Learning Optimal Deterministic Policies with Stochastic Policy Gradients

Alessandro Montenegro [1] **Marco Mussi** [1] **Alberto Maria Metelli** [1] **Matteo Papini** [1]

Abstract

Policy gradient (PG) methods are successful approaches to deal with continuous *reinforcement learning* (RL) problems. They learn stochastic parametric (hyper)policies by either exploring in the space of actions or in the space of parameters. Stochastic controllers, however, are often undesirable from a practical perspective because of their lack of robustness, safety, and traceability. In common practice, stochastic (hyper)policies are learned only to deploy their deterministic version. In this paper, we make a step towards the theoretical understanding of this practice. After introducing a novel framework for modeling this scenario, we study the global convergence to the *best deterministic* policy, under (weak) gradient domination assumptions. Then, we illustrate how to tune the exploration level used for learning to optimize the trade-off between the sample complexity and the performance of the deployed deterministic policy. Finally, we quantitatively compare action-based and parameter-based exploration, giving a formal guise to intuitive results.

1. Introduction

Within *reinforcement learning* (RL, Sutton & Barto, 2018) approaches, *policy gradient* (PG, Deisenroth et al., 2013) algorithms have proved very effective in dealing with real-world control problems. Their advantages include the applicability to continuous state and action spaces (Peters & Schaal, 2006), resilience to sensor and actuator noise (Gravell et al., 2020), robustness to partial observability (Azizzadenesheli et al., 2018), and the possibility of incorporating prior knowledge in the policy design phase (Ghavamzadeh & Engel, 2006), improving explainability (Likmeta et al., 2020). PG algorithms search directly in the space of parametric policies for the one that maximizes a performance function. Nonetheless, as always in RL, the *exploration* problem has to be addressed, and practical methods involve injecting noise in the actions or in the parameters. This limits the application of PG methods in many real-world scenarios, such as autonomous driving, industrial plants, and robotic controllers. Indeed, stochastic policies typically do not meet the *reliability*, *safety*, and *traceability* standards of this kind of applications.

The problem of learning *deterministic* policies has been explicitly addressed in the PG literature by Silver et al. (2014) with their *deterministic policy gradient*, which spawned very successful deep RL algorithms (Lillicrap et al., 2016; Fujimoto et al., 2018). This approach, however, is affected by several drawbacks, mostly due to its inherent *off-policy* nature. First, this makes DPG hard to analyze from a theoretical perspective: local convergence guarantees have been established only recently, and only under assumptions that are very demanding for deterministic policies (Xiong et al., 2022). Furthermore, its practical versions (DDPG, Lillicrap et al., 2016) are known to be very susceptible to hyperparameter tuning.

We study here a simpler and fairly common approach: that of learning *stochastic* policies with PG algorithms, then *deploying* the corresponding deterministic version, "switching off" the noise.[1] Intuitively, the amount of exploration (e.g., the variance of a Gaussian policy) should be selected wisely. Indeed, the smaller the exploration level, the closer the optimized objective is to that of a deterministic policy. At the same time, with a small exploration, learning can severely slow down and get stuck on bad local optima.

Policy gradient methods can be partitioned based on the space on which the exploration is carried out, distinguishing between: *action-based* (AB) and *parameter-based* (PB, Sehnke et al., 2010) exploration. The first, of which REINFORCE (Williams, 1992) and GPOMDP (Baxter & Bartlett, 2001; Sutton et al., 1999) are the progenitor algorithms, performs exploration in the action space, with a stochastic (e.g., Gaussian) *policy*. On the other hand, PB exploration, introduced by Parameter-Exploring Policy Gradients (PGPE, Sehnke et al., 2010), implements the exploration at the level of policy parameters by means of a stochastic *hyperpolicy*.

[1]Politecnico di Milano, Piazza Leonardo da Vinci 32, 20133, Milan, Italy. Correspondence to: Alessandro Montenegro <alessandro.montenegro@polimi.it>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

[1]This can be observed in several libraries (e.g., Raffin et al., 2021) and benchmarks (e.g., Duan et al., 2016).

The latter performs perturbations of the parameters of a (typically deterministic) action policy. Of course, this dualism only considers the simplest form of noise-based, *undirected* exploration. Efficient exploration in large-scale MDPs is a very active area of research, with a large gap between theory and practice (Ghavamzadeh et al., 2020), placing the matter well beyond the scope of this paper. Also, we consider noise magnitudes that are *fixed* during the learning process, as the common practice of *learning* the exploration parameters themselves breaks all known sample complexity guarantees of vanilla PG (see Appendix C).

To this day, a large effort has been put into providing convergence guarantees and sample complexity analyses for AB exploration algorithms (e.g., Papini et al., 2018; Yuan et al., 2022; Fatkhullin et al., 2023), while the theoretical analysis of PB exploration has been taking a back seat since (Zhao et al., 2011). We are not aware of any *global* convergence results for parameter-based PGs. Furthermore, even for AB exploration, current studies focus on the *convergence to the best stochastic policy*.

Original Contributions. In this paper, we make a step towards the theoretical understanding of the practice of *deploying* a deterministic policy learned with PG methods:

- We introduce a framework for modeling the practice of *deploying* a deterministic policy, by formalizing the notion of *white noise-based exploration*, allowing for a unified treatment of both AB and PB exploration.
- We study the *convergence to the best deterministic* policy for both AB and PB exploration. For this reason, we focus on the *global convergence*, rather than on the first-order stationary point (FOSP) convergence, and we leverage on commonly used *(weak) gradient domination* assumptions.
- We quantitatively show how the exploration level (i.e., noise) generates a trade-off between the sample complexity and the performance of the deployed deterministic policy. Then, we illustrate how it can be tuned to optimize such a trade-off, delivering sample complexity guarantees.

In light of these results, we compare the advantages and disadvantages of AB and PB exploration in terms of sample complexity and requested assumptions, giving a formal guise to intuitive results. We also elaborate on how the assumptions used in the convergence analysis can be reconnected to the basic characteristics of the MDP and the policy classes. We conclude with a numerical validation to empirically illustrate the discussed trade-offs. The proofs of the results presented in the main paper are reported in Appendix D.

2. Preliminaries

Notation. For a measurable set $\mathcal{X}$, we denote with $\Delta(\mathcal{X})$ the set of probability measures over $\mathcal{X}$. For $P \in \Delta(\mathcal{X})$, we denote with p its density function. With a little abuse of notation, we will interchangeably use $x \sim P$ or $x \sim p$ to denote that random variable x is sampled from the P. For $n \in \mathbb{N}$, we denote by $[\![n]\!] := \{1, \dots, n\}$.

Lipschitz Continuous and Smooth Functions. A function $f : \mathcal{X} \subseteq \mathbb{R}^d \to \mathbb{R}$ is *L-Lipschitz continuous* (L-LC) if $|f(\mathbf{x}) - f(\mathbf{x}')| \leqslant L\|\mathbf{x} - \mathbf{x}'\|_2$ for every $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$. f is *L_2-Lipschitz smooth* (L_2-LS) if it is continuously differentiable and its gradient $\nabla_{\mathbf{x}} f$ is L_2-LC, i.e., $\|\nabla_{\mathbf{x}} f(\mathbf{x}) - \nabla_{\mathbf{x}} f(\mathbf{x}')\|_2 \leqslant L_2\|\mathbf{x} - \mathbf{x}'\|_2$ for every $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$.

Markov Decision Processes. A Markov Decision Process (MDP, Puterman, 1990) is represented by $\mathcal{M} := (\mathcal{S}, \mathcal{A}, p, r, \rho_0, \gamma)$, where $\mathcal{S} \subseteq \mathbb{R}^{d_{\mathcal{S}}}$ and $\mathcal{A} \subseteq \mathbb{R}^{d_{\mathcal{A}}}$ are the measurable state and action spaces, $p : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ is the transition model, where $p(\mathbf{s}'|\mathbf{s}, \mathbf{a})$ specifies the probability density of landing in state $\mathbf{s}' \in \mathcal{S}$ by playing action $\mathbf{a} \in \mathcal{A}$ in state $\mathbf{s} \in \mathcal{S}$, $r : \mathcal{S} \times \mathcal{A} \to [-R_{\max}, R_{\max}]$ is the reward function, where $r(\mathbf{s}, \mathbf{a})$ specifies the reward the agent gets by playing action $\mathbf{a}$ in state $\mathbf{s}$, $\rho_0 \in \Delta(\mathcal{S})$ is the initial-state distribution, and $\gamma \in [0, 1]$ is the discount factor. A trajectory $\tau = (\mathbf{s}_{\tau,0}, \mathbf{a}_{\tau,0}, \dots, \mathbf{s}_{\tau,T-1}, \mathbf{a}_{\tau,T-1})$ of length $T \in \mathbb{N} \cup \{+\infty\}$ is a sequence of T state-action pairs. The *discounted return* of a trajectory τ is $R(\tau) := \sum_{t=0}^{T-1} \gamma^t r(\mathbf{s}_{\tau,t}, \mathbf{a}_{\tau,t})$.

Deterministic Parametric Policies. We consider a *parametric deterministic policy* $\boldsymbol{\mu}_{\boldsymbol{\theta}} : \mathcal{S} \to \mathcal{A}$, where $\boldsymbol{\theta} \in \Theta \subseteq \mathbb{R}^{d_\Theta}$ is the parameter vector belonging to the parameter space Θ. The performance of $\boldsymbol{\mu}_{\boldsymbol{\theta}}$ is assessed via the *expected return* $J_{\mathrm{D}} : \Theta \to \mathbb{R}$, defined as:

$$J_{\mathrm{D}}(\boldsymbol{\theta}) := \mathop{\mathbb{E}}_{\tau \sim p_{\mathrm{D}}(\cdot|\boldsymbol{\theta})} [R(\tau)], \tag{1}$$

where $p_{\mathrm{D}}(\tau; \boldsymbol{\theta}) := \rho_0(\mathbf{s}_{\tau,0}) \prod_{t=0}^{T-1} p(\mathbf{s}_{\tau,t+1}|\mathbf{s}_{\tau,t}, \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s}_{\tau,t}))$ is the density of trajectory τ induced by policy $\boldsymbol{\mu}_{\boldsymbol{\theta}}$.[2] The agent's goal consists of finding an optimal parameter $\boldsymbol{\theta}^*_{\mathrm{D}} \in \arg\max_{\boldsymbol{\theta} \in \Theta} J_{\mathrm{D}}(\boldsymbol{\theta})$ and we denote $J^*_{\mathrm{D}} := J_{\mathrm{D}}(\boldsymbol{\theta}^*_{\mathrm{D}})$.

Action-Based (AB) Exploration. In AB exploration, we consider a *parametric stochastic policy* $\pi_{\boldsymbol{\rho}} : \mathcal{S} \to \Delta(\mathcal{A})$, where $\boldsymbol{\rho} \in \mathcal{P}$ is the parameter vector belonging to the parameter space $\mathcal{P} \subseteq \mathbb{R}^{d_{\mathcal{P}}}$. The policy is used to sample actions $\mathbf{a}_t \sim \pi_{\boldsymbol{\rho}}(\cdot|\mathbf{s}_t)$ to be played in state $\mathbf{s}_t$ for *every step* t of interaction. The performance of $\pi_{\boldsymbol{\rho}}$ is assessed via the *expected return* $J_{\mathrm{A}} : \mathcal{P} \to \mathbb{R}$, defined as:

$$J_{\mathrm{A}}(\boldsymbol{\rho}) := \mathop{\mathbb{E}}_{\tau \sim p_{\mathrm{A}}(\cdot|\boldsymbol{\rho})} [R(\tau)], \qquad \text{where} \tag{2}$$

$p_{\mathrm{A}}(\tau; \boldsymbol{\rho}) := \rho_0(\mathbf{s}_{\tau,0}) \prod_{t=0}^{T-1} \pi_{\boldsymbol{\rho}}(\mathbf{a}_{\tau,t}|\mathbf{s}_{\tau,t}) p(\mathbf{s}_{\tau,t+1}|\mathbf{s}_{\tau,t}, \mathbf{a}_{\tau,t})$ is the density of trajectory τ induced by policy $\pi_{\boldsymbol{\rho}}$.[2] In AB exploration, we aim at learning $\boldsymbol{\rho}^*_{\mathrm{A}} \in \arg\max_{\boldsymbol{\rho} \in \mathcal{P}} J_{\mathrm{A}}(\boldsymbol{\rho})$ and we denote $J_{\mathrm{A}}{}^* := J_{\mathrm{A}}(\boldsymbol{\rho}^*_{\mathrm{A}})$. If $J_{\mathrm{A}}(\boldsymbol{\rho})$ is differentiable

[2] For both J_{D} (resp. J_{A}, J_{P}) and p_{D} (resp. p_{A}, p_{P}), we use the D (resp. A, P) subscript to denote that the dependence on $\boldsymbol{\theta}$ (resp. $\boldsymbol{\rho}$) is through a Deterministic policy (resp. Action-based exploration policy, Parameter-based exploration hyperpolicy).

w.r.t. $\boldsymbol{\rho}$, PG methods (Peters & Schaal, 2008) update the parameter $\boldsymbol{\rho}$ via gradient ascent: $\boldsymbol{\rho}_{t+1} \leftarrow \boldsymbol{\rho}_t + \zeta_t \widehat{\nabla}_{\boldsymbol{\rho}} J_{\mathrm{A}}(\boldsymbol{\rho}_t)$, where $\zeta_t > 0$ is the *step size* and $\widehat{\nabla}_{\boldsymbol{\rho}} J_{\mathrm{A}}(\boldsymbol{\rho})$ is an estimator of $\nabla_{\boldsymbol{\rho}} J_{\mathrm{A}}(\boldsymbol{\rho})$. In particular, the GPOMDP *estimator* is:[3]

$$\widehat{\nabla}_{\boldsymbol{\rho}} J_{\mathrm{A}}(\boldsymbol{\rho}) := \frac{1}{N} \sum_{i=1}^{N} \sum_{t=0}^{T-1} \left(\sum_{k=0}^{t} \nabla_{\boldsymbol{\rho}} \log \pi_{\boldsymbol{\rho}}(\mathbf{a}_{\tau_i,k} | \mathbf{s}_{\tau_i,k}) \right) \gamma^t r(\mathbf{s}_{\tau_i,t}, \mathbf{a}_{\tau_i,t}),$$

where N is the number of independent trajectories $\{\tau_i\}_{i=1}^{N}$ collected with policy $\pi_{\boldsymbol{\rho}}$ ($\tau_i \sim p_{\mathrm{A}}(\cdot; \boldsymbol{\rho})$), called *batch size*.

Parameter-Based (PB) Exploration. In PB exploration, we use a *parametric stochastic hyperpolicy* $\nu_{\boldsymbol{\rho}} \subseteq \Delta(\Theta)$, where $\boldsymbol{\rho} \in \mathbb{R}^{d_{\mathcal{P}}}$ is the parameter vector. The hyperpolicy is used to sample parameters $\boldsymbol{\theta} \sim \nu_{\boldsymbol{\rho}}$ to be plugged in the deterministic policy $\boldsymbol{\mu}_{\boldsymbol{\theta}}$ at the beginning of *every trajectory*. The performance index of $\nu_{\boldsymbol{\rho}}$ is $J_{\mathrm{P}} : \mathbb{R}^{d_{\boldsymbol{\rho}}} \to \mathbb{R}$, that is the expectation over $\boldsymbol{\theta}$ of $J_{\mathrm{D}}(\boldsymbol{\theta})$ defined as:[2]

$$J_{\mathrm{P}}(\boldsymbol{\rho}) := \mathop{\mathbb{E}}_{\boldsymbol{\theta} \sim \nu_{\boldsymbol{\rho}}} \left[J_{\mathrm{D}}(\boldsymbol{\theta}) \right].$$

PB exploration aims at learning $\boldsymbol{\rho}_{\mathrm{P}}^* \in \arg\max_{\boldsymbol{\rho} \in \mathcal{P}} J_{\mathrm{P}}(\boldsymbol{\rho})$ and we denote $J_{\mathrm{P}}^* := J_{\mathrm{P}}(\boldsymbol{\rho}_{\mathrm{P}}^*)$. If $J_{\mathrm{D}}(\boldsymbol{\rho})$ is differentiable w.r.t. $\boldsymbol{\rho}$, PGPE (Sehnke et al., 2010) updates the hyperparameter $\boldsymbol{\rho}$ via gradient accent: $\boldsymbol{\rho}_{t+1} \leftarrow \boldsymbol{\rho}_t + \zeta_t \widehat{\nabla}_{\boldsymbol{\rho}} J_{\mathrm{P}}(\boldsymbol{\rho}_t)$. In particular, PGPE uses an estimator of $\nabla_{\boldsymbol{\rho}} J_{\mathrm{P}}(\boldsymbol{\rho})$ defined as:

$$\widehat{\nabla}_{\boldsymbol{\rho}} J_{\mathrm{P}}(\boldsymbol{\rho}) = \frac{1}{N} \sum_{i=1}^{N} \nabla_{\boldsymbol{\rho}} \log \nu_{\boldsymbol{\rho}}(\boldsymbol{\theta}_i) R(\tau_i),$$

where N is the number of independent parameters-trajectories pairs $\{(\boldsymbol{\theta}_i, \tau_i)\}_{i=1}^{N}$, collected with hyperpolicy $\nu_{\boldsymbol{\rho}}$ ($\boldsymbol{\theta}_i \sim \nu_{\boldsymbol{\rho}}$ and $\tau_i \sim p_{\mathrm{D}}(\cdot; \boldsymbol{\theta}_i)$), called *batch size*.

3. White-Noise Exploration

We formalize a class of stochastic (hyper)policies widely employed in the practice of AB and PB exploration, namely *white noise-based (hyper)policies*. These policies $\pi_{\boldsymbol{\theta}}(\cdot|s)$ (resp. hyperpolicies $\nu_{\boldsymbol{\theta}}$) are obtained by adding a *white noise* $\boldsymbol{\epsilon}$ to the deterministic action $\mathbf{a} = \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s})$ (resp. to the parameter $\boldsymbol{\theta}$) independent of the state s (resp. parameter $\boldsymbol{\theta}$).

Definition 3.1 (White Noise)**.** *Let $d \in \mathbb{N}$ and $\sigma > 0$. A probability distribution $\Phi_d \in \Delta(\mathbb{R}^d)$ is a white-noise if:*

$$\mathop{\mathbb{E}}_{\boldsymbol{\epsilon} \sim \Phi_d} [\boldsymbol{\epsilon}] = \mathbf{0}_d, \qquad \mathop{\mathbb{E}}_{\boldsymbol{\epsilon} \sim \Phi_d} [\|\boldsymbol{\epsilon}\|_2^2] \leqslant d\sigma^2. \tag{3}$$

This definition complies with the zero-mean Gaussian distribution $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}_d, \boldsymbol{\Sigma})$, where $\mathbb{E}_{\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}_d, \boldsymbol{\Sigma})}[\|\boldsymbol{\epsilon}\|_2^2] = \mathrm{tr}(\boldsymbol{\Sigma}) \leqslant d\lambda_{\max}(\boldsymbol{\Sigma})$. In particular, for an isotropic Gaussian $\boldsymbol{\Sigma} = \sigma^2 \mathbf{I}_d$, we have that $\mathrm{tr}(\boldsymbol{\Sigma}) = d\sigma^2$. We now formalize the notion of *white noise-based (hyper)policy*.

Definition 3.2 (White noise-based policies)**.** *Let $\boldsymbol{\theta} \in \Theta$ and $\boldsymbol{\mu}_{\boldsymbol{\theta}} : \mathcal{S} \to \mathcal{A}$ be a parametric deterministic policy and let $\Phi_{d_{\mathcal{A}}}$ be a white noise (Definition 3.1). A* white noise-based policy $\pi_{\boldsymbol{\theta}} : \mathcal{S} \to \Delta(\mathcal{A})$ *is such that, for every state $\mathbf{s} \in \mathcal{S}$, action $\mathbf{a} \sim \pi_{\boldsymbol{\theta}}(\cdot|\mathbf{s})$ satisfies $\mathbf{a} = \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s}) + \boldsymbol{\epsilon}$ where $\boldsymbol{\epsilon} \sim \Phi_{d_{\mathcal{A}}}$ independently at every step.*

This definition considers stochastic policies $\pi_{\boldsymbol{\theta}}(\cdot|\mathbf{s})$ that are obtained by adding noise $\boldsymbol{\epsilon}$ fulfilling Definition 3.1, *sampled independently at every step*, to the action $\boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s})$ prescribed by the deterministic policy (i.e., AB exploration), resulting in playing action $\boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s}) + \boldsymbol{\epsilon}$. An analogous definition can be formulated for hyperpolicies.

Definition 3.3 (White noise-based hyperpolicies)**.** *Let $\boldsymbol{\theta} \in \Theta$ and $\boldsymbol{\mu}_{\boldsymbol{\theta}} : \mathcal{S} \to \mathcal{A}$ be a parametric deterministic policy and let Φ_{d_Θ} be a white-noise (Definition 3.1). A* white noise-based hyperpolicy $\nu_{\boldsymbol{\theta}} \in \Delta(\Theta)$ *is such that, for every parameter $\boldsymbol{\theta} \in \Theta$, parameter $\boldsymbol{\theta}' \sim \nu_{\boldsymbol{\theta}}$ satisfies $\boldsymbol{\theta}' = \boldsymbol{\theta} + \boldsymbol{\epsilon}$ where $\boldsymbol{\epsilon} \sim \Phi_{d_\Theta}$ independently in every trajectory.*

This definition considers stochastic hyperpolicies $\nu_{\boldsymbol{\theta}}$ obtained by adding noise $\boldsymbol{\epsilon}$ fulfilling Definition 3.1, *sampled independently at the beginning of each trajectory*, to the parameter $\boldsymbol{\theta}$ defining the deterministic policy $\boldsymbol{\mu}_{\boldsymbol{\theta}}$, resulting in playing deterministic policy $\boldsymbol{\mu}_{\boldsymbol{\theta}+\boldsymbol{\epsilon}}$ (i.e., PB exploration). Definitions 3.2 and 3.3 allow to represent a class of widely-used (hyper)policies, like Gaussian hyperpolicies and Gaussian policies with state-independent variance. Furthermore, once the parameter $\boldsymbol{\theta}$ is learned with either AB or PB exploration, *deploying* the corresponding deterministic policy (i.e., "switching off" the noise) is straightforward.[4] Finally, we remark that the noise can exhibit an inner structure, while it is required to be "white" among different realizations.

4. Fundamental Assumptions

In this section, we present the *fundamental* assumptions on the MDP (p and r), deterministic policy $\boldsymbol{\mu}_{\boldsymbol{\theta}}$, and white noise Φ. For the sake of generality, we will consider *abstract* assumptions in the next sections and, then, show their relation to the fundamental ones (see Appendix A for details).

Assumptions on the MDP. We start with the assumptions on the regularity of the MDP, i.e., on transition model p and reward function r, w.r.t. variations of the played action $\mathbf{a}$.

Assumption 4.1 (Lipschitz MDP ($\log p$, r) w.r.t. actions)**.** *The log transition model $\log p(\mathbf{s}'|\mathbf{s}, \cdot)$ and the reward function $r(\mathbf{s}, \cdot)$ are L_p-LC and L_r-LC, respectively, w.r.t. the action for every $\mathbf{s}, \mathbf{s}' \in \mathcal{S}$, i.e., for every $\mathbf{a}, \bar{\mathbf{a}} \in \mathcal{A}$:*

$$|\log p(\mathbf{s}'|\mathbf{s}, \mathbf{a}) - \log p(\mathbf{s}'|\mathbf{s}, \bar{\mathbf{a}})| \leqslant L_p \|\mathbf{a} - \bar{\mathbf{a}}\|_2, \tag{4}$$

$$|r(\mathbf{s}, \mathbf{a}) - r(\mathbf{s}, \bar{\mathbf{a}})| \leqslant L_r \|\mathbf{a} - \bar{\mathbf{a}}\|_2. \tag{5}$$

Assumption 4.2 (Smooth MDP ($\log p$, r) w.r.t. actions)**.**

[3] We limit our analysis to the GPOMDP estimator (Baxter & Bartlett, 2001), neglecting the REINFORCE one (Williams, 1992) since it is known that the latter suffers from larger variance.

[4] For white noise-based (hyper)policies there exists a *one-to-one mapping* between the parameter space of (hyper)policies and that of deterministic policies ($\mathcal{P} = \Theta$). For simplicity, we assume $\Theta = \mathbb{R}^{d_\Theta}$ and $\mathcal{A} = \mathbb{R}^{d_{\mathcal{A}}}$ (see Appendix C).

The log transition model $\log p(\mathbf{s}'|\mathbf{s},\cdot)$ *and the reward function* $r(\mathbf{s},\cdot)$ *are* $L_{2,p}$-LS *and* $L_{2,r}$-LS*, respectively, w.r.t. the action for every* $\mathbf{s},\mathbf{s}'\in\mathcal{S}$*, i.e., for every* $\mathbf{a},\overline{\mathbf{a}}\in\mathcal{A}$*:*

$$\|\nabla_{\mathbf{a}}\log p(\mathbf{s}'|\mathbf{s},\mathbf{a})-\nabla_{\mathbf{a}}\log p(\mathbf{s}'|\mathbf{s},\overline{\mathbf{a}})\|_2\leqslant L_{2,p}\|\mathbf{a}-\overline{\mathbf{a}}\|_2,$$
$$\|\nabla_{\mathbf{a}}r(\mathbf{s},\mathbf{a})-\nabla_{\mathbf{a}}r(\mathbf{s},\overline{\mathbf{a}})\|_2\leqslant L_{2,r}\|\mathbf{a}-\overline{\mathbf{a}}\|_2.$$

Intuitively, these assumptions ensure that when we perform AB and/or PB exploration altering the played action w.r.t. a deterministic policy, the effect on the environment dynamics and on reward (and on their gradients) is controllable.

Assumptions on the deterministic policy. We now move to the assumptions on the regularity of the deterministic policy $\boldsymbol{\mu}_{\boldsymbol{\theta}}$ w.r.t. the parameter $\boldsymbol{\theta}$.

Assumption 4.3 (Lipschitz deterministic policy $\boldsymbol{\mu}_{\boldsymbol{\theta}}$ w.r.t. parameters $\boldsymbol{\theta}$)**.** *The deterministic policy* $\boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s})$ *is* L_μ-LC *w.r.t. parameter for every* $\mathbf{s}\in\mathcal{S}$*, i.e., for every* $\boldsymbol{\theta},\overline{\boldsymbol{\theta}}\in\Theta$*:*

$$\|\boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s})-\boldsymbol{\mu}_{\overline{\boldsymbol{\theta}}}(\mathbf{s})\|_2\leqslant L_\mu\|\boldsymbol{\theta}-\overline{\boldsymbol{\theta}}\|_2. \tag{6}$$

Assumption 4.4 (Smooth deterministic policy $\boldsymbol{\mu}_{\boldsymbol{\theta}}$ w.r.t. parameters $\boldsymbol{\theta}$)**.** *The deterministic policy* $\boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s})$ *is* $L_{2,\mu}$-LS *w.r.t. parameter for every* $\mathbf{s}\in\mathcal{S}$*, i.e., for every* $\boldsymbol{\theta},\overline{\boldsymbol{\theta}}\in\Theta$*:*

$$\|\nabla_{\boldsymbol{\theta}}\boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s})-\nabla_{\boldsymbol{\theta}}\boldsymbol{\mu}_{\overline{\boldsymbol{\theta}}}(\mathbf{s})\|_2\leqslant L_{2,\mu}\|\boldsymbol{\theta}-\overline{\boldsymbol{\theta}}\|_2. \tag{7}$$

Similarly, these assumptions ensure that if we deploy an altered parameter $\boldsymbol{\theta}$, like in PB exploration, the effect on the played action (and on its gradient) is bounded.

