Viability of Future Actions: Robust Reinforcement Learning via Entropy Regularization

Pierre-François Massiani[*1] **Alexander von Rohr**[*1,2] **Lukas Haverbeck**[1]
Sebastian Trimpe[1]

[1] Institute for Data Science in Mechanical Engineering, RWTH Aachen University, Germany
[2] Learning Systems and Robotics Lab, Technical University of Munich, Germany
{massiani,lukas.haverbeck,trimpe}@dsme.rwth-aachen.de
alex.von.rohr@tum.de

Abstract

Despite the many recent advances in reinforcement learning (RL), the question of learning policies that robustly satisfy state constraints under disturbances remains open. This paper reveals how robustness arises naturally by combining two common practices in unconstrained RL: entropy regularization and constraints penalization. Our results provide a method to learn robust policies, model-free and with standard popular algorithms. We begin by showing how entropy regularization biases the constrained RL problem towards maximizing the number of future viable actions, which is a form of robustness. Then, we relax the safety constraints via penalties to obtain an unconstrained RL problem, which we show approximates its constrained counterpart arbitrarily closely. We support our findings with illustrative examples and on popular RL benchmarks.

1 Introduction

Safety and, more generally, viability [1], is the ability of a controller to keep the system away from a set of state constraints. Robustness extends the notion to an adversarial or noisy setting; a robust agent remains outside of the constraints in spite of the noise or adversary. It is then meaningful to ask how to encode these desirable properties into a reward or cost function for reinforcement learning (RL) or optimal control in a practical way. While this question is well-studied for viability [2, 3], it is much less for robustness. The purpose of this work is to reveal how robustness arises naturally from two common practices in RL; namely, entropy regularization [4] and constraints penalization [2]. Our results provide additional theoretical and empirical support for the notion that entropy regularization can effectively enhance robustness against disturbances and model uncertainties [5–7].

Incorporating robustness in the optimization objective begs the question of the precise definition of robustness considered. For instance, robustness in the sense of Hamilton-Jacobi reachability is *worst-case*; the goal is to satisfy the constraints despite adversarial, bounded disturbances [8]. The idea is appealing but comes at the price of specialized algorithms [9], often model-based and computationally expensive [8]. An alternative viewpoint is to consider the quantification of the robustness of a state as the number of viable actions available in that state, as proposed in [10, 11]. In this work, we extend this line of thought from states to policies by quantifying their robustness based on the long-term number of viable actions they consider. The cumulative discounted entropy of a stochastic policy directly expresses this metric as an on-policy property; the higher the cumulative entropy of a viable policy is, the more viable actions it considers along its trajectories; therefore,

[*]Equal contribution

17th European Workshop on Reinforcement Learning (EWRL 2024).

it is robust in this sense. We call this notion S-robustness and encode it in the optimization target via a dual formulation, which yields precisely the entropy-regularized RL problem in a modified environment where only viable actions are available. While the connection of S-robustness with standard, control-theoretic notions remains an open question, we show on illustrative examples that it is a meaningful definition as the mode of the resulting policy keeps the system further away from constraints as the Lagrange multiplier grows.

Making this definition of robustness relevant to state-of-the-art practices in RL then requires relating the unconstrained and state-constrained optimization problems, as most applications focus on the former [3]. This is the role of *constraints penalization* — or, more concisely, of penalties. In the absence of entropy regularization, they are known to make constraints violations suboptimal, and large enough penalties guarantee viable optimal controllers [2]. We add entropy regularization to this analysis to show that the combination of entropy regularization with sufficiently high penalties yields (approximately) robustly viable policy, with a degree of robustness controlled by the temperature parameter. Furthermore, the mode of the resulting policy is viable and robust.

Our results emphasize a benefit of entropy regularization that differs from what is commonly mentioned in the literature. Indeed, algorithms such as soft actor-critic (SAC) [4] are often praised for their excellent exploration and their robustness to the choice of hyperparameters [4]. Although crucial in practice, these strengths are relevant *during* learning. In contrast, we focus on the learned policy, that is, what occurs *after* learning is completed (assuming a successful optimization). While the guarantees we derive are theoretical tools, they enable an understanding of the optimal behavior and its properties that result from the problem setup. We argue that such an understanding is essential for a successful parameterization of learning problems. For instance, if robustness is a concern for the downstream application, knowing how different parameters influence it is valuable.

We frame our results in great generality to maximize their applicability. Specifically, we focus on *viability*: the agent should avoid visiting a set of constraints. In contrast with safety, visiting that set may or may not result in the termination of the episode; we allow for both settings. Furthermore, we allow penalizing states that are not yet constraints violations but inevitably lead there because of the dynamics; such states are called *unviable*. This setup is strictly more general than that of [2], which only covers the case of penalizing immediate constraints violations.

Contributions: We reveal how robust optimal controllers arise from the combination of entropy regularization together with sufficient constraints penalization, in theory and practice. Specifically:

- We introduce S-robustness, a new notion of robustness measuring the ability of a controller to preserve the number of viable actions, together with an ordering of safe controllers to compare their degree of S-robustness;
- We explain theoretically how entropy regularization of the viability-constrained RL problem promotes S-robustness (Theorem 6 and Corollary 7);
- We show that introducing sufficient penalization in the entropy-regularized, unconstrained RL problem solves its constrained counterpart arbitrarily closely, providing a model-free way to obtain robust and viable controllers (Theorem 8);
- We illustrate our theoretical findings on both illustrative examples and RL benchmarks, showcasing the relationship between the penalty, the temperature, and robustness.

Overall, our findings facilitate the choice of hyperparameters in applications where robustness is important. We discuss other approaches to robustness in RL in Section 2. We then expose necessary preliminaries in Section 3, and formalize S-robustness and the problem we consider in Section 4. Our theoretical findings are in Section 5, and our empirical results in Section 6.

2 Related work

Viability and safety in RL: There is a wide variety of definitions of safety in RL [3]. We consider the case of avoiding state constraints with certainty (level 3 in [3]). Such a definition of safety falls into the general problem of viability[2] [1]. Many specialized algorithms were developed to solve this safe RL problem, both model-free and model-based [12]. It has been shown in [2] that sufficient

[2]Arguably, the main difference is terminological and both concepts are formally equivalent. However, viability violations need not lead to catastrophic outcomes.

failure penalties enforce equivalence between the unconstrained and safety-constrained problems, making safe RL provably amenable to unconstrained algorithms. The difference with previous similar results [13] is that [2] concludes on *safety of the controllers*, which is stronger than strong duality since the latter only informs on the optimization objective value. Regardless, the above works only guarantee safety and neglect robustness, with the notable exception of [9], which implements ideas from Hamilton-Jacobi reachability analysis. We extend the analysis and proof methods of [2] to entropy-regularized RL, which naturally yields robustness in addition to safety.

Robustness in optimal control: Robustness is a well-studied topic in (optimal) control [14], and consists of preserving viability despite *model uncertainties*. Classical general approaches therein consist of robust model predictive control [15, 16] and Hamilton-Jacobi reachability analysis [8]. They provide *worst-case* guarantees, and recent methods such as scenario optimization [17] were developed to address quantitative uncertainty. While S-robustness fits in neither category yet, we expect that the *mode* of S-robust policies enjoys worst-case guarantees.

Robustness in RL: Achieving robustness for RL policies is an active area of research [18]. A common formalization is that of a two-player game between the agent and an adversary [19, 20]. This setup is akin to Hamilton-Jacobi reachability analysis, only with a discounted cost. These approaches allow achieving robustness through an adversary controlling, for instance, disturbances [20] or action noise [21], yielding worst-case robustness. However, such adversarially robust RL requires specialized algorithms and training the adversary. In contrast, entropy-regularized RL is a popular framework with many standard implementations, which, as we show, also yields robustness.

We are not the first to report that entropy-regularization leads to robustness. Some empirical [5, 6] and theoretical works [7] highlight the inherent robustness of entropy-regularized RL. Eysenbach and Levine [7] consider robustness to changes in the dynamic and reward transformations, whereas we introduce a new robustness notion for learning under constraints based on entropy and viability.

3 Preliminaries

We introduce in this section the concepts relevant to frame the optimization problems and their constraints. In particular, we address entropy-regularized optimal control and viability.

3.1 Entropy-regularized optimal control

We consider finite sets $\mathcal{X}$ and $\mathcal{A}$ called the state and action spaces, respectively, and deterministic dynamics $f : \mathcal{Q} \to \mathcal{X}$, where $\mathcal{Q} = \mathcal{X} \times \mathcal{A}$ is the state-action space. Then, a policy $\pi : \mathcal{Q} \to [0, 1]$ is a map whose partial evaluation in any $x \in \mathcal{X}$ is a probability mass function on $\mathcal{A}$. We write $\pi(\cdot \mid x)$ and denote the set of all policies by Π. The state at time $t \in \mathbb{N}$ from initial state $x \in \mathcal{X}$ and following the policy $\pi \in \Pi$ is the random variable $X(t; x, \pi)$, and the action taken by π at that time is $A(t; x, \pi)$. If the policy and initial state are clear from context, we simply write X_t and A_t.

