POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Algorithms for Reward-based Coherent Risk Measures in Risk-Averse Reinforcement Learning

Tesi di Laurea Magistrale in
Computer Science and Engineering - Ingegneria Informatica

Massimiliano Bonetti, 944205

Advisor:
Prof. Marcello Restelli

Co-advisors:
Dott. Lorenzo Bisi

Academic year:
2020-2021

Abstract: In real-world problems such as robotics, finance and healthcare, the risk is always present and it is important to take it into consideration in order to limit the chance of rare but dangerous events. The literature on risk-averse reinforcement learning has touched coherent risk measures based on the long-term return or reward-based risk measures that are not coherent. Here we present two new risk-averse objectives that are both coherent and based on the reward: the reward-based mean-mean absolute deviation (Mean-RMAD) and the reward-based conditional value at risk (RCVaR), showing the importance of coherence with an example. We prove that these risk measures bound the corresponding return-based risk measures, so increasing one of the former measures increases also the return-based version. We develop algorithms for these risk measures with guaranteed monotonic improvement. Furthermore, a meta-algorithm allows to solve the RCVaR optimisation by optimising instead a sequence of risk-neutral problems. Finally, we conduct an empirical analysis about how these approaches are effective in retrieving behaviours with different levels of risk-aversion on a financial environment and on noisy and challenging environments from PyBullet.

Key-words: reinforcement learning; risk-averse; coherent risk measure; reward-based measure; mean absolute deviation; CVaR

1. Introduction

Reinforcement learning (RL) (Sutton and Barto, 2018) is learning what to do over time in an environment, i.e. which action to choose in a situation, in order to maximize a reward signal in the classic risk-neutral case or also to minimize some forms of risks by achieving a trade-off between the reward and the considered risk. The learner is called the agent that interacts with the so-called environment, which is influenced by the actions of the agent and it gives as response rewards and observations that represent the environment's state. Reinforcement Learning methods have become very popular due to their ability of solving sequential decision making problems that we encounter very often in the real life, like in robotics, finance, healthcare (Yu et al., 2019), video games (Mnih et al., 2013), energy management (Mason and Grijalva, 2019). In fact Reinforcement Learning can be applied when the environment is too complex to be solved exactly with analytical methods or when the dynamics of the environment are unknown or are difficult to be modeled. In the classic risk-neutral stream of literature there are powerful solution methods like Trust Region Policy Optimization (TRPO) (Schulman et al., 2017a), Proximal Policy Optimization (PPO) (Schulman et al., 2017b), Soft Actor Critic (SAC) (Haarnoja et al., 2018)

that efficiently maximize the expected value of the cumulative discounted rewards (called expected return). Usually when dealing with realistic problems, like finance, robotics and healthcare, we want also to manage the *risk* in order to avoid bad events that can happen even if they are not very common. The risk can be split into three categories (Papini, 2021; García and Fernández, 2015): *inherent risk*, which is due to the stochasticity of the environment; *model risk*, that refers to the imperfect knowledge of the environment which makes the consequences of its actions difficult to predict; *action risk*, which is due to the stochasticity of the actions done with intention by the agent, typically in order to do exploration. The action risk is under direct control of the agent. The inherent risk can be addressed by optimizing specific objective functions called risk measures, differently from the commonly used expected return. The model risk can be reduced using safe policy updates, that we will provide thanks to the monotonic improvement guarantee of our algorithms.
Risk-averse Reinforcement Learning is not a new subject, several risk measures were introduced (Shapiro et al., 2021) like: conditional value at risk (CVaR) (Tamar et al., 2015b; Chow et al., 2015), variance-related measures (Di Castro et al., 2012; Tamar and Mannor, 2013; Sobel, 1982), utility function (Shen et al., 2014), entropic risk measure (Nass et al., 2019). More interesting are the coherent risk measures (Shapiro et al., 2021; Tamar et al., 2015a), that are characterized by convexity, monotonicity, translation equivariance and positive homogeneity. These properties allow for example to obtain solutions that are more rational like avoiding policies that always give the lowest possible reward. Furthermore, in Bisi et al. (2020) was introduced a new risk measure based on the reward instead of the return: the Mean-Volatility, which smooths the trajectories avoiding shocks.
In this work we want to capture the advantages of the coherence properties and of the reward-based measures by introducing two new risk measures that are both coherent and reward-based: the Mean-RMAD and the RCVaR, where the Mean is the normalized expected return, the RMAD stands for Reward-based Mean Absolute Deviation and RCVaR is the Reward-based Conditional Value at Risk. Furthermore we provide safe updates, thanks to the Performance Difference Lemma that allows to develop a TRPO-like algorithm for both measures with guaranteed monotonic improvement, which leads also to PPO for the RMAD. Moreover, we can use any risk-neutral Reinforcement Learning algorithm to optimize the CVaR maintaining the monotonic improvement.
We recall some Reinforcement Learning concepts about the risk measures in Section 2, that will be useful in the rest part of the document. In Section 3 we introduce the Mean-RMAD and the RCVaR, we compare them together and with other risk measures and we explain the utility of the coherence properties with a practical example. In Section 4 we explain RMAD-TRPO and RMAD-PPO, while Section 5 is dedicated to the algorithms for the RCVaR. Finally, in Section 6, we conduct an empirical analysis of the new algorithms on a financial environment, on two challenging robotic environments Hopper and Walker from PyBullet (Coumans and Bai, 2016–2021) and on some easier environments like a multi-armed bandit problem and an environment called *Point Reacher* (Bisi, 2022).

2. Preliminaries

In this section, we provide the necessary background that will be employed in the following of the document.

Mathematical Background. Given a measurable space $(\mathcal{X}, \sigma_{\mathcal{X}})$, where $\mathcal{X}$ is a set and $\sigma_{\mathcal{X}}$ a σ-algebra, we denote with $\Delta_{\mathcal{X}}$ the set of probability measures and with $\mathscr{B}(\mathcal{X})$ the set of bounded measurable functions. Given any probability $\mathcal{P} \in \Delta_{\mathcal{X}}$, $\mathbb{P} = (\mathcal{X}, \sigma_{\mathcal{X}}, \mathcal{P})$ is a probability space. On this space we define uncertain outcomes $Z = Z(x)$, which are random functions over the outcomes $x \in \mathcal{X}$. The space $\mathcal{Z}_{\mathcal{X}}$ of allowable random functions Z we deal with is $\mathcal{Z} \coloneqq \mathcal{L}_p(\mathcal{X}, \sigma_{\mathcal{X}}, \mathcal{P})$, where $p \in [1, \infty)$, so the random variable Z has a finite p-th order moment. We denote with $\succeq$ a *pointwise partial order* over $\mathcal{Z}$: given $Z, Z' \in \mathcal{Z}$, $Z \succeq Z'$ means that $Z(x) \geqslant Z'(x)$ for $\mathcal{P}$-almost all $x \in \mathcal{X}$.

Markov Decision Processes. We consider discrete-time, discounted Markov Decision Process (MDP) with infinite time-horizon (Puterman, 2014). An MDP is a tuple $(\mathcal{S}, \mathcal{A}, P, R, \gamma, \mu)$, where:

- $\mathcal{S}$ is the (continuous) state space, that contains all the possible states of the environment;
- $\mathcal{A}$ is the (continuous) action space, that contains the possible actions that an agent can do in any state;
- $P(\cdot|s,a) \in \Delta_{\mathcal{S}} \quad \forall s \in \mathcal{S}, a \in \mathcal{A}$ is the Markovian transition kernel that indicates the probability of reaching a specific state when performing action a in state s;
- $R : \mathcal{S} \times \mathcal{A} \to [-R_{max}, R_{max}]$ is the bounded reward function;
- $\gamma \in (0,1)$ is the discount factor, that exponentially discounts future rewards;
- $\mu(\cdot) \in \Delta_{\mathcal{S}}$ is the starting-state distribution that indicates the probability of starting in a specific state.

The agent's behaviour is determined by a Markovian, stationary policy, defined as the mapping $\pi : \mathcal{S} \to \Delta_{\mathcal{A}}$, where $\pi(a|s)$ is the probability of performing action a while being on state s. We will sometimes restrict our attention to parametric policies $\pi_{\boldsymbol{\theta}} \in \Pi_{\Theta}$ identified by a vector of parameters $\boldsymbol{\theta} \in \Theta \subseteq \mathbb{R}^m, m \geqslant 1$.
Following policy π, the interaction between the agent and the environment determines a trajectory $\tau =$

$(s_0, a_0, r_0, s_1, a_1, r_1, ...)$, where $s_0 \sim \mu(\cdot)$, $a_t \sim \pi(\cdot|s)$, $r_t = R(s_t, a_t)$ and $s_{t+1} \sim P(\cdot|s_t, a_t)$ for all $t \geqslant 0$. We call $\mathcal{T}$ the set of all possible trajectories. A trajectory is a *random variable* whose probability density, given some policy π, is:

$$p_\pi(\tau) = \mu(s_0) \prod_{t=0}^{\infty} \pi(a_t|s_t) P(s_{t+1}|s_t, a_t).$$

Given a trajectory τ followed by the agent, the discounted cumulative reward is called *return* and it is defined as follows:

$$G_\gamma(\tau) \coloneqq \sum_{t=0}^{\infty} \gamma^t R(s_t, a_t),$$

where γ is dropped when clear from the context. The conditional expectation of the return w.r.t. a policy π given some state s is called *value function* and it is defined as:

$$V_\pi(s) \coloneqq \mathop{\mathbb{E}}_{\substack{a_t \sim \pi(\cdot|s_t) \\ s_{t+1} \sim P(\cdot|s_t, a_t)}} \left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t) \mid s_0 = s \right] = \mathop{\mathbb{E}}_{\tau|\pi} \left[G(\tau) \mid s_0 = s \right].$$

The value of deviating for one step from the policy π by taking action a is instead measured by the *action value function*:

$$Q_\pi(s, a) \coloneqq \mathop{\mathbb{E}}_{\substack{s_{t+1} \sim P(\cdot|s_t, a_t) \\ a_{t+1} \sim \pi(\cdot|s_{t+1})}} \left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t) \mid s_0 = s, a_0 = a \right] = \mathop{\mathbb{E}}_{\tau \sim p_\pi(\cdot)} \left[G(\tau) \mid s_0 = s, a_0 = a \right],$$

and the gain of such deviation is given by the *advantage function*:

$$A_\pi(s, a) \coloneqq Q_\pi(s, a) - V_\pi(s).$$

These function enjoy fundamental recursive relations called Bellman equations (Bellman, 1954):

$$V_\pi(s) = \mathop{\mathbb{E}}_{a \sim \pi(\cdot|s)} \left[R(s, a) \right] + \gamma \mathop{\mathbb{E}}_{\substack{s' \sim P(\cdot|s,a) \\ a \sim \pi(\cdot|s)}} \left[V_\pi(s') \right],$$

$$Q_\pi(s, a) = R(s, a) + \gamma \mathop{\mathbb{E}}_{s' \sim P(\cdot|s,a)} \left[V_\pi(s') \right].$$

When considering parametric policies, it is convenient to evaluate the agent performance w.r.t. a scalar criterion called *expected return*:

$$J_\pi^\mu \coloneqq \mathbb{E}_{s_0 \sim \mu(\cdot)} \left[V_\pi(s_0) \right] = \mathop{\mathbb{E}}_{\tau \sim p_\pi(\cdot)} \left[G(\tau) \right],$$

which represents the expectation of the return w.r.t. the starting-state distribution μ[1].
The (discounted) state occupancy measure induced by policy π is defined as:

$$d_{\mu,\pi}(s) \coloneqq (1 - \gamma) \int_{\mathcal{S}} \mu(s_0) \sum_{t=0}^{\infty} p_\pi(s_0 \xrightarrow{t} s) \mathrm{d}s_0,$$

where $p_\pi(s_0 \xrightarrow{t} s)$ is the probability of reaching state s after t steps starting from state s_0 and following policy π, [2]. Similarly, for each state-action pair (s, a) we define, overloading the notation, the following measure:

$$d_{\mu,\pi}(s, a) \coloneqq d_{\mu,\pi}(s) \pi(a|s).$$

The expectation of the reward w.r.t. the state occupancy measure is called *normalized expected return* (sometimes we will use *n. expected return* for short):

$$\bar{J}_\pi^\mu \coloneqq \mathop{\mathbb{E}}_{\substack{s \sim d_{\mu,\pi}(\cdot) \\ a \sim \pi(\cdot|s)}} \left[R(s, a) \right].$$

The reason for the name is its close relationship with its *unnormalized* version $\bar{J}_\pi = (1 - \gamma) J_\pi$, as pointed out in (D'Oro et al., 2019; Bisi et al., 2020). The expected return J_π is also a standard maximization *objective* for the *risk-neutral* setting. This is particularly useful when dealing with parametric policies that are differentiable w.r.t. to their parameters: in such case, a (local) solution to the previous optimization can be obtained by means of *gradient ascent*, exploiting the Policy Gradient Theorem (Sutton and Barto, 2018), which provides a formula to compute such gradient:

$$\nabla_{\boldsymbol{\theta}} J_{\boldsymbol{\theta}} = \mathop{\mathbb{E}}_{\substack{s_0 \sim \mu(\cdot), \\ a_t \sim \pi(\cdot|s_t) \\ s_{t+1} \sim P(\cdot|s_t, a_t)}} \left[\nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a|s) Q_{\boldsymbol{\theta}}(s, a) \right].$$

[1] We drop the dependence from μ whenever it is clear from the context. When using parametric policies, we may abbreviate the dependence on $\pi_{\boldsymbol{\theta}}$ with the subscript $\boldsymbol{\theta}$.

[2] More details are provided in the Appendix A.

Risk-Measures. We recall here some useful definitions from the risk-related literature.

Definition 2.1 (Risk-Measure, (Shapiro et al., 2021)). *Given a probability space $(\mathcal{X}, \sigma_{\mathcal{X}}, \mathcal{P})$, and some uncertain outcome $Z \in \mathcal{Z}_{\mathcal{X}}$, we call* risk-measure *a function $\mathrm{f}(Z, \mathcal{P})$ which maps Z into the extended real line $\bar{\mathbb{R}} = \mathbb{R} \cup \{+\infty\} \cup \{-\infty\}$.*

A common and intuitive risk-measure is the *variance*:

$$\mathrm{Var}(Z, \mathcal{P}) \coloneqq \mathop{\mathbb{E}}_{x \sim \mathcal{P}(\cdot)} \left[\left(Z(x) - \mathop{\mathbb{E}}_{x \sim \mathcal{P}(\cdot)} [Z(x)] \right)^2 \right].$$

Trade-off criteria balancing expectation and variance of the random outcome are commonly used in risk-averse optimization. One of the most common one is the *(penalized) mean-variance* risk-measure:

$$\mathrm{MVar}^{\beta}(Z, \mathcal{P}) \coloneqq \mathop{\mathbb{E}}_{x \sim \mathcal{P}(\cdot)} [Z(x)] - \beta \mathrm{Var}(Z(x), \mathcal{P}),$$

where variance is *soft*-constrained, by considering it as a penalty weighted by some risk-aversion coefficient β. In order to guarantee that optimizing a risk-measure induces a *rational* behavior, such measure needs to respect some axioms.

Definition 2.2 (Coherent Risk-Measure, (Artzner et al., 1999)). *A risk-measure η, defined w.r.t. the uncertain outcome Z, is* coherent *if it satisfies the following properties for all $Z, Z' \in \mathcal{Z}$:*

- Concavity[3]*: $\eta(tZ + (1-t)Z') \geqslant t\eta(Z) + (1-t)\eta(Z') \quad \forall t \in [0, 1]$.*
- Monotonicity*: If $Z \succeq Z'$, then $\eta(Z) \geqslant \eta(Z')$.*
- Translation Equivariance*: $\forall a \in \mathbb{R} : \eta(Z + a) = \eta(Z) + a$.*
- Positive Homogeneity*: $\forall t > 0 : \eta(tZ) = t\eta(Z)$.*

Many coherent risk-measures (CRMs) have been developed (Shapiro et al., 2021, Ch. 6), we recall here the most important ones for this work. The Conditional Value at Risk (CVaR,Rockafellar et al. (2000)) with risk-aversion level $\alpha \in (0, 1)$ is a CRM:

$$\mathrm{CVaR}^{\alpha}(Z, \mathcal{P}) \coloneqq \max_{\rho} \left\{ \rho - \frac{1}{\alpha} \mathop{\mathbb{E}}_{x \sim \mathcal{P}(\cdot)} \left[(Z(x) - \rho)_- \right] \right\}. \tag{1}$$

The value $\rho^{\star}$ which maximizes the above optimization problem is called *value-at-risk* (VaR). When no probability atoms are present, a more straightforward formulation for CVaR is:

$$\mathrm{CVaR}^{\alpha}(Z, \mathcal{P}) = \mathop{\mathbb{E}}_{x \sim \mathcal{P}(\cdot)} [Z(x) | Z(x) < \mathrm{VaR}^{\alpha}(Z, \mathcal{P})]$$

and its VaR coincides with the α-quantile of the cumulative distribution. The CVaR captures the mean of the worst outcomes, in this way its optimization reduces the bad events. The Mean-Absolute-Deviation (MAD), defined as:

$$\mathrm{MAD}(Z, \mathcal{P}) \coloneqq \mathop{\mathbb{E}}_{x \sim \mathcal{P}(\cdot)} \left[\left| Z(x) - \mathop{\mathbb{E}}_{x \sim \mathcal{P}(\cdot)} [Z(x)] \right| \right], \tag{2}$$

is not a CRM. However, the corresponding penalized trade-off with the mean, called Mean-MAD (MMAD), and defined as follows:

$$\mathrm{MMAD}^{\lambda}(Z, \mathcal{P}) \coloneqq \mathop{\mathbb{E}}_{x \sim \mathcal{P}(\cdot)} [Z(x)] - \lambda \mathrm{MAD}(Z, \mathcal{P}), \tag{3}$$

is coherent whenever $0 \leqslant \lambda \leqslant \frac{1}{2}$ (Shapiro et al., 2021). It includes the classic mean and it penalizes the deviations from the mean, that can cause high variability of the results. Differently from the variance, the MAD doesn't square the deviations, but they affect only with the distance from the mean, while the variance weights more deviations bigger than one and less the deviations smaller than one.

Risk-Averse Reinforcement Learning. We consider a risk-averse optimization reinforcement learning context, in which the agent does not seek to maximize the risk-neutral objective J_{π}, but some risk-averse variant of it, typically a risk-measure. We introduce the following distinction among the risk-measures of interest for reinforcement learning, which classifies them according to the considered probability spaces and uncertain outcomes.

Definition 2.3 (Return-based and Reward-based Measures). *A risk-measure $\mathrm{f}(Z, \mathcal{P})$ defined w.r.t. a probability space $\mathbb{P}$ and an uncertain outcome Z is called:*

- return-based*, if $\mathbb{P} = (\mathcal{T}, \sigma_{\mathcal{T}}, p_{\pi})$, and $Z = G(\tau)$;*
- reward-based*, if $\mathbb{P} = (\mathcal{S} \times \mathcal{A}, \sigma_{\mathcal{S} \times \mathcal{A}}, d_{\mu,\pi})$, and $Z = R(s, a)$.*

[3] In the minimization formulation we have to substitute this property with *convexity*.

As discussed in Bisi et al. (2020), the return-based risk measures can capture only the risk on the return, thus they are insensitive to short-term risk; while the reward-based risk measures captures short-term risk because they consider the per-step reward, smoothing the trajectories that avoid shocks. Reinforcement literature has only employed, until lately, *return-based* risk-averse measure as:

- the *return* mean-variance: $\text{MVar}^{\beta}(G, p_\pi)$ (Di Castro et al., 2012; Tamar and Mannor, 2013; Sobel, 1982),
- the conditional value-at-risk *over returns*: $\text{CVaR}^{\alpha}(G, p_\pi)$ (Chow et al., 2017; Tamar et al., 2015b; Chow et al., 2015),
- the *return* Mean-MAD: $\text{MMAD}^{\lambda}(G, p_\pi)$ (Tamar et al., 2015a),

together with many other return-based criteria (Howard and Matheson, 1972; Nass et al., 2019). Recent works have though considered a measure called *reward-volatility* or *per-step reward-volatility* (Bisi et al., 2020; Zhang et al., 2020):

$$\text{Var}(R, d_{\mu,\pi}) \coloneqq \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}} \left[\left(R(s,a) - \bar{J}_\pi\right)^2\right].$$

A trade-off objective, called *mean-volatility*, has also been defined in those works as:

$$\text{MVar}^{\beta}(R, d_{\mu,\pi}) \coloneqq \bar{J}_\pi - \beta\text{Var}(R, d_{\mu,\pi}).$$

According to our classification, this is the solely *reward-based* risk-averse objective analysed in literature so far. This objective has shown to have many advantages from the optimization viewpoint. First, Bellman expectation equations are available, hence, policy evaluation can be carried out in an efficient way. Secondly, a policy gradient theorem can be obtained, which allow to pursue local policy optimization. Interestingly, it is possible to follow the gradient in a safe way, by employing a trust-region approach (TRVO, Bisi et al. (2020)). Moreover, it has been shown by Zhang et al. (2020) that the optimization of this objective can be turned in sequence of risk-neutral problems featuring a modified reward, solvable with any state-of-the-art approach. These works have also demonstrated that reward volatility upper bounds the return variance:

$$\text{Var}(G, p_\pi) \leq \frac{1}{(1-\gamma)^2}\text{Var}(R, d_{\mu,\pi}), \tag{4}$$

thus, minimizing the first one also helps reducing the latter one.

3. Coherent Reward-based Risk-Averse Objectives

Despite its interesting mathematical properties, the mean-volatility objective is *not coherent*. This may result in some drawbacks, which are extensively illustrated in Section 3.2. In order to avoid these negative effects, we introduce in this section two *coherent* and *reward-based* risk-measures for risk-averse reinforcement learning: the *reward-based* Mean-MAD and the *reward-based* CVaR. The reasons behind the choice of these particular two coherent measures among a large variety of options, are related to possibility of deriving effective algorithms for their optimization, a topic that will be better clarified later in this section.

3.1. Mean-RMAD and RCVaR

We explicitly state here the formal definitions for the two risk-measure of interest, which directly follows from (1), (2) and (3) applied to the reward-based context of Definition 2.3.

Definition 3.1 (Mean-RMAD). *Given an MDP $\mathcal{M}$ and a policy π, we define the* reward-based mean absolute deviation (RMAD) *as:*

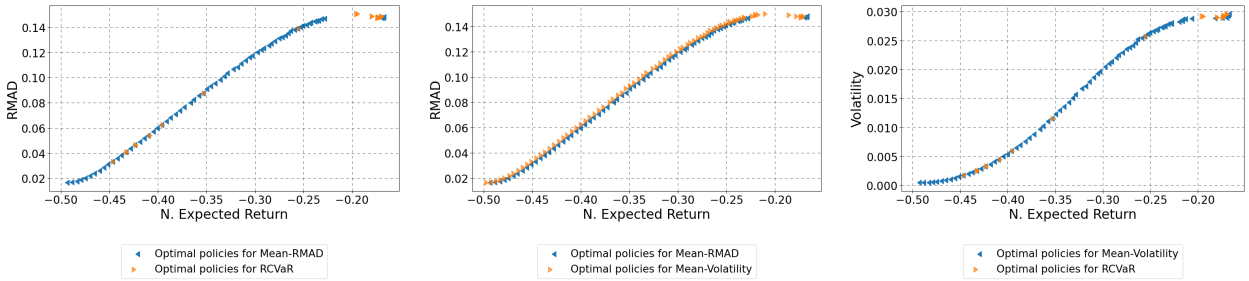
$$\text{MAD}(R, d_{\mu,\pi}) \coloneqq \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}} \left[\left|R(s,a) - \bar{J}_\pi\right|\right].$$

By setting a risk-aversion factor λ, we can define also a trade-off measure called reward-based Mean-MAD (Mean-RMAD)*:*

$$\text{MMAD}^{\lambda}(R, d_{\mu,\pi}) \coloneqq \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}} [R(s,a)] - \lambda\text{MAD}(R, d_{\mu,\pi}).$$

Definition 3.2 (RCVaR). *Given an MDP $\mathcal{M}$, a policy π and $\alpha \in (0,1)$, we define the* reward-based conditional value-at-risk (RCVaR) *as:*

$$\text{CVaR}^{\alpha}(R, d_{\mu,\pi}) \coloneqq \max_{\rho} \left\{ \rho - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}} \left[(R(s,a) - \rho)_-\right] \right\},$$



(a) Mean-RMAD vs RCVaR (b) Mean-RMAD vs Volatility (c) Mean-Volatility vs RCVaR

Figure 1: Comparison between the optimal policies of Mean-RMAD, RCVaR and Mean-Volatility, obtained with brute force on the environment *Point Reacher* (Bisi, 2022), more details are in Appendix H.3. Policies optimized w.r.t. different criteria are denoted with different colors for each Pareto plot. Pareto frontiers share some points but are not completely overlapped.

and the value ρ_π^α *for which the above program is optimal is* $\mathrm{VaR}^\alpha(R, d_{\mu,\pi})$ *(called RVaR).*

The following proposition relates the introduced reward-based risk measures with their corresponding return-based versions.

Proposition 3.1. *Given an MDP* $\mathcal{M}$ *and a policy* π, $\forall \alpha \in (0,1)$, *the following relationships hold between reward-based and return-based risk-measures:*

$$\mathrm{MAD}(G, p_\pi) \leq \frac{\mathrm{MAD}(R, d_{\mu,\pi})}{(1-\gamma)}, \tag{5}$$

$$\mathrm{CVaR}^\alpha(G, p_\pi) \geq \frac{\mathrm{CVaR}^\alpha(R, d_{\mu,\pi})}{(1-\gamma)}. \tag{6}$$

This result tells us that optimizing reward-based risk measures amounts to bound the corresponding return-based risk measure, similarly to what happens for reward-volatility in Equation (4). It is also possible to establish relationships among reward-based measures.

Proposition 3.2. *Given an MDP* $\mathcal{M}$ *and a policy* π, $\forall \alpha \in (0,1)$, *the following relationships hold among reward-based risk-measures:*

$$\mathrm{CVaR}^\alpha(R, d_{\mu,\pi}) \geq \mathrm{MMAD}^\lambda(R, d_{\mu,\pi}) \quad \textit{if} \quad \lambda = \frac{1}{2\alpha}, \tag{7}$$

$$(\mathrm{MAD}(R, d_{\mu,\pi}))^2 \leq \mathrm{Var}(R, d_{\mu,\pi}). \tag{8}$$

Interestingly, the previous proposition lower bounds the RCVaR value of some policy w.r.t. the Mean-RMAD one. Therefore, optimizing the latter quantity can also see as a proxy to optimize the former one. However, these objectives does not coincides. This can be better understood, for instance, by looking at the example provided in Figure 1a. Here we compare the *three reward-based risk measures*: Mean-RMAD, RCVaR and Mean-Volatility on a simple task called *Point-Reacher*. As shown in the figure, the optimal policies of these three risk measures can be different, meaning that they capture different preferences w.r.t. risk, thus, the best risk measure to use in practice may depend on the problem at hand.

Our choice of analysing these two risk-measures is motivated by the particular combination of properties they enjoy. We compare in Table 1 the measures in exam with both state-of-the-art measures and other coherent reward-based measures, which have not been employed in literature yet, and, thus, could have been considered in our analysis too. The dimensions under which we analyse them involve both coherence features and reinforcement learning properties. Anticipating some of the results we will derive in the next session, we show that the chosen risk-measures, beyond being coherent, enjoy expectation Bellman equations and a formulation of the Performance Difference Lemma (Kakade and Langford, 2002). These features are fundamental for the development of *policy gradient* approaches, *safe improvement bounds* and effective *trust-region* approaches. Looking at the table, it is possible to notice that RCVaR and Mean-RMAD share indeed these properties with Mean-Volatility, which is not coherent though, due to the lack of the monotonicity property.

Risk Measure	Coherency	Convexity	Monotonicity	Translation Equivariance	Positive Homogeneity	Bellman equations	PDL
Mean-RMAD	✓	✓	✓	✓	✓	✓	✓
RCVaR	✓	✓	✓	✓	✓	✓	✓
Mean-Volatility	X	✓	X	✓	X	✓	✓
Mean-Variance	X	✓	X	✓	X	✓	NK
Mean-Squared Root Volatility*	X	✓	X	✓	✓	NK	NK
Mean-Semi-Volatility*	X	✓	✓	✓	X	✓	NK
Mean-Squared Root Semi-Volatility*	✓	✓	✓	✓	✓	NK	NK
RVaR and VaR	X	X	✓	✓	✓	NK	NK
CVaR	✓	✓	✓	✓	✓	✓	NK
Utility model	X	✓	✓	X	✓	NK	NK
Entropic risk measure	X	✓	✓	✓	X	✓	NK

Table 1: A recap on the interesting properties enjoyed by a number of return-based and reward-based risk measures. In gold there are the new risk measures introduced in this document. "NK" means Not Known. The risk-measures with an asterisk are defined in the Appendix E.

3.2. The Importance of Coherence: a Motivating Example

Risk-measures have originally been developed by the financial literature as a way of computing the necessary amount of cash that need to be reserved to shield against some potential risk (Artzner et al., 1999). In this context, the coherence axioms listed in 2.2 have been selected in order to guarantee a rational behavior in some situations (see also Artzner et al. (1999) and Tamar et al. (2015a)). In particular, each property can be motivated by its practical implications:

1. *Concavity*: it ensures that diversification is always beneficial to risk, a property shared also by other common approaches (Markowitz, 1952).
2. *Monotonicity*: it makes choices resulting in lower reward for each outcome riskier. It avoids risk-averse optimization to converge to degenerate solutions.
3. *Translation Equivariance*: it allows to exclude deterministic components from risk computation. In a reinforcement learning perspective, it also allow reward translation by a constant quantity, which may be beneficial in some tasks in order to enhance exploration.
4. *Positive Homogeneity*: it encodes the intuition that multiplying the exposition directly maps to risk. From a reinforcement learning viewpoint, this property allows reward scaling, which may be useful in practice in case of extreme reward ranges (very low, or very high) to avoid precision or overflow errors.

By looking at Table 1, it is possible to notice that important risk-measures as Mean-Variance, Mean-Volatility and Entropic Risk-Measure (ERM, (Howard and Matheson, 1972)) lack positive homogeneity or monotonicity properties. Therefore, optimizing these risk-averse objective may result in irrational behaviors. Violating the monotonicity is of particular concern, since it permits the agent to possibly converge to solutions which should be excluded instead. With the following example we will try to provide the reader some further intuition on this point.