Assumptions 4.1 and 4.3 are standard in the DPG literature (Silver et al., 2014). Assumption 4.2, instead, can be interpreted as the counterpart of the *Q-function smoothness* used in the DPG analysis (Kumar et al., 2020; Xiong et al., 2022), while Assumption 4.4 has been used to study the convergence of DPG (Xiong et al., 2022). Similar conditions to our Assumption 4.1 were adopted by Pirotta et al. (2015), but measuring the continuity of p in the Kantorovich metric, a weaker requirement that, unfortunately, does not come with a corresponding smoothness condition.

Assumptions on the (hyper)policies. We introduce the assumptions on the score functions of the white noise Φ.

Assumption 4.5 (Bounded Scores of Φ)**.** *Let* $\Phi\in\Delta(\mathbb{R}^d)$ *be a white noise with variance bound* $\sigma>0$ *(Definition 3.1) and density* ϕ*.* ϕ *is differentiable in its argument and there exists a universal constant* $c>0$ *such that:*

(i) $\mathbb{E}_{\boldsymbol{\epsilon}\sim\Phi}[\|\nabla_{\boldsymbol{\epsilon}}\log\phi(\boldsymbol{\epsilon})\|_2^2]\leqslant cd\sigma^{-2}$*;*
(ii) $\mathbb{E}_{\boldsymbol{\epsilon}\sim\Phi}[\|\nabla^2_{\boldsymbol{\epsilon}}\log\phi(\boldsymbol{\epsilon})\|_2]\leqslant c\sigma^{-2}$*.*

Intuitively, this assumption is equivalent to the more common ones requiring the boundedness of the expected norms of the score function and its gradient (Papini et al., 2022; Yuan et al., 2022, see Appendix E). Note that a zero-mean Gaussian $\Phi=\mathcal{N}(\mathbf{0}_d,\boldsymbol{\Sigma})$ fulfills Assumption 4.5. Indeed, one has $\nabla_{\boldsymbol{\epsilon}}\log\phi(\boldsymbol{\epsilon})=\boldsymbol{\Sigma}^{-1}\boldsymbol{\epsilon}$ and $\nabla^2_{\boldsymbol{\epsilon}}\log\phi(\boldsymbol{\epsilon})=\boldsymbol{\Sigma}^{-1}$. Thus, $\mathbb{E}[\|\nabla_{\boldsymbol{\epsilon}}\log\phi(\boldsymbol{\epsilon})\|_2^2]=\mathrm{tr}(\boldsymbol{\Sigma}^{-1})\leqslant d\lambda_{\min}(\boldsymbol{\Sigma})^{-1}$ and $\mathbb{E}[\|\nabla^2_{\boldsymbol{\epsilon}}\log\phi(\boldsymbol{\epsilon})\|_2]=\lambda_{\min}(\boldsymbol{\Sigma})^{-1}$. In particular, for an isotropic Gaussian $\boldsymbol{\Sigma}=\sigma^2\mathbf{I}$, we have $\lambda_{\min}(\boldsymbol{\Sigma})=\sigma^2$, fulfilling Assumption 4.5 with $c=1$.

5. Deploying Deterministic Policies

In this section, we study the performance J_{D} of the *deterministic* policy $\boldsymbol{\mu}_{\boldsymbol{\theta}}$, when the parameter $\boldsymbol{\theta}$ is learned via AB or PB white noise-based exploration (Section 3). We will refer to this scenario as *deploying* the parameters, which reflects the common practice of "switching off the noise" once the learning process is over.

PB Exploration. Let us start with PB exploration by observing that for white noise-based hyperpolicies (Definition 3.3), we can express the expected return J_{P} as a function of J_{D} and of the noise $\boldsymbol{\epsilon}$ for every $\boldsymbol{\theta}\in\Theta$:

$$J_{\mathrm{P}}(\boldsymbol{\theta})=\mathop{\mathbb{E}}_{\boldsymbol{\epsilon}\sim\Phi_{d_\Theta}}[J_{\mathrm{D}}(\boldsymbol{\theta}+\boldsymbol{\epsilon})]. \tag{8}$$

This illustrates that PB exploration can be obtained by *perturbing the parameter* $\boldsymbol{\theta}$ of a deterministic policy $\boldsymbol{\mu}_{\boldsymbol{\theta}}$ via the noise $\boldsymbol{\epsilon}\sim\Phi_{d_\Theta}$. To achieve guarantees on the deterministic performance J_{D} of a parameter $\boldsymbol{\theta}$ learned with PB exploration, we enforce the following regularity condition.

Assumption 5.1 (Lipschitz J_{D} w.r.t. $\boldsymbol{\theta}$)**.** *J_D is L_J-LC in the parameter $\boldsymbol{\theta}$, i.e., for every $\boldsymbol{\theta},\boldsymbol{\theta}'\in\Theta$:*

$$|J_D(\boldsymbol{\theta})-J_D(\boldsymbol{\theta}')|\leqslant L_J\|\boldsymbol{\theta}-\boldsymbol{\theta}'\|_2. \tag{9}$$

When the MDP and the deterministic policy are LC as in Assumptions 4.1 and 4.3, L_J is $O((1-\gamma)^{-2})$ (see Table 2 in Appendix A for the full expression). This way, we guarantee that the perturbation $\boldsymbol{\epsilon}$ on the parameter $\boldsymbol{\theta}$ determines a variation on function J_{D} depending on the magnitude of $\boldsymbol{\epsilon}$, which allows obtaining the following result.

Theorem 5.1 (**Deterministic deployment of parameters learned with PB white-noise exploration**)**.** *If the hyperpolicy complies with Definition 3.3, under Assumption 5.1:*

(i) (Uniform bound) *for every* $\boldsymbol{\theta}\in\Theta$*, it holds that* $|J_D(\boldsymbol{\theta})-J_P(\boldsymbol{\theta})|\leqslant L_J\sqrt{d_\Theta}\sigma_P$*;*
(ii) (J_{D} upper bound) *let* $\boldsymbol{\theta}_P^*\in\arg\max_{\boldsymbol{\theta}\in\Theta}J_P(\boldsymbol{\theta})$*, it holds that:* $J_D^*-J_D(\boldsymbol{\theta}_P^*)\leqslant 2L_J\sqrt{d_\Theta}\sigma_P$*;*
(iii) (J_{D} lower bound) *there exists an MDP, a deterministic policy class $\boldsymbol{\mu}_{\boldsymbol{\theta}}$ fulfilling Assumption 5.1, and a noise complying with Definition 3.1, such that* $J_D^*-J_D(\boldsymbol{\theta}_P^*)\geqslant 0.28L_J\sqrt{d_\Theta}\sigma_P$*.*

Some observations are in order. *(i)* shows that the performance of the hyperpolicy $J_{\mathrm{P}}(\boldsymbol{\theta})$ is representative of the deterministic performance $J_{\mathrm{D}}(\boldsymbol{\theta})$ up to an additive term depending on $L_J\sqrt{d_\Theta}\sigma_{\mathrm{P}}$. As expected, this term grows with the Lipschitz constant L_J of the function J_{D}, with the standard deviation σ_{P} of the additive noise, and with the dimensionality of the parameter space d_Θ. In particular, this implies that $\lim_{\sigma_{\mathrm{P}}\to 0^+}J_{\mathrm{P}}(\boldsymbol{\theta})=J_{\mathrm{D}}(\boldsymbol{\theta})$. *(ii)* is a consequence

of (i) and provides an *upper bound* between the optimal performance obtained if we were able to directly optimize the deterministic policy $\max_{\boldsymbol{\theta}\in\Theta} J_{\mathrm{D}}(\boldsymbol{\theta})$ and the performance of the parameter $\boldsymbol{\theta}^*_{\mathrm{P}}$ learned by optimizing $J_{\mathrm{P}}(\boldsymbol{\theta})$, i.e., via PB exploration, *when deployed on the deterministic policy*. Finally, (iii) provides a lower bound to the same quantity on a specific instance of MDP and hyperpolicy, proving that the dependence on $L_J\sqrt{d_\Theta}\sigma_{\mathrm{P}}$ is *tight* up to constant terms.

AB Exploration. Let us move to the AB exploration case, where understanding the effect of the noise is more complex since it is applied to every action *independently at every step*. To this end, we introduce the notion of *non-stationary* deterministic policy $\underline{\boldsymbol{\mu}} = (\boldsymbol{\mu}_t)_{t=0}^{T-1}$, where at time step t the deterministic policy $\boldsymbol{\mu}_t : \mathcal{S} \to \mathcal{A}$ is played, and its expected return (with abuse of notation) is $J_{\mathrm{D}}(\underline{\boldsymbol{\mu}}) = \mathbb{E}_{\tau\sim p_{\mathrm{D}}(\cdot|\underline{\boldsymbol{\mu}})}[R(\tau)]$ where $p_{\mathrm{D}}(\cdot|\underline{\boldsymbol{\mu}}) := \rho_0(\mathbf{s}_{\tau,0})\prod_{t=0}^{T-1} p(\mathbf{s}_{\tau,t+1}|\mathbf{s}_{\tau,t},\boldsymbol{\mu}_t(\mathbf{s}_{\tau,t}))$. Let $\underline{\boldsymbol{\epsilon}} = (\boldsymbol{\epsilon}_t)_{t=0}^{T-1} \sim \Phi_{d_{\mathcal{A}}}^T$ be a sequence of noises sampled independently, we denote with $\underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}} + \underline{\boldsymbol{\epsilon}} = (\boldsymbol{\mu}_{\boldsymbol{\theta}} + \boldsymbol{\epsilon}_t)_{t=0}^{T-1}$ the non-stationary policy that, at time t, *perturbs the action* as $\boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s}_t) + \boldsymbol{\epsilon}_t$. Since the noise is independent on the state, we express J_{A} as a function of J_{D} for every $\boldsymbol{\theta}\in\Theta$ as follows:

$$J_{\mathrm{A}}(\boldsymbol{\theta}) = \mathop{\mathbb{E}}_{\underline{\boldsymbol{\epsilon}}\sim\Phi_{d_{\mathcal{A}}}^T}\left[J_{\mathrm{D}}(\underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}} + \underline{\boldsymbol{\epsilon}})\right]. \tag{10}$$

Thus, to ensure that the parameter learned with AB exploration achieves performance guarantees when evaluated as a deterministic policy, we need to enforce some regularity condition on J_{D} as a function of $\underline{\boldsymbol{\mu}}$.

Assumption 5.2 (Lipschitz J_{D} w.r.t. $\underline{\boldsymbol{\mu}}$)**.** *$J_D$ of the non-stationary deterministic policy $\underline{\boldsymbol{\mu}}$ is $(L_t)_{t=0}^{T-1}$-*LC *in the non-stationary policy, i.e., for every $\underline{\boldsymbol{\mu}}, \underline{\boldsymbol{\mu}}'$:*

$$|J_D(\underline{\boldsymbol{\mu}}) - J_D(\underline{\boldsymbol{\mu}}')| \leqslant \sum_{t=0}^{T-1} L_t \sup_{\mathbf{s}\in\mathcal{S}} \left\|\boldsymbol{\mu}_t(\mathbf{s}) - \boldsymbol{\mu}'_t(\mathbf{s})\right\|_2. \tag{11}$$

Furthermore, we denote $L := \sum_{t=0}^{T-1} L_t$.

When the MDP is LC as in Assumptions 4.1, L is $O((1-\gamma)^{-2})$ (see Table 2 in Appendix A for the full expression). The assumption enforces that changing the deterministic policy at step t from $\boldsymbol{\mu}_t$ to $\boldsymbol{\mu}'_t$, the variation of J_{D} is controlled by the action distance (in the worst state $\mathbf{s}$) multiplied by a *time-dependent* Lipschitz constant. This form of condition allows us to show the following result.

Theorem 5.2 (**Deterministic deployment of parameters learned with AB white-noise exploration**)**.** *If the policy complies with Definition 3.2 and under Assumption 5.2:*

- *(i)* (Uniform bound) *for every $\boldsymbol{\theta}\in\Theta$, it holds that: $|J_D(\boldsymbol{\theta}) - J_A(\boldsymbol{\theta})| \leqslant L\sqrt{d_{\mathcal{A}}}\sigma_A$;*
- *(ii)* (J_{D} upper bound) *letting $\boldsymbol{\theta}^*_A \in \arg\max_{\boldsymbol{\theta}\in\Theta} J_A(\boldsymbol{\theta})$, it holds that $J^*_D - J_D(\boldsymbol{\theta}^*_A) \leqslant 2L\sqrt{d_{\mathcal{A}}}\sigma_A$;*
- *(iii)* (J_{D} lower bound) *there exists an MDP, a deterministic policy class $\boldsymbol{\mu}_{\boldsymbol{\theta}}$ fulfilling Assumption 5.1, and a noise complying with Definition 3.1, such that $J^*_D - J_D(\boldsymbol{\theta}^*_A) \geqslant 0.28L\sqrt{d_{\mathcal{A}}}\sigma_A$.*

Similarly to Theorem 5.1, (i) and (ii) provide an upper bound on the difference between the policy performance $J_{\mathrm{A}}(\boldsymbol{\theta})$ and the corresponding deterministic policy $J_{\mathrm{D}}(\boldsymbol{\theta})$, and on the performance of $\boldsymbol{\theta}^*_{\mathrm{A}}$ when deployed on a deterministic policy. Clearly, also in the AB exploration, we have that $\lim_{\sigma_{\mathrm{A}}\to 0^+} J_{\mathrm{A}}(\boldsymbol{\theta}) = J_{\mathrm{D}}(\boldsymbol{\theta})$. As in the PB case, (iii) shows that the upper bound (ii) is tight up to constant terms.

Finally, let us note that our bounds for PB exploration depend on the dimension of the parameter space d_Θ that is replaced by that of the action space $d_{\mathcal{A}}$ in AB exploration.

6. Global Convergence Analysis

In this section, we present our main results about the convergence of AB and PB *white noise-based exploration* to a *global optimal parameter* $\boldsymbol{\theta}^*_{\mathrm{D}}$ for the performance of the deterministic policy J_{D}. Let $K\in\mathbb{N}$ be the number of *iterations* and N the *batch size*; given an accuracy threshold $\epsilon > 0$, our goal is to bound the *sample complexity* NK to fulfill the following *last-iterate global* convergence condition:

$$J^*_{\mathrm{D}} - \mathbb{E}\left[J_{\mathrm{D}}(\boldsymbol{\theta}_K)\right] \leqslant \epsilon, \tag{12}$$

where $\boldsymbol{\theta}_K$ is the (hyper)parameter at the end of learning. We start in Section 6.1, introducing the *abstract* assumptions and providing a general convergence analysis applicable to both AB and PB exploration for learning the corresponding objective (J_{A} or J_{P}). Then, in Section 6.2, we derive the convergence guarantees on the deterministic objective J_{D} for AB and PB exploration, respectively. Our results are first presented for a *fixed* white noise variance σ^2 to highlight the trade-off between sample complexity and performance, then extended to an ϵ-*adaptive* choice of σ.

6.1. General Global Convergence Analysis

In this section, we provide a global convergence analysis for a generic stochastic first-order algorithm optimizing the differentiable objective function $J_\dagger$ on the parameters space $\Theta\subseteq\mathbb{R}^d$, that can be instanced for both AB (setting $J_\dagger = J_{\mathrm{A}}$) and PB (setting $J_\dagger = J_{\mathrm{P}}$) exploration, when optimizing the corresponding objective. At every iteration $k\in[\![K]\!]$, the algorithm performs the gradient ascent update:

$$\boldsymbol{\theta}_{k+1} \leftarrow \boldsymbol{\theta}_k + \zeta_k \widehat{\nabla}_{\boldsymbol{\theta}} J_\dagger(\boldsymbol{\theta}_k), \tag{13}$$

where $\zeta_k > 0$ is the *step size* and $\widehat{\nabla}_{\boldsymbol{\theta}} J_\dagger(\boldsymbol{\theta}_k)$ is an *unbiased* estimate of $\nabla_{\boldsymbol{\theta}} J_\dagger(\boldsymbol{\theta}_k)$. We denote $J^*_\dagger = \max_{\boldsymbol{\theta}\in\Theta} J_\dagger(\boldsymbol{\theta})$ and we enforce the following standard assumptions.

Assumption 6.1 (Weak gradient domination for $J_\dagger$)**.** *There exist $\alpha > 0$ and $\beta \geqslant 0$ such that for every $\boldsymbol{\theta}\in\Theta$ it holds that $J^*_\dagger - J_\dagger(\boldsymbol{\theta}) \leqslant \alpha\|\nabla_{\boldsymbol{\theta}} J_\dagger(\boldsymbol{\theta})\|_2 + \beta$.*

Assumption 6.1 is the gold standard for the global conver-

gence of stochastic optimization (Yuan et al., 2022; Masiha et al., 2022; Fatkhullin et al., 2023). Note that, when $\beta = 0$, we recover the (strong) *gradient domination* (GD) property: $J_\dagger^* - J_\dagger(\boldsymbol{\theta}) \leqslant \alpha \|\nabla_{\boldsymbol{\theta}} J(_\dagger\boldsymbol{\theta})\|_2$ for all $\boldsymbol{\theta} \in \Theta$. GD is stricter than WGD and requires that $J_\dagger$ has no local optima. Instead, WGD admits local maxima as long as their performance is β-close to the globally optimal one.[5]

Assumption 6.2 (Smooth $J_\dagger$ w.r.t. parameters $\boldsymbol{\theta}$)**.** *$J_\dagger$ is $L_{2,\dagger}$-LS w.r.t. parameters $\boldsymbol{\theta}$, i.e., for every $\boldsymbol{\theta}, \boldsymbol{\theta}' \in \Theta$:*

$$\|\nabla_{\boldsymbol{\theta}} J_\dagger(\boldsymbol{\theta}') - \nabla_{\boldsymbol{\theta}} J_\dagger(\boldsymbol{\theta})\|_2 \leqslant L_{2,\dagger} \|\boldsymbol{\theta}' - \boldsymbol{\theta}\|_2. \tag{14}$$

Assumption 6.2 is ubiquitous in the convergence analysis of policy gradient algorithms (Papini et al., 2018; Agarwal et al., 2021; Yuan et al., 2022; Bhandari & Russo, 2024), which is usually studied as an instance of (non-convex) *smooth* stochastic optimization. The smoothness of $J_\dagger \in \{J_\mathrm{A}, J_\mathrm{P}\}$ can be: (i) inherited from the deterministic objective J_D (originating, in turn, from the regularity of the MDP) and of the deterministic policy $\boldsymbol{\mu}_{\boldsymbol{\theta}}$ (Asm. 4.1 and 4.4); or (ii) enforced through the properties on the white noise Φ (Asm. 4.5). The first result was observed in a similar form by Pirotta et al. (2015, Theorem 3), while a generalization of the second was established by Papini et al. (2022) and refined by Yuan et al. (2022).

Assumption 6.3 (Bounded estimator variance $\widehat{\nabla}_{\boldsymbol{\theta}} J_\dagger(\boldsymbol{\theta})$)**.** *The estimator $\widehat{\nabla}_{\boldsymbol{\theta}} J_\dagger(\boldsymbol{\theta})$ computed with batch size N has a bounded variance, i.e., there exists $V_\dagger \geqslant 0$ such that, for every $\boldsymbol{\theta} \in \Theta$, we have $\mathbb{V}\mathrm{ar}[\widehat{\nabla}_{\boldsymbol{\theta}} J_\dagger(\boldsymbol{\theta})] \leqslant V_\dagger / N$.*

Assumption 6.3 guarantees that the gradient estimator is characterized by a bounded variance $V_\dagger$ which scales with the batch size N. Under Assumption 4.5 (and 4.3 for GPOMDP), the term $V_\dagger$ can be further characterized (see Table 2 in Appendix A).

We are now ready to state the global convergence result.

Theorem 6.1. *Consider an algorithm running the update rule of Equation (13). Under Assumptions 6.1, 6.2, and 6.3, with a suitable constant step size, to guarantee $J_\dagger^* - \mathbb{E}[J_\dagger(\boldsymbol{\theta}_K)] \leqslant \epsilon + \beta$ the sample complexity is at most:*

$$NK = \frac{16\alpha^4 L_{2,\dagger} V_\dagger}{\epsilon^3} \log \frac{\max\{0, J_\dagger^* - J_\dagger(\boldsymbol{\theta}_0) - \beta\}}{\epsilon}. \tag{15}$$

This result establishes a convergence of order $\widetilde{O}(\epsilon^{-3})$[6] to the global optimum $J_\dagger^*$ of the general objective $J_\dagger$. Recalling that $J_\dagger \in \{J_\mathrm{A}, J_\mathrm{P}\}$, Theorem 6.1 provides: (i) the first global convergence guarantee for PGPE for PB exploration (setting $J_\dagger = J_\mathrm{P}$) and (ii) a global convergence guarantee for PG (e.g., GPOMDP) for AB exploration of the same order (up to logarithmic terms in ϵ^{-1}) of the state-of-the-art one of Yuan et al. (2022) (setting $J_\dagger = J_\mathrm{A}$). Note that our guarantee is obtained for a *constant* step size and holds for the last parameter $\boldsymbol{\theta}_K$, delivering a *last-iterate* result, rather than a *best-iterate* one as in (Yuan et al., 2022, Corollary 3.7). Clearly, this result is not yet our ultimate goal since, we need to assess how far the performance of the learned parameter $\boldsymbol{\theta}_K$ is from that of the optimal deterministic objective J_D^*.

6.2. Global Convergence of PGPE and GPOMDP

In this section, we provide results on the global convergence of PGPE and GPOMDP with white-noise exploration. The sample complexity bounds are summarized in Table 1 and presented extensively in Appendix D. They all follow from our general Theorem 6.1 and our results on the deployment of deterministic policies from Section 5.

PGPE. We start by commenting on the sample complexity of PGPE for a constant, generic hyperpolicy variance σ_P, shown in the first column (Table 1). First, the guarantee on $J_D^* - \mathbb{E}[J_D(\boldsymbol{\theta}_K)]$ contains the additional variance-dependent term $3L_P\sqrt{d_\Theta}\sigma_P$ originating from the deterministic deployment. Second, the sample complexity scales with $\widetilde{O}(\epsilon^{-3})$. Third, by enforcing the smoothness of the MDP and of the deterministic policy (Asm. 4.2 and 4.4), we improve the dependence on d_Θ and on σ_P at the price of an additional $(1-\gamma)^{-1}$ factor.

A choice of σ_P which adapts to ϵ allows us to achieve the global convergence on the deterministic objective J_D, up to $\epsilon + \beta$ only. Moving to the second column (Table 1), we observe that the convergence rate becomes $\widetilde{O}(\epsilon^{-7})$, which reduces to $\widetilde{O}(\epsilon^{-5})$ with the additional smoothness assumptions, which also improve the dependence on *both* $(1-\gamma)^{-1}$ and d_Θ. The slower rate ϵ^{-5} or ϵ^{-7}, compared to the ϵ^{-3} of the fixed-variance case, is easily explained by the more challenging requirement of converging to the *optimal deterministic policy* rather than the *optimal stochastic hyperpolicy*, as for standard PGPE. Note that we have set the standard deviation equal to $\sigma_\mathrm{P} = \frac{\epsilon}{6L_P\sqrt{d_\Theta}} = O(\epsilon(1-\gamma)^2 d_\Theta^{-1/2})$ that, as expected, decreases with the desired accuracy ϵ.[7]

GPOMDP. We now consider the global convergence of GPOMDP, starting again with a generic policy variance σ_A (third column, Table 1). The result is similar to that of PGPE with three notable exceptions. First, an additional $(1-\gamma)^{-1}$ factor appears in the sample complexity due to the variance bound of GPOMDP (Papini et al., 2022). This suggests that GPOMDP struggles more than PGPE in long-horizon environments, as already observed by Zhao et al.

[5] In this section, we will assume that $J_\dagger$ (i.e., either J_A or J_P) is already endowed with the WGD property. In Section 7, we illustrate how it can be obtained in several common scenarios.

[6] The $\widetilde{\mathcal{O}}(\cdot)$ notation hides logarithmic factors.

[7] These results should be interpreted as a demonstration that global convergence to deterministic policies *is possible* rather than a *practical recipe* to set the value of σ_P. We do hope that our theory can guide the design of practical solutions in future works.

	PGPE		GPOMDP	
	Generic σ_P	$\sigma_P = \epsilon/(6L_P\sqrt{d_\Theta})$	Generic σ_A	$\sigma_A = \epsilon/(6L_A\sqrt{d_\mathcal{A}})$
Without smoothness	Under Asm. 6.1, 4.1, 4.3, 4.5		Under Asm. 6.1, 4.1, 4.3, 4.4, 4.5	
	$\frac{\alpha^4 d_\Theta^2}{\sigma_\text{P}^4(1-\gamma)^4\epsilon^3}$ (Theorem D.4)	$\frac{\alpha^4 d_\Theta^4}{(1-\gamma)^{12}\epsilon^7}$ (Theorem D.5)	$\frac{\alpha^4 d_\mathcal{A}^2}{\sigma_\text{A}^4(1-\gamma)^5\epsilon^3}$ (Theorem D.8)	$\frac{\alpha^4 d_\mathcal{A}^4}{(1-\gamma)^{13}\epsilon^7}$ (Theorem D.9)
With smoothness	With additional Asm. 4.2, 4.4		With additional Asm. 4.2	
	$\frac{\alpha^4 d_\Theta}{\sigma_\text{P}^2(1-\gamma)^5\epsilon^3}$ (Theorem D.4)	$\frac{\alpha^4 d_\Theta^2}{(1-\gamma)^9\epsilon^5}$ (Theorem D.5)	$\frac{\alpha^4 d_\mathcal{A}}{\sigma_\text{A}^2(1-\gamma)^6\epsilon^3}$ (Theorem D.8)	$\frac{\alpha^4 d_\mathcal{A}^2}{(1-\gamma)^{10}\epsilon^5}$ (Theorem D.9)
$J_\text{D}^* - \mathbb{E}[J_\text{D}(\boldsymbol{\theta}_K)] \leqslant$	$\epsilon + \beta + 3\sigma_P L_P\sqrt{d_\Theta}$	$\epsilon + \beta$	$\epsilon + \beta + 3\sigma_\text{A} L_A\sqrt{d_\mathcal{A}}$	$\epsilon + \beta$

Table 1. Sample complexity $NK = \widetilde{O}(\cdot)$ of GPOMDP and PGPE to converge to a deterministic optimal policy, retaining only dependencies on ϵ, $(1-\gamma)^{-1}$, σ_A, σ_P, d_Θ, $d_\mathcal{A}$, and α. Task-dependent constants L_P and L_A are $O((1-\gamma)^{-2})$—see Table 2 in Appendix A.