We also consider $r : \mathcal{Q} \to \mathbb{R}$ a reward function, which is necessarily bounded since $\mathcal{Q}$ is finite. The return of a policy $\pi \in \Pi$ from initial state $x \in \mathcal{X}$ is then

$$G(x, \pi) = \sum_{t=0}^{\infty} \gamma^t r(X_t, A_t), \tag{1}$$

where $\gamma \in [0, 1)$ is the discount factor. It quantifies the problem-specific time-horizon: a smaller γ disregards delayed rewards, even though this effect can be overcome if the said rewards have large magnitude. The expected return is then $\bar{G}(x, \pi) = \mathbb{E}[G(x, \pi)]$. With $\mathcal{H}$ as the entropy, we introduce the discounted total entropy of a policy $\pi \in \Pi$ from the state $x \in \mathcal{X}$ as

$$S(x, \pi) = \sum_{t=0}^{\infty} \gamma^t \mathcal{H}(\pi(\cdot \mid X_t)), \tag{2}$$

and its expectation $\bar{S}(x, \pi) = \mathbb{E}[S(x, \pi)]$. The objective of entropy-regularized optimal control is then to find an optimal policy, that is, a policy $\pi_{\mathrm{opt}} \in \Pi$ such that

$$\pi_{\mathrm{opt}} \in \arg\max_{\pi \in \Pi} \bar{G}(x, \pi) + \alpha \bar{S}(x, \pi), \quad \forall x \in \mathcal{X}, \tag{3}$$

where $\alpha \in \mathbb{R}_{\geq 0}$ is a design parameter called the *temperature*. It is known that there exists an optimal policy [4]. Specifically, one can be computed by leveraging the optimal soft-Q-value function $q : \mathcal{Q} \to \mathbb{R}$, which is the only fixed point of the optimal soft-Bellman operator [22]:

$$q(x,a) = r(x,a) + \gamma\alpha \ln \left[\sum_{b \in \mathcal{A}} \exp\left(\frac{1}{\alpha} q(x',b) \right) \right], \quad \forall (x,a) \in \mathcal{Q}, \tag{4}$$

where we defined the shorthand $x' = f(x,a)$. An equivalent definition is [22, Theorem 16]

$$q(x,a) = \max_{\pi \in \Pi} r(x,a) + \gamma \bar{G}(x',\pi) + \alpha\gamma \bar{S}(x',\pi). \tag{5}$$

Once q is known, the following policy solves (3):

$$\pi_{\text{opt}}(a \mid x) = \text{softmax}\left[\frac{1}{\alpha} Q(x,\cdot) \right](a) = \frac{\exp\left[\frac{1}{\alpha} Q(x,a) \right]}{\sum_{b \in \mathcal{A}} \exp\left[\frac{1}{\alpha} Q(x,b) \right]}, \quad \forall (x,a) \in \mathcal{Q}. \tag{6}$$

3.2 Viability

We consider a set of states $\mathcal{X}_{\text{C}} \subset \mathcal{X}$ that the system should never visit. This constraint may encode safety concerns, for instance, but it can generally contain any states that are deemed undesirable for the problem at hand. Avoiding $\mathcal{X}_{\text{C}}$ is a dynamic concern, and some states that are not in $\mathcal{X}_{\text{C}}$ themselves may still lead there inevitably. We address this through viability theory [1, Chapter 2].

Definition 1 (Viability kernel)**.** The viability kernel $\mathcal{X}_{\text{V}}$ is the set of states from where $\mathcal{X}_{\text{C}}$ can be avoided at all times almost surely:

$$\mathcal{X}_{\text{V}} = \{ x \in \mathcal{X} \mid \exists \pi \in \Pi, \forall t \in \mathbb{N}_{>0}, \mathbb{P}[X_t \notin \mathcal{X}_{\text{C}}] = 1 \}. \tag{7}$$

By definition, any state that is not in the viability kernel leads to $\mathcal{X}_{\text{C}}$ in finite time. Such states are called *unviable*. The viability kernel is therefore the largest set that enables recursive feasibility of the problem of avoiding transitions into $\mathcal{X}_{\text{C}}$. A closely related concept is the *viable set*, which is the set of state-action pairs that preserve viability [10]:

$$\mathcal{Q}_{\text{V}} = \{ (x,a) \in \mathcal{Q} \mid x \in \mathcal{X}_{\text{V}} \wedge f(x,a) \in \mathcal{X}_{\text{V}} \}. \tag{8}$$

We also define the unviable set $\mathcal{Q}_{\text{U}} = \mathcal{Q} \setminus \mathcal{Q}_{\text{V}}$, and the critical set $\mathcal{Q}_{\text{crit}} = \mathcal{Q}_{\text{U}} \cap (\mathcal{X}_{\text{V}} \times \mathcal{A})$ [23].

Definition 2. Let $\pi \in \Pi$. We say that π is *viable from the state* $x \in \mathcal{X}$ if $\mathbb{P}[X_t \notin \mathcal{X}_{\text{C}}] = 1$ for all $t \in \mathbb{N}_{>0}$. We say that π is *viable* if it is viable from any $x \in \mathcal{X}_{\text{V}}$. Furthermore, for any $\delta > 0$, we say that π is δ-*viable* if $\max_{\mathcal{Q}_{\text{crit}}} \pi \leq \delta$. We denote the set of policies viable from the state x by $\Pi_{\text{V}}(x)$ and that of viable policies by Π_{V}.

By definition of the viability kernel, the condition for a viable policy can be replaced with $\mathbb{P}[X_t \in \mathcal{X}_{\text{V}}] = 1$ for all $t \in \mathbb{N}$. In the next section, we consider an optimal control problem over the set of safe policies and dual relaxations thereof. For this, we introduce dynamic indicators.

Definition 3 (Dynamic indicator)**.** Let $c : \mathcal{Q} \to \mathbb{R}_{\geq 0}$ and the associated discounted risk

$$\rho(x,\pi) = \sum_{t=0}^{\infty} \gamma^t c(X_t, A_t). \tag{9}$$

We say that c is a dynamic indicator of $\mathcal{X}_{\text{C}}$ if, for all $x \in \mathcal{X}_{\text{V}}$, $\mathbb{E}[\rho(x,\pi)] > 0$ if, and only if, $\pi \notin \Pi_{\text{V}}(x)$.

The notion is independent of $\gamma \in (0,1)$. For instance, the composition of the indicator function of $\mathcal{X}_{\text{C}}$ with the dynamics is a dynamic indicator of $\mathcal{X}_{\text{C}}$ [2, Lemma 1].

Remark 4 (Recovering from constraints violation)**.** Our results hold in the two settings where visiting $\mathcal{X}_{\text{C}}$ terminates the episode or not. The second case is fully consistent with the setup of infinite time-horizon optimal control that precedes. Then, actions taken from $\mathcal{X}_{\text{C}}$ may map back into $\mathcal{X}_{\text{V}}$: trajectories leaving $\mathcal{X}_{\text{V}}$ may only return there *after* visiting $\mathcal{X}_{\text{C}}$. We even have $\mathcal{X}_{\text{C}} \cap \mathcal{X}_{\text{V}} \neq \emptyset$ in general, and the intersection is composed of states with actions that map in $\mathcal{X}_{\text{V}} \setminus \mathcal{X}_{\text{C}}$. The first case, however, is not naturally framed in infinite time-horizon. Indeed, while adding an absorbing state with null reward and dynamic indicator as in [2] effectively cuts the sums in $G(x,\pi)$ and $\rho(x,\pi)$, the sum in $S(x,\pi)$ cannot be handled similarly without additional notation. In the interest of conciseness and clarity, we thus only introduce formally the case of non-terminal $\mathcal{X}_{\text{C}}$.

4 Problem formulation

We consider a standard reinforcement learning problem with dynamics f, constraint set $\mathcal{X}_{\mathrm{C}}$, viability kernel $\mathcal{X}_{\mathrm{V}}$, and return G as defined in Section 3. We aim to learn policies that find a trade-off between robustness and return maximization. To arrive at such a trade-off, we start with the minimal requirement that policies should be viable, meaning that the maximum objective we can achieve is

$$\max_{\pi \in \Pi_{\mathrm{V}}} \bar{G}(x, \pi). \tag{10}$$

This represents the solution with no explicit robustness consideration. To introduce robustness starting from here, we ask the following:

Problem 1. How do we bias the solution of (10) towards robustness?

Problem 2. How do we solve the resulting constrained and biased problem with classical algorithms?

We address Problem 1 in Section 5.2 by weighting G in the objective of (10) with a quantification of a meaningful notion of robustness and Problem 2 in Section 5.3 via a Lagrangian relaxation of the constraint $\pi \in \Pi_{\mathrm{V}}$, hereby approximately solving the biased modification of (10) arbitrarily closely.

5 Theoretical results

In this section, we present our definition of S-robustness and derive S-robustness properties of entropy-regularized RL problems, first with constraints, and then without. The proofs are in Appendix B.

5.1 Robustness as preserving future viable options

As motivated in the introduction, we extend the line of thought of Heim et al. [11], which defines the "natural robustness of a state $x \in \mathcal{X}_{\mathrm{V}}$" as the number of viable actions available in that state, that is, the cardinality of $\mathcal{Q}_{\mathrm{V}}[x]$. They call this quantity the safety measure.

To generalize this notion to policies, an idea is to consider (a transformation of) the safety measure of every state visited by the policy, and to aggregate the result across time and realizations. By choosing the aggregations as the infimum and expectation, respectively, this defines $\mathbb{E}[\inf_t \mathcal{H}(u_V(\cdot \mid X_t))]$ as the robustness of a policy $\pi \in \Pi_{\mathrm{V}}$ from a state $x \in \mathcal{X}_{\mathrm{V}}$, where $u_V(\cdot \mid x)$ is the uniform distribution with support $\mathcal{Q}_{\mathrm{V}}[x]$. In other words, it is the average of the minimum value of (a transformation of) the safety measure across trajectories.

Unfortunately, this definition has the problem of being *off-policy*, which encourages, for instance, visiting states where most viable actions lead to only having only few viable options remaining, as long as at least one path guarantees many choices. A compelling example illustrating why a policy with a high such metric should not be called robust is in Appendix A. This motivates an *on-policy* notion by using, in each state, *the number of viable actions the policy considers* instead of the safety measure, which yields the metric $\mathbb{E}[\inf_t \mathcal{H}(\pi(\cdot \mid X_t))]$ for $\pi \in \Pi_{\mathrm{V}}$. Indeed, since π is viable by assumption, its entropy in each state measures the number of viable actions it considers there. A policy maximizing this quantity is forced into a trade-off between its current entropy and avoiding states that lead to only few viable options, and it assigns higher probabilities to states that preserve the number of those future viable options. In other words, the on-policy nature of the metric forces policies to commit to the actions that contribute to it. It is thus intuitive that the *mode* of such a policy achieves a meaningful form of robustness. In what follows, we use a discounted sum rather than an infimum over time for compatibility with RL.