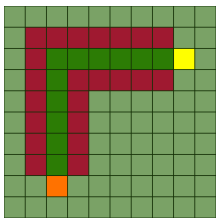
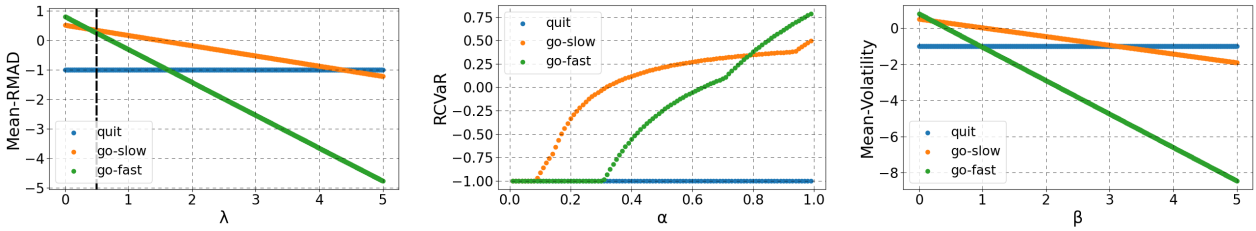


Figure 2: Graphical representation of the Grid-World Garden environment. The orange square is the starting-state, while the yellow square is the goal-state in which the agent receives the highest reward.



(a) Mean-RMAD (b) RCVaR (c) Mean-Volatility

Figure 3: Evaluation of three types of policies on the environment Garden: the go-fast policy is the one that reaches the goal state with high speed, the go-slow policy goes to the goal state with low speed, while the idle policy is the one that goes on the grass and stops in that absorbing state, thus in practice it can hurt the people that are around in the garden. We considered the risk measures: Mean-RMAD, RCVaR and Mean-Volatility. The vertical dashed line in Figure 3a indicates that over that value the risk measure is no more coherent. For a given risk-aversion level the policy with the highest value is the best one.

The Garden Example. Consider a real-world scenario in which a gardener-agent has to learn how to cut a hedge, and it must avoid collisions with flowers or people walking on the garden grass. In order to teach the agent how to accomplish this task, rewards are designed to penalize behaviours that may be dangerous for humans visiting the garden. We consider a simplified $2D$ model of a garden, represented by a grid-world with stochastic transitions, for which a pictorial representation is given in Figure 2. The agent can control his direction and speed, with higher speeds increasing the chance of losing control over its direction. Positive reinforce is provided for approaching and reaching some goal state. Collisions correspond instead to negative rewards and terminate the episode. Importantly, the lowest possible reward is achieved if the agent enters the grass, where people may be hit. To better understand the importance of coherence in this setting, we evaluate some fixed policies according to coherent and non-coherent criteria: RCVaR, Mean-RMAD, and Mean-Volatility. For each of these objectives, the performance was measured w.r.t. different risk-aversion levels. We considered three policies:

- *go-fast*: it follows the path to the goal-state with high speed (two steps per action);
- *go-slow*: it follows the path to the goal-state with low speed;
- *quit*: it quits the task prematurely, by immediately touching the grass.

We notice that the *quit* policy has the worse risk-neutral performance, giving always the lowest possible reward. On the other hand, by instantly quitting the task, this behaviour allows to obtain a low variability for the reward, hence, it may be preferred by an extremely risk-averse agent.We recall that, entering the grass, the gardener agent might risk to hurt the people in the area surrounding the hedge, a behaviour that we explicitly tried to discourage by providing a high penalty. Figures 3a, 3b, 3c show the performance of each policies according to, respectively, Mean-RMAD, RCVaR and Mean-Volatility. Thanks to the monotonicity property, coherent risk-measure assign the lowest value to the *quit* policy, since it always returns the lowest possible reward. Namely, we can see that this policy is never the best for Mean-RMAD with $0 \leq \lambda \leq 0.5$ or RCVaR. However, it can be noticed that the *quit* policy is selected for higher level of λ or β, the reason being that monotonicity is no longer guaranteed. Therefore, such risk-aversion levels may induce convergence of some learning algorithm towards dangerous policies as a byproduct of an excessive aversion to risk. While this phenomenon can be avoided a priori for Mean-RMAD by limiting the range of λ to the boundaries for which coherence is guaranteed, the same cannot be done with Mean-Volatility, hence, unwanted behaviors can only be spotted a posteriori.

The latter worked example certainly represents an oversimplified and unrealistic application of reinforcement learning to real-world tasks. However, it is quite indicative of the importance of coherence from a practical viewpoint. In general, whenever learning happens *online*, especially in a real setting, it is necessary to shield against potential bad outcomes in advance, in order to avoid that novel policies develop unwanted behaviors. A way to achieve this aim consists in instructing artificial agents to follow a coherent and risk-averse objective, which guarantees at the same time to bound the variability in the performance and to avoid degenerate outcomes. Another key tool for making online algorithms reliable is to provide *safe* guarantees to ensure a stable performance improvement. In the next sections we will show how to develop algorithm with such a property for both the considered risk-measure.

4. Mean-RMAD Optimization

In this section we focus on the following risk-averse reinforcement learning problem:

$$\max_{\pi \in \Pi_\Theta} \eta_\pi^\lambda := \text{MMAD}^\lambda(R, d_{\mu,\pi}),$$

where we re-defined Mean-RMAD with a new symbol to shorten the notation and, at the same time, highlight that it is the risk-averse objective we are optimizing in this context. We will also denote the RMAD of policy π with $\omega_\pi := \text{MAD}(R, d_{\mu,\pi})$.
We first derive some fundamental properties of the Mean-RMAD measures, and, then, we apply the obtained results to build algorithms to solve the related risk-averse reinforcement learning problem. By extending well-known results in safe policy optimization (Schulman et al., 2017a), we can obtain monotonic improvement guarantees, which form the basis for the development of practical trust-region approaches that can effectively employed for the solution of complex high-dimensional tasks.
Thus, in order to optimize the target objective, we focus on policy search approaches. Due to the presence of the L_1 norm in the definition of Mean-MAD, we have to resort to follow a sub-gradient, but this does not prevent us to obtain a convergent method.

4.1. Value functions and Bellman Expectation Equations

Here we introduce the value function for the RMAD as follows:

$$W_\pi(s) := \mathop{\mathbb{E}}_{\substack{a_t \sim \pi(\cdot|s_t) \\ s_{t+1} \sim P(\cdot|s_t, a_t)}} \left[\sum_{t=0}^{\infty} \gamma^t \left| R(s_t, a_t) - \bar{J}_\pi \right| | s_0 = s \right].$$

Similarly, we define also the action-value function as:

$$X_\pi(s, a) := \mathop{\mathbb{E}}_{\substack{s_{t+1} \sim P(\cdot|s_t, a_t) \\ a_{t+1} \sim \pi(\cdot|s_{t+1})}} \left[\sum_{t=0}^{\infty} \gamma^t \left| R(s_t, a_t) - \bar{J}_\pi \right| | s_0 = s, a_0 = a \right].$$

The value function of the Mean-RMAD can then be obtained as a linear combination of the classic risk-neutral value functions and the RMAD one:

$$\begin{aligned} V_\pi^\lambda(s) :=& V_\pi(s) - \lambda W_\pi(s) \\ =& \mathop{\mathbb{E}}_{\substack{a_t \sim \pi(\cdot|s_t) \\ s_{t+1} \sim P(\cdot|s_t, a_t)}} \left[\sum_{t=0}^{\infty} \gamma^t \left(R(s_t, a_t) - \lambda \left| R(s_t, a_t) - \bar{J}_\pi \right| \right) | s_0 = s \right]. \end{aligned}$$

The same hold for the Mean-RMAD action-value function too:

$$\begin{aligned} Q_\pi^\lambda(s, a) :=& Q_\pi(s, a) - \lambda X_\pi(s, a) \\ =& \mathop{\mathbb{E}}_{\substack{s_{t+1} \sim P(\cdot|s_t, a_t) \\ a_{t+1} \sim \pi(\cdot|s_{t+1})}} \left[\sum_{t=0}^{\infty} \gamma^t \left(R(s_t, a_t) - \lambda \left| R(s_t, a_t) - \bar{J}_\pi \right| \right) | s_0 = s, a_0 = a \right]. \end{aligned}$$

Interestingly, this action-value function enjoys a Bellman expectation equation:

$$Q_\pi^\lambda(s, a) = R(s, a) - \lambda \left| R(s, a) - \bar{J}_\pi \right| + \gamma \mathop{\mathbb{E}}_{s' \sim P(\cdot|s,a)} [V_\pi^\lambda(s')].$$

The proof of this result is provided in Appendix G.5. The definition of the advantage functions of the RMAD and of the Mean-RMAD automatically follow, respectively:

$$\begin{aligned} A_\pi^\omega(s, a) &:= -(X_\pi(s, a) - W_\pi(s)), \\ A_\pi^\lambda(s, a) &:= Q_\pi^\lambda(s, a) - V_\pi^\lambda(s) = A_\pi(s, a) + \lambda A_\pi^\omega(s, a), \end{aligned}$$

where we included the minus sign because we prefer lower values of the RMAD. Obtaining a Bellman-like expectation equation is a fundamental step to perform efficient policy evaluation, and provides an important basis to develop policy search techniques. These results are similar to what has been achieved for Mean-Volatility objective by Bisi et al. (2020). For both the latter measure and the Mean-RMAD is not possible, though, to provide also the Bellman optimality equations. Such tools enable the development of value-based optimization approaches in the risk-neutral setting. This is due to the fact that the aforementioned risk-measures are not *time-consistent*, thus, it is not possible to recursively decompose the maximization through timesteps (Ruszczyński, 2010).

4.2. Mean-RMAD Policy Subgradient

Since the development of dynamic programming and value-based methods is prevented from the lack of optimality equations, we resort to produce policy gradient approaches to solve our risk-averse objective, by restricting our attention to parametric policies. The next theorem provide us a fundamental tool for this purpose.

Theorem 4.1 (RMAD Policy Subgradient). *Given an MDP $\mathcal{M}$ and a (differentiable) parametric policy $\pi_{\boldsymbol{\theta}}, \boldsymbol{\theta} \in \Theta$ we have the following result:*

$$\partial_{\boldsymbol{\theta}} \omega_{\boldsymbol{\theta}} \in (1-\gamma) \mathop{\mathbb{E}}_{\tau \sim p_\pi(\cdot)} \left[\sum_{t=0}^{\infty} \gamma^t \nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a_t|s_t) \left(X_\pi(s_t, a_t) - \psi_\pi Q_\pi(s_t, a_t) \right) \right],$$

where ψ_π is defined as:

$$\psi_\pi \coloneqq \mathop{\mathbb{E}}_{\substack{s \sim d_{\mu,\pi}(\cdot) \\ a \sim \pi(\cdot|s)}} \left[\operatorname{sign}(R(s,a) - \bar{J}_\pi) \right], \tag{9}$$

which we call mean sign deviation.

A subgradient for the trade-off measure directly follows from the previous theorem from linearity. The latter can then be employed to obtain a stochastic subgradient method. Necessary conditions for convergence of this kind of algorithms have been analysed in Boyd and Mutapcic (2008). The paper shows that, when optimizing a concave function, if step sizes are square-summable but not summable we have convergence to the optimal point in expectation, in probability and almost sure convergence for a stochastic subgradient ascent method. For a non-concave objective as the Mean-RMAD, this result translate to a guarantee of convergence to a *local* optimal point.

4.3. Safe Improvement Guarantees and Trust-Region Algorithms

Monotonic Performance Improvement. We extend the Performance Difference Lemma of Kakade and Langford (2002) to the Mean-RMAD in order to develop a trust region method (Schulman et al., 2017a).

Lemma 4.1 (Mean-RMAD Performance Difference Lemma). *The difference of the performance in terms of Mean-RMAD between two policies π and $\tilde{\pi}$ is lower bounded by the expected Mean-RMAD advantage minus the absolute value of the expected mean advantage weighted by the risk-aversion factor λ:*

$$\eta^\lambda_{\tilde{\pi}} - \eta^\lambda_\pi \geqslant (1-\gamma) \mathop{\mathbb{E}}_{\tau|\tilde{\pi}} \left[\sum_t \gamma^t A^\lambda_\pi(s_t, a_t) \right] - \lambda(1-\gamma) \left| \mathop{\mathbb{E}}_{\tau|\tilde{\pi}} \left[\sum_t \gamma^t A_\pi(s_t, a_t) \right] \right|.$$

From this lemma we obtain the safe improvement bound in Theorem 4.2.

Theorem 4.2 (Mean-RMAD Safe Improvement Bound). *Consider the following approximation of $\eta^\lambda_{\tilde{\pi}}$, replacing the state-occupancy density of the old policy $d_{\mu,\pi}$:*

$$L^\lambda_\pi(\tilde{\pi}) \coloneqq \eta^\lambda_\pi + \int_{\mathcal{S}} d_{\mu,\pi}(s) \int_{\mathcal{A}} \tilde{\pi}(a|s) A^\lambda_\pi(s,a) \; da \; ds.$$

Then, the performance of $\tilde{\pi}$ can be bounded as follows:

$$\eta^\lambda_{\tilde{\pi}} \geqslant L^\lambda_\pi(\tilde{\pi}) - \frac{4\gamma\epsilon_\lambda}{1-\gamma} \alpha^2_{KL} - \lambda(1-\gamma)M,$$

where:

$$\alpha^2_{KL} = D^{max}_{KL}(\pi, \tilde{\pi}) = \max_s D_{KL}(\pi(\cdot|s), \tilde{\pi}(\cdot|s)),$$

$$\epsilon_\lambda = \max_{s,a} \left| A^\lambda_\pi(s,a) \right|, \quad \epsilon = \max_{s,a} \left| A_\pi(s,a) \right|,$$

$$M \coloneqq |A^{\tilde{\pi}}_\pi| + \frac{4\epsilon\gamma}{(1-\gamma)^2} \alpha^2_{KL},$$

$$A^{\tilde{\pi}}_\pi \coloneqq \mathop{\mathbb{E}}_{\tau|\pi} \left[\sum_{t=0}^{\infty} \gamma^t \mathop{\mathbb{E}}_{a \sim \tilde{\pi}(\cdot, s_t)} [A_\pi(s_t, a)] \right].$$

By optimizing the Safe Improvement Bound we get RMAD-TRPO, described in Algorithm 1.

Algorithm 1 RMAD-TRPO

1: **Input:** initial policy parameter $\boldsymbol{\theta}_0$, batch size N, number of iterations K, discount factor γ.
2: **for** $k = 0, ..., K-1$ **do**
3: Collect N trajectories with $\boldsymbol{\theta}_k$ to obtain dataset D_N.
4: Compute the advantage values $A^{\lambda}_{\boldsymbol{\theta}_k}(s,a)$ and $A^{\boldsymbol{\theta}}_{\boldsymbol{\theta}_k}(s,a)$.
5: Solve the constrained optimization problem:

$$\boldsymbol{\theta}_{k+1} = \arg\max_{\boldsymbol{\theta}\in\Theta} \left\{ L^{\lambda}_{k}(\boldsymbol{\theta}) - \frac{4\gamma\epsilon_{\lambda}}{1-\gamma}\alpha^2_{KL} - \lambda(1-\gamma)M \right\},$$

$$\text{where} \quad L^{\lambda}_{k}(\boldsymbol{\theta}) = \eta^{\lambda}_{\boldsymbol{\theta}_k} + \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi_{\boldsymbol{\theta}_k}} \\ a\sim\pi_{\boldsymbol{\theta}}(\cdot|s)}} A^{\lambda}_{\boldsymbol{\theta}_k}(s,a)$$

$$\text{and} \quad M = |A^{\boldsymbol{\theta}}_{\boldsymbol{\theta}_k}| + \frac{4\epsilon\gamma}{(1-\gamma)^2}\alpha^2_{KL}$$

$$\text{and} \quad A^{\boldsymbol{\theta}}_{\boldsymbol{\theta}_k} = \mathop{\mathbb{E}}_{\tau|\pi_{\boldsymbol{\theta}_k}} \left[\sum_{t=0}^{\infty} \gamma^t \mathop{\mathbb{E}}_{a\sim\pi_{\boldsymbol{\theta}}(\cdot,s_t)} [A_{\boldsymbol{\theta}_k}(s_t,a)] \right]$$

$$\text{and} \quad \epsilon_{\lambda} = \max_{s,a} \left| A^{\lambda}_{\boldsymbol{\theta}_k}(s,a) \right|, \quad \epsilon = \max_{s,a} |A_{\boldsymbol{\theta}_k}(s,a)|$$

$$\text{and} \quad \alpha^2_{KL} = D^{max}_{KL}(\pi_{\boldsymbol{\theta}_k}, \pi_{\boldsymbol{\theta}}) = \max_{s\in\mathcal{S}} D_{KL}(\pi_{\boldsymbol{\theta}_k}(\cdot|s), \pi_{\boldsymbol{\theta}}(\cdot|s)).$$

6: **end for**

The TRPO version for the Mean-RMAD has the same guarantee of monotonic improvement (Corollary 4.1) of the original risk-neutral method.

Corollary 4.1 (Monotonic Improvement of RMAD-TRPO). *By optimizing the Mean-RMAD Safe Improvement Bound of Theorem 4.1 at each iteration k, we obtain a monotonic improvement of the Mean-RMAD:*

$$\eta^{\lambda}_{\pi_{k+1}} \geqslant \eta^{\lambda}_{\pi_k} \quad \forall k \geqslant 0,$$

where $\eta^{\lambda}_{\pi_k}$ is the Mean-RMAD of policy π_k at iteration k.

Practical version of RMAD-TRPO. The practical version of RMAD-TRPO solves a problem that uses a constraint on the Kullback-Leibler divergence instead of a penalty, as it is done in TRPO (Schulman et al., 2017a):

$$\arg\max_{\boldsymbol{\theta}\in\Theta} \left\{ \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot) \\ a\sim\pi_{\boldsymbol{\theta}_k}(\cdot|s)}} \left[\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)} A^{\lambda}_{\boldsymbol{\theta}_k}(s,a) \right] - \lambda \left| \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot) \\ a\sim\pi_{\boldsymbol{\theta}_k}(\cdot|s)}} \left[\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)} A_{\boldsymbol{\theta}_k}(s,a) \right] \right| \right\} \tag{10}$$

$$\text{subject to} \mathop{\mathbb{E}}_{s\sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot)} [D_{KL}(\pi_{\boldsymbol{\theta}_k}(\cdot|s), \pi_{\boldsymbol{\theta}}(\cdot|s))] \leqslant \delta,$$

where D_{KL} is the Kullback-Leibler divergence, the derivation is in Appendix B. Proposition 4.1 provides a subgradient of the objective function of problem 10.

Proposition 4.1. *Given a (differentiable) parametric policy $\pi_{\boldsymbol{\theta}}, \boldsymbol{\theta} \in \Theta$, this subgradient of the Mean-RMAD:*

$$\partial_{\boldsymbol{\theta}}\eta^{\lambda}_{\boldsymbol{\theta}} \ni (1-\gamma) \mathop{\mathbb{E}}_{\tau\sim p_{\pi}(\cdot)} \left[\sum_{t=0}^{\infty} \gamma^t \nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a_t|s_t) \left(Q_{\pi}(s_t,a_t) - \lambda X_{\pi}(s_t,a_t) + \lambda \psi_{\pi} Q_{\pi}(s_t,a_t) \right) \right]$$

is also a subgradient of the objective function of problem 10.
ψ_{π} was defined in equation 9.

Following the practical version of TRPO Schulman et al. (2017a), at each iteration the algorithm does: compute *a search direction*, it does a linear approximation of the objective function using a subgradient which is, as stated in Proposition 4.1, a subgradient of the Mean-RMAD, it does a quadratic approximation of the constraint and it uses the conjugate gradient algorithm; line search in this direction, it ensures that the constraint is satisfied and the objective function improves.

RMAD-PPO. We can create a version of PPO (Schulman et al., 2017b) for the Mean-RMAD objective using the fact that one subgradient with respect to θ of the Mean-RMAD objective is:

$$\partial_{\boldsymbol{\theta}} \eta^{\lambda}_{\boldsymbol{\theta}} \ni (1-\gamma) \mathop{\mathbb{E}}_{\tau \sim p_{\pi_{old}}} \left[\sum_{t=0}^{\infty} \gamma^t \nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a_t|s_t) A^{\lambda,\psi}_{\pi_{old}}(s_t, a_t) \right]$$

and this is the gradient of:

$$(1-\gamma) \mathop{\mathbb{E}}_{\tau \sim p_{\pi_{old}}} \left[\sum_{t=0}^{\infty} \gamma^t \frac{\pi_{\boldsymbol{\theta}}(a_t|s_t)}{\pi_{\boldsymbol{\theta}_{old}}(a_t|s_t)} A^{\lambda,\psi}_{\pi_{old}}(s_t, a_t) \right],$$

where we used the advantage function of the transformed reward $\tilde{R}(s,a) \coloneqq R(s,a) - \lambda \left| R(s,a) - \bar{J}_{\pi} \right| + \lambda \psi_{\pi} R(s,a)$:

$$A^{\lambda,\psi}_{\pi}(s_t, a_t) \coloneqq A^{\lambda}_{\pi}(s_t, a_t) + \lambda \psi_{\pi} A_{\pi}(s_t, a_t) = A_{\pi}(s_t, a_t) + \lambda A^{\omega}_{\pi}(s_t, a_t) + \lambda \psi_{\pi} A_{\pi}(s_t, a_t).$$

So the main objective of RMAD-PPO is:

$$(1-\gamma) \mathop{\mathbb{E}}_{\tau \sim p_{\pi_{old}}} \left[\sum_{t=0}^{\infty} \gamma^t \min \left(r_{\boldsymbol{\theta}}(s_t, a_t) A^{\lambda,\psi}_{\pi_{old}}(s_t, a_t), \operatorname{clip}\left(r_{\boldsymbol{\theta}}(s_t, a_t), 1-\epsilon, 1+\epsilon \right) A^{\lambda,\psi}_{\pi_{old}}(s_t, a_t) \right) \right],$$

where $r_{\boldsymbol{\theta}}(s_t, a_t) = \frac{\pi_{\boldsymbol{\theta}}(a_t|s_t)}{\pi_{\boldsymbol{\theta}_{\text{old}}}(a_t|s_t)}$.
Then, we can add an entropy bonus and a squared-error loss for the value function and we can optimize the objective like in Schulman et al. (2017b).

5. RCVaR Optimization

We will indicate the RCVaR of policy π for risk-aversion level α with η^{α}_{π} and the RVaR with ρ^{α}_{π} [4].

5.1. Decomposition and Optimization via risk-neutral RL methods

In order to optimize the RCVaR we exchange the maximization with respect to the policy with the maximization with respect to ρ:

$$\begin{aligned}
\max_{\pi} \{\eta^{\alpha}_{\pi}\} &= \max_{\pi} \left\{ \max_{\rho} \left\{ \rho - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s \sim d_{\mu,\pi}(\cdot) \\ a \sim \pi(\cdot|s)}} \left[(R(s,a) - \rho)_{-} \right] \right\} \right\} \\
&= \max_{\rho} \left\{ \max_{\pi} \left\{ (1-\gamma) \mathop{\mathbb{E}}_{\substack{s_0 \sim \mu(\cdot), \\ a_t \sim \pi(\cdot|s_t) \\ s_{t+1} \sim P(\cdot|s_t, a_t)}} \left[\sum_{t=0}^{\infty} \gamma^t \left(\rho - \frac{1}{\alpha} (R(s,a) - \rho)_{-} \right) \right] \right\} \right\},
\end{aligned}$$

which can be solved in a block-coordinate fashion. For a fixed ρ, the inner problem is an MDP with the transformed reward $\tilde{R}(s,a) = \rho - \frac{1}{\alpha}(R(s,a) - \rho)_{-}$, that allows to apply any risk-neutral RL method. While for a fixed policy π, the solution of the outer problem is the RVaR (Rockafellar and Uryasev, 2001). Our approach is described in Algorithm 2, it alternates the calculation of the RVaR with the optimization of an MDP. Unfortunately, we have not found a similar decomposition for the Mean-RMAD, but future work may investigate more its properties. Our algorithm is similar to MVPI of Zhang et al. (2020), where they decompose the Mean-Volatility using the Fenchel duality, they exchange the two maximization terms and they apply the block cyclic coordinate ascent method by alternating the calculation of the normalized expected return with the optimization of an MDP. Another similar algorithm is Risk-Averse policy Optimization by State Augmentation (ROSA) of (Bisi, 2022), where the exchange between the maximization with respect to the policy and the maximization with another variable allows to apply any risk-neutral RL algorithm to a sequence of MDP. Its inner optimization problem is not an MDP, so ROSA requires also the augmentation of the state at each iteration, that may cause an increase of the sample complexity, while our method doesn't require it.

[4] We drop the dependence from π whenever it is clear from the context.

Algorithm 2 RCVaR Block cyclic coordinate ascent (RCVaR-BCCA)

1: **Input:** initial policy parameter $\boldsymbol{\theta}_0$, batch size N, number of iterations K, discount factor γ, risk-neutral RL algorithm A (e.g. PPO, TRPO, etc.)).
2: **for** $k = 0, ..., K-1$ **do**
3: Collect a batch $\{\tau_i\}_{i=1}^N$ of N trajectories with $\boldsymbol{\theta}_k$ to obtain dataset D_N.
4: Compute the RVaR:

$$\rho^{\alpha}_{\pi_{\boldsymbol{\theta}_k}} = \arg\max_{\rho\in\mathbb{R}} \left\{ (1-\gamma) \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot)\\ a_t\sim\pi_{\boldsymbol{\theta}_k}(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{\infty} \gamma^t \left(\rho - \frac{1}{\alpha} \left(R(s,a) - \rho \right)_- \right) \right] \right\}.$$

5: Feed $\{\tau_i\}_{i=1}^N$ into A and solve the problem:

$$\boldsymbol{\theta}_{k+1} = \arg\max_{\boldsymbol{\theta}\in\Theta} \left\{ (1-\gamma) \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot)\\ a_t\sim\pi_{\boldsymbol{\theta}}(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{\infty} \gamma^t \left(\rho^{\alpha}_{\pi_{\boldsymbol{\theta}_k}} - \frac{1}{\alpha} \left(R(s,a) - \rho^{\alpha}_{\pi_{\boldsymbol{\theta}_k}} \right)_- \right) \right] \right\}$$

to obtain $\pi_{\boldsymbol{\theta}_{k+1}}$.
6: **end for**

RCVaR-BCCA is characterized by the monotonic improvement guarantee of the RCVaR.

Theorem 5.1 (Monotonic Policy Improvement for block cyclic coordinate ascent)**.** *Following Algorithm 2, the RCVaR grows monotonically:*

$$\eta^{\alpha}_{\pi_{k+1}} \geqslant \eta^{\alpha}_{\pi_k} \quad \forall k \geqslant 0,$$

where $\eta^{\alpha}_{\pi_k}$ is the RCVaR of policy π_k at iteration k.

In the experiments we used RCVaR-BCCA with TRPO and PPO, which we call RCVaR-TRPO, respectively. The reason is that RCVaR-TRPO allows safe updates that reduce the model risk and because TRPO and PPO are able to tackle complex and large-scale control problems (Heess et al., 2017).

5.2. An Alternate Derivation for Trust-Region Approaches

RCVaR-TRPO can be obtained also by exploiting the performance difference lemma for the RCVaR. We just need to define a new advantage function on the transformed reward $\tilde{R}(s,a) \coloneqq -\frac{1}{\alpha}\left(R(s,a)-\rho\right)_-$:

$$A^{\rho}_{\pi}(s,a) \coloneqq Q^{\rho}_{\pi}(s,a) - V^{\rho}_{\pi}(s),$$

where the value function is:

$$V^{\rho}_{\pi}(s) \coloneqq \mathop{\mathbb{E}}_{\substack{a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{\infty} \gamma^t \left(-\frac{1}{\alpha} \left(R(s,a) - \rho \right)_- \right) | s_0 = s \right]$$

and the action value function is:

$$Q^{\rho}_{\pi}(s,a) \coloneqq \mathop{\mathbb{E}}_{\substack{s_{t+1}\sim P(\cdot|s_t,a_t)\\ a_{t+1}\sim\pi(\cdot|s_{t+1})}} \left[\sum_{t=0}^{\infty} \gamma^t \left(-\frac{1}{\alpha} \left(R(s,a) - \rho \right)_- \right) | s_0 = s, a_0 = a \right].$$

Lemma 5.1 (RCVaR Performance Difference Lemma)**.** *The difference of the performance in terms of RCVaR between two policies π and $\tilde{\pi}$ is lower bounded by the expected advantage function calculated on a modified MDP with new reward $\widetilde{R} = -\left(R - \rho^{\alpha}_{\pi}\right)_-$:*

$$\eta^{\alpha}_{\tilde{\pi}} - \eta^{\alpha}_{\pi} \geqslant (1-\gamma) \mathop{\mathbb{E}}_{\tau|\tilde{\pi}} [\sum_t \gamma^t A^{\rho^{\alpha}}_{\pi}(s_t,a_t)],$$

where ρ^{α} is the RVaR ρ^{α}_{π}.

From the performance difference lemma we obtain the following safe improvement bound.

Theorem 5.2 (RCVaR Safe Improvement Bound). *Consider the following approximation of $\eta^{\alpha}_{\widetilde{\pi}}$, replacing the state-occupancy density of the old policy $d_{\mu,\pi}$:*

$$L^{\rho^{\alpha}}_{\pi}(\widetilde{\pi}) \coloneqq \eta^{\alpha}_{\pi} + \frac{1}{\alpha}\int_{\mathcal{S}} d_{\mu,\pi}(s)\int_{\mathcal{A}} \widetilde{\pi}(a|s) A^{\rho^{\alpha}}_{\pi}(s,a)\ da\ ds,$$

where ρ^{α} is the RVaR ρ^{α}_{π}. Then, the performance of $\widetilde{\pi}$ can be bounded as follows:

$$\eta_{\widetilde{\pi}} \geqslant L^{\rho^{\alpha}}_{\pi}(\widetilde{\pi}) - \frac{4\gamma\epsilon}{1-\gamma}\alpha^2_{KL},$$

where:

$$\alpha^2_{KL} = D^{max}_{KL}(\pi,\widetilde{\pi}) = \max_s D_{KL}(\pi(\cdot|s),\widetilde{\pi}(\cdot|s)),$$
$$\epsilon = \max_{s,a}\left|A^{\rho^{\alpha}}_{\pi}(s,a)\right|.$$

Thanks to the safe improvement bound we get RCVaR-TRPO, which is Algorithm 3.

Algorithm 3 RCVaR-TRPO

1: **Input:** initial policy parameter $\boldsymbol{\theta}_0$, batch size N, number of iterations K, discount factor γ.
2: **for** $k = 0, ..., K-1$ **do**
3: Collect N trajectories with $\boldsymbol{\theta}_k$ to obtain dataset D_N.
4: Compute $\rho^{\alpha}_{\boldsymbol{\theta}_k}$ and advantage values $A^{\rho^{\alpha}}_{\boldsymbol{\theta}_k}(s,a)$.
5: Solve the constrained optimization problem:

$$\boldsymbol{\theta}_{k+1} = \arg\max_{\boldsymbol{\theta}\in\Theta}\left\{L^{\rho^{\alpha}}_{k}(\boldsymbol{\theta}) - \frac{4\gamma\epsilon}{1-\gamma}\alpha^2_{KL}\right\},$$
$$\text{where}\quad L^{\rho^{\alpha}}_{k}(\boldsymbol{\theta}) = \eta_{\boldsymbol{\theta}_k} + \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}\\ a\sim\pi_{\boldsymbol{\theta}}(\cdot|s)}} A^{\rho^{\alpha}}_{\boldsymbol{\theta}_k}(s,a)$$
$$\text{and}\quad \epsilon = \max_{s,a}\left|A^{\rho^{\alpha}}_{\boldsymbol{\theta}_k}(s,a)\right|$$
$$\text{and}\quad \alpha^2_{KL} = D^{max}_{KL}(\pi_{\boldsymbol{\theta}_k},\pi_{\boldsymbol{\theta}}).$$

6: **end for**

The algorithm still have the monotonic improvement guarantee of RCVaR.