(2011). Second, the dependence on the dimensionality of the parameter space d_Θ is replaced with the dimensionality of the action space $d_\mathcal{A}$. This is expected and derives from the nature of exploration that is performed in the parameter space for PGPE and in the action space for GPOMPD. Finally, the smoothness of the deterministic policy (Asm. 4.4) is always needed. Adding also the smoothness of the MDP (Asm. 4.2), we lose a $d_\mathcal{A}$ factor getting a $(1-\gamma)^{-1}$ one.

Again, a careful ϵ-dependent choice of σ_A allows us to achieve global convergence on the deterministic objective J_D. In the last column (Table 1), we can notice that the convergence rates display the same dependence on ϵ as in PGPE. However, the dependence on the effective horizon $(1-\gamma)^{-1}$ is worse. In this case, the additional smoothness assumption improves the dependency on $d_\mathcal{A}$ and $(1-\gamma)^{-1}$.

7. About the Weak Gradient Domination

So far, we have assumed WGD for the AB J_A and PB J_P (Asm. 6.1). In this section, we discuss several scenarios in which such an assumption holds.

7.1. Inherited Weak Gradient Domination

We start by discussing the case in which the deterministic policy objective J_D already enjoys the (W)GD property.

Assumption 7.1 (Weak gradient domination for J_D)**.** *There exist $\alpha_D > 0$ and $\beta_D \geqslant 0$ such that for every $\boldsymbol{\theta} \in \Theta$ it holds that $J_D^* - J_D(\boldsymbol{\theta}) \leqslant \alpha_D \|\nabla_{\boldsymbol{\theta}} J_D(\boldsymbol{\theta})\|_2 + \beta_D$.*

Although the notion of WGD has been mostly applied to stochastic policies in the literature (Liu et al., 2020; Yuan et al., 2022), there is no reason why it should not be plausible for deterministic policies. Bhandari & Russo (2024) provide sufficient conditions for the performance function not to have any local optima, which is a stronger condition, without discriminating between deterministic and stochastic policies (see their Remark 1). Moreover, one of their examples is linear-quadratic regulators with *deterministic* linear policies.

We show that, under Lipschiztianity and smoothness of the MDP and the deterministic policy (Asm. 4.1 and 4.4), this is sufficient to enforce the WGD property for both the PB J_P and the AB J_A objectives. Let us start with J_P.

Theorem 7.1 (Inherited weak gradient domination for J_P)**.** *Under Assumptions 4.1, 4.2, 4.3, 4.4, 7.1, for every $\boldsymbol{\theta} \in \Theta$:*

$$J_P^* - J_P(\boldsymbol{\theta}) \leqslant \alpha_D \|\nabla_{\boldsymbol{\theta}} J_P(\boldsymbol{\theta})\|_2 + \beta_D + (\alpha_D L_2 + L_P)\sigma_P\sqrt{d_\Theta},$$

where $L_2 = O((1-\gamma)^{-3})$ (full expression in Lemma E.2).

The result shows that the WGD property of J_D entails that of J_P with the same α_D coefficient, but a different $\beta = \beta_\text{D}(\alpha_\text{D} L_2 + L_P)\sigma_\text{P}\sqrt{d_\Theta}$ that accounts for the gap between the two objectives encoded in σ_P. Note that even if J_D enjoys a (strong) GD (i.e., $\beta_\text{D} = 0$), in general, J_P inherits a WGD property. In the setting of Theorem 7.1, convergence in the sense of $J_\text{D}^* - \mathbb{E}[J_\text{D}(\boldsymbol{\theta}_K)] \leqslant \epsilon + \beta_\text{D}$ can be achieved with $\widetilde{O}(\alpha_\text{D}^6 \epsilon^{-5} d_\Theta^2 (1-\gamma)^{-11})$ samples by carefully setting the hyperpolicy variance (see Theorem D.12 for details).

An analogous result can be obtained for AB exploration.

Theorem 7.2 (Inherited weak gradient domination on J_A)**.** *Under Assumptions 4.1, 4.2, 4.3, 4.4, 7.1, for every $\boldsymbol{\theta} \in \Theta$:*

$$J_A^* - J_A(\boldsymbol{\theta}) \leqslant \alpha_D \|\nabla_{\boldsymbol{\theta}} J_A(\boldsymbol{\theta})\|_2 + \beta_D + (\alpha_D \psi + L_A)\sigma_A\sqrt{d_\mathcal{A}},$$

where $\psi = O((1-\gamma)^{-4})$ (full expression in the proof).

The sample complexity, in this case, is $\widetilde{O}(\alpha_D^6 \epsilon^{-5} d_\mathcal{A}^2 (1-\gamma)^{-14})$ (see Theorem D.13 for details).

7.2. Policy-induced Weak Gradient Domination

When the objective function does not enjoy weak gradient domination in the space of deterministic policies, we can still have WGD w.r.t. *stochastic* policies if they satisfy a condition known as Fisher-non-degeneracy (Liu et al., 2020; Ding et al., 2022). As far as we know, WGD by Fisher-non-

degeneracy is a peculiar property of AB exploration that has no equivalent in PB exploration. White-noise policies satisfying Assumption 4.5 are Fisher-non-degenerate under the following standard assumption (Liu et al., 2020).

Assumption 7.2 (Explorability)**.** *There exists* $\lambda_E > 0$ *s.t.* $\mathbb{E}_{\pi_{\boldsymbol{\theta}}}[\nabla_{\boldsymbol{\theta}} \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s}) \nabla_{\boldsymbol{\theta}} \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s})^\top] \succeq \lambda_E \mathbf{I}$ *for all* $\boldsymbol{\theta} \in \Theta$*, where the expectation over states is induced by the* stochastic *policy.*

We can use this fact to prove WGD for white-noise policies.

Theorem 7.3 (Policy-induced weak gradient domination)**.** *Under Assumptions 4.5 and 7.2, we have:*

$$J_A{}^* - J_A(\boldsymbol{\theta}) \leqslant C \frac{\sqrt{d_{\mathcal{A}}}\sigma_A}{\lambda_E} \|\nabla_{\boldsymbol{\theta}} J_A(\boldsymbol{\theta})\|_2 + \frac{\sqrt{\epsilon_{\text{bias}}}}{1-\gamma},$$

for some numerical constant $C > 0$*. Thus, Assumption 6.1 (*$\dagger$*=A) is satisfied with* $\alpha = C\frac{\sqrt{d_{\mathcal{A}}}\sigma_A}{\lambda_E}$ *and* $\beta = \frac{\sqrt{\epsilon_{\text{bias}}}}{1-\gamma}$.

Here ϵ_{bias} is the *compatible-critic error*, which can be very small for rich policy classes (Ding et al., 2022).[8] We can leverage this to prove the global convergence of GPOMDP as in Section 7.1, this time to $J_{\text{D}} - \mathbb{E}[J_{\text{D}}(\boldsymbol{\theta})] \leqslant \epsilon + \frac{\sqrt{\epsilon_{\text{bias}}}}{1-\gamma}$.

Tuning σ_{A}, we can achieve a sample complexity of $\widetilde{O}(\epsilon^{-1}\lambda_E^{-4} d_{\mathcal{A}}^4 (1-\gamma)^{-10})$ (see Theorem D.16 for details) This seems to violate the $\Omega(\epsilon^{-2})$ lower bound by Azar et al. (2013). However, the factor λ_E can depend on $\sigma_{\text{A}} = O(\epsilon)$ in highly non-trivial ways and, thus, can hide additional factors of ϵ. For this reason, the results granted by the Fisher-non-degeneracy of white-noise policies are not compared with the ones granted by inherited WGD from Section 7.1. Intuitively, λ_E encodes some difficulties of exploration that are absent in "nice" MDPs satisfying Assumption 7.1. See Appendix D.4 for further discussion and omitted proofs.

8. Related Works

In this section, we provide a discussion of previous works that addressed similar questions to the ones considered in this paper. Additional related works in Appendix B.

Convergence rates. The convergence of PG to stationary points at a rate of $O(\epsilon^{-4})$ was clear at least since (Sutton et al., 1999), although the recent work by Yuan et al. (2022) clarifies several aspects of the analysis and the required assumptions. Variants of REINFORCE with faster convergence, based on stochastic variance reduction, were explored much later (Papini et al., 2018; Xu et al., 2019), and the $O(\epsilon^{-3})$ rate of (Xu et al., 2020) is now believed to be optimal due to lower bounds from nonconvex stochastic optimization (Arjevani et al., 2023). The same holds for second-order methods (Shen et al., 2019; Arjevani et al., 2020). Although the convergence properties of PGPE are analogous to those of PG, they have not received the same attention, with the exception of (Xu et al., 2020), where the $O(\epsilon^{-3})$ rate is proved for a variance-reduced version of PGPE. Studying the convergence of PG to globally optimal policies under additional assumptions is a more recent endeavor, pioneered by works such as Scherrer & Geist (2014), Fazel et al. (2018), Bhandari & Russo (2024). These works introduced to the policy gradient literature the concept of gradient domination, or gradient *dominance*, or Polyak-Łojasiewicz condition, which has a long history in the optimization literature (Lojasiewicz, 1963; Polyak et al., 1963; Karimi et al., 2016). Several works study the iteration complexity of policy gradient with exact gradients (e.g., Agarwal et al., 2021; Mei et al., 2020; Li et al., 2021). These results are restricted to specific policy classes (e.g., softmax, direct tabular parametrization) for which gradient domination is guaranteed. A notable exception is the study of sample-based *natural* policy gradient for general smooth policies (Agarwal et al., 2021). As for vanilla sample-based PG (i.e., GPOMDP), Liu et al. (2020) were the first to study the sample complexity of this algorithm in converging to a global optimum. They also introduced the concept of Fisher-non-degeneracy (Ding et al., 2022), which allows to exploit a form of gradient domination for a general class of policies. We refer the reader to (Yuan et al., 2022) which achieves a better $\widetilde{O}(\epsilon^{-3})$ sample complexity under weaker assumptions. More sophisticated algorithms, such as variance-reduced methods mentioned above, can achieve even better sample complexity. The current state of the art is (Fatkhullin et al., 2023): $\widetilde{O}(\epsilon^{-2.5})$ for hessian-free and $\widetilde{O}(\epsilon^{-2})$ for second-order algorithms. The latter is optimal up to logarithmic terms (Azar et al., 2013). When instantiated to Gaussian policies, all of the works mentioned in this paragraph implicitly assume that the covariance parameters are fixed. In this case, our Theorem D.4 recovers the $\widetilde{O}(\epsilon^{-3})$ rate of Yuan et al. (2022, Corollary 3.7), the best-known result for GPOMDP under general WGD.

Deterministic policies. Value-based RL algorithms, such as Q-learning, naturally produce deterministic policies as their final solution, while most policy-gradient methods must search, by design, in a space of non-degenerate stochastic policies. In (Sutton et al., 1999), this is presented as an opportunity rather than as a limitation since the optimal policy is often stochastic for partially observable problems. The possibility of deploying deterministic policies only is one of the appeals of PGPE and related evolutionary techniques (Schwefel, 1993), but also of model-based approaches (Deisenroth & Rasmussen, 2011). In the context of action-based policy search, the DPG algorithm by Silver et al. (2014) was the first to search in a space of deterministic policies. Differently from PGPE, stochastic policies are run during the learning process for exploration purposes, similarly to value-based methods. Moreover, the distribution mismatch due to off-policy sampling is largely ignored. Nonetheless, popular deep RL algorithms were derived from

[8] A formal definition of $\varepsilon_{\text{bias}}$ can be found in Appendix D.4.

DPG (Lillicrap et al., 2016; Fujimoto et al., 2018). (Xiong et al., 2022) proved the convergence of *on-policy* (hence, fully deterministic) DPG to a stationary point, with $O(\epsilon^{-4})$ sample complexity. However, they rely on an explorability assumption (Asm. 4 in their paper) that is standard for stochastic policies, but very demanding for deterministic policies. A more practical way of achieving fully deterministic DPG was proposed by Saleh et al. (2022), who also provide a discussion of the advantages of deterministic policies. Unsurprisingly, truly deterministic learning is only possible under strong assumptions on the regularity of the environment. In this paper, for PG, we considered the more common scenario of evaluating stochastic policies at training time, only to deploy a good deterministic policy in the end. PGPE, by design, does the same with hyperpolicies.

9. Numerical Validation

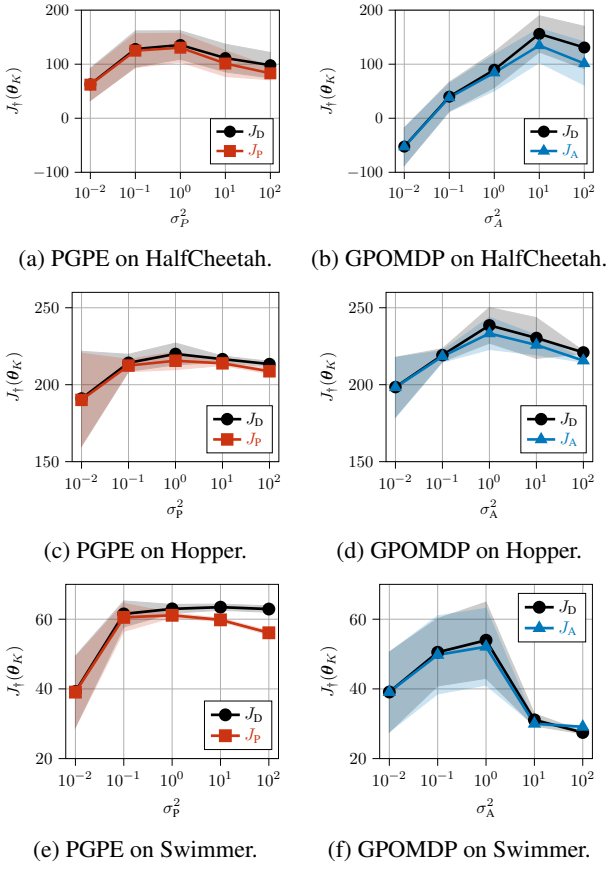
In this section, we empirically validate the theoretical results presented in the paper. We conduct a study on the gap in performance between the deterministic objective J_{D} and the ones of GPOMDP and PGPE (respectively J_{A} and J_{P}) by varying the value of their exploration parameters (σ_{A} and σ_{P}, respectively). Details on the employed versions of PGPE and GPOMDP can be found in Appendix G. Additional experimental results can be found in Appendix H.

We run PGPE and GPOMDP for $K = 2000$ iterations with batch size $N = 100$ on three environments from the MuJoCo (Todorov et al., 2012) suite: *Swimmer-v4* ($T = 200$), *Hopper-v4* ($T = 100$), and *HalfCheetah-v4* ($T = 100$). For all the environments the deterministic policy is linear in the state and the noise is Gaussian. We consider $\sigma^2_{\dagger} \in \{0.01, 0.1, 1, 10, 100\}$. More details in Appendix H.1.[9]

From Figure 1, we note that as the exploration parameter grows, the distance of $J_{\mathrm{P}}(\boldsymbol{\theta}_K)$ and $J_{\mathrm{A}}(\boldsymbol{\theta}_K)$ from $J_{\mathrm{D}}(\boldsymbol{\theta}_K)$ increases, coherently with Theorems 5.1 and 5.2. Among the tested values for σ_{P} and σ_{A}, some lead to the highest values of $J_{\mathrm{D}}(\boldsymbol{\theta}_K)$. Empirically, we note that PGPE delivers the best deterministic policy with $\sigma^2_{\mathrm{P}} = 10$ for *Swimmer* and with $\sigma^2_{\mathrm{P}} = 1$ for the other environments. GPOMDP performs the best with $\sigma^2_{\mathrm{A}} = 1$ for *Swimmer*, and with $\sigma^2_{\mathrm{A}} = 10$ in the other cases. These outcomes agree with the theoretical results in showing that there exists an optimal value for $\sigma_{\dagger}$.

We can also appreciate the trade-off between GPOMDP and PGPE w.r.t. d_Θ and T, by comparing the best values of J_{D} found by the two algorithms in each environment. GPOMDP is better than PGPE in *Hopper* and *HalfCheetah*. Indeed, such environments are characterized by higher values of d_Θ. Instead in *Swimmer*, PGPE performs better than GPOMDP, since T is higher and d_Θ is lower.

[9]The code is available at https://github.com/MontenegroAlessandro/MagicRL.



(a) PGPE on HalfCheetah. (b) GPOMDP on HalfCheetah.

(c) PGPE on Hopper. (d) GPOMDP on Hopper.

(e) PGPE on Swimmer. (f) GPOMDP on Swimmer.

Figure 1. Variance study on Mujoco (5 runs, mean $\pm$ 95% C.I.).

10. Conclusions

In this work, we have perfected recent theoretical results on the global convergence of policy gradient algorithms to address the practical problem of finding a good *deterministic* parametric policy. We have studied the effects of noise on the learning process and identified a theoretical value of the variance of the (hyper)policy that allows to find a good deterministic policy using a polynomial number of samples. We have compared the two common forms of noisy exploration, action-based and parameter-based, both from a theoretical and an empirical perspective.

Our work paves the way for several exciting research directions. First, our theoretical selection of the policy variance is not practical, but our theoretical findings should guide the design of sound and efficient adaptive-variance schedules. We have shown how white-noise exploration preserves *weak* gradient domination—the natural next question is whether a sufficient amount of noise can smooth or even eliminate the local optima of the objective function. Finally, we have focused on "vanilla" policy gradient methods, but our ideas could be applied to more advanced algorithms, such as the ones recently proposed by Fatkhullin et al. (2023), to find optimal deterministic policies with $\widetilde{O}(\epsilon^{-2})$ samples.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

Acknowledgements

Funded by the European Union – Next Generation EU within the project NRPP M4C2, Investment 1.,3 DD. 341 - 15 march 2022 – FAIR – Future Artificial Intelligence Research – Spoke 4 - PE00000013 - D53C22002380006.

References

Agarwal, A., Kakade, S. M., Lee, J. D., and Mahajan, G. On the theory of policy gradient methods: Optimality, approximation, and distribution shift. *Journal of Machine Learning Research (JMLR)*, 22:98:1–98:76, 2021.

Ahmed, Z., Roux, N. L., Norouzi, M., and Schuurmans, D. Understanding the impact of entropy on policy optimization. In *International Conference on Machine Learning (ICML)*, volume 97 of *Proceedings of Machine Learning Research*, pp. 151–160. PMLR, 2019.

Allgower, E. L. and Georg, K. *Numerical continuation methods - an introduction*, volume 13 of *Springer series in computational mathematics*. Springer, 1990.

Arjevani, Y., Carmon, Y., Duchi, J. C., Foster, D. J., Sekhari, A., and Sridharan, K. Second-order information in non-convex stochastic optimization: Power and limitations. In *Proceedings of the Annual Conference on Learning Theory (COLT)*, volume 125 of *Proceedings of Machine Learning Research*, pp. 242–299. PMLR, 2020.

Arjevani, Y., Carmon, Y., Duchi, J. C., Foster, D. J., Srebro, N., and Woodworth, B. E. Lower bounds for non-convex stochastic optimization. *Math. Program.*, 199(1):165–214, 2023.

Azar, M. G., Munos, R., and Kappen, H. J. Minimax PAC bounds on the sample complexity of reinforcement learning with a generative model. *Machine Learning*, 91(3): 325–349, 2013.

Azizzadenesheli, K., Yue, Y., and Anandkumar, A. Policy gradient in partially observable environments: Approximation and convergence. *arXiv preprint arXiv:1810.07900*, 2018.

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

Bhandari, J. and Russo, D. Global optimality guarantees for policy gradient methods. *Operations Research*, 2024.

Bolland, A., Louppe, G., and Ernst, D. Policy gradient algorithms implicitly optimize by continuation. *Transactions on Machine Learning Research*, 2023.

Deisenroth, M. P. and Rasmussen, C. E. PILCO: A model-based and data-efficient approach to policy search. In *International Conference on Machine Learning (ICML)*, pp. 465–472. Omnipress, 2011.

Deisenroth, M. P., Neumann, G., and Peters, J. A survey on policy search for robotics. *Foundations and Trends in Robotics*, 2(1-2):1–142, 2013.

Ding, Y., Zhang, J., and Lavaei, J. On the global optimum convergence of momentum-based policy gradient. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 151 of *Proceedings of Machine Learning Research*, pp. 1910–1934. PMLR, 2022.

Duan, Y., Chen, X., Houthooft, R., Schulman, J., and Abbeel, P. Benchmarking deep reinforcement learning for continuous control. In *International Conference on Machine Learning (ICML)*, pp. 1329–1338. PMLR, 2016.

Fatkhullin, I., Barakat, A., Kireeva, A., and He, N. Stochastic policy gradient methods: Improved sample complexity for fisher-non-degenerate policies. In *International Conference on Machine Learning (ICML)*, volume 202 of *Proceedings of Machine Learning Research*, pp. 9827–9869. PMLR, 2023.

Fazel, M., Ge, R., Kakade, S. M., and Mesbahi, M. Global convergence of policy gradient methods for the linear quadratic regulator. In *International Conference on Machine Learning (ICML)*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1466–1475. PMLR, 2018.

Fujimoto, S., van Hoof, H., and Meger, D. Addressing function approximation error in actor-critic methods. In *International Conference on Machine Learning (ICML)*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1582–1591. PMLR, 2018.

Ghavamzadeh, M. and Engel, Y. Bayesian policy gradient algorithms. *Advances in Neural Information Processing Systems (NeurIPS)*, 19, 2006.

Ghavamzadeh, M., Lazaric, A., and Pirotta, M. Exploration in reinforcement learning. Tutorial at AAAI'20, 2020.

Gravell, B., Esfahani, P. M., and Summers, T. Learning optimal controllers for linear systems with multiplicative noise via policy gradient. *IEEE Transactions on Automatic Control*, 66(11):5283–5298, 2020.

Kakade, S. M. A natural policy gradient. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 1531–1538. MIT Press, 2001.

Karimi, H., Nutini, J., and Schmidt, M. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *Machine Learning and Knowledge Discovery in Databases: European Conference (ECML PKDD)*, pp. 795–811. Springer, 2016.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Kucera, V. Optimal control: Linear quadratic methods: Brian d. o. anderson and john b. moore. *Autom.*, 28(5): 1068–1069, 1992.

Kumar, H., Kalogerias, D. S., Pappas, G. J., and Ribeiro, A. Zeroth-order deterministic policy gradient. *arXiv preprint arXiv:2006.07314*, 2020.

Li, G., Wei, Y., Chi, Y., Gu, Y., and Chen, Y. Softmax policy gradient methods can take exponential time to converge. In *Proceedings of the Annual Conference on Learning Theory (COLT)*, volume 134 of *Proceedings of Machine Learning Research*, pp. 3107–3110. PMLR, 2021.

Likmeta, A., Metelli, A. M., Tirinzoni, A., Giol, R., Restelli, M., and Romano, D. Combining reinforcement learning with rule-based controllers for transparent and general decision-making in autonomous driving. *Robotics and Autonomous Systems*, 131:103568, 2020.

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. In *International Conference on Learning Representations (ICLR)*, 2016.

Liu, Y., Zhang, K., Basar, T., and Yin, W. An improved analysis of (variance-reduced) policy gradient and natural policy gradient methods. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Lojasiewicz, S. Une propriété topologique des sous-ensembles analytiques réels. *Les équations aux dérivées partielles*, 117:87–89, 1963.

Masiha, S., Salehkaleybar, S., He, N., Kiyavash, N., and Thiran, P. Stochastic second-order methods improve best-known sample complexity of sgd for gradient-dominated functions. *Advances in Neural Information Processing Systems (NeurIPS)*, 35:10862–10875, 2022.

Mei, J., Xiao, C., Szepesvári, C., and Schuurmans, D. On the global convergence rates of softmax policy gradient methods. In *International Conference on Machine Learning (ICML)*, volume 119 of *Proceedings of Machine Learning Research*, pp. 6820–6829. PMLR, 2020.

Metelli, A. M., Papini, M., Faccio, F., and Restelli, M. Policy optimization via importance sampling. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 5447–5459, 2018.

Metelli, A. M., Papini, M., Montali, N., and Restelli, M. Importance sampling techniques for policy optimization. *J. Mach. Learn. Res.*, 21:141:1–141:75, 2020.

Metelli, A. M., Papini, M., D'Oro, P., and Restelli, M. Policy optimization as online learning with mediator feedback. In *AAAI Conference on Artificial Intelligence (AAAI)*, pp. 8958–8966. AAAI Press, 2021.

Papini, M., Binaghi, D., Canonaco, G., Pirotta, M., and Restelli, M. Stochastic variance-reduced policy gradient. In *International Conference on Machine Learning (ICML)*, volume 80 of *Proceedings of Machine Learning Research*, pp. 4023–4032. PMLR, 2018.

Papini, M., Battistello, A., and Restelli, M. Balancing learning speed and stability in policy gradient via adaptive exploration. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 108 of *Proceedings of Machine Learning Research*, pp. 1188–1199. PMLR, 2020.

Papini, M., Pirotta, M., and Restelli, M. Smoothing policies and safe policy gradients. *Machine Learning*, 111(11): 4081–4137, 2022.

Peters, J. and Schaal, S. Policy gradient methods for robotics. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2219–2225. IEEE, 2006.

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

Peters, J., Vijayakumar, S., and Schaal, S. Natural actor-critic. In *European Conference on Machine Learning (ECML)*, volume 3720 of *Lecture Notes in Computer Science*, pp. 280–291. Springer, 2005.

Pirotta, M., Restelli, M., and Bascetta, L. Policy gradient in lipschitz markov decision processes. *Machine Learning*, 100:255–283, 2015.

Polyak, B. T. et al. Gradient methods for minimizing functionals. *Zhurnal vychislitel'noi matematiki i matematicheskoi fiziki*, 3(4):643–653, 1963.

Puterman, M. L. Markov decision processes. *Handbooks in operations research and management science*, 2:331–434, 1990.

Raffin, A., Hill, A., Gleave, A., Kanervisto, A., Ernestus, M., and Dormann, N. Stable-baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research (JMLR)*, 22(268):1–8, 2021.

Saleh, E., Ghaffari, S., Bretl, T., and West, M. Truly deterministic policy optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Scherrer, B. and Geist, M. Local policy search in a convex space and conservative policy iteration as boosted policy search. In *Machine Learning and Knowledge Discovery in Databases: European Conference (ECML PKDD)*, volume 8726 of *Lecture Notes in Computer Science*, pp. 35–50. Springer, 2014.

Schwefel, H.-P. P. *Evolution and optimum seeking: the sixth generation*. John Wiley & Sons, Inc., 1993.

Sehnke, F., Osendorfer, C., Rückstieß, T., Graves, A., Peters, J., and Schmidhuber, J. Parameter-exploring policy gradients. *Neural Networks*, 23(4):551–559, 2010. The International Conference on Artificial Neural Networks (ICANN).

Shani, L., Efroni, Y., and Mannor, S. Adaptive trust region policy optimization: Global convergence and faster rates for regularized mdps. In *AAAI*, pp. 5668–5675. AAAI Press, 2020.

Shen, Z., Ribeiro, A., Hassani, H., Qian, H., and Mi, C. Hessian aided policy gradient. In *International Conference on Machine Learning (ICML)*, volume 97 of *Proceedings of Machine Learning Research*, pp. 5729–5738. PMLR, 2019.