Definition 5. We say that $\pi_1 \in \Pi_{\mathrm{V}}$ is *less S-robust* than $\pi_2 \in \Pi_{\mathrm{V}}$, and write $\pi_1 \preceq \pi_2$, if

$$\bar{S}(x, \pi_1) \leq \bar{S}(x, \pi_2), \quad \forall x \in \mathcal{X}_{\mathrm{V}}. \tag{11}$$

5.2 Biasing for robustness

With this definition, the maximum entropy policy is the most robust one, and is the maximum element for the partial order $\preceq$:

$$\pi^{\star}_{\mathrm{ent}} = \arg\max_{\pi \in \Pi_{\mathrm{V}}} \bar{S}(x, \pi), \quad \forall x \in \mathcal{X}_{\mathrm{V}}. \tag{12}$$

The trade-off between optimality and robustness is achieved by the relative weighting of both terms:

$$\pi^\star_\alpha = \arg\max_{\pi\in\Pi_{\mathrm{V}}} \bar{G}(x,\pi) + \alpha\bar{S}(x,\pi), \quad \forall x \in \mathcal{X}_{\mathrm{V}}. \tag{13}$$

This is precisely the entropy-regularized RL problem in the modified state-action space $\mathcal{Q}_{\mathrm{V}}$ with temperature $\alpha \in \mathbb{R}_{\geq 0}$.

5.2.1 Behavior for increasing temperatures

For $\alpha = 0$, (13) recovers (10); we are maximizing the return over viable policies with no concerns about robustness. As α increases, entropy is more and more prevalent in the objective of (13), whose solution converges to $\pi^\star_{\mathrm{ent}}$. This is best seen through the soft-value function.

Theorem 6. Consider the soft-Q-value functions Q_{ent} and Q_α of (12) and (13), respectively and for all $\alpha \in \mathbb{R}_{\geq 0}$. Then, $\max_{\mathcal{Q}_{\mathrm{V}}} |\frac{1}{\alpha}Q_\alpha - Q_{\mathrm{ent}}| \to 0$ as $\alpha \to \infty$.

Corollary 7. Denote by $\pi^\star_{\mathrm{ent}}$ and $\pi^\star_\alpha$ the solutions of (12) and (13), respectively and for all $\alpha \in \mathbb{R}_{\geq 0}$. Then, the map $\alpha \mapsto \pi^\star_\alpha$ is monotonic for $\preceq$ and $\max_{\mathcal{Q}_{\mathrm{V}}} |\pi^\star_\alpha - \pi^\star_{\mathrm{ent}}| \to 0$ as $\alpha \to \infty$.

In particular, Corollary 7 implies that the solution of (13) gets more and more robust as α increases.

5.3 Relaxing safety constraints with penalties

A drawback of (13) that hinders its practicality is that it explicitly involves both the viability kernel $\mathcal{X}_{\mathrm{V}}$ and the viable set $\mathcal{Q}_{\mathrm{V}}$, which are unknown in model-free situations. We now leverage a Lagrangian relaxation of these viability constraints to make the problem amenable to model-free algorithms. The results in this section extend those of [2] to the case of an objective involving entropy regularization.

In this section, we consider c, a dynamic indicator function of $\mathcal{X}_{\mathrm{C}}$, and ρ, the associated discounted risk (Definition 3). We are interested in the following penalized problem

$$\pi^\star_{\alpha,p} = \arg\max_{\pi\in\Pi} \bar{G}(x,\pi) + \alpha\bar{S}(x,\pi) - p\rho(x,\pi), \tag{14}$$

where $p \in \mathbb{R}_{\geq 0}$ is a penalty parameter. It is known that in the case $\alpha = 0$, (14) and (10) share the same solutions if p is large enough [2, Theorem 2]. Unfortunately, this result does not directly carry to the case $\alpha > 0$: from (6), $\pi^\star_{\alpha,p}(a \mid x) > 0$ for all $(x,a) \in \mathcal{Q}$, and thus in particular $\pi^\star_{\alpha,p} \notin \Pi_{\mathrm{V}}$. However, scaling the penalty remains possible if one accepts to trade viability for δ-viability.

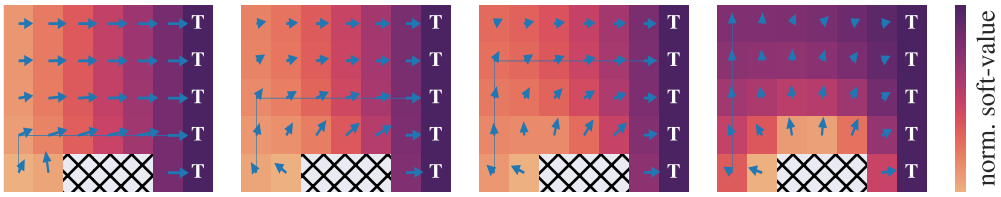
Theorem 8. For any $\delta > 0$, $\epsilon > 0$, and $\alpha > 0$, there exists $p^\star \in \mathbb{R}_{\geq 0}$ such that, for all $p > p^\star$, the optimal policy of (14) $\pi^\star_{\alpha,p}$ is δ-viable and

$$\max_{\mathcal{Q}_{\mathrm{V}}} |\pi^\star_{\alpha,p} - \pi^\star_\alpha| < \epsilon. \tag{15}$$

Sketch of proof. The penalty enforces an upper-bound on the soft Q-value of state-actions in $\mathcal{Q}_{\mathrm{crit}}$ (Lemma 11). Values there thus decrease arbitrarily low as the penalty increases, while it remains lower-bounded on $\mathcal{Q}_{\mathrm{V}}$. This, in turn, shows δ-safety of $\pi^\star_{\alpha,p}$ for p large enough. Therefore, the value function of (14) approximates to that of (10) on $\mathcal{Q}_{\mathrm{V}}$, and $\pi^\star_{\alpha,p}$ gets arbitrarily close to $\pi^\star_\alpha$. □

In practice, the weakening to δ-viability is not problematic, as one is particularly concerned about the *mode* of the policy learned via entropy regularization, which is safe if δ is low enough in Theorem 8. One of the purposes of the next section is to demonstrate empirically the robustness of this mode.

Conclusion on the problems Finally, we are able to answer Problems 1 and 2 based on the following arguments. Entropy regularization in the presence of constraints biases the learning problem to favor S-robustness, with the temperature coefficient monotonically controlling the degree of S-robustness. The viability constraints of (13) can be relaxed by a Lagrangian formulation at the price of trading viability for δ-viability. Specifically, the solution of (14) approximates arbitrarily closely that of (13), provided that the penalty is sufficiently high. Put together, these results provide a model-free way to approximate robust controllers with tunable degrees of robustness.



(a) $\alpha = 1$ (b) $\alpha = 4$ (c) $\alpha = 8$ (d) Max. ent.

Figure 1: **Fenced cliff — Robustness as a function of α:** An entropy regularized policy avoids states with unviable actions (d). The degree is controlled by the temperature parameter α. As it increases (a–c), the policy moves away from the constraints, getting more robust but taking longer to reach the target. The mode of the policy is shown as a thin blue line.

6 Empirical results

We demonstrate the robustness of entropy-regularized, penalized policies in two ways. We start with a discrete gridworld, which reveals, through simple dynamics, the effects and interactions of the temperature and penalty. Second, we evaluate entropy-regularized neural network policies on RL benchmarks. We go beyond our theoretical results by evaluating the robustness to external disturbances to complement previous observations that show SAC leads to robust policies [5, 6].

6.1 Cliff walking

Our gridworld is an adaptation of the cliff environment [24, Example 6.6]. Three states in the middle of the bottom row represent the cliff; the constraint set $\mathcal{X}_\mathrm{C}$ the agent should robustly avoid. The right column represents the target of escaping the cliff. Both the cliff and target states are invariant under all actions. Otherwise, the dynamics follow the direction of the chosen action and map back into the current state if the agent hits a border. Actions outside of the cliff and target get a -1 reward.

6.1.1 Constraints and entropy enforce a robustness–performance trade-off

The constrained version of the environment — the fenced cliff — only offers three actions to an agent neighboring the cliff, imposing a lower achievable entropy in those states. This observation is key in understanding why entropy regularization avoids them, yielding robustness (Fig. 1.d).