Corollary 5.1 (Monotonic Improvement of RCVaR-TRPO). *By optimizing the RCVaR Safe Improvement Bound of Theorem 5.2 at each iteration k, we obtain a monotonic improvement of the RCVaR:*

$$\eta^{\alpha}_{\pi_{k+1}} \geqslant \eta^{\alpha}_{\pi_k} \quad \forall k \geqslant 0,$$

where $\eta^{\alpha}_{\pi_k}$ is the RCVaR of policy π_k at iteration k.

Practical version of RCVaR-TRPO. Like in TRPO and RMAD-TRPO, the practical version of RCVaR-TRPO uses a constraint on the Kullback-Leibler divergence:

$$\arg\max_{\boldsymbol{\theta}\in\Theta} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot)\\ a\sim\pi_{\boldsymbol{\theta}_k}(\cdot|s)}}\left[\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)} A^{\rho^{\alpha}}_{\boldsymbol{\theta}_k}(s,a)\right] \tag{11}$$
$$\text{subject to} \mathop{\mathbb{E}}_{s\sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot)}\left[D_{KL}(\pi_{\boldsymbol{\theta}_k}(\cdot|s),\pi_{\boldsymbol{\theta}}(\cdot|s))\right] \leqslant \delta.$$

The derivation is very similar to that of TRPO (Schulman et al., 2017a). This version needs also the gradient of the RCVaR, as states Proposition 5.1.

Proposition 5.1. *Given a (differentiable) parametric policy $\pi_{\boldsymbol{\theta}}, \boldsymbol{\theta}\in\Theta$, this Policy Gradient Theorem version of the gradient of the RCVaR:*

$$\nabla_{\boldsymbol{\theta}}\eta^{\alpha}_{\boldsymbol{\theta}} = \frac{1}{\alpha}(1-\gamma) \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi_{\boldsymbol{\theta}}(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}}\left[(\nabla_{\boldsymbol{\theta}}\log\pi_{\boldsymbol{\theta}}(a_t|s_t))|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k} Q^{\rho^{\alpha}}_{\pi_k}(s_t,a_t)\right],$$

is also the gradient of the objective function of problem 11.

Then the practical algorithm follows the practical version of TRPO, by doing: compute *a search direction*, using the gradient of Proposition 5.1; line search in this direction.

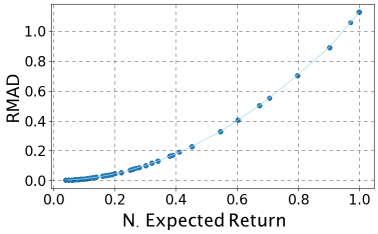
Differences Between RCVaR-BCCA with TRPO and RCVaR-TRPO. Both Algorithm 2 with TRPO and Algorithm 3 have the monotonic improvement of the RCVaR. The only difference is that at each iteration the first algorithm considers the transformed reward $\rho_\pi^\alpha - \frac{1}{\alpha}\left(R(s,a) - \rho_\pi^\alpha\right)_-$ while the second algorithm considers $-\frac{1}{\alpha}\left(R(s,a) - \rho_\pi^\alpha\right)_-$. At a certain iteration ρ_π^α is constant and equal to $RVaR_{\boldsymbol{\theta}_k}^\alpha$, so it doesn't change the solution of the optimization of the mean advantage function. But if we estimate the advantage function with bootstrapping, the values at a certain iteration depend on the values of the previous iteration; so it becomes important to consider the same reward scale at each iteration, which is preserved with $\rho_\pi^\alpha - \frac{1}{\alpha}\left(R(s,a) - \rho_\pi^\alpha\right)_-$ and not with $-\frac{1}{\alpha}\left(R(s,a) - \rho_\pi^\alpha\right)_-$. In fact in a challenging environment like Hopper we have found that using $\rho_\pi^\alpha - \frac{1}{\alpha}\left(R(s,a) - \rho_\pi^\alpha\right)_-$ gives a more stable learning.

6. Experiments

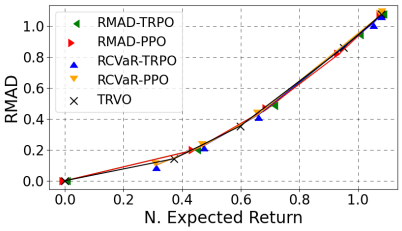
We performed an empirical analysis of the following algorithm, developed in the previous sections: RMAD-TRPO (Algorithm 1), RMAD-PPO (Section 4.3), RCVaR-TRPO (Algorithm 2 with TRPO) and RCVaR-PPO (Algorithm 2 with PPO). We compare these new algorithm with TRVO in terms of convergence speed and quality of the retrieved approximated Pareto frontier. The algorithms were tested on two toy problems: a continuous multi-armed bandit problem and an environment called *Point Reacher* (Bisi, 2022); and on challenging environments: Hopper and Walker from PyBullet (Coumans and Bai, 2016–2021) and a trading environment based on real financial data. The objective of these environments is to show the risk-sensitivity, the trade-off between the normalized expected return and the risk, the speed of convergence and the ability to optimize the considered risk measure of the newly introduced algorithms.
The results are the average of 5 independent runs and in each environment we considered 5 risk-aversion levels for each risk measure. The details about the hyperparameters of the algorithms can be found in the appendix.

6.1. Continuous Multi-Armed Bandit

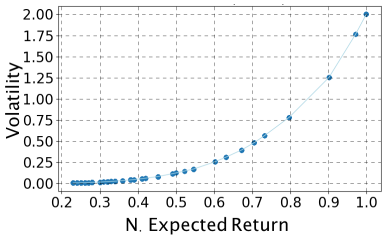
Environment description. Here we consider a multi-armed bandit (MAB) problem with a continuous space of actions $[-1; 1]$. When the agent takes action a it receives a reward distributed as $\mathcal{N}(1 - |a|, ((1 - |a|)^2 + 0.00001)^2 \cdot 2)$. The optimal risk-neutral action is 0 which has the greatest normalized expected return, but in this problem higher normalized expected return means also higher risk. Therefore the agent must trade off between the mean and the risk according to its risk aversion.



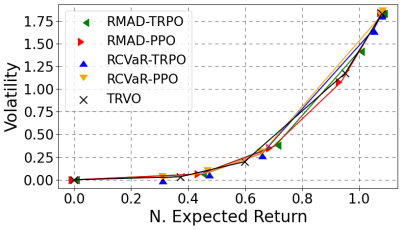
(a) Mean, RMAD of optimal policies



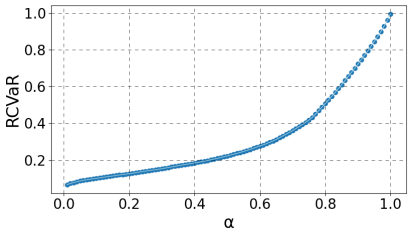
(b) Mean, RMAD of trained policies



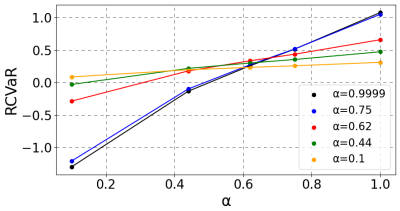
(c) Mean, Volatility of optimal policies



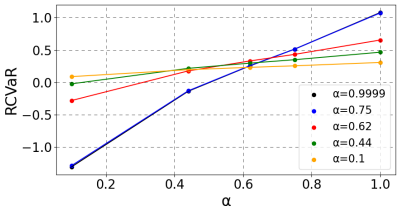
(d) Mean, Volatility of trained policies



(e) Optimal RCVaR



(f) RCVaR values from RCVaR-TRPO



(g) RCVaR values from RCVaR-PPO

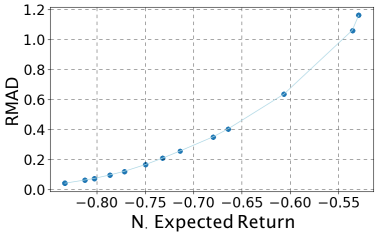
Figure 4: Results obtained on the environment MAB. In Figures 4a and 4c there are the optimal policies for a risk-aversion factor in $[0; 10]$, these are the policies that maximize the Mean-RMAD or the Mean-Volatility objectives. While in Figure 4e, the points represent the optimal policies for the RCVaR with the specified alpha value. The optimal policies were obtained by applying brute force on the policies with actions from 0 to 1 with step 0.01, each evaluated with 100000 steps. Figure 4b shows the trade-off between the normalized expected return and the RMAD of the policies trained with RMAD-TRPO, RMAD-PPO, RCVaR-TRPO, RCVaR-PPO and TRVO, while Figure 4d shows the trade-off between the normalized expected return and the Volatility. The frontiers report the policies obtained with the following risk-aversion levels: λ equal to 0, 0.58, 0.73, 1.2 and 10 for the Mean-RMAD; α equal to 0.9999, 0.75, 0.62, 0.44 and 0.1 for the RCVaR; β equal to 0, 0.25, 0.6, 2 and 10 for the Mean-Volatility. Figures 4f and 4g display the RCVaR for different values of α of the policies trained with RCVaR-TRPO and RCVaR-PPO respectively, which tried to optimize the RCVaR with α indicated in the legend. The lines that connect the points are showed only for readability.

Results. In Figures 4a, 4c and 4e, values of policies with actions from 0 to 1 with step 0.01 are shown, we did not consider the actions from -1 to 0 given that they are equivalent to the previous ones. In Figure 4a we can see the optimal policies for the Mean-RMAD for some risk-aversion factors in the range $[0; 10]$, while in 4c the Mean-Volatility is considered. In Figure 4e optimal values are shown for the RCVaR for α from 0.01 to 0.99 with step 0.01.
The obtained frontiers in Figures 4b and 4d approximate the frontiers composed by the optimal policies in Figures 4a and 4c obtained with the brute force. In Figures 4f and 4g we can see that for $\alpha = 0.1$ the highest RCVaR is achieved by the policy that tried to maximize the RCVaR with $\alpha = 0.1$; while for $\alpha = 0.44$ the best policy is the one that tried to maximize the RCVaR with $\alpha = 0.44$ and so on with the other values of α. Furthermore, the highest values of the RCVaR for each α of Figures 4f and 4g are similar to the optimal values obtained with brute force in Figure 4e. Thus the algorithms have found the optimal policies or policies very similar to the optimal ones.

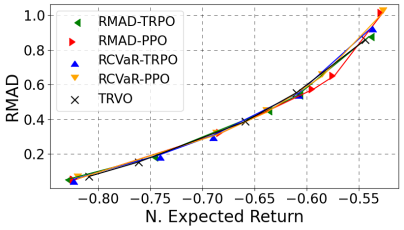
6.2. Noisy Point Reacher

Environment description. We consider a modified version of *Point Reacher* (Bisi, 2022), in which the agent controls a point mass that moves along the real line in order to bring it to a target location in the minimum number of steps. The agent chooses a continuous action in $[-2; 2]$ and the state of the environment is described by the position of the mass in $[-10; 10]$. If the agent takes action a and the state is s, then the new state is

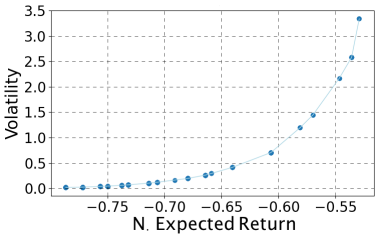
$s' \sim \mathcal{N}(s + a, a^2 + 0.01)$ and the reward is $r = -(0.1|s'| + (a-1)_+^2)$. The goal is to move the point as near as possible to the origin. Each episode has length 10 and the initial state is drawn uniformly in $[-5.1; -5] \cup [5; 5.1]$. We added more noise in order to show the trade-off between the normalized expected return and the risk. If the agent performs action a the new state is $s' \sim \mathcal{N}(s + a, (|a| + 0.01)^{0.26})$ and the reward is distributed as $\mathcal{N}(-(0.1|s'|)^{0.26}, |a|^3)$. The goal is always to move the point to the origin, but the higher the speed the higher the risk.



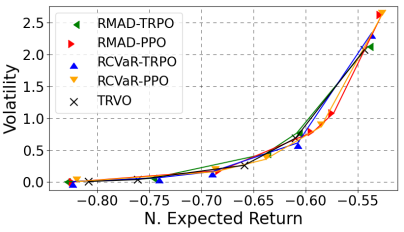
(a) Mean, RMAD of optimal policies



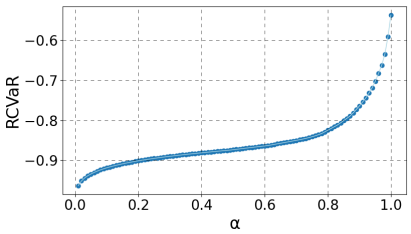
(b) Mean, RMAD of trained policies



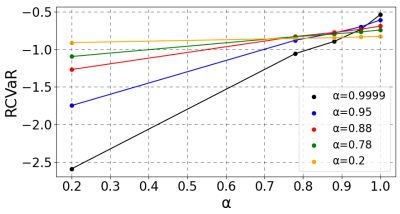
(c) Mean, Volatility of optimal policies



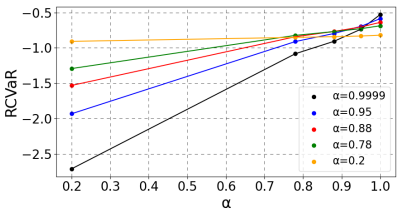
(d) Mean, Volatility of trained policies



(e) Optimal RCVaR



(f) RCVaR values from RCVaR-TRPO

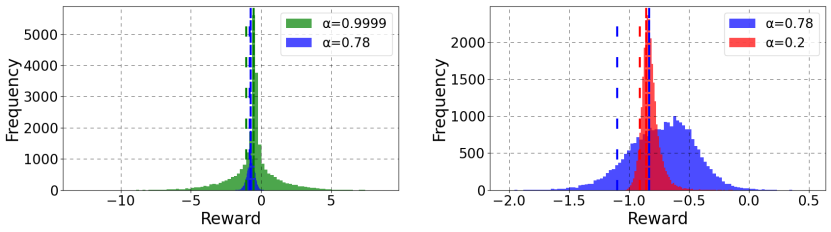


(g) RCVaR values from RCVaR-PPO

Figure 5: Results obtained on the environment Noisy Point Reacher. In Figures 5a and 5c there are the optimal policies for a risk-aversion factor in $[0; 10]$, these are the policies that maximize the Mean-RMAD or the Mean-Volatility objectives. While in Figure 5e, the points represent the optimal policies for the RCVaR with the specified alpha value. The optimal policies were obtained by applying brute force. Figure 5b shows the trade-off between the normalized expected return and the RMAD of the policies trained with RMAD-TRPO, RMAD-PPO, RCVaR-TRPO, RCVaR-PPO and TRVO, while Figure 5d shows the trade-off between the normalized expected return and the Volatility. The frontiers report the policies obtained with the following risk-aversion levels: λ equal to 0, 0.2, 0.26, 0.45 and 1.5 for the Mean-RMAD; α equal to 0.9999, 0.95, 0.88, 0.78 and 0.2 for the RCVaR; λ equal to 0, 0.1, 0.2, 0.8 and 2 for the Mean-Volatility. Figures 5f and 5g display the RCVaR for different values of α of the policies trained with RCVaR-TRPO and RCVaR-PPO respectively, which tried to optimize the RCVaR with α indicated in the legend. The lines that connect the points are showed only for readability.

Results. In Figure 5a, 5c and 5e we considered only the policies that move the point mass directly to the goal with a fixed action (we considered actions from 0 to 2 with step 0.01) or with an action equal to the distance from the goal if the distance is smaller than the fixed action, so we didn't considered the policies that move away the point mass from the target location, because they are not optimal. In Figure 5a there are the optimal policies (obtained with brute force) for the Mean-RMAD for some risk-aversion factors in the range $[0; 10]$, while in 5c it is considered the Mean-Volatility objective. In Figure 5e optimal values are shown for the RCVaR for α from 0.01 to 0.99 with step 0.01. The frontiers in Figures 5b and 5d, coming from the training, approximate the frontiers composed by the optimal policies in Figures 5a and 5c. The results of Figures 5f and 5g come from RCVaR-TRPO and RCVaR-PPO and we can see that for $\alpha = 0.2$ the highest RCVaR is achieved by the policy that tried to maximize the RCVaR with $\alpha = 0.2$; while for $\alpha = 0.78$ the best policy is the one that tried to maximize the RCVaR with $\alpha = 0.78$ and so on with the other values of α. Furthermore, the highest values

of the RCVaR for each α of Figures 5f and 5g are similar to the optimal values obtained with brute force in Figure 5e. These results indicate that the algorithms have found policies that have a performance very similar to the optimal one.



(a) $\alpha = 0.9999$ and $\alpha = 0.78$ (b) $\alpha = 0.78$ and $\alpha = 0.2$

Figure 6: Reward distributions obtained from the policy trained with RCVaR-TRPO that tried to optimize the RCVaR for α indicated in the legend. 25000 rewards were sampled using the trained models on the modified Noisy Point Reacher environment characterized by the transition kernel 12. The dashed lines are the RCVaR values obtained from the policy of the same color during validation. The densely dashed lines are the RCVaR with α equal to the first one in the legend, while the loosely dashed lines refer to the RCVaR with α equal to the second one in the legend. Similar results were obtained with RCVaR-PPO.

Figure 6 shows the reward distributions obtained by sampling with policies trained to optimize RCVaR with different level of α. For this example we used the trained RCVaR-TRPO models on a slightly modified MDP characterized by the following transition kernel (Bisi, 2022):

$$\tilde{P}(\cdot|s,a) = \gamma P(\cdot|s,a) + (1-\gamma)\mu_0(\cdot), \tag{12}$$

where P is the transition kernel of the original environment, in this case Hopper. At each step, the next state is sampled according to the original kernel with probability γ, while it is sampled according to the initial state distribution with probability $(1-\gamma)$. At steady-state, this sampling process causes the faced states to be distributed according to the discounted state occupancy measure (Thomas, 2014). Always in Figure 6 we can see the capability of RCVaR-TRPO (similar results were obtained with RCVaR-PPO) of maximizing the RCVaR by increasing the values of the rewards that are less than the VaR: when we reduce α the RCVaR will take into consideration only the small rewards, in fact in the distribution coming from $\alpha = 0.999$ the rewards span from -5, while in the distribution coming from $\alpha = 0.999$ the rewards span from -1.5 and in the last distribution from -1.

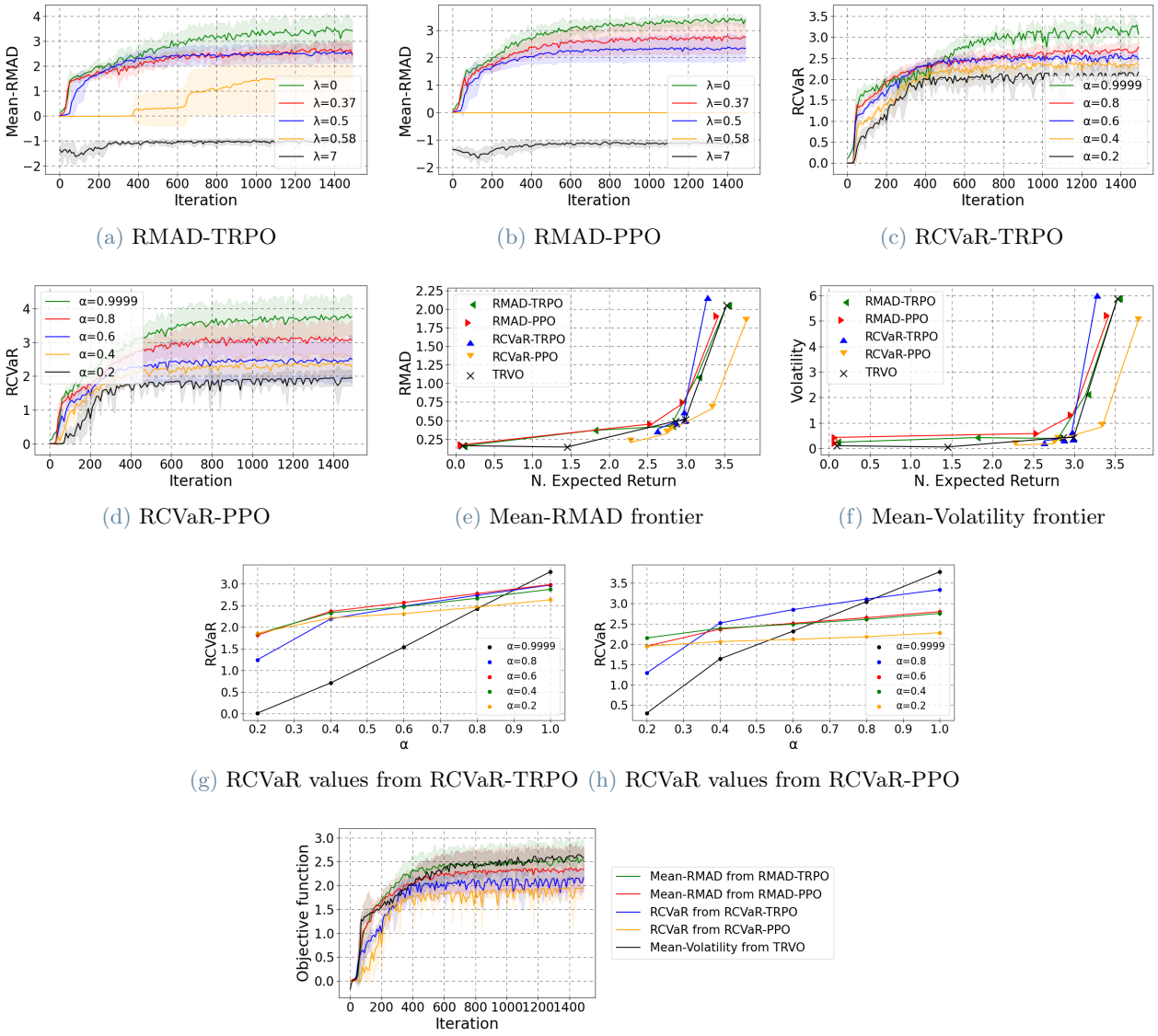
6.3. Robotic Locomotion

Environments description. We considered two challenging environments in the robotic setting: Hopper and Walker. We used the implementation of PyBullet (Coumans and Bai, 2016–2021) and we set the maximum length of each episode to 500. The state of the robot is made up of its position, its speed and the time remaining in the episode, while the actions consists of torques applicable to various joints. The state space and the action space are continuous and high-dimensional. The reward is equal to a linear combination of: a bonus for being alive, a bonus for its distance from the initial position, a cost for large actions in absolute value and a cost if the joints of the robot are at their limit. The dynamics is deterministic so to obtain a sensible environment for risk-averse optimization we added a Gaussian noise to the actions with zero mean and variance the action to the power of 6 and we added another noise to the reward with zero mean and range in $[-0.5; +0.5]$. In particular, the reward noise can assume only two values x and $-x$, each with probability equal to 0.5, where the value x is proportional to the action to the power of 6 and can be only in the range $[-0.5; +0.5]$. These noises model the fact that high torques have more unpredictable effects on the state of the system.
Only in Hopper, we modified the reward function in order to show the monotonicity property of the Mean-RMAD and of the RCVaR: the electricity cost was changed from -2 to -1.5, the stall torque cost from -0.1 to -0.05, the alive reward was set to $+2$, we multiplied the progress reward by a factor of 1.5 and the resulting reward is clipped to be always greater than or equal to 0.5. In this way if we consider also the noise on the reward that is between -0.5 and 0.5, we obtain that the reward of the environment is always greater than or equal to zero and we can exploit the monotonicity property in order to avoid the policy that commits suicide, in which the robot falls and gets zero until the end of the episode. So if we use the RCVaR or the Mean-RMAD with $0 \leqslant \lambda \leqslant 0.5$, we have the guarantee that the optimal policy is not the one that commits suicide because all other policies give a reward that is always greater or equal to zero. While if we use for example the Mean-Volatility we don't have this guarantee, it can happen that for a certain λ the optimal policy is the one that

kills itself, because it has a very low Volatility.
In Hopper we used curriculum learning for RCVaR-TRPO and RCVaR-PPO by starting with $\alpha = 1$ and by reducing it in an exponential way to the considered $\bar{\alpha}$:

$$\alpha_k = 0.5^{\frac{k}{\tau}}(1 - \bar{\alpha}) + \bar{\alpha},$$

where k is the current iteration number, τ is a time constant, $\bar{\alpha}$ is the α value of the RCVaR which we want to optimize and α_k is the α value considered at iteration k. We used curriculum learning because in this environment at the beginning of the learning the agent goes frequently to an absorbing state that gives always the lowest reward which is zero, almost filling the batch with these rewards. So it happens that for small values of α the VaR is equal to the lowest possible reward and it causes the gradient of the RCVaR to become 0. We used curriculum learning also in Walker, but there the absorbing states do not give the lowest possible reward, so probably RCVaR-TRPO and RCVaR-PPO work also without curriculum learning.



(a) RMAD-TRPO (b) RMAD-PPO (c) RCVaR-TRPO

(d) RCVaR-PPO (e) Mean-RMAD frontier (f) Mean-Volatility frontier

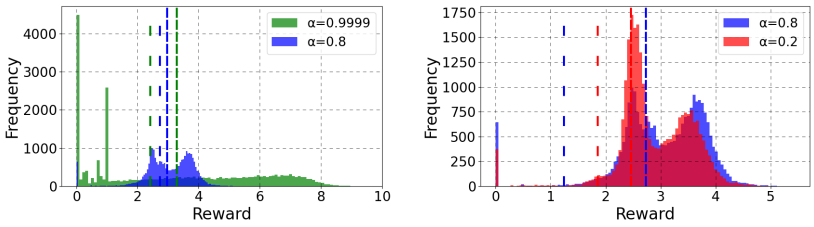
(g) RCVaR values from RCVaR-TRPO (h) RCVaR values from RCVaR-PPO

(i) Algorithms comparison

Figure 7: Results obtained on the environment Hopper, with shaded area representing the standard deviation, while the solid lines represent the mean. The first four figures display the learning curves of the algorithms: RMAD-TRPO, RMAD-PPO, RCVaR-TRPO and RCVaR-PPO respectively, for different risk-aversion levels. Figure 7e shows the trade-off between the normalized expected return and the RMAD of the policies trained with the previous algorithms, while and 7f shows the trade-off between the normalized expected return and the Volatility. The frontiers report the policies obtained from the previous learning curves with that algorithms and risk-aversion levels. Figures 7g and 7h display the RCVaR for different values of α of the policies trained with RCVaR-TRPO and RCVaR-PPO respectively, which tried to optimize the RCVaR with α indicated in the legend. Figure 7i reports as comparison the learning curves of the newly introduced algorithms and of the baseline TRVO, we considered the Mean-RMAD with $\lambda = 0.5$ for RMAD-TRPO and RMAD-PPO, the RCVaR with $\alpha = 0.2$ for RCVaR-TRPO and RCVaR-PPO and the Mean-Volatility with $\beta = 0.5$ for TRVO. The lines that connect the points of Figures 7e, 7f, 7g and 7h are showed only for readability.

Hopper results. In Figure 7 are reported: the learning curves of RMAD-TRPO, RMAD-PPO, RCVaR-TRPO and RCVaR-PPO; the trade-offs between the normalized expected return and the RMAD and the Volatility; the RCVaR values obtained from the RCVaR-TRPO and RCVaR-PPO models for different values of α; a comparison with TRVO. From the learning curves we can see that all algorithms achieved convergence for different risk-aversion levels. In the frontiers of Figures 7e and 7f we can see that the algorithms have found policies that can trade-off between the mean and the risk: we have risk-neutral policies with high normalized expected return and high RMAD and Volatility; policies that have low normalized expected return and low RMAD and Volatility; other policies that can obtain a good mean even if not high with a quite low RMAD and Volatility. This is a very noisy environment so it can happen that the policy which tried to optimize a certain $\bar{\alpha}$ is not better than all other policies for the RCVaR with $\alpha = \bar{\alpha}$, as it is in Figures 7g and 7h, but it is still a good policy in the sense that it beats some other policies for some values of α: for example the policy that generated

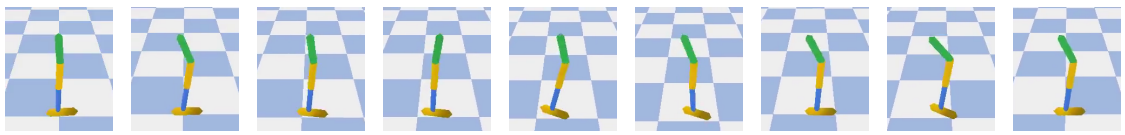
the green dots in Figure 7g is not the best one for any value of α but it is never the worst and for α equal to 0.2, 0.4, 0.6 and 0.8 it is very near to the highest values of the RCVaR. Thanks to the monotonicity property, we obtained that for all values of α that we used (0.9999, 0.8, 0.6, 0.4 and 0.2) and for $0 \leqslant \lambda \leqslant 0.5$ (we used 0, 0.37 and 0.5) the found policies with RCVaR-TRPO, RCVaR-PPO, RMAD-TRPO and RMAD-PPO did never commit suicide. Finally, in Figure 7i we can see that the algorithms have achieved convergence almost at the same time.



(a) $\alpha = 0.9999$ and $\alpha = 0.8$ (b) $\alpha = 0.8$ and $\alpha = 0.2$

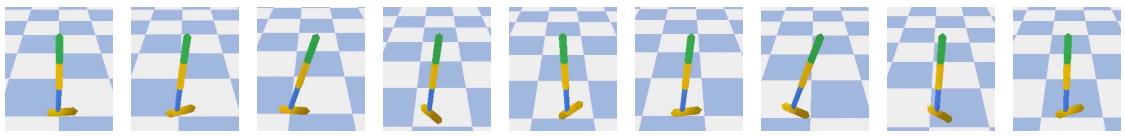
Figure 8: Reward distributions obtained from the policy trained with RCVaR-TRPO that tried to optimize the RCVaR for α indicated in the legend. 25000 rewards were sampled using the trained models on the modified Hopper environment characterized by the transition kernel 12. The dashed lines are the RCVaR values obtained from the policy of the same color during validation. The densely dashed lines are the RCVaR with α equal to the first one in the legend, while the loosely dashed lines refer to the RCVaR with α equal to the second one in the legend. Similar results were obtained with RCVaR-PPO.