Silver, D., Lever, G., Heess, N., Degris, T., Wierstra, D., and Riedmiller, M. Deterministic policy gradient algorithms. In *International Conference on Machine Learning (ICML)*, pp. 387–395. PMLR, 2014.

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

Sutton, R. S., McAllester, D., Singh, S., and Mansour, Y. Policy gradient methods for reinforcement learning with function approximation. *Advances in Neural Information Processing Systems (NeurIPS)*, 12, 1999.

Todorov, E., Erez, T., and Tassa, Y. Mujoco: A physics engine for model-based control. In *IEEE/RSJ international conference on intelligent robots and systems*, pp. 5026–5033. IEEE, 2012.

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

Xiong, H., Xu, T., Zhao, L., Liang, Y., and Zhang, W. Deterministic policy gradient: Convergence analysis. In *Uncertainty in Artificial Intelligence (UAI)*, pp. 2159–2169. PMLR, 2022.

Xu, P., Gao, F., and Gu, Q. An improved convergence analysis of stochastic variance-reduced policy gradient. In *Uncertainty in Artificial Intelligence (UAI)*, volume 115 of *Proceedings of Machine Learning Research*, pp. 541–551. AUAI Press, 2019.

Xu, P., Gao, F., and Gu, Q. Sample efficient policy gradient methods with recursive variance reduction. In *International Conference on Learning Representations (ICLR)*. OpenReview.net, 2020.

Yuan, R., Gower, R. M., and Lazaric, A. A general sample complexity analysis of vanilla policy gradient. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pp. 3332–3380. PMLR, 2022.

Zhao, T., Hachiya, H., Niu, G., and Sugiyama, M. Analysis and improvement of policy gradient estimation. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 262–270, 2011.

A. Assumptions and Constants: Quick Reference

As mentioned in Section 7, we can start from fundamental assumptions on the MDP and the (hyper)policy classes to satisfy more abstract assumptions that can be used directly in convergence analyses. Figure 2 shows the relationship between the assumptions, and Table 2 the constants obtained in the process. All proofs of the assumptions' implications can be found in Appendix E.

	$L_\dagger$ (Lipschitz)	$L_{2,\dagger}$ (Smooth)		$V_\dagger$ (Variance bound)
AB Exploration ($\dagger$=A)	$\frac{L_p R_{\max}}{(1-\gamma)^2} + \frac{L_r}{1-\gamma}$	$\frac{2L_p^2 L_\mu^2 R_{\max}}{(1-\gamma)^3} + \frac{2L_\mu^2 L_p L_r + L_{2,\mu} L_{2,p} R_{\max}}{(1-\gamma)^2} + \frac{L_{2,\mu} L_{2,r}}{1-\gamma}$	$\frac{R_{\max} c (d_{\mathcal{A}}+1)(L_\mu^2 + L_{2,\mu})}{\sigma_{\text{A}}^2 (1-\gamma)^2}$ ‡	$\frac{R_{\max} c d_{\mathcal{A}} L_\mu^2}{\sigma_{\text{A}}^2 (1-\gamma)^3}$
Assumptions:	4.1	4.1, 4.2, 4.3, 4.4	4.3, 4.4, 4.5	4.3, 4.5
Reference:	Lemma E.1	Lemma D.7		Lemma D.6
PB Exploration ($\dagger$=P)	$\frac{L_p L_\mu R_{\max}}{(1-\gamma)^2} + \frac{L_r L_\mu}{1-\gamma}$	$\frac{2L_p^2 L_\mu^2 R_{\max}}{(1-\gamma)^3} + \frac{2L_\mu^2 L_p L_r + L_{2,\mu} L_{2,p} R_{\max}}{(1-\gamma)^2} + \frac{L_{2,\mu} L_{2,r}}{1-\gamma}$	$\frac{R_{\max} c (d_\Theta + 1)}{\sigma_{\text{P}}^2 (1-\gamma)^2}$	$\frac{R_{\max} c d_\Theta}{\sigma_{\text{P}}^2 (1-\gamma)^2}$
Assumptions:	4.1, 4.3	4.1, 4.2, 4.3, 4.4	4.5	4.5
Reference:	Lemma E.1	Lemma D.3		Lemma D.2

Table 2. Bounds to the Lipschitz and smoothness constants for the AB and PB objectives (J_{A} and J_{P}) and variance of the GPOMDP and PGPE estimators. Both presented bounds on $L_{2,\dagger}$ hold under different sets of assumptions. ‡ if $\sigma_{\text{A}} < \sqrt{d_{\mathcal{A}}}$.

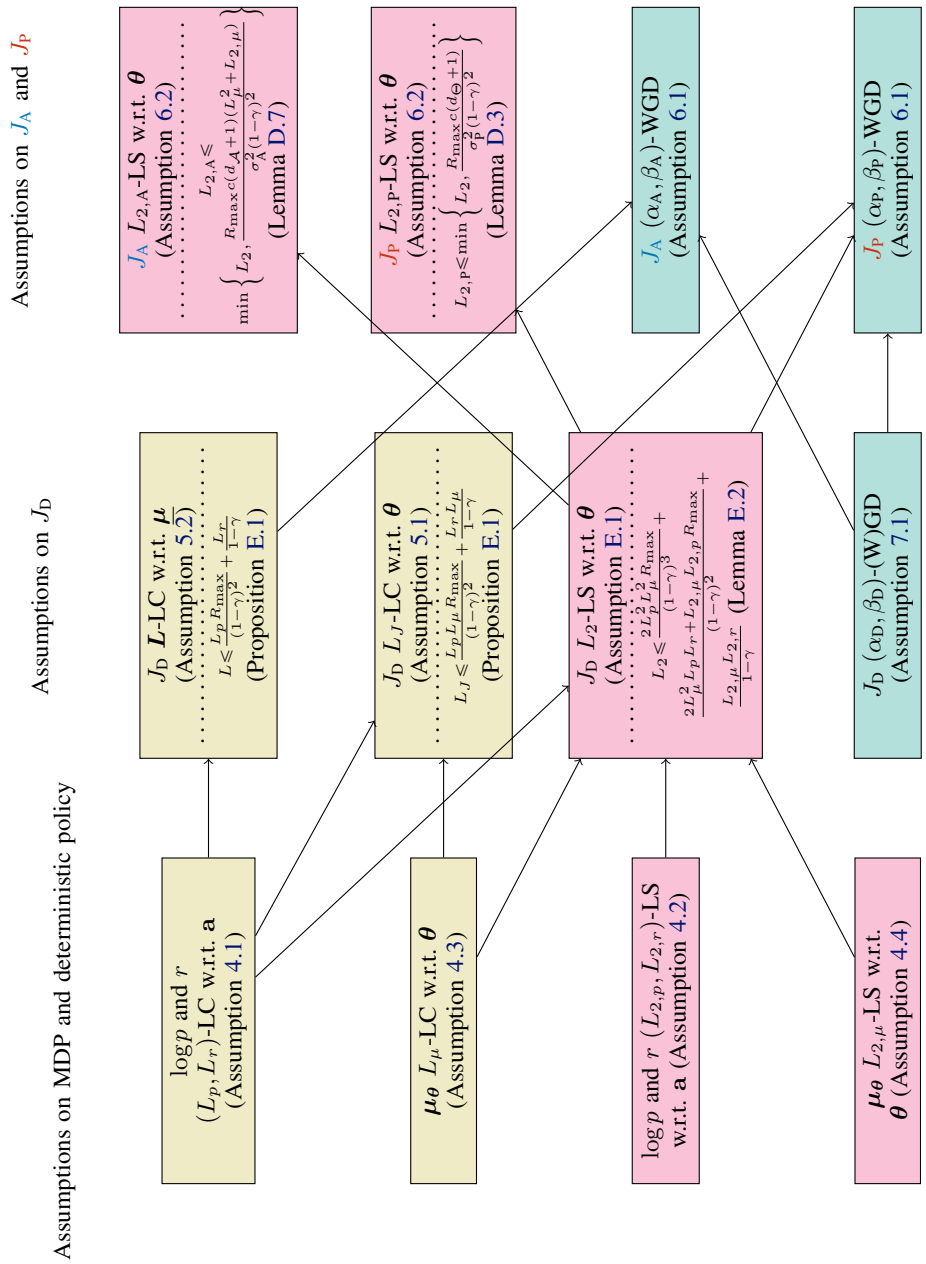


Figure 2. Dependency of some of the assumptions presented in the paper.

B. Additional Related Works

Policy variance. When optimizing Gaussian policies with policy-gradient methods, the scale parameters (those of the variance or, more in general, of the covariance matrix of the policy) are typically fixed in theory, and optimized via gradient descent in practice. To the best of our knowledge, there is no satisfying theory of the effects of a varying policy (or hyperpolicy) variance on the convergence rates of PG (or PGPE). Ahmed et al. (2019) were the first to take into serious consideration the impact of the policy stochasticity on the geometry of the objective function, although their focus was on entropy regularization. Papini et al. (2020), focusing on monotonic improvement rather than convergence, proposed to use second-order information to overcome the greediness of gradient updates, arguing that the latter is particularly harmful for scale parameters. Bolland et al. (2023) propose to study PG with Gaussian policies under the lens of *optimization by continuation* (Allgower & Georg, 1990), that is, as a sequence of smoothed version of the deterministic policy optimization problem. Unfortunately, the theory of optimization by continuation is rather scarce. We studied the impact of a *fixed* policy variance on the number of samples needed to find a good *deterministic* policy. We hope that this can provide some insight on how to design adaptive policy-variance strategies in future work. We remark here that the common practice of *learning* the exploration parameters together with all the other policy parameters breaks all of the known convergence results of GPOMDP, since the smoothness of the stochastic objective is inversely proportional to the policy variance (Papini et al., 2022). In this regard, entropy-regularized policy optimization is different, and is better studied using mirror descent theory, rather than stochastic gradient descent theory (Shani et al., 2020).

Comparing AB and PB exploration. A classic on the topic is the paper by Zhao et al. (2011). They prove upper bounds on the variance of the REINFORCE and PGPE estimators, highlighting the better dependence on the task horizon of the latter. The idea that variance reduction does not tell the whole story about the efficiency of policy gradient methods is rather recent (Ahmed et al., 2019). We revisited the comparison of action-based and parameter based methods under the lens of modern sample complexity theory. We reached similar conclusions but achieved, we believe, a more complete understanding of the matter. To our knowledge, the only other work that thoroughly compares AB and PB exploration is (Metelli et al., 2018; 2020; 2021), where the trade-off between the task horizon and the number of policy parameters is discussed both in theory and experiments, but in the context of trust-region methods.

C. Additional Considerations

We only considered (hyper)policy variances $\sigma_{\text{A}}^2, \sigma_{\text{P}}^2$ that are *fixed* for the duration of the learning process, albeit they can be set as functions of problem-dependent constants and of the desired accuracy ϵ. This is due to our focus on convergence guarantees based on smooth optimization theory, as explained in the following.

Remark C.1 (About learning the (hyper)policy variance)**.** *It is a well established practice to parametrize the policy variance and learn these exploration parameters via gradient descent together with all the other policy parameters (again, for examples, see Duan et al., 2016; Raffin et al., 2021). The same is true for parameter-based exploration (Schwefel, 1993; Sehnke et al., 2010). However, it is easy to see that an adaptive (in the sense of time-varying) policy variance breaks the sample complexity guarantees of GPOMDP (Yuan et al., 2022) and its variance-reduced variants (e.g., Liu et al., 2020). That is because these guarantees all rely on Assumption 6.2, or equivalent smoothness conditions, and obtain sample complexity upper bounds that scale with the smoothness constant $L_{2,A}$. However, the latter can depend* inversely *on σ_A^2, as already observed by Papini et al. (2022) for Gaussian policies. Thus, unconstrained learning of σ_A breaks the convergence guarantees. Analogous considerations hold for PGPE with adaptive hyperpolicy variance. Different considerations apply to entropy-regularized policy optimization methods, which were not considered in this paper, mostly because they converge to a surrogate objective that is even further from optimal deterministic performance. These methods are better analyzed using the theory of mirror descent. We refer the reader to (Shani et al., 2020).*

In order to properly define the white noise-based (hyper)policies, we need that $\boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s}) + \boldsymbol{\epsilon} \in \mathcal{A}$ (for AB exploration) and $\boldsymbol{\theta} + \boldsymbol{\epsilon} \in \Theta$ (for PB exploration), we will assume that $\mathcal{A} = \mathbb{R}^{d_{\mathcal{A}}}$ and $\Theta = \mathbb{R}^{d_\Theta}$ for simplicity.

Remark C.2 (About $\mathcal{A} = \mathbb{R}^{d_{\mathcal{A}}}$ and $\Theta = \mathbb{R}^{d_\Theta}$ assumption)**.** *We have assumed that the action space $\mathcal{A}$ and the parameter space Θ correspond to $\mathbb{R}^{d_{\mathcal{A}}}$ and $\mathbb{R}^{d_\Theta}$, respectively. If this is not the case, we can easily alter the transition model p and the reward function r (for the AB exploration), and the deterministic policy $\mu_{\boldsymbol{\theta}}$ (for the PB exploration) by means of a* retraction *function. Let $\mathcal{X} \subseteq \mathbb{R}^d$ be a measurable set, a retraction function $\iota_{\mathcal{X}} : \mathbb{R}^d \to \mathcal{X}$ is such that $\iota_{\mathcal{X}}(x) = x$ if $x \in \mathcal{X}$, i.e., it is the identity over $\mathcal{X}$.*

- *For the AB exploration, we redefine the transition model as $\overline{p}(\mathbf{s}'|\mathbf{s}, \mathbf{a}) := p(\mathbf{s}'|\mathbf{s}, \boldsymbol{\iota}_{\mathcal{A}}(\mathbf{a}))$ for every $\mathbf{s}, \mathbf{s}' \in \mathcal{S}$ and $\mathbf{a} \in \mathcal{A}$. Furthermore, we redefine the reward function as $\overline{r}(\mathbf{s}, \mathbf{a}) := r(\mathbf{s}, \iota_{\mathcal{A}}(\mathbf{a}))$ for every $\mathbf{s} \in \mathcal{S}$ and $\mathbf{a} \in \mathcal{A}$.*
- *For the PB exploration, we redefine the deterministic policy as $\overline{\boldsymbol{\mu}}_{\boldsymbol{\theta}}(\mathbf{s}) := \boldsymbol{\mu}_{\boldsymbol{\iota}_\Theta(\boldsymbol{\theta})}$, for every $\boldsymbol{\theta} \in \Theta$.*

D. Proofs

D.1. Proofs from Section 5

Lemma D.1. *Let $L>0$, consider the function $f:\mathbb{R}\to\mathbb{R}$ defined for every $x\in\mathbb{R}$ as follows:*

$$f(x)=\begin{cases}0 & \text{if } x<-1/L \text{ or } x>2/L\\ Lx+1 & \text{if } -1/L\leqslant x<0\\ 1-\frac{L}{2}x & \text{if } 0\leqslant x\leqslant 2/L\end{cases}. \tag{16}$$

Consider the function $\psi_\sigma:\mathbb{R}\to\mathbb{R}_{\geqslant 0}$ defined for every $x\in\mathbb{R}$ as follows:

$$\psi_\sigma(x)=\begin{cases}\frac{1}{2\sqrt{3}\sigma} & \text{if } -\sqrt{3}\sigma\leqslant x\leqslant\sqrt{3}\sigma\\ 0 & \text{otherwise}\end{cases}, \tag{17}$$

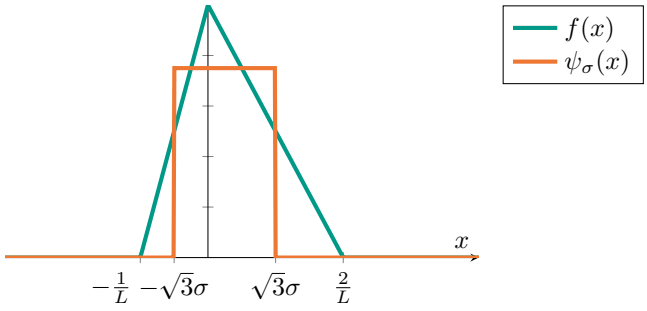
i.e., the p.d.f. of a uniform distribution with zero mean and variance σ^2. Let $f_\sigma:=f*\psi_\sigma$, let $x^*=\arg\max_{x\in\mathbb{R}} f(x)$, and let $x^*_\sigma=\arg\max_{x\in\mathbb{R}} f_\sigma(x)$. Then f is L-LC and, if $\sqrt{3}\sigma\leqslant 1/L$, it holds that $f(x^*)-f(x^*_\sigma)=L\sigma/(2\sqrt{3})$.

Proof. Let us first verify that the distribution whose p.d.f. is ϕ_σ has zero mean and variance σ^2:

$$\int_{\mathbb{R}}\psi_\sigma(x)x\mathrm{d}x=0, \tag{18}$$

$$\int_{\mathbb{R}}\psi_\sigma(x)x^2\mathrm{d}x=2\int_0^{\sqrt{3}\sigma}\psi_\sigma(x)x^2\mathrm{d}x=\sigma^2. \tag{19}$$

Under the assumption $\sqrt{3}\sigma\leqslant 1/L$, functions f and ψ_σ can be represented as follows:



Let us now compute the convolution:

$$f_\sigma(x)=f*\psi_\sigma=\int_{\mathbb{R}}\psi_\sigma(x-t)f(t)\mathrm{d}t. \tag{20}$$

It is clear that the global optimum of function f_σ is located in the interval given by $|x|\leqslant 1/L$. This combined, with the assumption $\sqrt{3}\sigma\leqslant 1/L$, allows to simplify the integral as:

$$\int_{\mathbb{R}}\psi_\sigma(x-t)f(t)\mathrm{d}t=\int_{x-\sqrt{3}\sigma}^{0}\frac{1}{2\sqrt{3}\sigma}(Lt+1)\mathrm{d}t+\int_0^{x+\sqrt{3}\sigma}\frac{1}{2\sqrt{3}\sigma}\left(1-\frac{L}{2}t\right)\mathrm{d}t \tag{21}$$

$$1-=\frac{1}{2\sqrt{3}\sigma}\left(\frac{L}{2}(x-\sqrt{3}\sigma)^2+\frac{L}{4}(x+\sqrt{3}\sigma)^2\right). \tag{22}$$

The latter is a concave (quadratic) function of x, which is maximized for $x^*_\sigma=\sigma/\sqrt{3}$. Noticing that $x^*=\arg\max_{x\in\mathbb{R}} f(x)$=0, we have:

$$f(x^*)-f(x^*_\sigma)=f(0)-f(\sigma/\sqrt{3})=1-\left(1-\frac{L\sigma}{2\sqrt{3}}\right)=\frac{L\sigma}{2\sqrt{3}}. \tag{23}$$

□

Theorem 5.1 (Deterministic deployment of parameters learned with PB white-noise exploration). *If the hyperpolicy complies with Definition 3.3, under Assumption 5.1:*

(i) (Uniform bound) *for every* $\boldsymbol{\theta} \in \Theta$, *it holds that* $|J_D(\boldsymbol{\theta}) - J_P(\boldsymbol{\theta})| \leqslant L_J \sqrt{d_\Theta} \sigma_P$;
(ii) (J_D upper bound) *let* $\boldsymbol{\theta}_P^* \in \arg\max_{\boldsymbol{\theta} \in \Theta} J_P(\boldsymbol{\theta})$, *it holds that:* $J_D^* - J_D(\boldsymbol{\theta}_P^*) \leqslant 2 L_J \sqrt{d_\Theta} \sigma_P$;
(iii) (J_D lower bound) *there exists an MDP, a deterministic policy class* $\boldsymbol{\mu}_{\boldsymbol{\theta}}$ *fulfilling Assumption 5.1, and a noise complying with Definition 3.1, such that* $J_D^* - J_D(\boldsymbol{\theta}_P^*) \geqslant 0.28 L_J \sqrt{d_\Theta} \sigma_P$.

Proof. Before starting the derivation, we remark that:

$$J_{\mathrm{P}}(\boldsymbol{\theta}) = \mathop{\mathbb{E}}_{\boldsymbol{\epsilon} \sim \Phi_{d_\Theta}} \left[J_{\mathrm{D}}(\boldsymbol{\theta} + \boldsymbol{\epsilon}) \right], \tag{24}$$

where $\mathbb{E}_{\boldsymbol{\epsilon} \sim \Phi_{d_\Theta}} [\|\boldsymbol{\epsilon}\|_2^2] \leqslant d_\Theta \sigma_{\mathrm{P}}^2$. From Assumption 5.1, we can easily derive (i):

$$\begin{aligned}
|J_{\mathrm{D}}(\boldsymbol{\theta}) - J_{\mathrm{P}}(\boldsymbol{\theta})| &= |J_{\mathrm{D}}(\boldsymbol{\theta}) - \mathop{\mathbb{E}}_{\boldsymbol{\epsilon} \sim \Phi_{d_\Theta}} \left[J_{\mathrm{D}}(\boldsymbol{\theta} + \boldsymbol{\epsilon}) \right]| && (25)\\
&\leqslant \mathop{\mathbb{E}}_{\boldsymbol{\epsilon} \sim \Phi_{d_\Theta}} \left[|J_{\mathrm{D}}(\boldsymbol{\theta}) - J_{\mathrm{D}}(\boldsymbol{\theta} + \boldsymbol{\epsilon})| \right] && (26)\\
&\leqslant L_J \mathop{\mathbb{E}}_{\boldsymbol{\epsilon} \sim \Phi_{d_\Theta}} \left[\|\boldsymbol{\epsilon}\|_2 \right] && (27)\\
&\leqslant L_J \sqrt{\mathop{\mathbb{E}}_{\boldsymbol{\epsilon} \sim \Phi_{d_\Theta}} \left[\|\boldsymbol{\epsilon}\|_2^2 \right]} && (28)\\
&\leqslant L_J \sigma_{\mathrm{P}} \sqrt{d_\Theta}. && (29)
\end{aligned}$$

For (ii), let $\boldsymbol{\theta}^* \in \arg\max_{\boldsymbol{\theta} \in \Theta} J_{\mathrm{D}}(\boldsymbol{\theta})$, we have:

$$\begin{aligned}
\max_{\boldsymbol{\theta} \in \Theta} J_{\mathrm{D}}(\boldsymbol{\theta}) - J_{\mathrm{D}}(\boldsymbol{\theta}_{\mathrm{P}}^*) &= J_{\mathrm{D}}(\boldsymbol{\theta}^*) - J_{\mathrm{D}}(\boldsymbol{\theta}_{\mathrm{P}}^*) \pm J_{\mathrm{P}}(\boldsymbol{\theta}^*) && (30)\\
&\leqslant J_{\mathrm{D}}(\boldsymbol{\theta}^*) - J_{\mathrm{P}}(\boldsymbol{\theta}^*) + J_{\mathrm{P}}(\boldsymbol{\theta}_{\mathrm{P}}^*) - J_{\mathrm{D}}(\boldsymbol{\theta}_{\mathrm{P}}^*) && (31)\\
&\leqslant 2 \max_{\boldsymbol{\theta} \in \Theta} |J_{\mathrm{D}}(\boldsymbol{\theta}) - J_{\mathrm{P}}(\boldsymbol{\theta})| && (32)\\
&\leqslant 2 L_J \sigma_{\mathrm{P}} \sqrt{d_\Theta}, && (33)
\end{aligned}$$

where line (31) follows from $J_{\mathrm{P}}(\boldsymbol{\theta}_{\mathrm{P}}^*) = \max_{\boldsymbol{\theta} \in \Theta} J_{\mathrm{P}}(\boldsymbol{\theta}) \geqslant J_{\mathrm{P}}(\boldsymbol{\theta}^*)$, and line (32), follows by applying twice result (i).

