Reinforcement Learning in Configurable Continuous Environments

Alberto Maria Metelli [1] **Emanuele Ghelfi** [1] **Marcello Restelli** [1]

Abstract

Configurable Markov Decision Processes (Conf-MDPs) have been recently introduced as an extension of the usual MDP model to account for the possibility of *configuring* the environment to improve the agent's performance. Currently, there is still no suitable algorithm to solve the learning problem for real-world Conf-MDPs. In this paper, we fill this gap by proposing a trust-region method, Relative Entropy Model Policy Search (REMPS), able to learn both the policy and the MDP configuration in continuous domains without requiring the knowledge of the true model of the environment. After introducing our approach and providing a finite-sample analysis, we empirically evaluate REMPS on both benchmark and realistic environments by comparing our results with those of the gradient methods.

1. Introduction

The overall goal of Reinforcement Learning (RL, Sutton & Barto, 1998) is to make an agent learn a behavior that maximizes the amount of reward it collects during its interaction with the environment. Most of the problems tackled by RL are typically modeled as a *Markov Decision Process* (MDP, Puterman, 2014) in which the environment is considered a fixed entity and cannot be adjusted. Nevertheless, there exist several real-world motivational examples in which partial control over the environment can be exercised by the agent itself or by an external supervisor (Metelli et al., 2018). For instance, in a car racing problem, the vehicle can be set up to better suit the driver's needs. The entity that performs the configuration can be either the driver itself (agent) or a track engineer (supervisor). With the phrase *environment configuration*, we refer to the activity of altering some environmental parameters to improve the performance of the agent's policy. This scenario has been recently formalized as a *Configurable Markov Decision Process* (Conf-MDP, Metelli et al., 2018). As in traditional RL, in a Conf-MDP the agent looks for the best policy but, in the meantime, there exists an entity entitled to configure the environment with the shared goal of maximizing the final performance of the policy. The nature of this new kind of interaction with the environment cannot be modeled either within the agent's action space or with a multi-agent framework. Indeed, the configuration activity cannot be placed at the same level as the agent's learning process. Configuring the environment may be more expensive and dangerous than updating the agent's policy and may occur on a different time scale w.r.t. the agent's learning process. Furthermore, while the entity that configures the environment must be aware of the presence of the agent (in order to wisely choose the environment configuration), the agent may not be aware of the fact that the configuration is taking place, perceiving the changes in the environment just as a non-stationarity. It is worth noting that the configuration process is rather different from the idea of changing the environment just to encourage learning in the original environment. While in the Conf-MDP framework the environment is altered because we *can* decide to change it (e.g., when a Formula 1 driver selects a car that better fits their driving abilities), in other works, like Ciosek & Whiteson (2017) and Florensa et al. (2017), the configuration is limited to a simulator and does not affect the real environment. Recently, an approach similar to Conf-MDPs, including also and explicit cost for altering the environment, has been proposed (Silva et al., 2018).

Learning in a Conf-MDP, therefore, means finding an agent's policy π together with an environment configuration p that, jointly, maximize the total reward. In Metelli et al. (2018), a safe-learning algorithm, *Safe Policy Model Iteration* (SPMI), is presented to solve the learning problem in the Conf-MDP framework. The basic idea is to optimize a lower bound of the performance improvement to ensure a monotonic increase of the total reward (Kakade & Langford, 2002; Pirotta et al., 2013). Although this approach succeeded in showing the benefits of configuring the environment in some illustrative examples, it is quite far from being applicable to real-world scenarios. We believe there are two significant limitations of SPMI. First of all, it is only applicable to problems with a finite state-action space, while the most interesting Conf-MDP examples have, at

[1] Politecnico di Milano, 32, Piazza Leonardo da Vinci, Milan, Italy. Correspondence to: Alberto Maria Metelli <albertomaria.metelli@polimi.it>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

least, a continuous state space (e.g., the car configuration problem). Second, it requires full knowledge of the environment dynamics. This latter limitation is the most relevant as, in reality, we almost never know the true environment dynamics, and even if a model is available it could be too approximate or too complex and computationally expensive (e.g., the fluid-dynamic model of a car).

In this paper, we propose a new learning algorithm for the Conf-MDP problem that overcomes the main limitations of SPMI. *Relative Entropy Model Policy Search* (REMPS) belongs to the *trust-region* class of methods (Schulman et al., 2015) and takes inspiration from REPS (Peters et al., 2010). REMPS operates with parametric policies $\pi_{\boldsymbol{\theta}}$ and configurations $p_{\boldsymbol{\omega}}$ and can be endowed with an approximate configuration model $\widehat{p}_{\boldsymbol{\omega}}$ that can be estimated from interaction with the environment. At each iteration, REMPS performs two phases: *optimization* and *projection*. In the optimization phase, we aim at identifying a new stationary distribution for the Conf-MDP that maximizes the total reward in a neighborhood of the current stationary distribution. This notion of neighborhood is encoded in our approach as a KL–divergence constraint. However, this distribution may fall outside the space of representable distributions, given the parametrization of the policy and of the configuration. Thus, the projection phase performs a moment projection in order to find an approximation of this stationary distribution in terms of representable policies and configurations.

Our framework shares some aspects with Utility Maximizing Design (Keren et al., 2017); although it assumes that the environment and the applicable modifications are known to the planner, whereas in our setting the agent only knows the environment parameters but ignores their effect on the transition probabilities. Controlling the learning process by employing the KL-divergence was previously done in the Linearly Solvable MDPs (Todorov, 2007) and its extensions (Todorov, 2009; Guan et al., 2014; Busic & Meyn, 2018) considering a penalty, rather than a constraint, to account for the cost of changing the transition probabilities.

In principle, the learning process in a parametric Conf-MDP can be carried out by a standard stochastic gradient method (Sutton et al., 2000; Peters & Schaal, 2008). We can easily adapt the classic REINFORCE (Williams, 1992) and G(PO)MDP (Baxter & Bartlett, 2001) estimators for learning the configuration parameters (see Appendix B). However, we believe that a first-order method does not scale to relevant situations that are of motivating interest in the Conf-MDP framework. For instance, it may be convenient to select a new configuration that makes the performance of the current policy worse because, in this new configuration, we have a much better chance of learning high-performing policies. We argue that this behavior is impossible by using a gradient method, as the gradient update direction attempts to improve performance for all parameters, including those in the transition model. This example justifies the choice of a trust-region method that allows a closed-form optimization in a controlled region. It has been proved empirically that these methods, also in the policy-search framework, are able to overcome local maxima (Levine & Koltun, 2013).

The contribution of this paper is threefold: algorithmic, theoretical and empirical. We start in Section 2 by recalling the definition of MDP, Conf-MDP and some notions of RL. Section 3 introduces our algorithm, REMPS, whose theoretical analysis is provided in Section 4. Section 5 shows how to equip REMPS with an approximation of the environment, while Section 6 presents the experimental evaluation. Finally, in Section 7 we discuss the results and provide some future research directions. The proofs of all results can be found in Appendix A.

2. Preliminaries

A discrete-time Markov Decision Process (MDP, Puterman, 2014) is defined by the tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, p, r, \mu, \gamma)$, where $\mathcal{S}$ and $\mathcal{A}$ are the state space and the action space respectively, p is the transition model that provides, for every state-action pair $(s, a) \in \mathcal{S} \times \mathcal{A}$, a probability distribution over the next state $p(\cdot|s, a)$, r is the reward model defining the reward collected by the agent $r(s, a, s')$ when performing action $a \in \mathcal{A}$ in state $s \in \mathcal{S}$ and landing on state $s' \in \mathcal{S}$, μ is the distribution of the initial state and $\gamma \in [0, 1]$ is the discount factor. The behavior of an agent is defined by means of a policy π that provides a probability distribution over the actions $\pi(\cdot|s)$ for every state $s \in \mathcal{S}$. A Configurable Markov Decision Process (Conf-MDP, Metelli et al., 2018) is defined as $\mathcal{CM} = (\mathcal{S}, \mathcal{A}, r, \mu, \gamma, \mathcal{P}, \Pi)$ and extends the MDP definition by considering a configuration space $\mathcal{P}$ instead of a single transition model p and adds a policy space Π to account for the possible limitations of the agent. The performance of a policy-configuration pair (π, p) is defined in terms of the expected return:

$$J_{\pi,p} = \mathop{\mathbb{E}}_{\substack{S_0 \sim \mu \\ A_t \sim \pi(\cdot|S_t) \\ S_{t+1} \sim p(\cdot|S_t, A_t)}} \left[\sum_{t=0}^{+\infty} \gamma^t r(S_t, A_t, S_{t+1}) \right]. \quad (1)$$

When $\gamma = 1$, the previous equation diverges; therefore, we resort to the expected average reward:

$$J_{\pi,p} = \liminf_{H \to +\infty} \mathop{\mathbb{E}}_{\substack{S_0 \sim \mu \\ A_t \sim \pi(\cdot|S_t) \\ S_{t+1} \sim p(\cdot|S_t, A_t)}} \left[\frac{1}{H} \sum_{t=0}^{H-1} r(S_t, A_t, S_{t+1}) \right]. \quad (2)$$

Sometimes we will refer to the state transition kernel $p^{\pi}(s'|s) = \int_{\mathcal{A}} \pi(a|s) p(s'|s, a) \mathrm{d}a$. For any policy $\pi \in \Pi$ and environment configuration $p \in \mathcal{P}$ we can define the (γ-*discounted*) *stationary distribution*:

$$d_{\pi,p}(s) = (1 - \gamma)\mu(s) + \gamma \int_{\mathcal{S}} d_{\pi,p}(s') p^{\pi}(s|s') \mathrm{d}s', \quad (3)$$

which represents the discounted number of times state $s \in \mathcal{S}$ is visited under policy π and configuration p. $d_{\pi,p}(s)$ surely exists for $\gamma < 1$. For the case $\gamma = 1$, we will assume that the Markov chain p^π is ergodic for any $\pi \in \Pi$ and $p \in \mathcal{P}$, so that a unique stationary distribution $d_{\pi,p}$ exists. We define the state-action stationary distribution as $d_{\pi,p}(s,a) = d_{\pi,p}(s)\pi(a|s)$ and the state-action-next-state stationary distribution as $d_{\pi,p}(s,a,s') = d_{\pi,p}(s)\pi(a|s)p(s'|s,a)$. In this way, we can unify Equations (1) and (2) as:

$$J_{\pi,p} = \mathop{\mathbb{E}}_{S,A,S' \sim d_{\pi,p}} \left[r(S,A,S') \right]. \tag{4}$$

Sometimes, given a stationary distribution d, we will indicate with J_d the corresponding performance. We will denote with $\mathcal{D}_{\Pi,\mathcal{P}}$ the set of all stationary distributions $d_{\pi,p}$ induced by Π and $\mathcal{P}$. In this paper, we assume that the agent policy belongs to a parametric policy space $\Pi_\Theta = \{\pi_{\boldsymbol{\theta}} : \boldsymbol{\theta} \in \Theta \subseteq \mathbb{R}^p\}$ as well as the environment configuration $\mathcal{P}_\Omega = \{p_{\boldsymbol{\omega}} : \boldsymbol{\omega} \in \Omega \subseteq \mathbb{R}^q\}$. Thus, the learning problem in a Conf-MDP can be rephrased as finding the optimal policy and configuration parametrizations:

$$\boldsymbol{\theta}^*, \boldsymbol{\omega}^* = \mathop{\arg\max}_{\boldsymbol{\theta} \in \Theta, \boldsymbol{\omega} \in \Omega} J_{\pi_{\boldsymbol{\theta}}, p_{\boldsymbol{\omega}}}. \tag{5}$$

3. Relative Entropy Model Policy Search

In this section, we introduce an algorithm to solve the learning problem in the Conf-MDP framework that can be effectively applied to continuous state-action spaces. *Relative Entropy Model Policy Search* (REMPS), imports several ideas from the classic REPS (Peters et al., 2010); in particular, the use of a constraint to ensure that the resulting new stationary distribution is sufficiently close to the current one. REMPS consists of two subsequent phases: *optimization* and *projection*. In the optimization phase (Section 3.1) we look for the stationary distribution d' that optimizes the performance (4). This search is limited to the space of distributions that are not too dissimilar from the current stationary distribution $d_{\pi,p}$. The notion of dissimilarity is formalized in terms of a threshold $\kappa > 0$ on the KL-divergence. The resulting distribution d' may not fall within the space of the representable stationary distributions given our parametrization $\mathcal{D}_{\Pi_\Theta,\mathcal{P}_\Omega}$. Therefore, similarly to Daniel et al. (2012), in the projection phase (Section 3.2) we need to retrieve a policy $\pi_{\boldsymbol{\theta}}$ and a configuration $p_{\boldsymbol{\omega}}$ inducing a stationary distribution $d_{\pi_{\boldsymbol{\theta}},p_{\boldsymbol{\omega}}} \in \mathcal{D}_{\Pi_\Theta,\mathcal{P}_\Omega}$ as close as possible to d'.

3.1. Optimization

The optimization problem can be stated in terms of stationary distributions only. Given a stationary distribution d (e.g., the one used to collect samples, i.e., $d_{\pi,p}$) and a KL–divergence threshold $\kappa > 0$, we look for a new stationary distribution d' solving the optimization problem PRIMAL$_\kappa$:

$$\max_{d' \in \Delta(\mathcal{S}\times\mathcal{A}\times\mathcal{S})} J_{d'} \qquad \text{s.t.} \quad D_{\mathrm{KL}}(d' \| d) \le \kappa,$$

where, for a given set $\mathcal{X}$, we have denoted with $\Delta(\mathcal{X})$ the set of all probability distributions over $\mathcal{X}$ and $D_{\mathrm{KL}}(d'\|d) = \mathbb{E}_{S,A,S'\sim d'}\left[\log \frac{d'(S,A,S')}{d(S,A,S')}\right]$ is the KL-divergence between d' and d. It is worth noting that, unlike REPS, we do not impose a constraint on the validity of the stationary distribution w.r.t. the transition model (constraint (7) in Peters et al. (2010)), as we have the possibility to change it. With similar mathematical tools we can solve PRIMAL$_\kappa$ in closed form.

Theorem 3.1. *Let d be a distribution over $\mathcal{S}\times\mathcal{A}\times\mathcal{S}$ and $\kappa > 0$ a KL-divergence threshold. The solution d' of the problem PRIMAL$_\kappa$, for $\kappa > 0$, is given by:*

$$d'(s,a,s') \propto d(s,a,s') \exp\left(\frac{1}{\eta} r(s,a,s')\right), \tag{6}$$

where η is the unique solution of the dual problem DUAL$_\kappa$:

$$\min_{\eta \in [0,+\infty)} g(\eta) = \eta \log \mathop{\mathbb{E}}_{S,A,S'\sim d}\left[\exp\left(\frac{1}{\eta} r(S,A,S') + \kappa\right)\right].$$

Thus, to find the optimal solution of PRIMAL$_\kappa$ we must first determine η, by solving DUAL$_\kappa$. It can be proved, as done in REPS, that with a change of variable $\overline{\eta} = 1/\eta$, we have that $g(\overline{\eta})$ is a convex function (Boyd & Vandenberghe, 2004), and therefore DUAL$_\kappa$ can be easily solved using standard optimization tools. Given a value of η, the new stationary distribution d' is defined by the exponential reweighting of each (s,a,s') triple with its reward $r(s,a,s')$. Moreover, given a stationary distribution d', we can derive a representation of a policy π' and a configuration p' inducing d'.

Corollary 3.1. *The solution d' of PRIMAL$_\kappa$ is induced by the configuration p' and the policy π' defined as:*

$$p'(s'|s,a) \propto p(s'|s,a) \exp\left(\frac{1}{\eta} r(s,a,s')\right),$$

$$\pi'(a|s) \propto \pi(a|s) \mathop{\mathbb{E}}_{S'\sim p(\cdot|s,a)}\left[\exp\left(\frac{1}{\eta} r(s,a,S')\right)\right].$$

In practice, we do not have access to the actual sampling distribution $d_{\pi,p}$, so we cannot compute the exact solution of the dual problem DUAL$_\kappa$. As in REPS, all expectations must be estimated from samples. Given a dataset $\{(s_i, a_i, s'_i, r_i)\}_{i=1}^N$ of N i.i.d. samples drawn from $d_{\pi,p}$, the empirical dual problem $\widetilde{\mathrm{DUAL}}_\kappa$ becomes:

$$\min_{\eta \in [0,+\infty)} \widetilde{g}(\eta) = \eta \log \frac{1}{N} \sum_{i=1}^N \exp\left(\frac{1}{\eta} r_i + \kappa\right),$$

which yields the solution $\widetilde{\eta}$ inducing the distribution $\widetilde{d}'$ defined by Equation (6). We discuss the effect of the finite sample size in Section 4.3.

3.2. Projection

The solution d' of the PRIMAL$_\kappa$ problem does not belong, in general, to the class of stationary distributions $\mathcal{D}_{\Pi_\Theta,\mathcal{P}_\Omega}$

induced by Π_Θ and $\mathcal{P}_\Omega$. For this reason, we look for a parametric policy $\pi_{\boldsymbol{\theta}}$ and a parametric configuration $p_{\boldsymbol{\omega}}$ that induce a stationary distribution $d_{\pi_{\boldsymbol{\theta}},p_{\boldsymbol{\omega}}}$ as close as possible to d', by performing a moment projection (PROJ_d):[1]

$$\begin{aligned}\boldsymbol{\theta}', \boldsymbol{\omega}' &= \operatorname*{arg\,min}_{\boldsymbol{\theta}\in\Theta, \boldsymbol{\omega}\in\Omega} D_{\mathrm{KL}}\left(d' \| d_{\pi_{\boldsymbol{\theta}},p_{\boldsymbol{\omega}}}\right) \\ &= \operatorname*{arg\,max}_{\boldsymbol{\theta}\in\Theta, \boldsymbol{\omega}\in\Omega} \mathop{\mathbb{E}}_{S,A,S'\sim d'}\left[\log d_{\pi_{\boldsymbol{\theta}},p_{\boldsymbol{\omega}}}(S,A,S')\right].\end{aligned}$$

However, this problem is very hard to solve as computing the functional form of $d_{\pi_{\boldsymbol{\theta}},p_{\boldsymbol{\omega}}}$ is complex and cannot be performed in closed form for most cases. If the state space and the action space are finite, we can formulate the problem by defining a set of constraints $d'(s) = (1-\gamma)\mu(s) + \gamma \sum_{s'\in\mathcal{S}} \sum_{a\in\mathcal{A}} d'(s')\pi_{\boldsymbol{\theta}}(a|s')p_{\boldsymbol{\omega}}(s|s',a)$, $\forall s \in \mathcal{S}$ to enforce the nature of the stationary distribution. Nevertheless, in most of the relevant cases, the problem remains intractable as the state space could be very large. Therefore, we consider more convenient projection approaches that we will justify from a theoretical standpoint in Section 4.1. A first relaxation consists in finding an approximation of the transition kernel ${p'}^{\pi'}$ induced by d' (PROJ_{p^π}):

$$\begin{aligned}\boldsymbol{\theta}', \boldsymbol{\omega}' &= \operatorname*{arg\,min}_{\boldsymbol{\theta}\in\Theta, \boldsymbol{\omega}\in\Omega} \mathop{\mathbb{E}}_{S\sim d'}\left[D_{\mathrm{KL}}\left({p'}^{\pi'}(\cdot|S) \| p_{\boldsymbol{\omega}}^{\pi_{\boldsymbol{\theta}}}(\cdot|S)\right)\right] \\ &= \operatorname*{arg\,max}_{\boldsymbol{\theta}\in\Theta, \boldsymbol{\omega}\in\Omega} \mathop{\mathbb{E}}_{S,A,S'\sim d'}\left[\log p_{\boldsymbol{\omega}}^{\pi_{\boldsymbol{\theta}}}(S'|S)\right].\end{aligned}$$

Clearly, we need to be able to compute the functional form of the state transition kernel $p_{\boldsymbol{\omega}}^{\pi_{\boldsymbol{\theta}}}$, which is only possible when considering finite action spaces. Indeed, in such case, we just have to marginalize over the (finite) action space as: $p_{\boldsymbol{\omega}}^{\pi_{\boldsymbol{\theta}}}(s'|s) = \sum_{a\in\mathcal{A}} \pi_{\boldsymbol{\theta}}(a|s)p_{\boldsymbol{\omega}}(s'|s,a)$. When also the action space is infinite, we resort to separate projections for the policy and the transition model ($\mathrm{PROJ}_{\pi,p}$):

$$\begin{aligned}\boldsymbol{\theta}' &= \operatorname*{arg\,min}_{\boldsymbol{\theta}\in\Theta} \mathop{\mathbb{E}}_{S\sim d'}\left[D_{\mathrm{KL}}\left(\pi'(\cdot|S) \| \pi_{\boldsymbol{\theta}}(\cdot|S)\right)\right] \\ &= \operatorname*{arg\,max}_{\boldsymbol{\theta}\in\Theta} \mathop{\mathbb{E}}_{S,A,S'\sim d'}\left[\log \pi_{\boldsymbol{\theta}}(A|S)\right], \\ \boldsymbol{\omega}' &= \operatorname*{arg\,min}_{\boldsymbol{\omega}\in\Omega} \mathop{\mathbb{E}}_{S,A\sim d'}\left[D_{\mathrm{KL}}\left(p'(\cdot|S,A) \| p_{\boldsymbol{\omega}}(\cdot|S,A)\right)\right] \\ &= \operatorname*{arg\,max}_{\boldsymbol{\omega}\in\Omega} \mathop{\mathbb{E}}_{S,A,S'\sim d'}\left[\log p_{\boldsymbol{\omega}}(S'|S,A)\right].\end{aligned}$$

Similarly to what happens during the optimization phase, we only have access to a finite dataset of N samples. Moreover, here we are faced with an additional challenge, i.e., we need to compute expectations w.r.t. d', but our samples are collected with $d_{\pi,p}$. This can be cast as an off-distribution estimation problem and therefore we resort to importance weighting (Owen, 2013). In the importance weighting estimation, each sample (s_i, a_i, s'_i) is reweighted by the likelihood of being generated by d', i.e., by $w_i = \exp(r_i/\widetilde{\eta})$.[2] In the following, we will denote the approximate projections

[1] When using samples, the moment projection is equivalent to the maximum likelihood estimation.