Figure 8 shows the reward distributions obtained by sampling from policies trained to optimize RCVaR with different level of α; for small values of α the model avoid to have small rewards, by concentrating the distribution to higher values, but this causes also to have less big rewards. The peaks in zero are due to the falls of the robot, in fact if we use a less risk-averse policy we obtain more falls, while the policies that are more risk-averse fall less times.



(a) (b) (c) (d) (e) (f) (g) (h) (i)

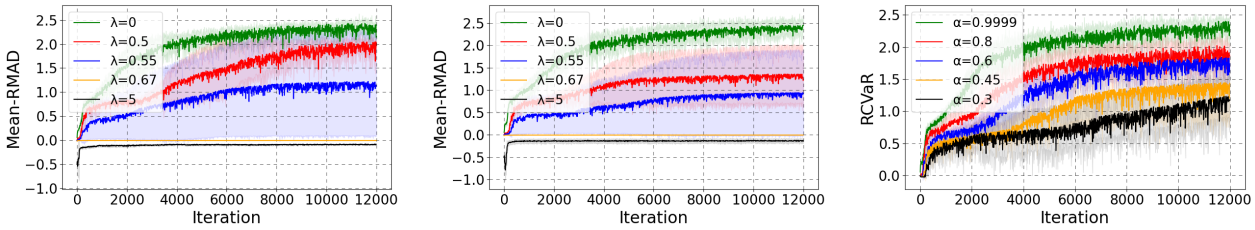
Figure 9: Frames from the Hopper environment obtained using the risk-neutral policy trained with RMAD-TRPO for risk-aversion factor 0.



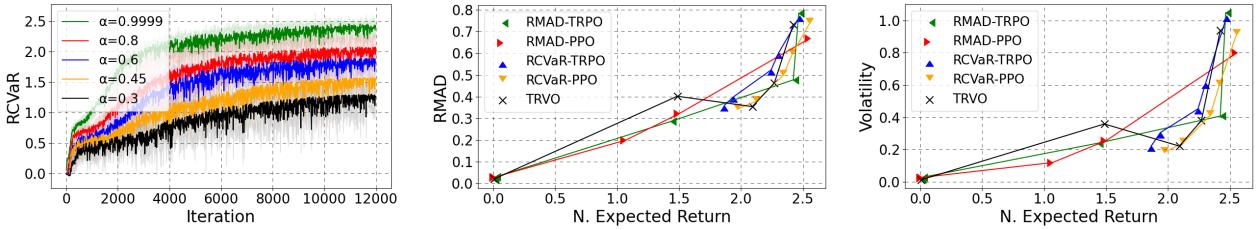
(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 10: Frames from the Hopper environment obtained using the risk-averse policy trained with RMAD-TRPO for risk-aversion factor 0.5.

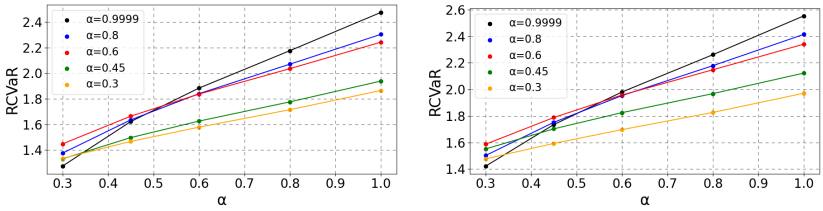
Figure 9 shows some frames of the policy risk-neutral obtained from the learning. The policy risk-neutral moves by oscillating the top part of the robot, in this way it can achieve a high normalized expected return but there is a higher risk of falling. While in Figure 10 the frames of a policy risk-averse show a more cautious behaviour that maintains fixed the top part of the robot, allowing to move forward with less risk of falling. The risk-neutral behavior of Figure 9 is observed also with the following combination of algorithms and risk-aversion levels: RMAD-PPO for risk-aversion factor 0, RCVaR-TRPO with alpha 0.999, RCVaR-PPO with alpha 0.999 and TRVO for risk-aversion factor 0. While the risk-averse behavior of Figure 10 is observed also with the following combination of algorithms and risk-aversion levels: RMAD-PPO for risk-aversion factor 0.5, RCVaR-TRPO with alpha 0.2, RCVaR-PPO with alpha 0.2 and TRVO for risk-aversion factor 0.5. These policies come from one run, but the other four runs gave similar policies.



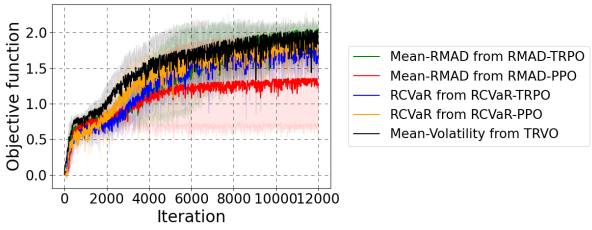
(a) RMAD-TRPO (b) RMAD-PPO (c) RCVaR-TRPO



(d) RCVaR-PPO (e) Mean-RMAD frontier (f) Mean-Volatility frontier



(g) RCVaR values from RCVaR-TRPO (h) RCVaR values from RCVaR-PPO



(i) Algorithms comparison

Figure 11: Results obtained on the environment Walker, with shaded area representing the standard deviation, while the solid lines represent the mean. The first five figures display the learning curves of the algorithms: RMAD-TRPO, RMAD-PPO, RCVaR-TRPO and RCVaR-PPO respectively, for different risk-aversion levels. Figure 11e shows the trade-off between the normalized expected return and the RMAD of the policies trained with the previous algorithms, while and 11f shows the trade-off between the normalized expected return and the Volatility. The frontiers report the policies obtained from the previous learning curves with that algorithms and risk-aversion levels. Figures 11g and 11h display the RCVaR for different values of α of the policies trained with RCVaR-TRPO and RCVaR-PPO respectively, which tried to optimize the RCVaR with α indicated in the legend. Figure 11i reports as comparison the learning curves of the newly introduced algorithms and of the baseline TRVO, we considered the Mean-RMAD with $\lambda = 0.5$ for RMAD-TRPO and RMAD-PPO, the RCVaR with $\alpha = 0.6$ for RCVaR-TRPO and RCVaR-PPO and the Mean-Volatility with $\beta = 0.5$ for TRVO. The lines that connect the points of Figures 11e, 11f, 11g and 11h are showed only for readability.

Walker results. In Figure 11 are reported: the learning curves of RMAD-TRPO, RMAD-PPO, RCVaR-TRPO and RCVaR-PPO; the trade-offs between the normalized expected return and the RMAD and the Volatility; the RCVaR values obtained from the RCVaR-TRPO and RCVaR-PPO models for different values of α; a comparison with TRVO. Also in Walker, almost all algorithms achieved convergence as shown in Figures 11a-11d. In Figures 11e and 11f we obtained similar frontiers to that coming from TRVO; RCVaR-TRPO and RCVaR-PPO need just a smaller value of α in order to find policies with less MAD and less Volatility. The frontiers show a variety of policies that found a trade-off between the normalized expected return and the RMAD and the Volatility. In Figure 11i, the various methods have comparable learning curves, reaching convergence almost at the same time.

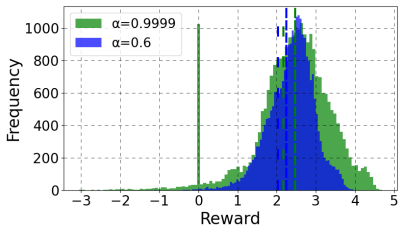
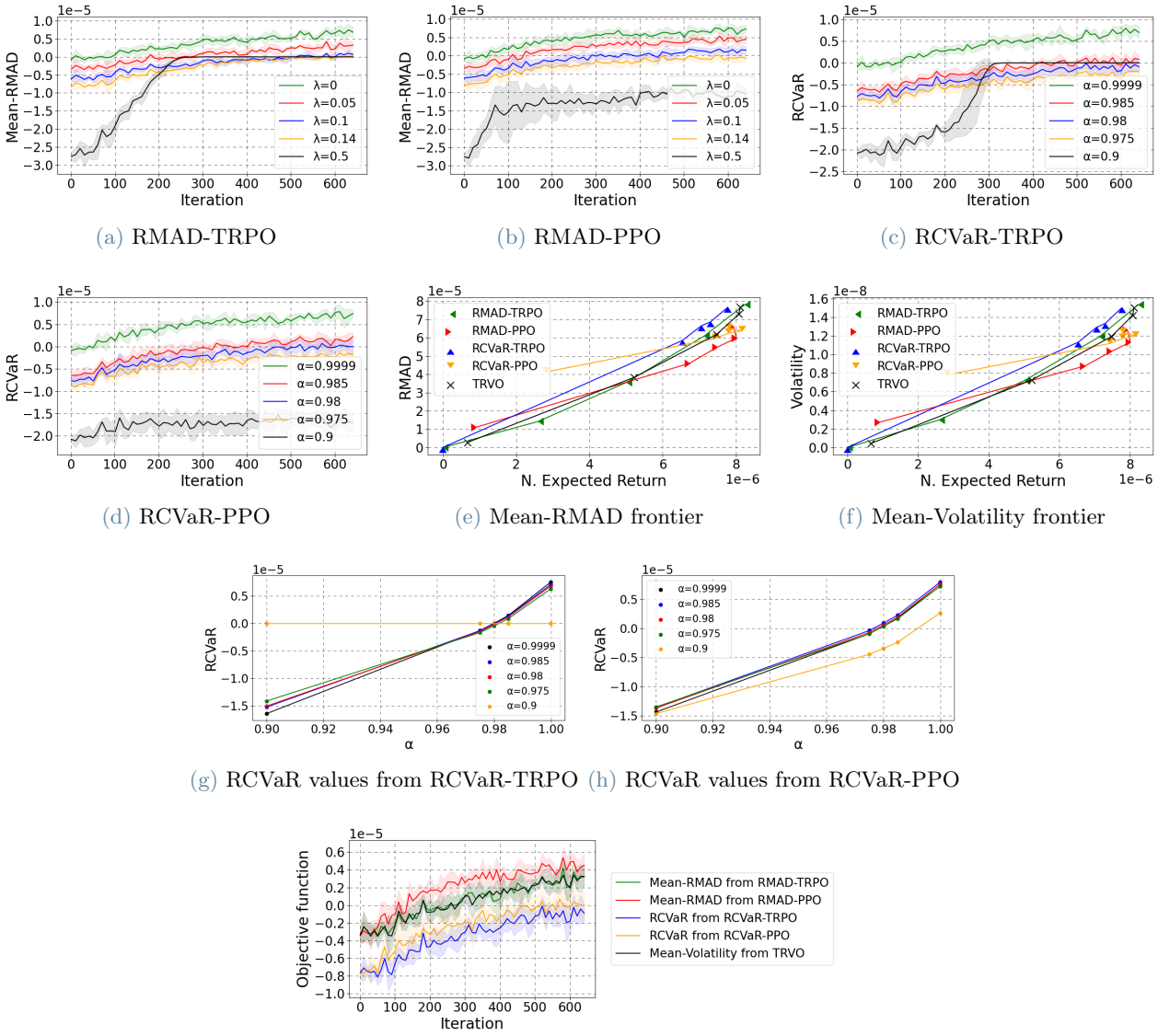


Figure 12: Reward distributions obtained from the policy trained with RCVaR-TRPO that tried to optimize the RCVaR for α indicated in the legend. 25000 rewards were sampled using the trained models on the modified Walker environment characterized by the transition kernel 12. The dashed lines are the RCVaR values obtained from the policy of the same color during validation. The densely dashed lines are the RCVaR with α equal to the first one in the legend, while the loosely dashed lines refer to the RCVaR with α equal to the second one in the legend. Similar results were obtained with RCVaR-PPO.

Figure 12 shows the reward distributions obtained by sampling from policies trained to optimize RCVaR with different level of α. With $\alpha = 0.6$, RCVaR-TRPO found more conservative policies that give rewards in a smaller but positive range. The risk-averse policy has in addition no peak in zero, which means that the robot falls almost never.

6.4. Trading

The environment consists in a simulated trading task on the Foreign Exchange (FOREX) market. The agent can trade a fixed amount of dollars, based on exchange rate prices taken from real 2017 open data. An episode includes a day of trading from 01:00 to 21:29 with a step of one minute. There are three possible actions: short position (0), flat position (1) and long position (2). When the agent does action a_t, the reward is equal to $r_t = a_t(p_t - p_{t-1}) - c|a_t - a_{t-1}|$, where p_t is the price at time step t and c is a fee equal to 10^{-6}. In each episode the agent start from a random day. The prices in a day are 1230, but the episode has length equal to 1169, because the agent has access to the last 60 differences of price between a time step and its previous one. The state includes also the time remaining in the episode and the last taken action.



(a) RMAD-TRPO (b) RMAD-PPO (c) RCVaR-TRPO

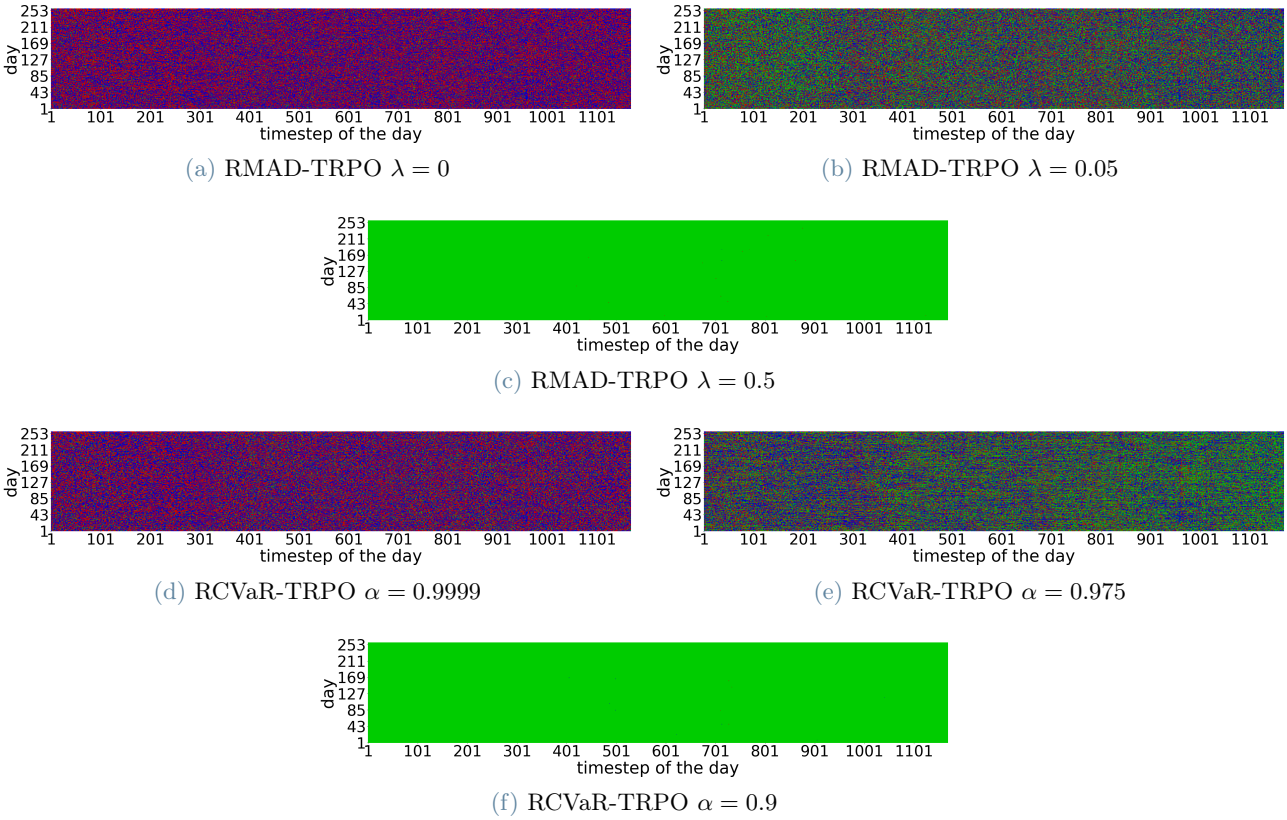
(d) RCVaR-PPO (e) Mean-RMAD frontier (f) Mean-Volatility frontier

(g) RCVaR values from RCVaR-TRPO (h) RCVaR values from RCVaR-PPO

(i) Algorithms comparison

Figure 13: Results obtained on the environment Trading, with shaded area representing the standard deviation, while the solid lines represent the mean. The first five figures display the learning curves of the algorithms: RMAD-TRPO, RMAD-PPO, RCVaR-TRPO and RCVaR-PPO respectively, for different risk-aversion levels. Figure 13e shows the trade-off between the normalized expected return and the RMAD of the policies trained with the previous algorithms, while and 13f shows the trade-off between the normalized expected return and the Volatility. Figures 13g and 13h display the RCVaR for different values of α of the policies trained with RCVaR-TRPO and RCVaR-PPO respectively, which tried to optimize the RCVaR with α indicated in the legend. Figure 13i reports as comparison the learning curves of the newly introduced algorithms and of the baseline TRVO, we considered the Mean-RMAD with $\lambda = 0.05$ for RMAD-TRPO and RMAD-PPO, the RCVaR with $\alpha = 0.98$ for RCVaR-TRPO and RCVaR-PPO and the Mean-Volatility with $\beta = 250$ for TRVO. The lines that connect the points of Figures 13e, 13f, 13g and 13h are showed only for readability.

Figures 13a-13d display the learning curves. The training was stop after 6 million steps because from that point the algorithms start to overfit, this allowed to have similar performance in the training set and in the validation set. The validation was done on three days per month taken randomly, while the other days were used for training. From Figures 13e and 13f we can see the good trade-off between the normalized expected return and the RMAD and the Volatility obtained by RMAD-TRPO, RMAD-PPO and TRVO. While RCVaR-TRPO and RCVaR-PPO found essentially only two types of policies: one aggressive risk-neutral policy and one conservative risk-averse policy, as it can be seen also in Figures 13g and 13h. In Figure 13i we can see that the various methods have a similar growth, meaning that they have a comparable sample efficiency. They achieve different values only because we are showing different risk measures according to the algorithm, for example for RMAD-TRPO we are showing the learning curve in terms of the Mean-RMAD, while for RCVaR-TRPO we are showing the RCVaR.



(a) RMAD-TRPO $\lambda = 0$

(b) RMAD-TRPO $\lambda = 0.05$

(c) RMAD-TRPO $\lambda = 0.5$

(d) RCVaR-TRPO $\alpha = 0.9999$

(e) RCVaR-TRPO $\alpha = 0.975$

(f) RCVaR-TRPO $\alpha = 0.9$

Figure 14: Actions selected during the year on the Trading Environment by the policy trained with the corresponding algorithm and risk-aversion level indicated under each figure. On the x axis there are the timesteps of the day and on the y axis there are the days of the year. The green dots indicate the flat action, the red dots are the short position, while the blue dots are the long position.

In Figure 14 we can see the actions selected by the policies trained with RMAD-TRPO and RCVaR-TRPO for different risk-aversion levels. The figures with a lot of red and blue dots represent aggressive and risk-neutral policies, while if there are a lot of green dots the figure represents a risk-averse policy. For each algorithm, the risk-averse policies tend to choose the flat action more times than what the risk-neutral policy does, until the most risk-averse policy does nothing and exits from the market. In fact, in Figures 14a and 14d we can see that the risk neutral policies trade a lot on the market. While in Figures 14b and 14e the policies are more risk-averse and sometimes choose the flat action. The most risk-averse policies don't want any risk so they exit from the market by doing nothing, as it can be seen in Figures 14c and 14f. These results indicate that the algorithms are risk sensitive, they are able to find aggressive policies that maximize the mean, policies that give a good mean but with lower risk depending on the risk aversion level and policies that don't want any risk at all. Similar results were obtained also with RMAD-PPO, RCVaR-PPO and TRVO.

7. Related work

We have found risk measures that are coherent, reward-based and that can be optimized with efficient algorithms with same training and convergence properties of state-of-the-art risk-neutral methods, guaranteeing also the monotonic improvement of the performance which allows safe updates. In the literature, there are several risk measures based on the return: CVaR (Tamar et al., 2015b; Chow et al., 2015) (Bäuerle and Ott, 2011), variance-related measures (Di Castro et al., 2012; Tamar and Mannor, 2013) (Sobel, 1982), utility function (Shen et al., 2014), entropic risk measure (Nass et al., 2019). The first work about a reward-based risk measure, the Mean-Volatility, is Bisi et al. (2020). It is a measure similar to the Mean-RMAD but it is not coherent. Shapiro et al. (2021) illustrates risk-averse optimization and analyzes the coherence properties of several risk measures: utility model, CVaR, VaR, entropic risk measure, mean-variance, mean-deviation and mean-semideviation.
The paper Zhang et al. (2020) showed that any RL algorithm can be used to optimize the Mean-Volatility, by decomposing the problem in a double optimization problem and using block cyclic coordinate ascent like we have done in Section 5.1. In Bisi (2022) the author develops the algorithm ROSA, in which you can use any RL method to optimize some return-based risk measures: mean-variance, mean-volatility, CVaR, utility function and entropic risk measure, but at the cost of a greater number of samples due to the augmentation of the state

space, while the RCVaR doesn't need it thanks to its reward-based nature. An algorithm similar to RMAD-TRPO and RCVaR-TRPO is TRVO (Bisi et al., 2020), that merged together two streams: risk-averse objective functions and safe policy updates. The first one reduces the inherent risk, that comes from the stochastic nature of the environment; the second one reduces the model risk, related to the imperfect knowledge of the environment. RMAD-TRPO and RCVaR-TRPO add also the coherence, which provides rational solutions.
In Tamar et al. (2015a) the authors develop policy gradient for coherent risk measures, but it doesn't provide safe updates. The Performance Difference Lemma was firstly introduced in Kakade and Langford (2002), which allows to create the famous risk-neutral algorithm TRPO of Schulman et al. (2017a), that provides safe updates. An approximation of TRPO is Proximal Policy Optimization (PPO) of Schulman et al. (2017b), both algorithms are used to deal with complex control problems. The paper Boyd and Mutapcic (2008) analyses the subgradient method which can be used to optimize the Safe Improvement Bound of the Mean-RMAD.

8. Conclusions

In this work, we tackled the problem of trading-off the mean and the risk with two new risk measures that are both coherent and based on the reward, which allow to smooth the trajectories, they avoid large deviations in the reward and they allow to obtain solutions that are rational, for example they avoid to choose a policy that gives always the lowest possible reward, they encourage diversification and they allow to translate or to scale the reward without changing the optimal policies. These measures bound the corresponding return-based measures and have interesting relations with other risk measures based on the reward and on the return. We obtained trust-region algorithms for both measures that guarantee safe improvement updates. For the RCVaR we can also apply any classic risk-neutral Reinforcement Learning algorithm maintaining the monotonic improvement of the performance. We showed the risk-sensitivity, the sample efficiency and the results of these algorithms on some environments for which we can find the optimal policy with brute-force and on more challenging environments like Hopper, Walker and Trading, where we obtained similar convergence speed to that of TRVO. Future studies can further develop the theoretical part of these algorithms: whether they can achieve the global optimal policy under certain assumptions, whether there are conditions that allow to get an epsilon optimal policy and the convergence rate to a local or to a global optimal policy. The paper Neu et al. (2017) shows that the policy update of the exact version of TRPO can be expressed in closed form, thus it is equivalent to the MDP-E algorithm of Even-Dar et al. (2009) implying that TRPO converges to the optimal policy in the stationary setting and in environments with finite action space and finite state space. Shani et al. (2019) shows the convergence rate to the global optimum of the sample-based version of TRPO. To conclude, we obtained methods that share the convergence speed of state-of-the-art risk-neutral algorithms while taking into consideration the risk and having the guarantee that the found solutions have good properties thanks to the coherence of the considered risk measures.

References

P. Artzner, F. Delbaen, J.-M. Eber, and D. Heath. Coherent measures of risk. *Mathematical finance*, 9(3): 203–228, 1999. Publisher: Wiley Online Library.

R. Bellman. The theory of dynamic programming. *Bulletin of the American Mathematical Society*, 60(6): 503–515, 1954.

L. Bisi. *Algorithms for risk-averse reinforcement learning.* PhD thesis, Politecnico di Milano, 2022.

L. Bisi, L. Sabbioni, E. Vittori, M. Papini, and M. Restelli. Risk-Averse Trust Region Optimization for Reward-Volatility Reduction. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 4583–4589, 2020.

S. Boyd and A. Mutapcic. Stochastic subgradient methods. *Lecture Notes for EE364b, Stanford University*, 2008.

N. Bäuerle and J. Ott. Markov decision processes with average-value-at-risk criteria. *Mathematical Methods of Operations Research*, 74:361–379, 12 2011. doi: 10.1007/s00186-011-0367-0.

Y. Chow, A. Tamar, S. Mannor, and M. Pavone. Risk-sensitive and robust decision-making: a cvar optimization approach. *CoRR*, abs/1506.02188, 2015. URL `http://arxiv.org/abs/1506.02188`.

Y. Chow, M. Ghavamzadeh, L. Janson, and M. Pavone. Risk-constrained reinforcement learning with percentile risk criteria. *JMLR*, 18(1):6070–6120, 2017. Publisher: JMLR. org.

E. Coumans and Y. Bai. Pybullet, a python module for physics simulation for games, robotics and machine learning. http://pybullet.org, 2016–2021.

D. Di Castro, A. Tamar, and S. Mannor. Policy gradients with variance related risk criteria, 2012. URL https://arxiv.org/abs/1206.6404.

H. Dong and M. K. Nakayama. A tutorial on quantile estimation via monte carlo. In *International Conference on Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing*, pages 3–30. Springer, 2018.

P. D'Oro, A. M. Metelli, A. Tirinzoni, M. Papini, and M. Restelli. Gradient-aware model-based policy search. *CoRR*, abs/1909.04115, 2019. URL http://arxiv.org/abs/1909.04115.

E. Even-Dar, S. M. Kakade, and Y. Mansour. Online markov decision processes. *Mathematics of Operations Research*, 34(3):726–736, 2009. ISSN 0364765X, 15265471. URL http://www.jstor.org/stable/40538442.

J. García and F. Fernández. A comprehensive survey on safe reinforcement learning. *J. Mach. Learn. Res.*, 16: 1437–1480, 2015.

T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. *CoRR*, abs/1801.01290, 2018. URL http://arxiv.org/abs/1801.01290.

N. Heess, D. TB, S. Sriram, J. Lemmon, J. Merel, G. Wayne, Y. Tassa, T. Erez, Z. Wang, S. M. A. Eslami, M. A. Riedmiller, and D. Silver. Emergence of Locomotion Behaviours in Rich Environments. *CoRR*, abs/1707.02286, 2017.

A. Hill, A. Raffin, M. Ernestus, A. Gleave, A. Kanervisto, R. Traore, P. Dhariwal, C. Hesse, O. Klimov, A. Nichol, M. Plappert, A. Radford, J. Schulman, S. Sidor, and Y. Wu. Stable baselines. https://github.com/hill-a/stable-baselines, 2018.

R. A. Howard and J. E. Matheson. Risk-sensitive Markov decision processes. *Management science*, 18(7): 356–369, 1972. Publisher: INFORMS.

S. Kakade and J. Langford. Approximately optimal approximate reinforcement learning. In *IN PROC. 19TH INTERNATIONAL CONFERENCE ON MACHINE LEARNING*, pages 267–274, 2002.

H. Markowitz. Portfolio Selection. *The Journal of Finance*, 7(1):77–91, 1952.

K. Mason and S. Grijalva. A review of reinforcement learning for autonomous building energy management. *Computers & Electrical Engineering*, 78:300–312, 2019. ISSN 0045-7906. doi: https://doi.org/10.1016/j.compeleceng.2019.07.019. URL https://www.sciencedirect.com/science/article/pii/S0045790618333421.

V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. A. Riedmiller. Playing atari with deep reinforcement learning. *CoRR*, abs/1312.5602, 2013. URL http://arxiv.org/abs/1312.5602.

D. Nass, B. Belousov, and J. Peters. Entropic risk measure in policy search. *CoRR*, abs/1906.09090, 2019. URL http://arxiv.org/abs/1906.09090.

G. Neu, A. Jonsson, and V. Gómez. A unified view of entropy-regularized markov decision processes, 2017. URL https://arxiv.org/abs/1705.07798.

M. Papini. *Safe Policy Optimization*. PhD thesis, Politecnico di Milano, 2021.

M. Papini, M. Pirotta, and M. Restelli. Smoothing policies and safe policy gradients, 2019.

M. L. Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

R. T. Rockafellar and S. P. Uryasev. Conditional value-at-risk for general loss distributions. *Corporate Finance and Organizations eJournal*, 2001.

R. T. Rockafellar, S. Uryasev, and others. Optimization of conditional value-at-risk. *Journal of risk*, 2:21–42, 2000.

A. Ruszczyński. Risk-averse dynamic programming for Markov decision processes. *Mathematical programming*, 125(2):235–261, 2010. Publisher: Springer.

J. Schulman, S. Levine, P. Moritz, M. I. Jordan, and P. Abbeel. Trust region policy optimization, 2017a.

J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347, 2017b. URL http://arxiv.org/abs/1707.06347.

L. Shani, Y. Efroni, and S. Mannor. Adaptive trust region policy optimization: Global convergence and faster rates for regularized mdps. *CoRR*, abs/1909.02769, 2019. URL http://arxiv.org/abs/1909.02769.

A. Shapiro, D. Dentcheva, and A. Ruszczynski. *Lectures on stochastic programming: modeling and theory*. SIAM, 2021.

Y. Shen, R. Huang, C. Yan, and K. Obermayer. Risk-averse reinforcement learning for algorithmic trading. In *2014 IEEE Conference on Computational Intelligence for Financial Engineering Economics (CIFEr)*, pages 391–398, 2014. doi: 10.1109/CIFEr.2014.6924100.

M. J. Sobel. The variance of discounted markov decision processes. *Journal of Applied Probability*, 19(4): 794–802, 1982. ISSN 00219002. URL http://www.jstor.org/stable/3213832.

R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

A. Tamar and S. Mannor. Variance adjusted actor critic algorithms, 2013. URL https://arxiv.org/abs/1310.3697.

A. Tamar, Y. Chow, M. Ghavamzadeh, and S. Mannor. Policy gradient for coherent risk measures. *CoRR*, abs/1502.03919, 2015a. URL http://arxiv.org/abs/1502.03919.

A. Tamar, Y. Glassner, and S. Mannor. Optimizing the cvar via sampling. In *AAAI*, 2015b.

P. Thomas. Bias in natural actor-critic algorithms. In E. P. Xing and T. Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 441–448, Bejing, China, 22–24 Jun 2014. PMLR. URL https://proceedings.mlr.press/v32/thomas14.html.

C. Yu, J. Liu, and S. Nemati. Reinforcement learning in healthcare: A survey. *CoRR*, abs/1908.08796, 2019. URL http://arxiv.org/abs/1908.08796.

S. Zhang, B. Liu, and S. Whiteson. Per-step reward: A new perspective for risk-averse reinforcement learning. *CoRR*, abs/2004.10888, 2020. URL https://arxiv.org/abs/2004.10888.