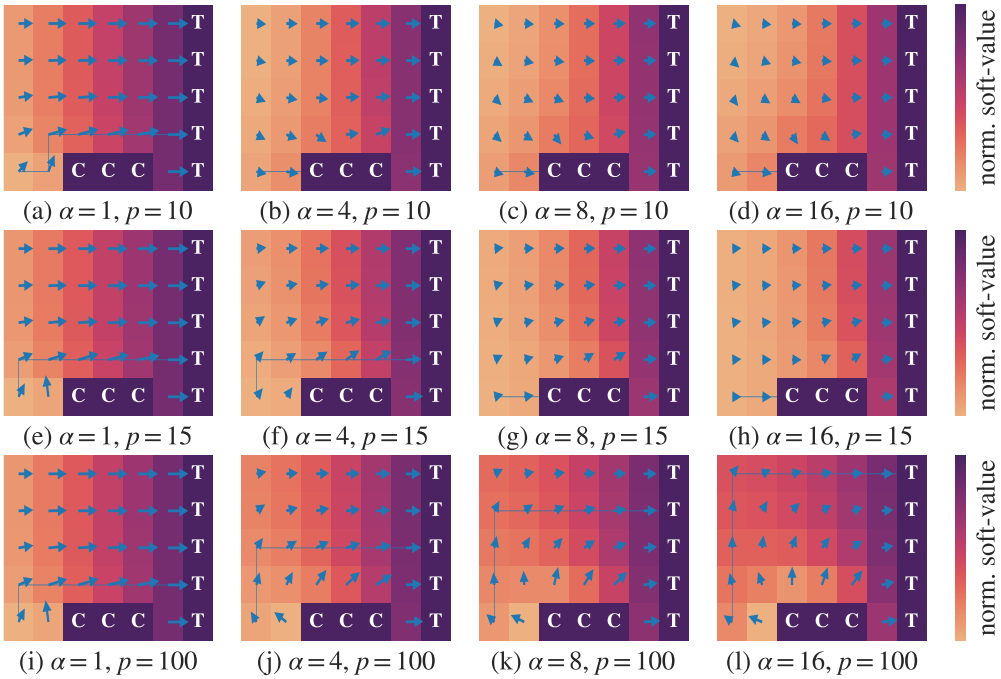
Indeed, in the absence of any reward, the maximum entropy policy favors transitioning away from states neighboring the constraints due to the aforementioned upper bound on immediate entropy. In turn, the immediate entropy of the policy in the 2-step neighbors is also reduced since some transitions are less desirable. The same logic applies recursively "outwards" from states with unviable actions, and the policy prefers the top corners as these are as far as possible from the constraint (Fig. 1.d) The trade-off between short- and long-term entropy depends on the discount factor γ.

In contrast, adding a nonzero reward (Fig. 5.a–c) finds a trade-off between entropy — that is, moving randomly — and reaching the goal state to avoid the negative reward. The agent thus takes more risks to collect rewards while preserving some distance from the constraints. This trade-off between performance and robustness is controlled by the temperature parameter α: High values favor entropy (and, thus, robustness by what precedes), whereas lower ones favor performance.

While high robustness may be desirable, it comes at the price of suboptimality. Too high a temperature may entirely prevent task completion if the path to it is inherently risky, leading to unsuccessful learning outcomes due to poor choice of hyperparameters.

6.1.2 Unconstrained cliff walking

Sufficient constraints penalization enables solving the constrained problem (Fig. 2), consistently with Theorem 8. The example shows the robustness–performance trade-off with different temperatures and penalties. Importantly, entropy and penalties are now competing, and insufficient penalties get overcome by high temperatures, degrading safety. The penalty thus needs to scale with the temperature to ensure δ-safety with a low δ, as shown in Appendix C.



(a) $\alpha = 1, p = 10$ (b) $\alpha = 4, p = 10$ (c) $\alpha = 8, p = 10$ (d) $\alpha = 16, p = 10$

(e) $\alpha = 1, p = 15$ (f) $\alpha = 4, p = 15$ (g) $\alpha = 8, p = 15$ (h) $\alpha = 16, p = 15$

(i) $\alpha = 1, p = 100$ (j) $\alpha = 4, p = 100$ (k) $\alpha = 8, p = 100$ (l) $\alpha = 16, p = 100$

Figure 2: **Unconstrained cliff — Safety and robustness as functions of α and p:** Safety and robustness can be achieved by penalizing (p) the constraints $\mathcal{X}_C$ and adjusting the temperature (α).

6.2 Reinforcement learning benchmarks

This section demonstrates the performance robustness trade-off in entropy-regularized RL on two popular benchmarks. For this, we train a neural network policy using the SAC implementation of Huang et al. [25] with different fixed temperature values (no automatic tuning). The hyperparameters are reported in Appendix E. For each α, we train on 25 random seeds and evaluate the trained policy in terms of return and robustness.

To evaluate robustness, we take the mode of each trained policy and add a disturbance sampled from a uniform distribution $\mathcal{U}(-\epsilon, \epsilon)$ to the action at each environment step. An episode is successful if the agent obeys the constraint despite the input disturbance. We test each trained policy for 100 episodes and report the success rate. Consistent with our theoretical results, we find (i) entropy-regularization decreases the return by avoiding high-value states with many unviable actions; (ii) the mode of entropy-regularized policies is more robust to disturbances as the training temperature increases.

6.2.1 Pendulum

We modify the `Pendulum-v1` environment [26] as follows to make it suitable for a study on robustness: (i) the initial state is the still upright position; (ii) the constraints consist of angles with magnitude beyond 90° and the penalty is 90; and (iii) the reward is the quadratic distance to a target angle of 45°, which is outside of the viability kernel since the agent exerts bounded torque. See Appendix D.1 for further details.

The results are shown in Fig. 3. All policies lean towards the target state but avoid leaving the viability kernel and reaching the constraints. Sufficient penalties emulate the boundary of the viability kernel, which reduces the effective number of available actions when leaning to one side. This pushes entropy-regularized policies away from the target state, as can be seen in the lower return — the maximum entropy policy keeps the pendulum upright. The results show a robustness–performance trade-off between staying upright and leaning as far as possible towards the target, which is controlled by the temperature α. Furthermore, the mode of the entropy-regularized policy can cope with significantly higher disturbances when trained with higher temperatures.

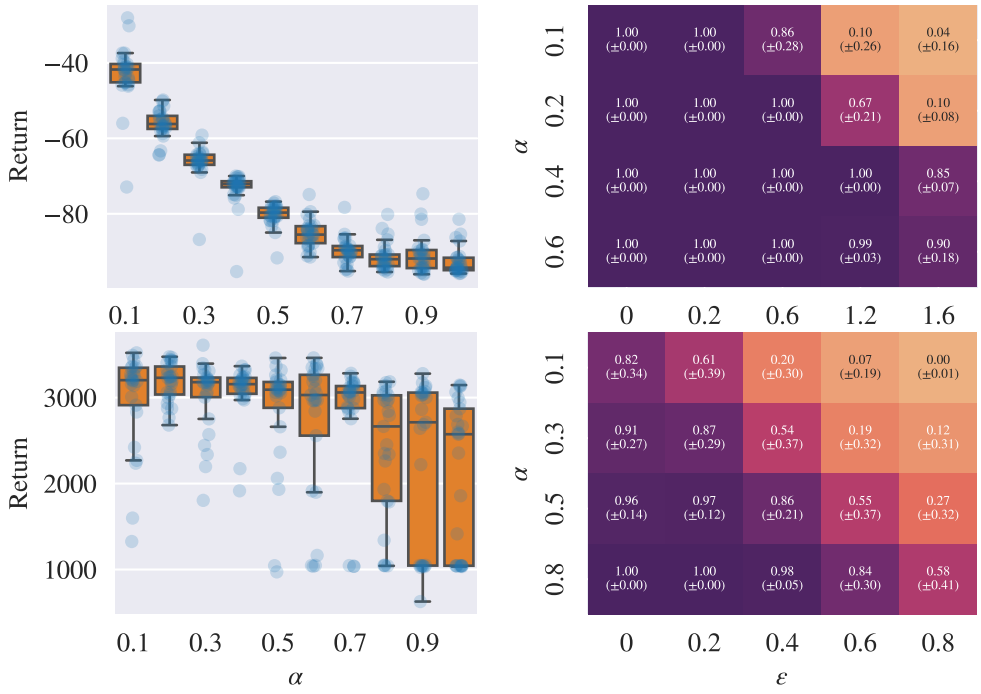


Figure 3: **Learning robust policies with SAC:** Evaluation on the `Pendulum-v1` (top row) and `Hopper-v2` (bottom row) environments. *Left:* As the temperature increases, the mode of the policy trades return for robustness and gets more conservative. *Right:* The mode of the stochastic policy is more robust to disturbances as the temperature increases.

6.2.2 Hopper

We repeat the same experiment as in the previous section for a modified `Hopper-v2` environment [26]. We modify the environment by penalizing the "unhealthy" states with a penalty of $p = 300$. The results are shown in Fig. 3. The gait becomes slower but more robust to disturbances as the training temperature increases. Interestingly, as the temperature is increased, the training finds two distinct robust behaviors. One is the intended hopping forward; the other is standing still and only collecting the healthy reward.

7 Limitations

In this section, we point towards some of the limitations of our analysis. First, and maybe most importantly, we do not yet establish a formal connection between S-robustness and classical robustness in optimal control. The main difficulty is that entropy-regularized policies have full support and explore every state with positive probability, while most classical robustness notions are about surely preserving a margin to unsafe states, represented by the bound on the allowed disturbances. A promising approach for future research is thus to focus on the mode of the stochastic entropy policy. Indeed, our results already empirically demonstrate that it can achieve robustness to disturbances.

A further limitation is that we restrict our analysis to optimal value functions. While this is necessary to understand the learning problem, we inherit practical issues that come with solving (14) with specific RL algorithms, such as local minima and function approximation errors. We have encountered this problem already in the evaluation of the hopper. A direction for future research is ensuring that the learning outcome has the desired robustness properties using, for example, formal certification [27].

8 Conclusion

We discuss the properties of the entropy-regularized solution to optimal control problems with constraints or penalties. We argue that the entropy of a viable policy measures a form of robustness

and show that the combination of entropy regularization and constraints penalization is sufficient to learn robustly viable policies in model-free RL.

We expect our findings to inform practitioners when applying RL algorithms such as SAC. While entropy regularization has mainly been developed as an exploration mechanism [4], it biases the policy to robustly move away from states with low rewards, which means remaining viable if one uses constraints penalties. If such a behavior is desirable for the given application, our analysis suggests avoiding a common practice of annealing the temperature α during learning. On the other hand, high temperatures or minimum entropy constraints [28] can make parts of the state space unreachable, lead to conservative policies, and may even entirely prevent task completion. Overall, our results enable principled decisions regarding this crucial parameter.