[2] The likelihood is given by the density ratio $\frac{d'(s,a,s')}{d_{\pi,p}(s,a,s')} \propto \exp(r(s,a,s')/\widetilde{\eta})$.

Algorithm 1 Relative Entropy Model Policy Search

Initialize $\boldsymbol{\theta}_0, \boldsymbol{\omega}_0$ arbitrarily
for $t = 0, 1, ...$ until convergence[3] **do**
  Collect N samples $\{(s_i, a_i, s'_i, r_i)\}_{i=1}^N$ with $d_{\pi_{\boldsymbol{\theta}_t},p_{\boldsymbol{\omega}_t}}$
  (*Optimization*) Compute $\widetilde{\eta}$ and $\widetilde{d'}$ solving the $\widetilde{\mathrm{DUAL}}_\kappa$
  (*Projection*) Perform the projection of $\widetilde{d'}$ and obtain $\boldsymbol{\theta}_{t+1}$ and $\boldsymbol{\omega}_{t+1}$
end for

with $\widetilde{\mathrm{PROJ}}$. A summary of the objective functions for the different projection approaches, their applicability, and the corresponding estimators are reported in Table 1.

The full REMPS problem can be stated as the composition of optimization and projection, i.e., $\mathrm{REMPS}_\kappa = \mathrm{PROJ} \circ \mathrm{PRIMAL}_\kappa$, and the corresponding problem from samples as $\widetilde{\mathrm{REMPS}}_\kappa = \widetilde{\mathrm{PROJ}} \circ \widetilde{\mathrm{PRIMAL}}_\kappa$. Refer to Algorithm 1 for a high-level pseudocode of REMPS.

4. Theoretical Analysis

In this section, we elaborate on three theoretical aspects of REMPS. First of all, we provide three inequalities that bound the difference of performance when changing the policy and the model in terms of distributional divergences between stationary distributions, policies and models (Section 4.1). Secondly, we present a sensitivity study of the hyper-parameter κ (i.e., the KL-divergence threshold) of REMPS (Section 4.2). Finally, we discuss a finite-sample analysis of the single step of REMPS (Section 4.3). In the following, we will not constrain the policy and the configuration spaces to be parametric spaces, so we will omit the parameter space dependence in the symbols Π and $\mathcal{P}$. Furthermore, we will consider the following assumptions.

Assumption 4.1. *(Uniformly bounded reward) For any $s, s' \in \mathcal{S}, a \in \mathcal{A}$ it holds that: $|r(s,a,s')| \leq r_{\max} < +\infty$.*

Assumption 4.2. *(Ergodicity) Let $\pi \in \Pi$ and $p \in \mathcal{P}$, the* ergodicity coefficient *(Seneta, 1988) of the Markov chain induced by π and p is defined as:*

$$\tau(p^\pi) = \frac{1}{2} \sup_{s,s'\in\mathcal{S}} \left\| p^\pi(\cdot|s) - p^\pi(\cdot|s') \right\|.$$

If $\gamma = 1$, for any $\pi \in \Pi$ and $p \in \mathcal{P}$ we assume $\tau(p^\pi) \leq \tau_{\max}$.

4.1. Performance Bounds

We start with the following result that bounds the absolute difference of total reward with a dissimilarity index between the stationary distributions.

[3] The algorithm can be stopped after a fixed number of iterations $N_{\max}$ or if the performance improvement between two consecutive iterations is too small.

Table 1. Applicability, exact objective function and corresponding estimator for the three projections presented. w_i is the (non-normalized) importance weight defined as $w_i = \exp(r_i/\widetilde{\eta})$.

Projection	$\|\mathcal{S}\| = \infty$	$\|\mathcal{A}\| = \infty$	Exact objective	Estimated objective
PROJ_d	✗	✗	$\mathbb{E}_{S,A,S' \sim d'}\left[\log d_{\pi_{\boldsymbol{\theta}}, p_{\boldsymbol{\omega}}}(S,A,S')\right]$	$\frac{1}{N}\sum_{i=1}^N w_i \log d_{\pi_{\boldsymbol{\theta}}, p_{\boldsymbol{\omega}}}(s_i,a_i,s_i')$
PROJ_{p^π}	✓	✗	$\mathbb{E}_{S,A,S' \sim d'}\left[\log p_{\boldsymbol{\omega}}^{\pi_{\boldsymbol{\theta}}}(S'\|S)\right]$	$\frac{1}{N}\sum_{i=1}^N w_i \log p_{\boldsymbol{\omega}}^{\pi_{\boldsymbol{\theta}}}(s_i'\|s_i)$
$\mathrm{PROJ}_{\pi,p}$	✓	✓	$\mathbb{E}_{S,A,S' \sim d'}\left[\log \pi_{\boldsymbol{\theta}}(A\|S)\right]$ $\mathbb{E}_{S,A,S' \sim d'}\left[\log p_{\boldsymbol{\omega}}(S'\|S,A)\right]$	$\frac{1}{N}\sum_{i=1}^N w_i \log \pi_{\boldsymbol{\theta}}(a_i\|s_i)$ $\frac{1}{N}\sum_{i=1}^N w_i \log p_{\boldsymbol{\omega}}(s_i'\|s_i,a_i)$

Proposition 4.1. *Let d and d' be two stationary distributions, then it holds that:*

$$|J_{d'} - J_d| \le r_{\max} \|d' - d\|_1 \le r_{\max}\sqrt{2D_{\mathrm{KL}}(d' \| d)}.$$

This result justifies the use of the projection PROJ_d, since minimizing the KL-divergence between the stationary distributions allows controlling the performance difference. As we have seen in Section 3.2, the PROJ_d is typically intractable. Therefore, we now prove that performing the projection of the state transition kernel (PROJ_{p^π}) still allows controlling the performance difference.

Corollary 4.1. *Let p^π and $p'^{\pi'}$ two transition kernels, inducing the stationary distributions d and d' respectively, then, under Assumption 4.2, it holds that:*

$$|J_{d'} - J_d| \le r_{\max}\rho\sqrt{2\mathop{\mathbb{E}}_{S\sim d'}\left[D_{\mathrm{KL}}\left(p'^{\pi'}(\cdot|S)\|p^\pi(\cdot|S)\right)\right]},$$

where $\rho = \frac{\gamma}{1-\gamma}$ if $\gamma < 1$ or $\rho = \frac{1}{1-\tau_{\max}}$ if $\gamma = 1$.

Finally, the following result provides a justification for the separate projections of policy and model ($\mathrm{PROJ}_{\pi,p}$) .

Lemma 4.1. *Let (π, p) and (π', p') be two policy-configuration pairs and let p^π and $p'^{\pi'}$ the corresponding transition kernels, then for any state $s \in \mathcal{S}$, it holds that:*

$$D_{\mathrm{KL}}\left(p'^{\pi'}(\cdot|s)\|p^\pi(\cdot|s)\right) \le D_{\mathrm{KL}}\left(\pi'(\cdot|s)\|\pi(\cdot|s)\right) + \mathop{\mathbb{E}}_{A\sim\pi'(\cdot|s)}\left[D_{\mathrm{KL}}\left(p'(\cdot|s,A)\|p(\cdot|s,A)\right)\right].$$

As an immediate consequence, thanks to the monotonicity property, the inequality remains valid when taking the expectation w.r.t. $S \sim d'$. Thus, we are able to bound the right-hand side of Corollary 4.1 isolating the contribution of policy and configuration.

4.2. Sensitivity to the KL threshold

We analyze how the performance of the solution of PRIMAL_κ changes when the KL-divergence threshold κ varies. The following result upper bounds the reduction in performance between the optimal solution d of PRIMAL_κ and the optimal solution d' of $\mathrm{PRIMAL}_{\kappa'}$ when $\kappa' \le \kappa$.

Proposition 4.2. *Let d and d' be the solutions of PRIMAL_κ and $\mathrm{PRIMAL}_{\kappa'}$ respectively with $\kappa' \le \kappa$, having d_0 as sampling distribution. Then, it holds that:*

$$J_d - J_{d'} \le r_{\max}\|d - d_0\|_1\left(1 - \frac{\kappa'}{\kappa}\right). \tag{7}$$

This result is general and can be applied broadly to the class of trust-region methods, when using the KL-divergence as a constraint to define the trust-region.

4.3. Finite-sample Analysis

Now we present a finite-sample analysis of the single step of REMPS. In particular, our goal is to upper bound the difference $J_{d'} - J_{\widetilde{\pi}'', \widetilde{p}''}$ where d' is the solution of the exact problem PRIMAL_κ and $d_{\widetilde{\pi}'', \widetilde{p}''}$ is the solution obtained after projecting $\widetilde{d}'$ onto $\mathcal{D}_{\Pi,\mathcal{P}}$. Due to the similarities between the two algorithms, large part of our analysis applies to also REPS (Peters et al., 2010). We will denote $\mathcal{D}_d = \{d' \in \Delta(\mathcal{S}\times\mathcal{A}\times\mathcal{S}) : d' \propto d\exp(r/\eta) : \eta \in [0, +\infty)\}$ the set of possible solutions to the PRIMAL_κ problem.We will consider the following additional assumptions.

Assumption 4.3. *(Finite pseudo-dimension) Given a policy $\pi \in \Pi$ and a transition model $p \in \mathcal{P}$, the pseudo-dimensions of the hypothesis spaces $\{\frac{d}{d_{\pi,p}} : d \in \mathcal{D}_{d_{\pi,p}}\}$, $\{\frac{d}{d_{\pi,p}} r : d \in \mathcal{D}_{d_{\pi,p}}\}$, $\{\frac{d}{d_{\pi,p}}\log\frac{d}{d_{\pi,p}} : d \in \mathcal{D}_{d_{\pi,p}}\}$ and $\{\frac{d}{d_{\pi,p}}\log d' : d \in \mathcal{D}_{d_{\pi,p}}, d' \in \mathcal{D}_{\Pi,\mathcal{P}}\}$ are bounded by $v < +\infty$.*

Assumption 4.4. *(Finite β–moments) There exist $\beta \in (1,2)$, such that*

$$\mathop{\mathbb{E}}_{S,A,S'\sim d_{\pi,p}}\left[\left|\frac{d(S,A,S')}{d_{\pi,p}(S,A,S')}\right|^\beta\right]^{1/\beta} \quad \textit{and}$$

$$\mathop{\mathbb{E}}_{S,A,S'\sim d_{\pi,p}}\left[\left|\frac{d(S,A,S')}{d_{\pi,p}(S,A,S')}\log d'(S,A,S')\right|^\beta\right]^{1/\beta}$$

are bounded for all $d \in \mathcal{D}_{d_{\pi,p}}$ and $d' \in \mathcal{D}_{\Pi,\mathcal{P}}$.

Assumption 4.3 requires that all the involved hypothesis spaces (for the solution of the PRIMAL_κ and PROJ) are characterized by a finite pseudo-dimension. This assumption is necessary to state learning theory guarantees. Assumption 4.4 is more critical as it requires that the involved loss functions (used to solve the PRIMAL_κ and PROJ) have a uniformly bounded (over the hypothesis space) moment of order $\beta \in (1, 2)$. In particular, the first line states that the exponentiated β-Rényi divergence (Cortes et al., 2010, see Equation (34)) between d and $d_{\pi,p}$ is finite for some $\beta \in (1, 2)$. This requirement allows an analysis based on Cortes et al. (2013) for unbounded loss function with bounded moments. A more straightforward analysis can be made by assuming that the involved loss functions are uniformely bounded and using more traditional tools (Mohri et al., 2012) (see Appendix A.4.4). However, we believe this latter requirement is too restrictive. Therefore, we report below the general statement, under Assumption 4.4.

Theorem 4.1. *(Finite–Sample Bound) Let $\pi \in \Pi$ and $p \in \mathcal{P}$ be the current policy and transition model. Let $\kappa > 0$ be the KL–divergence threshold. Let $d' \in \mathcal{D}_{d_{\pi,p}}$ be the solution of the PRIMAL$_\kappa$ problem and $d_{\widetilde{\pi}'',\widetilde{p}''} \in \mathcal{D}_{\Pi,\mathcal{P}}$ be the solution of the $\widetilde{\text{REMPS}}_\kappa$ problem with PROJ$_d$ computed with $N > 0$ samples collected with $d_{\pi,p}$. Then, under Assumptions 4.1, 4.3 and 4.4, for any $\alpha \in (1, \beta)$, there exist two constants χ, ξ and a function $\zeta(N) = \mathcal{O}(\log N)$ depending on α, and on the samples, such that for any $\delta \in (0, 1)$, with probability at least $1 - 4\delta$ it holds that:*

$$J_{d'} - J_{\widetilde{\pi}'',\widetilde{p}''} \leq \sqrt{2} r_{\max} \underbrace{\sup_{d \in \mathcal{D}_{d_{\pi,p}}} \inf_{\overline{d} \in \mathcal{D}_{\Pi,\mathcal{P}}} \sqrt{D_{\mathrm{KL}}(d \| \overline{d})}}_{\text{approximation error}} + \underbrace{r_{\max}\chi\sqrt{\epsilon} + r_{\max}\zeta(N)\epsilon + r_{\max}\xi\epsilon^2}_{\text{estimation error}},$$

where $\epsilon = 2^{\frac{\alpha+2}{2\alpha}} \sqrt{\frac{v \log \frac{2eN}{v} + \log \frac{8}{\delta}}{N^{\frac{2(\alpha-1)}{\alpha}}}} \Gamma\left(\alpha, \sqrt{\frac{v \log \frac{2eN}{v} + \log \frac{8}{\delta}}{N^{\frac{2(\alpha-1)}{\alpha}}}}\right)$, *depending on the pseudo-dimension bound $v < +\infty$ and* $\Gamma(\alpha, \tau) = \frac{\alpha-1}{\alpha} + \frac{1}{\alpha}\left(\frac{\alpha}{\alpha-1}\right)^{\alpha-1}\left(1 + \left(\frac{\alpha-1}{\alpha}\right)^{\alpha-1} \log \frac{1}{\tau}\right)^{\frac{\alpha-1}{\alpha}}$.

Proof Sketch. The idea of the proof is to decouple the contributions of (i) $\widetilde{\text{PRIMAL}}_\kappa$ and (ii) $\widetilde{\text{PROJ}}_d$ to the final error:

$$J_{d'} - J_{\widetilde{\pi}'',\widetilde{p}''} = \underbrace{J_{d'} - J_{\widetilde{d}'}}_{\text{(i)}} + \underbrace{J_{\widetilde{d}'} - J_{\widetilde{\pi}'',\widetilde{p}''}}_{\text{(ii)}},$$

where $\widetilde{d}'$ is the solution of $\widetilde{\text{PRIMAL}}_\kappa$. (i) is the contribution of the *estimation* error due to the finite number of samples used to solve $\widetilde{\text{PRIMAL}}_\kappa$. It is analyzed in Lemma A.3, exploiting the sensitivity analysis of Lemma 4.2. (ii) includes the contribution of an *approximation* error due to the space in which we represent the stationary distributions $\mathcal{D}_{\Pi,\mathcal{P}}$ and an estimation error due to the finite number of samples used to perform the projection $\widetilde{\text{PROJ}}_d$. This term can be decomposed using Lemma A.4. Lemmas A.7-A.10 are then employed to state learning theory bounds over the various terms that allow completing the proof of the main statement. □

The estimation error is dominated by $\sqrt{\epsilon}$. Ignoring logarithmic terms, we have that $J_{d'} - J_{\widetilde{\pi}'',\widetilde{p}''} = \widetilde{\mathcal{O}}\big(N^{-\frac{2(\alpha-1)}{4\alpha}}\big)$. In this analysis, we considered the case in which the projection

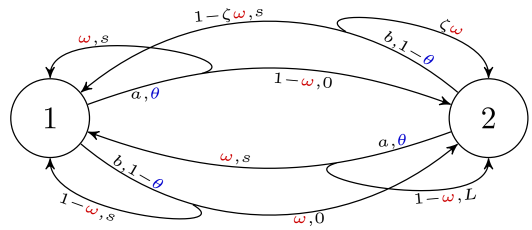

Figure 1. The Chain Domain.

is performed over the stationary distribution (PROJ$_d$).[4] The result can be easily extended to the case in which we resort to PROJ$_{p^\pi}$ or PROJ$_{\pi,p}$ (Corollary A.1).

5. Approximation of the Transition Model

The formulation of REMPS we presented above, requires access to a representation of the environment model $p_{\boldsymbol{\omega}}$, depending on a vector of parameters $\boldsymbol{\omega}$. Although the parameters that can be configured are usually known; the environment dynamics is unknown in a model-free scenario. Even when an environment model is available it may be too imprecise or too complex to be used effectively. In principle, we could resort to a general model-based RL approach to effectively approximate the transition model (Deisenroth & Rasmussen, 2011; Nagabandi et al., 2018). However, in our scenario, we need to learn a mapping from state-action-configuration triples to a new state. Our approach is based on a simple maximum likelihood estimation. Given a dataset of experience $\{(s_i, a_i, s'_i, \boldsymbol{\omega}_i)\}_{i=1}^N$ (possibly collected with different policies π_i and different configurations $\boldsymbol{\omega}_i$) and given an approximation space $\widehat{\mathcal{P}}_\Omega$ we solve the problem:

$$\max_{\widehat{p} \in \widehat{\mathcal{P}}_\Omega} \frac{1}{N} \sum_{i=1}^{N} \log \widehat{p}(s'_i | s_i, a_i, \boldsymbol{\omega}_i), \tag{8}$$

where we made explicit that the distribution of the next state s'_i depends also on the configuration.[5] Given the model approximation, we can run REMPS by replacing p with $\widehat{p}$. We do not impose any restriction on the specific model class $\widehat{\mathcal{P}}_\Omega$ (e.g., neural network, Gaussian process) and on the moment in which the fitting phase has to be performed (e.g., at the beginning of the training or every m iterations).

6. Experiments

In this section, we provide the experimental evaluation of REMPS on three domains: a simple chain domain (Section 6.1, Figure 1), the classical Cartpole (Section 6.2) and

[4] Note that Assumption 4.4 ensures that the approximation error is finite, since the KL-divergence is the 1-Rényi divergence and the Rényi divergence is non-decreasing in the order β (Van Erven & Harremos, 2014).

[5] Notice that the configuration parameters $\boldsymbol{\omega}$ are an input of the approximate model.

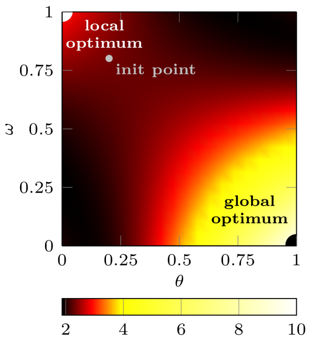

Figure 2. Return surface of the Chain domain.

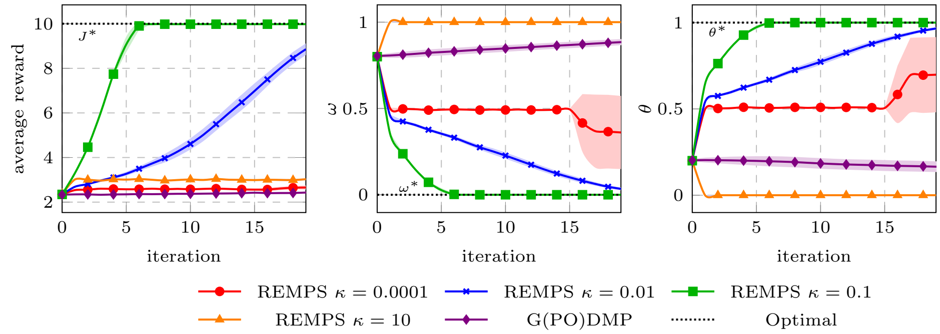

Figure 3. Average reward, configuration parameter ω, and policy parameter θ, as a function of the number of iterations for REMPS with different values of κ and G(PO)MDP. 20 runs, 95% c.i.

a more challenging car-configuration task based on TORCS (Section 6.3). In the first two experiments, we compare REMPS with the extension of G(PO)MDP to the policy-configuration learning (Appendix B), whereas in the last experiment we evaluate REMPS against REPS, the latter used for policy learning only. Full experimental details are reported in Appendix D.

6.1. Chain Domain

The Chain Domain (Figure 1) is an illustrative example of Conf-MDP. There are two states 1 and 2 and the agent can perform two actions a (forward) and b (backward). The agent is forced to play every action with the same probability in both states, i.e., $\pi_\theta(a|s) = \theta$ and $\pi_\theta(b|s) = 1 - \theta$ for all $s \in \{1, 2\}$ and $\theta \in [0, 1]$. The environment can be configured via the parameter $\omega \in [0, 1]$, that is the probability of action failure. Action a, if successful, takes the agent to state 2, whereas action b, if successful, takes the agent to state 1. When one action fails, the other is executed. The agent gets a high reward, $L > 0$, if, starting from state 1, it successfully executes action a, while it gets a smaller reward, l ($0 < l < L$) if it lands in state 2 starting from 1 but by performing action b. The agent gets an even smaller reward, s ($0 < s < l$), when it lands in state 1. The parameter $\zeta \in [0, 1]$ is not configurable and has been added to avoid symmetries in the return surface.

The main goal of this experiment is to show the benefits of our algorithm compared to a simple gradient method, assuming to know the exact environment model. The return surface is characterized by two local maxima (Figure 2). If the system is initialized in a suitable region (as in Figure 2), to reach the global maximum we need to change the model in order to worsen the current policy performance. In Figure 3, we compare our algorithm REMPS using PROJ_{p^π} with different values of κ, against G(PO)MDP adapted to model learning (see Appendix B). We can see that G(PO)MDP, besides the slow convergence, moves in the direction of the local maximum. Instead, for some appropriate values of the hyperparameter (e.g., $\kappa \in \{0.1, 0.01\}$) REMPS is able to reach the global optimum. It is worth noting that too small a value of κ (e.g., $\kappa = 0.0001$) prevents escaping the basin of attraction of the local maximum. Likewise, for too large κ (e.g., $\kappa = 10$) the estimated quantities are too uncertain and therefore we are not able to reach the global optimum as well. Hyperparameter values and further experiments, including the effect of the different projection strategies, no-configuration cases, and the comparison with SPMI (Metelli et al., 2018), are reported in Appendix D.1.