A. Discounted state occupancy measure

Let us define more precisely the discounted state occupancy measure. We define the t-step state transition density under policy π, $\forall s, s' \in \mathcal{S}$:

$$
\begin{aligned}
&p_\pi(s \xrightarrow{0} s') \coloneqq \delta(s' - s),\\
&p_\pi(s \xrightarrow{1} s') \coloneqq p_\pi(s'|s) \coloneqq \mathop{\mathbb{E}}_{a\sim\pi(\cdot|s)}[P(s'|s,a)]),\\
&p_\pi(s \xrightarrow{t+1} s') \coloneqq \mathop{\mathbb{E}}_{\tilde{s}\sim p_\pi(\cdot|s)}\left[p_\pi(\tilde{s} \xrightarrow{t} s')\right] \quad \forall t > 1,
\end{aligned}
$$

where δ is the Dirac delta distribution, the discounted state occupancy measures is:

$$
\begin{aligned}
d_\pi(s'|s) &\coloneqq (1-\gamma)\sum_{t=0}^{\infty}\gamma^t p_\pi(s \xrightarrow{t} s'),\\
d_{\mu,\pi}(s) &\coloneqq \mathop{\mathbb{E}}_{s_0\sim\mu(\cdot)}\left[d_\pi(s|s_0)\right],
\end{aligned}
$$

where $(1-\gamma)$ is a normalization constant.
Lemma A.1 will be useful in the analysis of the estimators and in order to prove the relations between different risk measures.

Lemma A.1 (Trajectory distribution and occupancy measure relation (D'Oro et al., 2019)). *Let π be a policy and $f : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ be a bounded measurable function. The trajectory distribution and the (discounted) occupancy measure are related by this equation:*

$$(1-\gamma) \mathop{\mathbb{E}}_{\substack{s_0 \sim \mu(\cdot), \\ a_t \sim \pi(\cdot|s_t) \\ s_{t+1} \sim P(\cdot|s_t, a_t)}} \left[\sum_{t=0}^{\infty} \gamma^t f(s_t, a_t) \right] = \mathop{\mathbb{E}}_{\substack{s \sim d_{\mu,\pi}(\cdot) \\ a \sim \pi(\cdot|s)}} [f(s,a)] .$$

B. Practical version of RMAD-TRPO

For the practical version of RMAD-TRPO we followed Schulman et al. (2017a).
We start from the original maximization problem:

$$\arg\max_{\boldsymbol{\theta} \in \Theta} \left[L_k^{\lambda}(\pi) - \frac{4\gamma\epsilon_{\lambda}}{1-\gamma} D_{KL}^{max}(\pi_{\boldsymbol{\theta}_k}, \pi_{\boldsymbol{\theta}}) - \lambda(1-\gamma)|A_{\boldsymbol{\theta}_k}^{\boldsymbol{\theta}}| - \lambda \frac{4\epsilon\gamma}{(1-\gamma)} D_{KL}^{max}(\pi_{\boldsymbol{\theta}_k}, \pi_{\boldsymbol{\theta}}) \right] .$$

In order to take large steps in a robust way we use a constraint on the KL divergence, i.e., a trust region constraint:

$$\arg\max_{\boldsymbol{\theta} \in \Theta} \left[L_k^{\lambda}(\pi) - \lambda(1-\gamma)|A_{\boldsymbol{\theta}_k}^{\boldsymbol{\theta}}| \right]$$
$$\text{subject to } D_{KL}^{max}(\pi_{\boldsymbol{\theta}_k}, \pi_{\boldsymbol{\theta}}) \leqslant \delta.$$

A heuristic approximation considers the average KL divergence:

$$\arg\max_{\boldsymbol{\theta} \in \Theta} \left[L_k^{\lambda}(\pi) - \lambda(1-\gamma)|A_{\boldsymbol{\theta}_k}^{\boldsymbol{\theta}}| \right]$$
$$\text{subject to } \mathop{\mathbb{E}}_{s \sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot)} [D_{KL}(\pi_{\boldsymbol{\theta}_k}(\cdot|s), \pi_{\boldsymbol{\theta}}(\cdot|s))] \leqslant \delta,$$

We can expand $L_k^{\lambda}(\pi)$ and $A_{\boldsymbol{\theta}_k}^{\boldsymbol{\theta}}$ and remove $\eta_{\boldsymbol{\theta}_k}$:

$$\arg\max_{\boldsymbol{\theta} \in \Theta} \left[\mathop{\mathbb{E}}_{\substack{s \sim d_{\mu,\pi_{\boldsymbol{\theta}_k}} \\ a \sim \pi_{\boldsymbol{\theta}}(\cdot|s)}} A_{\boldsymbol{\theta}_k}^{\lambda}(s,a) - \lambda(1-\gamma) \left| \mathop{\mathbb{E}}_{\tau|\pi_{\boldsymbol{\theta}_k}} \left[\sum_{t=0}^{\infty} \gamma^t \mathop{\mathbb{E}}_{a \sim \pi_{\boldsymbol{\theta}}(\cdot, s_t)} [A_{\boldsymbol{\theta}_k}(s_t, a)] \right] \right| \right]$$
$$\text{subject to } \mathop{\mathbb{E}}_{s \sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot)} [D_{KL}(\pi_{\boldsymbol{\theta}_k}(\cdot|s), \pi_{\boldsymbol{\theta}}(\cdot|s))] \leqslant \delta.$$

Finally we use the importance sampling, the optimization problem becomes:

$$\arg\max_{\boldsymbol{\theta} \in \Theta} \left[\mathop{\mathbb{E}}_{\substack{s \sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot) \\ a \sim \pi_{\boldsymbol{\theta}_k}(\cdot|s)}} \left[\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)} A_{\boldsymbol{\theta}_k}^{\lambda}(s,a) \right] - \lambda \left| \mathop{\mathbb{E}}_{\substack{s \sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot) \\ a \sim \pi_{\boldsymbol{\theta}_k}(\cdot|s)}} \left[\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)} A_{\boldsymbol{\theta}_k}(s_t, a) \right] \right| \right]$$
$$\text{subject to } \mathop{\mathbb{E}}_{s \sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot)} [D_{KL}(\pi_{\boldsymbol{\theta}_k}(\cdot|s), \pi_{\boldsymbol{\theta}}(\cdot|s))] \leqslant \delta.$$

C. Hidden-concavity counterexamples

C.1. Mean-RMAD

The Mean-RMAD risk measure is not concave with respect to the state-action occupancy measure, thus it has not the hidden-concavity property (or hidden-convexity in case of costs). Here there is a counterexample to the hidden-concavity.
Let us consider a MAB with two actions. The first action gives always a reward of 0, the second action gives always a reward of 1. Let us call also $\eta^{\lambda}(d)$ the Mean-RMAD of the (discounted) state-action occupancy measure d. For a risk-aversion factor of 0.4, t equal to 0.2 and $d_1(action1) = 1, d_1(action2) = 0, d_2(action1) = 0, d_2(action2) = 1$, we obtain:

$$\eta^{\lambda}(d_1) = 0,$$
$$\eta^{\lambda}(d_2) = 1,$$
$$td_1 + (1-t)d_2 = [0.2, 0.8],$$
$$\eta^{\lambda}(td_1 + (1-t)d_2) = 0.8 - 0.4 * 0.32 = 0.672 < t\eta^{\lambda}(d_1) + (1-t)\eta^{\lambda}(d_2) = 0.8,$$

which violates the concavity of η^λ:

$$\eta^\lambda(td_1 + (1-t)d_2) \geqslant t\eta^\lambda(d_1) + (1-t)\eta^\lambda(d_2) \quad \forall t \in [0,1], \forall d_1, d_2.$$

C.2. RCVaR

The reward-based RCVaR has not the hidden-concavity property (or hidden-convexity in case of costs). Here there is a counterexample to the hidden-concavity. Let us call $\rho^\alpha(d)$ and $\eta^\alpha(d)$ the RVaR and the RCVaR of the (discounted) state-action occupancy measure d, respectively. We consider two discounted state-action occupancy measures $d_{\mu,\pi,1} = [0.1, 0.15, 0.2, 0.16, 0.19, 0.2]$ and $d_{\mu,\pi,2} = [0.2, 0.1, 0.1, 0.4, 0.1, 0.1]$, the rewards $[-5, 0, -2, 3, 4, 0.5]$ and $\alpha = 0.14$, we obtain for the first occupancy measure:

$$\rho^\alpha(d_{\mu,\pi,1}) = -2,$$
$$\eta^\alpha(d_{\mu,\pi,1}) = -4.142857143,$$

while for the second one:

$$\rho^\alpha(d_{\mu,\pi,2}) = -5,$$
$$\eta^\alpha(d_{\mu,\pi,2}) = -5.$$

For $t = 0.6$ the linear combination of the two occupancy measures is
$d_{\mu,\pi,t} = td_{\mu,\pi,1} + (1-t)\nu_{\mu,\pi,2} = [0.14, 0.13, 0.16, 0.266, 0.154, 0.16]$ and with this occupancy measure:

$$\rho^\alpha(d_{\mu,\pi,t}) = -2,$$
$$\eta^\alpha(d_{\mu,\pi,t}) = -5,$$

while the linear combination of the two RCVaR is $t\eta^\alpha(d_{\mu,\pi,1}) + (1-t)\eta^\alpha(d_{\mu,\pi,2}) = -4.485714286$ so:

$$\eta^\alpha(d_{\mu,\pi,t}) = -5 < t\eta^\alpha(d_{\mu,\pi,1}) + (1-t)\eta^\alpha(d_{\mu,\pi,2}) = -4.485714286,$$

which violates the hidden-concavity property. These two occupancy measures can be obtained by considering a finite-horizon MDP with horizon equal to one, six states and one action per state that brings back to itself, so we obtain that each discounted state-action occupancy measure is equal to the chosen initial state probability. Or we can consider a finite-horizon MDP with horizon equal to one, one state and six actions from this state to itself and initial probability of the unique state equal to one, in this MDP the discounted state-action occupancy measure is equal to the chosen policy.

D. Estimators

In this section we provide the estimators needed to optimize the Mean-RMAD and the RCVaR. Let us consider that the trajectories are sampled with policy π and that each episode is run for T steps so that each sampled trajectory has a length equal to T. The following estimators will be a bit biased due to the finite length T of the sampled trajectories, but we assume that T is big enough to have $\gamma^T \approx 0$, in this way the part of the trajectory that is not consider doesn't influence much the estimator.

D.1. Estimators for Mean-RMAD optimization

First we consider an unbiased estimator of the normalized expected return:

$$\hat{\bar{J}}_\pi := (1-\gamma)\frac{1}{N}\sum_{i=0}^{N-1}\sum_{t=0}^{T-1}\gamma^t R_i(s_t, a_t),$$

where N is the batch size.
Proof.
We want to prove:

$$\mathop{\mathbb{E}}_{\substack{s_0\sim\mu,\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}}[\hat{\bar{J}}_\pi] = \bar{J}_\pi.$$

First, we consider:

$$\left|(1-\gamma)\mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}}\left[\sum_{t=T}^{\infty}\gamma^t R(s_t,a_t)\right]\right| = \gamma^T\left|(1-\gamma)\mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}}\left[\sum_{t=T}^{\infty}\gamma^{t-T} R(s_t,a_t)\right]\right| \leqslant \gamma^T R_{max}.$$

so for sufficiently large T the quantity is arbitrary near to zero and we can ignore the term. Thus:

$$\bar{J}_\pi \approx (1-\gamma) \mathop{\mathbb{E}}_{\substack{s_0\sim\mu,\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{T-1} \gamma^t R(s_t,a_t)\right].$$

Let us consider the expected value of the estimator $\hat{\bar{J}}_\pi$:

$$\begin{aligned}
\mathop{\mathbb{E}}_{\substack{s_0\sim\mu,\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} [\hat{\bar{J}}_\pi] &= \mathop{\mathbb{E}}_{\substack{s_0\sim\mu,\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[(1-\gamma)\frac{1}{N}\sum_{i=0}^{N-1}\sum_{t=0}^{T-1}\gamma^t R_i(s_t,a_t)\right] \\
&= \frac{1}{N}\sum_{i=0}^{N-1}(1-\gamma) \mathop{\mathbb{E}}_{\substack{s_0\sim\mu,\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{T-1}\gamma^t R_i(s_t,a_t)\right] \\
&\approx \frac{1}{N}\sum_{i=0}^{N-1}\bar{J}_\pi = \bar{J}_\pi.
\end{aligned}$$

■

An estimator of the RMAD that uses one batch of sampled trajectories is:

$$\hat{\omega}_\pi \coloneqq (1-\gamma)\frac{1}{N}\sum_{i=0}^{N-1}\sum_{t=0}^{T-1}\gamma^t |R_i(s_t,a_t) - \hat{\bar{J}}_{1,\pi}|,$$

where N is the batch size. It has a biased greater than or equal to zero, but it is asymptotically unbiased[5]. A conservative estimator, with $-2\omega_\pi \leqslant bias \leqslant 0$ but asymptotically unbiased, is:

$$\hat{\omega}_\pi \coloneqq (1-\gamma)\frac{1}{N}\sum_{i=0}^{N-1}\sum_{t=0}^{T-1}\gamma^t \left(R_i(s_t,a_t) - \hat{\bar{J}}_{\pi,1}\right) \operatorname{sign}\left(R_i(s_t,a_t) - \hat{\bar{J}}_{\pi,1}\right),$$

it requires three independent batches of trajectories: one for $\hat{\bar{J}}_{\pi,1}$, one for $\hat{\bar{J}}_{\pi,2}$ and one for the rewards $\{R_i\}_{i=0}^{N-1}$.
Proof.
Regarding the first estimator:

$$\hat{\omega}_\pi \coloneqq (1-\gamma)\frac{1}{N}\sum_{i=0}^{N-1}\sum_{t=0}^{T-1}\gamma^t |R_i(s_t,a_t) - \hat{\bar{J}}_\pi|,$$

its bias is:

$$\begin{aligned}
bias &= \mathop{\mathbb{E}}_{\tau_1}\Big[\mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \hat{\omega}_\pi\Big] - \omega_\pi \\
&= \left(\frac{1}{N}\sum_{i=0}^{N-1}(1-\gamma) \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{T-1}\gamma^t \mathop{\mathbb{E}}_{\tau_1}|R_i(s_t,a_t) - \hat{\bar{J}}_{1,\pi}|\right]\right) - \omega_\pi,
\end{aligned}$$

we use the Jensen's inequality:

$$\begin{aligned}
bias &\geqslant \left(\frac{1}{N}\sum_{i=0}^{N-1}(1-\gamma) \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{T-1}\gamma^t |\mathop{\mathbb{E}}_{\tau_1}(R_i(s_t,a_t) - \hat{\bar{J}}_{1,\pi})|\right]\right) - \omega_\pi \\
&= \left(\frac{1}{N}\sum_{i=0}^{N-1}(1-\gamma) \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{T-1}\gamma^t |R_i(s_t,a_t) - \bar{J}_\pi|\right]\right) - \omega_\pi = 0
\end{aligned}$$

[5]Sometimes an estimator which is asymptotically unbiased is called also consistent, even if it is different from weak consistency and from consistency in mean square.

Asymptotically the bias is:

$$\begin{aligned}
\lim_{N\to\infty} \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \hat{\omega}_\pi &= \lim_{N\to\infty} \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[(1-\gamma)\frac{1}{N}\sum_{i=0}^{N-1}\sum_{t=0}^{T-1}\gamma^t|R_i(s_t,a_t)-\hat{\bar{J}}_\pi|\right] \\
&= \lim_{N\to\infty} \frac{1}{N}\sum_{i=0}^{N-1}(1-\gamma) \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{T-1}\gamma^t|R_i(s_t,a_t)-\hat{\bar{J}}_\pi|\right] \\
&= \lim_{N\to\infty} \frac{1}{N}\sum_{i=0}^{N-1}(1-\gamma) \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{T-1}\gamma^t|R_i(s_t,a_t)-\bar{J}_\pi|\right] \quad \text{almost surely} \\
&= \omega_\pi,
\end{aligned}$$

considering that, for $N\to\infty$, $\hat{\bar{J}}_\pi$ tends to $\bar{J}_\pi$ almost surely for the strong law of large numbers.
While for this second estimator:

$$\hat{\omega}_\pi \coloneqq (1-\gamma)\frac{1}{N}\sum_{i=0}^{N-1}\sum_{t=0}^{T-1}\gamma^t\left(R_i(s_t,a_t)-\hat{\bar{J}}_{1,\pi}\right)\operatorname{sign}\left(R_i(s_t,a_t)-\hat{\bar{J}}_{2,\pi}\right),$$

we get:

$$\begin{aligned}
bias &= \mathop{\mathbb{E}}_{\tau_2}\mathop{\mathbb{E}}_{\tau_1}\Big[\mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \hat{\omega}_\pi\Big] - \omega_\pi \\
&= \left(\frac{1}{N}\sum_{i=0}^{N-1}(1-\gamma) \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{T-1}\gamma^t \mathop{\mathbb{E}}_{\tau_1}(R_i(s_t,a_t)-\hat{\bar{J}}_\pi,1)\mathop{\mathbb{E}}_{\tau_2}\operatorname{sign}(R_i(s_t,a_t)-\hat{\bar{J}}\pi,2)\right]\right)-\omega_\pi \\
&= \left(\frac{1}{N}\sum_{i=0}^{N-1}(1-\gamma) \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{T-1}\gamma^t(R_i(s_t,a_t)-\hat{\bar{J}}_{2,\pi})\mathop{\mathbb{E}}_{\tau_2}\operatorname{sign}(R_i(s_t,a_t)-\hat{\bar{J}}\pi,2)\right]\right)-\omega_\pi
\end{aligned}$$

we have that $-1 \leqslant \mathbb{E}_{\tau_2}\operatorname{sign}(R_i(s_t,a_t)-\hat{\bar{J}}\pi,2) \leqslant 1$ and so, since $-|x| \leqslant x*y \leqslant |x| for -1 \leqslant y \leqslant 1$, $-|R_i(s_t,a_t)-\bar{J}_\pi| \leqslant (R_i(s_t,a_t)-\bar{J}_\pi)\,\mathbb{E}_{\tau_2}\operatorname{sign}(R_i(s_t,a_t)-\hat{\bar{J}}\pi,2) \leqslant |R_i(s_t,a_t)-\bar{J}_\pi|$, from which:

$$-2\omega_\pi \leqslant bias \leqslant 0.$$

Analogously to the first estimator, if the sizes of the two batches of $\hat{\bar{J}}_{\pi,1}$ and $\hat{\bar{J}}_{\pi,2}$ tend to infinity then the two estimators of the normalized expected return tend to the normalized expected return almost surely for the strong law of the large numbers. So we obtain that the estimator tends to the RMAD almost surely, thus it is asymptotically unbiased.

■

An estimator of the mean sign deviation ψ_π is:

$$\hat{\psi}_\pi \coloneqq (1-\gamma)\frac{1}{N}\sum_{i=0}^{N-1}\sum_{t=0}^{T-1}\gamma^t\operatorname{sign}(R_i(s_t,a_t)-\hat{\bar{J}}_\pi),$$

which uses one batch of sampled trajectories with size N. It is asymptotically unbiased almost surely as the previous estimators.
Analogously as before, one estimator of the subgradient of the RMAD (4.1) is:

$$(1-\gamma)\frac{1}{N}\sum_{i=0}^{N-1}\left[\sum_{t=0}^{T-1}\nabla_{\boldsymbol{\theta}}\log\pi_{\boldsymbol{\theta}}(a_t|s_t)\sum_{t=0}^{T-1}\gamma^t\left(|R(s_t,a_t)-\bar{J}_{\boldsymbol{\theta}}|-\psi_{\boldsymbol{\theta}}R(s_t,a_t)\right)\right],$$

which is in general biased, but asymptotically is unbiased almost surely because $\hat{\bar{J}}_{\boldsymbol{\theta}}$ tends to $\bar{J}_{\boldsymbol{\theta}}$ and $\hat{\psi}_{\boldsymbol{\theta}}$ tends to $\psi_{\boldsymbol{\theta}}$ almost surely.
Proof.

$$
\begin{aligned}
bias &= \mathop{\mathbb{E}}_{\tau_1}[\mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \hat{\psi}_\pi] - \psi_\pi \\
&= \left(\frac{1}{N}\sum_{i=0}^{N-1}(1-\gamma) \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{T-1} \gamma^t \mathop{\mathbb{E}}_{\tau_1} \operatorname{sign}(R_i(s_t,a_t) - \hat{\bar{J}}_\pi) \right] \right) - \psi_\pi,
\end{aligned}
$$

$\mathbb{E}_{\tau_1} \operatorname{sign}(R_i(s_t,a_t) - \hat{\bar{J}}\pi, 1)$ is a quantity between -1 and 1, so the bias can be:

$$-1 - \psi_\pi \leqslant bias \leqslant 1 - \psi_\pi.$$

Analogously to the estimators of the RMAD, if N tends to infinity the estimator tends to the mean sign deviation almost surely due to the normalized expected return estimator that tends to the normalized expected return almost surely, thus it is asymptotically unbiased.

■

D.2. Estimators for RCVaR optimization

The RVaR can be estimated in this way: sample a batch of trajectories; weight each sampled reward with γ^t where t is the timestep at which the reward was sampled during the episode; sort the rewards in ascending order; finally select the reward $\bar{R}$ for which the sum of the weights of the rewards that are smaller than $\bar{R}$ is $\frac{\alpha N}{(1-\gamma)}$. $\hat{RVaR} = \bar{R}$ is the estimated RVaR. This estimator has a bias that tends in distribution to zero when the batch size tends to infinity, more details are provided in Dong and Nakayama (2018).
Using an estimator of the RVaR, we can estimate the RCVaR with:

$$\hat{RCVaR} = \hat{RVaR} - \frac{1}{\alpha}(1-\gamma)\frac{1}{N}\sum_{i=0}^{N-1}\left[\sum_{t=0}^{T-1}\gamma^t\left(R(s_t,a_t) - \hat{RVaR}\right)_-\right],$$

and with one batch. It is asymptotically unbiased almost surely for the strong law of the large numbers if the estimator of the RVaR tends asymptotically to the RVaR almost surely.
The same happens with the gradient of the RCVaR, using the previous estimator of the RVaR we can estimate the gradient with one batch with:

$$\frac{1}{\alpha}(1-\gamma)\frac{1}{N}\sum_{i=0}^{N-1}\left[\sum_{t=0}^{T-1}\nabla_{\boldsymbol{\theta}}\log\pi_{\boldsymbol{\theta}}(a_t|s_t)\left(R(s_t,a_t) - \hat{RVaR}\right)_-\right],$$

which is asymptotically unbiased almost surely for the strong law of the large numbers if the estimator of the RVaR tends asymptotically to the RVaR almost surely.
An estimator of the RVaR that is asymptotically unbiased almost surely can be obtained with the following method. The RVaR can be written in this way (Rockafellar and Uryasev, 2001):

$$RVaR_\pi^\alpha = \arg\max_{\rho\in\mathbb{R}}\left\{(1-\gamma)\mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot)\\ a_t\sim\pi_{\boldsymbol{\theta}_k}(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}}\left[\sum_{t=0}^{\infty}\gamma^t\left(\rho - \frac{1}{\alpha}(R(s,a)-\rho)_-\right)\right]\right\}$$

Thus, we can create the needed estimator of the RVaR with one batch by solving the following problem:

$$\hat{RVaR} = \arg\max_{\rho\in\mathbb{R}}\left\{(1-\gamma)\frac{1}{N}\sum_{i=0}^{N-1}\left[\sum_{t=0}^{T-1}\gamma^t\left(\rho - \frac{1}{\alpha}(R(s,a)-\rho)_-\right)\right]\right\}.$$

It is asymptotically unbiased almost surely for the strong law of the large numbers, because this function:

$$(1-\gamma)\frac{1}{N}\sum_{i=0}^{N-1}\left[\sum_{t=0}^{T-1}\gamma^t\left(\rho - \frac{1}{\alpha}(R(s,a)-\rho)_-\right)\right].$$

which is inside the argmax, tends to the function inside the argmax of the formula of the RVaR:

$$(1-\gamma) \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot)\\ a_t\sim\pi_{\boldsymbol{\theta}_k}(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{\infty}\gamma^t\left(\rho-\frac{1}{\alpha}\left(R(s,a)-\rho\right)_-\right)\right],$$

almost surely, given that it is just a mean. In order to solve the maximization problem we can use algorithms for convex optimization, thanks to Proposition D.1:

Proposition D.1. *The function:*

$$f(\rho)=(1-\gamma) \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot)\\ a_t\sim\pi_{\boldsymbol{\theta}_k}(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{\infty}\gamma^t\left(\rho-\frac{1}{\alpha}\left(R(s,a)-\rho\right)_-\right)\right]$$

is concave with respect to ρ. The proof is in Appendix D.1.

For example we can exploit this subgradient:

$$(1-\gamma)\frac{1}{N}\sum_{i=0}^{N-1}\left[\sum_{t=0}^{T-1}\gamma^t\left(1-\frac{1}{\alpha}\frac{1}{2}\left(-\operatorname{sign}\left(R(s,a)-\rho\right)+1\right)\right)\right]$$

and use subgradient ascent.

E. Definitions of other risk-measures

Now we present some other risk measures that can be put together into relation.
The *Squared Root Volatility* is:

$$\nu_\pi \coloneqq \sqrt{\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}}\left[\left(R(s,a)-\bar{J}_\pi\right)^2\right]}$$

from which comes the *Mean-Squared Root Volatility*:

$$\bar{J}_\pi-\beta\nu_\pi,$$

for a certain risk-aversion factor β.
The *Semi-Volatility* is:

$$\nu^2_{-,\pi} \coloneqq \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}}\left[\left(R(s,a)-\bar{J}_\pi\right)_-^2\right]$$

wich its squared root $\nu_{-,\pi}$, called *Squared Root Semi-Volatility*, which allow to create the *Mean-Semi-Volatility*:

$$\bar{J}_\pi-\beta\nu^2_{-,\pi}$$

and the *Mean-Squared Root Semi-Volatility*:

$$\bar{J}_\pi-\beta\nu_{-,\pi}$$

for a certain risk-aversion factor β. We defined the Mean-Squared Root Volatility, the Mean-Semi-Volatility and the Mean-Squared Root Semi-Volatility because their properties were showed in Table 1. Interestingly, the Mean-Squared Root Semi-Volatility is coherent for $\beta\in[0,1]$ (Shapiro et al., 2021).
The *Semi-Variance* is:

$$\sigma^2_{-,\pi} \coloneqq \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot)\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}}\left[\left(G-J_\pi\right)_-^2\right].$$

and its squared root $\sigma_{-,\pi}$.
The *Semi-RMAD* is:

$$\omega_{-,\pi} \coloneqq \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}}\left[\left(R(s,a)-\bar{J}_\pi\right)_-\right]$$

and its analogue on the return, the *Semi-MAD*, is:

$$\omega^G_{-,\pi} = \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}}\left[\left(G-J_\pi\right)_-\right]$$

The *Standard Deviation* is:

$$\sigma_\pi = \sqrt{\mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}}[(G-J_\pi)]}.$$

In order to be concise we will use these symbols for the risk measures defined in Section 3:

$$\begin{aligned}
\omega_\pi &\coloneqq \mathrm{MAD}(R, d_{\mu,\pi}),\\
d_\pi^2 &\coloneqq \mathrm{Var}(R, d_{\mu,\pi}),\\
\omega_\pi^G &\coloneqq \mathrm{MAD}(G, p_\pi),\\
\eta_\pi^{\lambda,G} &\coloneqq \mathrm{MMAD}^\lambda(G, p_\pi),\\
\eta_\pi^{\alpha,G} &\coloneqq \mathrm{CVaR}^\alpha(G, p_\pi),\\
\sigma_\pi^2 &\coloneqq \mathrm{Var}(G, p_\pi).
\end{aligned}$$

F. Other relations between risk-measures

1. The first relation is between the RMAD and the Semi-RMAD:
   (a)
   $$\omega_\pi = 2\omega_{-,\pi}$$
   (b) and it happens also between the return-based MAD and the return-based Semi-MAD:
   $$\omega_\pi^G = 2\omega_{-,\pi}^G.$$

2. Here we present other relations between reward-based risk measures and return-based risk measures that were not showed in Section 3, $\forall\gamma \in (0,1)$:
   (a)
   $$\sigma_-^2 \leqslant \frac{\nu_-^2}{(1-\gamma)^2},$$
   (b)
   $$\sigma_- \leqslant \frac{\nu_-}{(1-\gamma)},$$
   (c)
   $$\sigma_\pi \leqslant \frac{\nu_\pi}{(1-\gamma)},$$
   (d)
   $$\omega_{-,\pi}^G \leqslant \frac{\omega_{-,\pi}}{(1-\gamma)}.$$

3. Furthermore here there are other relations between reward-based risk measures:
   (a)
   $$\omega_{-,\pi} \leqslant \nu_{-,\pi},$$
   (b)
   $$\omega_\pi \leqslant 2\nu_{-,\pi},$$
   (c)
   $$\omega_\pi \leqslant \nu_\pi,$$
   (d)
   $$(\omega_{-,\pi})^2 \leqslant \nu_{-,\pi}^2,$$
   (e)
   $$(\omega_\pi)^2 \leqslant 4\nu_{-,\pi}^2.$$

4. Finally we can obtain similar relations between return-based risk measures, $\forall\alpha,\gamma \in (0,1)$:
   (a)
   $$\eta_\pi^{\alpha,G} \geqslant J_\pi - \frac{1}{2\alpha}\omega_\pi^G = \text{ Mean-MAD with risk-aversion factor } \lambda = \frac{1}{\alpha},$$
   (b)
   $$\omega_\pi^G \leqslant \sigma_\pi,$$

(c)

$$\eta_\pi^{\alpha,G} \geqslant J_\pi - \frac{1}{2\alpha}\sigma_\pi = \text{ Mean-Standard Deviation with risk-aversion factor } \beta = \frac{1}{2\alpha},$$

(d)

$$\omega_{-,\pi}^G \leqslant \sigma_{-,\pi},$$

(e)

$$\omega_\pi^G \leqslant 2\sigma_{-,\pi},$$

(f)

$$(\omega_\pi^G)^2 \leqslant \sigma_\pi^2,$$

(g)

$$(\omega_{-,\pi}^G)^2 \leqslant \sigma_{-,\pi}^2,$$

(h)

$$(\omega_\pi^G)^2 \leqslant 4\sigma_{-,\pi}^2.$$

Proof.

1. (a)

$$\begin{aligned}\omega_{-,\pi} &= \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}}\left[\left(R(s,a)-\bar{J}_\pi\right)_-\right] = \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}}\left[\frac{\left|R(s,a)-\bar{J}_\pi\right| - \left(R(s,a)-\bar{J}_\pi\right)}{2}\right] \\ &= \frac{\omega_\pi}{2} - \frac{1}{2}\underbrace{\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}}\left[\left(R(s,a)-\bar{J}_\pi\right)\right]}_{=0} = \frac{\omega_\pi}{2}.\end{aligned}$$

.