Acknowledgments and Disclosure of Funding

The authors thank Zeheng Gong for help with the empirical results. Computations were performed with computing resources granted by RWTH Aachen University under project rwth1626.

References

[1] Jean-Pierre Aubin, Alexandre M Bayen, and Patrick Saint-Pierre. *Viability theory: new directions*. Springer Science & Business Media, 2011.

[2] Pierre-François Massiani, Steve Heim, Friedrich Solowjow, and Sebastian Trimpe. Safe Value Functions. *IEEE Transactions on Automatic Control*, 2023.

[3] Lukas Brunke, Melissa Greeff, Adam W Hall, Zhaocong Yuan, Siqi Zhou, Jacopo Panerati, and Angela P Schoellig. Safe learning in robotics: From learning-based control to safe reinforcement learning. *Annual Review of Control, Robotics, and Autonomous Systems*, 5:411–444, 2022.

[4] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, pages 1861–1870. PMLR, 2018.

[5] Tuomas Haarnoja, Vitchyr Pong, Aurick Zhou, Murtaza Dalal, Pieter Abbeel, and Sergey Levine. Composable deep reinforcement learning for robotic manipulation. In *IEEE International Conference on Robotics and Automation*, pages 6244–6251, 2018. doi: 10.1109/ICRA.2018.8460756.

[6] Tuomas Haarnoja, Sehoon Ha, Aurick Zhou, Jie Tan, George Tucker, and Sergey Levine. Learning to walk via deep reinforcement learning. In *Proceedings of Robotics: Science and Systems*, 2019. doi: 10.15607/RSS.2019.XV.011.

[7] Benjamin Eysenbach and Sergey Levine. Maximum entropy RL (provably) solves some robust RL problems. In *International Conference on Learning Representations*, 2022.

[8] Somil Bansal, Mo Chen, Sylvia Herbert, and Claire J Tomlin. Hamilton-jacobi reachability: A brief overview and recent advances. In *Conference on Decision and Control*, pages 2242–2253, 2017.

[9] Kai-Chieh Hsu, Vicenç Rubies-Royo, Claire J Tomlin, and Jaime F Fisac. Safety and liveness guarantees through reach-avoid reinforcement learning. *arXiv preprint arXiv:2112.12288*, 2021.

[10] Steve Heim and Alexander Badri-Spröwitz. Beyond basins of attraction: Quantifying robustness of natural dynamics. *IEEE Transactions on Robotics*, 35(4):939–952, 2019.

[11] Steve Heim, Alexander Rohr, Sebastian Trimpe, and Alexander Badri-Spröwitz. A Learnable Safety Measure. In *Conference on Robot Learning*, pages 627–639. PMLR, May 2020.

[12] Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *International conference on machine learning*, pages 22–31. PMLR, 2017.

[13] Santiago Paternain, Luiz Chamon, Miguel Calvo-Fullana, and Alejandro Ribeiro. Constrained reinforcement learning has zero duality gap. *Advances in Neural Information Processing Systems*, 32, 2019.

[14] Kemin Zhou, John Doyle, and Keith Glover. *Robust and optimal control*. Prentice Hall, 1996.

[15] Lars Grüne and Jürgen Pannek. *Nonlinear Model Predictive Control*. Springer, 2 edition, 2017.

[16] Daniel Limon, Teodoro Alamo, Davide M Raimondo, D Muñoz De La Peña, José Manuel Bravo, Antonio Ferramosca, and Eduardo F Camacho. Input-to-state stability: a unifying framework for robust model predictive control. *Nonlinear Model Predictive Control: Towards New Challenging Applications*, pages 1–26, 2009.

[17] Giuseppe Carlo Calafiore and Marco C Campi. The scenario approach to robust control design. *IEEE Transactions on automatic control*, 51(5):742–753, 2006.

[18] Janosch Moos, Kay Hansel, Hany Abdulsamad, Svenja Stark, Debora Clever, and Jan Peters. Robust reinforcement learning: A review of foundations and recent advances. *Machine Learning and Knowledge Extraction*, 4(1):276–315, 2022. ISSN 2504-4990. doi: 10.3390/make4010013.

[19] Jun Morimoto and Kenji Doya. Robust reinforcement learning. *Neural Computation*, 17(2): 335–359, 2005. doi: 10.1162/0899766053011528.

[20] Lerrel Pinto, James Davidson, Rahul Sukthankar, and Abhinav Gupta. Robust adversarial reinforcement learning. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 2817–2826, 2017.

[21] Chen Tessler, Yonathan Efroni, and Shie Mannor. Action robust reinforcement learning and applications in continuous control. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pages 6215–6224, 2019.

[22] Ofir Nachum, Mohammad Norouzi, Kelvin Xu, and Dale Schuurmans. Bridging the gap between value and policy based reinforcement learning. *Advances in neural information processing systems*, 30, 2017.

[23] Pierre-François Massiani, Steve Heim, and Sebastian Trimpe. On exploration requirements for learning safety constraints. In *Learning for Dynamics and Control*, pages 905–916. PMLR, 2021.

[24] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[25] Shengyi Huang, Rousslan Fernand Julien Dossa, Chang Ye, Jeff Braga, Dipam Chakraborty, Kinal Mehta, and João G.M. Araújo. Cleanrl: High-quality single-file implementations of deep reinforcement learning algorithms. *Journal of Machine Learning Research*, 23(274):1–18, 2022. URL `http://jmlr.org/papers/v23/21-1342.html`.

[26] Mark Towers, Jordan K. Terry, Ariel Kwiatkowski, John U. Balis, Gianluca de Cola, Tristan Deleu, Manuel Goulão, Andreas Kallinteris, Arjun KG, Markus Krimmel, Rodrigo Perez-Vicente, Andrea Pierré, Sander Schulhoff, Jun Jet Tai, Andrew Tan Jin Shen, and Omar G. Younis. Gymnasium, March 2023. URL `https://zenodo.org/record/8127025`.

[27] Björn Lütjens, Michael Everett, and Jonathan P. How. Certified adversarial robustness for deep reinforcement learning. In Leslie Pack Kaelbling, Danica Kragic, and Komei Sugiura, editors, *Proceedings of the Conference on Robot Learning*, volume 100 of *Proceedings of Machine Learning Research*, pages 1328–1337, 2020.

[28] Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, Pieter Abbeel, et al. Soft actor-critic algorithms and applications. *arXiv preprint arXiv:1812.05905*, 2018.

A Counterexample for an off-policy metric of robustness

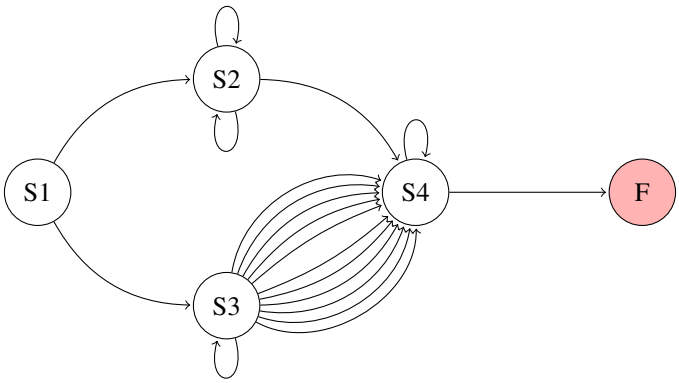


Figure 4: **Counterexample for an off-policy metric of robustness based on entropy.** The constraint set is $\mathcal{X}_{\mathrm{C}} = \{F\}$.

Figure 4 illustrates a counterexample demonstrating why the number of viable actions, expressed as $\mathbb{E}_\pi[\inf_t \mathcal{H}(u_V(\cdot \mid X_t)]$, does not quantify a meaningful notion of robustness.

Consider two policies: π_1, that goes from $S1$ to $S2$, and π_2, that goes from $S1$ to $S3$. Consider the case — which differs from Figure 4 — where there is only one action leading from $S3$ to $S4$. Then, arguably, a meaningful notion of robustness should say that π_1 is more robust than π_2; indeed, in the presence of action perturbations, both may end up in $S4$, but π_1 takes a path with more "chances" to avoid — or delay — going there. Adding additional actions from $S3$ to $S4$ does not change this reasoning, and π_1 should still be considered more robust than π_2 in the situation of Figure 4. Yet, the value of $\mathbb{E}_{\pi_2}[\inf_t \mathcal{H}(u_V(\cdot \mid X_t))]$ is higher than that of $\mathbb{E}_{\pi_1}[\inf_t \mathcal{H}(u_V(\cdot \mid X_t))]$, since the former is increased by the additional actions. Choosing this metric thus classifies π_2 as more robust than π_1, which is the opposite of the desired outcome.

The property of the above metric allowing this example is that the policy only needs to visit states with a high number of viable actions, without considering whether those actions preserve a high number of viable actions themselves, as long as at least one does. In other words, the policy does not need to commit to the viable actions that contribute to the value of its metric. This observation calls for an *on-policy* metric of robustness, which we achieve by replacing u_V in the previous proposition by the policy itself, yielding $\mathbb{E}[\inf_t \mathcal{H}(\pi(\cdot \mid X_t))]$. Then, there are policies that visit $S2$ with a metric higher than the value achieved by any policy that visits $S3$.

B Proofs

In all of this section, we identify real functions defined on $\mathcal{Q}_{\mathrm{V}}$ with vectors in $\mathbb{R}^{|\mathcal{Q}_{\mathrm{V}}|}$, and use operators defined on the functions on their vector representations indistinctly.