6.2. Cartpole

The Cartpole domain (Widrow & Smith, 1964; Barto et al., 1983) is a continuous-state and finite-action environment. We add to the standard Cartpole domain the possibility to configure the cart force, via the parameter ω. We consider in the reward function an additional penalization proportional to the applied force, so that an optimal agent should find the smallest force that allows the pole to remain in a vertical position (details in Appendix D.2.1). The goal of this experiment is to test the ability of REMPS to learn jointly the policy and the environment configuration in a continuous state environment, as well as the effect of replacing the exact environment model with an approximator, trained just at the beginning of the learning process.

In Figure 4, we compare the performance of REMPS, with the two projection strategies PROJ_{p^π} and $\text{PROJ}_{\pi,p}$, and G(PO)MDP, starting from a fixed value of the model parameter ($\omega_0 = 8$), both for the case of exact model and approximate model. In the exact case, the performance of REMPS are similar to those of G(PO)MDP. The latter is even faster to achieve a good performance, although it shows a larger variance across the runs. No significant difference can be found between PROJ_{p^π} and $\text{PROJ}_{\pi,p}$ in this case. Instead, in the approximated scenario, REMPS notably outperforms G(PO)MDP, which shows a very unstable

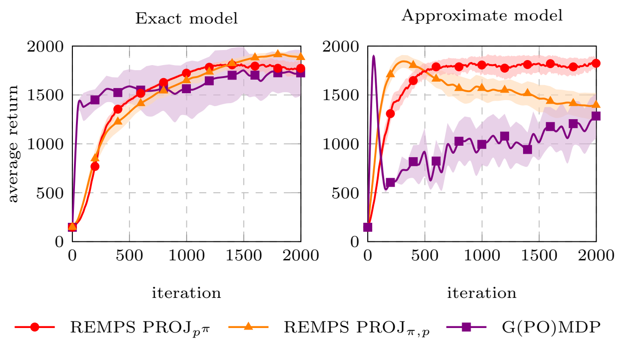

Figure 4. Average return as a function of the number of iterations for the Cartpole experiment when the environment model is exact (left) or approximated from samples (right) comparing REMPS with PROJ$_{p^\pi}$, PROJ$_{\pi,p}$ and G(PO)MDP. 20 runs, 95% c.i.

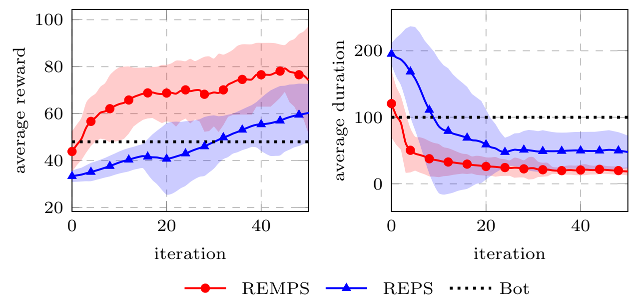

Figure 5. Average reward and episode duration as a function of the number of iterations for the TORCS experiment comparing REMPS, REPS and the bot. 10 runs, 80% c.i.

curve. Indeed, constraining the search in a trust-region, as REMPS does by means of κ, is even more important in the approximate case, since the estimated quantities are affected by further uncertainty (injected by the approximated model of the environment). It is worth noting that in this case the difference between PROJ$_{p^\pi}$ and PROJ$_{\pi,p}$ is more visible. Indeed, PROJ$_{\pi,p}$ is less precise than PROJ$_{p^\pi}$ (being a relaxation) and thus, when projecting d', it trusts the approximate model moving towards a suboptimal configuration. Further details on the Cartpole experiments are given in Appendix D.2.

6.3. Driving and Configuring with TORCS

The Open Racing Car Simulator TORCS (Wymann et al., 2000) is a car racing simulation that allows simulating driving races. TORCS has been used several times in RL (Loiacono et al., 2010; Koutník et al., 2013; Lillicrap et al., 2015; Mnih et al., 2016). We modified TORCS adding the possibility to configure the car parameters taking inspiration from the "Car Setup Competition" (Loiacono et al., 2013, details in Appendix D.3.1). The agent's observation is a low-dimensional representation of the car's sensors (including speed, focus and wheel speeds), while the action space is composed of steering and acceleration/braking (continuous).

The goal of this experiment is to show the ability of REMPS to learn policy and configuration in a continuous state-action space, like a car racing scenario. We consider a configuration space made of three parameters: rear and front wing orientation and brake repartition between front and rear. We start with a policy pretrained via behavioral cloning, using samples collected with a driving bot (snakeoil). Using the same bot, we collect a dataset of episodes with different parameter values, used to train an approximation of the environment. In Figure 5, we compare the average reward and the average lap time for REMPS (with PROJ$_{\pi,p}$), in which we act on both the policy and the model, and REPS, in which only policy learning is enabled. We can notice that REMPS is able to reach performances larger than those achievable without configuring the environment. In this experiment, we can appreciate another remarkable benefit of environment configurability: configuring the environment can also speed up the learning process (online performance), as clearly visible in Figure 5. Full experimental results can be found in Appendix D.3.

7. Discussion and Conclusions

Environment configurability is a relevant property of many real-world domains, with significant potential benefits when accounted by the RL algorithms. In this paper, we proposed a novel trust-region algorithm, REMPS, which takes advantage of this possibility to jointly learn an agent policy and an environment configuration. Unlike previous works, REMPS can be employed in continuous state-action spaces and does not require the knowledge of the exact environment dynamics. Furthermore, we derived several interesting properties of REMPS, especially we provided a finite-sample analysis for the single step of REMPS. Finally, the experimental evaluation showed that configuring the environment, on the one hand, allows the agent to learn highly performing policies; on the other hand, it might speed up the learning process itself. Moreover, REMPS showed the ability to overcome some of the limitations of gradient methods when employed to configure environments, even in the presence of approximate models. The future research directions include a more-in-depth analysis of REMPS, especially by studying the effect of dynamically modifying the KL-divergence threshold κ and the extension of the theoretical analysis for finite-time guarantees as well as for accounting of the approximate model of the environment. More generally, it is interesting to investigate diverse applications of the Conf-MDP framework, with particular attention of removing the knowledge of the agent policy space.

References

Barto, A. G., Sutton, R. S., and Anderson, C. W. Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE transactions on systems, man, and cybernetics*, (5):834–846, 1983.

Baxter, J. and Bartlett, P. L. Infinite-horizon policy-gradient estimation. *Journal of Artificial Intelligence Research*, 15:319–350, 2001.

Boyd, S. and Vandenberghe, L. *Convex optimization*. Cambridge university press, 2004.

Busic, A. and Meyn, S. Action-constrained markov decision processes with kullback-leibler cost. In *Conference On Learning Theory (COLT)*, 2018.

Ciosek, K. A. and Whiteson, S. Offer: Off-environment reinforcement learning. In *AAAI*, pp. 1819–1825, 2017.

Cortes, C., Mansour, Y., and Mohri, M. Learning bounds for importance weighting. In *Advances in Neural Information Processing Systems*, pp. 442–450, 2010.

Cortes, C., Greenberg, S., and Mohri, M. Relative deviation learning bounds and generalization with unbounded loss functions. *arXiv preprint arXiv:1310.5796*, 2013.

Daniel, C., Neumann, G., and Peters, J. Hierarchical relative entropy policy search. In *Artificial Intelligence and Statistics*, pp. 273–281, 2012.

Deisenroth, M. P. and Rasmussen, C. E. PILCO: A model-based and data-efficient approach to policy search. In Getoor, L. and Scheffer, T. (eds.), *Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011*, pp. 465–472. Omnipress, 2011.

Florensa, C., Held, D., Wulfmeier, M., Zhang, M., and Abbeel, P. Reverse curriculum generation for reinforcement learning. In *Conference on Robot Learning*, pp. 482–495, 2017.

Gelfand, I. M., Silverman, R. A., et al. *Calculus of variations*. Courier Corporation, 2000.

Guan, P., Raginsky, M., and Willett, R. M. Online markov decision processes with kullback–leibler control cost. *IEEE Transactions on Automatic Control*, 59(6):1423–1438, 2014.

Kakade, S. and Langford, J. Approximately optimal approximate reinforcement learning. In Sammut, C. and Hoffmann, A. G. (eds.), *Machine Learning, Proceedings of the Nineteenth International Conference (ICML 2002), University of New South Wales, Sydney, Australia, July 8-12, 2002*, pp. 267–274. Morgan Kaufmann, 2002.

Keren, S., Pineda, L., Gal, A., Karpas, E., and Zilberstein, S. Equi-reward utility maximizing design in stochastic environments. *HSDIP 2017*, pp. 19, 2017.

Koutník, J., Cuccu, G., Schmidhuber, J., and Gomez, F. Evolving large-scale neural networks for vision-based reinforcement learning. In *Proceedings of the 15th Annual Conference on Genetic and Evolutionary Computation*, GECCO '13, pp. 1061–1068, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-1963-8.

Levine, S. and Koltun, V. Guided policy search. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *JMLR Workshop and Conference Proceedings*, pp. 1–9. JMLR.org, 2013.

Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., and Wierstra, D. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

Loiacono, D., Prete, A., Lanzi, P. L., and Cardamone, L. Learning to overtake in torcs using simple reinforcement learning. In *IEEE Congress on Evolutionary Computation*, pp. 1–8, July 2010. doi: 10.1109/CEC.2010.5586191.

Loiacono, D., Cardamone, L., and Lanzi, P. L. Simulated car racing championship: Competition software manual. *arXiv preprint arXiv:1304.1672*, 2013.

Metelli, A. M., Mutti, M., and Restelli, M. Configurable markov decision processes. In Dy, J. G. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 3488–3497. PMLR, 2018.

Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T. P., Harley, T., Silver, D., and Kavukcuoglu, K. Asynchronous methods for deep reinforcement learning. In Balcan, M. and Weinberger, K. Q. (eds.), *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pp. 1928–1937. JMLR.org, 2016.

Mohri, M., Rostamizadeh, A., and Talwalkar, A. *Foundations of machine learning*. MIT press, 2012.

Nagabandi, A., Kahn, G., Fearing, R. S., and Levine, S. Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7559–7566. IEEE, 2018.

Owen, A. B. *Monte Carlo theory, methods and examples*. 2013.

Peters, J. and Schaal, S. Reinforcement learning of motor skills with policy gradients. *Neural networks*, 21(4):682–697, 2008.

Peters, J., Mülling, K., and Altun, Y. Relative entropy policy search. In *AAAI*, pp. 1607–1612. Atlanta, 2010.

Pirotta, M., Restelli, M., Pecorino, A., and Calandriello, D. Safe policy iteration. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *JMLR Workshop and Conference Proceedings*, pp. 307–315. JMLR.org, 2013.

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

Schulman, J., Levine, S., Abbeel, P., Jordan, M. I., and Moritz, P. Trust region policy optimization. In Bach, F. R. and Blei, D. M. (eds.), *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pp. 1889–1897. JMLR.org, 2015.

Seneta, E. Perturbation of the stationary distribution measured by ergodicity coefficients. *Advances in Applied Probability*, 20(1):228–230, 1988.

Silva, R., Melo, F. S., and Veloso, M. What if the world were different? gradient-based exploration for new optimal policies. *EPiC Series in Computing*, 55:229–242, 2018.

Sutton, R. S. and Barto, A. G. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

Sutton, R. S., McAllester, D. A., Singh, S. P., and Mansour, Y. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems*, pp. 1057–1063, 2000.

Todorov, E. Linearly-solvable markov decision problems. In *Advances in Neural Information Processing Systems*, pp. 1369–1376, 2007.

Todorov, E. Efficient computation of optimal actions. *Proceedings of the national academy of sciences*, 106(28): 11478–11483, 2009.

Van Erven, T. and Harremos, P. Rényi divergence and kullback-leibler divergence. *IEEE Transactions on Information Theory*, 60(7):3797–3820, 2014.

Widrow, B. and Smith, F. W. Pattern-recognizing control systems. *Computer and information sciences*, 288, 1964.

Williams, R. J. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992.

Wymann, B., Espié, E., Guionneau, C., Dimitrakakis, C., Coulom, R., and Sumner, A. Torcs, the open racing car simulator. 4:6, 2000.

A. Proofs and Derivations

In this Appendix, we provide the proofs and the derivations of the results presented in the main paper.

A.1. Proofs of Section 3.1

Theorem 3.1. *Let d be a distribution over $\mathcal{S} \times \mathcal{A} \times \mathcal{S}$ and $\kappa > 0$ a KL-divergence threshold. The solution d' of the problem PRIMAL$_\kappa$, for $\kappa > 0$, is given by:*

$$d'(s,a,s') \propto d(s,a,s') \exp\left(\frac{1}{\eta} r(s,a,s')\right), \tag{6}$$

where η is the unique solution of the dual problem DUAL$_\kappa$:

$$\min_{\eta \in [0,+\infty)} g(\eta) = \eta \log \mathop{\mathbb{E}}_{S,A,S' \sim d} \left[\exp\left(\frac{1}{\eta} r(S,A,S') + \kappa \right) \right].$$

Proof. For the sake of brevity, we define $\mathcal{X} = \mathcal{S} \times \mathcal{A} \times \mathcal{S}$ and $(s,a,s') = x \in \mathcal{X}$. We restate the PRIMAL$_\kappa$ problem in a more explicit form:

$$\max_{d'} \int_{\mathcal{X}} d'(x) r(x) \mathrm{d}x \tag{9}$$

$$\text{s.t.} \int_{\mathcal{X}} d'(x) \log \frac{d'(x)}{d(x)} \mathrm{d}x \le \kappa \tag{10}$$

$$\int_{\mathcal{X}} d'(x) \mathrm{d}x = 1, \tag{11}$$

where we simply made explicit the constraint guaranteeing that d' must sum up to one. Note that we do not need to ensure that $d'(x) \ge 0$ for all $x \in \mathcal{X}$ since this is guaranteed by the KL-divergence constraint. We solve the optimization problem using the Lagrange multipliers. We denote with $\eta \ge 0$ the Lagrange multiplier associated with the KL constraint (10) and with λ the multiplier associated with the constraint (11). The Lagrangian function becomes:

$$\begin{aligned}\mathcal{L}\left(d', \eta, \lambda\right) &= \int_{\mathcal{X}} d'(x) r(x) \mathrm{d}x + \eta \left(\kappa - \int_{\mathcal{X}} d'(x) \log \frac{d'(x)}{d(x)} \mathrm{d}x \right) + \lambda \left(1 - \int_{\mathcal{X}} d'(x) \mathrm{d}x \right) \\ &= \int_{\mathcal{X}} d'(x) \left(r(x) - \eta \log \frac{d'(x)}{d(x)} - \lambda \right) \mathrm{d}x + \eta\kappa + \lambda. \end{aligned} \tag{12}$$

Taking the functional derivative of $\mathcal{L}$ w.r.t. d' and applying a simple form of the Euler-Lagrange equation (Gelfand et al., 2000), we get:

$$\frac{\delta \mathcal{L}}{\delta d'(x)} = r(x) - \eta \log \frac{d'(x)}{d(x)} - \eta - \lambda = 0 \implies d'(x) = d(x) \exp\left(\frac{r(x)}{\eta} \right) \exp\left(-1 - \frac{\lambda}{\eta} \right). \tag{13}$$

We can derive an expression for d' by enforcing the constraint (11):

$$\exp\left(-1 - \frac{\lambda}{\eta} \right)^{-1} = \int_{\mathcal{X}} d(x) \exp\left(\frac{r(x)}{\eta} \right) \mathrm{d}x \implies d'(x) = \frac{d(x) \exp\left(\frac{r(x)}{\eta} \right)}{\int_{\mathcal{X}} d(x) \exp\left(\frac{r(x)}{\eta} \right) \mathrm{d}x}. \tag{14}$$

Substituting (13) into the Lagrangian function (12) and recalling (14), we obtain the dual function:

$$\begin{aligned} g(\eta, \lambda) &= \exp\left(-1 - \frac{\lambda}{\eta} \right) \int_{\mathcal{X}} d(x) \exp\left(\frac{r(x)}{\eta} \right) \left\{ r(x) - \eta \log \left[\exp\left(\frac{r(x)}{\eta} \right) \exp\left(-1 - \frac{\lambda}{\eta} \right) \right] - \lambda \right\} \mathrm{d}x + \eta\kappa + \lambda \\ &= \eta \exp\left(-1 - \frac{\lambda}{\eta} \right) \int_{\mathcal{X}} d(x) \exp\left(\frac{r(x)}{\eta} \right) \mathrm{d}x + \eta\kappa + \lambda \\ &= \eta + \eta\kappa + \lambda \\ &= \eta \log \left[\exp\left(-1 - \frac{\lambda}{\eta} \right)^{-1} \right] + \eta\kappa \\ &= \eta \log \int_{\mathcal{X}} d(x) \exp\left(\frac{r(x)}{\eta} \right) \mathrm{d}x + \eta\kappa \\ &= \eta \log \int_{\mathcal{X}} d(x) \exp\left(\frac{r(x)}{\eta} + \kappa \right) \mathrm{d}x. \end{aligned}$$

Making the change of variable $\overline{\eta} = 1/\eta$, we have that $\frac{1}{\overline{\eta}} \log \int_{\mathcal{X}} d(x) \exp\left(\overline{\eta} r(x)\right) \mathrm{d}x$ is convex (Boyd & Vandenberghe, 2004). Moreover, $\frac{\kappa}{\overline{\eta}}$ is strictly convex (as $\frac{\partial^2}{\partial \overline{\eta}^2} \frac{\kappa}{\overline{\eta}} = \frac{2\kappa}{\overline{\eta}^3} > 0$ for $\kappa > 0$), therefore their sum is strictly convex. Furthermore, function g is proper as it admits at least one feasible point (e.g., $\eta = 1$). Thus, being g strictly convex and proper, the optimization problem admits a unique solution (Boyd & Vandenberghe, 2004). $\square$

Corollary 3.1. *The solution d' of PRIMAL$_\kappa$ is induced by the configuration p' and the policy π' defined as:*

$$p'(s'|s,a) \propto p(s'|s,a) \exp\left(\frac{1}{\eta} r(s,a,s')\right),$$
$$\pi'(a|s) \propto \pi(a|s) \mathop{\mathbb{E}}_{S' \sim p(\cdot|s,a)} \left[\exp\left(\frac{1}{\eta} r(s,a,S')\right)\right].$$

Proof. Recall the definition of $d'(s,a,s') = d'(s)\pi'(a|s)p'(s'|s,a)$. Therefore we have:

$$p'(s'|s,a) = \frac{d'(s,a,s')}{d'(s)\pi'(a|s)} = \frac{d'(s,a,s')}{d'(s,a)} = \frac{d'(s,a,s')}{\int_{\mathcal{S}} d'(s',s,a)\mathrm{d}s'}.$$

Now, we substitute the expression of d':

$$\begin{aligned}
p'(s'|s,a) &= \frac{d(s,a,s') \exp\left(\frac{r(s,a,s')}{\eta}\right)}{\int_{\mathcal{S}} d(s,a,s') \exp\left(\frac{r(s,a,s')}{\eta}\right) \mathrm{d}s'} \\
&= \frac{d(s)\pi(a|s)p(s'|s,a) \exp\left(\frac{r(s,a,s')}{\eta}\right)}{d(s)\pi(a|s) \int_{\mathcal{S}} p(s'|s,a) \exp\left(\frac{r(s,a,s')}{\eta}\right) \mathrm{d}s'} \\
&= \frac{p(s'|s,a) \exp\left(\frac{r(s,a,s')}{\eta}\right)}{\int_{\mathcal{S}} p(s'|s,a) \exp\left(\frac{r(s,a,s')}{\eta}\right) \mathrm{d}s'}.
\end{aligned}$$

In a similar way for the policy, recall that $d'(s,a) = d'(s)\pi'(a|s)$, we have:

$$\pi'(a|s) = \frac{d'(s,a)}{d'(s)} = \frac{\int_{\mathcal{S}} d'(s,a,s')\mathrm{d}s'}{\int_{\mathcal{A}}\int_{\mathcal{S}} d'(s,a,s')\mathrm{d}s'\mathrm{d}a}.$$

Now, we substitute the expression of d' again:

$$\begin{aligned}
\pi'(a|s) &= \frac{\int_{\mathcal{S}} d(s,a,s') \exp\left(\frac{r(s,a,s')}{\eta}\right) \mathrm{d}s'}{\int_{\mathcal{A}}\int_{\mathcal{S}} d(s,a,s') \exp\left(\frac{r(s,a,s')}{\eta}\right) \mathrm{d}s'\mathrm{d}a} \\
&= \frac{d(s)\pi(a|s) \int_{\mathcal{S}} p(s'|s,a) \exp\left(\frac{r(s,a,s')}{\eta}\right) \mathrm{d}s'}{d(s) \int_{\mathcal{A}} \pi(a|s) \int_{\mathcal{S}} p(s'|s,a) \exp\left(\frac{r(s,a,s')}{\eta}\right) \mathrm{d}s'\mathrm{d}a} \\
&= \frac{\pi(a|s) \int_{\mathcal{S}} p(s'|s,a) \exp\left(\frac{r(s,a,s')}{\eta}\right) \mathrm{d}s'}{\int_{\mathcal{A}} \pi(a|s) \int_{\mathcal{S}} p(s'|s,a) \exp\left(\frac{r(s,a,s')}{\eta}\right) \mathrm{d}s'\mathrm{d}a}.
\end{aligned}$$

□

A.2. Proofs of Section 4.1

Proposition 4.1. *Let d and d' be two stationary distributions, then it holds that:*

$$|J_{d'} - J_d| \le r_{\max} \|d' - d\|_1 \le r_{\max}\sqrt{2D_{\mathrm{KL}}(d'\|d)}.$$

Proof. The first inequality is obtained with the following simple derivation:

$$|J_d - J_{d'}| = \left|\int \left(d(s,a,s') - d'(s,a,s')\right) r(s,a,s')\mathrm{d}s\mathrm{d}a\mathrm{d}s'\right| \le r_{\max} \int \left|d(s,a,s') - d'(s,a,s')\right| \mathrm{d}s\mathrm{d}a\mathrm{d}s'.$$

The second inequality is a straightforward application of the Pinsker's inequality. □

Corollary 4.1. *Let p^π and $p'^{\pi'}$ two transition kernels, inducing the stationary distributions d and d' respectively, then, under Assumption 4.2, it holds that:*

$$|J_{d'} - J_d| \le r_{\max}\rho\sqrt{2 \mathop{\mathbb{E}}_{S \sim d'} \left[D_{\mathrm{KL}}\left(p'^{\pi'}(\cdot|S)\|p^\pi(\cdot|S)\right)\right]},$$

where $\rho = \frac{\gamma}{1-\gamma}$ if $\gamma < 1$ or $\rho = \frac{1}{1-\tau_{\max}}$ if $\gamma = 1$.