(b) The proof is very similar to the one before.

2. (a)

$$\begin{aligned}\sigma_{-,\pi}^2 &\coloneqq \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}}\left[\left(\sum_{t=0}^\infty \gamma^t R(s_t,a_t) - \frac{\bar{J}_\pi}{1-\gamma}\right)_-^2\right] \\ &= \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}}\left[\left(\sum_{t=0}^\infty \gamma^t (R(s_t,a_t) - \bar{J}_\pi)\right)_-^2\right],\end{aligned}$$

we use the fact that the negative part of the sum is less than or equal to the sum of the negative parts:

$$\leqslant \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}}\left[\left(\sum_{t=0}^\infty (\gamma^t R(s_t,a_t) - \bar{J}_\pi)_-\right)^2\right],$$

γ^t is always positive so can be moved outside the negative part:

$$= \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}}\left[\left(\sum_{t=0}^\infty \gamma^t (R(s_t,a_t) - \bar{J}_\pi)_-\right)^2\right],$$

we can use the Cauchy-Schwarz inequality:

$$\begin{aligned}&\leqslant \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}}\left[\left(\sum_{t=0}^\infty \gamma^t\right)\left(\sum_{t=0}^\infty \gamma^t (R(s_t,a_t) - \bar{J}_\pi)_-^2\right)\right] \\ &= \frac{1}{1-\gamma}\mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}}\left[\sum_{t=0}^\infty \gamma^t (R(s_t,a_t) - \bar{J}_\pi)_-^2\right],\end{aligned}$$

finally we use Lemma A.1

$$= \frac{1}{(1-\gamma)^2} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}} \left[(R(s,a) - \bar{J}_\pi)_-^2\right]$$
$$= \frac{\nu_-^2}{(1-\gamma)^2}.$$

(b) The relation comes by making the square root of both sides of $\sigma_-^2 \leqslant \frac{\nu_-^2}{(1-\gamma)^2}$, given that they are always greater than or equal to zero.

(c) The relation comes by making the square root of both sides of $\sigma^2 \leqslant \frac{\nu^2}{(1-\gamma)^2}$, given that they are always greater than or equal to zero.

(d) This relation can obtained using the proof of 5 by substituting the absolute value with the negative part function.

3. (a)

$$\omega_{-,\pi} \coloneqq \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}} \left[\left(R(s,a) - \bar{J}_\pi\right)_-\right]$$
$$= \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}} \left[\sqrt{\left(R(s,a) - \bar{J}_\pi\right)_-^2}\right]$$
$$\leqslant \sqrt{\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}} \left[\left(R(s,a) - \bar{J}_\pi\right)_-^2\right]} = \nu_{-,\pi}.$$

(b) This inequality comes from $\omega_{-,\pi} \leqslant \nu_{-,\pi}$ and using $\omega_\pi = 2\omega_{-,\pi}$.

(c) This inequality can be obtained by making the squared root of both sides of $(\omega_\pi)^2 \leqslant d_\pi^2$ (Proposition 3.2), because both sides are always positive.

(d) This inequality can be obtained by making the squared of both sides of $\omega_{-,\pi} \leqslant \nu_{-,\pi}$, because both sides are always positive.

(e) This relation results by combining $\omega_{-,\pi}^2 \leqslant \nu_{-,\pi}^2$ with $\omega_\pi = 2\omega_{-,\pi}$.

4. These inequalities are between return-based risk measures so we have a different probability, the return instead of the reward and the expected return instead of the normalized expected return, which don't alter the proofs of the corresponding reward-based risk measures, that can still be applied with minor modifications.

■

G. Proofs

G.1. Proof of Proposition 3.1 (Relations between reward-based risk measures and the corresponding return based versions)

First we prove the relation (5) between the two MADs:

$$\mathrm{MAD}(G, p_\pi) \coloneqq \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\left|\sum_{t=0}^{\infty}\gamma^t R(s_t,a_t) - \frac{\bar{J}_\pi}{1-\gamma}\right|\right]$$
$$= \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\left|\sum_{t=0}^{\infty}\gamma^t (R(s_t,a_t) - \bar{J}_\pi)\right|\right],$$

we use the fact that the absolute value of the sum is less that or equal to the sum of the absolute values:

$$\leqslant \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{\infty}|\gamma^t R(s_t,a_t) - \bar{J}_\pi|\right],$$

γ^t is always positive so can be moved outside the absolute value:

$$= \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{\infty}\gamma^t|R(s_t,a_t)-\bar{J}_\pi|\right],$$

finally we use Lemma A.1:

$$= \frac{1}{(1-\gamma)} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}} \left[|R(s,a)-\bar{J}_\pi|\right]$$
$$= \frac{\mathrm{MAD}(R,d_{\mu,\pi})}{(1-\gamma)}.$$

Now we prove the relation (6) between the two CVARs:

$$\begin{aligned}
\mathrm{CVaR}^\alpha(G,p_\pi) &= \max_\rho \left\{ \rho - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[(G-\rho)_-\right] \right\} \\
&= \max_\rho \left\{ \rho - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\left(\sum_{t=0}^{\infty}\gamma^t R(s_t,a_t)-\rho\right)_-\right] \right\} \\
&= \max_\rho \left\{ \rho - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\left(\sum_{t=0}^{\infty}\gamma^t \left(R(s_t,a_t)-(1-\gamma)\rho\right)\right)_-\right] \right\},
\end{aligned}$$

we put $\frac{1}{(1-\gamma)}\mathrm{VaR}^\alpha(R,d_{\mu,\pi})$ in ρ:

$$\geqslant \frac{1}{(1-\gamma)}\mathrm{VaR}^\alpha(R,d_{\mu,\pi}) - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\left(\sum_{t=0}^{\infty}\gamma^t \left(R(s_t,a_t)-\mathrm{VaR}^\alpha(R,d_{\mu,\pi})\right)\right)_-\right],$$

use the fact that the negative part of the sum is less than or equal to the sum of the negative parts:

$$\begin{aligned}
&\geqslant \frac{1}{(1-\gamma)}\mathrm{VaR}^\alpha(R,d_{\mu,\pi}) - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{\infty}\gamma^t \left(R(s_t,a_t)-\mathrm{VaR}^\alpha(R,d_{\mu,\pi})\right)_-\right] \\
&= \frac{1}{(1-\gamma)} \left(\mathrm{VaR}^\alpha(R,d_{\mu,\pi}) - \frac{1}{\alpha}(1-\gamma) \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{\infty}\gamma^t \left(R(s_t,a_t)-\mathrm{VaR}^\alpha(R,d_{\mu,\pi})\right)_-\right] \right),
\end{aligned}$$

finally we use Lemma A.1:

$$\begin{aligned}
&= \frac{1}{(1-\gamma)} \left(\mathrm{VaR}^\alpha(R,d_{\mu,\pi}) - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}} \left[\left(R(s,a)-\mathrm{VaR}^\alpha(R,d_{\mu,\pi})\right)_-\right] \right) \\
&= \frac{1}{(1-\gamma)}\mathrm{CVaR}^\alpha(R,d_{\mu,\pi}).
\end{aligned}$$

■

G.2. Proof of Proposition 3.2 (Relations between different risk measures)

Let us begin with relation (7)

$$\begin{aligned}
\mathrm{CVaR}^{\alpha}(R, d_{\mu,\pi}) =& \max_{\rho} \left\{ \rho - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot) \\ a\sim\pi(\cdot|s)}} \left[(R(s,a) - \rho)_- \right] \right\} \\
\geqslant& \bar{J}_\pi - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot) \\ a\sim\pi(\cdot|s)}} \left[\left(R(s,a) - \bar{J}_\pi \right)_- \right] \\
=& \bar{J}_\pi - \frac{1}{\alpha} \omega_{-,\pi} = \text{ Mean Semi-RMAD with } \lambda = \frac{1}{\alpha} \\
=& \bar{J}_\pi - \frac{1}{2\alpha} \mathrm{MAD}(R, d_{\mu,\pi}) = \mathrm{MMAD}^{\lambda}(R, d_{\mu,\pi}) \text{ with } \lambda = \frac{1}{2\alpha}.
\end{aligned}$$

Now we prove (8):

$$\left(\mathrm{MAD}(R, d_{\mu,\pi})\right)^2 := \left(\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot) \\ a\sim\pi(\cdot|s)}} \left[\left| R(s,a) - \bar{J}_\pi \right| \right] \right)^2 = \left(\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot) \\ a\sim\pi(\cdot|s)}} \left[\sqrt{\left(R(s,a) - \bar{J}_\pi \right)^2} \right] \right)^2,$$

using the Jensen's inequality we obtain:

$$\begin{aligned}
\left(\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot) \\ a\sim\pi(\cdot|s)}} \left[\sqrt{\left(R(s,a) - \bar{J}_\pi \right)^2} \right] \right)^2 \leqslant& \left(\sqrt{\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot) \\ a\sim\pi(\cdot|s)}} \left[\left(R(s,a) - \bar{J}_\pi \right)^2 \right]} \right)^2 \\
=& \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot) \\ a\sim\pi(\cdot|s)}} \left[\left(R(s,a) - \bar{J}_\pi \right)^2 \right] \\
=& \mathrm{VaR}^{\alpha}(R, d_{\mu,\pi}),
\end{aligned}$$

so the squared RMAD is less than or equal to the Volatility:

$$\left(\mathrm{MAD}(R, d_{\mu,\pi})\right)^2 \leqslant \mathrm{VaR}^{\alpha}(R, d_{\mu,\pi}).$$

■

G.3. Positive homogeneity

Proposition G.1. *Given an MDP $\mathcal{M}_1$ and a risk measure η that satisfies the positive homogeneity property, if we multiply all the rewards of $\mathcal{M}_1$ by a positive quantity we obtain a new MDP $\mathcal{M}_2$ with the same optimal policies of $\mathcal{M}_1$ with respect to the risk measure η.*

Proof.
Given two policies π_1 and π_2 for which $\eta_{\pi_1}^{\mathcal{M}_1} < \eta_{\pi_2}^{\mathcal{M}_1}$, where $\eta_{\pi_1}^{\mathcal{M}_1}$ is the risk measure value of policy π_1 on MDP $\mathcal{M}_1$, if we multiply all rewards of the MDP $\mathcal{M}_1$ by a positive constant quantity $t \in \mathbb{R}^+$ we obtain:

$$\begin{aligned}
\eta_{\pi_1}^{\mathcal{M}_2} &= t\eta_{\pi_1}^{\mathcal{M}_1}, \\
\eta_{\pi_2}^{\mathcal{M}_2} &= t\eta_{\pi_2}^{\mathcal{M}_1},
\end{aligned}$$

thanks to the positive homogeneity property of the risk measure η. From $\eta_{\pi_1}^{\mathcal{M}_1} < \eta_{\pi_2}^{\mathcal{M}_1}$, we get:

$$t\eta_{\pi_1}^{\mathcal{M}_1} < t\eta_{\pi_2}^{\mathcal{M}_1},$$

which is:

$$\eta_{\pi_1}^{\mathcal{M}_2} < \eta_{\pi_2}^{\mathcal{M}_2},$$

so the partial relation between the two policies remains the same and this happens also with the equality $\eta_{\pi_1}^{\mathcal{M}_1} = \eta_{\pi_2}^{\mathcal{M}_1}$, thus the optimal policies with respect to the risk measure η of $\mathcal{M}_2$ are the same optimal policies of the original environment $\mathcal{M}_1$.

■

G.4. Translation equivariance

Proposition G.2. *Given an MDP $\mathcal{M}_1$ and a risk measure η that satisfies the translation equivariance property, if we add a constant quantity to all rewards of $\mathcal{M}_1$ we obtain a new MDP $\mathcal{M}_2$ with the same optimal policies of $\mathcal{M}_1$ with respect to the risk measure η.*

Proof.
Given two policies π_1 and π_2 for which $\eta_{\pi_1}^{\mathcal{M}_1} < \eta_{\pi_2}^{\mathcal{M}_1}$, where $\eta_{\pi_1}^{\mathcal{M}_1}$ is the risk measure value of policy π_1 on MDP $\mathcal{M}_1$, if we add a constant quantity $a \in \mathbb{R}$ to all rewards of the MDP $\mathcal{M}_1$ we obtain:

$$\eta_{\pi_1}^{\mathcal{M}_2} = \eta_{\pi_1}^{\mathcal{M}_1} + a,$$
$$\eta_{\pi_2}^{\mathcal{M}_2} = \eta_{\pi_2}^{\mathcal{M}_1} + a,$$

thanks to the translation equivariance property of the risk measure η. From $\eta_{\pi_1}^{\mathcal{M}_1} < \eta_{\pi_2}^{\mathcal{M}_1}$, we get:

$$\eta_{\pi_1}^{\mathcal{M}_1} + a < \eta_{\pi_2}^{\mathcal{M}_1} + a,$$

which is:

$$\eta_{\pi_1}^{\mathcal{M}_2} < \eta_{\pi_2}^{\mathcal{M}_2},$$

so the partial relation between the two policies remains the same and this happens also with the equality $\eta_{\pi_1}^{\mathcal{M}_1} = \eta_{\pi_2}^{\mathcal{M}_1}$, thus the optimal policies with respect to the risk measure η of $\mathcal{M}_2$ are the same optimal policies of the original environment $\mathcal{M}_1$.

■

G.5. Value functions of the RMAD and of the Mean-RMAD

The action-value function of the Mean-RMAD is:

$$\begin{aligned}
Q_\pi^\lambda(s,a) :=& Q_\pi(s,a) - \lambda X_\pi(s,a) \\
=& \mathop{\mathbb{E}}_{\substack{s_{t+1}\sim P(\cdot|s_t,a_t)\\ a_{t+1}\sim\pi(\cdot|s_{t+1})}} \left[\sum_{t=0}^{\infty} \gamma^t R(s_t,a_t) | s_0 = s, a_0 = a\right] \\
& - \lambda \mathop{\mathbb{E}}_{\substack{s_{t+1}\sim P(\cdot|s_t,a_t)\\ a_{t+1}\sim\pi(\cdot|s_{t+1})}} \left[\sum_{t=0}^{\infty} \gamma^t \left|R(s_t,a_t) - \bar{J}_\pi\right| | s_0 = s, a_0 = a\right] \\
=& \mathop{\mathbb{E}}_{\substack{s_{t+1}\sim P(\cdot|s_t,a_t)\\ a_{t+1}\sim\pi(\cdot|s_{t+1})}} \left[\sum_{t=0}^{\infty} \gamma^t R(s_t,a_t) | s_0 = s, a_0 = a\right] \\
& + \mathop{\mathbb{E}}_{\substack{s_{t+1}\sim P(\cdot|s_t,a_t)\\ a_{t+1}\sim\pi(\cdot|s_{t+1})}} \left[\sum_{t=0}^{\infty} \gamma^t \left(-\lambda \left|R(s_t,a_t) - \bar{J}_\pi\right|\right) | s_0 = s, a_0 = a\right] \\
=& \mathop{\mathbb{E}}_{\substack{s_{t+1}\sim P(\cdot|s_t,a_t)\\ a_{t+1}\sim\pi(\cdot|s_{t+1})}} \left[\sum_{t=0}^{\infty} \gamma^t R(s_t,a_t) + \sum_{t=0}^{\infty} \gamma^t \left(-\lambda \left|R(s_t,a_t) - \bar{J}_\pi\right|\right) | s_0 = s, a_0 = a\right] \\
=& \mathop{\mathbb{E}}_{\substack{s_{t+1}\sim P(\cdot|s_t,a_t)\\ a_{t+1}\sim\pi(\cdot|s_{t+1})}} \left[\sum_{t=0}^{\infty} \gamma^t \left(R(s_t,a_t) - \lambda \left|R(s_t,a_t) - \bar{J}_\pi\right|\right) | s_0 = s, a_0 = a\right]
\end{aligned}$$

and its state-value function is:

$$\begin{aligned}
V_\pi^\lambda(s) :=& \mathop{\mathbb{E}}_{\substack{a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{\infty} \gamma^t R(s_t,a_t) | s_0 = s\right] \\
& - \lambda \mathop{\mathbb{E}}_{\substack{a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{\infty} \gamma^t \left|R(s_t,a_t) - \bar{J}_\pi\right| | s_0 = s\right] \\
=& \mathop{\mathbb{E}}_{\substack{a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{\infty} \gamma^t \left(R(s_t,a_t) - \lambda \left|R(s_t,a_t) - \bar{J}_\pi\right|\right) | s_0 = s\right] \\
=& \mathop{\mathbb{E}}_{a\sim\pi(\cdot|s)} [Q_\pi^\lambda(s,a)].
\end{aligned}$$

From the previous definitions we obtain the Bellman equation of the Mean-RMAD after some steps

$$
\begin{aligned}
Q_\pi^\lambda(s,a) &= \mathop{\mathbb{E}}_{\substack{s_{t+1}\sim P(\cdot|s_t,a_t)\\ a_{t+1}\sim\pi(\cdot|s_{t+1})}} \left[\sum_{t=0}^{\infty}\gamma^t\left(R(s_t,a_t)-\lambda\left|R(s_t,a_t)-\bar{J}_\pi\right|\right)|s_0=s,a_0=a\right] \\
&= \mathop{\mathbb{E}}_{\substack{s_{t+1}\sim P(\cdot|s_t,a_t)\\ a_{t+1}\sim\pi(\cdot|s_{t+1})}} \Big[\left(R(s,a)-\lambda\left|R(s,a)-\bar{J}_\pi\right|\right) \\
&\quad +\sum_{t=1}^{\infty}\gamma^t\left(R(s_t,a_t)-\lambda\left|R(s_t,a_t)-\bar{J}_\pi\right|\right)|s_0=s,a_0=a\Big] \\
&= \mathop{\mathbb{E}}_{\substack{s_{t+1}\sim P(\cdot|s_t,a_t)\\ a_{t+1}\sim\pi(\cdot|s_{t+1})}} \left[\left(R(s,a)-\lambda\left|R(s,a)-\bar{J}_\pi\right|\right)\right] \\
&\quad + \mathop{\mathbb{E}}_{\substack{s_{t+1}\sim P(\cdot|s_t,a_t)\\ a_{t+1}\sim\pi(\cdot|s_{t+1})}} \left[\sum_{t=1}^{\infty}\gamma^t\left(R(s_t,a_t)-\lambda\left|R(s_t,a_t)-\bar{J}_\pi\right|\right)|s_0=s,a_0=a\right] \\
&= (1-\gamma)\left(R(s,a)-\lambda\left|R(s,a)-\bar{J}_\pi\right|\right) \\
&\quad + \mathop{\mathbb{E}}_{\substack{s_{t+1}\sim P(\cdot|s_t,a_t)\\ a_{t+1}\sim\pi(\cdot|s_{t+1})}} \left[\sum_{t=1}^{\infty}\gamma^t\left(R(s_t,a_t)-\lambda\left|R(s_t,a_t)-\bar{J}_\pi\right|\right)|s_1=s'\right] \\
&= (1-\gamma)\left(R(s,a)-\lambda\left|R(s,a)-\bar{J}_\pi\right|\right) \\
&\quad + \gamma \mathop{\mathbb{E}}_{\substack{a_t\sim\pi(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[\sum_{t=0}^{\infty}\gamma^t\left(R(s_t,a_t)-\lambda\left|R(s_t,a_t)-\bar{J}_\pi\right|\right)|s_0=s'\right] \\
&= \left(R(s,a)-\lambda\left|R(s,a)-\bar{J}_\pi\right|\right)+\gamma \mathop{\mathbb{E}}_{s'\sim P(\cdot|s,a)}[V_\pi^\lambda(s')]
\end{aligned}
$$

here:

$$
Q_\pi^\lambda(s,a) = \left(R(s,a)-\lambda\left|R(s,a)-\bar{J}_\pi\right|\right)+\gamma \mathop{\mathbb{E}}_{s'\sim P(\cdot|s,a)}[V_\pi^\lambda(s')].
$$

■

G.6. Proof of Theorem 4.1 (RMAD Policy Subgradient)

We follow a similar argument as in Theorem 2 of Bisi et al. (2020). First we need to prove:

$$
\omega_\pi = (1-\gamma)\int_{\mathcal{S}}\mu(s)W_\pi(s)ds,
$$

which requires Lemma 1 of Papini et al. (2019):

Lemma G.1. *Any integrable function $f:\mathcal{S}\to\mathbb{R}$ that can be recursively defined as:*

$$
f(s) = g(s)+\gamma\int_{\mathcal{S}}P^\pi(s'|s)f(s')ds',
$$

where $g:\mathcal{S}\to\mathbb{R}$ is any integrable function, is equal to:

$$
f(s) = \frac{1}{1-\gamma}\int_{\mathcal{S}}d_\pi(s'|s)g(s')ds'.
$$

From the definition of W_π we have, using Lemma G.1:

$$
W_\pi(s) = \mathop{\mathbb{E}}_{a\sim\pi(\cdot|s)}\left[\left|R(s,a)-\bar{J}_\pi\right|\right]+\gamma \mathop{\mathbb{E}}_{s'\sim P^\pi(\cdot|s)}[W_\pi(s')]] = \frac{1}{(1-\gamma)}\mathop{\mathbb{E}}_{\substack{s'\sim d_\pi(\cdot|s)\\ a\sim\pi(\cdot|s)}}\left[|R(s',a)-J_\pi|\right],
$$

from which we obtain:

$$
\omega_\pi = \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}}\left[\left|R(s,a)-\bar{J}_\pi\right|\right] = (1-\gamma)\mathop{\mathbb{E}}_{s\sim\mu}[W_\pi(s)] = (1-\gamma)\int_{\mathcal{S}}\mu(s)W_\pi(s)ds.
$$

Now we can consider the gradient of the value functions of the RMAD, $\forall s, a \in \mathcal{S} \times \mathcal{A}$:

$$\partial_{\boldsymbol{\theta}} X_\pi(s,a) \ni -(1-\gamma)\operatorname{sign}(R(s,a)-\bar{J}_\pi)\nabla \bar{J}_\pi + \gamma \mathop{\mathbb{E}}_{s'\sim P(\cdot|s,a)}[\nabla_{\boldsymbol{\theta}} W_\pi(s')],$$

$$\begin{aligned}
\partial_{\boldsymbol{\theta}} W_\pi(s) =& \partial_{\boldsymbol{\theta}} \int_{\mathcal{A}} \pi_{\boldsymbol{\theta}}(a|s) X_\pi(s,a) da \\
=& \int_{\mathcal{A}} \left[\nabla_{\boldsymbol{\theta}} \pi_{\boldsymbol{\theta}}(a|s) X_\pi(s,a) + \pi_{\boldsymbol{\theta}}(a|s)\partial_{\boldsymbol{\theta}} X_\pi(s,a)\right] da \\
\ni& \int_{\mathcal{A}} \left[\nabla_{\boldsymbol{\theta}} \pi_{\boldsymbol{\theta}}(a|s) X_\pi(s,a) - \pi_{\boldsymbol{\theta}}(a|s)(1-\gamma)\operatorname{sign}(R(s,a)-\bar{J}_\pi)\nabla \bar{J}_\pi\right] da \\
&+ \gamma \int_{\mathcal{S}} P^\pi(s'|s)\partial_{\boldsymbol{\theta}} W_\pi(s') ds',
\end{aligned}$$

we use lemma G.1

$$= \frac{1}{1-\gamma}\int_{\mathcal{S}} d_\pi(s'|s)\int_{\mathcal{A}} \left[\nabla_{\boldsymbol{\theta}} \pi_{\boldsymbol{\theta}}(a|s') X_\pi(s',a) - \pi_{\boldsymbol{\theta}}(a|s')(1-\gamma)\operatorname{sign}(R(s',a)-\bar{J}_\pi)\nabla \bar{J}_\pi\right] da ds'.$$

To obtain the subgradient of the RMAD we use $\omega_\pi = \int_{\mathcal{S}} \mu(s) W_\pi(s) ds$:

$$\begin{aligned}
\partial_{\boldsymbol{\theta}} \omega_\pi &= \int_{\mathcal{S}} \mu(s)\partial_{\boldsymbol{\theta}} W_\pi(s) ds \\
&\ni \frac{1}{1-\gamma}\int_{\mathcal{S}} d_{\mu,\pi}(s)\int_{\mathcal{A}} \left[\nabla_{\boldsymbol{\theta}} \pi_{\boldsymbol{\theta}}(a|s) X_\pi(s,a) - \pi_{\boldsymbol{\theta}}(a|s)(1-\gamma)\operatorname{sign}(R(s,a)-\bar{J}_\pi)\nabla \bar{J}_\pi\right] da ds \\
&= \frac{1}{1-\gamma}\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}}\left[\nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a|s) X_\pi(s,a)\right] - \nabla \bar{J}_\pi \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}}\left[\operatorname{sign}(R(s,a)-\bar{J}_\pi)\right] \\
&= \mathop{\mathbb{E}}_{\tau\sim p_\pi(\cdot)}\left[\sum_{t=0}^{\infty}\gamma^t \nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a_t|s_t) X_\pi(s_t,a_t)\right] - \nabla \bar{J}_\pi (1-\gamma)\mathop{\mathbb{E}}_{\tau\sim p_\pi(\cdot)}\left[\sum_{t=0}^{\infty}\gamma^t \operatorname{sign}(R(s_t,a_t)-\bar{J}_\pi)\right] \\
&= \mathop{\mathbb{E}}_{\tau\sim p_\pi(\cdot)}\left[\sum_{t=0}^{\infty}\gamma^t \nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a_t|s_t) X_\pi(s_t,a_t)\right] - \nabla \bar{J}_\pi \psi_\pi \\
&= \mathop{\mathbb{E}}_{\tau\sim p_\pi(\cdot)}\left[\sum_{t=0}^{\infty}\gamma^t \nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a_t|s_t) X_\pi(s_t,a_t)\right] - \psi_\pi \mathop{\mathbb{E}}_{\tau\sim p_\pi(\cdot)}\left[\sum_{t=0}^{\infty}\gamma^t \nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a_t|s_t) Q_\pi(s_t,a_t)\right] \\
&= \mathop{\mathbb{E}}_{\tau\sim p_\pi(\cdot)}\left[\sum_{t=0}^{\infty}\gamma^t \nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a_t|s_t)\left(X_\pi(s_t,a_t) - \psi_\pi Q_\pi(s_t,a_t)\right)\right],
\end{aligned}$$

where ψ_π is what we call *mean sign deviation*:

$$\psi_\pi \coloneqq \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}}\left[\operatorname{sign}(R(s,a)-\bar{J}_\pi)\right].$$

■

G.7. Proof of Lemma 4.1 (Mean-RMAD Performance Difference Lemma)

We start from the objective function:

$$\eta_{\tilde{\pi}}^\lambda = (1-\gamma)\mathop{\mathbb{E}}_{\tau|\tilde{\pi}}\left[\sum_t \gamma^t (R(s_t,a_t) - \lambda|R(s_t,a_t) - \bar{J}_{\tilde{\pi}}|)\right]$$

$$= (1-\gamma)\mathop{\mathbb{E}}_{\tau|\tilde{\pi}}\left[V_\pi^\lambda(s_0) - V_\pi^\lambda(s_0) + \sum_t \gamma^t (R(s_t,a_t) - \lambda|R(s_t,a_t) - \bar{J}_{\tilde{\pi}}|)\right]$$

$$= \eta_\pi^\lambda + (1-\gamma)\mathop{\mathbb{E}}_{\tau|\tilde{\pi}}\left[\sum_t \gamma^t (R(s_t,a_t) - \lambda|R(s_t,a_t) - \bar{J}_{\tilde{\pi}}| + \gamma V_\pi^\lambda(s_{t+1}) - V_\pi^\lambda(s_t))\right].$$

We use the result in Kakade and Langford (2002) :

$$\bar{J}_{\tilde{\pi}} = \bar{J}_\pi + (1-\gamma)\mathop{\mathbb{E}}_{\tau|\tilde{\pi}}\left[\sum_t \gamma^t A_\pi(s_t,a_t)\right]$$

here

$$|R(s_t,a_t)-\bar{J}_{\widetilde{\pi}}| = \left|R(s_t,a_t)-\bar{J}_{\pi}-(1-\gamma)\mathop{\mathbb{E}}_{\tau|\widetilde{\pi}}\left[\sum_t \gamma^t A_\pi(s_t,a_t)\right]\right|$$
$$\leqslant |R(s_t,a_t)-\bar{J}_{\pi}| + \left|(1-\gamma)\mathop{\mathbb{E}}_{\tau|\widetilde{\pi}}\left[\sum_t \gamma^t A_\pi(s_t,a_t)\right]\right|.$$

We return to the objective function:

$$\begin{aligned}\eta^\lambda_{\widetilde{\pi}} - \eta^\lambda_\pi \geqslant &(1-\gamma)\mathop{\mathbb{E}}_{\tau|\widetilde{\pi}}\left[\sum_t \gamma^t (R(s_t,a_t)-\lambda|R(s_t,a_t)-\bar{J}_\pi| - \lambda\left|(1-\gamma)\mathop{\mathbb{E}}_{\tau|\widetilde{\pi}}\left[\sum_t \gamma^t A_\pi(s_t,a_t)\right]\right|\right.\\ &\left.+\gamma V^\lambda_\pi(s_{t+1}) - V^\lambda_\pi(s_t))\right]\\ =&(1-\gamma)\mathop{\mathbb{E}}_{\tau|\widetilde{\pi}}\left[\sum_t \gamma^t (R(s_t,a_t)-\lambda|R(s_t,a_t)-\bar{J}_\pi| + \gamma V^\lambda_\pi(s_{t+1}) - V^\lambda_\pi(s_t))\right]\\ &-\lambda(1-\gamma)\mathop{\mathbb{E}}_{\tau|\widetilde{\pi}}\left[\sum_t \gamma^t \left|(1-\gamma)\mathop{\mathbb{E}}_{\tau|\widetilde{\pi}}\left[\sum_t \gamma^t A_\pi(s_t,a_t)\right]\right|\right]\\ =&(1-\gamma)\mathop{\mathbb{E}}_{\tau|\widetilde{\pi}}\left[\sum_t \gamma^t A^\lambda_\pi(s_t,a_t)\right] - \lambda(1-\gamma)\left|\mathop{\mathbb{E}}_{\tau|\widetilde{\pi}}\left[\sum_t \gamma^t A_\pi(s_t,a_t)\right]\right|.\end{aligned}$$

■

G.8. Proof of Theorem 4.2 (Mean-RMAD Safe Improvement Bound)