To prove (iii) we construct the MDP $(\{s\}, \mathbb{R}^{d_\Theta}, p, r, \rho_0, \gamma)$ (i.e., a bandit), where $r(s, \mathbf{a}) = \frac{1}{d_\Theta} \sum_{i=1}^{d_\Theta} f(a_i)$, where f is defined in Lemma D.1 and $\boldsymbol{\mu}_{\boldsymbol{\theta}}(s) = \boldsymbol{\theta}$ with $\boldsymbol{\theta} \in \mathbb{R}^{d_\Theta}$. Thus, we can compute the expected return as follows:

$$J_{\mathrm{D}}(\boldsymbol{\theta}) = \frac{1-\gamma^T}{1-\gamma} \cdot \frac{1}{d_\Theta} \sum_{i=1}^{d_\Theta} f(\theta_i). \tag{34}$$

Let us compute its Lipschitz constant recalling that f is L-LC thanks to Lemma D.1. In particular, we take $\boldsymbol{\theta} = \mathbf{0}_{d_\Theta}$ and $\boldsymbol{\theta}' = -\eta \mathbf{1}_{d_\Theta}$ with $\eta \in (0, 1/L)$, recalling that $\|\boldsymbol{\theta} - \boldsymbol{\theta}'\|_2 = \eta \sqrt{d_\Theta}$ and that $f(\theta_i) = 1$ and $f(\theta_i') = -\eta L + 1$, we have:

$$\begin{aligned}
|J_{\mathrm{D}}(\boldsymbol{\theta}) - J_{\mathrm{D}}(\boldsymbol{\theta}')| &= \left| \frac{1-\gamma^T}{1-\gamma} \cdot \frac{1}{d_\Theta} \sum_{i=1}^{d_\Theta} f(\theta_i) - \frac{1-\gamma^T}{1-\gamma} \cdot \frac{1}{d_\Theta} \sum_{i=1}^{d_\Theta} f(\theta_i') \right| && (35)\\
&= \frac{1-\gamma^T}{1-\gamma} \cdot \frac{1}{d_\Theta} \sum_{i=1}^{d_\Theta} |f(\theta_i) - f(\theta_i')| && (36)\\
&= \frac{1-\gamma^T}{1-\gamma} \cdot \frac{L}{d_\Theta} \sum_{i=1}^{d_\Theta} |\theta_i - \theta_i'| && (37)\\
&= \frac{1-\gamma^T}{1-\gamma} L \eta && (38)\\
&= \frac{1-\gamma^T}{1-\gamma} \cdot \frac{L}{\sqrt{d_\Theta}} \|\boldsymbol{\theta} - \boldsymbol{\theta}'\|_2. && (39)
\end{aligned}$$

Thus, we have that $J_{\mathrm{D}}(\boldsymbol{\theta})$ is $\left(\frac{1-\gamma^T}{1-\gamma} \cdot \frac{L}{\sqrt{d_\Theta}} \right)$-LC. By naming $L_J = \frac{1-\gamma^T}{1-\gamma} \cdot \frac{L}{\sqrt{d_\Theta}}$, we have $L = \frac{1-\gamma}{1-\gamma^T} \sqrt{d_\Theta} L_J$. We now consider the additive noise $\Phi_{d_\Theta} = \otimes_{i=1}^{d_\Theta} \mathrm{Uni}([-\sqrt{3}\sigma, \sqrt{3}\sigma])$, i.e., the d_Θ-dimensional uniform distribution with independent components over the hypercube $[-\sqrt{3}\sigma, \sqrt{3}\sigma]^{d_\Theta}$. From Lemma D.1, we know that each dimension has variance σ^2,

consequently:

$$\mathop{\mathbb{E}}_{\boldsymbol{\epsilon}\sim\Phi_{d_\Theta}}[\|\boldsymbol{\epsilon}\|_2^2]=\sum_{i=1}^{d_\Theta}\mathop{\mathbb{E}}_{\epsilon_i\sim\mathrm{Uni}([-\sqrt{3}\sigma,\sqrt{3}\sigma])}[\epsilon_i^2]=d_\Theta\sigma^2, \tag{40}$$

thus complying with Definition 3.2. Consequently:

$$J_{\mathrm{P}}(\boldsymbol{\theta})=\mathop{\mathbb{E}}_{\boldsymbol{\epsilon}\sim\Phi_{d_\Theta}}[J_{\mathrm{D}}(\boldsymbol{\theta}+\boldsymbol{\epsilon})]=\sum_{i=1}^{d_\Theta}\mathop{\mathbb{E}}_{\epsilon_i\sim\mathrm{Uni}([-\sqrt{3}\sigma,\sqrt{3}\sigma])}[f(\theta_i+\epsilon_i)]=\sum_{i=1}^{d_\Theta}(f*\psi_\sigma)(\theta_i), \tag{41}$$

where ψ_σ is the p.d.f. of the considered uniform distribution as defined in Lemma D.1. From Lemma D.1 and observing that both J_{D} and J_{P} decompose into a sum over the d_Θ dimensions, we have for $\sqrt{3}\sigma<1/L$:

$$\boldsymbol{\theta}^*=\mathop{\arg\max}_{\boldsymbol{\theta}\in\mathbb{R}^{d_\Theta}}J_{\mathrm{D}}(\boldsymbol{\theta})=\mathbf{0}_{d_\Theta},\qquad \boldsymbol{\theta}^*_{\mathrm{P}}=\mathop{\arg\max}_{\boldsymbol{\theta}\in\mathbb{R}^{d_\Theta}}J_{\mathrm{P}}(\boldsymbol{\theta})=\frac{\sigma}{\sqrt{3}}\mathbf{1}_{d_\Theta}. \tag{42}$$

It follows that:

$$J_{\mathrm{D}}(\boldsymbol{\theta}^*)-J_{\mathrm{D}}(\boldsymbol{\theta}^*_{\mathrm{P}})=J_{\mathrm{D}}(\mathbf{0}_{d_\Theta})-J_{\mathrm{D}}\left(\frac{\sigma}{\sqrt{3}}\mathbf{1}_{d_\Theta}\right) \tag{43}$$

$$=\frac{1-\gamma^T}{1-\gamma}\frac{1}{d_\Theta}\sum_{i=1}^{d_\Theta}f(0)-f(\sigma/\sqrt{3}) \tag{44}$$

$$=\frac{1-\gamma^T}{1-\gamma}\frac{L\sigma}{2\sqrt{3}} \tag{45}$$

$$=\frac{1}{2\sqrt{3}}L_J\sqrt{d_\Theta}\sigma. \tag{46}$$

□

Theorem 5.2 (Deterministic deployment of parameters learned with AB white-noise exploration). *If the policy complies with Definition 3.2 and under Assumption 5.2:*

(i) (Uniform bound) *for every* $\boldsymbol{\theta}\in\Theta$*, it holds that:* $|J_D(\boldsymbol{\theta})-J_A(\boldsymbol{\theta})|\leqslant L\sqrt{d_{\mathcal{A}}}\sigma_A$*;*

(ii) (J_{D} upper bound) *letting* $\boldsymbol{\theta}^*_A\in\arg\max_{\boldsymbol{\theta}\in\Theta}J_A(\boldsymbol{\theta})$*, it holds that* $J^*_D-J_D(\boldsymbol{\theta}^*_A)\leqslant 2L\sqrt{d_{\mathcal{A}}}\sigma_A$*;*

(iii) (J_{D} lower bound) *there exists an MDP, a deterministic policy class* $\boldsymbol{\mu}_{\boldsymbol{\theta}}$ *fulfilling Assumption 5.1, and a noise complying with Definition 3.1, such that* $J^*_D-J_D(\boldsymbol{\theta}^*_A)\geqslant 0.28L\sqrt{d_{\mathcal{A}}}\sigma_A$*.*

Proof. From Assumption 5.2, noting that $J_{\mathrm{D}}(\boldsymbol{\theta})=J_{\mathrm{D}}(\underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}})$ we can easily derive (i):

$$|J_{\mathrm{D}}(\boldsymbol{\theta})-J_{\mathrm{A}}(\boldsymbol{\theta})|=\left|J_{\mathrm{D}}(\boldsymbol{\theta})-\mathop{\mathbb{E}}_{\underline{\boldsymbol{\epsilon}}\sim\Phi^T_{d_{\mathcal{A}}}}\left[J_{\mathrm{D}}(\underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}}+\underline{\boldsymbol{\epsilon}})\right]\right| \tag{47}$$

$$=\left|J_{\mathrm{D}}(\underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}})-\mathop{\mathbb{E}}_{\underline{\boldsymbol{\epsilon}}\sim\Phi^T_{d_{\mathcal{A}}}}\left[J_{\mathrm{D}}(\underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}}+\underline{\boldsymbol{\epsilon}})\right]\right| \tag{48}$$

$$\leqslant\mathop{\mathbb{E}}_{\underline{\boldsymbol{\epsilon}}\sim\Phi^T_{d_{\mathcal{A}}}}\left[\sum_{t=0}^{T-1}L_t\sup_{s_t\in\mathcal{S}}\|\boldsymbol{\mu}_{\boldsymbol{\theta}}(s_t)-(\boldsymbol{\mu}_{\boldsymbol{\theta}}(s_t)+\boldsymbol{\epsilon}_t)\|_2\right] \tag{49}$$

$$=\sum_{t=0}^{T-1}L_t\mathop{\mathbb{E}}_{\boldsymbol{\epsilon}\sim\Phi_{d_{\mathcal{A}}}}[\|\boldsymbol{\epsilon}\|_2^2] \tag{50}$$

$$\leqslant L\sqrt{d_{\mathcal{A}}}\sigma_{\mathrm{A}}. \tag{51}$$

For (ii), let $\boldsymbol{\theta}^*\in\arg\max_{\boldsymbol{\theta}\in\Theta}J_{\mathrm{D}}(\boldsymbol{\theta})$, we have:

$$\max_{\boldsymbol{\theta}\in\Theta}J_{\mathrm{D}}(\boldsymbol{\theta})-J_{\mathrm{D}}(\boldsymbol{\theta}^*_{\mathrm{A}})=J_{\mathrm{D}}(\boldsymbol{\theta}^*)-J_{\mathrm{D}}(\boldsymbol{\theta}^*_{\mathrm{A}})\pm J_{\mathrm{A}}(\boldsymbol{\theta}^*) \tag{52}$$

$$\leqslant J_{\mathrm{D}}(\boldsymbol{\theta}^*)-J_{\mathrm{A}}(\boldsymbol{\theta}^*)+J_{\mathrm{A}}(\boldsymbol{\theta}^*_{\mathrm{A}})-J_{\mathrm{D}}(\boldsymbol{\theta}^*_{\mathrm{A}}) \tag{53}$$

$$\leqslant 2\max_{\boldsymbol{\theta}\in\Theta}|J_{\mathrm{D}}(\boldsymbol{\theta})-J_{\mathrm{A}}(\boldsymbol{\theta})| \tag{54}$$

$$\leqslant 2L\sigma_{\mathrm{A}}\sqrt{d_{\mathcal{A}}}, \tag{55}$$

where line (53) follows from $J_{\mathrm{A}}(\boldsymbol{\theta}^*_{\mathrm{A}}) = \max_{\boldsymbol{\theta}\in\Theta} J_{\mathrm{A}}(\boldsymbol{\theta}) \geqslant J_{\mathrm{A}}(\boldsymbol{\theta}^*)$, and line (54) follows by applying twice result (i). The proof of (iii) is identical to that of Theorem 5.1 since, for the particular instance, we have enforced $\boldsymbol{\mu}_{\boldsymbol{\theta}}(s) = \boldsymbol{\theta}$ (which implies $d_{\mathcal{A}} = d_{\Theta}$) and, thus, AB exploration is equivalent to PB exploration. □

D.2. Proofs from Section 6

Lemma D.2 (Variance of $\widehat{\nabla}_{\boldsymbol{\theta}} J_{\mathrm{P}}(\boldsymbol{\theta})$ bounded)**.** *Under Assumption 4.5, the variance the PGPE estimator with batch size N is bounded for every $\boldsymbol{\theta}\in\Theta$ as:*

$$\mathbb{V}\mathrm{ar}\left[\widehat{\nabla}_{\boldsymbol{\theta}} J_{P}(\boldsymbol{\theta})\right] \leqslant \frac{R_{\max}^2 \xi_2 (1-\gamma^T)^2}{N(1-\gamma)^2} \leqslant \frac{R_{\max}^2 \xi_1^2}{N(1-\gamma)^2}.$$

with $\xi_2 \leqslant c d_{\Theta} \sigma_P^{-2}$.

Proof. We recall that the estimator employed by PGPE in its update rule is:

$$\widehat{\nabla}_{\boldsymbol{\theta}} J_{\mathrm{P}}(\boldsymbol{\theta}) = \frac{1}{N}\sum_{i=1}^{N} \nabla_{\boldsymbol{\theta}} \log \nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}_i) R(\tau_i),$$

where N is the number of parameter configuration tested (on one trajectory) at each iteration. Thus, we can compute the variance of such an estimator as:

$$\begin{aligned}
\underset{\boldsymbol{\theta}'\sim\nu_{\boldsymbol{\theta}}}{\mathbb{V}\mathrm{ar}}\left[\widehat{\nabla}_{\boldsymbol{\theta}} J_{\mathrm{P}}(\boldsymbol{\theta}')\right] &= \frac{1}{N} \underset{\boldsymbol{\theta}'\sim\nu_{\boldsymbol{\theta}}}{\mathbb{V}\mathrm{ar}}\left[\nabla_{\boldsymbol{\theta}} \log \nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}') R(\tau_1)\right] \\
&= \frac{1}{N} \underset{\boldsymbol{\theta}'\sim\nu_{\boldsymbol{\theta}}}{\mathbb{E}}\left[\left\|\nabla_{\boldsymbol{\theta}} \log \nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}')\right\|_2^2 R(\tau_1)^2\right] \\
&\leqslant \frac{R_{\max}^2 \xi_1^2 (1-\gamma^T)^2}{N(1-\gamma)^2},
\end{aligned}$$

where the last line follows form Assumption 4.5 and Lemma E.4 after having defined $\xi_2 = \mathbb{E}_{\boldsymbol{\theta}'\sim\nu_{\boldsymbol{\theta}}}\left[\left\|\nabla_{\boldsymbol{\theta}} \log \nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}')\right\|_2^2\right]$ and from the fact that, given a trajectory τ, $R(\tau)$ is defined as:

$$R(\tau) = \sum_{t=0}^{T-1} \gamma^t r(s_{\tau,t}, \mathbf{a}_{\tau,t}),$$

with $r(\mathbf{s},\mathbf{a}) \in [-R_{\max}, R_{\max}]$ for every $\mathbf{s}\in\mathcal{S}$ and $\mathbf{a}\in\mathcal{A}$. □

Lemma D.3 (Bounded J_{P} Hessian)**.** *Under Assumption 4.5 and using a hyperpolicy complying with Definition 3.2, $\forall \boldsymbol{\theta}\in\Theta$ it holds that:*

$$\left\|\nabla^2_{\boldsymbol{\theta}} J_{P}(\boldsymbol{\theta})\right\|_2 \leqslant L_{2,P} \frac{R_{\max}(1-\gamma^T)}{1-\gamma}(\xi_2 + \xi_3),$$

where $\xi_2 \leqslant c d_{\Theta} \sigma_P^{-2}$ and $\xi_3 \leqslant c \sigma_P^{-2}$. Furthermore, under Assumptions 4.1, 4.3, 4.2, and 4.4, and using a hyperpolicy complying with Definition 3.2, $\forall \boldsymbol{\theta}\in\Theta$ it holds that:

$$\left\|\nabla^2_{\boldsymbol{\theta}} J_{P}(\boldsymbol{\theta})\right\|_2 \leqslant L_2,$$

where L_2 is bounded as in Lemma E.2.

Proof. The performance index J_{P} of a hyperpolicy $\nu_{\boldsymbol{\theta}}$ can be seen as the expectation over the sampling of a parameter configuration $\boldsymbol{\theta}'$ from the hyperpolicy $\nu_{\boldsymbol{\theta}}$, or as the perturbation according to the realization $\boldsymbol{\epsilon}$ of a sub-gaussian noise σ_{P} of the parameter configuration of the deterministic policy $\boldsymbol{\mu}_{\boldsymbol{\theta}}$.

Using the first characterization we can write:

$$J_{\mathrm{P}}(\boldsymbol{\theta}) = \underset{\boldsymbol{\theta}'\sim\nu_{\boldsymbol{\theta}}}{\mathbb{E}}\left[J_{\mathrm{D}}(\boldsymbol{\theta}')\right]. \tag{56}$$

Equivalently, we can write:

$$J_{\mathrm{P}}(\boldsymbol{\theta}) = \underset{\boldsymbol{\epsilon}\sim\Phi}{\mathbb{E}}\left[J_{\mathrm{D}}(\boldsymbol{\theta}+\boldsymbol{\epsilon})\right]. \tag{57}$$

By using the latter, we have that:

$$\begin{aligned}\left\|\nabla^2_{\boldsymbol{\theta}} J_{\mathrm{P}}(\boldsymbol{\theta})\right\|_2 &= \left\|\nabla^2_{\boldsymbol{\theta}} \mathop{\mathbb{E}}_{\boldsymbol{\epsilon}\sim g}\left[J_{\mathrm{D}}(\boldsymbol{\theta}+\boldsymbol{\epsilon})\right]\right\|_2 \\ &= \left\|\mathop{\mathbb{E}}_{\boldsymbol{\epsilon}\sim g}\left[\nabla^2_{\boldsymbol{\theta}} J_{\mathrm{D}}(\boldsymbol{\theta}+\boldsymbol{\epsilon})\right]\right\|_2 \\ &\leqslant \mathop{\mathbb{E}}_{\boldsymbol{\epsilon}\sim g}\left[\left\|\nabla^2_{\boldsymbol{\theta}} J_{\mathrm{D}}(\boldsymbol{\theta}+\boldsymbol{\epsilon})\right\|_2\right] \\ &\leqslant L_2, \end{aligned} \tag{58}$$

where the last inequality simply follows from Assumption E.1.

By using Equation (56), instead, we have the following:

$$\begin{aligned}\nabla^2_{\boldsymbol{\theta}} J_{\mathrm{P}}(\boldsymbol{\theta}) &= \nabla^2_{\boldsymbol{\theta}} \mathop{\mathbb{E}}_{\boldsymbol{\theta}'\sim\nu_{\boldsymbol{\theta}}}\left[J_{\mathrm{D}}(\boldsymbol{\theta}')\right] \\ &= \int \nabla^2_{\boldsymbol{\theta}}\left(\nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}') J_{\mathrm{D}}(\boldsymbol{\theta}')\right) \mathrm{d}\boldsymbol{\theta}' \\ &= \int \nabla_{\boldsymbol{\theta}}\left(\nabla_{\boldsymbol{\theta}}\nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}') J_{\mathrm{D}}(\boldsymbol{\theta}') + \nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}')\nabla_{\boldsymbol{\theta}} J_{\mathrm{D}}(\boldsymbol{\theta}')\right) \mathrm{d}\boldsymbol{\theta}' \\ &= \int \nabla_{\boldsymbol{\theta}}\left(\nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}')\left(\nabla_{\boldsymbol{\theta}}\log\nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}') J_{\mathrm{D}}(\boldsymbol{\theta}')\right)\right) \mathrm{d}\boldsymbol{\theta}' \\ &= \int \nabla_{\boldsymbol{\theta}}\nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}')\nabla_{\boldsymbol{\theta}}\log\nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}') J_{\mathrm{D}}(\boldsymbol{\theta}') + \nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}')\left(\nabla^2_{\boldsymbol{\theta}}\log\nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}') J_{\mathrm{D}}(\boldsymbol{\theta}') + \nabla_{\boldsymbol{\theta}}\log\nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}')\nabla_{\boldsymbol{\theta}} J_{\mathrm{D}}(\boldsymbol{\theta}')\right) \mathrm{d}\boldsymbol{\theta}' \\ &= \int \nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}')\left(\nabla_{\boldsymbol{\theta}}\log\nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}')\nabla_{\boldsymbol{\theta}}\log\nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}')^\top J_{\mathrm{D}}(\boldsymbol{\theta}') + \nabla^2_{\boldsymbol{\theta}}\log\nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}') J_{\mathrm{D}}(\boldsymbol{\theta}')\right) \mathrm{d}\boldsymbol{\theta}' \\ &= \mathop{\mathbb{E}}_{\boldsymbol{\theta}'\sim\nu_{\boldsymbol{\theta}}}\left[\left(\nabla_{\boldsymbol{\theta}}\log\nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}')\nabla_{\boldsymbol{\theta}}\log\nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}')^\top + \nabla^2_{\boldsymbol{\theta}}\log\nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}')\right) J_{\mathrm{D}}(\boldsymbol{\theta}')\right]. \end{aligned}$$

Now, given the previous argument, it follows that:

$$\begin{aligned}\left\|\nabla^2_{\boldsymbol{\theta}} J_{\mathrm{P}}(\boldsymbol{\theta})\right\|_2 &= \left\|\mathop{\mathbb{E}}_{\boldsymbol{\theta}'\sim\nu_{\boldsymbol{\theta}}}\left[\left(\nabla_{\boldsymbol{\theta}}\log\nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}')\nabla_{\boldsymbol{\theta}}\log\nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}')^\top + \nabla^2_{\boldsymbol{\theta}}\log\nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}')\right) J_{\mathrm{D}}(\boldsymbol{\theta}')\right]\right\|_2 \\ &\leqslant \mathop{\mathbb{E}}_{\boldsymbol{\theta}'\sim\nu_{\boldsymbol{\theta}}}\left[\left\|\nabla_{\boldsymbol{\theta}}\log\nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}')\right\|_2^2 \left|J_{\mathrm{D}}(\boldsymbol{\theta}')\right| + \left\|\nabla^2_{\boldsymbol{\theta}}\log\nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}')\right\|_2 \left|J_{\mathrm{D}}(\boldsymbol{\theta}')\right|\right] \\ &\leqslant \frac{R_{\max}(1-\gamma^T)}{1-\gamma}\left(\xi_2+\xi_3\right). \end{aligned} \tag{59}$$

We employ Lemma E.4 to bound ξ_2 and ξ_3. □

Theorem D.4 (Global convergence of PGPE - Fixed σ_{P}). *Under Assumptions 6.1 (with $J_\dagger = J_P$), 4.1, 4.3, 4.5, with a suitable constant step size, to guarantee $J_D^* - \mathbb{E}[J_D(\boldsymbol{\theta}_K)] \leqslant \epsilon + \beta + 3L_P\sqrt{d_\Theta}\sigma_P$, where $3L_P\sqrt{d_\Theta}\sigma_P = O(\sqrt{d_\Theta}\sigma_P(1-\gamma)^{-2})$ the sample complexity of PGPE is at most:*

$$NK = \widetilde{O}\left(\frac{\alpha^4 d_\Theta^2}{\sigma_P^4(1-\gamma)^4\epsilon^3}\right). \tag{60}$$

Furthermore, under Assumptions 4.2 and 4.4, the same guarantee is obtained with a sample complexity at most:

$$NK = \widetilde{O}\left(\frac{\alpha^4 d_\Theta}{\sigma_P^2(1-\gamma)^5\epsilon^3}\right). \tag{61}$$

Proof. We first apply Theorem F.1 with $J_\dagger = J_{\mathrm{P}}$, recalling that the assumptions enforced in the statement entail those of Theorem F.1:

$$J_{\mathrm{P}}{}^* - \mathbb{E}[J_{\mathrm{P}}(\boldsymbol{\theta}_K)] \leqslant \epsilon + \beta \qquad \text{with} \quad NK = \frac{16\alpha^2 L_{2,\mathrm{P}} V_{\mathrm{P}}}{\epsilon^3}\log\frac{\max\{0, J_{\mathrm{P}}{}^* - J_{\mathrm{P}}(\boldsymbol{\theta}_0) - \beta\}}{\epsilon^3}. \tag{62}$$

By Theorem 5.1 (i) and (ii), we have that:

$$J_{\mathrm{D}}^* - \mathbb{E}\left[J_{\mathrm{D}}(\boldsymbol{\theta}_K)\right] = \left(J_{\mathrm{D}}^* - J_{\mathrm{P}}{}^*\right) + \mathbb{E}\left[J_{\mathrm{P}}(\boldsymbol{\theta}_K) - J_{\mathrm{D}}(\boldsymbol{\theta}_K)\right] + J_{\mathrm{P}}{}^* - \mathbb{E}[J_{\mathrm{P}}(\boldsymbol{\theta}_K)] \leqslant J_{\mathrm{P}}{}^* - \mathbb{E}[J_{\mathrm{P}}(\boldsymbol{\theta}_K)] + 3L_J\sqrt{d_\Theta}\sigma_{\mathrm{P}}. \tag{63}$$

After renaming $L_P := L_J$ for the sake of exposition, the result follows by replacing in the sample complexity NK the bounds on L_P, $L_{2,\mathrm{P}}$, and V_P from Table 2 under the two set of assumptions and retaining only the desired dependences with the Big-$\widetilde{O}$ notation.

□

Theorem D.5 (**Global convergence of PGPE - ϵ-adaptive σ_P**). *Under Assumptions 6.1 (with $J_\dagger = J_P$), 4.1, 4.3, 4.5, with a suitable constant step size and $\sigma_P = \frac{\epsilon}{6L_P\sqrt{d_\Theta}} = O(\epsilon(1-\gamma)^2 d_\Theta^{-1/2})$, to guarantee $J_D^* - \mathbb{E}[J_D(\boldsymbol{\theta}_K)] \leqslant \epsilon + \beta$ the sample complexity of PGPE is at most:*

$$NK = \widetilde{O}\left(\frac{\alpha^4 d_\Theta^4}{(1-\gamma)^{12}\epsilon^7}\right). \tag{64}$$

Furthermore, under Assumptions 4.2 and 4.4, the same guarantee is obtained with a sample complexity at most:

$$NK = \widetilde{O}\left(\frac{\alpha^4 d_\Theta^2}{(1-\gamma)^{9}\epsilon^5}\right). \tag{65}$$

Proof. We apply Theorem D.4 with $\epsilon \leftarrow \epsilon/2$ and set σ_P so that:

$$3L_J\sqrt{d_\Theta}\sigma_\mathrm{P} = \frac{\epsilon}{2} \Longrightarrow \sigma_\mathrm{P} = \frac{\epsilon}{6L_J\sqrt{d_\Theta}}. \tag{66}$$

After renaming $L_P := L_J$ for the sake of exposition, the result follows substituting this value in the sample complexity and bounding the constant L_P as in Table 2. □

Lemma D.6 (Variance of $\widehat{\nabla}_{\boldsymbol{\theta}} J_\mathrm{A}(\boldsymbol{\theta})$ bounded). *Under Assumptions 4.3 and 4.5, the variance the GPOMDP estimator with batch size N is bounded for every $\boldsymbol{\theta} \in \Theta$ as:*

$$\mathbb{V}\mathrm{ar}\left[\widehat{\nabla}_{\boldsymbol{\theta}} J_\mathrm{A}(\boldsymbol{\theta})\right] \leqslant \frac{R_{\max}^2 \xi_2 (1-\gamma^T)}{N(1-\gamma)^3} \leqslant \frac{R_{\max}^2 \xi_2}{N(1-\gamma)^3}.$$

with $\xi_2 \leqslant c d_\mathcal{A} \sigma_A^{-2} L_\mu^2$.

Proof. It follows from Lemma 29 of Papini et al. (2022) and from the application of Lemma E.3 to bound ξ_2. □

Lemma D.7 (Bounded J_A Hessian). *Under Assumptions 4.3, 4.4, and 4.5 $\forall \boldsymbol{\theta} \in \Theta$ it holds that:*

$$\left\|\nabla^2_{\boldsymbol{\theta}} J_\mathrm{A}(\boldsymbol{\theta})\right\|_2 \leqslant \frac{R_{\max}\left(1-\gamma^{T+1}\right)}{(1-\gamma)^2}(\upsilon_2 + \upsilon_3),$$

where $\upsilon_2 \leqslant c d_\mathcal{A} \sigma - 2_A L_\mu^2$ and $\upsilon_3 \leqslant c\sigma_A^{-2} L_\mu^2 + c\sqrt{d_\mathcal{A}}\sigma_A^{-1} L_{2,\mu}$. Furthermore, under Assumptions 4.1, 4.3, 4.2, and 4.4, $\forall \boldsymbol{\theta} \in \Theta$ it holds that:

$$\left\|\nabla^2_{\boldsymbol{\theta}} J_\mathrm{A}(\boldsymbol{\theta})\right\|_2 \leqslant L_2,$$

where L_2 is bounded in Lemma E.2.