Proof. If $\gamma < 1$, the statement is obtained starting from Theorem 4.1 and bounding $\|d' - d\|_1$ as in Proposition 3.1 of Metelli et al.

(2018):

$$\left\|d' - d\right\|_1 = \frac{\gamma}{1-\gamma} \mathop{\mathbb{E}}_{S\sim d'} \left[\left\| {p'}^{\pi'}(\cdot|S) - p^{\pi}(\cdot|S) \right\|_1 \right].$$

For the case $\gamma = 1$, we start from the following inequality provided in Seneta (1988) (Section 2, taking $p = \infty$) that we rewrite in our notation:

$$\begin{aligned}
\left\|d' - d\right\|_1 &\le \sum_{k=0}^{\infty} \tau\left(p^{\pi}\right)^k \int_{\mathcal{S}} \left| \int_{\mathcal{S}} d'(s) \left({p'}^{\pi'}(s'|s) - p^{\pi}(s'|s) \right) \mathrm{d}s \right| \mathrm{d}s' \\
&\le \int_{\mathcal{S}} d'(s) \int_{\mathcal{S}} \left| {p'}^{\pi'}(s'|s) - p^{\pi}(s'|s) \right| \mathrm{d}s' \mathrm{d}s \sum_{k=0}^{\infty} \tau_{\max}^k \\
&\le \frac{1}{1-\tau_{\max}} \mathop{\mathbb{E}}_{S\sim d'} \left[\left| p^{\pi}(\cdot|S) - {p'}^{\pi'}(\cdot|S) \right| \right],
\end{aligned}$$

where we exploited Assumption 4.2 for the bound $\tau\left(p^{\pi}\right) \le \tau_{\max}$. An application of Pinsker's inequality concludes the proof. ☐

Lemma 4.1. *Let (π, p) and (π', p') be two policy-configuration pairs and let p^{π} and ${p'}^{\pi'}$ the corresponding transition kernels, then for any state $s \in \mathcal{S}$, it holds that:*

$$\begin{aligned}
D_{\mathrm{KL}}\left({p'}^{\pi'}(\cdot|s) \| p^{\pi}(\cdot|s) \right) \le\ & D_{\mathrm{KL}}\left(\pi'(\cdot|s) \| \pi(\cdot|s) \right) \\
&+ \mathop{\mathbb{E}}_{A\sim\pi'(\cdot|s)} \left[D_{\mathrm{KL}}\left(p'(\cdot|s,A) \| p(\cdot|s,A) \right) \right].
\end{aligned}$$

Proof. We can prove the result by exploiting the properties of the KL-divergence:

$$\begin{aligned}
D_{\mathrm{KL}}\left({p'}^{\pi'}(\cdot|s) \| p^{\pi}(\cdot|s) \right) &= \int_{\mathcal{S}} {p'}^{\pi'}(s'|s) \log \frac{{p'}^{\pi'}(s'|s)}{p^{\pi}(s'|s)} \mathrm{d}s' \\
&= \int_{\mathcal{S}} \int_{\mathcal{A}} p'(s'|s,a)\pi'(a|s)\mathrm{d}a \log \frac{\int_{\mathcal{A}} p'(s'|s,a)\pi'(a|s)\mathrm{d}a}{\int_{\mathcal{A}} p(s'|s,a)\pi(a|s)\mathrm{d}a} \mathrm{d}s' \\
&\le \int_{\mathcal{S}} \int_{\mathcal{A}} p'(s'|s,a)\pi'(a|s) \log \frac{p'(s'|s,a)\pi'(a|s)}{p(s'|s,a)\pi(a|s)} \mathrm{d}a\mathrm{d}s' \qquad (15) \\
&= \int_{\mathcal{A}} \pi'(a|s) \int_{\mathcal{S}} p'(s'|s,a) \log \frac{p'(s'|s,a)}{p(s'|s,a)} \mathrm{d}s'\mathrm{d}a + \int_{\mathcal{A}} \pi'(a|s) \log \frac{\pi'(a|s)}{\pi(a|s)} \mathrm{d}a \\
&= \mathop{\mathbb{E}}_{A\sim\pi'(\cdot|s)} \left[D_{\mathrm{KL}}\left(p'(\cdot|s,A) \| p(\cdot|s,A) \right) \right] + D_{\mathrm{KL}}\left(\pi'(\cdot|s) \| \pi(\cdot|s) \right),
\end{aligned}$$

where (15) is an application of the integral log-sum inequality $\int_{\mathcal{X}} f(x) \log \frac{f(x)}{g(x)} \mathrm{d}x \ge F \log \frac{F}{G}$, where $F = \int_{\mathcal{X}} f(x)\mathrm{d}x$ and $G = \int_{\mathcal{X}} g(x)\mathrm{d}x$, that holds for any pair of integrable non negative functions f and g. ☐

A.3. Proofs of Section 4.2

In order to prove Proposition 4.2, we need a preliminary Lemma. Suppose that $\kappa' \le \kappa$, then the KL constraint is more restrictive, thus, we expect $J_{d'} \le J_d$. Let us consider a new class distributions $d_\alpha = \alpha d + (1-\alpha)d_0$, with $\alpha \in [0,1]$. Ideally, we could increase α until we saturate the constraint κ', getting a sort of projection of d over the region that satisfies the constraint induced by κ'.

Lemma A.1. *Let d and d' be the solution of the problems PRIMAL$_\kappa$ and PRIMAL$_{\kappa'}$ with $\kappa' \le \kappa$. Let $d_\alpha = \alpha d + (1-\alpha)d_0$ with $\alpha \in [0,1]$. If $D_{\mathrm{KL}}(d_\alpha \| d_0) = \kappa'$, then $\alpha \ge \frac{\kappa'}{\kappa}$.*

Proof. We use the convexity of the KL divergence: $D_{\mathrm{KL}}(\alpha\mu_1 + (1-\alpha)\mu_2 \| \alpha\nu_1 + (1-\alpha)\nu_2) \le \alpha D_{\mathrm{KL}}(\mu_1\|\nu_1) + (1-\alpha)D_{\mathrm{KL}}(\mu_2\|\nu_2)$ for $\alpha \in [0,1]$. Take $\mu_1 = d$, $\mu_2 = \nu_1 = \nu_2 = d_0$:

$$\begin{aligned}
\kappa' = D_{\mathrm{KL}}(d_\alpha\|d_0) = D_{\mathrm{KL}}(\alpha d + (1-\alpha)d_0 \| \alpha d_0 + (1-\alpha)d_0) \le \\
\le \alpha D_{\mathrm{KL}}(d\|d_0) + (1-\alpha)D_{\mathrm{KL}}(d_0\|d_0) = \alpha D_{\mathrm{KL}}(d\|d_0).
\end{aligned}$$

Therefore, observing that $D_{\mathrm{KL}}(d\|d_0) \le \kappa$:

$$\alpha \ge \frac{\kappa'}{D_{\mathrm{KL}}(d\|d_0)} \ge \frac{\kappa'}{\kappa}. \qquad (16)$$

☐

We can now prove Proposition 4.2.

Proposition 4.2. *Let d and d' be the solutions of PRIMAL$_\kappa$ and PRIMAL$_{\kappa'}$ respectively with $\kappa' \leq \kappa$, having d_0 as sampling distribution. Then, it holds that:*

$$J_d - J_{d'} \leq r_{\max} \|d - d_0\|_1 \left(1 - \frac{\kappa'}{\kappa}\right). \tag{7}$$

Proof. Consider the $\alpha' \in [0,1]$, as defined in Lemma A.1, such that $D_{\mathrm{KL}}(d_{\alpha'} \| d_0) = \kappa'$. We start observing that being d' the optimal solution with constraint κ' and since $d_{\alpha'}$ fulfills the constraint, we surely have $J_{d'} \geq J_{d_{\alpha'}}$. Consider the following sequence of inequalities:

$$\begin{aligned} J_d - J_{d'} &\leq J_d - J_{d_{\alpha'}} \\ &\leq r_{\max} \|d - d_{\alpha'}\|_1 \\ &\leq r_{\max} \|(1-\alpha')(d - d_0)\|_1 \\ &= r_{\max}(1-\alpha') \|d - d_0\|_1. \end{aligned}$$

Applying Lemma A.1 we get $1 - \alpha' \leq 1 - \frac{\kappa'}{\kappa}$, from which the result follows. □

A.4. Proofs of Section 4.3

For sake of brevity, we will denote with $\mathcal{X} = \mathcal{S} \times \mathcal{A} \times \mathcal{S}$ and with $x = (s, a, s')$ a state-action-next-state triple. In order to make the presentation clearer, we revise in the following the formulation of the optimization problems involved in REMPS.

A.4.1. Formulation of the optimization problems

The REMPS problem takes as input a stationary distribution $d_{\pi,p} \in \mathcal{D}_{\Pi,\mathcal{P}}$ and a KL–divergence threshold κ and provides as output a new stationary distribution in the space $\mathcal{D}_{\Pi,\mathcal{P}}$. This process is dived into two consecutive phases: *optimization* and *projection*.

Optimization In the *optimization* phase, given a KL–divergence threshold $\kappa > 0$, let $(\pi, p) \in \Pi \times \mathcal{P}$ be the current policy-configuration pair inducing a stationary distribution $d_{\pi,p}$, we seek for a new stationary distribution d' that solves the following optimization problem PRIMAL$_\kappa$:

$$\begin{aligned} \max_{d \in \Delta(\mathcal{X})} \; & J_d = \mathop{\mathbb{E}}_{X \sim d} \left[r(X) \right] \\ \text{s.t.} \quad & D_{\mathrm{KL}}(d \| d_{\pi,p}) = \mathop{\mathbb{E}}_{X \sim d} \left[\log \frac{d(X)}{d_{\pi,p}(X)} \right] \leq \kappa. \end{aligned}$$

This problem, yields to the solution:

$$d'(x) = \frac{d_{\pi,p}(x) \exp\left(\frac{1}{\eta} r(x)\right)}{\int_{\mathcal{X}} d_{\pi,p}(x) \exp\left(\frac{1}{\eta} r(x)\right) \mathrm{d}x}, \quad x \in \mathcal{X}, \tag{17}$$

where η is the unique solution of the dual problem DUAL$_\kappa$:

$$\min_{\eta \in [0,\infty)} \eta \log \mathop{\mathbb{E}}_{X \sim d_{\pi,p}} \left[\exp\left(\frac{1}{\eta} r(X) + \kappa \right) \right]. \tag{18}$$

In practice we have no access to $d_{\pi,p}$. Therefore, we need to estimate the expectations from samples using a dataset $\{(s_i, a_i, s'_i, r_i)\}_{i=1}^N = \{(x_i, r_i)\}_{i=1}^N$ (note that $r_i = r(x_i)$) of N samples collected with $d_{\pi,p}$. Notice that we have only access to an empirical estimate of $d_{\pi,p}$, which is $\widehat{d}_{\pi,p}(x) = \frac{1}{N}\sum_{i=1}^N \delta(x - x_i)$ uniform on the seen xs. Using $\widehat{d}_{\pi,p}$ we want to evaluate the performance of a candidate distribution d defined over the seen xs. For this purpose, we perform an importance weighting procedure. We define the weight $w(x_i) = \frac{d(x_i)}{\widehat{d}_{\pi,p}(x_i)} = N d(x_i)$. The problem we aim to solve becomes $\widetilde{\text{PRIMAL}}_\kappa$:

$$\begin{aligned} \max_{d \in \Delta(\{x_i : i \in \{1,2,\dots,N\}\})} \; & \widetilde{J}_d = \frac{1}{N} \sum_{i=1}^N w(x_i) r(x_i) = \sum_{i=1}^N d(x_i) r(x_i) \\ \text{s.t.} \quad & \widetilde{D}_{\mathrm{KL}}(d \| d_{\pi,p}) = \frac{1}{N} \sum_{i=1}^N w(x_i) \log w(x_i) = \sum_{i=1}^N d(x_i) \left(\log d(x_i) + \log N \right) \leq \kappa. \end{aligned}$$

This problem yields a solution which is defined only over the seen state-action-next-state triples:

$$d'(x_i) = \frac{\exp\left(\frac{1}{\widetilde{\eta}} r(x_i)\right)}{\frac{1}{N}\sum_{j=1}^{N} \exp\left(\frac{1}{\widetilde{\eta}} r(x_j)\right)}, \quad i \in \{1, 2, \dots, N\}, \tag{19}$$

where $\widetilde{\eta}$ is the unique solution of the dual problem $\widetilde{\mathrm{DUAL}}_{\kappa}$:

$$\min_{\eta \in [0,\infty)} \eta \log \frac{1}{N} \sum_{i=1}^{N} \exp\left(\frac{1}{\eta} r(x_i) + \kappa\right). \tag{20}$$

Once we solved this problem, the new distribution over the whole $\mathcal{X}$ is characterized by just the Lagrange multiplier $\widetilde{\eta}$:

$$\widetilde{d}'(x) = \frac{d_{\pi,p}(x) \exp\left(\frac{1}{\widetilde{\eta}} r(x)\right)}{\int_{\mathcal{X}} d_{\pi,p}(x) \exp\left(\frac{1}{\widetilde{\eta}} r(x)\right) \mathrm{d}x}, \quad x \in \mathcal{X}. \tag{21}$$

We denote the performance of the new stationary distribution $\widetilde{d}'$ as with $J_{\widetilde{d}'} = \mathbb{E}_{X \sim \widetilde{d}'}[r(X)]$.

Projection In the *projection* phase we aim at finding the best representation of the stationary distribution we got from the optimization phase in a given hypothesis space $\mathcal{D}_{\Pi,\mathcal{P}}$. Let d' be the solution of PRIMAL_{κ}, the projection problem PROJ can be stated as the moment-projection of d' onto $\mathcal{D}_{\Pi,\mathcal{P}}$. According to the three projections presented in Section 3.2, we have:

$$\mathrm{PROJ}_d \qquad \max_{\pi'' \in \Pi, p'' \in \mathcal{P}} H(d' \| d_{\pi'',p''}) = \mathop{\mathbb{E}}_{X \sim d'}\left[\log d_{\pi'',p''}(X)\right] + c,$$

$$\mathrm{PROJ}_{p^{\pi}} \qquad \max_{\pi'' \in \Pi, p'' \in \mathcal{P}} \mathbb{H}({p'}^{\pi'} \| {p''}^{\pi''}) = \mathop{\mathbb{E}}_{S,A,S' \sim d'}\left[H({p'}^{\pi'}(\cdot|S) \| {p''}^{\pi''}(\cdot|S))\right] = \mathop{\mathbb{E}}_{S,A,S' \sim d'}\left[\log {p''}^{\pi''}(\cdot|S)\right] + c,$$

$$\mathrm{PROJ}_{\pi,p} \qquad \max_{\pi'' \in \Pi} \mathbb{H}(\pi' \| \pi'') = \mathbb{E}_{S,A,S' \sim d'}\left[H(\pi'(\cdot|S) \| \pi''(\cdot|S))\right] = \mathbb{E}_{S,A,S' \sim d'}\left[\log \pi''(\cdot|S)\right] + c$$

$$\max_{p'' \in \mathcal{P}} \mathbb{H}(p' \| p'') = \mathbb{E}_{S,A,S' \sim d'}\left[H(p'(\cdot|S,A) \| p''(\cdot|S,A))\right] = \mathbb{E}_{S,A,S' \sim d'}\left[\log p''(\cdot|S,A)\right] + c,$$

where $H(d\|d')$ is the cross–entropy, since $D_{\mathrm{KL}}(d\|d') = H(d\|d') - H(d)$ and the entropy $H(d)$ is independent on d', and c denotes a constant that does not depend on the quantities we are optimizing on. Clearly, also in this case we need to consider the Monte Carlo estimates obtained from the very same samples $\{x_i\}_{i=1}^{N}$ collected with $d_{\pi,p}$. Let $\widetilde{d}'$ be the solution of $\widetilde{\mathrm{PRIMAL}}_{\kappa}$, the projection problem $\widetilde{\mathrm{PROJ}}$ becomes:

$$\widetilde{\mathrm{PROJ}}_d \qquad \max_{\pi'' \in \Pi, p'' \in \mathcal{P}} \widetilde{H}(\widetilde{d}' \| d_{\pi'',p''}) = \frac{1}{N} \sum_{i=1}^{N} w(x_i) \log d_{\pi'',p''}(x_i) + c,$$

$$\widetilde{\mathrm{PROJ}}_{p^{\pi}} \qquad \max_{\pi'' \in \Pi, p'' \in \mathcal{P}} \widetilde{\mathbb{H}}(\widetilde{{p'}^{\pi'}} \| {p''}^{\pi''}) = \frac{1}{N} \sum_{i=1}^{N} w(x_i) \log {p''}^{\pi''}(s_i'|s_i) + c,$$

$$\widetilde{\mathrm{PROJ}}_{\pi,p} \qquad \max_{\pi'' \in \Pi} \widetilde{\mathbb{H}}(\widetilde{\pi}' \| \pi'') = \frac{1}{N} \sum_{i=1}^{N} w(x_i) \log \pi''(a_i|s_i) + c$$

$$\max_{p'' \in \mathcal{P}} \widetilde{\mathbb{H}}(\widetilde{p}' \| p'') = \frac{1}{N} \sum_{i=1}^{N} w(x_i) \log p''(s_i'|s_i, a_i) + c,$$

A.4.2. Off-distribution estimation

Given a value of the Lagrange multiplier η inducing d, let us define the ratio importance weight $\widehat{w}(x)$ and the self-normalized importance weight $\widetilde{w}(x)$ as:

$$\widehat{w}(x) = \frac{d(x)}{d_{\pi,p}(x)} = \frac{\exp\left(\frac{1}{\eta} r(x)\right)}{\int_{\mathcal{X}} d_{\pi,p}(x) \exp\left(\frac{1}{\eta} r(x)\right) \mathrm{d}x}, \qquad \widetilde{w}(x) = \frac{\widehat{w}(x)}{\sum_{i=1}^{N} \widehat{w}(x_i)} = \frac{\exp\left(\frac{1}{\eta} r(x)\right)}{\sum_{i=1}^{N} \exp\left(\frac{1}{\eta} r(x_i)\right)}.$$

Thus, the off-distribution estimator $\widetilde{J}_d$ which is optimized by $\widetilde{\text{PRIMAL}}_\kappa$ is actually a *self–normalized importance weighting* estimate, opposed to the *ratio importance weighting* estimate $\widehat{J}_d$ which does not appear in the optimization problems, but will be useful in the following:

$$\widehat{J}_d = \frac{1}{N}\sum_{i=1}^{N} \widehat{w}(x_i)R(x_i), \qquad \widetilde{J}_d = \sum_{i=1}^{N} \widetilde{w}(x_i)R(x_i).$$

Analogously we can define the KL divergence estimators:

$$\widehat{D}_{KL}(d\|d_{\pi,p}) = \frac{1}{N}\sum_{i=1}^{N} \widehat{w}(x_i)\log \widehat{w}(x_i), \qquad \widetilde{D}_{KL}(d\|d_{\pi,p}) = \sum_{i=1}^{N} \widetilde{w}(x_i)\log\left(N\widetilde{w}(x_i)\right),$$

and, given a $d' \in \mathcal{D}_{\Pi,\mathcal{P}}$, we define the cross–entropy estimators:

$$\widehat{H}(d\|d') = \frac{1}{N}\sum_{i=1}^{N} \widehat{w}(x_i)\log d'(x_i), \qquad \widetilde{H}(d\|d') = \sum_{i=1}^{N} \widetilde{w}(x_i)\log d'(x_i).$$

It is well known that the ratio estimation is unbiased while the self-normalized estimator is biased but consistent (Owen, 2013).

A.4.3. Error Analysis

We have seen in the previous section that we need to solve both phases of the REMPS problem using the samples. Starting with $d_{\pi,p}$, PRIMAL_κ yields the solution d' whereas $\widetilde{\text{REMPS}}_\kappa$ provides the solution $d_{\widetilde{\pi}'',\widetilde{p}''}$ which is in terms derived from the $\widetilde{\text{PRIMAL}}_\kappa$ problem yielding $\widetilde{d}'$ and the $\widetilde{\text{PROJ}}$ problem. There are two sources of error in this process. First of all, $\widetilde{d}'$ is obtained from a finite sample and thus it may differ from d' (*estimation error*). Secondly, we limit to a hypothesis space $\mathcal{D}_{\Pi,\mathcal{P}}$ that may not be able to represent $\widetilde{d}'$ (*approximation error*). Furthermore, the projection is performed from samples as well (another source of estimation error). The goal of this analysis is to provide a bound to the quantity $J_{d'} - J_{\widetilde{\pi}'',\widetilde{p}''}$. To this end, we consider the following decomposition to isolate the contribution of the two phases:

$$J_{d'} - J_{\widetilde{\pi}'',\widetilde{p}''} = \underbrace{J_{d'} - J_{\widetilde{d}'}}_{\text{(i)}} + \underbrace{J_{\widetilde{d}'} - J_{\widetilde{\pi}'',\widetilde{p}''}}_{\text{(ii)}}.$$

Term (i) A typical approach, from Empirical Risk Minimization (ERM), to bound the estimation error is to add and subtract the empirical risk of the empirical risk minimizer $\widetilde{J}_{\widetilde{d}'}$ and exploit the fact that this quantity is larger (smaller in supervised learning) than the empirical risk of any other hypothesis in the hypothesis space (being ERM), in particular d'. However, in our framework, the hypothesis space changes since the constraint on the KL–divergence is estimated from samples and, in principle, it can impose more relaxed/tight conditions. For this purpose, we introduce a new distribution $\overline{d}$ which is the optimal solution to the PRIMAL_κ problem using the sample constraint. For this reason, $\widetilde{d}'$ and $\overline{d}$ are searched in the same hypothesis space and thus we can apply the theory from ERM. Clearly, we need to manage the discrepancy between $\overline{d}$ and d'. For this, we use the sensitivity analysis (Section 4.2). Let us define the discrepancy in the constraint for a given hypothesis d:

$$\Delta\kappa(d) = D_{\mathrm{KL}}(d\|d_{\pi,p}) - \widetilde{D}_{KL}(d\|d_{\pi,p}). \tag{22}$$

As a consequence $\widetilde{D}_{KL}(d\|d_{\pi,p}) \leq \kappa \iff D_{\mathrm{KL}}(d\|d_{\pi,p}) \leq \kappa + \Delta\kappa(d)$. Finally, we define $\Delta\kappa = \sup_{d\in\mathcal{D}_{d_{\pi,p}}} \Delta\kappa(d)$. We have the usual two cases. i) If $\Delta\kappa \leq 0$ then the exact constraint is always (i.e., for every hypothesis) tighter and thus $J_{\overline{d}} \geq J_d$. ii) If $\Delta\kappa > 0$ then there exists at least one hypothesis for which the constraint is looser; thus it might be that $J_{\overline{d}} \leq J_d$. In general, the following result holds.