We apply Theorem 1 from Schulman et al. (2017a) to the Mean-RMAD version of the Performance Difference Lemma:[6]

$$\eta^\lambda_{\widetilde{\pi}} \geqslant L^\lambda_\pi(\widetilde{\pi}) - \frac{4\gamma\epsilon_\lambda}{1-\gamma}\alpha^2_{KL} - \lambda(1-\gamma)\left|\mathop{\mathbb{E}}_{\tau|\widetilde{\pi}}\left[\sum_{t=0}^{\infty}\gamma^t A_\pi(s_t,a_t)\right]\right|,$$

where $\epsilon_\lambda = \max_{s,a}\left|A^\lambda_\pi(s,a)\right|$,

$$L^\lambda_\pi(\widetilde{\pi}) \coloneqq \eta^\lambda_\pi + \int_{\mathcal{S}} d_{\mu,\pi}(s)\int_{\mathcal{A}}\widetilde{\pi}(a|s)A^\lambda_\pi(s,a)\,\mathrm{d}a\,\mathrm{d}s,$$
$$\alpha^2_{KL} = D^{max}_{KL}(\pi,\widetilde{\pi}) = \max_s D_{KL}(\pi(\cdot|s),\widetilde{\pi}(\cdot|s))$$

and D_{KL} is the Kullback-Leibler divergence.
Since the last term depends on the trajectory obtained with the new policy we need to find a bound $M \geqslant 0$ such that:

$$M \geqslant \left|\mathop{\mathbb{E}}_{\tau|\widetilde{\pi}}\left[\sum_{t=0}^{\infty}\gamma^t A_\pi(s_t,a_t)\right]\right|.$$

Let us call:

$$\bar{A}(s) \coloneqq \mathop{\mathbb{E}}_{a\sim\widetilde{\pi}(\cdot,s)}[A_\pi(s,a)],$$
$$A^{\widetilde{\pi}}_\pi \coloneqq \mathop{\mathbb{E}}_{\tau|\pi}\left[\sum_{t=0}^{\infty}\gamma^t \mathop{\mathbb{E}}_{a\sim\widetilde{\pi}(\cdot,s_t)}[A_\pi(s_t,a)]\right] = \mathop{\mathbb{E}}_{\tau|\pi}\left[\sum_{t=0}^{\infty}\gamma^t \bar{A}(s_t)\right],$$
$$A \coloneqq \mathop{\mathbb{E}}_{\tau|\widetilde{\pi}}\left[\sum_{t=0}^{\infty}\gamma^t A_\pi(s_t,a_t)\right].$$

We want a relation for:

$$\begin{aligned}\left|A - A^{\widetilde{\pi}}_\pi\right| &= \left|\mathop{\mathbb{E}}_{\tau|\widetilde{\pi}}\left[\sum_{t=0}^{\infty}\gamma^t A_\pi(s_t,a_t)\right] - \mathop{\mathbb{E}}_{\tau|\pi}\left[\sum_{t=0}^{\infty}\gamma^t \bar{A}(s_t)\right]\right| = \left|\sum_{t=0}^{\infty}\gamma^t\left(\mathop{\mathbb{E}}_{\tau|\widetilde{\pi}}\left[A_\pi(s_t,a_t)\right] - \mathop{\mathbb{E}}_{\tau|\pi}\left[\bar{A}(s_t)\right]\right)\right|\\ &= \left|\sum_{t=0}^{\infty}\gamma^t\left(\mathop{\mathbb{E}}_{\tau|\widetilde{\pi}}\left[\bar{A}(s_t)\right] - \mathop{\mathbb{E}}_{\tau|\pi}\left[\bar{A}(s_t)\right]\right)\right| \leqslant \sum_{t=0}^{\infty}\gamma^t\left|\mathop{\mathbb{E}}_{\tau|\widetilde{\pi}}\left[\bar{A}(s_t)\right] - \mathop{\mathbb{E}}_{\tau|\pi}\left[\bar{A}(s_t)\right]\right|,\end{aligned}$$

[6] Comparing this bound to the results shown in the paper, the denominator term is not squared due to the normalization of the return.

now we use Lemma 3 from (Schulman et al., 2015):

$$\leqslant \sum_{t=0}^{\infty} \gamma^t 4\epsilon\alpha(1-(1-\alpha)^t) \leqslant \frac{4\epsilon\gamma\alpha_{KL}^2}{(1-\gamma)^2},$$

where $\epsilon = \max_{s,a} |A_\pi(s,a)|$.
From this we can obtain:

$$|A| = |A + A_\pi^{\widetilde{\pi}} - A_\pi^{\widetilde{\pi}}| \leqslant |A_\pi^{\widetilde{\pi}}| + |A - A_\pi^{\widetilde{\pi}}| \leqslant |A_\pi^{\widetilde{\pi}}| + \frac{4\epsilon\gamma}{(1-\gamma)^2}\alpha_{KL}^2.$$

So we can set $M \coloneqq |A_\pi^{\widetilde{\pi}}| + \frac{4\epsilon\gamma}{(1-\gamma)^2}\alpha_{KL}^2$ to obtain:

$$\eta_{\widetilde{\pi}}^{\lambda} \geqslant L_\pi^\lambda(\widetilde{\pi}) - \frac{4\gamma\epsilon_\lambda}{1-\gamma}\alpha_{KL}^2 - \lambda(1-\gamma)M.$$

■

G.9. Proof of Corollary 4.1 (Monotonic Improvement of RMAD-TRPO)

Let us call $B_k(\pi)$:

$$B_k(\pi) = L_k^\lambda(\pi) - \frac{4\gamma\epsilon_\lambda}{1-\gamma}\alpha_{KL}^2 - \lambda(1-\gamma)M = L_k^\lambda(\pi) - \frac{4\gamma\epsilon_\lambda}{1-\gamma}\alpha_{KL}^2 - \lambda(1-\gamma)|A_{\boldsymbol{\theta}_k}^{\boldsymbol{\theta}}| - \lambda\frac{4\epsilon\gamma}{(1-\gamma)}\alpha_{KL}^2$$

the Mean-RMAD Safe Improvement Bound of policy π.
We have that $\eta_{\pi_k}^\lambda = B_k(\pi_k)$ because:

- $L_k^\lambda(\pi_k) = \eta_{\pi_k}$ due to $\int_{\mathcal{A}} \pi_k(a|s) A_{\pi_k}^\lambda(s,a)\ \mathrm{d}a = 0$;
- $-\frac{4\gamma\epsilon_\lambda}{1-\gamma}\alpha_{KL}^2$ and $-\lambda\frac{4\epsilon\gamma}{(1-\gamma)}\alpha_{KL}^2$ are zero because $\alpha_{KL}^2 = D_{KL}^{max}(\pi_k,\pi_k) = 0$;
- $|A_{\pi_k}^{\pi_k}|$ is zero because $\mathbb{E}_{a\sim\pi_k(\cdot,s)}[A_{\pi_k}(s,a)] = 0$.

From this last fact and from $\eta_{\pi_{k+1}}^\lambda \geqslant B_k(\pi_{k+1})$ of Mean-RMAD the Safe Improvement Bound we obtain:

$$\eta_{\pi_{k+1}}^\lambda - \eta_{\pi_k}^\lambda \geqslant B_k(\pi_{k+1}) - B_k(\pi_k).$$

Thus, by maximizing B_k at each iteration k, we guarantee that the Mean-RMAD is non-decreasing.

■

G.10. Proof of Proposition 4.1

Let us find a subgradient of:

$$\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot)\\ a\sim\pi_{\boldsymbol{\theta}_k}(\cdot|s)}}\left[\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)}A_{\boldsymbol{\theta}_k}^\lambda(s,a)\right] - \lambda\left|\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot)\\ a\sim\pi_{\boldsymbol{\theta}_k}(\cdot|s)}}\left[\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)}A_{\boldsymbol{\theta}_k}(s,a)\right]\right|. \tag{13}$$

The first term is differentiable so we can consider its gradient:

$$\nabla_{\boldsymbol{\theta}} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot)\\ a\sim\pi_{\boldsymbol{\theta}_k}(\cdot|s)}}\left[\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)}A_{\boldsymbol{\theta}_k}^\lambda(s,a)\right] = \nabla_{\boldsymbol{\theta}}\int_{\mathcal{S}} d_{\mu,\pi_{\boldsymbol{\theta}_k}}(s)\int_{\mathcal{A}}\pi_{\boldsymbol{\theta}}(a|s)A_{\boldsymbol{\theta}_k}^\lambda(s,a)\ \mathrm{d}a\ \mathrm{d}s,$$

we decompose the advantage function with $A_\pi^\lambda(s,a) = A_\pi(s,a) + \lambda A_\pi^\omega(s,a)$:

$$= \nabla_{\boldsymbol{\theta}}\int_{\mathcal{S}} d_{\mu,\pi_{\boldsymbol{\theta}_k}}(s)\int_{\mathcal{A}}\pi_{\boldsymbol{\theta}}(a|s)A_{\boldsymbol{\theta}_k}(s,a)\ \mathrm{d}a\ \mathrm{d}s + \lambda\nabla_{\boldsymbol{\theta}}\int_{\mathcal{S}} d_{\mu,\pi_{\boldsymbol{\theta}_k}}(s)\int_{\mathcal{A}}\pi_{\boldsymbol{\theta}}(a|s)A_{\boldsymbol{\theta}_k}^\omega(s,a)\ \mathrm{d}a\ \mathrm{d}s. \tag{14}$$

Regarding the risk-neutral advantage function we recover the gradient of the normalized expected return:

$$\begin{aligned}
\nabla_{\boldsymbol{\theta}}\int_{\mathcal{S}} d_{\mu,\pi_{\boldsymbol{\theta}_k}}(s)\int_{\mathcal{A}}\pi_{\boldsymbol{\theta}}(a|s)A_{\boldsymbol{\theta}_k}(s,a)\ \mathrm{d}a\ \mathrm{d}s &= \int_{\mathcal{S}} d_{\mu,\pi_{\boldsymbol{\theta}_k}}(s)\int_{\mathcal{A}}\nabla_{\boldsymbol{\theta}}\pi_{\boldsymbol{\theta}}(a|s)A_{\boldsymbol{\theta}_k}(s,a)\ \mathrm{d}a\ \mathrm{d}s \\
&= \int_{\mathcal{S}} d_{\mu,\pi_{\boldsymbol{\theta}_k}}(s)\int_{\mathcal{A}}\nabla_{\boldsymbol{\theta}}\pi_{\boldsymbol{\theta}}(a|s)(Q_{\boldsymbol{\theta}_k}(s,a) - V_{\boldsymbol{\theta}_k}(s))\ \mathrm{d}a\ \mathrm{d}s \\
&= \int_{\mathcal{S}} d_{\mu,\pi_{\boldsymbol{\theta}_k}}(s)\int_{\mathcal{A}}\nabla_{\boldsymbol{\theta}}\pi_{\boldsymbol{\theta}}(a|s)Q_{\boldsymbol{\theta}_k}(s,a)\ \mathrm{d}a\ \mathrm{d}s,
\end{aligned}$$

because $\int_{\mathcal{S}} d_{\mu,\pi_{\boldsymbol{\theta}_k}}(s) \int_{\mathcal{A}} \nabla_{\boldsymbol{\theta}} \pi_{\boldsymbol{\theta}}(a|s) V_{\boldsymbol{\theta}_k}(s) \, \mathrm{d}a \, \mathrm{d}s = \int_{\mathcal{S}} d_{\mu,\pi_{\boldsymbol{\theta}_k}}(s) V_{\boldsymbol{\theta}_k}(s) \int_{\mathcal{A}} \nabla_{\boldsymbol{\theta}} \pi_{\boldsymbol{\theta}}(a|s) \, \mathrm{d}a \, \mathrm{d}s = 0$, now we use the log trick:

$$= \int_{\mathcal{S}} d_{\mu,\pi_{\boldsymbol{\theta}_k}}(s) \int_{\mathcal{A}} \pi_{\boldsymbol{\theta}}(a|s) \nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a|s) Q_{\boldsymbol{\theta}_k}(s,a) \, \mathrm{d}a \, \mathrm{d}s,$$

if we evaluate it in $\boldsymbol{\theta} = \boldsymbol{\theta}_k$ we obtain:

$$\int_{\mathcal{S}} d_{\mu,\pi_{\boldsymbol{\theta}_k}}(s) \int_{\mathcal{A}} \pi_{\boldsymbol{\theta}_k}(a|s) \left(\nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a|s)\right)|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k} Q_{\boldsymbol{\theta}_k}(s,a) \, \mathrm{d}a \, \mathrm{d}s = \nabla_{\boldsymbol{\theta}} \bar{J}_{\boldsymbol{\theta}}|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k}.$$

Analogously we obtain for the second term of the formula 14:

$$\nabla_{\boldsymbol{\theta}} \lambda \int_{\mathcal{S}} d_{\mu,\pi_{\boldsymbol{\theta}_k}}(s) \int_{\mathcal{A}} \pi_{\boldsymbol{\theta}}(a|s) A^{\omega}_{\boldsymbol{\theta}_k}(s,a) \, \mathrm{d}a \, \mathrm{d}s = -\lambda \int_{\mathcal{S}} d_{\mu,\pi_{\boldsymbol{\theta}_k}}(s) \int_{\mathcal{A}} \pi_{\boldsymbol{\theta}}(a|s) \nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a|s) X_{\boldsymbol{\theta}_k}(s,a) \, \mathrm{d}a \, \mathrm{d}s,$$

if we evaluate it in $\boldsymbol{\theta} = \boldsymbol{\theta}_k$ we obtain:

$$-\lambda \int_{\mathcal{S}} d_{\mu,\pi_{\boldsymbol{\theta}_k}}(s) \int_{\mathcal{A}} \pi_{\boldsymbol{\theta}_k}(a|s) \left(\nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a|s)\right)|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k} X_{\boldsymbol{\theta}_k}(s,a) \, \mathrm{d}a \, \mathrm{d}s = -\lambda \nabla_{\boldsymbol{\theta}} \omega_{\boldsymbol{\theta}}|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k} - \lambda \psi_{\boldsymbol{\theta}_k} \nabla_{\boldsymbol{\theta}} \bar{J}^{\boldsymbol{\theta}}|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k}.$$

Now we consider the second term of the objective function 13:

$$\begin{aligned}
&\nabla_{\boldsymbol{\theta}} \left| \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot) \\ a\sim\pi_{\boldsymbol{\theta}_k}(\cdot|s)}} \left[\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)} A_{\boldsymbol{\theta}_k}(s,a)\right] \right| = \\
&= \operatorname{sign}\left(\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot) \\ a\sim\pi_{\boldsymbol{\theta}_k}(\cdot|s)}} \left[\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)} A_{\boldsymbol{\theta}_k}(s,a)\right] \right) \nabla_{\boldsymbol{\theta}} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot) \\ a\sim\pi_{\boldsymbol{\theta}_k}(\cdot|s)}} \left[\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)} A_{\boldsymbol{\theta}_k}(s,a)\right] \\
&= \operatorname{sign}\left(\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot) \\ a\sim\pi_{\boldsymbol{\theta}_k}(\cdot|s)}} \left[\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)} A_{\boldsymbol{\theta}_k}(s,a)\right] \right) \int_{\mathcal{S}} d_{\mu,\pi_{\boldsymbol{\theta}_k}}(s) \int_{\mathcal{A}} \nabla_{\boldsymbol{\theta}} \pi_{\boldsymbol{\theta}}(a|s) Q_{\boldsymbol{\theta}_k}(s,a) \, \mathrm{d}a \, \mathrm{d}s,
\end{aligned}$$

but the gradient doesn't exist in $\boldsymbol{\theta} = \boldsymbol{\theta}_k$, because in $\boldsymbol{\theta} = \boldsymbol{\theta}_k$ the term inside the absolute value becomes

$$\begin{aligned}
\mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot) \\ a\sim\pi_{\boldsymbol{\theta}_k}(\cdot|s)}} \left[\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)} A_{\boldsymbol{\theta}_k}(s,a)\right] &= \mathop{\mathbb{E}}_{s\sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot)} \mathop{\mathbb{E}}_{a\sim\pi_{\boldsymbol{\theta}_k}(\cdot|s)} [A_{\boldsymbol{\theta}_k}(s,a)] \\
&= \mathop{\mathbb{E}}_{s\sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot)} \mathop{\mathbb{E}}_{a\sim\pi_{\boldsymbol{\theta}_k}(\cdot,s)} [Q_{\boldsymbol{\theta}_k}(s,a) - V_{\boldsymbol{\theta}_k}(s)] \\
&= \mathop{\mathbb{E}}_{s\sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot)} \left[\mathop{\mathbb{E}}_{a\sim\pi_{\boldsymbol{\theta}_k}(\cdot,s)} [Q_{\boldsymbol{\theta}_k}(s,a)] - V_{\boldsymbol{\theta}_k}(s) \right] \\
&= \mathop{\mathbb{E}}_{s\sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot)} [V_{\boldsymbol{\theta}_k}(s) - V_{\boldsymbol{\theta}_k}(s)] = 0.
\end{aligned}$$

So we have to use the subgradient method. In $\boldsymbol{\theta} = \boldsymbol{\theta}_k$ and using the log trick,
$\int_{\mathcal{S}} d_{\mu,\pi_{\boldsymbol{\theta}_k}}(s) \int_{\mathcal{A}} \nabla_{\boldsymbol{\theta}} \pi_{\boldsymbol{\theta}}(a|s) Q_{\boldsymbol{\theta}_k}(s,a) \, \mathrm{d}a \, \mathrm{d}s$ becomes $\nabla_{\boldsymbol{\theta}} \bar{J}_{\boldsymbol{\theta}}|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k}$, so the set of the subgradients in $\boldsymbol{\theta} = \boldsymbol{\theta}_k$ is:

$$\left[-\nabla_{\boldsymbol{\theta}} \bar{J}_{\boldsymbol{\theta}}|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k}; \nabla_{\boldsymbol{\theta}} \bar{J}_{\boldsymbol{\theta}}|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k}\right].$$

We can take $-\psi_{\boldsymbol{\theta}_k} \nabla_{\boldsymbol{\theta}} \bar{J}_{\boldsymbol{\theta}}|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k}$ since $-1 \leqslant \psi_{\boldsymbol{\theta}_k} \leqslant 1$. Finally, one possible subgradient of the objective function 13 is:

$$\nabla_{\boldsymbol{\theta}} \bar{J}_{\boldsymbol{\theta}}|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k} - \lambda \nabla_{\boldsymbol{\theta}} \omega_{\boldsymbol{\theta}}|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k} - \lambda \psi_{\boldsymbol{\theta}_k} \nabla_{\boldsymbol{\theta}} \bar{J}_{\boldsymbol{\theta}}|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k} + \lambda \psi_{\boldsymbol{\theta}_k} \nabla_{\boldsymbol{\theta}} \bar{J}_{\boldsymbol{\theta}}|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k} = \nabla_{\boldsymbol{\theta}} \bar{J}_{\boldsymbol{\theta}}|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k} - \lambda \nabla_{\boldsymbol{\theta}} \omega_{\boldsymbol{\theta}}|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k} = \nabla_{\boldsymbol{\theta}} \eta^{\lambda}_{\boldsymbol{\theta}}|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k}.$$

■

G.11. Proof of Proposition D.1

$$
\begin{aligned}
f(t\rho_1 + (1-t)\rho_2) =& t\rho_1 + (1-t)\rho_2 - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}} \left[(R(s,a) - t\rho_1 - (1-t)\rho_2)_-\right] \\
=& t\rho_1 + (1-t)\rho_2 - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}} \left[(tR(s,a) - t\rho_1 + (1-t)R(s,a) - (1-t)\rho_2)_-\right] \\
\geqslant& t\rho_1 + (1-t)\rho_2 - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}} \left[(tR(s,a) - t\rho_1)_- + ((1-t)R(s,a) - (1-t)\rho_2)_-\right] \\
=& t\rho_1 + (1-t)\rho_2 - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}} \left[(tR(s,a) - t\rho_1)_-\right] - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}} \left[((1-t)R(s,a) - (1-t)\rho_2)_-\right] \\
=& t\rho_1 - t\frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}} \left[(R(s,a) - \rho_1)_-\right] + (1-t)\rho_2 - (1-t)\frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}} \left[(R(s,a) - \rho_2)_-\right] \\
=& tf(\rho_1) + (1-t)f(\rho_2).
\end{aligned}
$$

■

G.12. Proof of Theorem 5.1 (Monotonic Policy Improvement for block cyclic coordinate ascent)

The proof is similar to the proof of proposition 1 of Zhang et al. (2020):

$$
\begin{aligned}
\eta^{\alpha}_{\pi_{k+1}}[R(s,a)] =& \max_{\rho} \left\{ \rho - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi_{k+1}}(\cdot)\\ a\sim\pi_{k+1}(\cdot|s)}} \left[(R(s,a) - \rho)_-\right] \right\} \\
\geqslant& \rho^{\alpha}_{\pi_k}[R(s,a)] - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi_{k+1}}(\cdot)\\ a\sim\pi_{k+1}(\cdot|s)}} \left[\left(R(s,a) - \rho^{\alpha}_{\pi_k}[R(s,a)]\right)_-\right] \\
\geqslant& \rho^{\alpha}_{\pi_k}[R(s,a)] - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi_{k}}(\cdot)\\ a\sim\pi_{k}(\cdot|s)}} \left[\left(R(s,a) - \rho^{\alpha}_{\pi_k}[R(s,a)]\right)_-\right]
\end{aligned}
$$

(because π_{k+1} is the maximizer of that expression with the value at risk fixed)

$$
\begin{aligned}
=& \max_{\rho} \left\{ \rho - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi_{k}}(\cdot)\\ a\sim\pi_{k}(\cdot|s)}} \left[(R(s,a) - \rho)_-\right] \right\} \\
=& \eta^{\alpha}_{\pi_k}[R(s,a)].
\end{aligned}
$$

■

G.13. Proof of Lemma 5.1 (RCVaR Performance Difference Lemma)

$$
\begin{aligned}
\eta^{\alpha}_{\widetilde{\pi}} - \eta^{\alpha}_{\pi} =& \max_{\rho} \left\{ \rho - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\widetilde{\pi}}(\cdot)\\ a\sim\widetilde{\pi}(\cdot|s)}} \left[(R(s,a) - \rho)_-\right] \right\} - \max_{\rho} \left\{ \rho - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot)\\ a\sim\pi(\cdot|s)}} \left[(R(s,a) - \rho)_-\right] \right\} \\
=& * .
\end{aligned}
$$

Let us consider $\rho_\pi^\alpha = \arg\max_\rho \left\{ \rho - \frac{1}{\alpha} \mathbb{E}_{\substack{s\sim d_{\mu,\pi}(\cdot) \\ a\sim\pi(\cdot|s)}} \left[(R(s,a) - \rho)_- \right] \right\}$ (which is the RVaR):

$$
\begin{aligned}
* \geqslant & \rho_\pi^\alpha - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\widetilde{\pi}}(\cdot) \\ a\sim\widetilde{\pi}(\cdot|s)}} \left[(R(s,a) - \rho_\pi^\alpha)_- \right] - \rho_\pi^\alpha + \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot) \\ a\sim\pi(\cdot|s)}} \left[(R(s,a) - \rho_\pi^\alpha)_- \right] \\
= & - \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\widetilde{\pi}}(\cdot) \\ a\sim\widetilde{\pi}(\cdot|s)}} \left[(R(s,a) - \rho_\pi^\alpha)_- \right] + \frac{1}{\alpha} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot) \\ a\sim\pi(\cdot|s)}} \left[(R(s,a) - \rho_\pi^\alpha)_- \right] \\
= & \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\widetilde{\pi}}(\cdot) \\ a\sim\widetilde{\pi}(\cdot|s)}} \left[-\frac{1}{\alpha} (R(s,a) - \rho_\pi^\alpha)_- \right] - \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi}(\cdot) \\ a\sim\pi(\cdot|s)}} \left[-\frac{1}{\alpha} (R(s,a) - \rho_\pi^\alpha)_- \right] \\
= & J_{\widetilde{\pi}}^{\rho^\alpha} - J_\pi^{\rho^\alpha} \\
= & (1-\gamma) \mathop{\mathbb{E}}_{\tau|\widetilde{\pi}} \left[\sum_t \gamma^t A_\pi^{\rho^\alpha}(s_t, a_t) \right],
\end{aligned}
$$

where $J_\pi^{\rho^\alpha}$ is the normalized expected return of the modified MDP with reward $\widetilde{R} = -\frac{1}{\alpha}(R - \rho^\alpha)_-$ following policy π.

■

G.14. Proof of Theorem 5.2 (RCVaR Safe Improvement Bound)

We apply Theorem 1 from Schulman et al. (2017a) to the RCVaR version of the Performance Difference Lemma (Lemma 5.1), obtaining:[7]

$$
\eta_{\widetilde{\pi}}^\alpha \geqslant \eta_\pi^\alpha + (1-\gamma) \mathop{\mathbb{E}}_{\tau|\pi} \left[\sum_{t=0}^{\infty} \gamma^t \mathop{\mathbb{E}}_{a\sim\widetilde{\pi}(\cdot,s_t)} \left[A_\pi^{\rho^\alpha}(s_t, a_t) \right] \right] - \frac{4\gamma\epsilon}{(1-\gamma)} \alpha_{KL}^2,
$$

where $\epsilon_\lambda = \max_{s,a} \left| A_\pi^{\rho^\alpha}(s,a) \right|$.

■

G.15. Proof of Corollary 5.1 (Monotonic Improvement of RCVaR-TRPO)

Let us call

$$
B_k(\pi) = \eta_{\pi_k}^\alpha + \frac{1}{\alpha}(1-\gamma) \mathop{\mathbb{E}}_{\tau|\pi_k} \left[\sum_{t=0}^{\infty} \gamma^t \mathop{\mathbb{E}}_{a\sim\pi(\cdot,s_t)} \left[A_{\pi_k}^{\rho^\alpha}(s_t, a_t) \right] \right] - \frac{1}{\alpha} \frac{4\gamma\epsilon}{(1-\gamma)} \alpha_{KL}^2.
$$

We have that $\eta_{\pi_k}^\alpha = B_k(\pi_k)$ because:

- $\frac{1}{\alpha}(1-\gamma)\, \mathbb{E}_{\tau|\pi_k} \left[\sum_{t=0}^{\infty} \gamma^t\, \mathbb{E}_{a\sim\pi_k(\cdot,s_t)} \left[A_{\pi_k}^{\rho^\alpha}(s_t, a_t) \right] \right]$ is zero because $\mathbb{E}_{a\sim\pi_k(\cdot,s_t)} \left[A_{\pi_k}^{\rho^\alpha}(s_t, a_t) \right] = 0$;
- $-\frac{1}{\alpha} \frac{4\gamma\epsilon}{(1-\gamma)} \alpha_{KL}^2$ is zero because $\alpha_{KL}^2 = D_{KL}^{max}(\pi_k, \pi_k) = 0$.

From this last fact and from $\eta_{\pi_{k+1}}^\alpha \geqslant B_k(\pi_{k+1})$ of the RCVaR Safe Improvement Bound we obtain:

$$
\eta_{\pi_{k+1}}^\alpha - \eta_{\pi_k}^\alpha \geqslant B_k(\pi_{k+1}) - B_k(\pi_k).
$$

Thus, by maximizing B_k at each iteration k, we guarantee that the RCVaR is non-decreasing.

■

[7] Comparing this bound to the results shown in the paper, the denominator term is not squared due to the normalization of the return.

G.16. Proof of Proposition 5.1

Let us do the gradient with respect to $\boldsymbol{\theta}$ of the objective function 11:

$$\begin{aligned}
&\nabla_{\boldsymbol{\theta}} \mathop{\mathbb{E}}_{\substack{s\sim d_{\mu,\pi_{\boldsymbol{\theta}_k}}(\cdot)\\ a\sim\pi_{\boldsymbol{\theta}_k}(\cdot|s)}} \left[\frac{\pi_{\boldsymbol{\theta}}(a|s)}{\pi_{\boldsymbol{\theta}_k}(a|s)} A^{\rho^\alpha}_{\boldsymbol{\theta}_k}(s,a)\right] = \\
=&\nabla_{\boldsymbol{\theta}} \int_{\mathcal{S}} d_{\mu,\pi_{\boldsymbol{\theta}_k}}(s) \int_{\mathcal{A}} \pi_{\boldsymbol{\theta}}(a|s) A^{\rho^\alpha}_{\pi_k}(s,a)\ \mathrm{d}a\ \mathrm{d}s \\
=&\int_{\mathcal{S}} d_{\mu,\pi_{\boldsymbol{\theta}_k}}(s) \int_{\mathcal{A}} \nabla_{\boldsymbol{\theta}} \pi_{\boldsymbol{\theta}}(a|s) A^{\rho^\alpha}_{\pi_k}(s,a)\ \mathrm{d}a\ \mathrm{d}s \\
=&\int_{\mathcal{S}} d_{\mu,\pi_{\boldsymbol{\theta}_k}}(s) \int_{\mathcal{A}} \pi_{\boldsymbol{\theta}}(a|s) \nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a|s) A^{\rho^\alpha}_{\pi_k}(s,a)\ \mathrm{d}a\ \mathrm{d}s \\
=&\int_{\mathcal{S}} d_{\mu,\pi_{\boldsymbol{\theta}_k}}(s) \int_{\mathcal{A}} \pi_{\boldsymbol{\theta}}(a|s) \nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a|s) Q^{\rho^\alpha}_{\pi_k}(s,a)\ \mathrm{d}a\ \mathrm{d}s.
\end{aligned}$$

If we evaluate it in $\boldsymbol{\theta} = \boldsymbol{\theta}_k$ we obtain:

$$\int_{\mathcal{S}} d_{\mu,\pi_{\boldsymbol{\theta}_k}}(s) \int_{\mathcal{A}} \pi_{\boldsymbol{\theta}_k}(a|s) (\nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a|s))|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k} Q^{\rho^\alpha}_{\pi_k}(s,a)\ \mathrm{d}a\ \mathrm{d}s,$$

which is, using Lemma A.1:

$$(1-\gamma) \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi_k(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[(\nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a_t|s_t))|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k} Q^{\rho^\alpha}_{\pi_k}(s_t,a_t)\right], \tag{15}$$

where $Q^{\rho^\alpha}_{\pi_k}$ is the action value function obtained from the modified MDP with reward $\widetilde{R}(s,a) = \rho^\alpha_{\pi_{\boldsymbol{\theta}_k}} - \frac{1}{\alpha}\left(R(s,a) - \rho^\alpha_{\pi_{\boldsymbol{\theta}_k}}\right)_-$. Excluding the first term $\rho^\alpha_{\pi_{\boldsymbol{\theta}_k}}$, that acts as a baseline, the transformed reward is $-\frac{1}{\alpha}\left(R(s,a) - \rho^\alpha_{\pi_{\boldsymbol{\theta}_k}}\right)_-$, so the gradient 15 is the PGT version of the gradient of the RCVaR. In fact the gradient of the RCVaR is (Tamar et al., 2015a):

$$-\frac{1}{\alpha}(1-\gamma) \mathop{\mathbb{E}}_{\substack{s_0\sim\mu(\cdot),\\ a_t\sim\pi_k(\cdot|s_t)\\ s_{t+1}\sim P(\cdot|s_t,a_t)}} \left[(\nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a_t|s_t))|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k} \sum_{t=0}^{\infty} \gamma^t \left(R(s_t,a_t) - \rho^\alpha_{\pi_{\boldsymbol{\theta}_k}}\right)_-\right].$$

■

H. Experiments details

For completeness, we show the learning curves that were not showed in the experiments part. We report also the hyperparameters of the algorithms for reproducibility. The algorithms RMAD-TRPO, RMAD-PPO, RCVaR-TRPO, RCVaR-PPO and TRVO were implemented by adapting the code of TRPO and PPO of Stable Baselines Hill et al. (2018). We used also the default multilayer perceptron (MLP) policy of Stable Baselines, which is a neural network with two layers with 64 hidden neurons each.