Proof. Under Assumption 4.5, by a slight modification of the proof of Lemma 4.4 by Yuan et al. (2022) (in which we consider a finite horizon T), it follows that:

$$\left\|\nabla^2_{\boldsymbol{\theta}} J_\mathrm{A}(\boldsymbol{\theta})\right\|_2 \leqslant \frac{R_{\max}\left(1-(T+1)\gamma^T + T\gamma^{T+1}\right)}{(1-\gamma)^2}(\upsilon_1 + \upsilon_2) \leqslant \frac{R_{\max}\left(1-\gamma^T\right)}{(1-\gamma)^2}(\upsilon_1 + \upsilon_2).$$

As in the proof of Theorem E.1, we introduce the following convenient expression for the trajectory density function having fixed a sequence of noise $\underline{\boldsymbol{\epsilon}} \sim \Phi_{d_\mathcal{A}}^T$:

$$p_\mathrm{D}(\tau; \underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}} + \underline{\boldsymbol{\epsilon}}) = \rho_0(s_{\tau,0}) \prod_{t=0}^{T-1} p(s_{\tau,t+1} | s_{\tau,t}, \boldsymbol{\mu}_{\boldsymbol{\theta}}(s_{\tau,t}) + \boldsymbol{\epsilon}_t).$$

This allows us to express the function $J_\mathrm{A}(\boldsymbol{\theta})$, for a generic $\boldsymbol{\theta} \in \Theta$, as:

$$J_\mathrm{A}(\boldsymbol{\theta}) = \mathop{\mathbb{E}}_{\underline{\boldsymbol{\epsilon}} \sim \Phi_{d_\mathcal{A}}^T} \left[\int_\tau p_\mathrm{D}(\tau; \underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}} + \underline{\boldsymbol{\epsilon}}) \sum_{t=0}^{T-1} \gamma^t r(s_{\tau,t}, \boldsymbol{\mu}_{\boldsymbol{\theta}}(s_{\tau,t}) + \boldsymbol{\epsilon}_t) \mathrm{d}\tau\right].$$

With a slight abuse of notation, let us call $J_{\mathrm{D}}(\underline{\boldsymbol{\mu}_{\boldsymbol{\theta}}}+\underline{\boldsymbol{\epsilon}})$ the following quantity:

$$J_{\mathrm{D}}(\underline{\boldsymbol{\mu}_{\boldsymbol{\theta}}}+\underline{\boldsymbol{\epsilon}}) := \int_{\tau} p_{\mathrm{D}}(\tau; \underline{\boldsymbol{\mu}_{\boldsymbol{\theta}}}+\underline{\boldsymbol{\epsilon}}) \sum_{t=0}^{T-1} \gamma^t r(s_{\tau,t}, \boldsymbol{\mu}_{\boldsymbol{\theta}}(s_{\tau,t})+\boldsymbol{\epsilon}_t)\mathrm{d}\tau.$$

Now, considering the norm of the hessian w.r.t. $\boldsymbol{\theta}$ of J_{A}, we have that:

$$\left\|\nabla^2_{\boldsymbol{\theta}} J_{\mathrm{A}}(\boldsymbol{\theta})\right\|_2 \leqslant \mathop{\mathbb{E}}_{\underline{\boldsymbol{\epsilon}}\sim\Phi^T_{d_{\mathcal{A}}}} \left[\left\|\nabla^2_{\boldsymbol{\theta}} J_{\mathrm{D}}(\underline{\boldsymbol{\mu}_{\boldsymbol{\theta}}}+\underline{\boldsymbol{\epsilon}})\right\|_2\right] \leqslant L_2,$$

which follows from Assumptions E.1. □

Theorem D.8 (**Global convergence of GPOMDP - Fixed σ_{A}**). *Under Assumptions 6.1 (with $J_\dagger = J_A$), 4.1, 4.3, 4.4, 4.5, with a suitable constant step size, to guarantee $J_D^* - \mathbb{E}[J_D(\boldsymbol{\theta}_K)] \leqslant \epsilon + \beta + 3L_A\sqrt{d_{\mathcal{A}}}\sigma_A$, where $3L_A\sqrt{d_{\mathcal{A}}}\sigma_A = O(\sqrt{d_{\mathcal{A}}}\sigma_A(1-\gamma)^{-2})$ the sample complexity of GPOMDP is at most:*

$$NK = \widetilde{O}\left(\frac{\alpha^4 d_{\mathcal{A}}^2}{\sigma_A^4(1-\gamma)^5\epsilon^3}\right). \tag{67}$$

Furthermore, under Assumption 4.2, the same guarantee is obtained with a sample complexity at most:

$$NK = \widetilde{O}\left(\frac{\alpha^4 d_{\mathcal{A}}}{\sigma_A^2(1-\gamma)^6\epsilon^3}\right). \tag{68}$$

Proof. We first apply Theorem F.1 with $J_\dagger = J_{\mathrm{A}}$, recalling that the assumptions enforced in the statement entail those of Theorem F.1:

$$J_{\mathrm{A}}{}^* - \mathbb{E}[J_{\mathrm{A}}(\boldsymbol{\theta}_K)] \leqslant \epsilon + \beta \qquad \text{with} \quad NK = \frac{16\alpha^2 L_{2,\mathrm{A}} V_{\mathrm{A}}}{\epsilon^3}\log\frac{\max\{0, J_{\mathrm{A}}{}^* - J_{\mathrm{A}}(\boldsymbol{\theta}_0) - \beta\}}{\epsilon^3}. \tag{69}$$

By Theorem 5.2 (i) and (ii), we have that:

$$J_{\mathrm{D}}^* - \mathbb{E}\left[J_{\mathrm{D}}(\boldsymbol{\theta}_K)\right] = (J_{\mathrm{D}}^* - J_{\mathrm{A}}{}^*) + \mathbb{E}\left[J_{\mathrm{A}}(\boldsymbol{\theta}_K) - J_{\mathrm{D}}(\boldsymbol{\theta}_K)\right] + J_{\mathrm{A}}{}^* - \mathbb{E}[J_{\mathrm{A}}(\boldsymbol{\theta}_K)] \leqslant J_{\mathrm{A}}{}^* - \mathbb{E}[J_{\mathrm{A}}(\boldsymbol{\theta}_K)] + 3L\sqrt{d_{\mathcal{A}}}\sigma_{\mathrm{A}}. \tag{70}$$

After renaming $L_A := L$ for the sake of exposition, the result follows by replacing in the sample complexity NK the bounds on L_A, $L_{2,\mathrm{A}}$, and V_{A} from Table 2 under the two set of assumptions and retaining only the desired dependences with the Big-$\widetilde{O}$ notation. □

Theorem D.9 (**Global convergence of GPOMDP - ϵ-adaptive σ_{P}**). *Under Assumptions 6.1 (with $J_\dagger = J_{\mathrm{A}}$), 4.1, 4.3, 4.4, 4.5, with a suitable constant step size and setting $\sigma_A = \frac{\epsilon}{6L_A\sqrt{d_{\mathcal{A}}}} = O(\epsilon(1-\gamma)^2 d_{\mathcal{A}}^{-1/2})$, to guarantee $J_D^* - \mathbb{E}[J_D(\boldsymbol{\theta}_K)] \leqslant \epsilon + \beta$ the sample complexity of GPOMDP is at most:*

$$NK = \widetilde{O}\left(\frac{\alpha^4 d_{\mathcal{A}}^4}{(1-\gamma)^{13}\epsilon^7}\right). \tag{71}$$

Furthermore, under Assumption 4.2, the same guarantee is obtained with a sample complexity at most:

$$NK = \widetilde{O}\left(\frac{\alpha^4 d_{\mathcal{A}}^2}{(1-\gamma)^{10}\epsilon^5}\right). \tag{72}$$

Proof. We apply Theorem D.8 with $\epsilon \leftarrow \epsilon/2$ and set σ_{A} so that:

$$3L\sqrt{d_{\mathcal{A}}}\sigma_{\mathrm{A}} = \frac{\epsilon}{2} \Longrightarrow \sigma_{\mathrm{A}} = \frac{\epsilon}{6L\sqrt{d_{\mathcal{A}}}}. \tag{73}$$

After renaming $L_A := L$ for the sake of exposition, the result follows substituting this value in the sample complexity and bounding the constant L_A as in Table 2. □

D.3. Proofs from Section 7.1

Lemma D.10. *Under Assumptions 4.1, 4.3, 4.2, 4.4, 7.1, and using a hyperpolicy complying with Definition 3.2, $\forall \boldsymbol{\theta} \in \Theta$ it holds that:*

$$J_D^* - J_D(\boldsymbol{\theta}) \leqslant \alpha_D \|\nabla_{\boldsymbol{\theta}} J_P(\boldsymbol{\theta})\|_2 + \beta_D + \alpha_D L_2 \sigma_P \sqrt{d_\Theta}.$$

Proof. We start by observing that

$$J_{\mathrm{P}}(\boldsymbol{\theta}) = \mathop{\mathbb{E}}_{\boldsymbol{\theta}' \sim \nu_{\boldsymbol{\theta}}} \left[J_{\mathrm{D}}(\boldsymbol{\theta}') \right] = \mathop{\mathbb{E}}_{\boldsymbol{\epsilon} \sim \Phi} \left[J_{\mathrm{D}}(\boldsymbol{\theta} + \boldsymbol{\epsilon}) \right].$$

From this fact, we can proceed as follows:

$$\begin{aligned} \nabla_{\boldsymbol{\theta}} J_{\mathrm{P}}(\boldsymbol{\theta}) &= \nabla_{\boldsymbol{\theta}} \mathop{\mathbb{E}}_{\boldsymbol{\epsilon} \sim \Phi} \left[J_{\mathrm{D}}(\boldsymbol{\theta} + \boldsymbol{\epsilon}) \right] \\ &= \mathop{\mathbb{E}}_{\boldsymbol{\epsilon} \sim \Phi} \left[\nabla_{\boldsymbol{\theta}} J_{\mathrm{D}}(\boldsymbol{\theta} + \boldsymbol{\epsilon}) \right]. \end{aligned}$$

For what follows, we define $\tilde{\boldsymbol{\theta}}_{\boldsymbol{\epsilon}}$ as an intermediate parameter configuration between $\boldsymbol{\theta}$ and $\boldsymbol{\theta} + \boldsymbol{\epsilon}$. More formally, let $\lambda \in [0,1]$, then $\tilde{\boldsymbol{\theta}}_{\boldsymbol{\epsilon}} = \lambda \boldsymbol{\theta} + (1-\lambda)(\boldsymbol{\theta} + \boldsymbol{\epsilon})$. We can proceed by rewriting the term $\nabla_{\boldsymbol{\theta}} J_{\mathrm{D}}(\boldsymbol{\theta} + \boldsymbol{\epsilon})$ exploiting the first-order Taylor expansion centered in $\boldsymbol{\epsilon}$: there exists a $\lambda \in [0,1]$ such that

$$\begin{aligned} \mathop{\mathbb{E}}_{\boldsymbol{\epsilon} \sim g} \left[\nabla_{\boldsymbol{\theta}} J_{\mathrm{D}}(\boldsymbol{\theta} + \boldsymbol{\epsilon}) \right] &= \mathop{\mathbb{E}}_{\boldsymbol{\epsilon} \sim g} \left[\nabla_{\boldsymbol{\theta}} J_{\mathrm{D}}(\boldsymbol{\theta}) + \boldsymbol{\epsilon}^T \nabla^2_{\boldsymbol{\theta}} J_{\mathrm{D}}(\tilde{\boldsymbol{\theta}}_{\boldsymbol{\epsilon}}) \right] \\ &= \nabla_{\boldsymbol{\theta}} J_{\mathrm{D}}(\boldsymbol{\theta}) + \mathop{\mathbb{E}}_{\boldsymbol{\epsilon} \sim \Phi} \left[\boldsymbol{\epsilon}^T \nabla^2_{\boldsymbol{\theta}} J_{\mathrm{D}}(\tilde{\boldsymbol{\theta}}_{\boldsymbol{\epsilon}}) \right]. \end{aligned}$$

Now, we can consider the 2-norm of the gradient:

$$\begin{aligned} \|\nabla_{\boldsymbol{\theta}} J_{\mathrm{P}}(\boldsymbol{\theta})\| = \left\| \nabla_{\boldsymbol{\theta}} J_{\mathrm{D}}(\boldsymbol{\theta}) + \mathop{\mathbb{E}}_{\boldsymbol{\epsilon} \sim \Phi} \left[\boldsymbol{\epsilon}^T \nabla^2_{\boldsymbol{\theta}} J_{\mathrm{D}}(\tilde{\boldsymbol{\theta}}_{\boldsymbol{\epsilon}}) \right] \right\|_2 &\geqslant \|\nabla_{\boldsymbol{\theta}} J_{\mathrm{D}}(\boldsymbol{\theta})\|_2 - \left\| \mathop{\mathbb{E}}_{\boldsymbol{\epsilon} \sim \Phi} \left[\boldsymbol{\epsilon}^T \nabla^2_{\boldsymbol{\theta}} J_{\mathrm{D}}(\tilde{\boldsymbol{\theta}}_{\boldsymbol{\epsilon}}) \right] \right\|_2 \\ &\geqslant \|\nabla_{\boldsymbol{\theta}} J_{\mathrm{D}}(\boldsymbol{\theta})\|_2 - L_2 \mathop{\mathbb{E}}_{\boldsymbol{\epsilon} \sim \Phi} \left[\|\boldsymbol{\epsilon}\|_2 \right] \qquad (74) \\ &\geqslant \frac{1}{\alpha_{\mathrm{D}}} \left(J^*_{\mathrm{D}} - J_{\mathrm{D}}(\boldsymbol{\theta}) \right) - \frac{\beta_{\mathrm{D}}}{\alpha_{\mathrm{D}}} - L_2 \mathop{\mathbb{E}}_{\boldsymbol{\epsilon} \sim \Phi} \left[\|\boldsymbol{\epsilon}\|_2 \right] \qquad (75) \\ &\geqslant \frac{1}{\alpha_{\mathrm{D}}} \left(J^*_{\mathrm{D}} - J_{\mathrm{D}}(\boldsymbol{\theta}) \right) - \frac{\beta_{\mathrm{D}}}{\alpha_{\mathrm{D}}} - L_2 \sigma_{\mathrm{P}} \sqrt{d_\Theta}, \end{aligned}$$

where Equation (74) follows from Assumption E.1, and Equation (75) follows from Assumption 6.1. Thus, it simply follows that:

$$J^*_{\mathrm{D}} - J_{\mathrm{D}}(\boldsymbol{\theta}) \leqslant \alpha_{\mathrm{D}} \|\nabla_{\boldsymbol{\theta}} J_{\mathrm{P}}(\boldsymbol{\theta})\|_2 + \beta_{\mathrm{D}} + \alpha_{\mathrm{D}} L_2 \sigma_{\mathrm{P}} \sqrt{d_\Theta}.$$

□

Theorem 7.1 (Inherited weak gradient domination for J_{P})**.** *Under Assumptions 4.1, 4.2, 4.3, 4.4, 7.1, for every $\boldsymbol{\theta} \in \Theta$:*

$$J_P{}^* - J_P(\boldsymbol{\theta}) \leqslant \alpha_D \|\nabla_{\boldsymbol{\theta}} J_P(\boldsymbol{\theta})\|_2 + \beta_D + (\alpha_D L_2 + L_P) \sigma_P \sqrt{d_\Theta},$$

where $L_2 = O((1-\gamma)^{-3})$ (full expression in Lemma E.2).

Proof. We recall that under the assumptions in the statement, the results of Lemma D.10 and of Theorem 5.1 hold In particular, we need the result from Theorem 5.1, saying that $\forall \boldsymbol{\theta} \in \Theta$ it holds that

$$J_{\mathrm{P}}(\boldsymbol{\theta}) - L_J \sigma_{\mathrm{P}} \sqrt{d_\Theta} \leqslant J_{\mathrm{D}}(\boldsymbol{\theta}) \leqslant J_{\mathrm{P}}(\boldsymbol{\theta}) + L_J \sigma_{\mathrm{P}} \sqrt{d_\Theta}. \qquad (76)$$

Thus, using the result of Lemma D.10, we need to work on the left-hand side of the following inequality:

$$J^*_{\mathrm{D}} - J_{\mathrm{D}}(\boldsymbol{\theta}) \leqslant \alpha_{\mathrm{D}} \|\nabla_{\boldsymbol{\theta}} J_{\mathrm{P}}(\boldsymbol{\theta})\|_2 + \beta_{\mathrm{D}} + \alpha_{\mathrm{D}} L_2 \sigma_{\mathrm{P}} \sqrt{d_\Theta}.$$

Moreover, by definition of J_{P}, we have that $J^*_{\mathrm{D}} \geqslant J_{\mathrm{P}}{}^*$. Thus, it holds that:

$$\begin{aligned} J^*_{\mathrm{D}} - J_{\mathrm{D}}(\boldsymbol{\theta}) &\geqslant J_{\mathrm{P}}{}^* - J_{\mathrm{P}}(\boldsymbol{\theta}) \\ &\geqslant J_{\mathrm{P}}{}^* - J_{\mathrm{P}}(\boldsymbol{\theta}) - L_J \sqrt{d_\Theta}, \end{aligned}$$

where the last line follows from Line (76). We rename $L_P := L_J$ in the statement.

□

Lemma D.11. *Under Assumptions 7.1, 4.1, 4.3, 4.2, 4.4, using a policy complying with Definition 3.2, $\forall \boldsymbol{\theta} \in \Theta$, it holds that:*

$$J^*_D - J_D(\boldsymbol{\theta}) \leqslant \alpha_D \|\nabla_{\boldsymbol{\theta}} J_A(\boldsymbol{\theta})\|_2 + \beta_D + \alpha_D \psi \sigma_A \sqrt{d_{\mathcal{A}}},$$

where

$$\psi = L_\mu \left(\frac{L_p^2 R_{\max} \gamma}{(1-\gamma)^4} + \frac{(L_r L_p + R_{\max} L_{2,p} + L_p L_r \gamma)}{(1-\gamma)^2} + \frac{L_{2,r}}{1-\gamma} \right) (1-\gamma^T).$$

Proof. As in the proof of Theorem E.1, we introduce the following convenient expression for the trajectory density function having fixed a sequence of noise $\underline{\boldsymbol{\epsilon}} \sim \Phi_{d_{\mathcal{A}}}^T$:

$$p_{\mathrm{D}}(\tau; \underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}} + \underline{\boldsymbol{\epsilon}}) = \rho_0(s_{\tau,0}) \prod_{t=0}^{T-1} p(s_{\tau,t+1} | s_{\tau,t}, \boldsymbol{\mu}_{\boldsymbol{\theta}}(s_{\tau,t}) + \boldsymbol{\epsilon}_t). \tag{77}$$

Also in this case, we denote with $p_{\mathrm{D}}(\tau_{0:l}; \underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}} + \underline{\boldsymbol{\epsilon}})$ the density function of a trajectory prefix of length l:

$$p_{\mathrm{D}}(\tau_{0:l}; \underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}} + \underline{\boldsymbol{\epsilon}}) = \rho_0(s_{\tau,0}) \prod_{t=0}^{l-1} p(s_{\tau,t+1} | s_{\tau,t}, \boldsymbol{\mu}_{\boldsymbol{\theta}}(s_{\tau,t}) + \boldsymbol{\epsilon}_t). \tag{78}$$

From the proof of Proposition E.1, considering a generic parametric configuration $\boldsymbol{\theta} \in \Theta$, we can write the AB performance index $J_{\mathrm{A}}(\boldsymbol{\theta})$ as:

$$\begin{aligned} J_{\mathrm{A}}(\boldsymbol{\theta}) &= \mathop{\mathbb{E}}_{\underline{\boldsymbol{\epsilon}} \sim \Phi_{d_{\mathcal{A}}}^T} \left[\int_\tau p_{\mathrm{D}}(\tau; \underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}} + \underline{\boldsymbol{\epsilon}}) \sum_{t=0}^{T-1} \gamma^t r(s_t, \boldsymbol{\mu}_{\boldsymbol{\theta}}(s_t) + \boldsymbol{\epsilon}_t) \mathrm{d}\tau \right] \\ &= \mathop{\mathbb{E}}_{\underline{\boldsymbol{\epsilon}} \sim \Phi_{d_{\mathcal{A}}}^T} \left[\underbrace{\sum_{t=0}^{T-1} \int_{\tau_{0:t}} p_{\mathrm{D}}(\tau_{0:t}; \underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}} + \underline{\boldsymbol{\epsilon}}) \gamma^t r(s_t, \boldsymbol{\mu}_{\boldsymbol{\theta}}(s_t) + \boldsymbol{\epsilon}_t) \mathrm{d}\tau_{0:t}}_{=: f(\underline{\boldsymbol{\epsilon}})} \right], \end{aligned}$$

moreover, by using the Taylor expansion centered in $\underline{\boldsymbol{\epsilon}} = \underline{\mathbf{0}}$, for $\widetilde{\underline{\boldsymbol{\epsilon}}} = x \underline{\boldsymbol{\epsilon}}$ (for some $x \in [0,1]$) the following holds:

$$\begin{aligned} J_{\mathrm{A}}(\boldsymbol{\theta}) &= \mathop{\mathbb{E}}_{\underline{\boldsymbol{\epsilon}} \sim \Phi_{d_{\mathcal{A}}}^T} \left[f(\underline{\mathbf{0}}) + \sum_{t=0}^{T-1} \boldsymbol{\epsilon}_t^\top \nabla_{\boldsymbol{\epsilon}_t} f(\underline{\boldsymbol{\epsilon}})|_{\underline{\boldsymbol{\epsilon}} = \widetilde{\underline{\boldsymbol{\epsilon}}}} \right] \\ &= J_{\mathrm{D}}(\boldsymbol{\theta}) + \sum_{t=0}^{T-1} \mathop{\mathbb{E}}_{\underline{\boldsymbol{\epsilon}} \sim \Phi_{d_{\mathcal{A}}}^T} [\boldsymbol{\epsilon}_t^\top \nabla_{\boldsymbol{\epsilon}_t} f(\underline{\boldsymbol{\epsilon}})|_{\underline{\boldsymbol{\epsilon}} = \widetilde{\underline{\boldsymbol{\epsilon}}}}]. \end{aligned}$$

Here, we are interested in the gradient of J_{A}:

$$\begin{aligned} \nabla_{\boldsymbol{\theta}} J_{\mathrm{A}}(\boldsymbol{\theta}) &= \nabla_{\boldsymbol{\theta}} J_{\mathrm{D}}(\boldsymbol{\theta}) + \sum_{t=0}^{T-1} \nabla_{\boldsymbol{\theta}} \mathop{\mathbb{E}}_{\underline{\boldsymbol{\epsilon}} \sim \Phi_{d_{\mathcal{A}}}^T} [\boldsymbol{\epsilon}_t^\top \nabla_{\boldsymbol{\epsilon}_t} f(\underline{\boldsymbol{\epsilon}})|_{\underline{\boldsymbol{\epsilon}} = \widetilde{\underline{\boldsymbol{\epsilon}}}}] \\ &= \nabla_{\boldsymbol{\theta}} J_{\mathrm{D}}(\boldsymbol{\theta}) + \sum_{t=0}^{T-1} \mathop{\mathbb{E}}_{\underline{\boldsymbol{\epsilon}} \sim \Phi_{d_{\mathcal{A}}}^T} [\boldsymbol{\epsilon}_t^\top \nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\epsilon}_t} f(\underline{\boldsymbol{\epsilon}})|_{\underline{\boldsymbol{\epsilon}} = \widetilde{\underline{\boldsymbol{\epsilon}}}}]. \end{aligned}$$

Now, considering the norm of the gradient we have:

$$\begin{aligned} \|\nabla_{\boldsymbol{\theta}} J_{\mathrm{A}}(\boldsymbol{\theta})\|_2 &\geqslant \|\nabla_{\boldsymbol{\theta}} J_{\mathrm{D}}(\boldsymbol{\theta})\|_2 - \left\| \sum_{t=0}^{T-1} \mathop{\mathbb{E}}_{\underline{\boldsymbol{\epsilon}} \sim \Phi_{d_{\mathcal{A}}}^T} [\boldsymbol{\epsilon}_t^\top \nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\epsilon}_t} f(\underline{\boldsymbol{\epsilon}})|_{\underline{\boldsymbol{\epsilon}} = \widetilde{\underline{\boldsymbol{\epsilon}}}}] \right\|_2 \\ &\geqslant \frac{1}{\alpha_{\mathrm{D}}} (J_{\mathrm{D}}^* - J_{\mathrm{D}}(\boldsymbol{\theta})) - \frac{\beta_{\mathrm{D}}}{\alpha_{\mathrm{D}}} - \left\| \sum_{t=0}^{T-1} \mathop{\mathbb{E}}_{\underline{\boldsymbol{\epsilon}} \sim \Phi_{d_{\mathcal{A}}}^T} [\boldsymbol{\epsilon}_t^\top \nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\epsilon}_t} f(\underline{\boldsymbol{\epsilon}})|_{\underline{\boldsymbol{\epsilon}} = \widetilde{\underline{\boldsymbol{\epsilon}}}}] \right\|_2 \\ &\geqslant \frac{1}{\alpha_{\mathrm{D}}} (J_{\mathrm{D}}^* - J_{\mathrm{D}}(\boldsymbol{\theta})) - \frac{\beta_{\mathrm{D}}}{\alpha_{\mathrm{D}}} - \sum_{t=0}^{T-1} \mathop{\mathbb{E}}_{\boldsymbol{\epsilon}_t \sim \Phi_{d_{\mathcal{A}}}} [\|\boldsymbol{\epsilon}_t\|_2^2]^{1/2} \mathop{\mathbb{E}}_{\underline{\boldsymbol{\epsilon}} \sim \Phi_{d_{\mathcal{A}}}^T} [\|\nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\epsilon}_t} f(\underline{\boldsymbol{\epsilon}})|_{\underline{\boldsymbol{\epsilon}} = \widetilde{\underline{\boldsymbol{\epsilon}}}}\|_2^2]^{1/2} \\ &\geqslant \frac{1}{\alpha_{\mathrm{D}}} (J_{\mathrm{D}}^* - J_{\mathrm{D}}(\boldsymbol{\theta})) - \frac{\beta_{\mathrm{D}}}{\alpha_{\mathrm{D}}} - \sigma_{\mathrm{A}} \sqrt{d_{\mathcal{A}}} \sum_{t=0}^{T-1} \mathop{\mathbb{E}}_{\underline{\boldsymbol{\epsilon}} \sim \Phi_{d_{\mathcal{A}}}^T} [\|\nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\epsilon}_t} f(\underline{\boldsymbol{\epsilon}})|_{\underline{\boldsymbol{\epsilon}} = \widetilde{\underline{\boldsymbol{\epsilon}}}}\|_2^2]^{1/2}, \end{aligned}$$

where the second inequality is by Assumption 7.1. Re-arranging the last inequality, we have:

$$J_{\mathrm{D}}^{*}-J_{\mathrm{D}}(\boldsymbol{\theta})\leqslant\alpha_{\mathrm{D}}\left\|\nabla_{\boldsymbol{\theta}}J_{\mathrm{A}}(\boldsymbol{\theta})\right\|_{2}+\beta_{\mathrm{D}}+\alpha_{\mathrm{D}}\sigma_{\mathrm{A}}\sqrt{d_{\mathcal{A}}}\sum_{t=0}^{T-1}\mathop{\mathbb{E}}_{\underline{\boldsymbol{\epsilon}}\sim\Phi_{d_{\mathcal{A}}}^{T}}\left[\left\|\nabla_{\boldsymbol{\theta}}\nabla_{\boldsymbol{\epsilon}_{t}}f(\underline{\boldsymbol{\epsilon}})|_{\underline{\boldsymbol{\epsilon}}=\underline{\tilde{\boldsymbol{\epsilon}}}}\right\|_{2}^{2}\right]^{1/2}.$$