Lemma A.2. *Let d', $\overline{d}$ as defined before. The following bound holds:*

$$J_{d'} \leq J_{\overline{d}} + 2r_{\max}\max\left\{0, \min\left\{\frac{1}{2}, \frac{\Delta\kappa}{\kappa}\right\}\right\}. \tag{23}$$

Proof. If $J_{d'} - J_{\overline{d}} \leq 0$ then the theorem holds. Otherwise, it must be that $\Delta\kappa(d') \geq 0$ (this is because we defined $\overline{d}$ as the optimal solution under the sample-based constraint). We define d_α as in Proposition 4.2, so we get:

$$\begin{aligned} J_{d'} - J_{\overline{d}} &\leq J_{d'} - J_{d_\alpha} \\ &\leq r_{\max}\left(1 - \frac{\kappa}{\kappa + \Delta\kappa(d')}\right)\|d' - d_{\pi,p}\|_1 \\ &\leq r_{\max}\frac{\Delta\kappa(d')}{\kappa + \Delta\kappa(d')}\|d' - d_{\pi,p}\|_1 \end{aligned}$$

$$\leq 2r_{\max} \min\left\{\frac{1}{2}, \frac{\Delta\kappa(d')}{\kappa}\right\}$$
$$\leq 2r_{\max} \min\left\{\frac{1}{2}, \frac{\Delta\kappa}{\kappa}\right\},$$

where we exploited the fact that $\|d' - d_{\pi,p}\|_1 \leq 2$, $\frac{\Delta\kappa(d')}{\kappa+\Delta\kappa(d')} \leq \frac{\Delta\kappa(d')}{\kappa}$, being $\Delta\kappa(d') \geq 0$, and $\frac{\Delta\kappa(d')}{\kappa+\Delta\kappa(d')} \leq \frac{1}{2}$ being $\Delta\kappa(d') \leq \kappa$ and finally $\Delta\kappa(d') \leq \Delta\kappa$. Taking the max between the two cases we get the result. $\square$

Notice that $\max\left\{0, \min\left\{\frac{1}{2}, \frac{\Delta\kappa}{\kappa}\right\}\right\} \leq \frac{|\Delta\kappa|}{\kappa} = \frac{1}{\kappa}\sup_{d\in\mathcal{D}_{d_{\pi,p}}}\left|\widetilde{D}_{KL}(d\|d_{\pi,p}) - D_{\mathrm{KL}}(d\|d_{\pi,p})\right|$, which is convenient for using ERM theory. Now we are ready to bound $J_{d'} - J_{\widetilde{d'}}$.

Lemma A.3. *Let d' and $\widetilde{d'}$ be the solutions of the PRIMAL$_\kappa$ and $\widetilde{\text{PRIMAL}}_\kappa$ problems, the latter using $N > 0$ i.i.d. samples collected with $d_{\pi,p}$. Let $\kappa > 0$ be the KL–divergence threshold. Then, it holds that:*

$$J_{d'} - J_{\widetilde{d'}} \leq 2\sup_{d\in\mathcal{D}_{d_{\pi,p}}}\left|J_d - \widetilde{J}_d\right| + \frac{2r_{\max}}{\kappa}\sup_{d\in\mathcal{D}_{d_{\pi,p}}}\left|\widetilde{D}_{KL}(d\|d_{\pi,p}) - D_{\mathrm{KL}}(d\|d_{\pi,p})\right|. \tag{24}$$

Proof. We use a very simple argument of ERM combined with the previous result. Let $\overline{d}$ be defined as before, we have:

$$\begin{aligned}
J_{d'} - J_{\widetilde{d'}} &\leq J_{\overline{d}} - J_{\widetilde{d'}} + 2r_{\max}\max\left\{0, \min\left\{\frac{1}{2}, \frac{\Delta\kappa}{\kappa}\right\}\right\} \\
&\leq J_{\overline{d}} - J_{\widetilde{d'}} + \frac{2r_{\max}}{\kappa}|\Delta\kappa| \\
&= J_{\overline{d}} - J_{\widetilde{d'}} + \frac{2r_{\max}}{\kappa}|\Delta\kappa| \pm \widetilde{J}_{\widetilde{d'}} \\
&\leq J_{\overline{d}} - \widetilde{J}_{\overline{d}} + \widetilde{J}_{\widetilde{d'}} - J_{\widetilde{d'}} + \frac{2r_{\max}}{\kappa}|\Delta\kappa| \\
&\leq 2\sup_{d\in\mathcal{D}_{d_{\pi,p}}}\left|J_d - \widetilde{J}_d\right| + \frac{2r_{\max}}{\kappa}\sup_{d\in\mathcal{D}_{d_{\pi,p}}}\left|\widetilde{D}_{KL}(d\|d_{\pi,p}) - D_{\mathrm{KL}}(d\|d_{\pi,p})\right|,
\end{aligned}$$

where we exploited the fact that $\widetilde{J}_{\overline{d}} \leq \widetilde{J}_{\widetilde{d'}}$, being $\widetilde{d'}$ the ERM over the same hypothesis space. $\square$

Term (ii) To bound this second term it is useful to recall the property of the KL–divergence $D_{\mathrm{KL}}(d\|d') = H(d\|d') - H(d)$, where $H(d\|d')$ is the cross-entropy between d and d' and $H(d)$ is the entropy of d. When performing the projection, we are minimizing the term $H(d\|d')$ since $H(d)$ does not depend on d'. We can state the following result for $\widetilde{\text{PROJ}}_d$.

Lemma A.4. *Let $\widetilde{d'}$ and $d_{\widetilde{\pi}'',\widetilde{p}''}$ be the solutions of the $\widetilde{\text{PRIMAL}}_\kappa$ and $\widetilde{\text{PROJ}}_d$ problems using $N > 0$ i.i.d. samples collected with $d_{\pi,p}$. Let $\kappa > 0$ be the KL–divergence threshold. Then, it holds that:*

$$J_{\widetilde{d'}} - J_{\widetilde{\pi}'',\widetilde{p}''} \leq r_{\max}\sqrt{2\sup_{d\in\mathcal{D}_{d_{\pi,p}}}\inf_{d'\in\mathcal{D}_{\Pi,\mathcal{P}}} D_{\mathrm{KL}}(d\|d')} + r_{\max}\sqrt{2\sup_{d\in\mathcal{D}_{d_{p,\pi}}}\sup_{d'\in\mathcal{D}_{\Pi,\mathcal{P}}}\left|\widehat{H}(d\|d') - H(d\|d')\right|}. \tag{25}$$

Proof. Let us define:

$$\epsilon_2 = \sup_{d\in\mathcal{D}_{d_{\pi,p}}}\sup_{d'\in\mathcal{D}_{\Pi,\mathcal{P}}}\left|\widehat{H}(d\|d') - H(d\|d')\right|. \tag{26}$$

Consider the best approximation of $\widetilde{d'}$ contained in $\mathcal{D}_{\Pi,\mathcal{P}}$, let us call it d^*, i.e., $d^* = \arg\min_{d\in\mathcal{D}_{\Pi,\mathcal{P}}} H(\widetilde{d'}\|d)$. Then we can state the following inequalities:

$$J_{\widetilde{d'}} - J_{\widetilde{\pi}'',\widetilde{p}''} \leq r_{\max}\left\|\widetilde{d'} - d_{\widetilde{\pi}'',\widetilde{p}''}\right\|_1$$
$$\leq r_{\max}\sqrt{2D_{\mathrm{KL}}(\widetilde{d'}\|d_{\widetilde{\pi}'',\widetilde{p}''})} \tag{27}$$
$$= r_{\max}\sqrt{2H(\widetilde{d'}\|d_{\widetilde{\pi}'',\widetilde{p}''}) - 2H(\widetilde{d'})}$$
$$\leq r_{\max}\sqrt{2\widehat{H}(\widetilde{d'}\|d_{\widetilde{\pi}'',\widetilde{p}''}) - 2H(\widetilde{d'}) + \epsilon_2} \tag{28}$$
$$= r_{\max}\sqrt{2\left(\frac{1}{N}\sum_{i=1}^{N}\widehat{w}(x_i)\right)\widetilde{H}(\widetilde{d'}\|d_{\widetilde{\pi}'',\widetilde{p}''}) - 2H(\widetilde{d'}) + \epsilon_2} \tag{29}$$

$$\leq r_{\max}\sqrt{2\left(\frac{1}{N}\sum_{i=1}^{N}\widehat{w}(x_i)\right)\widetilde{H}(\widetilde{d}'\|d^*) - 2H(\widetilde{d}') + \epsilon_2} \tag{30}$$

$$= r_{\max}\sqrt{2\widehat{H}(\widetilde{d}'\|d^*) - 2H(\widetilde{d}') + \epsilon_2}$$

$$\leq r_{\max}\sqrt{2H(\widetilde{d}'\|d^*) - 2H(\widetilde{d}') + 2\epsilon_2} \tag{31}$$

$$= r_{\max}\sqrt{2D_{\mathrm{KL}}(\widetilde{d}'\|d^*) + 2\epsilon_2} \tag{32}$$

$$\leq r_{\max}\sqrt{2D_{\mathrm{KL}}(\widetilde{d}'\|d^*)} + r_{\max}\sqrt{2\epsilon_2}$$

$$\leq r_{\max}\sqrt{2\sup_{d\in\mathcal{D}_{d_{\pi,p}}}\inf_{d'\in\mathcal{D}_{\Pi,\mathcal{P}}} D_{\mathrm{KL}}(d\|d')} + r_{\max}\sqrt{2\epsilon_2}, \tag{33}$$

where line (27) follows from Pinsker inequality, lines (28) and (31) follow from the hypothesis, line (30) follows from the fact that $d_{\widetilde{\pi}'',\widetilde{p}''}$ is ERM, line (32) follows from the inequality $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$ and lines (29) and (31) follow from the fact that $\left(\frac{1}{N}\sum_{i=1}^{N}\widehat{w}(x_i)\right)\widetilde{H}(d'\|d_{\widetilde{\pi}'',\widetilde{p}''}) = \widehat{H}(\widetilde{d}'\|d_{\widetilde{\pi}'',\widetilde{p}''})$. ◻

It is pretty straightforward to extent the previous result to the other two projections.

Corollary A.1. *Let $\widetilde{d}'$ and $d_{\widetilde{\pi}'',\widetilde{p}''}$ be the solutions of the $\widetilde{\mathit{PRIMAL}}_{\kappa}$ and $\widetilde{\mathit{PROJ}}_{p^{\pi}}$ problems using $N > 0$ i.i.d. samples collected with $d_{\pi,p}$. Let $\kappa > 0$ be the KL–divergence threshold. Then, it holds that:*

$$J_{\widetilde{d}'} - J_{d_{\widetilde{\pi}'',\widetilde{p}''}} \leq r_{\max}\rho\sqrt{2\sup_{d'\in\mathcal{D}_{d_{\pi,p}}}\inf_{p''^{\pi''}\in\mathcal{P}^{\Pi}}\mathop{\mathbb{E}}_{S\sim d}\left[D_{\mathrm{KL}}\left(p'^{\pi'}(\cdot|S)\|p''^{\pi''}(\cdot|S)\right)\right]}$$

$$+ r_{\max}\rho\sqrt{2\sup_{d\in\mathcal{D}_{d_{p,\pi}}}\sup_{p''^{\pi''}\in\mathcal{P}^{\Pi}}\left|\widehat{\mathbb{H}}(p'^{\pi'}\|p''^{\pi''}) - \mathbb{H}(p'^{\pi'}\|p''^{\pi''})\right|},$$

where we denote with $\mathcal{P}^{\Pi} = \{p^{\pi} : p \in \mathcal{P},\ \pi \in \Pi\}$ the set of state transition kernels induced by $\mathcal{P}$ and Π.

Let $\widetilde{d}'$ and $d_{\widetilde{\pi}'',\widetilde{p}''}$ be the solutions of the $\widetilde{\mathit{PRIMAL}}_{\kappa}$ and $\widetilde{\mathit{PROJ}}_{\pi,p}$ problems using $N > 0$ i.i.d. samples collected with $d_{\pi,p}$. Let $\kappa > 0$ be the KL–divergence threshold. Then, it holds that:

$$J_{\widetilde{d}'} - J_{d_{\widetilde{\pi}'',\widetilde{p}''}} \leq r_{\max}\rho\sqrt{2\sup_{d'\in\mathcal{D}_{d_{\pi,p}}}\inf_{\pi''\in\Pi}\mathop{\mathbb{E}}_{S\sim d}\left[D_{\mathrm{KL}}\left(\pi'(\cdot|S)\|\pi''(\cdot|S)\right)\right]}$$

$$+ r_{\max}\rho\sqrt{2\sup_{d\in\mathcal{D}_{d_{p,\pi}}}\sup_{\pi''\in\Pi}\left|\widehat{\mathbb{H}}(\pi'\|\pi'') - \mathbb{H}(\pi'\|\pi'')\right|}$$

$$+ r_{\max}\rho\sqrt{2\sup_{d'\in\mathcal{D}_{d_{\pi,p}}}\inf_{p''\in\mathcal{P}}\mathop{\mathbb{E}}_{S,A\sim d}\left[D_{\mathrm{KL}}\left(p'(\cdot|S,A)\|p''(\cdot|S,A)\right)\right]}$$

$$+ r_{\max}\rho\sqrt{2\sup_{d\in\mathcal{D}_{d_{p,\pi}}}\sup_{p''\in\mathcal{P}}\left|\widehat{\mathbb{H}}(p'\|p'') - \mathbb{H}(p'\|p'')\right|}.$$

Proof. The result is obtained using an approach analogous to that of Lemma A.4, using Corollary 4.1 and Lemma 4.1. ◻

From now on we will limit our attention to the case of PROJ_d. Putting all together we get the following result.

Theorem A.1. *(Error Decomposition) Let $\pi \in \Pi$ and $p \in \mathcal{P}$ be the current policy and transition model respectively. Let $\kappa > 0$ be the KL–divergence threshold. Let $d' \in \mathcal{D}_{d_{\pi,p}}$ be the solution of the PRIMAL_{κ} problem and $d_{\widetilde{\pi}'',\widetilde{p}''} \in \mathcal{D}_{\Pi,\mathcal{P}}$ be the solution of the $\widetilde{\mathit{REMPS}}_{\kappa}$ problem computed with $N > 0$ i.i.d. samples collected with $d_{\pi,p}$. Then, under Assumptions 4.1, it holds that:*

$$J_{d'} - J_{\widetilde{\pi}'',\widetilde{p}''} \leq 2\sup_{d\in\mathcal{D}_{d_{\pi,p}}}\left|J_d - \widetilde{J}_d\right| + \frac{2r_{\max}}{\kappa}\sup_{d\in\mathcal{D}_{d_{\pi,p}}}\left|\widetilde{D}_{KL}(d\|d_{\pi,p}) - D_{\mathrm{KL}}(d\|d_{\pi,p})\right|$$

$$+ r_{\max}\sqrt{2\sup_{d\in\mathcal{D}_{d_{\pi,p}}}\inf_{\overline{d}\in\mathcal{D}_{\Pi,\mathcal{P}}} D_{\mathrm{KL}}(\overline{d}\|d')} + r_{\max}\sqrt{2\sup_{d\in\mathcal{D}^{d_{p,\pi}}}\sup_{d'\in\mathcal{D}_{\Pi,\mathcal{P}}}\left|\widehat{H}(d\|d') - H(d\|d')\right|}.$$

Proof. Just sum together Lemma A.3 and Lemma A.4. ◻

A.4.4. FINITE–SAMPLE ANALYSIS FOR BOUNDED PROBABILITY DENSITIES

In the following, we will provide a finite–sample analysis of REMPS under the following assumption on the involved distributions.

Assumption A.1. *(Finite* $\sup$*, non–zero* $\inf$*) For every* $\pi \in \Pi$ *and a transition model* $p \in \mathcal{P}$*, for every* $d \in \mathcal{D}_{d_{\pi,p}}$ *and for every* $s, s' \in \mathcal{S}$ *and* $a \in \mathcal{A}$ *it holds that* $0 < m \leq d(s,a,s') \leq M < +\infty$ *and* $0 < m \leq d_{\pi,p}(s,a,s') \leq M < +\infty$*.*

This assumption ensures that all loss functions we are considering are uniformly bounded and allows us to state the sequence of useful facts.

Lemma A.5. *For any* $d \in \mathcal{D}_{d_{\pi,p}}$ *and for any* $d' \in \mathcal{D}_{\Pi,\mathcal{P}}$*. Under Assumption A.1, the following facts hold:*

1. *the importance weights are bounded above and below:* $\frac{m}{M} \leq \widehat{w}(x) \leq \frac{M}{m}$*;*
2. *the empirical KL divergence is bounded:* $\left|\widehat{D}_{KL}(d \| d_{\pi,p})\right| \leq \max\left\{\frac{1}{e}, \frac{M}{m}\log\frac{M}{m}\right\}$*;*
3. *the empirical cross–entropy is bounded:* $\left|\widehat{H}(d' \| d)\right| \leq \max\left\{-\frac{M}{m}\log m, \frac{M}{m}\log M\right\}$*;*
4. $\left|\widehat{D}_{KL}(d \| d_{\pi,p}) - \widetilde{D}_{KL}(d \| d_{\pi,p})\right| \leq \max\left\{\log\frac{M}{m} + 1, -2\log\frac{m}{M} - 1, \log N + 1\right\}\left|\frac{1}{N}\sum_{i=1}^N \widehat{w}(x_i) - 1\right|$*;*
5. $\left|\widehat{J}_d - \widetilde{J}_d\right| \leq r_{\max}\left|\frac{1}{N}\sum_{i=1}^N \widehat{w}(x_i) - 1\right|$*.*

Proof.

1. Immediate consequence of Assumption A.1, just observing that $\widehat{w}(x) = d(x)/d_{\pi,p}(x)$.
2. $|\widehat{D}_{KL}(d \| d_{\pi,p})| \leq \frac{1}{N}\sum_{i=1}^N |\widehat{w}(x_i)\log\widehat{w}(x_i)|$. Now, we know that $\widehat{w}(x) \leq \frac{M}{m}$ and that the function $|y\log y|$ has a local maximum whose value is $1/e$. As a consequence, $|\widehat{w}(x)\log\widehat{w}(x)| \leq \max\{1/e, M/m\log(M/m)\}$.
3. $|\widehat{H}(d \| d')| \leq \frac{1}{N}\sum_{i=1}^N |\widehat{w}(x_i)\log d'(x_i)|$. The maximum is attained when both $\widehat{w}(x)$ and $|\log d'(x)|$ are maximum. $\widehat{w}(x) \leq M/m$, while $|\log d'(x)| \leq \max\{-\log m, \log M\}$.
4. The absolute derivative of $y\log y$ is $|\log y + 1|$. Consider the term $\widehat{w}(x_i)\log\widehat{w}(x_i)$, we know that $m/M \leq \widehat{w}(x_i) \leq M/m$, therefore the maximum absolute derivative has value $\max\{\log(M/m) + 1, -\log(m/M) - 1\}$. Consider the term $N\widetilde{w}(x_i) = N\widehat{w}(x_i)/\sum_{i=1}^N \widehat{w}(x_i)$. We know that $(m/M)^2 \leq N\widetilde{w}(x_i) \leq N$, thus the maximum absolute derivative has value $\max\{\log(N) + 1, -2\log(m/M) - 1\}$. Since the Lipschitz constant of an average is smaller or equal to the Lipschitz constant of each term, we get the result.
5. Consider the following inequalities:

$$
\begin{aligned}
\left|\widehat{J}_d - \widetilde{J}_d\right| &= \left|\frac{1}{N}\sum_{i=1}^N \widehat{w}(x_i)r(x_i) - \frac{\sum_{i=1}^N \widehat{w}(x_i)r(x_i)}{\sum_{i=1}^N \widehat{w}(x_i)}\right| \\
&= \left|\frac{\sum_{i=1}^N \widehat{w}(x_i)r(x_i)}{\sum_{i=1}^N \widehat{w}(x_i)}\left(\frac{1}{N}\sum_{i=1}^N \widehat{w}(x_i) - 1\right)\right| \\
&\leq r_{\max}\left|\frac{1}{N}\sum_{i=1}^N \widehat{w}(x_i) - 1\right|.
\end{aligned}
$$

□

We report now a standard result of learning theory that we are going to use extensively throughout the analysis (Mohri et al., 2012).

Theorem A.2. *Let* $\mathcal{H}$ *be a family real-valued functions and let* $\mathcal{G} = \{L_h(x) : h \in \mathcal{H}\}$ *be the family of loss functions associated to* $\mathcal{H}$*. Assume that* $\mathrm{Pdim}(\mathcal{G}) = v$ *and that the loss function* L *is bounded by* M*. Then, for any* $\delta \in (0,1)$*, with probability at least* $1-\delta$*, for all* $h \in \mathcal{H}$ *it holds that:*

$$
\mathop{\mathbb{E}}_{X}\left[L_h(X)\right] \leq \frac{1}{N}\sum_{i=1}^N L_h(x_i) + M\sqrt{\frac{8v\log\frac{2eN}{v} + 8\log\frac{4}{\delta}}{N}},
$$

and also, with probability at least $1-\delta$, for all $h \in \mathcal{H}$ it holds that:

$$\frac{1}{N}\sum_{i=1}^{N} L_h(x_i) \leq \mathop{\mathbb{E}}_{X}\left[L_h(X)\right] + M\sqrt{\frac{8v\log\frac{2eN}{v} + 8\log\frac{4}{\delta}}{N}}.$$

Using this result, we immediately derive the following.