H.1. Grid-Worlds

H.1.1. Grid-World Bridge

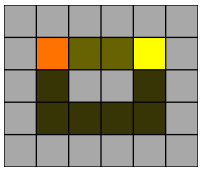
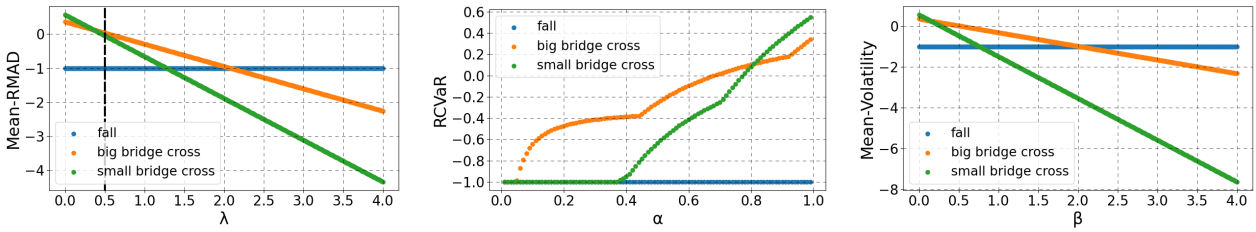


Figure 15: Graphical representation of the Grid-World Bridge environment. The orange square is the starting-state, while the yellow square is the goal-state in which the agent receives a big reward. There are two bridges: the small bridge allows to go to the goal-state in a few steps but there is a high risk of falling from the bridge and obtaining the lowest reward; the big bridge requires more steps to reach the goal-state but there is less risk of falling from the bridge. The grey squares represent a ravine, if the agent falls into the ravine it gets the lowest reward possible.

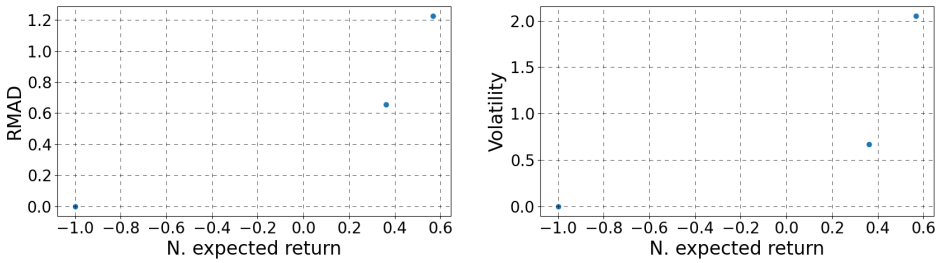
We created another Grid-World to show the utility of the coherence similarly as it was done with the Grid-World Garden. The Grid-World Bridge is represented in Figure 15. The state is the tuple (x, y) with the x coordinate and the y coordinate of the position of the agent in the environment, where $0 \leqslant x \leqslant 5$ and $0 \leqslant y \leqslant 4$. The agent's actions are go left, up, right or down. If the agent is on the bridge or wants to go there because it is on the starting-state then the probability of moving to the desired direction by on step is 0.7 otherwise a random direction is taken, while if we consider the big bridge the probability is 0.97. The grey squares and the goal-state are absorbing state, so once the agent is there it remains there until the end of the episode, which has length equal to 7. In the goal-state the reward is 2.5. When the agent is not in an absorbing state the reward it receives a reward equal to $0.5(D_i - D_{i-1}) + 0.2$, where 0.2 is an alive bonus, D_i is the Manhattan distance from the goal-state at step i which incentivizes the agent to reach the goal-state. In the grey squares the reward is -1, which is the lowest possible and it allows to apply the monotonicity property.



(a) Mean-RMAD (b) RCVaR (c) Mean-Volatility

Figure 16: Evaluation of three types of policies on the environment Bridge: the small bridge cross policy is the one that goes on the small bridge, the big bridge cross policy goes on the big bridge, while the falling policy is the one that falls into the ravine with one step. We considered the risk measures: Mean-RMAD, RCVaR and Mean-Volatility. The vertical dashed line in Figure 16a indicates that over that value the risk measure is no more coherent. For a given risk-aversion level the policy with the highest value is the best one. The policies were evaluated with 10000 steps.

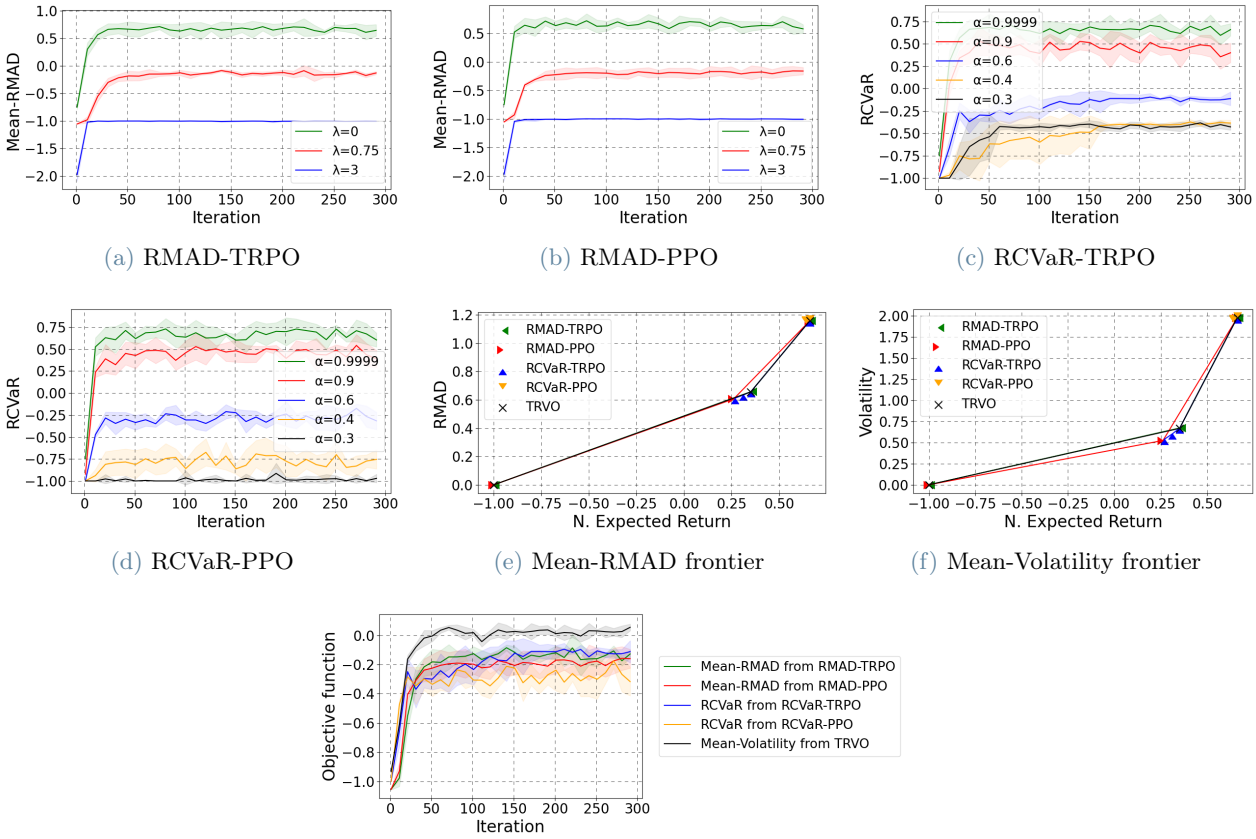
In Figure 16 we evaluated three policies: the one that goes to the goal-state through the small bridge, the one that goes to the goal-state through the big bridge and the one that falls into the ravine in one step. These three policies are reasonably optimal depending on the risk-aversion level and on the objective function. The falling policy is the most risk-averse and it is the one that must not be chosen if the risk-measure is coherent, because it gives always the lowest possible reward. In fact, we can see that for $0 \leqslant \lambda \leqslant 0.5$ the optimal policy for the Mean-RMAD is not the falling one, while for the RCVaR the falling policy is never optimal, because the RCVaR is always coherent. Regarding the Mean-Volatility, we do not have the guarantee that for a risk-aversion factor the falling policy is not optimal.



(a) Mean-RMAD policies (b) Mean-Volatility policies

Figure 17: The first figure shows the trade-off between the normalized expected return and the RMAD of the three policies: the one that goes to the goal-state through the small bridge, the one that goes to the goal-state through the big bridge and the one that falls into the ravine in one step. While the second figure shows the trade-off between the normalized expected return and the Volatility. The policies were evaluated with 10000 steps.

In Figure 17 there are the trade-off between the normalized expected return and the RMAD and the Volatility obtained from the previous three policies.



(a) RMAD-TRPO (b) RMAD-PPO (c) RCVaR-TRPO

(d) RCVaR-PPO (e) Mean-RMAD frontier (f) Mean-Volatility frontier

(g) Algorithms comparison

Figure 18: Further results obtained on the environment Bridge, with shaded area representing the standard deviation, while the solid lines represent the mean. The first four figures display the learning curves of the algorithms: RMAD-TRPO, RMAD-PPO, RCVaR-TRPO and RCVaR-PPO respectively, for different risk-aversion levels. Figure 18g reports as comparison the learning curves of the newly introduced algorithms and of the baseline TRVO, we considered the Mean-RMAD with $\lambda = 0.5$ for RMAD-TRPO and RMAD-PPO, the RCVaR with $\alpha = 0.6$ for RCVaR-TRPO and RCVaR-PPO and the Mean-Volatility with $\beta = 0.5$ for TRVO.

In Figure 18 we used our methods and TRVO on this environment. Each method was run for 300000 steps with batch size equal to 1000 steps. The validation part used for the frontiers and for the RCVaR values comprised 1000 episodes. We used a Gaussian policy on top of the MLP with variance trainable and initialized to 0 and bias initialized to 0. The discount factor is 0.86.

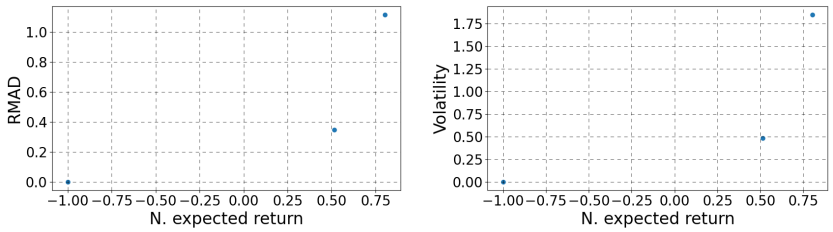
RMAD-TRPO, RCVaR-TRPO, TRVO. The hyperparameters are: maximum Kullback-Leibler divergence (KL): 0.005, conjugate-gradient iterations: 15, generalized advantage estimation factor: 0.95, entropy coefficient: 0.01, conjugate-gradient damping: 0.1, value function step size: 0.001, value function iterations: 10.

RMAD-PPO, RCVaR-PPO. The hyperparameters are: clipping parameter: 0.2, entropy coefficient: 0.01, number of epochs: 10, stepsize: 0.0003, minibatch size: 64, generalized advantage estimation factor: 0.95, learning rate scheduling: linear.

For curriculum learning of RCVaR-TRPO and RCVaR-PPO, we set the time constant to $10(1-\bar{\alpha})$, where $\bar{\alpha}$ is the α value of the RCVaR to optimize.

H.1.2. Grid-World Garden

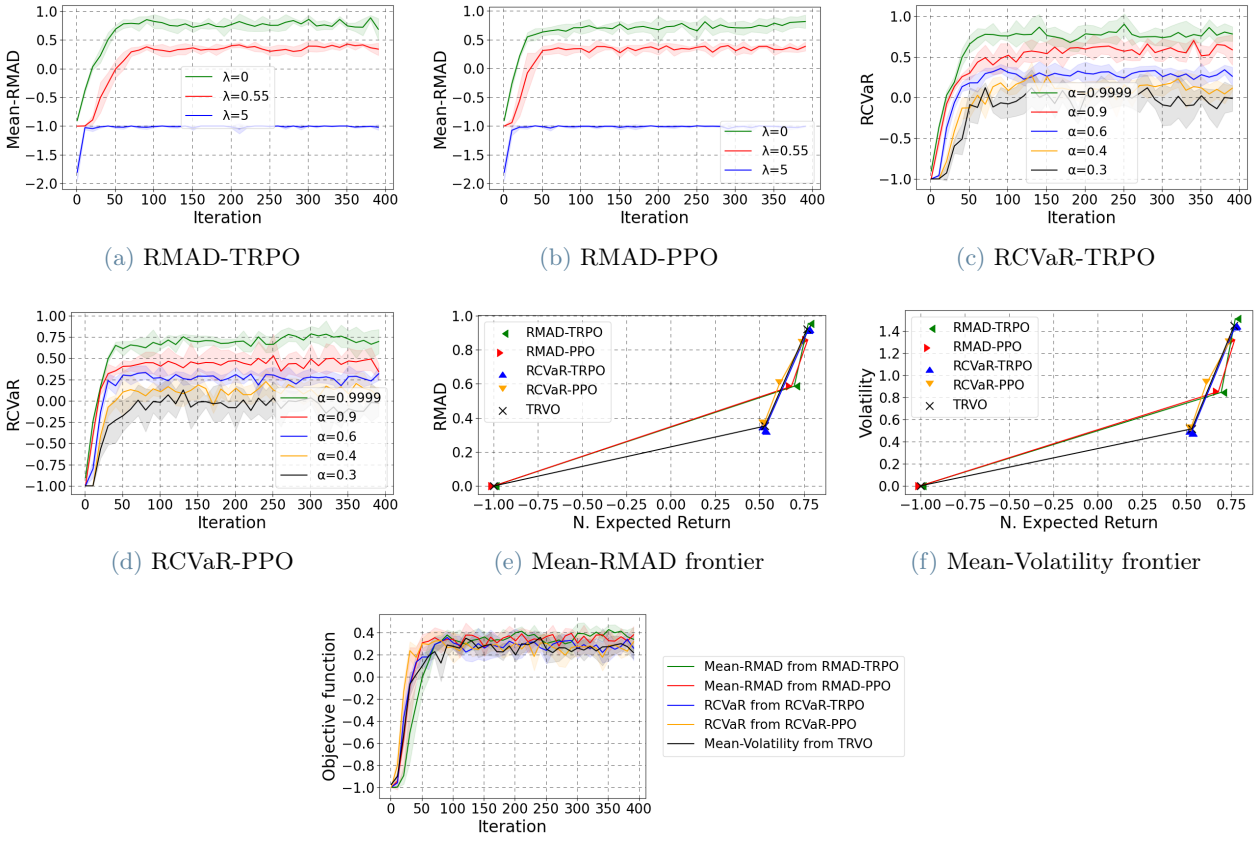
The state is the tuple (x, y) with the x coordinate and the y coordinate of the position of the agent in the environment, where $0 \leqslant x \leqslant 9$ and $0 \leqslant y \leqslant 9$. The agent can choose the direction of the step:left, up, right or down; and the speed of the step: one step or two steps. If the speed is high then the desired direction is taken with probability 0.86 otherwise a random direction is taken, while if the speed is low the probability is 0.9. The goal-state and grass part (including the people) are absorbing states and the agent remains there until the end of the episode which has length equal to 14. The goal-state gives a reward of 2.5. When the agent is not in an absorbing state the reward it receives a reward equal to $0.375(D_i - D_{i-1}) + 0.2$, where 0.2 is an alive bonus, D_i is the Manhattan distance from the goal-state at step i which incentivizes the agent to reach the goal-state. In the grass the reward is -1, which is the lowest possible and it allows to apply the monotonicity property.



(a) Mean-RMAD policies (b) Mean-Volatility policies

Figure 19: The first figure shows the trade-off between the normalized expected return and the RMAD of the three policies: go-fast, go-slow and quit. While the second figure shows the trade-off between the normalized expected return and the Volatility.

In Figure 19 there are the trade-off between the normalized expected return and the RMAD and the Volatility obtained from the policies: go-fast, go-slow and quit, that were described in Section 3.2. These policies were evaluated with 10000 steps.



(a) RMAD-TRPO (b) RMAD-PPO (c) RCVaR-TRPO

(d) RCVaR-PPO (e) Mean-RMAD frontier (f) Mean-Volatility frontier

(g) Algorithms comparison

Figure 20: Further results obtained on the environment Garden, with shaded area representing the standard deviation, while the solid lines represent the mean. The first four figures display the learning curves of the algorithms: RMAD-TRPO, RMAD-PPO, RCVaR-TRPO and RCVaR-PPO respectively, for different risk-aversion levels. Figure 20g reports as comparison the learning curves of the newly introduced algorithms and of the baseline TRVO, we considered the Mean-RMAD with $\lambda = 0.55$ for RMAD-TRPO and RMAD-PPO, the RCVaR with $\alpha = 0.6$ for RCVaR-TRPO and RCVaR-PPO and the Mean-Volatility with $\beta = 0.5$ for TRVO.

In Figure 20 we used our methods and TRVO on this environment. Each method was run for 400000 steps with batch size equal to 1000 steps. The validation part used for the frontiers and for the RCVaR values comprised 1000 episodes. We used a Gaussian policy on top of the MLP with variance trainable and initialized to 0 and bias initialized to 0. The discount factor is 0.92.

RMAD-TRPO, RCVaR-TRPO, TRVO. The hyperparameters are: maximum Kullback-Leibler divergence (KL): 0.005, conjugate-gradient iterations: 15, generalized advantage estimation factor: 0.95, entropy coefficient: 0.01, conjugate-gradient damping: 0.1, value function step size: 0.001, value function iterations: 10.

RMAD-PPO, RCVaR-PPO. The hyperparameters are: clipping parameter: 0.2, entropy coefficient: 0.01, number of epochs: 10, stepsize: 0.0003, minibatch size: 64, generalized advantage estimation factor: 0.95, learning rate scheduling: linear.

For curriculum learning of RCVaR-TRPO and RCVaR-PPO, we set the time constant to $40(1 - \bar{\alpha})$, where $\bar{\alpha}$ is the α value of the RCVaR to optimize.

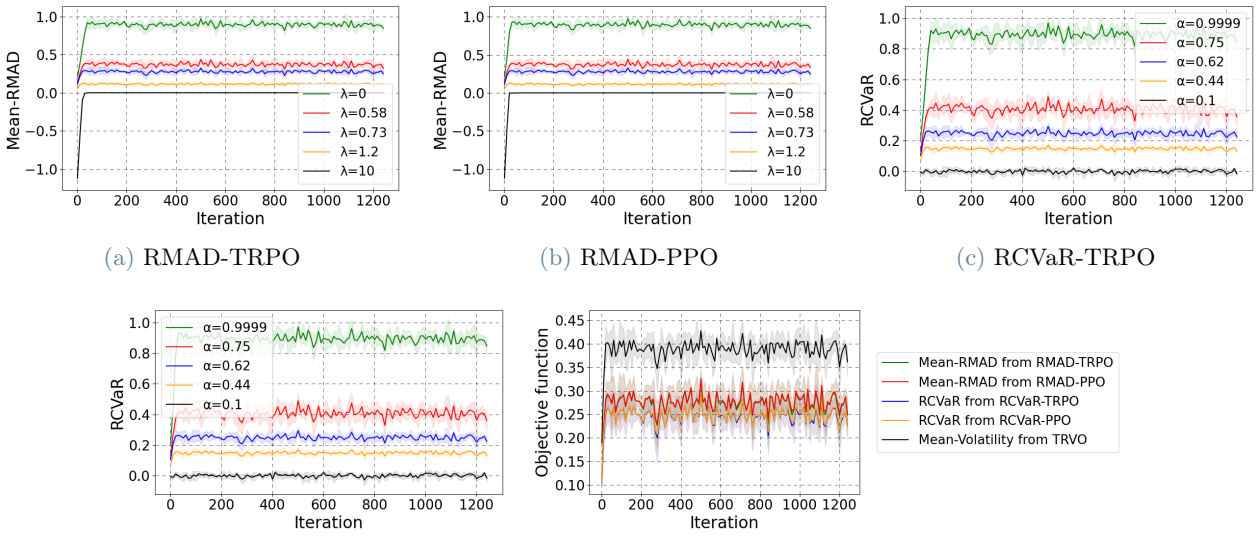
H.2. MAB

Each method was run for 500000 steps with batch size equal to 400 steps. The validation part used for the frontiers and for the RCVaR values comprised 200 episodes. We used a Gaussian policy on top of the MLP with variance non trainable and set to e^{-4} and bias initialized to 0.8. Further results are showed in Figure 21.

RMAD-TRPO, RCVaR-TRPO, TRVO. The hyperparameters are: maximum Kullback-Leibler divergence (KL): 0.01, conjugate-gradient iterations: 15, generalized advantage estimation factor: 0.95, entropy

coefficient: 0, conjugate-gradient damping: 0.1, value function step size: 0.001, value function iterations: 10.

RMAD-PPO, RCVaR-PPO. The hyperparameters are: clipping parameter: 0.2, entropy coefficient: 0, number of epochs: 10, stepsize: 0.0003, minibatch size: 64, generalized advantage estimation factor: 0.95, learning rate scheduling: linear.



(a) RMAD-TRPO (b) RMAD-PPO (c) RCVaR-TRPO

(d) RCVaR-PPO (e) Algorithms comparison

Figure 21: Further results obtained on the environment MAB, with shaded area representing the standard deviation, while the solid lines represent the mean. The first four figures display the learning curves of the algorithms: RMAD-TRPO, RMAD-PPO, RCVaR-TRPO and RCVaR-PPO respectively, for different risk-aversion levels. Figure 21e reports as comparison the learning curves of the newly introduced algorithms and of the baseline TRVO, we considered the Mean-RMAD with $\lambda = 0.73$ for RMAD-TRPO and RMAD-PPO, the RCVaR with $\alpha = 0.62$ for RCVaR-TRPO and RCVaR-PPO and the Mean-Volatility with $\beta = 0.6$ for TRVO.

H.3. Point Reacher

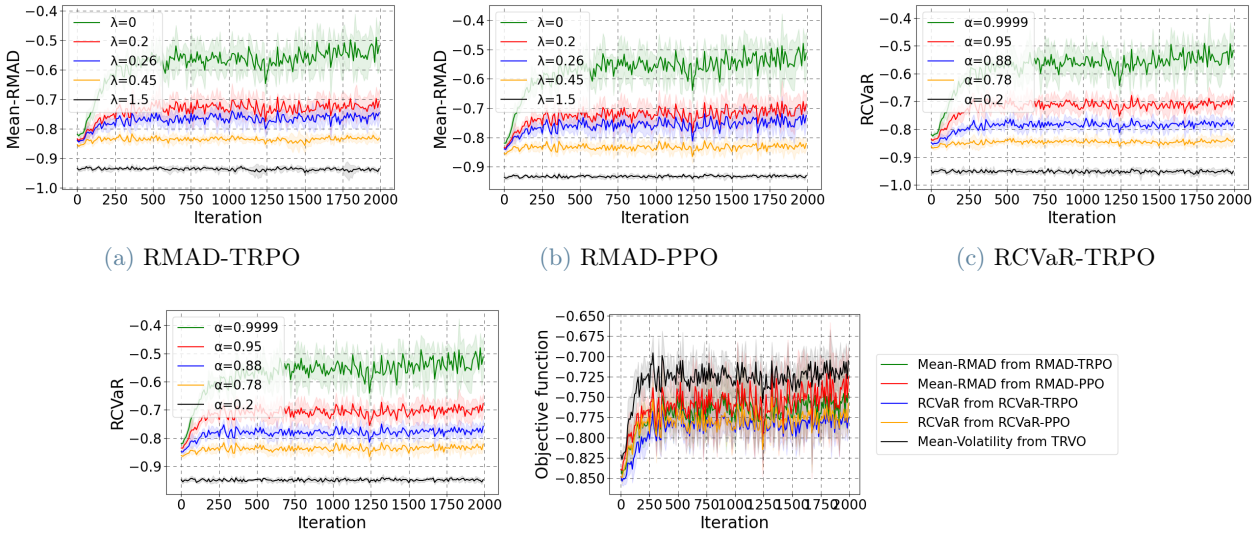
We considered only the policies that move directly to the goal with a fixed action (from 0 to 2 with step 0.01) or with an action equal to the distance from the goal if the distance is smaller than the fixed action, so we didn't considered the policies that go away from the target location. The policies were evaluated with 10000 episodes. We used different risk aversion factors from 0 to 2 with step 0.005 for the Mean-RMAD and for the Mean-Volatility, while for the RCVaR we used α from 0.01 to 0.99 with step 0.01.

H.4. Noisy Point Reacher

Each method was run for 2 million steps with batch size equal to 2000 steps. The validation part used for the frontiers and for the RCVaR values comprised 2000 episodes. We used a Gaussian policy on top of the MLP with variance non trainable and set to e^{-4} and bias initialized to 0. The discount factor is 0.9. More results are presented in Figure 22.

RMAD-TRPO, RCVaR-TRPO, TRVO. The hyperparameters are: maximum Kullback-Leibler divergence (KL): 0.005, conjugate-gradient iterations: 15, generalized advantage estimation factor: 0.95, entropy coefficient: 0, conjugate-gradient damping: 0.1, value function step size: 0.001, value function iterations: 10.

RMAD-PPO, RCVaR-PPO. The hyperparameters are: clipping parameter: 0.2, entropy coefficient: 0, number of epochs: 10, stepsize: 0.0003, minibatch size: 64, generalized advantage estimation factor: 0.95, learning rate scheduling: linear.



(a) RMAD-TRPO (b) RMAD-PPO (c) RCVaR-TRPO

(d) RCVaR-PPO (e) Algorithms comparison

Figure 22: Further results obtained on the environment Noisy Point Reacher, with shaded area representing the standard deviation, while the solid lines represent the mean. The four five figures display the learning curves of the algorithms: RMAD-TRPO, RMAD-PPO, RCVaR-TRPO and RCVaR-PPO respectively, for different risk-aversion levels. Figure 22e reports as comparison the learning curves of the newly introduced algorithms and of the baseline TRVO, we considered the Mean-RMAD with $\lambda = 0.26$ for RMAD-TRPO and RMAD-PPO, the RCVaR with $\alpha = 0.88$ for RCVaR-TRPO and RCVaR-PPO and the Mean-Volatility with $\beta = 0.2$ for TRVO.

H.5. Robotic Locomotion

H.5.1. Hopper

Each method was run for 12 million steps with batch size equal to 8000 steps. The validation part used for the frontiers and for the RCVaR values comprised 50 episodes. We used a Gaussian policy on top of the MLP with variance trainable and initialized to 1 and bias initialized to 0. The discount factor is 0.999.

RMAD-TRPO, RCVaR-TRPO, TRVO. The hyperparameters are: maximum Kullback-Leibler divergence (KL): 0.005, conjugate-gradient iterations: 15, generalized advantage estimation factor: 0.95, entropy coefficient: 0, conjugate-gradient damping: 0.1, value function step size: 0.001, value function iterations: 10.

RMAD-PPO, RCVaR-PPO. The hyperparameters are: clipping parameter: 0.2, entropy coefficient: 0, number of epochs: 10, stepsize: 0.0003, minibatch size: 64, generalized advantage estimation factor: 0.95, learning rate scheduling: linear.

For curriculum learning of RCVaR-TRPO and RCVaR-PPO, we set the time constant to $200(1 - \bar{\alpha})$, where $\bar{\alpha}$ is the α value of the RCVaR to optimize.

H.5.2. Walker

Each method was run for 48 million steps with batch size equal to 4000 steps. The validation part used for the frontiers and for the RCVaR values comprised 50 episodes. We used a Gaussian policy on top of the MLP with variance non trainable and set to e^{-2} and bias initialized to 0. The discount factor is 0.999.

RMAD-TRPO, RCVaR-TRPO, TRVO. The hyperparameters are: maximum Kullback-Leibler divergence (KL): 0.01, conjugate-gradient iterations: 15, generalized advantage estimation factor: 0.95, entropy coefficient: 0, conjugate-gradient damping: 0.1, value function step size: 0.001, value function iterations: 10.

RMAD-PPO, RCVaR-PPO. The hyperparameters are: clipping parameter: 0.2, entropy coefficient: 0, number of epochs: 10, stepsize: 0.0003, minibatch size: 64, generalized advantage estimation factor: 0.95, learning rate scheduling: linear.

For curriculum learning of RCVaR-TRPO and RCVaR-PPO, we set the time constant to $200(1-\bar{\alpha})$, where $\bar{\alpha}$ is the α value of the RCVaR to optimize.

H.6. Trading

Each method was run for 6 million steps with batch size equal to 9352 steps. The validation part used for the frontiers and for the RCVaR values comprised 50 episodes. We used a Boltzmann policy on top of the MLP with variance trainable and initialized to 1 and bias initialized to 0. The discount factor is 0.999.

RMAD-TRPO, RCVaR-TRPO, TRVO. The hyperparameters are: maximum Kullback-Leibler divergence (KL): 0.005, conjugate-gradient iterations: 15, generalized advantage estimation factor: 0.95, entropy coefficient: 0.01, conjugate-gradient damping: 0.1, value function step size: 0.001, value function iterations: 5.

RMAD-PPO, RCVaR-PPO. The hyperparameters are: clipping parameter: 0.2, entropy coefficient: 0, number of epochs: 10, stepsize: 0.0003, minibatch size: 64, generalized advantage estimation factor: 0.95, learning rate scheduling: linear.

Abstract in lingua italiana

In problemi del mondo reale come la robotica, la finanza e la sanità il rischio è sempre presente ed è importante tenerlo in considerazione in modo da limitare la possibilità di rari ma pericolosi eventi. La letteratura sull'apprendimento per rinforzo avverso al rischio ha indagato misure di rischio coerenti basate sul return del lungo termine e misure di rischio coerenti basate sul reward. Qui presentiamo due nuovi obiettivi avversi al rischio che sono sia coerenti sia basati sul reward: la media-deviazione media assoluta (Mean-RMAD) e il valore condizionale a rischio (RCVaR) basati sul reward, mostrando l'importanza della coerenza con un esempio. Dimostriamo che queste misure di rischio limitano il valore delle corrispondenti misure di rischio basate sul return, quindi se si incrementa una delle precedenti misure si incrementa anche la versione basata sul return. Sviluppiamo algoritmi per queste misure di rischio con la garanzia di miglioramento monotono della misura. Inoltre, un meta-algoritmo permette di risolvere la massimizzazione del RCVaR ottimizzando una sequenza di problemi neutrali al rischio. Infine, svolgiamo un'analisi empirica riguardo come questi approcci sono efficaci nel trovare comportamenti per diversi livelli di avversione al rischio su un ambiente finanziario e su ambienti rumorosi e impegnativi che provengono da PyBullet.

Parole chiave: apprendimento per rinforzo; avversione al rischio; misura di rischio coerente; misura basata sul reward; deviazione media assoluta; CVaR