In order to conclude the proof, we need to bound the term $\sum_{t=0}^{T-1}\mathbb{E}_{\underline{\boldsymbol{\epsilon}}\sim\Phi_{d_{\mathcal{A}}}^{T}}\left[\left\|\nabla_{\boldsymbol{\theta}}\nabla_{\boldsymbol{\epsilon}_{t}}f(\underline{\boldsymbol{\epsilon}})|_{\underline{\boldsymbol{\epsilon}}=\underline{\tilde{\boldsymbol{\epsilon}}}}\right\|_{2}^{2}\right]^{1/2}$. From the proof of Proposition E.1, for any index $k\in[\![T]\!]$, we have that:

$$\nabla_{\boldsymbol{\epsilon}_{k}}f(\underline{\boldsymbol{\epsilon}})=\mathop{\mathbb{E}}_{\tau\sim p_{\mathrm{D}}(\cdot;\underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}}+\underline{\boldsymbol{\epsilon}})}\left[\sum_{t=k}^{T-1}\gamma^{t}r(s_{t},\mu_{\boldsymbol{\theta}}(s_{t})+\boldsymbol{\epsilon}_{t})\nabla_{\boldsymbol{\epsilon}_{k}}\log p(s_{k+1}|s_{k},\mu_{\boldsymbol{\theta}}(s_{k})+\boldsymbol{\epsilon}_{k})+\gamma^{k}\nabla_{\boldsymbol{\epsilon}_{k}}r(s_{k},\mu_{\boldsymbol{\theta}}(s_{k})+\boldsymbol{\epsilon}_{k})\right],$$

from which we can derive $\nabla_{\boldsymbol{\theta}}\nabla_{\boldsymbol{\epsilon}_{k}}f(\underline{\boldsymbol{\epsilon}})$ as follows:

$$\begin{aligned}
&\nabla_{\boldsymbol{\theta}}\nabla_{\boldsymbol{\epsilon}_{k}}f(\underline{\boldsymbol{\epsilon}})\\
&=\nabla_{\boldsymbol{\theta}}\int_{\tau}p_{\mathrm{D}}(\tau;\underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}}+\underline{\boldsymbol{\epsilon}})\left(\sum_{t=k}^{T-1}\gamma^{t}r(s_{t},\mu_{\boldsymbol{\theta}}(s_{t})+\boldsymbol{\epsilon}_{t})\nabla_{\boldsymbol{\epsilon}_{k}}\log p(s_{k+1}|s_{k},\mu_{\boldsymbol{\theta}}(s_{k})+\boldsymbol{\epsilon}_{k})+\gamma^{k}\nabla_{\boldsymbol{\epsilon}_{k}}r(s_{k},\mu_{\boldsymbol{\theta}}(s_{k})+\boldsymbol{\epsilon}_{k})\right)\mathrm{d}\tau\\
&=\underbrace{\nabla_{\boldsymbol{\theta}}\int_{\tau}p_{\mathrm{D}}(\tau;\underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}}+\underline{\boldsymbol{\epsilon}})\sum_{t=k}^{T-1}\gamma^{t}r(s_{t},\mu_{\boldsymbol{\theta}}(s_{t})+\boldsymbol{\epsilon}_{t})\nabla_{\boldsymbol{\epsilon}_{k}}\log p(s_{k+1}|s_{k},\mu_{\boldsymbol{\theta}}(s_{k})+\boldsymbol{\epsilon}_{k})\mathrm{d}\tau}_{\text{(i)}}\\
&\quad+\underbrace{\nabla_{\boldsymbol{\theta}}\int_{\tau}p_{\mathrm{D}}(\tau;\underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}}+\underline{\boldsymbol{\epsilon}})\gamma^{k}\nabla_{\boldsymbol{\epsilon}_{k}}r(s_{k},\mu_{\boldsymbol{\theta}}(s_{k})+\boldsymbol{\epsilon}_{k})\mathrm{d}\tau}_{\text{(ii)}}.
\end{aligned}$$

We will consider the terms (i) and (ii) separately. However, we first need to clarify what happens when we try to compute the gradient w.r.t. $\boldsymbol{\theta}$ and $\boldsymbol{\epsilon}_t$, for a generic $t\in\{0,\dots,T-1\}$. To this purpose let $g(\cdot,\mathbf{a})$ be a generic differentiable function of the action $\mathbf{a}=\mu_{\boldsymbol{\theta}}(s_t)+\boldsymbol{\epsilon}_t$. The norm of its gradient w.r.t. $\boldsymbol{\epsilon}_t$ can be written as:

$$\begin{aligned}
\left\|\nabla_{\boldsymbol{\epsilon}_{t}}g(\cdot,\mathbf{a})|_{\mathbf{a}=\mu_{\boldsymbol{\theta}}(s_{t})+\boldsymbol{\epsilon}_{t}}\right\|_{2}&=\left\|\nabla_{\mathbf{a}}g(\cdot,\mathbf{a})|_{\mathbf{a}=\mu_{\boldsymbol{\theta}}(s_{t})+\boldsymbol{\epsilon}_{t}}\nabla_{\boldsymbol{\epsilon}_{t}}(\mu_{\boldsymbol{\theta}}(s_{t})+\boldsymbol{\epsilon}_{t})\right\|_{2}\\
&=\left\|\nabla_{\mathbf{a}}g(\cdot,\mathbf{a})|_{\mathbf{a}=\mu_{\boldsymbol{\theta}}(s_{t})+\boldsymbol{\epsilon}_{t}}\right\|_{2}.
\end{aligned}$$

On the other hand, the norm of the gradient of g w.r.t. $\boldsymbol{\theta}$ can be written as:

$$\begin{aligned}
\left\|\nabla_{\boldsymbol{\theta}}g(\cdot,\mathbf{a})|_{\mathbf{a}=\mu_{\boldsymbol{\theta}}(s_{t})+\boldsymbol{\epsilon}_{t}}\right\|_{2}&=\left\|\nabla_{\mathbf{a}}g(\cdot,\mathbf{a})|_{\mathbf{a}=\mu_{\boldsymbol{\theta}}(s_{t})+\boldsymbol{\epsilon}_{t}}\nabla_{\boldsymbol{\theta}}(\mu_{\boldsymbol{\theta}}(s_{t})+\boldsymbol{\epsilon}_{t})\right\|_{2}\\
&=\left\|\nabla_{\mathbf{a}}g(\cdot,\mathbf{a})|_{\mathbf{a}=\mu_{\boldsymbol{\theta}}(s_{t})+\boldsymbol{\epsilon}_{t}}\nabla_{\boldsymbol{\theta}}\mu_{\boldsymbol{\theta}}(s_{t})\right\|_{2}.
\end{aligned}$$

Moreover, the norm of the gradient w.r.t. $\boldsymbol{\theta}$ of the gradient of g w.r.t. $\boldsymbol{\epsilon}_t$, can be written as:

$$\begin{aligned}
\left\|\nabla_{\boldsymbol{\theta}}\nabla_{\boldsymbol{\epsilon}_{t}}g(\cdot,\mathbf{a})|_{\mathbf{a}=\mu_{\boldsymbol{\theta}}(s_{t})+\boldsymbol{\epsilon}_{t}}\right\|_{2}&=\left\|\nabla_{\boldsymbol{\theta}}\nabla_{\mathbf{a}}g(\cdot,\mathbf{a})|_{\mathbf{a}=\mu_{\boldsymbol{\theta}}(s_{t})+\boldsymbol{\epsilon}_{t}}\right\|_{2}\\
&=\left\|\nabla_{\mathbf{a}}^{2}g(\cdot,\mathbf{a})|_{\mathbf{a}=\mu_{\boldsymbol{\theta}}+\boldsymbol{\epsilon}_{t}}\nabla_{\boldsymbol{\theta}}\mu_{\boldsymbol{\theta}}(s_{t})\right\|_{2}.
\end{aligned}$$

Having said this, we can proceed by analyzing the terms (i) and (ii).

The term (i) can be rewritten as:

$$\begin{aligned}
\text{(i)}&=\nabla_{\boldsymbol{\theta}}\int_{\tau}p_{\mathrm{D}}(\tau;\underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}}+\underline{\boldsymbol{\epsilon}})\sum_{t=k}^{T-1}\gamma^{t}r(s_{t},\mu_{\boldsymbol{\theta}}(s_{t})+\boldsymbol{\epsilon}_{t})\nabla_{\boldsymbol{\epsilon}_{k}}\log p(s_{k+1}|s_{k},\mu_{\boldsymbol{\theta}}(s_{k})+\boldsymbol{\epsilon}_{k})\mathrm{d}\tau\\
&=\sum_{t=k}^{T-1}\gamma^{t}\nabla_{\boldsymbol{\theta}}\int_{\tau_{0:t}}p_{\mathrm{D}}(\tau_{0:t};\underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}}+\underline{\boldsymbol{\epsilon}})r(s_{t},\mu_{\boldsymbol{\theta}}(s_{t})+\boldsymbol{\epsilon}_{t})\nabla_{\boldsymbol{\epsilon}_{k}}\log p(s_{k+1}|s_{k},\mu_{\boldsymbol{\theta}}(s_{k})+\boldsymbol{\epsilon}_{k})\mathrm{d}\tau_{0:t}\\
&=\sum_{t=k}^{T-1}\gamma^{t}\mathop{\mathbb{E}}_{\tau_{0:t}\sim p_{\mathrm{D}}(\cdot;\underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}}+\underline{\boldsymbol{\epsilon}})}\Big[\nabla_{\boldsymbol{\theta}}\log p_{\mathrm{D}}(\tau_{0:t},\underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}}+\underline{\boldsymbol{\epsilon}})r(s_{t},\mu_{\boldsymbol{\theta}}(s_{t})+\boldsymbol{\epsilon}_{t})\nabla_{\boldsymbol{\epsilon}_{k}}\log p(s_{k+1}|s_{k},\mu_{\boldsymbol{\theta}}(s_{k})+\boldsymbol{\epsilon}_{k})\\
&\qquad+\nabla_{\boldsymbol{\theta}}r(s_{t},\mu_{\boldsymbol{\theta}}(s_{t})+\boldsymbol{\epsilon}_{t})\nabla_{\boldsymbol{\epsilon}_{k}}\log p(s_{k+1}|s_{k},\mu_{\boldsymbol{\theta}}(s_{k})+\boldsymbol{\epsilon}_{k})\\
&\qquad+r(s_{t},\mu_{\boldsymbol{\theta}}(s_{t})+\boldsymbol{\epsilon}_{t})\nabla_{\boldsymbol{\theta}}\nabla_{\boldsymbol{\epsilon}_{k}}\log p(s_{k+1}|s_{k},\mu_{\boldsymbol{\theta}}(s_{k})+\boldsymbol{\epsilon}_{k})\Big]
\end{aligned}$$

$$= \sum_{t=k}^{T-1} \gamma^t \mathop{\mathbb{E}}_{\tau_{0:t}\sim p_{\mathrm{D}}(\cdot;\underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}}+\underline{\boldsymbol{\epsilon}})} \Bigg[\sum_{j=0}^{t-1} \nabla_{\boldsymbol{\theta}} \log p(s_{j+1}|s_j, \mu_{\boldsymbol{\theta}}(s_j)+\boldsymbol{\epsilon}_j) r(s_t, \mu_{\boldsymbol{\theta}}(s_t)+\boldsymbol{\epsilon}_t) \nabla_{\boldsymbol{\epsilon}_k} \log p(s_{k+1}|s_k, \mu_{\boldsymbol{\theta}}(s_k)+\boldsymbol{\epsilon}_k)$$
$$+ \nabla_{\boldsymbol{\theta}} r(s_t, \mu_{\boldsymbol{\theta}}(s_t)+\boldsymbol{\epsilon}_t) \nabla_{\boldsymbol{\epsilon}_k} \log p(s_{k+1}|s_k, \mu_{\boldsymbol{\theta}}(s_k)+\boldsymbol{\epsilon}_k)$$
$$+ r(s_t, \mu_{\boldsymbol{\theta}}(s_t)+\boldsymbol{\epsilon}_t) \nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\epsilon}_k} \log p(s_{k+1}|s_k, \mu_{\boldsymbol{\theta}}(s_k)+\boldsymbol{\epsilon}_k) \Bigg].$$

We need to bound its norm, thus we can proceed as follows:

$$\begin{aligned}
&\|(i)\|_2 \\
&\leqslant \sum_{t=k}^{T-1} \gamma^t \mathop{\mathbb{E}}_{\tau_{0:t}\sim p_{\mathrm{D}}(\cdot;\underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}}+\underline{\boldsymbol{\epsilon}})} \Bigg[\sum_{j=0}^{t-1} \|\nabla_{\boldsymbol{\theta}} \log p(s_{j+1}|s_j, \mu_{\boldsymbol{\theta}}(s_j)+\boldsymbol{\epsilon}_j)\|_2 \, |r(s_t, \mu_{\boldsymbol{\theta}}(s_t)+\boldsymbol{\epsilon}_t)| \, \|\nabla_{\boldsymbol{\epsilon}_k} \log p(s_{k+1}|s_k, \mu_{\boldsymbol{\theta}}(s_k)+\boldsymbol{\epsilon}_k)\|_2 \\
&\qquad + \|\nabla_{\boldsymbol{\theta}} r(s_t, \mu_{\boldsymbol{\theta}}(s_t)+\boldsymbol{\epsilon}_t)\|_2 \, \|\nabla_{\boldsymbol{\epsilon}_k} \log p(s_{k+1}|s_k, \mu_{\boldsymbol{\theta}}(s_k)+\boldsymbol{\epsilon}_k)\|_2 \\
&\qquad + |r(s_t, \mu_{\boldsymbol{\theta}}(s_t)+\boldsymbol{\epsilon}_t)| \, \|\nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\epsilon}_k} \log p(s_{k+1}|s_k, \mu_{\boldsymbol{\theta}}(s_k)+\boldsymbol{\epsilon}_k)\|_2 \Bigg] \\
&\leqslant L_\mu L_{\log p}^2 R_{\max} \sum_{t=k}^{T-1} t\gamma^t + (L_\mu L_r L_{\log p} + R_{\max} L_\mu L_{2,\log p}) \sum_{t=k}^{T-1} \gamma^t \\
&= L_\mu L_{\log p}^2 R_{\max} \gamma \left(\frac{1 - T\gamma^{T-1} + (T-1)\gamma^T}{(1-\gamma)^2} - \frac{1 - k\gamma^{k-1} + (k-1)\gamma^k}{(1-\gamma)^2} \right) + (L_\mu L_r L_{\log p} + R_{\max} L_\mu L_{2,\log p}) \frac{\gamma^k - \gamma^T}{1-\gamma} \\
&\leqslant L_\mu L_{\log p}^2 R_{\max} \gamma \left(\frac{1 - \gamma^T}{(1-\gamma)^2} - \frac{1 - k\gamma^{k-1}}{(1-\gamma)^2} \right) + (L_\mu L_r L_{\log p} + R_{\max} L_\mu L_{2,\log p}) \frac{\gamma^k - \gamma^T}{1-\gamma} \\
&\leqslant L_\mu L_{\log p}^2 R_{\max} \gamma \left(\frac{1 - \gamma^T}{(1-\gamma)^2} - \frac{1 - k\gamma^{k-1}}{(1-\gamma)^2} \right) + (L_\mu L_r L_{\log p} + R_{\max} L_\mu L_{2,\log p}) \frac{\gamma^k - \gamma^T}{1-\gamma}.
\end{aligned}$$

Finally, we have to sum over $k \in [\![T]\!]$:

$$\begin{aligned}
\sum_{k=0}^{T-1} \|(i)\|_2 &\leqslant \sum_{k=0}^{T-1} L_\mu L_{\log p}^2 R_{\max} \gamma \left(\frac{1 - \gamma^T}{(1-\gamma)^2} - \frac{1 - k\gamma^{k-1}}{(1-\gamma)^2} \right) + (L_\mu L_r L_{\log p} + R_{\max} L_\mu L_{2,\log p}) \frac{\gamma^k - \gamma^T}{1-\gamma} \\
&= L_\mu L_{\log p}^2 R_{\max} \gamma \left(T \frac{1-\gamma^T}{(1-\gamma)^2} - \frac{T}{(1-\gamma)^2} + \frac{1-\gamma^T}{(1-\gamma)^4} - \frac{T\gamma^{T-1}}{(1-\gamma)^3} \right) \\
&\quad + (L_\mu L_r L_{\log p} + R_{\max} L_\mu L_{2,\log p}) \frac{1 - 2\gamma^T + \gamma^{T+1}}{(1-\gamma)^2} \\
&\leqslant L_\mu L_{\log p}^2 R_{\max} \gamma \frac{1-\gamma^T}{(1-\gamma)^4} + (L_r L_{\log p} + R_{\max} L_{2,\log p}) L_\mu \frac{1-\gamma^T}{(1-\gamma)^2}.
\end{aligned}$$

The term (ii) can be rewritten as:

$$\begin{aligned}
\text{(ii)} &= \nabla_{\boldsymbol{\theta}} \int_{\tau} p_{\mathrm{D}}(\tau; \underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}} + \underline{\boldsymbol{\epsilon}}) \gamma^k \nabla_{\boldsymbol{\epsilon}_k} r(s_k, \mu_{\boldsymbol{\theta}}(s_k) + \boldsymbol{\epsilon}_k) \mathrm{d}\tau \\
&= \nabla_{\boldsymbol{\theta}} \int_{\tau_{0:k}} p_{\mathrm{D}}(\tau_{0:k}; \underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}} + \underline{\boldsymbol{\epsilon}}) \gamma^k \nabla_{\boldsymbol{\epsilon}_k} r(s_k, \mu_{\boldsymbol{\theta}}(s_k) + \boldsymbol{\epsilon}_k) \mathrm{d}\tau_{0:k} \\
&= \gamma^k \int_{\tau_{0:k}} p_{\mathrm{D}}(\tau_{0:k}; \underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}} + \underline{\boldsymbol{\epsilon}}) \left(\nabla_{\boldsymbol{\theta}} \log p_{\mathrm{D}}(\tau_{0:k}; \underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}} + \underline{\boldsymbol{\epsilon}}) \nabla_{\boldsymbol{\epsilon}_k} r(s_k, \mu_{\boldsymbol{\theta}}(s_k) + \boldsymbol{\epsilon}_k) + \nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\epsilon}_k} r(s_k, \mu_{\boldsymbol{\theta}}(s_k) + \boldsymbol{\epsilon}_k) \right) \\
&= \gamma^k \mathop{\mathbb{E}}_{\tau_{0:k}\sim p_{\mathrm{D}}(\cdot;\underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}}+\underline{\boldsymbol{\epsilon}})} \left[\sum_{j=0}^{k-1} \nabla_{\boldsymbol{\theta}} \log p(s_{j+1}|s_j, \mu_{\boldsymbol{\theta}}(s_j)+\boldsymbol{\epsilon}_j) \nabla_{\boldsymbol{\epsilon}_k} r(s_k, \mu_{\boldsymbol{\theta}}(s_k)+\boldsymbol{\epsilon}_k) + \nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\epsilon}_k} r(s_k, \mu_{\boldsymbol{\theta}}(s_k)+\boldsymbol{\epsilon}_k) \right].
\end{aligned}$$

We need to bound its norm, thus we can proceed as follows:

$$\|\text{(ii)}\|_2 \leqslant \gamma^k \mathop{\mathbb{E}}_{\tau_{0:k}\sim p_{\mathrm{D}}(\cdot;\underline{\boldsymbol{\mu}}_{\boldsymbol{\theta}}+\underline{\boldsymbol{\epsilon}})} \left[\sum_{j=0}^{k-1} \|\nabla_{\boldsymbol{\theta}} \log p(s_{j+1}|s_j, \mu_{\boldsymbol{\theta}}(s_j)+\boldsymbol{\epsilon}_j)\|_2 \, \|\nabla_{\boldsymbol{\epsilon}_k} r(s_k, \mu_{\boldsymbol{\theta}}(s_k)+\boldsymbol{\epsilon}_k)\|_2 \right.$$

$$+\|\nabla_{\boldsymbol{\theta}}\nabla_{\boldsymbol{\epsilon}_k} r(s_k,\mu_{\boldsymbol{\theta}}(s_k)+\boldsymbol{\epsilon}_k)\|_2\Big]$$
$$\leqslant L_\mu L_p L_r k\gamma^k + L_\mu L_{2,r}\gamma^k.$$

Finally, we have to sum over $k\in[\![T]\!]$:

$$\begin{aligned}\sum_{k=0}^{T-1}\|(\mathrm{ii})\|_2 &= L_\mu L_p L_r \sum_{k=0}^{T-1} k\gamma^k + L_\mu L_{2,r}\sum_{k=0}^{T-1}\gamma^k\\ &\leqslant L_\mu L_p L_r \gamma\frac{1-T\gamma^{T-1}+(T-1)\gamma^T}{(1-\gamma)^2} + L_\mu L_{2,r}\frac{1-\gamma^T}{1-\gamma}\\ &\leqslant L_\mu L_p L_r \gamma\frac{1-\gamma^T}{(1-\gamma)^2} + L_\mu L_{2,r}\frac{1-\gamma^T}{1-\gamma}.\end{aligned}$$

Putting together the bounds on (i) and (ii):

$$\begin{aligned}&\sum_{t=0}^{T-1}\mathop{\mathbb{E}}_{\underline{\boldsymbol{\epsilon}}\sim\Phi_{d_{\mathcal{A}}}^T}[\|\nabla_{\boldsymbol{\theta}}\nabla_{\boldsymbol{\epsilon}_t} f(\underline{\boldsymbol{\epsilon}})|_{\underline{\boldsymbol{\epsilon}}=\tilde{\underline{\boldsymbol{\epsilon}}}}\|_2^2]^{1/2}\\ &\leqslant \sum_{k=0}^{T-1}\|(i)\|_2 + \sum_{k=0}^{T-1}\|(\mathrm{ii})\|_2\\ &\leqslant L_\mu L_p^2 R_{\max}\gamma\frac{1-\gamma^T}{(1-\gamma)^4} + (L_r L_p + R_{\max}L_{2,p})L_\mu\frac{1-\gamma^T}{(1-\gamma)^2}\\ &\quad + L_\mu L_p L_r\gamma\frac{1-\gamma^T}{(1-\gamma)^2} + L_\mu L_{2,r}\frac{1-\gamma^T}{1-\gamma}\\ &\leqslant L_\mu\left(\frac{L_p^2 R_{\max}\gamma}{(1-\gamma)^4} + \frac{(L_r L_p + R_{\max}L_{2,p} + L_p L_r\gamma)}{(1-\gamma)^2} + \frac{L_{2,r}}{1-\gamma}\right)(1-\gamma^T),\end{aligned}$$

which concludes the proof. □

Theorem 7.2 (Inherited weak gradient domination on J_{A})**.** *Under Assumptions 4.1, 4.2, 4.3, 4.4, 7.1, for every $\boldsymbol{\theta}\in\Theta$:*

$$J_A{}^* - J_A(\boldsymbol{\theta}) \leqslant \alpha_D\|\nabla_{\boldsymbol{\theta}} J_A(\boldsymbol{\theta})\|_2 + \beta_D + (\alpha_D\psi + L_A)\sigma_A\sqrt{d_{\mathcal{A}}},$$

where $\psi = O((1-\gamma)^{-4})$ (full expression in the proof).

Proof. This proof directly follows from the combination of Lemma D.11 and Theorem E.1, and we can proceed as in the proof of Theorem 7.1. Indeed, recalling that L is

$$L = \frac{\gamma+\gamma^{1+T}(T-1)-T\gamma^T}{(1-\gamma)^2}L_{\log p}R_{\max} + \frac{1-\gamma^T}{1-\gamma}L_r,$$

from Theorem E.1, it follows that:

$$J_{\mathrm{A}}(\boldsymbol{\theta}) - L\sigma_{\mathrm{A}}\sqrt{d_{\mathcal{A}}} \leqslant J_{\mathrm{D}}(\boldsymbol{\theta}) \leqslant J_{\mathrm{A}}(\boldsymbol{\theta}) + L\sigma_{\mathrm{A}}\sqrt{d_{\mathcal{A}}}. \tag{79}$$

Analogously to the proof of Theorem 7.1, it is useful to notice that by definition of J_{A}, we have $J_{\mathrm{D}}^* \geqslant J_{\mathrm{A}}{}^*$. Thus, it holds that:

$$\begin{aligned}J_{\mathrm{D}}^* - J_{\mathrm{D}}(\boldsymbol{\theta}) &\geqslant J_{\mathrm{A}}{}^* - J_{\mathrm{D}}(\boldsymbol{\theta})\\ &\geqslant J_{\mathrm{A}}{}^* - J_{\mathrm{A}}(\boldsymbol{\theta}) - L\sigma_{\mathrm{A}}\sqrt{d_{\mathcal{A}}},\end{aligned}$$

where the last line follows from Line (79). We rename $L_A := L$ in the statement. □

Theorem D.12 (Global convergence of PGPE - Inherited WGD)**.** *Consider the PGPE algorithm. Under Assumptions 4.1, 4.3, 4.2, 4.4, 4.5, 7.1, with a suitable constant step size and setting $\sigma_P = \frac{\epsilon}{(\alpha_D L_2 + 4L_J)\sqrt{d_\Theta}} = O(\epsilon(1-\gamma)^3 d_\Theta^{-1/2})$, to guarantee $J_D^* - \mathbb{E}[J_D(\boldsymbol{\theta}_K)] \leqslant \epsilon + \beta_D$ the sample complexity is at most:*

$$NK = \widetilde{O}\left(\frac{\alpha_D^6 d_\Theta^2}{(1-\gamma)^{11}\epsilon^5}\right). \tag{80}$$

Proof. Simply apply Theorem F.1 and Theorem 5.1 to obtain a guarantee of $J_{\mathrm{D}}^{*}-\mathbb{E}[J_{\mathrm{D}}(\boldsymbol{\theta}_K)]\leqslant\epsilon/2+\beta_{\mathrm{D}}+(\alpha_{\mathrm{D}}L_2+L_J)\sqrt{d_\Theta}\sigma_{\mathrm{P}}+3L_J\sqrt{d_\Theta}\sigma_{\mathrm{P}}=\epsilon+\beta_{\mathrm{D}}+(\alpha_{\mathrm{D}}L_2+4L_J)\sqrt{d_\Theta}\sigma_{\mathrm{P}}$. Then, we set σ_{P} to ensure that $(\alpha_{\mathrm{D}}L_2+4L_J)\sqrt{d_\Theta}\sigma_{\mathrm{P}}=\epsilon/2$. □

Theorem D.13 (Global convergence of GPOMDP - Inherited WGD)**.** *Consider the GPOMDP algorithm. Under Assumptions 4.1, 4.3, 4.2, 4.4, 4.5, 7.1, with a suitable constant step size and setting* $\sigma_A=\frac{\epsilon}{(\alpha_D\Psi+4L)\sqrt{d_{\mathcal{A}}}}=O(\epsilon(1-\gamma)^4d_{\mathcal{A}}^{-1/2})$*, to guarantee* $J_D^{*}-\mathbb{E}[J_D(\boldsymbol{\theta}_K)]\leqslant\epsilon+\beta_D$ *the sample complexity is at most:*

$$NK=\widetilde{O}\left(\frac{\alpha_D^6d_{\mathcal{A}}^2}{(1-\gamma)^{14}\epsilon^5}\right). \tag{81}$$

Proof. Simply apply Theorem F.1 and Theorem 5.2 to obtain a guarantee of $J_{\mathrm{D}}^{*}-\mathbb{E}[J_{\mathrm{D}}(\boldsymbol{\theta}_K)]\leqslant\epsilon/2+\beta_{\mathrm{D}}+(\alpha_{\mathrm{D}}\Psi+L)\sqrt{d_\Theta}\sigma_{\mathrm{A}}+3L\sqrt{d_\Theta}\sigma_{\mathrm{A}}=\epsilon+\beta_{\mathrm{D}}+(\alpha_{\mathrm{D}}\Psi+4L)\sqrt{d_\Theta}\sigma_{\mathrm{A}}$. Then, we set σ_{A} to ensure that $(\alpha_{\mathrm{D}}\Psi+4L)\sqrt{d_{\mathcal{A}}}\sigma_{\mathrm{A}}=\epsilon/2$. □

D.4. Proofs from Section 7.2

In this section, we focus on AB exploration with white-noise policies (Definition 3.2), and give the proofs that were omitted in Section 7.2. We denote by $\upsilon_{\boldsymbol{\theta}}(\cdot,\cdot)$ the state-action distribution induced by the (stochastic) policy $\pi_{\boldsymbol{\theta}}$, and, with some abuse of notation, $\upsilon_{\boldsymbol{\theta}}(\cdot)$ to denote the corresponding state distribution. We denote by $A^{\boldsymbol{\theta}}:\mathcal{S}\times\mathcal{A}\to\mathbb{R}$ the advantage function of $\pi_{\boldsymbol{\theta}}$ (for the standard definitions, see Sutton & Barto, 2018).