Lemma A.6. *Under Assumption A.1, each of these events hold with probability at least $1-\delta$:*

$$(\mathcal{E}_1) \qquad \forall d \in \mathcal{D}_{d_{\pi,p}}: \quad \left|\frac{1}{N}\sum_{i=1}^{N}\widehat{w}(x_i) - 1\right| \leq \frac{M}{m}\sqrt{\frac{8v\log\frac{2eN}{v} + 8\log\frac{8}{\delta}}{N}};$$

$$(\mathcal{E}_2) \qquad \forall d \in \mathcal{D}_{d_{\pi,p}}: \quad \left|\widehat{J}_d - J_d\right| \leq r_{\max}\frac{M}{m}\sqrt{\frac{8v\log\frac{2eN}{v} + 8\log\frac{8}{\delta}}{N}};$$

$$(\mathcal{E}_3) \qquad \forall d \in \mathcal{D}_{d_{\pi,p}}: \quad \left|\widehat{D}_{KL}(d\|d_{\pi,p}) - D_{\mathrm{KL}}(d\|d_{\pi,p})\right| \leq \max\left\{\frac{1}{e}, \frac{M}{m}\log\frac{M}{m}\right\}\sqrt{\frac{8v\log\frac{2eN}{v} + 8\log\frac{8}{\delta}}{N}};$$

$$(\mathcal{E}_4) \qquad \forall d \in \mathcal{D}_{d_{\pi,p}}, \forall d' \in \mathcal{D}_{\Pi,\mathcal{P}}: \quad \left|\widehat{H}(d'\|d) - H(d'\|d)\right| \leq \frac{M}{m}\max\left\{-\log m, \log M\right\}\sqrt{\frac{8v\log\frac{2eN}{v} + 8\log\frac{8}{\delta}}{N}}.$$

Proof. It is a trivial application of Theorem A.2, by observing that we need a bilateral bound, by carefully defining the maximum of each function involved and exploiting Assumption 4.3. □

We can now put all together.

Theorem A.3. *(Finite–Sample Bound under Assumption A.1) Let $\pi \in \Pi$ and $p \in \mathcal{P}$ be the current policy and transition model respectively. Let $\kappa > 0$ be the KL–divergence threshold. Let $d' \in \mathcal{D}_{d_{\pi,p}}$ be the solution of the PRIMAL$_\kappa$ problem and $d_{\widetilde{\pi}'',\widetilde{p}''} \in \mathcal{D}_{\Pi,\mathcal{P}}$ be the solution of the $\widetilde{\text{REMPS}}_\kappa$ problem computed with $N > 0$ samples collected with $d_{\pi,p}$. Then, under Assumptions 4.1, 4.3 and A.1, there exists a constant ϕ and function $\psi(N) = \mathcal{O}(\log N)$, such that for any $\delta \in (0,1)$, with probability at least $1-4\delta$ it holds that:*

$$J_d - J_{\widetilde{\pi}'',\widetilde{p}''} \leq \sqrt{2}r_{\max}\sup_{d \in \mathcal{D}_{d_{\pi,p}}}\inf_{\overline{d} \in \mathcal{D}_{\Pi,\mathcal{P}}}\sqrt{D_{\mathrm{KL}}(d\|\overline{d})} + r_{\max}\phi\sqrt[4]{\frac{8v\log\frac{2eN}{v} + 8\log\frac{8}{\delta}}{N}} + r_{\max}\psi(N)\sqrt{\frac{8v\log\frac{2eN}{v} + 8\log\frac{8}{\delta}}{N}}.$$

Proof. We start from Theorem A.1 and we bound each term using Lemma A.5 and Lemma A.6. Let us start with $\sup_{d\in\mathcal{D}_{d_{\pi,p}}}\left|J_d - \widetilde{J}_d\right|$:

$$\begin{aligned}
\sup_{d \in \mathcal{D}_{d_{\pi,p}}}\left|J_d - \widetilde{J}_d\right| &= \sup_{d \in \mathcal{D}_{d_{\pi,p}}}\left|J_d - \widetilde{J}_d \pm \widehat{J}_d\right| \\
&\leq \sup_{d \in \mathcal{D}_{d_{\pi,p}}}\left|J_d - \widehat{J}_d\right| + \sup_{d \in \mathcal{D}_{d_{\pi,p}}}\left|\widehat{J}_d - \widetilde{J}_d\right| \\
&\leq \sup_{d \in \mathcal{D}_{d_{\pi,p}}}\left|J_d - \widehat{J}_d\right| + r_{\max}\sup_{d \in \mathcal{D}_{d_{\pi,p}}}\left|\frac{1}{N}\sum_{i=1}^{N}\widehat{w}(x_i) - 1\right| \\
&\leq 2r_{\max}\frac{M}{m}\sqrt{\frac{8v\log\frac{2eN}{v} + 8\log\frac{8}{\delta}}{N}},
\end{aligned}$$

where we exploited events $(\mathcal{E}_1)$ and $(\mathcal{E}_2)$. Consider $\sup_{d\in\mathcal{D}_{d_{\pi,p}}}\left|\widetilde{D}_{KL}(d\|d_{\pi,p}) - D_{\mathrm{KL}}(d\|d_{\pi,p})\right|$:

$$\begin{aligned}
\sup_{d \in \mathcal{D}_{d_{\pi,p}}}\left|\widetilde{D}_{KL}(d\|d_{\pi,p}) - D_{\mathrm{KL}}(d\|d_{\pi,p})\right| &= \sup_{d \in \mathcal{D}_{d_{\pi,p}}}\left|\widetilde{D}_{KL}(d\|d_{\pi,p}) - D_{\mathrm{KL}}(d\|d_{\pi,p}) \pm \widehat{D}_{KL}(d\|d_{\pi,p})\right| \\
&\leq \sup_{d \in \mathcal{D}_{d_{\pi,p}}}\left|D_{KL}(d\|d_{\pi,p}) - \widehat{D}_{KL}(d\|d_{\pi,p})\right| + \sup_{d \in \mathcal{D}_{d_{\pi,p}}}\left|\widetilde{D}_{KL}(d\|d_{\pi,p}) - \widehat{D}_{KL}(d\|d_{\pi,p})\right| \\
&\leq \sup_{d \in \mathcal{D}_{d_{\pi,p}}}\left|D_{KL}(d\|d_{\pi,p}) - \widehat{D}_{KL}(d\|d_{\pi,p})\right|
\end{aligned}$$

$$+\sup_{d\in\mathcal{D}_{d_{\pi,p}}} \max\left\{\log\frac{M}{m}+1, -2\log\frac{m}{M}-1, \log N+1\right\}\left|\frac{1}{N}\sum_{i=1}^{N}\widehat{w}(x_i)-1\right|$$

$$\leq \left(\max\left\{\frac{1}{e},\frac{M}{m}\log\frac{M}{m}\right\}+\max\left\{\log\frac{M}{m}+1, -2\log\frac{m}{M}-1, \log N+1\right\}\right)\sqrt{\frac{8v\log\frac{2eN}{v}+8\log\frac{8}{\delta}}{N}}$$

$$\leq f(N)\sqrt{\frac{8v\log\frac{2eN}{v}+8\log\frac{8}{\delta}}{N}},$$

where we defined $f(N)=\max\left\{\frac{1}{e},\frac{M}{m}\log\frac{M}{m}\right\}+\max\left\{\log\frac{M}{m}+1,-2\log\frac{m}{M}-1,\log N+1\right\}=\mathcal{O}(\log N)$ and we exploited events $(\mathcal{E}_1)$ and $(\mathcal{E}_3)$. Finally, the term $\sup_{d\in\mathcal{D}^{d_{p,\pi}}}\sup_{d'\in\mathcal{D}_{\Pi,\mathcal{P}}}\left|\widehat{H}(d\|d')-H(d\|d')\right|$ can be bounded using Lemma A.6. Let us define $c=\frac{M}{m}\max\{-\log m,\log M\}$ and $\epsilon=\sqrt{\frac{8v\log\frac{2eN}{v}+8\log\frac{8}{\delta}}{N}}$ and we put all together we get:

$$J_{d'}-J_{\widetilde{\pi}'',\widetilde{p}''}\leq 4r_{\max}\epsilon+\frac{2r_{\max}}{\kappa}f(N)\epsilon+r_{\max}\sqrt{2}\sup_{d\in\mathcal{D}_{d_{\pi,p}}}\inf_{\overline{d}\in\mathcal{D}_{\Pi,\mathcal{P}}}\sqrt{D_{\mathrm{KL}}(d\|\overline{d})}+r_{\max}\sqrt{2c\epsilon}$$

$$=r_{\max}\sqrt{2}\sup_{d\in\mathcal{D}_{d_{\pi,p}}}\inf_{\overline{d}\in\mathcal{D}_{\Pi,\mathcal{P}}}\sqrt{D_{\mathrm{KL}}(d\|\overline{d})}+r_{\max}\sqrt{\epsilon}\left(\left(4+\frac{2}{\kappa}f(N)\right)\sqrt{\epsilon}+\sqrt{2c}\right)$$

$$=r_{\max}\sqrt{2}\sup_{d\in\mathcal{D}_{d_{\pi,p}}}\inf_{\overline{d}\in\mathcal{D}_{\Pi,\mathcal{P}}}\sqrt{D_{\mathrm{KL}}(d\|\overline{d})}+r_{\max}\phi\sqrt{\epsilon}+r_{\max}\psi(N)\epsilon,$$

where we renamed $\psi(N)=4+\frac{2}{\kappa}f(N)$ and $\phi=\sqrt{2c}$. Notice that $\psi_\kappa(N)=\mathcal{O}(\log N)$. Since we made a union bound over the events $(\mathcal{E}_1)$, $(\mathcal{E}_2)$, $(\mathcal{E}_3)$ and $(\mathcal{E}_4)$, the statement holds with probability $1-4\delta$. □

A.4.5. Finite–Sample Analysis for finite β-moments

In the following, we consider a more realistic set of assumptions (Assumption 4.4). This second analysis poses two main challenges. First, we are not guaranteed that the involved loss functions have finite supremum. This problem can be tackled by resorting to learning bounds that are applicable to unbounded loss functions with bounded moments (Cortes et al., 2013). Second, the analysis of the KL–divergence estimated with self–normalized importance sampling is more complex.

The main theoretical tool we are going to use in the following comes from Cortes et al. (2013).

Theorem A.4. *Let $\mathcal{H}$ be a family real-valued functions and let $\mathcal{G}=\{L_h(x):h\in\mathcal{H}\}$ be the family of loss functions associated to $\mathcal{H}$. Assume that $\mathrm{Pdim}(\mathcal{G})=v$ and that there exists $\alpha\in(1,2)$ such that $\sup_{h\in\mathcal{H}}L_\alpha(h)=\mathbb{E}_X\left[|L_h(X)|^\alpha\right]<+\infty$. Let $\widehat{L}_\alpha(h)=\frac{1}{N}\sum_{i=1}^N|L_h(x_i)|^\alpha$. Then, for any $\delta\in(0,1)$, with probability at least $1-\delta$, for all $h\in\mathcal{H}$ it holds that:*

$$\mathop{\mathbb{E}}_{X}\left[L_h(X)\right]\leq\frac{1}{N}\sum_{i=1}^{N}L_h(x_i)+2^{\frac{\alpha+2}{2\alpha}}\sqrt[\alpha]{L_\alpha(h)}\sqrt{\frac{v\log\frac{2eN}{v}+\log\frac{4}{\delta}}{N^{\frac{2(\alpha-1)}{\alpha}}}}\Gamma\left(\alpha,\sqrt{\frac{v\log\frac{2eN}{v}+\log\frac{4}{\delta}}{N^{\frac{2(\alpha-1)}{\alpha}}}}\right),$$

and also, with probability at least $1-\delta$, for all $h\in\mathcal{H}$ it holds that:

$$\frac{1}{N}\sum_{i=1}^{N}L_h(x_i)\leq\mathop{\mathbb{E}}_{X}\left[L_h(X)\right]+2^{\frac{\alpha+2}{2\alpha}}\sqrt[\alpha]{\widehat{L}_\alpha(h)}\sqrt{\frac{v\log\frac{2eN}{v}+\log\frac{4}{\delta}}{N^{\frac{2(\alpha-1)}{\alpha}}}}\Gamma\left(\alpha,\sqrt{\frac{v\log\frac{2eN}{v}+\log\frac{4}{\delta}}{N^{\frac{2(\alpha-1)}{\alpha}}}}\right),$$

where $\Gamma(\alpha,\epsilon)=\frac{\alpha-1}{\alpha}+\frac{1}{\alpha}\left(\frac{\alpha}{\alpha-1}\right)^{\alpha-1}\left(1+\left(\frac{\alpha-1}{\alpha}\right)^{\alpha-1}\log\frac{1}{\epsilon}\right)^{\frac{\alpha-1}{\alpha}}$.

In the following statements, we make use of the Rényi divergence between probability distributions (Cortes et al., 2010). Given two probability distributions P and Q admitting p and q as density functions. The α–Rényi divergence between p and q is given by:

$$D_\alpha(p\|q)=\frac{1}{1-\alpha}\log\mathop{\mathbb{E}}_{X\sim q}\left[\left(\frac{p(X)}{q(X)}\right)^\alpha\right],\qquad for\,\alpha\in[0,\infty]. \tag{34}$$

We define the exponentiated Rényi divergence as $d_\alpha(p\|q)=\exp\left(D_\alpha(p\|q)\right)$.

We start by showing a trivial application of Theorem A.4 for bounding in probability several deviations of interest.

Lemma A.7. *Let us define $\epsilon=2^{\frac{\beta+2}{2\beta}}\sqrt{\frac{v\log\frac{2eN}{v}+\log\frac{8}{\delta}}{N^{\frac{2(\beta-1)}{\beta}}}}\Gamma\left(\beta,\sqrt{\frac{v\log\frac{2eN}{v}+\log\frac{8}{\delta}}{N^{\frac{2(\beta-1)}{\beta}}}}\right)$. Under Assumption 4.4, each of these events holds with probability at least $1-\delta$:*

$$(\mathcal{E}_1) \qquad \forall d \in \mathcal{D}_{d_{\pi,p}}: \quad \left| \frac{1}{N}\sum_{i=1}^N \widehat{w}(x_i) - 1 \right| \le \max\left\{ \sqrt[\beta]{d_\beta(d\|d_{\pi,p})}, \sqrt[\beta]{\widehat{d}_\beta(d\|d_{\pi,p})} \right\}\epsilon;$$

$$(\mathcal{E}_2) \qquad \forall d \in \mathcal{D}_{d_{\pi,p}}: \quad \left| \widehat{J}_d - J_d \right| \le r_{\max}\left\{ \sqrt[\beta]{d_\beta(d\|d_{\pi,p})}, \sqrt[\beta]{\widehat{d}_\beta(d\|d_{\pi,p})} \right\}\epsilon;$$

$$(\mathcal{E}_4) \qquad \forall d \in \mathcal{D}_{d_{\pi,p}}, \forall d' \in \mathcal{D}_{\Pi,\mathcal{P}}:$$

$$\left| \widehat{H}(d'\|d) - H(d'\|d) \right| \le \max\left\{ \mathop{\mathbb{E}}_{X\sim d_{\pi,p}}\left[\left| \frac{d(X)}{d_{\pi,p}(X)} \log d'(X) \right|^\beta \right]^{1/\beta}, \left(\frac{1}{N}\sum_{i=1}^N \left| \frac{d(x_i)}{d_{\pi,p}(x_i)} \log d'(x_i) \right|^\beta \right)^{1/\beta} \right\}\epsilon.$$

Proof. It is a simple application of Theorem A.4, using Assumption 4.4 and applying definition (34). □

Concerning the KL–divergence, the derivation is a bit more complicated. We first need the following technical lemma.

Lemma A.8. *Under Assumption 4.4, for any $\alpha \in (1, \beta)$, the following inequality holds:*

$$\mathop{\mathbb{E}}_{X\sim d_{\pi,p}}\left[\left| \frac{d(x)}{d_{\pi,p}(x)} \log \frac{d(x)}{d_{\pi,p}(x)} \right|^\alpha \right]^{1/\alpha} \le \frac{1}{e} + \frac{\alpha}{\beta-\alpha} \mathop{\mathbb{E}}_{X\sim d_{\pi,p}}\left[\left| \frac{d(x)}{d_{\pi,p}(x)} \right|^\beta \right]^{1/\alpha} = \frac{1}{e} + \frac{\alpha}{\beta-\alpha} d_\beta(d\|d_{\pi,p})^{\beta/\alpha}. \tag{35}$$

Proof. Let $y = d(x)/d_{\pi,p}(x)$. We start proving that the following inequality hold for all $\alpha > 1$:

$$|y \log y| \le \max\left\{ \frac{1}{e}, \frac{y^\alpha}{\alpha - 1} \right\}. \tag{36}$$

Let $g(y) = |y\log y|$. For $y \in [0,1]$ we know that $y \log y$ is negative, thus $g(y) = -y\log y$ that has $1/e$ as maximum. Just take the derivative $\partial g/\partial y = -\log y - 1 = 0 \implies y = 1/e \implies g(1/e) = 1/e$. Clearly the second derivative is negative, thus $1/e$ is a maximum and at the extremes $g(0) = g(1) = 0 < 1/e$. We prove that for $y \in [1,\infty)$, $g(y) = y\log y \le \frac{y^\alpha}{\alpha-1}$. It suffices to prove that $\log y \le \frac{y^{\alpha-1}}{\alpha-1}$. Consider the function $h(y) = \log y - \frac{y^{\alpha-1}}{\alpha-1}$, it is enough to prove that $h(y) \le 0$ for all $y \in [1,\infty)$. We know that $h(1) = -\frac{1}{\alpha-1} < 0$ and $h(\infty) = -\infty$ and continuous. Therefore we consider the derivative:

$$\frac{\partial h}{\partial y} = \frac{1}{y} - y^{\alpha-2} \le 0 \implies y \ge 1. \tag{37}$$

Thus $h(y)$ is monotonically decreasing in $[1,\infty)$ and therefore the statement holds. Now we observe that $\max\{x,y\} \le x + y$ for $x, y \ge 0$ and we get using Minkowski:

$$\begin{aligned}\mathop{\mathbb{E}}_{X\sim d_{\pi,p}}\left[\left| \frac{d(x)}{d_{\pi,p}(x)} \log \frac{d(x)}{d_{\pi,p}(x)} \right|^\alpha \right]^{1/\alpha} &\le \mathop{\mathbb{E}}_{X\sim d_{\pi,p}}\left[\left(\frac{1}{e} + \frac{1}{\gamma-1}\left(\frac{d(x)}{d_{\pi,p}(x)} \right)^\gamma \right)^\alpha \right]^{1/\alpha} \\ &\le \frac{1}{e} + \frac{1}{\gamma-1} \mathop{\mathbb{E}}_{X\sim d_{\pi,p}}\left[\left(\frac{d(x)}{d_{\pi,p}(x)} \right)^{\gamma\alpha} \right]^{1/\alpha}.\end{aligned}$$

By taking $\gamma\alpha = \beta$ we get the result. □

An immediate consequence is the following result.

Lemma A.9. *For any $\alpha \in (1,2)$, let $\epsilon = 2^{\frac{\alpha+2}{2\alpha}} \sqrt{\frac{v \log \frac{2eN}{v} + \log \frac{8}{\delta}}{N^{\frac{2(\alpha-1)}{\alpha}}}}\,\Gamma\left(\alpha, \sqrt{\frac{v \log \frac{2eN}{v} + \log \frac{8}{\delta}}{N^{\frac{2(\alpha-1)}{\alpha}}}}\right)$. For any $\alpha \in (1,\beta)$, under Assumption 4.4, the following inequality holds with probability $1-\delta$:*

$$(\mathcal{E}_3) \qquad \forall d \in \mathcal{D}_{d_{\pi,p}}:$$

$$\left| \widehat{D}_{KL}(d\|d_{\pi,p}) - D_{\mathrm{KL}}(d\|d_{\pi,p}) \right| \le \max\left\{ \frac{1}{e} + \frac{\alpha}{\beta-\alpha} \mathop{\mathbb{E}}_{x\sim q}\left[\left| \frac{d(x)}{d_{\pi,p}(x)} \right|^\beta \right]^{1/\alpha}, \left(\frac{1}{N}\sum_{i=1}^N |\widehat{w}(x_i)\log\widehat{w}(x_i)|^\beta \right)^{1/\beta} \right\}\epsilon.$$

Proof. It is a simple application of Theorem A.4, using Assumption 4.4 and Lemma A.8. □

Finally, we need the following result to relate the KL–divergence estimated with and without the self–normalized estimator.

Lemma A.10. *For any $d \in \mathcal{D}_{d_{\pi,p}}$ and for any $d' \in \mathcal{D}_{\Pi,\mathcal{P}}$. The following inequality holds:*

$$\left| \widehat{D}_{KL}(d\|d_{\pi,p}) - \widetilde{D}_{KL}(d\|d_{\pi,p}) \right| \le \left| \left(\frac{1}{N}\sum_{i=1}^N \widehat{w}(x_i) \right) \log\left(\frac{1}{N}\sum_{i=1}^N \widehat{w}(x_i) \right) \right| + 2\log N \left| \frac{1}{N}\sum_{i=1}^N \widehat{w}(x_i) - 1 \right|.$$

Proof. We perform some algebraic manipulation of the expression:

$$\widehat{D}_{KL}(d\|d_{\pi,p}) - \widetilde{D}_{KL}(d\|d_{\pi,p}) = \frac{1}{N}\sum_{i=1}^{N}\widehat{w}(x_i)\log\widehat{w}(x_i) - \frac{1}{N}\sum_{i=1}^{N}\frac{\widehat{w}(x_i)N}{\sum_{i=1}^{N}\widehat{w}(x_i)}\log\frac{\widehat{w}(x_i)N}{\sum_{i=1}^{N}\widehat{w}(x_i)}$$

$$= \frac{1}{N}\sum_{i=1}^{N}\widehat{w}(x_i)\log\frac{\widehat{w}(x_i)N}{\sum_{i=1}^{N}\widehat{w}(x_i)} + \left(\frac{1}{N}\sum_{i=1}^{N}\widehat{w}(x_i)\right)\log\left(\frac{1}{N}\sum_{i=1}^{N}\widehat{w}(x_i)\right) - \frac{1}{N}\sum_{i=1}^{N}\frac{\widehat{w}(x_i)N}{\sum_{i=1}^{N}\widehat{w}(x_i)}\log\frac{\widehat{w}(x_i)N}{\sum_{i=1}^{N}\widehat{w}(x_i)}$$

$$= \frac{1}{N}\sum_{i=1}^{N}\frac{\widehat{w}(x_i)N}{\sum_{i=1}^{N}\widehat{w}(x_i)}\log\frac{\widehat{w}(x_i)N}{\sum_{i=1}^{N}\widehat{w}(x_i)}\left(\frac{1}{N}\sum_{i=1}^{N}\widehat{w}(x_i) - 1\right) + \left(\frac{1}{N}\sum_{i=1}^{N}\widehat{w}(x_i)\right)\log\left(\frac{1}{N}\sum_{i=1}^{N}\widehat{w}(x_i)\right).$$

Now, consider the term:

$$\frac{1}{N}\sum_{i=1}^{N}\frac{\widehat{w}(x_i)N}{\sum_{i=1}^{N}\widehat{w}(x_i)}\log\frac{\widehat{w}(x_i)N}{\sum_{i=1}^{N}\widehat{w}(x_i)} = \sum_{i=1}^{N}\widetilde{w}(x_i)\log\widetilde{w}(x_i) + \log N.$$

Since the $\widetilde{w}(x_i)$ sum up to 1, the summation $\sum_{i=1}^{N}\widetilde{w}(x_i)\log\widetilde{w}(x_i)$ is maximized in absolute value when all $\widehat{w}(x_i)$ are equal, thus $|\sum_{i=1}^{N}\widetilde{w}(x_i)\log\widetilde{w}(x_i)| \le \log N$. By taking the absolute value of the full expression, we get the result. $\square$

Now we can put all together.