B.1 Proof of Theorem 6

Proof. For $\alpha \in \mathbb{R}_{>0}$, consider the operators $B^\star$ and $B^\star_\alpha$ defined on $\mathbb{R}^{\mathcal{Q}_{\mathrm{V}}}$ by, for all $h \in \mathbb{R}^{\mathcal{Q}_{\mathrm{V}}}$ and $(x, a) \in \mathcal{Q}_{\mathrm{V}}$,

$$(B^\star h)(x,a) = \gamma \ln \left[\sum_{b \in \mathcal{Q}_{\mathrm{V}}[x]} \exp[h(x', b)] \right],$$

$$(B^\star_\alpha h)(x,a) = \frac{1}{\alpha} r(x,a) + \gamma \ln \left[\sum_{b \in \mathcal{Q}_{\mathrm{V}}[x]} \exp[h(x', b)] \right],$$

where we introduced the shorthand $x' = f(x, a)$. The operator $B^\star$ is the optimal soft-Bellman operator associated with (12), and $B^\star_\alpha$ is that associated with (10), scaled by the factor $\frac{1}{\alpha}$. Let now

$R_\alpha = \frac{1}{\alpha} Q_\alpha$. By properties of the optimal Bellman operator, R_α and Q_{ent} are the unique fixed points of $B^\star_\alpha$ and $B^\star$, respectively.

Let us fix a strictly increasing sequence $(\alpha_n)_{n\in\mathbb{N}}$, and define for conciseness $R_{\alpha_n} = R_n$ and $B^\star_{\alpha_n} = B^\star_n$. The sequence $(R_n)_{n\in\mathbb{N}}$ is clearly bounded. Therefore, there exists $R \in \mathbb{R}^{|\mathcal{Q}_{\mathrm{V}}|}$ and $(m_n)_{n\in\mathbb{N}}$, strictly increasing, such that $R_{m_n} \to R$ as $n \to \infty$. We show $B^\star(R) = R$. For all $n \in \mathbb{N}$,

$$\begin{aligned}\|R - B^\star R\| &\leq \|R - R_{m_n}\| + \|B^\star_{m_n}(R_{m_n}) - B^\star(R_{m_n})\| + \|B^\star(R_{m_n}) - B^\star(R)\| \\ &\leq \|R - R_{m_n}\| + \frac{1}{\alpha_n}\|r\| + \|B^\star(R_{m_n}) - B^\star(R)\|,\end{aligned}$$

where we used the identity $B^\star_{m_n}(R_{m_n}) = R_{m_n}$ in the first inequality. By continuity of $B^\star$, we have $B^\star(R_{m_n}) \to B^\star(R)$, and thus the right-hand side converges to 0 as $n \to \infty$. We deduce that $R = B^\star(R)$. But then, by uniqueness of the fixed point of $B^\star$, $R = Q_{\mathrm{ent}}$. This shows that the bounded sequence $(R_n)_{n\in\mathbb{N}}$ admits a unique accumulation point Q_{ent}, and thus converges to Q_{ent}. We have thus shown that for any strictly increasing sequence $(\alpha_n)_{n\in\mathbb{N}}$, $R_{\alpha_n} \to Q_{\mathrm{ent}}$ as $n \to \infty$. This concludes the proof by the sequential characterization of convergence of a function. □

B.2 Proof of Corollary 7

Proof. We focus on monotonicity since convergence follows immediately from Theorem 6 and (6). Let α and β be in $\mathbb{R}_{\geq 0}$, with $\alpha \leq \beta$. Let $x \in \mathcal{X}_{\mathrm{V}}$. By definition of $\pi^\star_\alpha$ and $\pi^\star_\beta$, we have both

$$\begin{aligned}\bar{G}(x, \pi^\star_\alpha) + \alpha \bar{S}(x, \pi^\star_\alpha) &\geq \bar{G}(x, \pi^\star_\beta) + \alpha \bar{S}(x, \pi^\star_\beta), \\ \bar{G}(x, \pi^\star_\alpha) + \beta \bar{S}(x, \pi^\star_\alpha) &\leq \bar{G}(x, \pi^\star_\beta) + \beta \bar{S}(x, \pi^\star_\beta).\end{aligned}$$

Taking the difference and rearranging yields

$$(\alpha - \beta)(\bar{S}(x, \pi^\star_\alpha) - \bar{S}(x, \pi^\star_\beta)) \geq 0.$$

We deduce that the two factors have the same sign, i.e., $\bar{S}(x, \pi^\star_\alpha) \leq \bar{S}(x, \pi^\star_\alpha)$. Since this is valid for all $x \in \mathcal{X}_{\mathrm{V}}$, this concludes the proof. □

B.3 Proof of Theorem 8

We begin with two preliminary technical results on dynamic indicators. For any trajectory $\tau = [(x_n, a_n)]_{n\in\mathbb{N}} \subset \mathcal{Q}^{\mathbb{N}}$, we introduce the notations

$$\begin{aligned}T_{\mathrm{C}}(\tau) &= \min\{t \in \mathbb{N} \mid x_t \in \mathcal{X}_{\mathrm{C}}\}, \\ T_{\mathrm{R}}(\tau) &= \min\{t \in \mathbb{N} \mid x_t \in \mathcal{X}_{\mathrm{C}} \wedge x_{t+1} \notin \mathcal{X}_{\mathrm{C}}\},\end{aligned}$$

with the convention $\min \emptyset = \infty$.

Lemma 9 (Characterization of dynamic indicators)**.** Let $c : \mathcal{Q} \to \mathbb{R}_{\geq 0}$. Then, c is dynamic indicator of $\mathcal{X}_{\mathrm{C}}$ if, and only if, $c_{|\mathcal{Q}_{\mathrm{V}}} = 0$ and for any trajectory $\tau = [(x_t, a_t)]_{t\in\mathbb{N}} \subset \mathcal{Q}^{\mathbb{N}}$ such that $x_0 \in \mathcal{X}_{\mathrm{V}}$ and $T_{\mathrm{C}}(\tau) < \infty$, there exists $t \leq T_{\mathrm{R}}(\tau)$ such that $c(x_t, a_t) > 0$.

Proof. We first show the converse implication. Let $x \in \mathcal{X}_{\mathrm{V}}$ and $\pi \in \Pi$. If $\pi \notin \Pi_{\mathrm{V}}(x)$, then let $\tau = [(x_n, a_n)]_{n\in\mathbb{N}}$ be a trajectory starting from x and generated by π with positive probability such $T_{\mathrm{C}}(\tau) < \infty$. Let $t \leq T_{\mathrm{R}}(\tau)$ such that $c(x_t, a_t) > 0$. Then,

$$\bar{\rho}(x, \pi) \geq L \cdot \sum_{u=0}^{\infty} \gamma^u c(x_u, a_u) \geq L \cdot \gamma^t c(x_t, a_t) > 0,$$

where we defined $L = \mathbb{P}[((X(t; x, \pi), A(t; x, \pi)))_{t\in\{0,\dots,T\}} = ((x_t, a_t))_{t\in\{0,\dots,T\}}] > 0$ for conciseness. Conversely, if $\bar{\rho}(x, \pi) > 0$, then there must exist a trajectory $\tau = [(x_t, a_t)]_{t\in\mathbb{N}}$ starting from x and with positive probability such that $\sum_{t=0}^{\infty} \gamma^t c(x_t, a_t) > 0$. But there must then be a $t \in \mathbb{N}$ such that $(x_t, a_t) \notin \mathcal{Q}_{\mathrm{V}}$, by assumption on c. By definition, the trajectory τ reaches $\mathcal{X}_{\mathrm{C}}$ in finite time after that time t. Since τ has positive probability by following π, this shows that $\pi \notin \Pi_{\mathrm{V}}(x)$ and shows the implication.

For the converse implication, assume that c is a dynamic indicator of $\mathcal{X}_{\mathrm{C}}$. Let $(x, a) \in \mathcal{Q}_{\mathrm{V}}$, and take any policy $\pi \in \Pi_{\mathrm{V}}$ such that $\pi(a \mid x) = 1$ (such a policy exists by definition of the viability

kernel). Then, $0 = \bar{\rho}(x, \pi) \geq c(x, a)$, and thus $c_{|\mathcal{Q}_\mathrm{V}} = 0$. Next, let $\tau = [(x_t, a_t)]_{t\in\mathbb{N}}$ be such that $x_0 \in \mathcal{X}_\mathrm{V}$ and $T_\mathrm{C}(\tau) < \infty$. Introduce $T_\mathrm{V} = \max\{t \in \mathbb{N} \mid t \leq T_\mathrm{C}(\tau) \wedge (x_t, a_t) \in \mathcal{Q}_\mathrm{V}\}$, and define $(x, a) = (x_{T_\mathrm{V}+1}, a_{T_\mathrm{V}+1})$. For every $y \in \mathcal{X}$, define the set of actions that τ takes in state y,

$$\tau(y) = \{b \in \mathcal{A} \mid \exists t \in \mathbb{N}, (x_t, a_t) = (y, b)\}.$$

Let $\theta : \mathcal{X} \to \mathcal{A}$ be a map such that

- $\theta(x) = a$;
- for all $y \in \mathcal{X}_\mathrm{V}$ with $y \neq x$, $\theta(y) \in \mathcal{Q}_\mathrm{V}[y]$;
- for all $y \notin \mathcal{X}_\mathrm{V}$ with $\tau(y) \neq \emptyset$, $\theta(y) \in \tau(y)$.