Theorem 4.1. *(Finite–Sample Bound) Let $\pi \in \Pi$ and $p \in \mathcal{P}$ be the current policy and transition model. Let $\kappa > 0$ be the KL–divergence threshold. Let $d' \in \mathcal{D}_{d_{\pi,p}}$ be the solution of the PRIMAL$_\kappa$ problem and $d_{\widetilde{\pi}'',\widetilde{p}''} \in \mathcal{D}_{\Pi,\mathcal{P}}$ be the solution of the $\widetilde{\text{REMPS}}_\kappa$ problem with PROJ$_d$ computed with $N > 0$ samples collected with $d_{\pi,p}$. Then, under Assumptions 4.1, 4.3 and 4.4, for any $\alpha \in (1,\beta)$, there exist two constants χ, ξ and a function $\zeta(N) = \mathcal{O}(\log N)$ depending on α, and on the samples, such that for any $\delta \in (0,1)$, with probability at least $1-4\delta$ it holds that:*

$$J_{d'} - J_{\widetilde{\pi}'',\widetilde{p}''} \le \underbrace{\sqrt{2}r_{\max}\sup_{d\in\mathcal{D}_{d_{\pi,p}}}\inf_{\overline{d}\in\mathcal{D}_{\Pi,\mathcal{P}}}\sqrt{D_{\mathrm{KL}}(d\|\overline{d})}}_{\textit{approximation error}}$$

$$+ \underbrace{r_{\max}\chi\sqrt{\epsilon} + r_{\max}\zeta(N)\epsilon + r_{\max}\xi\epsilon^2}_{\textit{estimation error}},$$

where $\epsilon = 2^{\frac{\alpha+2}{2\alpha}}\sqrt{\frac{v\log\frac{2eN}{v}+\log\frac{8}{\delta}}{N^{\frac{2(\alpha-1)}{\alpha}}}}\Gamma\left(\alpha,\sqrt{\frac{v\log\frac{2eN}{v}+\log\frac{8}{\delta}}{N^{\frac{2(\alpha-1)}{\alpha}}}}\right)$, depending on the pseudo-dimension bound $v < +\infty$ and $\Gamma(\alpha,\tau) = \frac{\alpha-1}{\alpha} + \frac{1}{\alpha}\left(\frac{\alpha}{\alpha-1}\right)^{\alpha-1}\left(1+\left(\frac{\alpha-1}{\alpha}\right)^{\alpha-1}\log\frac{1}{\tau}\right)^{\frac{\alpha-1}{\alpha}}$.

Proof. We start from Theorem A.1 and we bound each term using Lemma A.7 and Lemma A.9. For brevity, we define $\epsilon = 2^{\frac{\alpha+2}{2\alpha}}\sqrt{\frac{v\log\frac{2eN}{v}+\log\frac{8}{\delta}}{N^{\frac{2(\alpha-1)}{\alpha}}}}\Gamma\left(\alpha,\sqrt{\frac{v\log\frac{2eN}{v}+\log\frac{8}{\delta}}{N^{\frac{2(\alpha-1)}{\alpha}}}}\right)$. Let us start with $\sup_{d\in\mathcal{D}_{d_{\pi,p}}}|J_d - \widetilde{J}_d|$:

$$\sup_{d\in\mathcal{D}_{d_{\pi,p}}}\left|J_d - \widetilde{J}_d\right| = \sup_{d\in\mathcal{D}_{d_{\pi,p}}}\left|J_d - \widetilde{J}_d \pm \widehat{J}_d\right|$$

$$\le \sup_{d\in\mathcal{D}_{d_{\pi,p}}}\left|J_d - \widehat{J}_d\right| + r_{\max}\sup_{d\in\mathcal{D}_{d_{\pi,p}}}\left|\frac{1}{N}\sum_{i=1}^{N}\widehat{w}(x_i) - 1\right|$$

$$\le 2r_{\max}\max\left\{\sqrt[\alpha]{d_\alpha(d\|d_{\pi,p})},\ \sqrt[\alpha]{\widehat{d}_\beta(d\|d_{\pi,p})}\right\}\epsilon,$$

where we exploited events $(\mathcal{E}_1)$ and $(\mathcal{E}_2)$ and simply observed that $\alpha < \beta$ and thus Lemma A.9 holds as well. Consider $\sup_{d\in\mathcal{D}_{d_{\pi,p}}}\left|\widetilde{D}_{KL}(d\|d_{\pi,p}) - D_{\mathrm{KL}}(d\|d_{\pi,p})\right|$:

$$\sup_{d\in\mathcal{D}_{d_{\pi,p}}}\left|\widetilde{D}_{KL}(d\|d_{\pi,p}) - D_{\mathrm{KL}}(d\|d_{\pi,p})\right| = \sup_{d\in\mathcal{D}_{d_{\pi,p}}}\left|\widetilde{D}_{KL}(d\|d_{\pi,p}) - D_{\mathrm{KL}}(d\|d_{\pi,p}) \pm \widehat{D}_{KL}(d\|d_{\pi,p})\right|$$

$$\le \sup_{d\in\mathcal{D}_{d_{\pi,p}}}\left|D_{KL}(d\|d_{\pi,p}) - \widehat{D}_{KL}(d\|d_{\pi,p})\right|$$

$$+ \left|\left(\frac{1}{N}\sum_{i=1}^{N}\widehat{w}(x_i)\right)\log\left(\frac{1}{N}\sum_{i=1}^{N}\widehat{w}(x_i)\right)\right| + 2\log N\left|\frac{1}{N}\sum_{i=1}^{N}\widehat{w}(x_i) - 1\right|.$$

To complete the derivation we have to analyze the term $z\log z$ with $z = \frac{1}{N}\sum_{i=1}^{N}\widehat{w}(x_i)$. Now using Lemma A.7 and defining $\tau = \max\left\{\sqrt[\alpha]{d_\alpha(d\|d_{\pi,p})},\ \sqrt[\alpha]{\widehat{d}_\beta(d\|d_{\pi,p})}\right\}\epsilon$ we know that $\max\{0, 1-\tau\} \le z \le 1+\tau$ as $z \ge 0$. Consider a value of $\tau \in [0,1]$ it is simple to prove that $(1+\tau)\log(1+\tau) \geq -(1-\tau)\log(1-\tau)$, therefore $|z \log z| \leq (1+\tau)\log(1+\tau)$. Therefore, we have:

$$\sup_{d \in \mathcal{D}_{d_{\pi,p}}} \left| \widetilde{D}_{KL}(d \| d_{\pi,p}) - D_{\mathrm{KL}}(d \| d_{\pi,p}) \right| \leq \max \left\{ \frac{1}{e} + \frac{\alpha}{\beta - \alpha} \mathop{\mathbb{E}}_{x \sim q} \left[\left| \frac{d(x)}{d_{\pi,p}(x)} \right|^{\beta} \right]^{1/\alpha}, \left(\frac{1}{N} \sum_{i=1}^{N} |\widehat{w}(x_i) \log \widehat{w}(x_i)|^{\beta} \right)^{1/\beta} \right\} \epsilon$$
$$+ \left(1 + \max \left\{ \sqrt[\alpha]{d_\alpha(d, d_{\pi,p})}, \sqrt[\alpha]{\widehat{d}_\alpha(d, d_{\pi,p})} \right\} \epsilon \right) \log \left(1 + \max \left\{ \sqrt[\alpha]{d_\alpha(d, d_{\pi,p})}, \sqrt[\alpha]{\widehat{d}_\alpha(d, d_{\pi,p})} \right\} \epsilon \right)$$
$$+ 2 \log N \max \left\{ \sqrt[\alpha]{d_\beta(d, d_{\pi,p})}, \sqrt[\alpha]{\widehat{d}_\alpha(d, d_{\pi,p})} \right\} \epsilon.$$

Finally, the term $\sup_{d \in \mathcal{D}^{d_{p,\pi}}} \sup_{d' \in \mathcal{D}_{\Pi,\mathcal{P}}} \left| \widehat{H}(d \| d') - H(d \| d') \right|$ can be bounded using Lemma A.7. We define:

$$f(\alpha) = \max \left\{ \frac{1}{e} + \frac{\alpha}{\beta - \alpha} \mathop{\mathbb{E}}_{x \sim q} \left[\left| \frac{d(x)}{d_{\pi,p}(x)} \right|^{\beta} \right]^{1/\alpha}, \left(\frac{1}{N} \sum_{i=1}^{N} |\widehat{w}(x_i) \log \widehat{w}(x_i)|^{\beta} \right)^{1/\beta}, \sqrt[\alpha]{d_\alpha(d, d_{\pi,p})}, \sqrt[\alpha]{\widehat{d}_\alpha(d, d_{\pi,p})}, \right.$$
$$\left. \mathop{\mathbb{E}}_{X \sim d_{\pi,p}} \left[\left| \frac{d(X)}{d_{\pi,p}(X)} \log d'(X) \right|^{\beta} \right]^{1/\beta}, \left(\frac{1}{N} \sum_{i=1}^{N} \left| \frac{d(x_i)}{d_{\pi,p}(x_i)} \log d'(x_i) \right|^{\beta} \right)^{1/\beta} \right\}.$$

Finally,

$$\begin{aligned}
J_{d'} - J_{\widetilde{\pi}'', \widetilde{p}''} &\leq 4 r_{\max} f(\alpha) \epsilon + \frac{2 r_{\max}}{\kappa} \left[f(\alpha)\epsilon + (1 + f(\alpha)\epsilon) \log(1 + f(\alpha)\epsilon) + 2 \log N f(\alpha) \epsilon \right] \\
&\quad + r_{\max} \sqrt{2} \sup_{d \in \mathcal{D}_{d_{\pi,p}}} \inf_{\overline{d} \in \mathcal{D}_{\Pi,\mathcal{P}}} \sqrt{D_{\mathrm{KL}}(d \| \overline{d})} + r_{\max} \sqrt{2 f(\alpha) \epsilon} \\
&\leq 4 r_{\max} f(\alpha)\epsilon + \frac{2 r_{\max}}{\kappa} \left(1 + 2\log N + f(\alpha)\epsilon\right) f(\alpha)\epsilon + r_{\max}\sqrt{2} \sup_{d \in \mathcal{D}_{d_{\pi,p}}} \inf_{\overline{d} \in \mathcal{D}_{\Pi,\mathcal{P}}} \sqrt{D_{\mathrm{KL}}(d \| \overline{d})} + r_{\max}\sqrt{2 f(\alpha)\epsilon} \\
&= \sqrt{2} r_{\max} \sup_{d \in \mathcal{D}_{d_{\pi,p}}} \inf_{\overline{d} \in \mathcal{D}_{\Pi,\mathcal{P}}} \sqrt{D_{\mathrm{KL}}(d \| \overline{d})} + r_{\max} \chi \sqrt{\epsilon} + r_{\max} \zeta(N) \epsilon + r_{\max} \xi \epsilon^2,
\end{aligned}$$

where we exploited the fact that $\log(1+x) \leq x$ and $\chi = \sqrt{2 f(\alpha)}$, $\zeta(N) = 4 + \frac{2}{\kappa}(1 + 2\log N) f(\alpha)$ and $\xi = \frac{2}{\kappa}$. Since we made a union bound over the events $(\mathcal{E}_1)$, $(\mathcal{E}_2)$, $(\mathcal{E}_3)$ and $(\mathcal{E}_4)$, the statement holds with probability $1 - 4\delta$. □

Remark The first set of assumptions allows to have a convergence rate of type $\widetilde{\mathcal{O}}(N^{-1/4})$, while the second allows for $\widetilde{\mathcal{O}}\left(N^{-\frac{2(\alpha-1)}{4\alpha}}\right)$. Note that $\frac{2(\alpha-1)}{4\alpha} \leq \frac{1}{4}$, but the assumption employed is significantly lighter.

B. Gradient Estimators for Parametric Configuration Learning

In this appendix, we provide the straightforward extensions of REINFORCE and G(PO)MDP gradient estimators that can be used to adapt policy gradient methods to the problem of learning parametric environment configurations. Let us start by recalling the P-Gradient Theorem, introduced in Metelli et al. (2018), which is the natural adaptation of the Policy Gradient Theorem of Sutton et al. (2000).

Theorem B.1 (P-Gradient Theorem, from (Metelli et al., 2018))**.** *Let $\mathcal{P}_\Omega$ be a class of parametric stochastic transition models differentiable in $\boldsymbol{\omega}$, let π be a policy. Then, the gradient of the expected return with respect to $\boldsymbol{\omega}$ is given by:*

$$\nabla_{\boldsymbol{\omega}} J_{\pi, p_{\boldsymbol{\omega}}} = \mathop{\mathbb{E}}_{(S,A,S') \sim d_{\pi, p_{\boldsymbol{\omega}}}} \left[\nabla_{\boldsymbol{\omega}} \log p_{\boldsymbol{\omega}}(S'|S,A) u_{\pi,p_{\boldsymbol{\omega}}}(S,A,S') \right],$$

where $u_{\pi,p_{\boldsymbol{\omega}}}(s,a,s') = r(s,a,s') + \gamma v_{\pi,p_{\boldsymbol{\omega}}}(s')$ is the state-action-next-state value function.

We can also simply derive the trajectory-based expression of the gradient w.r.t. the environment configuration parameters.

Proposition B.1. *Let $\mathcal{P}_\Omega$ be a class of parametric stochastic transition models differentiable in $\boldsymbol{\omega}$, let π be a policy. Then, the gradient of the expected return with respect to $\boldsymbol{\omega}$ is given by:*

$$\nabla_{\boldsymbol{\omega}} J_{\pi,p_{\boldsymbol{\omega}}} = \mathop{\mathbb{E}}_{\tau \sim \nu_{\pi,p_{\boldsymbol{\omega}}}} \left[\nabla_{\boldsymbol{\omega}} \log \nu_{\pi,p_{\boldsymbol{\omega}}}(\tau) R(\tau) \right] = \mathop{\mathbb{E}}_{\tau \sim \nu_{\pi,p_{\boldsymbol{\omega}}}} \left[\sum_{t=0}^{H-1} \nabla_{\boldsymbol{\omega}} \log p_{\boldsymbol{\omega}}(s_{\tau,t+1}|s_{\tau,t}, a_{\tau,t}) R(\tau) \right],$$

where $\nu_{\pi,p_{\boldsymbol{\omega}}}(\tau) = \mu(s_{\tau,0}) \prod_{t=0}^{H-1} \pi(a_{\tau,t}|s_{\tau,t}) p_{\boldsymbol{\omega}}(s_{\tau,t+1}|s_{\tau,t}, a_{\tau,t})$ is the trajectory density function and $R(\tau) = \sum_{t=0}^{H-1} \gamma^t r(s_{\tau,t}, a_{\tau,t}, s_{\tau,t+1})$ is the trajectory return.

Proof. Derives trivially from the linearity of the gradient and expectation and by applying the log-trick. □

We can now derive the REINFORCE and G(PO)MDP estimators for the gradient and the corresponding optimal baselines. We omit the derivations as they are analogous to the policy case and we denote with $\odot$ the element-wise product between vectors.

REINFORCE

$$\widehat{\nabla}_{\boldsymbol{\omega}}^{\mathrm{RF}} J_{\pi,p_{\boldsymbol{\omega}}} = \frac{1}{N}\sum_{i=1}^{N}\left(\sum_{t=0}^{H-1}\nabla_{\boldsymbol{\omega}}\log p_{\boldsymbol{\omega}}(s_{\tau_i,t+1}|s_{\tau_i,t},a_{\tau_i,t})\right)\odot\left(\sum_{t=0}^{H-1}\gamma^t r(s_{\tau_i,t},a_{\tau_i,t},s_{\tau_i,t+1})-\mathbf{b}_{\mathrm{RF}}\right)$$

$$\mathbf{b}_{\mathrm{RF}}^{*} = \frac{\mathbb{E}_{\tau\sim\nu_{\pi,p_{\boldsymbol{\omega}}}}\left[\left(\sum_{t=0}^{H-1}\nabla_{\boldsymbol{\omega}}\log p_{\boldsymbol{\omega}}(s_{\tau_i,t+1}|s_{\tau_i,t},a_{\tau_i,t})\right)^2 R(\tau)\right]}{\mathbb{E}_{\tau\sim\nu_{\pi,p_{\boldsymbol{\omega}}}}\left[\left(\sum_{t=0}^{H-1}\nabla_{\boldsymbol{\omega}}\log p_{\boldsymbol{\omega}}(s_{\tau_i,t+1}|s_{\tau_i,t},a_{\tau_i,t})\right)^2\right]}$$

G(PO)MDP

$$\widehat{\nabla}_{\boldsymbol{\omega}}^{\mathrm{G(PO)MDP}} J_{\pi,p_{\boldsymbol{\omega}}} = \frac{1}{N}\sum_{i=1}^{N}\sum_{t=0}^{H-1}\left(\sum_{l=0}^{t}\nabla_{\boldsymbol{\omega}}\log p_{\boldsymbol{\omega}}(s_{\tau_i,l+1}|s_{\tau_i,l},a_{\tau_i,l})\right)\odot\left(\gamma^t r(s_{\tau_i,t},a_{\tau_i,t},s_{\tau_i,t+1})-\mathbf{b}_{t,\mathrm{G(PO)MDP}}\right)$$

$$\mathbf{b}_{t,\mathrm{G(PO)MDP}}^{*} = \frac{\mathbb{E}_{\tau\sim\nu_{\pi,p_{\boldsymbol{\omega}}}}\left[\left(\sum_{l=0}^{t}\nabla_{\boldsymbol{\omega}}\log p_{\boldsymbol{\omega}}(s_{\tau_i,l+1}|s_{\tau_i,l},a_{\tau_i,l})\right)^2 \gamma^t r(s_{\tau,t},a_{\tau,t},s_{\tau,t+1})\right]}{\mathbb{E}_{\tau\sim\nu_{\pi,p_{\boldsymbol{\omega}}}}\left[\left(\sum_{l=0}^{t}\nabla_{\boldsymbol{\omega}}\log p_{\boldsymbol{\omega}}(s_{\tau_i,l+1}|s_{\tau_i,l},a_{\tau_i,l})\right)^2\right]}$$

C. Implementation Details

In this appendix, we discuss some practical issues about our implementation of REMPS.

C.1. Dual Regularization

The parameter η in the solution of the PRIMAL$_\kappa$ controls the greediness of the stationary distribution d'. A small η corresponds to a very greedy distribution since the reward of a triplet (s,a,s') is weighted by $1/\eta$, an high η makes the new distribution very similar to the sampling one. We employ a regularization on the dual adding two penalization terms to prevent η from assuming too extreme values:

$$\min_{\eta\in[0,\infty)} g(\eta) = \eta\log\underset{S,A,S'\sim d}{\mathbb{E}}\left[\exp\left(\frac{1}{\eta}r(S,A,S')+\kappa\right)\right]+\lambda\left(\eta+\frac{1}{\eta}\right), \tag{38}$$

where $\lambda \geq 0$ controls the magnitude of the regularization.

C.2. Policy Regularization

Additionally, we sometimes employ an L^2 regularization of the policy parameters in the projection phase:

$$\boldsymbol{\theta}',\boldsymbol{\omega}' = \operatorname*{arg\,min}_{\boldsymbol{\theta}\in\Theta,\boldsymbol{\omega}\in\Omega} D_{\mathrm{KL}}\left(d'\|d_{\pi_{\boldsymbol{\theta}},p_{\boldsymbol{\omega}}}\right)+\beta\left\|\boldsymbol{\theta}\right\|_2^2,$$

where $\beta \geq 0$ controls the magnitude of the policy regularization. The same regularization term can be applied to all types of projections. Notice that we do not apply regularization on model parameters.

D. Experimental Details

In this appendix, we report the description of the environments used in the experimental evaluation, the value of the hyperparameters employed and some additional experiments we did not include in the main paper.

D.1. Chain Domain

D.1.1. Hyperparameters

In Table 2 we report the hyperparameters used in the experiments on the Chain domain.

Table 2. Hyper-parameters used in the experiments on the Chain domain.

Parameter	Value
ζ	0.2
L	10
l	8
s	2
ω_0	0.8
θ_0	0.2
λ	0
β	0
Number of samples per iteration	500

D.1.2. Comparison of projection strategies

In Figure 6, we compare the different projection strategies together with the no-configuration cases. We can see that the best learning curve is attained by the PROJ_{p^π} that reaches the global optimum quickly. REMPS with $\text{PROJ}_{\pi,p}$ is unable to reach the global optimum, indeed the configuration parameter gets stuck to a suboptimal value (around 0.55), thus the performance is significantly worse w.r.t. PROJ_{p^π}. The same behavior, limited to the configuration parameter value, is displayed by the only-configuration (REMS, Relative Entropy Model Search) learning case. Finally, the only-policy (REPS, Relative Entropy Policy Search) learning moves the policy parameter towards zero, approaching the local optimum.