Define the policy $\pi(b \mid y) = \delta_{\theta(y)}(b)$. Clearly, $\pi \notin \Pi_\mathrm{V}(x)$, since it takes action a in x, and thus $\bar{\rho}(x, \pi) > 0$. Define now $\tau_{0:\mathrm{R}} = \{(y, b) \in \mathcal{Q} \mid \exists t \in \{0, \dots, T_\mathrm{R}(\tau)\}, (y, b) = (x_t, a_t)\}$. Crucially, π only explores state-action pairs that are either in $\mathcal{Q}_\mathrm{V}$ or in $\tau_{0:\mathrm{R}}$ when initialized in $\mathcal{X}_\mathrm{V}$. We thus obtain (almost-surely)

$$\begin{aligned}
\bar{\rho}(x, \pi) &= \sum_{t=0}^{\infty} \gamma^t c(X(t; x, \pi), A(t; x, \pi)) \\
&= \sum_{t=0}^{\infty} \sum_{(y,b)\in\tau_{0:\mathrm{R}}} \gamma^t c(y, b) \cdot \delta_y(X(t; x, \pi)) \cdot \delta_b(A(t; x, \pi)) \\
&= \sum_{(y,b)\in\tau_{0:\mathrm{R}}} c(y, b) \cdot \sum_{t=0}^{\infty} \gamma^t \cdot \delta_y(X(t; x, \pi)) \cdot \delta_b(A(t; x, \pi)),
\end{aligned}$$

where we leveraged the fact that $c_{|\mathcal{Q}_\mathrm{V}} = 0$. Therefore, there must be $(y, b) \in \tau_{0:\mathrm{R}}$ such that $c(y, b) > 0$, concluding the proof. □

Corollary 10. Let c be a dynamic indicator of $\mathcal{X}_\mathrm{C}$, ρ the associated discounted risk (9), and $q \in \mathcal{Q}_\mathrm{crit}$. For any $q \in \mathcal{Q}_\mathrm{crit}$, $c(q) + \gamma \min_{\pi\in\Pi} \bar{\rho}(f(q), \pi) > 0$.

Proof. From classical results on dynamic programming, there exists $\eta \in \Pi$ such that $\bar{\rho}(f(q), \eta) = \min_{\pi\in\Pi} \bar{\rho}(f(q), \pi)$. Furthermore, η can be chosen deterministic. Consider then the trajectory of η starting from $f(q)$, and prepend q to it. Let us call τ the resulting trajectory. It begins in $\mathcal{X}_\mathrm{V}$ and $T_\mathrm{C}(\tau) < \infty$, since $q \in \mathcal{Q}_\mathrm{crit}$. Therefore, by Lemma 9, there exists $t \leq T_\mathrm{R}(\tau)$ such that $c(x_t, a_t) > 0$, where $\tau = [(x_u, a_u)]_{u\in\mathbb{N}}$. The result follows from $c(q) + \bar{\rho}(f(q), \eta) \geq \gamma^t c(x_t, a_t) > 0$. □

The following lemma is the core of the proof of Theorem 8: it upper-bounds the soft-Q-value on $\mathcal{Q}_\mathrm{crit}$, and lower-bounds it on $\mathcal{Q}_\mathrm{V}$. It is a generalization to the present setting of [2, Lemma 2].

Lemma 11. There exists constants $v_1, u_1, u_2 \in \mathbb{R}$ and $u_3 \in \mathbb{R}_{>0}$ such that, for all $\alpha \in \mathbb{R}_{>0}$ and $p \in \mathbb{R}_{\geq 0}$,

$$\min_{\mathcal{Q}_\mathrm{V}} Q_{\alpha,p} \geq v_1, \tag{16}$$

$$\max_{\mathcal{Q}_\mathrm{crit}} Q_{\alpha,p} \leq u_1 + \alpha \cdot u_2 - p \cdot u_3. \tag{17}$$

Proof. By [28, Theorem 1], we have in particular for all $\alpha \in \mathbb{R}_{\geq 0}$ and $p \in \mathbb{R}_{\geq 0}$ that $Q_{\alpha,p} \geq Q^\pi_{\alpha,p}$, where $Q^\pi_{\alpha,p}$ is the soft Q-value function of an arbitrary policy $\pi \in \Pi$, that is, the only fixed point of the operator $B^\pi_{\alpha,p}$ defined for all $h \in \mathbb{R}^\mathcal{Q}$ as

$$(B^\pi_{\alpha,p} h)(x, a) = r(x, a) + \gamma \mathbb{E}[h(x', A_1) - \ln \pi(A_1 \mid x')],$$

for all $(x, a) \in \mathcal{Q}$ and where we defined the shorthand $x' = f(x, a)$. Take for π a deterministic policy. Then, $B^\pi_{\alpha,p}$ simplifies to the classical Bellman operator, and thus the soft-Q-value function coincides

with the classical Q-value function Q_p^π. Therefore, $Q_{\alpha,p} \geq Q_p^\pi$ for any $\pi \in \Pi$ deterministic. This holds in particular if $\pi \in \Pi_{\mathrm{V}}$, for which

$$\begin{aligned} Q_p^\pi(x,a) &= r(x,a) - p \cdot c(x,a) + \gamma(G(x',\pi) - p\rho(x',\pi)) \\ &= r(x,a) + \gamma G(x',\pi), \end{aligned}$$

for all $(x,a) \in \mathcal{Q}_{\mathrm{V}}$ and with $x' = f(x,a)$. Indeed, $c(x,a) = 0$ by Lemma 9 since $(x,a) \in \mathcal{Q}_{\mathrm{V}}$, and $\rho(x',\pi) = 0$ since $\pi \in \Pi_{\mathrm{V}}(x')$. Since the right-hand side is lower-bounded (by boundedness of r) by a constant independent of p and α, we deduce the existence of v_1 as announced.

In contrast, from [22, Theorem 16], it follows that for all $(x,a) \in \mathcal{Q}_{\mathrm{crit}}$

$$\begin{aligned} Q_{\alpha,p}(x,a) &= \max_{\pi\in\Pi} r(x,a) + \gamma\bar{G}(x',\pi) + \alpha\gamma S(x',\pi) - pc(x,a) - p\gamma\bar{\rho}(x',\pi) \\ &\leq \max_{\pi\in\Pi} \frac{1}{1-\gamma}(\sup_{\mathcal{Q}} r + \alpha\gamma|\mathcal{A}|\ln|\mathcal{A}|) - pc(x,a) - p\gamma\bar{\rho}(x',\pi) \\ &= u_1 + \alpha \cdot u_2 - p \cdot \left(c(x,a) + \gamma \min_{\pi\in\Pi} \bar{\rho}(x',\pi) \right), \end{aligned}$$

with $x' = f(x,a)$. Here, we have defined $u_1 = \frac{1}{1-\gamma}\sup_{\mathcal{Q}} r$ and $u_2 = \frac{1}{1-\gamma}|\mathcal{A}|\ln|\mathcal{A}|$. Furthermore, let $u_3 = \min_{q\in\mathcal{Q}_{\mathrm{crit}}}[c(q) + \min_{\pi\in\Pi}\bar{\rho}(f(q),\pi)]$, which is positive by Corollary 10 and since $\mathcal{Q}_{\mathrm{crit}}$ is finite. This yields the desired upper bound for all $(x,a) \in \mathcal{Q}_{\mathrm{crit}}$. Since the constants are independent of α and p, this concludes the proof. □

This enables showing convergence of the unconstrained, penalized soft-Q-value to its constrained counterpart.

Lemma 12. For all $\alpha > 0$, we have

$$\begin{aligned} \sup_{\mathcal{Q}_{\mathrm{crit}}} Q_{\alpha,p} &\xrightarrow[p\to\infty]{} -\infty, \\ \sup_{\mathcal{Q}_{\mathrm{V}}} |Q_{\alpha,p} - Q_\alpha| &\xrightarrow[p\to\infty]{} 0. \end{aligned}$$

Proof. The first claim follows immediately from Lemma 11. For the second one, recall from [22, Theorem 16] that for all $(x,a) \in \mathcal{Q}$

$$Q_{\alpha,p}(x,a) = \max_{\pi\in\Pi} r(x,a) + \gamma\bar{G}(x',\pi) + \alpha\gamma S(x',\pi) - pc(x,a) - p\gamma\bar{\rho}(x',\pi).$$

It follows that the function $p \mapsto Q_{\alpha,p}(x,a)$ is nonincreasing. If $(x,a) \in \mathcal{Q}_{\mathrm{V}}$, it is also lower-bounded by Lemma 11. Therefore, there exists $\bar{Q} : \mathcal{Q}_{\mathrm{V}} \to \mathbb{R}$ such that $\lim_{p\to\infty} Q_{\alpha,p} = \bar{Q}$, pointwise on $\mathcal{Q}_{\mathrm{V}}$ (and, thus, uniformly as well). We show $\bar{Q} = Q_\alpha$.

Let $(x,a) \in \mathcal{Q}_{\mathrm{V}}$ and $x' = f(x,a)$. For any $p \in \mathbb{R}_{\geq 0}$, the definition of $Q_{\alpha,p}$ as the unique fixed point [22] of the soft-Bellman operator associated to the reward $r - pc$ gives

$$Q_{\alpha,p}(x,a) = r(x,a) + \alpha\gamma \ln \left[\sum_{b\in\mathcal{A}} \exp\left[\frac{1}{\alpha} Q_{\alpha,p}(x',b) \right] \right],$$

recalling that $c(x,a) = 0$. Taking the limit as $p \to \infty$ on both sides yields

$$\begin{aligned} \bar{Q}(x,a) &= r(x,a) + \alpha\gamma \ln \left[\sum_{b\in\mathcal{A}} \exp\left[\frac{1}{\alpha} \lim_{p\to\infty} Q_{\alpha,p}(x',b) \right] \right] \\ &= r(x,a) + \alpha\gamma \ln \left[\sum_{b\in\mathcal{Q}_{\mathrm{V}}[x']} \exp\left[\frac{1}{\alpha}\bar{Q}(x',b) \right] + \sum_{b\in\mathcal{Q}_{\mathrm{crit}}[x']} 0 \right] \\ &= r(x,a) + \alpha\gamma \ln \left[\sum_{b\in\mathcal{Q}_{\mathrm{V}}[x']} \exp\left[\frac{1}{\alpha}\bar{Q}(x',b) \right] \right], \end{aligned}$$

where we used the fact that $\lim_{p\to\infty} Q_{\alpha,p}(x,b) = -\infty$ for all $b \in \mathcal{Q}_{\mathrm{crit}}[x']$. In other words, $\bar{Q}$ is a fixed point of the same Bellman operator as Q_α. Since that fixed point is unique and equal to Q_α, this concludes the proof. □

Proof of Theorem 8. Let δ, ϵ, and α be in $\mathbb{R}_{>0}$. Let $p_1 = \frac{u_1 - v_1}{u_3} + \frac{u_2 - \ln \delta}{u_3}\alpha$ and $p > p_1$. By Lemma 11, we have

$$\begin{aligned}
\pi^{\star}_{\alpha,p}(a \mid x) &= \frac{\exp\left[\frac{1}{\alpha}Q_{\alpha,p}(x,a)\right]}{\sum_{b\in\mathcal{A}}\exp\left[\frac{1}{\alpha}Q(x,b)\right]} \\
&\leq \exp\left[\frac{u_1 - v_1}{\alpha} + u_2 - \frac{u_3}{\alpha}p\right] \\
&< \delta,
\end{aligned}$$

for all $(x, a) \in \mathcal{Q}_{\text{crit}}$.

Next, it follows from Lemma 12 that $\exp\left[\frac{1}{\alpha}Q_{\alpha,p}(x,a)\right]$ converges to $\exp\left[\frac{1}{\alpha}Q_{\alpha}(x,a)\right]$ if $(x, a) \in \mathcal{Q}_{\text{V}}$, and to 0 if $(x, a) \in \mathcal{Q}_{\text{crit}}$. It follows immediately that there exists $p_2 \in \mathbb{R}_{\geq 0}$ such that, for all $(x, a) \in \mathcal{Q}_{\text{V}}$ and $p > p_2$,

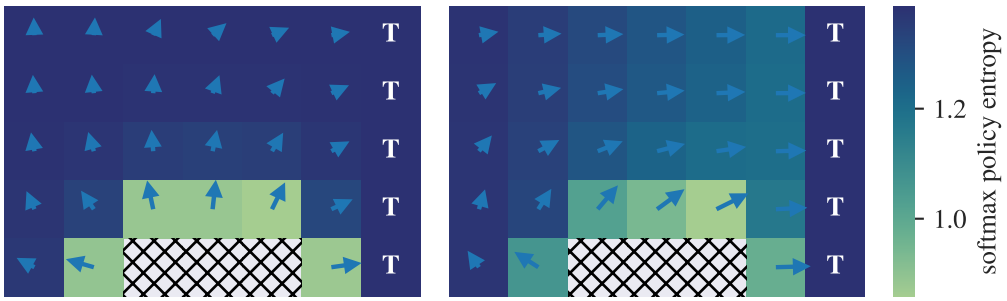
$$\left|\text{softmax}\left[\frac{1}{\alpha}Q_{\alpha,p}(x,\cdot)\right](a) - \text{softmax}\left[\frac{1}{\alpha}Q_{\alpha}(x,\cdot)\right](a)\right| < \epsilon.$$

The result follows by taking $p^{\star} = \max\{p_1, p_2\}$. □

C Extended cliff walking results

C.1 Policy entropy

In Fig. 5 we show the entropy of the optimal policy in each state for both the maximum entropy policy and the regularized policy.



(a) Maximum entropy (b) Regularized policy

Figure 5: **Maximum entropy policy (a) vs. regularized policy (b):** To avoid states with unviable actions, $\pi^{\star}_{\text{ent}}$ moves away from the constraint (a). The regularized policy ($\alpha = 4$) trades entropy for return and stays closer to the constraints (b). The colormap is the entropy of the optimal policy and the blue arrows the expected actions. Shorter arrows thus mean a higher action distribution entropy.

C.2 δ-viability of the optimal soft-max policy

We show the numerical results for δ-viability in Fig. 6.

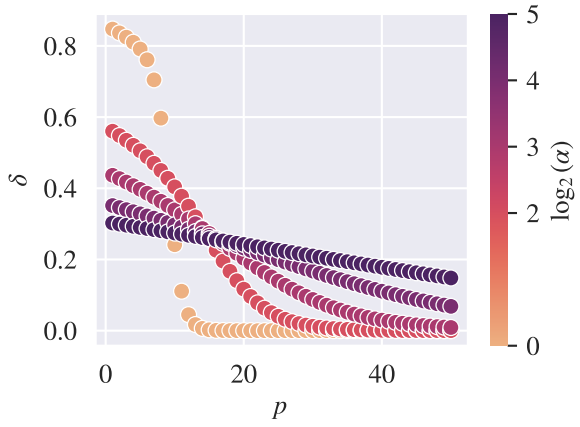


Figure 6: **Effect of the temperature and penalty on δ-viability. δ-viability:** As the penalty for $\mathcal{X}_{\text{C}}$ is increased δ quickly goes to zero.

D Extended RL benchmark results

D.1 Detail of the modified pendulum environment

In Fig 7 we show a sketch of the modified pendulum environment. Clearly, the upright position is the most robust state with the largest distance to the constraint set.

We set the reward function to

$$r(x, a) = -(\theta - \theta_{\text{target}})^2, \tag{18}$$

in order to instigate the agent to move towards the edge of the viability kernel.

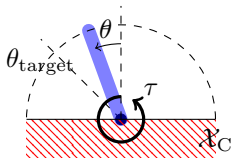


Figure 7: A sketch of the pendulum environment with target angle θ_{target} and constraint set $\mathcal{X}_{\text{C}}$

.

D.2 Success rate under disturbance as a function of α

We show in Fig. 8 and Fig. 9 the dependence between the training temperature temperature and the degree of disturbance the agent can withstand. Generally, as the training temperature increases so does the robustness of the mode of the learned policy. For the hopper environment (Fig. 9) the trend is intact but our results exhibit high variance over different training runs, especially for high temperatures. We attribute this to suboptimal solutions found during learning. When inspecting the trained policy's behavior we see that sometimes the agent learns to stand still in order to increase robustness.

Figure 8: **Robustness to disturbances for the pendulum environment:** Evaluation of a policy trained with different temperatures α. We show the mean and standard deviation of the success rate evaluated over training with 25 random seeds. As the magnitude ϵ of the disturbance is increased, the mode policy retains higher success rates for higher temperatures.

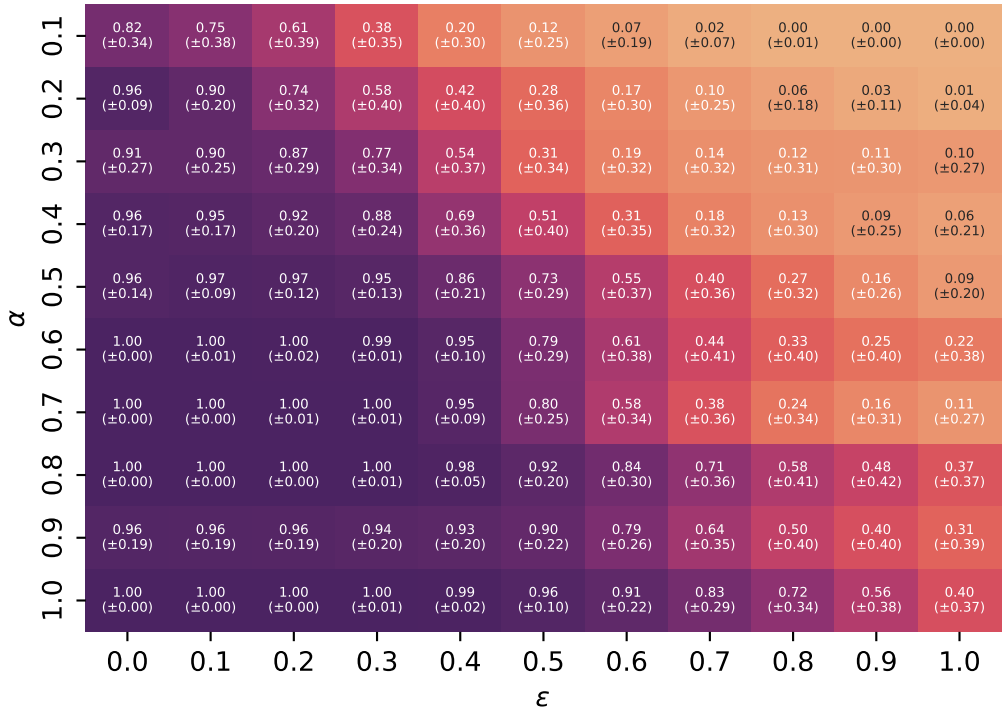


Figure 9: **Robustness to disturbances for the hopper environment:** Evaluation of a policy trained with different temperatures α. We show the mean and standard deviation of the success rate evaluated over training with 25 random seeds. As the magnitude ϵ of the disturbance is increased, the mode policy retains higher success rates for higher temperatures.

E Hyperparameters

All experiments in Sec. 6.2 were conducted on a compute cluster. In total around 50 core-hours on an Intel Xeon 8468 Sapphire and 450 core-hours on an NVIDIA H100 have been used to produce the results. The code will be published upon acceptance and is also part of the supplementary material of the submission.

After training, we evaluate the policy with the highest environment return seen during the training.

Table 1: Hyperparameters for tabular RL experiments.

Name	Value
Discount factor	0.95
Maximum number of iterations	$1,000$
Convergence tolerance	0.00001

Table 2: Hyperparameters for Deep RL experiments.

Name	Value (Pendulum)	Value (Hopper)
Penalty	90	300
Total steps	$250,000$	$600,000$
Buffer size	$250,000$	$600,000$
Batch size	256	256
Discount factor	0.99	0.99
Target smoothing coefficient	0.005	0.005
Policy learning rate	0.0003	0.0003
Q learning rate	0.001	0.001
Optimizer	Adam	Adam
Policy update frequency	2^{-1}	2^{-1}
Target network update frequency	1	1