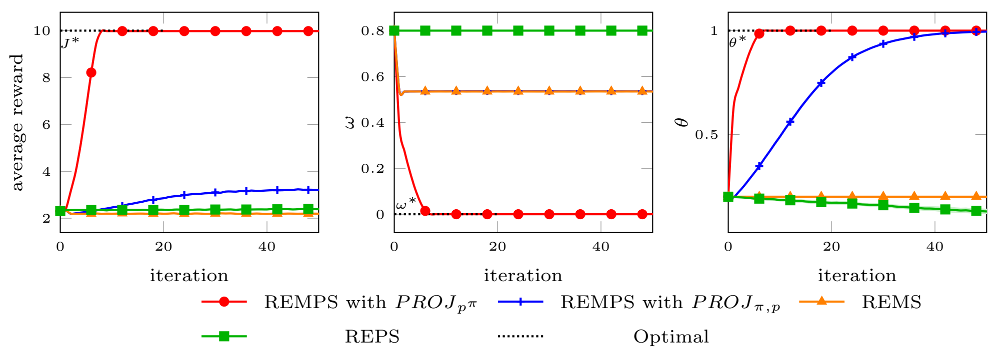

Figure 6. Average reward, configuration parameter (ω) and policy parameter (θ) in the Chain domain with different projection strategies, only-policy (REPS) and only-configuration (REMS) learning as a function of the number of iterations. 20 runs 95% c.i.

D.1.3. Effect of the policy and model spaces

The optimization phase (PRIMAL_κ) in REMPS is able to find in closed-form a new stationary distribution d' that optimizes our performance index subject to a trust-region constraint. As we have seen, this distribution is not typically representable in space $\mathcal{D}_{\Pi,\mathcal{P}}$ and, thus, we need to perform a projection. We analyze how the finite representation power of $\mathcal{D}_{\Pi,\mathcal{P}}$ affects performance. Figure 7 shows the performance of the best model-policy found as a function of κ and the value of PRIMAL_κ which is the expected return obtained by evaluating d' after solving the primal. We can see that the value of the primal is always larger than the performance after the projection, i.e., the performance of the new policy-configuration pair. As expected, the projection yields a degradation of performance.

D.1.4. Sensitivity to parameter initialization

REMPS behaves consistently with respect to a random initialization of model and policy parameters. In Figure 8, we can see that REMPS updates the model and policy parameters towards the global maximum while G(PO)MDP updates vary across the different initializations. In the G(PO)MDP learning curve it is possible to see clearly the two attractors.

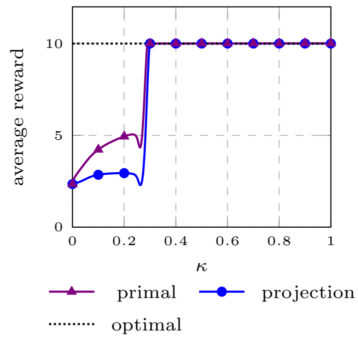

Figure 7. Average reward after PRIMAL$_\kappa$ (primal) and after PROJ$_d$ (projection) compared with the optimal performance, as a function of the KL-threshold κ.

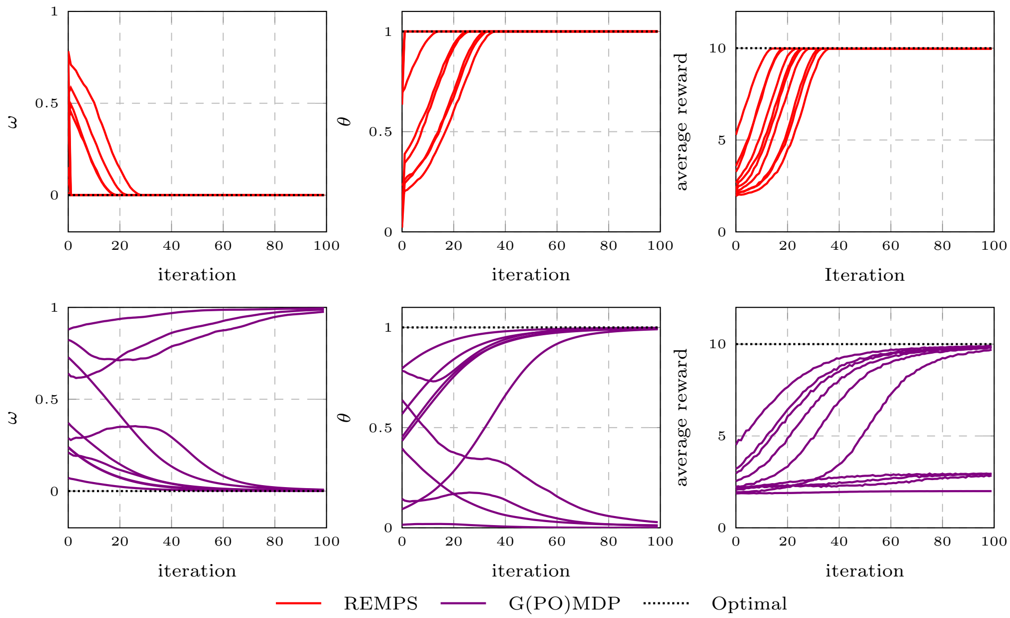

Figure 8. Configuration parameter (ω), policy parameter (θ) and average return in the Chain domain with random initialization of model and policy parameter. Comparison between G(PO)MDP and REMPS.

D.1.5. COMPARISON WITH SPMI

SPMI (Metelli et al., 2018) is, so far, the only algorithm proposed for Conf-MDPs. We report in the following the learning curves of SPMI. In Figure 9, we show the behavior of the SPMI variants on the chain experiment. We can easily notice that SPMI requires a huge number of iterations before convergence. While REMPS converges approximately after 10 iterations, SPMI requires a number of iterations in the order of 10^3. This is due to the conservative step size of safe approaches. SPMI, SPMI-alt, SPMI-sup and SPI-SMI reach the global maximum while SMI-SPI goes (very slowly) towards the local maximum. SMI-SPI is not able to reach the global maximum since it alternates a model improvement step with a policy improvement step, considering the two components in a separate way. We recall that SPMI is applicable to the chain experiment since this environment has a discrete state space and a discrete action space, while the standard version of this algorithm cannot be applied to the other domains considered in this paper (Cartpole and TORCS), having them a continuous state space.

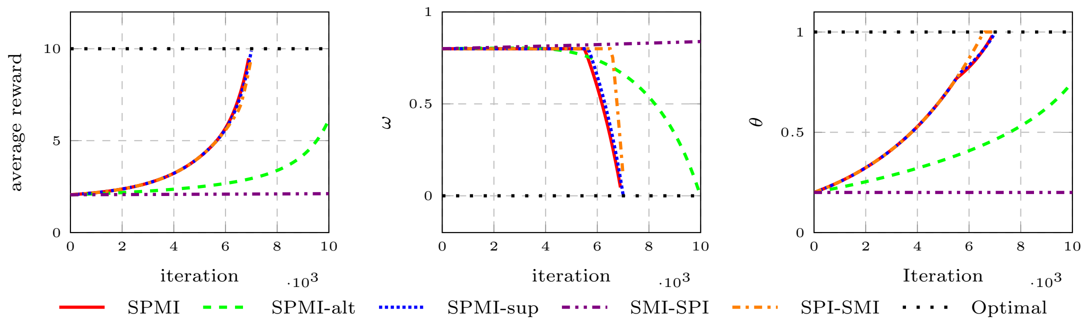

Figure 9. Average return, configuration parameter (ω) and policy parameter (θ) in the Chain domain comparing the different versions of SPMI.

D.2. Cartpole

D.2.1. Environment description

The Cartpole domain (Widrow & Smith, 1964; Barto et al., 1983) is a standard RL benchmark. The environment consists of a cart that moves along the horizontal axis and a pole that is anchored on the cart. The state space is continuous and is represented by the position of the cart x, by the cart velocity $\dot{x}$, by the pole angle γ with respect to the vertical, and by the pole angular velocity $\dot{\gamma}$. The action space is discrete and consists of two actions: left L and right R. The model parameter is represented by the force ω to be applied to the cart, which is the same for both actions, thus the resulting force is $\pm\omega$ based on the action. The parameter space is $\Omega = [0, 30]$. Each action, when performed, is affected by a noise term proportional to the applied force and independent for each state component. The goal is to keep the pole in a vertical position ($\gamma = 0$) as long as possible. The episode ends when the pole reaches a certain angle ($|\gamma| > \bar{\gamma}$) or after a predefined number of steps. We want to encourage smaller forces, to this end we use the following reward function:

$$r(s, a, s') = 10 - \frac{\omega^2}{20} - 20 \cdot (1 - \cos(\gamma)).$$

The first part of the reward function is a fixed bonus for each time step the pole is up and the pole angle is within the range $[-\bar{\gamma}, \bar{\gamma}]$. The second part of the reward is a penalty proportional to the force. The third part is a penalty proportional to the pole angle. Ideally the agent should learn to balance the pole with the smallest force possible, keeping it fixed in a vertical position.

D.2.2. Hyperparameters

In Table 3 we report the hyper-parameters used in the Cartpole experiments.

Table 3. Hyper-parameters used in the exact and approximate Cartpole experiment.

Parameter	Exact case	Approximate case
Number of samples per iteration	100000	50000
λ	0	0
β	0	10^{-4}
κ	10^{-3}	10^{-3}
ω_0	8	8

D.2.3. Policy and Model approximator

We evaluate the performance of our algorithm in the exact case (known model) and in the approximate case. In the exact case, we know the effect of the model parameters on the transition function, i.e., we know $p_\omega(\cdot|s, a)$. The policy $\pi_{\boldsymbol{\theta}}$ is softmax policy with a linear mapping in the state space $\mathbf{s} = (x, \dot{x}, \gamma, \dot{\gamma}, 1)$:

$$\pi_{\boldsymbol{\theta}}(a|s) = \frac{e^{\boldsymbol{\theta}_a^T \mathbf{s}}}{\sum_{a' \in \{L,R\}} e^{\boldsymbol{\theta}_{a'}^T \mathbf{s}}}, \quad a \in \{L, R\}.$$

For the approximate case, we assume the distribution over the next states can be approximated by a Gaussian distribution with diagonal covariance. We model the mean and the variance using two independent neural networks with the same input $(\mathbf{s}, a, \omega)$:

$$\begin{aligned}
&\widehat{p}(\cdot|\mathbf{s}, a, \omega) \sim \mathcal{N}\left(\mu(\mathbf{s}, a, \omega), \sigma^2(\mathbf{s}, a, \omega)\right), \\
&\mu(\mathbf{s}, a, \omega) = NN_\mu(\mathbf{s}, a, \omega), \\
&\sigma(\mathbf{s}, a, \omega) = \exp\left(NN_\sigma(\mathbf{s}, a, \omega)\right),
\end{aligned}$$

where NN_μ and NN_σ have the same architecture, i.e., one hidden layer made of 10 neurons with tanh activation. The training is performed just once at the beginning of training, using a dataset made of 10^5 samples collected with different configuration parameters ω (randomly generated).

D.2.4. Additional Results

In Figure 10, we show additional details of the experiments, reported in the main paper, both for the exact and approximate case.

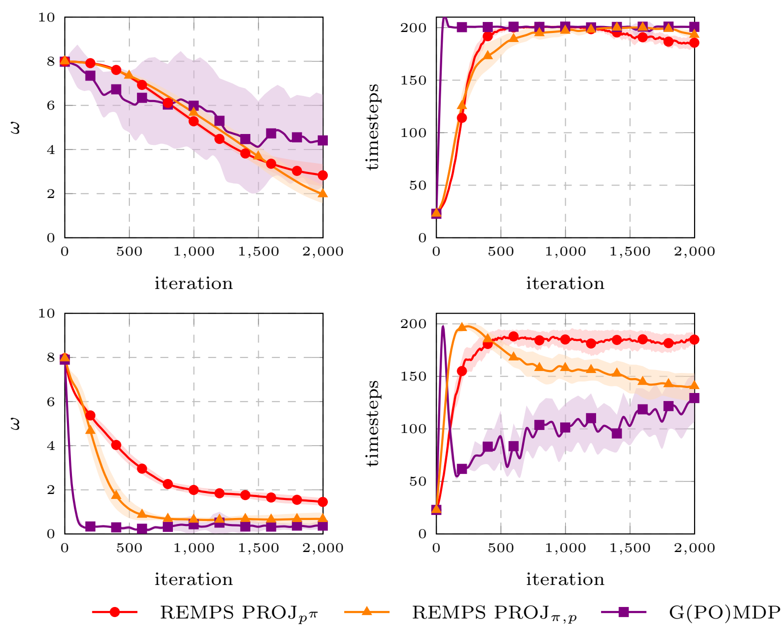

Figure 10. Configuration parameter (ω) and episode duration for the cartpole experiment comparing REMPS with PROJ$_{p^\pi}$, REMPS with PROJ$_{\pi,p}$ and G(PO)MDP. Top: ideal scenario. Bottom: approximated scenario. 20 runs, 95% c.i.

Configuring the environment could be more expensive than modifying the policy (e.g. one should stop a factory and change the machine configuration). In Figure 11, we show empirically the effect of performing an *alternated projection* PROJ$_{\text{alt}}$ on the performance. In this experiment we perform policy optimization every iteration, while the model optimization is performed only every 50 iterations, using the projection PROJ$_{p^\pi}$. In Figure 11, we show the comparison between the alternated projection and the other types of projections. We can notice that, although the is update of the configuration is performed rarely, we are able to reach pretty fast a good performance, at least comparable with PROJ$_{\pi,p}$.

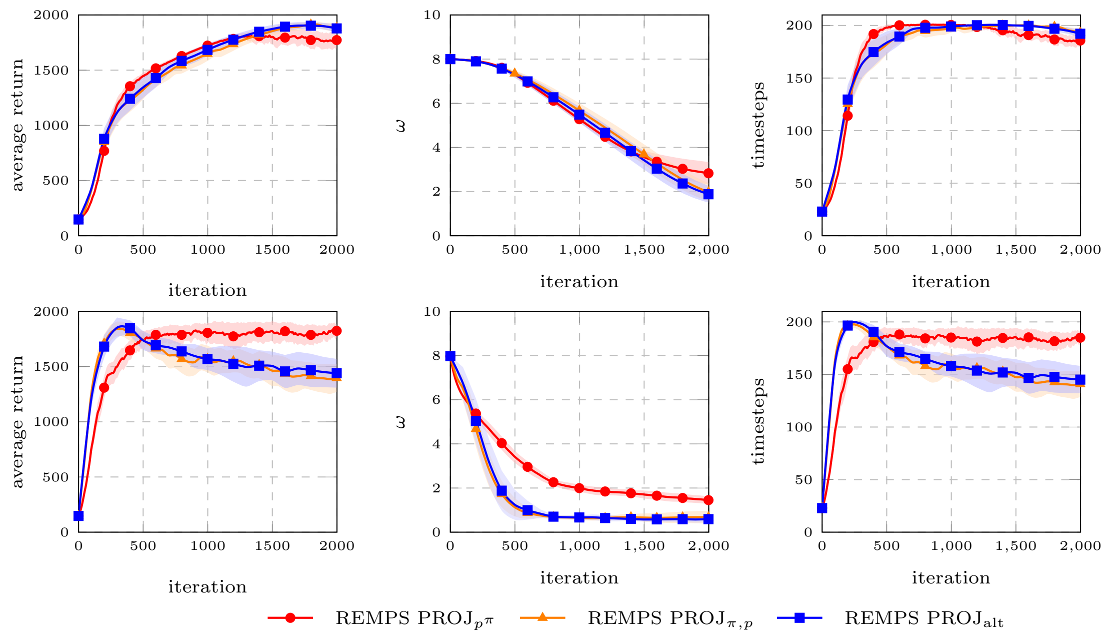

Figure 11. Average return, Configuration parameter (ω), episode duration for the cartpole experiment comparing REMPS PROJ$_{p^\pi}$, REMPS PROJ$_{\pi,p}$ and REMPS PROJ$_{\text{alt}}$. Top: ideal scenario. Bottom: approximated scenario. 20 runs, 95% c.i.

D.3. TORCS

D.3.1. Environment description

The state space of the TORCS environment is composed by 29 dimensions, $\mathcal{S} \subseteq \mathbb{R}^{29}$. The action space is composed by 2 dimensions, $\mathcal{A} \subseteq \mathbb{R}^2$: acceleration/brake action, where $+1$ indicates full acceleration and -1 full brake and steering angle, where -1 indicates maximum left steer and $+1$ maximum right steer. Among all possible parameters (Table 5), in our experiments we focused on configuring the Rear and Front Wings and the Front-Rare Brake Repartition. All configuration parameters are normalized in the range $[0, 1]$. The state space space is summarized in Table 4 and the configuration parameters in Table 5.

Table 4. State space of the TORCS experiment.

Parameter	Description
α	Angle between the car direction and the direction of the track axis.
rpm	Number of rotation per minute of the car engine.
v_x	Speed of the car along the longitudinal axis of the car.
v_y	Speed of the car along the transverse axis of the car.
v_z	Speed of the car along the Z axis of the car.
track	Vector of 19 range finder sensors: each sensors returns the distance between the track edge and the car within a range of 200 meters.
trackPos	Distance between the car and the track axis.
wheelSpinVel	Vector of 4 sensors representing the rotation speed of wheels.

We defined the reward function in the following way:

$$r(s, a, s') = v'_x \cdot \cos(\alpha'), \tag{39}$$

where v'_x is the velocity on the longitudinal direction of the car in state s' and α' is the angle between the car direction and the direction of the track axis. We give a penalty of -1000 if the agent runs backward, if it goes out of track or if the progress in the race is too small. The rationale behind this reward is to encourage the agent to go at high speed and to stay centered with respect to the track.

Table 5. Configuration space of the TORCS experiment. Underlined the parameters we configure in the experiment.

Parameter	Description
Rear Wing	Angle of the rear wing.
Front Wing	Angle of the front wing.
Front-Rear Brake Repartition	Repartition of the brake between the front and rear.
Front Anti-Roll Bar	Front Spring.
Rear Anti-Roll Bar	Rear Spring.
Front Left-Right Brake	Brake disk diameter of the front wheels.
Rear Left-Right Brake	Brake disk diameter of the rear wheels.

D.3.2. HYPERPARAMETERS

In Table 6 we report the hyper-parameters used in the TORCS experiments.

Table 6. Hyper-parameters used in the TORCS experiment.

Parameter	Value
Number of samples per iteration	20000
λ	0.1
β	0
κ	10^{-5}
ω_0	$(1,1,1)$

D.3.3. POLICY AND MODEL APPROXIMATORS

Policy approximator The policy we used in the TORCS experiments is a Gaussian Policy parameterized by a fully connected neural network:

$$\pi(\cdot|\boldsymbol{s}) \sim \mathcal{N}(\boldsymbol{\mu}(\boldsymbol{s}), \operatorname{diag}(\boldsymbol{\sigma}^2)),$$
$$\boldsymbol{\mu}(\boldsymbol{s}) = NN_\mu(\boldsymbol{s}),$$
$$\boldsymbol{\sigma} = \exp(\boldsymbol{\nu}),$$

where NN_μ and has one hidden layer with 64 neurons with tanh activations. The activation of the last layer of NN_μ is tanh since actions are limited in $[-1,1]$. The covariance matrix is diagonal and independent of the state. We initialize the policy fitting, via maximum likelihood, a scripted policy (snakeoil) using 45000 samples collected with 30 randomly generated values of the configurable parameters.

Model approximator We considered a Gaussian model to approximate the dynamics of the task:

$$\widehat{p}(\cdot|\mathbf{s},\mathbf{a},\boldsymbol{\omega}) \sim \mathcal{N}\left(\mu(\mathbf{s},\mathbf{a},\boldsymbol{\omega}), \operatorname{diag}(\boldsymbol{\sigma}^2)\right),$$
$$\mu(\mathbf{s},\mathbf{a},\boldsymbol{\omega}) = NN_\mu(\mathbf{s},\mathbf{a},\boldsymbol{\omega}),$$
$$\boldsymbol{\sigma} = \exp(\boldsymbol{\eta}).$$

The mean network is composed of two hidden layers of 64 neurons each with tanh activation. The covariance matrix is diagonal and independent of the state, action and configurable parameters. The model fitting is performed at the beginning of learning using the same samples employed for fitting the policy.

D.3.4. ADDITIONAL EXPERIMENTS

In Figure 12, we show the average speed as a function of the number of iterations and the value of the configurable parameters. First, we observe that, when configuring the environment is possible, the car reaches higher speeds. Second, we can see that all parameters tend to be moved towards zero. Indeed, a good behavior in the considered track consists in increasing the speed as much as possible. Therefore, the orientation of the wing tends to be reduced to increase the speed. A similar behavior is visible for the Front-Rear Brake Repartition.

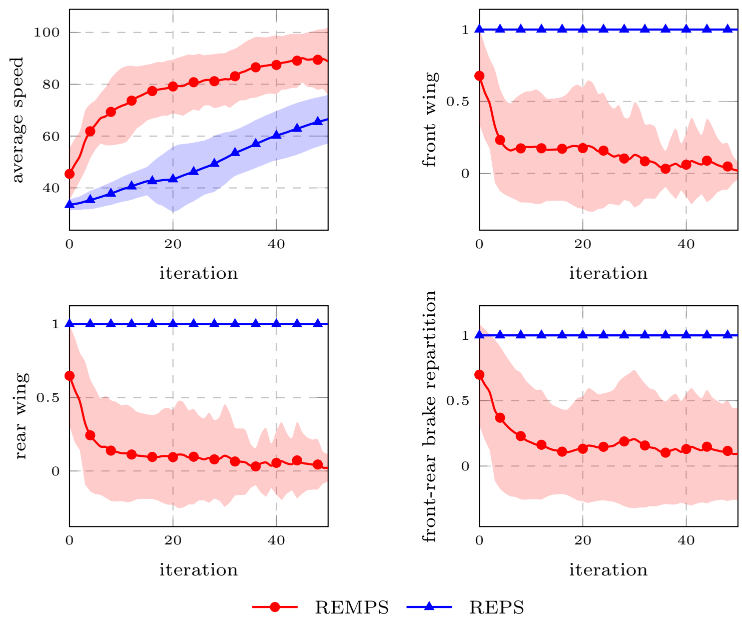

Figure 12. Average speed, configurable parameters values and episode duration as a function of the number of iterations for the TORCS experiment comparing REMPS and REPS. 10 runs, 80% c.i.