We first have to give a formal characterization of ϵ_{bias}. Equivalent definitions appeared in (Liu et al., 2020; Ding et al., 2022; Yuan et al., 2022), but the concept dates back at least to (Peters et al., 2005).

Definition D.1. *Let* $\ell(\boldsymbol{w};s,a,\boldsymbol{\theta})=\left(A^{\boldsymbol{\theta}}(s,a)-(1-\gamma)\boldsymbol{w}^\top\nabla_{\boldsymbol{\theta}}\log\pi_{\boldsymbol{\theta}}(a|s)\right)^2$*, and* $\boldsymbol{w}^\star(\boldsymbol{\theta})=\arg\min_{\boldsymbol{w}\in\mathbb{R}^{d_\Theta}}\mathbb{E}_{s,a\sim\upsilon_{\boldsymbol{\theta}}}[\ell(\boldsymbol{w};s,a,\boldsymbol{\theta})]$*. We define* ϵ_{bias} *as the smallest positive constant such that, for all* $\boldsymbol{\theta}\in\Theta$*,* $\mathbb{E}_{s,a\sim\upsilon_{\boldsymbol{\theta}^\star}}[\ell(\boldsymbol{w}^\star(\boldsymbol{\theta});s,a,\boldsymbol{\theta})]\leqslant\epsilon_{\mathrm{bias}}$*, where* $\boldsymbol{\theta}^\star\in\arg\max J_A(\boldsymbol{\theta})$.

We begin by showing that white-noise policies are Fisher-non-degenerate, in the sense of (Ding et al., 2022). First we need to introduce the concept of Fisher information matrix, that for stochastic policies is defined as (Kakade, 2001):

$$F(\boldsymbol{\theta}):=\mathop{\mathbb{E}}_{s,a\sim\upsilon^{\boldsymbol{\theta}}}[\nabla_{\boldsymbol{\theta}}\log\pi_{\boldsymbol{\theta}}(a|s)\nabla_{\boldsymbol{\theta}}\log\pi_{\boldsymbol{\theta}}(a|s)^\top]. \tag{82}$$

Lemma D.14. *Let* $\pi_{\boldsymbol{\theta}}$ *be a white-noise policy (Definition 3.2). Under Assumption 7.2, for all* $\boldsymbol{\theta}\in\Theta$*,* $F(\boldsymbol{\theta})\geq\lambda_F I$*, where*

$$\lambda_F:=\frac{\lambda_E}{\sigma_A^2}.$$

Proof. Let $\Sigma=\mathbb{E}_{\boldsymbol{\epsilon}\sim\Phi_{d_{\mathcal{A}}}}[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^\top]$ be the covariance matrix of the noise, which by definition has $\lambda_{\max}(\Sigma)\leqslant\sigma_A^2$. By a simple change of variable and Cramer-Rao's bound:

$$\begin{aligned}
F(\boldsymbol{\theta})&=\mathop{\mathbb{E}}_{s,a\sim\upsilon^{\boldsymbol{\theta}}}[\nabla_{\boldsymbol{\theta}}\log\pi_{\boldsymbol{\theta}}(a|s)\nabla_{\boldsymbol{\theta}}\log\pi_{\boldsymbol{\theta}}(a|s)^\top]\\
&=\mathop{\mathbb{E}}_{s\sim\upsilon^{\boldsymbol{\theta}}}\left[\nabla_{\boldsymbol{\theta}}\mu_{\boldsymbol{\theta}}(s)\mathop{\mathbb{E}}_{a\sim\pi_{\boldsymbol{\theta}}(\cdot|s)}\left[\nabla_{\boldsymbol{\epsilon}}\log\phi(\widetilde{\boldsymbol{\epsilon}})|_{\widetilde{\boldsymbol{\epsilon}}=a-\mu_{\boldsymbol{\theta}}(s)}\nabla_{\boldsymbol{\epsilon}}\log\phi(\widetilde{\boldsymbol{\epsilon}})^\top|_{\widetilde{\boldsymbol{\epsilon}}=a-\mu_{\boldsymbol{\theta}}(s)}\right]\nabla_{\boldsymbol{\theta}}\mu_{\boldsymbol{\theta}}(s)^\top\right]\\
&=\mathop{\mathbb{E}}_{s\sim\upsilon^{\boldsymbol{\theta}}}\left[\nabla_{\boldsymbol{\theta}}\mu_{\boldsymbol{\theta}}(s)\mathop{\mathbb{E}}_{\boldsymbol{\epsilon}\sim\Phi_{d_{\mathcal{A}}}}\left[\nabla_{\boldsymbol{\epsilon}}\log\phi(\boldsymbol{\epsilon})\nabla_{\boldsymbol{\epsilon}}\log\phi(\boldsymbol{\epsilon})^\top\right]\nabla_{\boldsymbol{\theta}}\mu_{\boldsymbol{\theta}}(s)^\top\right]\\
&\geq\mathop{\mathbb{E}}_{s\sim\upsilon^{\boldsymbol{\theta}}}\left[\nabla_{\boldsymbol{\theta}}\mu_{\boldsymbol{\theta}}(s)\Sigma^{-1}\nabla_{\boldsymbol{\theta}}\mu_{\boldsymbol{\theta}}(s)^\top\right] && \text{(Cramer-Rao)}\\
&\geq\frac{1}{\lambda_{\max}(\Sigma)}\mathop{\mathbb{E}}_{s\sim\upsilon^{\boldsymbol{\theta}}}\left[\nabla_{\boldsymbol{\theta}}\mu_{\boldsymbol{\theta}}(s)\nabla_{\boldsymbol{\theta}}\mu_{\boldsymbol{\theta}}(s)^\top\right]\\
&\geq\frac{\lambda_E}{\sigma_A^2}I.
\end{aligned}$$

□

We can then use Corollary 4.14 by Yuan et al. (2022), itself a refinement of Lemma 4.7 by Ding et al. (2022), to prove that J_{A} enjoys the WGD property.

Theorem 7.3 (Policy-induced weak gradient domination)**.** *Under Assumptions 4.5 and 7.2, we have:*

$$J_A{}^* - J_A(\boldsymbol{\theta}) \leqslant C\frac{\sqrt{d_{\mathcal{A}}}\sigma_A}{\lambda_E}\|\nabla_{\boldsymbol{\theta}} J_A(\boldsymbol{\theta})\|_2 + \frac{\sqrt{\epsilon_{\text{bias}}}}{1-\gamma},$$

for some numerical constant $C>0$. Thus, Assumption 6.1 (†=A) is satisfied with $\alpha = C\frac{\sqrt{d_{\mathcal{A}}}\sigma_A}{\lambda_E}$ and $\beta = \frac{\sqrt{\epsilon_{\text{bias}}}}{1-\gamma}$.

Proof. Corollary 4.14 by Yuan et al. (2022) tells us that, under Assumption D.1,

$$J_{\text{A}}{}^* - J_{\text{A}}(\boldsymbol{\theta}) \leqslant \frac{\xi}{\lambda_F}\|\nabla_{\boldsymbol{\theta}} J_{\text{A}}(\boldsymbol{\theta})\| + \frac{\sqrt{\epsilon_{\text{bias}}}}{1-\gamma},$$

whenever $F(\boldsymbol{\theta}) \geq \lambda_F I$ and $\mathbb{E}_{a\sim\pi_{\boldsymbol{\theta}}(\cdot|s)}[\|\nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a|s)\|^2] \leqslant \xi^2$ hold for all $\boldsymbol{\theta}\in\Theta$ and $s\in\mathcal{S}$. By Lemma D.14, and the fact that $\xi = \sqrt{cd_{\mathcal{A}}}\sigma_A^{-1}$ is a valid choice under Assumption 4.5, the previous display holds with

$$\frac{\xi}{\lambda_F} \leqslant \frac{\sqrt{cd_{\mathcal{A}}}\sigma_A^{-1}}{\lambda_E\sigma_A^{-2}} = \frac{\sqrt{cd_{\mathcal{A}}}\sigma_A}{\lambda_E},$$

the proof is concluded by letting $C=\sqrt{c}$, where c is the constant from Assumption 4.5. □

Finally, we can use the WGD property just established, with its values of α and β, to prove special cases of Theorems D.8 and D.9. The key difference with respect to the other sample complexity results presented in the paper is that the amount of noise σ_A has an effect on the α parameter of the WGD property.

We first consider the case of a generic σ_A:

Theorem D.15. *Consider the GPOMDP algorithm. Under Assumptions 4.1, 4.3, 4.4, 4.5, 7.2, and D.1, with a suitable constant step size, to guarantee $J_D^* - \mathbb{E}[J_D(\boldsymbol{\theta}_K)] \leqslant \epsilon + \frac{\sqrt{\epsilon_{\text{bias}}}}{1-\gamma} + 3L\sqrt{d_{\mathcal{A}}}\sigma_A$, where $3L\sqrt{d_{\mathcal{A}}}\sigma_A = O(\sqrt{d_{\mathcal{A}}}\sigma_A(1-\gamma)^{-2})$ the sample complexity is at most:*

$$NK = \widetilde{O}\left(\frac{d_{\mathcal{A}}^4}{\lambda_E^4(1-\gamma)^5\epsilon^3}\right). \tag{83}$$

Furthermore, under Assumption 4.2, the same guarantee is obtained with a sample complexity of at most:

$$NK = \widetilde{O}\left(\frac{d_{\mathcal{A}}^3\sigma_A^2}{\lambda_E^4(1-\gamma)^6\epsilon^3}\right). \tag{84}$$

Proof. By Theorem D.8 and Lemma 7.3. □

The first bound seem to have no dependence on σ_A. However, a complex dependence is hidden in λ_E^4. Also, it may seem that $\sigma_A \simeq 0$ is a good choice, especially for the second bound. However, λ_E can be very large (or infinite) for a (quasi-)deterministic policy.

If we instead set σ_A as in Section 6 in order to converge to a good deterministic policy (which, of course, completely ignores the complex dependencies of λ_E and ϵ_{bias} on σ_A), we obtain the following:

Theorem D.16. *Consider the GPOMDP algorithm. Under Assumptions 4.1, 4.3, 4.4, 4.5, 7.2, and D.1 with a suitable constant step size and setting $\sigma_A = \frac{\epsilon}{6L\sqrt{d_{\mathcal{A}}}} = O(\epsilon(1-\gamma)^2 d_{\mathcal{A}}^{-1/2})$, to guarantee $J_D^* - \mathbb{E}[J_D(\boldsymbol{\theta}_K)] \leqslant \epsilon + \frac{\sqrt{\epsilon_{\text{bias}}}}{1-\gamma}$ the sample complexity is at most:*

$$NK = \widetilde{O}\left(\frac{d_{\mathcal{A}}^6}{\lambda_E^4(1-\gamma)^{13}\epsilon^3}\right). \tag{85}$$

Furthermore, under Assumption 4.2, the same guarantee is obtained with a sample complexity of at most:

$$NK = \widetilde{O}\left(\frac{d_{\mathcal{A}}^4}{\lambda_E^4(1-\gamma)^{10}\epsilon}\right). \tag{86}$$

Proof. By Theorem D.9 and Lemma 7.3. □

The *apparently* better sample complexity w.r.t. Theorem D.9 is easily explained: using a small σ makes the α parameter of WGD from Lemma 7.3 smaller *if we ignore the effect of* λ_E, and smaller α yields faster convergence. However, Equation (86)

clearly shows that λ_E cannot be ignored. In particular, λ_E must be $O(\sigma_A^{1/4})$ not to violate the classic $\Omega(\epsilon^{-2})$ lower bound on the sample complexity (Azar et al., 2013). This may be of independent interest.

E. Assumptions' Implications

Lemma E.1 (L and L_J characterization)**.** *Assumption 4.1 implies Assumption 5.2 with:*

$$L_t \leqslant \frac{\gamma^{k+1}-\gamma^T}{1-\gamma} L_p R_{\max} + \gamma^k L_r, \tag{87}$$

$$L \leqslant \frac{\gamma(1-\gamma^T)}{(1-\gamma)^2} L_p R_{\max} + \frac{1-\gamma^T}{1-\gamma} L_r \leqslant \frac{\gamma L_p R_{\max}}{(1-\gamma)^2} + \frac{L_r}{1-\gamma}. \tag{88}$$

Assumption 4.1 and 4.3 imply Assumption 5.1 with $L_J \leqslant L L_\mu$.

Proof. In AB exploration, we introduce the following convenient expression for the trajectory density function having fixed a sequence of noise $\underline{\boldsymbol{\epsilon}}$:

$$p_{\mathrm{D}}(\tau; \underline{\boldsymbol{\mu}} + \underline{\boldsymbol{\epsilon}}) = \rho_0(s_{\tau,0}) \prod_{t=0}^{T-1} p(s_{\tau,t+1} | s_{\tau,t}, \boldsymbol{\mu}_t(s_{\tau,t}) + \boldsymbol{\epsilon}_t). \tag{89}$$

Furthermore, we denote with $p_{\mathrm{D}}(\tau_{0:l}; \underline{\boldsymbol{\mu}} + \underline{\boldsymbol{\epsilon}})$ the density function of a trajectory prefix of length l:

$$p_{\mathrm{D}}(\tau_{0:l}; \underline{\boldsymbol{\mu}} + \underline{\boldsymbol{\epsilon}}) = \rho_0(s_{\tau,0}) \prod_{t=0}^{l-1} p(s_{\tau,t+1} | s_{\tau,t}, \boldsymbol{\mu}_t(s_{\tau,t}) + \boldsymbol{\epsilon}_t). \tag{90}$$

Let us decompose $\underline{\boldsymbol{\mu}}' = \underline{\boldsymbol{\mu}}' + \underline{\boldsymbol{\epsilon}}$. We have:

$$\begin{aligned} J_{\mathrm{D}}(\underline{\boldsymbol{\mu}}') &= \int_\tau p_{\mathrm{D}}(\tau; \underline{\boldsymbol{\mu}} + \underline{\boldsymbol{\epsilon}}) \sum_{t=0}^{T-1} \gamma^t r(s_t, \boldsymbol{\mu}_t(s_t) + \boldsymbol{\epsilon}_t) \mathrm{d}\tau \\ &= \underbrace{\sum_{t=0}^{T-1} \int_{\tau_{0:t}} p_{\mathrm{D}}(\tau_{0:t}; \underline{\boldsymbol{\mu}} + \underline{\boldsymbol{\epsilon}}) \gamma^t r(s_t, \boldsymbol{\mu}_t(s_t) + \boldsymbol{\epsilon}_t) \mathrm{d}\tau_{0:t}}_{=: f(\underline{\boldsymbol{\epsilon}})}. \end{aligned}$$

Note that given the definition of $f(\underline{\boldsymbol{\epsilon}})$, we have that $f(\underline{\mathbf{0}_{d_{\mathcal{A}}}}) = J_{\mathrm{D}}(\underline{\boldsymbol{\mu}})$. Using Taylor expansion, we have for $\underline{\widetilde{\boldsymbol{\epsilon}}} = x\underline{\boldsymbol{\epsilon}}$, for some $x \in [0,1]$:

$$\begin{aligned} J_{\mathrm{D}}(\underline{\boldsymbol{\mu}}') &= f(\underline{\boldsymbol{\epsilon}}) \\ &= f(\underline{\mathbf{0}_{d_{\mathcal{A}}}}) + \underline{\boldsymbol{\epsilon}}^\top \nabla_{\underline{\boldsymbol{\epsilon}}} f(\underline{\boldsymbol{\epsilon}})|_{\underline{\boldsymbol{\epsilon}} = \underline{\widetilde{\boldsymbol{\epsilon}}}} \\ &= J_{\mathrm{D}}(\underline{\boldsymbol{\mu}}) + \sum_{t=0}^{T-1} \boldsymbol{\epsilon}_t^\top \nabla_{\boldsymbol{\epsilon}_t} f(\underline{\boldsymbol{\epsilon}})|_{\underline{\boldsymbol{\epsilon}} = \underline{\widetilde{\boldsymbol{\epsilon}}}} \\ &\leqslant J_{\mathrm{D}}(\underline{\boldsymbol{\mu}}) + \sum_{t=0}^{T-1} \|\boldsymbol{\epsilon}_t\|_2 \|\nabla_{\boldsymbol{\epsilon}_t} f(\underline{\boldsymbol{\epsilon}})|_{\underline{\boldsymbol{\epsilon}} = \underline{\widetilde{\boldsymbol{\epsilon}}}}\|_2. \end{aligned}$$

We want to find a bound for the $\|\nabla_{\boldsymbol{\epsilon}_t} f(\boldsymbol{\epsilon})|_{\underline{\boldsymbol{\epsilon}} = \underline{\widetilde{\boldsymbol{\epsilon}}}}\|_2^2$ which is different for every t. This will result in the Lipschitz constant L_t. We have for $k \in [\![0, T-1]\!]$:

$$\begin{aligned} \left\|\nabla_{\underline{\boldsymbol{\epsilon}}_k} f(\underline{\boldsymbol{\epsilon}})\right\|_2 &\leqslant \mathop{\mathbb{E}}_{\tau \sim p_{\mathrm{D}}(\cdot; \underline{\boldsymbol{\mu}} + \underline{\boldsymbol{\epsilon}})} \left[\sum_{t=0}^{T-1} \left\|\nabla_{\boldsymbol{\epsilon}_k} \log p_{\mathrm{D}}(\tau_{0:t}; \underline{\boldsymbol{\mu}} + \underline{\boldsymbol{\epsilon}})\right\|_2 \gamma^t |r(s_t, \boldsymbol{\mu}_t(s_t) + \boldsymbol{\epsilon}_t)| + \sum_{t=0}^{T-1} \gamma^t \|\nabla_{\boldsymbol{\epsilon}_k} r(s_t, \boldsymbol{\mu}_t(s_t) + \boldsymbol{\epsilon}_t)\|_2 \right] \\ &\leqslant \mathop{\mathbb{E}}_{\tau \sim p_{\mathrm{D}}(\cdot; \underline{\boldsymbol{\mu}} + \underline{\boldsymbol{\epsilon}})} \left[\sum_{t=0}^{T-1} \gamma^t |r(s_t, \boldsymbol{\mu}_t(\mathbf{s}_t) + \boldsymbol{\epsilon}_t)| \sum_{l=0}^{t-1} \|\nabla_{\boldsymbol{\epsilon}_k} p(s_{l+1} | \mathbf{s}_l, \boldsymbol{\mu}_l(s_l) + \boldsymbol{\epsilon}_l)\|_2 + \sum_{t=0}^{T-1} \gamma^t \|\nabla_{\boldsymbol{\epsilon}_k} r(s_t, \boldsymbol{\mu}_t(s_t) + \boldsymbol{\epsilon}_t)\|_2 \right] \\ &= \mathop{\mathbb{E}}_{\tau \sim p_{\mathrm{D}}(\cdot; \underline{\boldsymbol{\mu}} + \underline{\boldsymbol{\epsilon}})} \left[\sum_{t=k+1}^{T-1} \gamma^t |r(s_t, \boldsymbol{\mu}_t(s_t) + \boldsymbol{\epsilon}_t)| \|\nabla_{\boldsymbol{\epsilon}_k} p(s_{k+1} | s_k, \boldsymbol{\mu}_k(s_k) + \boldsymbol{\epsilon}_k)\|_2 + \gamma^k \|\nabla_{\boldsymbol{\epsilon}_k} r(s_k, \boldsymbol{\mu}_k(s_k) + \boldsymbol{\epsilon}_k)\|_2 \right] \end{aligned}$$

$$\leqslant \sum_{t=k+1}^{T-1} \gamma^t R_{\max} L_p + \gamma^k L_r$$
$$= \frac{\gamma^{k+1} - \gamma^T}{1-\gamma} L_p R_{\max} + \gamma^k L_r =: L_k.$$

Thus, we have:

$$\sum_{k=0}^{T-1} L_t = \sum_{k=0}^{T-1} \frac{\gamma^{k+1} - \gamma^T}{1-\gamma} L_p R_{\max} + \gamma^k L_r$$
$$= \frac{\gamma + \gamma^{1+T}(T-1) - T\gamma^T}{(1-\gamma)^2} L_p R_{\max} + \frac{1-\gamma^T}{1-\gamma} L_r$$
$$\leqslant \frac{\gamma(1-\gamma^T)}{(1-\gamma)^2} L_p R_{\max} + \frac{(1-\gamma^T)}{1-\gamma} L_r =: L.$$

For the PB exploration, we consider the trajectory density function:

$$p_{\mathrm{A}}(\tau; \boldsymbol{\theta} + \boldsymbol{\epsilon}) = \rho_0(s_{\tau,0}) \prod_{t=0}^{T-1} p(s_{\tau,t+1} | s_{\tau,t}, \boldsymbol{\mu}_{\boldsymbol{\theta}+\boldsymbol{\epsilon}}(s_{\tau,t})), \tag{91}$$

and the corresponding version for a trajectory prefix:

$$p_{\mathrm{D}}(\tau_{0:l}; \boldsymbol{\theta} + \boldsymbol{\epsilon}) = \rho_0(s_{\tau,0}) \prod_{t=0}^{l-1} p(s_{\tau,t+1} | s_{\tau,t}, \boldsymbol{\mu}_{\boldsymbol{\theta}+\boldsymbol{\epsilon}}(s_{\tau,t})). \tag{92}$$

With such a notation, we can write the $\boldsymbol{\theta}' = \boldsymbol{\theta} + \boldsymbol{\epsilon}$ index as follows:

$$J_{\mathrm{D}}(\boldsymbol{\theta}') = \int_{\tau} p_{\mathrm{D}}(\tau; \boldsymbol{\theta} + \boldsymbol{\epsilon}) \sum_{t=0}^{T-1} \gamma^t r(s_t, \boldsymbol{\mu}_{\boldsymbol{\theta}+\boldsymbol{\epsilon}}(s_t)) \mathrm{d}\tau = \underbrace{\int_{\tau} \sum_{t=0}^{T-1} p_{\mathrm{D}}(\tau_{0:t}; \boldsymbol{\theta} + \boldsymbol{\epsilon}) \gamma^t r(s_t, \boldsymbol{\mu}_{\boldsymbol{\theta}+\boldsymbol{\epsilon}}(s_t))}_{=: g(\boldsymbol{\epsilon})}.$$

We recall that $g(\mathbf{0}_{d_\Theta}) = J_{\mathrm{D}}(\boldsymbol{\theta})$. By using Taylor expansion, where $\widetilde{\boldsymbol{\epsilon}} = x\boldsymbol{\epsilon}$ for some $x \in [0,1]$:

$$J_{\mathrm{D}}(\boldsymbol{\theta}') = g(\boldsymbol{\epsilon}) \tag{93}$$
$$= g(\mathbf{0}_{d_\Theta}) + \boldsymbol{\epsilon}^\top \nabla_{\boldsymbol{\epsilon}} g(\boldsymbol{\epsilon})|_{\boldsymbol{\epsilon}=\widetilde{\boldsymbol{\epsilon}}} \tag{94}$$
$$\leqslant J_{\mathrm{D}}(\boldsymbol{\theta}) + \|\boldsymbol{\epsilon}\|_2 \|\nabla_{\boldsymbol{\epsilon}} g(\boldsymbol{\epsilon})|_{\boldsymbol{\epsilon}=\widetilde{\boldsymbol{\epsilon}}}\|_2. \tag{95}$$

We now bound the norm of the gradient:

$$\|\nabla_{\boldsymbol{\epsilon}} g(\boldsymbol{\epsilon})\|_2 \leqslant \mathop{\mathbb{E}}_{\tau \sim p_{\mathrm{A}}(\cdot; \boldsymbol{\mu}_{\boldsymbol{\theta}+\boldsymbol{\epsilon}})} \left[\sum_{t=0}^{T-1} \|\nabla_{\boldsymbol{\epsilon}} \log p(\tau_{0:t}; \boldsymbol{\theta} + \boldsymbol{\epsilon})\|_2 \gamma^t |r(s_t, \boldsymbol{\mu}_{\boldsymbol{\theta}+\boldsymbol{\epsilon}}(s_t))| + \sum_{t=0}^{T-1} \gamma^t \|\nabla_{\boldsymbol{\epsilon}} r(s_t, \boldsymbol{\mu}_{\boldsymbol{\theta}+\boldsymbol{\epsilon}}(s_t))\|_2 \right]$$
$$\leqslant \mathop{\mathbb{E}}_{\tau \sim p_{\mathrm{A}}(\cdot; \boldsymbol{\mu}_{\boldsymbol{\theta}+\boldsymbol{\epsilon}})} \left[\sum_{t=0}^{T-1} \gamma^t |r(s_t, \boldsymbol{\mu}_{\boldsymbol{\theta}+\boldsymbol{\epsilon}}(s_t))| \sum_{l=0}^{t-1} \|\nabla_{\mathbf{a}} \log p(s_{l+1} | s_l, \mathbf{a})|_{\mathbf{a}=\boldsymbol{\mu}_{\boldsymbol{\theta}+\epsilon}(s_t)}\|_2 \|\nabla_{\boldsymbol{\epsilon}} \boldsymbol{\mu}_{\boldsymbol{\theta}+\boldsymbol{\epsilon}}(s_l)\|_2 \right.$$
$$\left. + \sum_{t=0}^{T-1} \gamma^t \|\nabla_{\mathbf{a}} r(s_t, \mathbf{a})|_{\mathbf{a}=\boldsymbol{\mu}_{\boldsymbol{\theta}+\boldsymbol{\epsilon}}(s_t)}\|_2 \|\nabla_{\boldsymbol{\epsilon}} \boldsymbol{\mu}_{\boldsymbol{\theta}+\boldsymbol{\epsilon}}(s_t)\|_2 \right]$$
$$\leqslant \sum_{t=0}^{T-1} \gamma^t R_{\max} t L_p L_\mu + \frac{1-\gamma^T}{1-\gamma} L_r L_\mu$$
$$= \frac{\gamma + \gamma^{1+T}(T-1) - T\gamma^T}{(1-\gamma)^2} L_p L_\mu R_{\max} + \frac{1-\gamma^T}{1-\gamma} L_r L_\mu = L L_\mu.$$

□

Assumption E.1 (Smooth J_{D} w.r.t. parameter $\boldsymbol{\theta}$)**.** *J_D is L_2-LS w.r.t. parameter $\boldsymbol{\theta}$, i.e., for every $\boldsymbol{\theta}, \boldsymbol{\theta}' \in \Theta$, we have:*

$$\|\nabla_{\boldsymbol{\theta}} J_D(\boldsymbol{\theta}') - \nabla_{\boldsymbol{\theta}} J_D(\boldsymbol{\theta})\|_2 \leqslant L_2 \|\boldsymbol{\theta}' - \boldsymbol{\theta}\|_2. \tag{96}$$

Lemma E.2 (L_2 Characterization). *Assumptions 4.1, 4.3, 4.2, and 4.4 imply Assumption E.1 with*

$$L_2 \leqslant \frac{\gamma(1+\gamma)L_\mu^2 L_p^2 R_{\max}}{(1-\gamma)^3} + \frac{\gamma(2L_\mu^2 L_p L_r + L_{2,\mu}L_{2,p}R_{\max})}{(1-\gamma)^2} + \frac{L_{2,\mu}L_{2,r}}{1-\gamma}.$$

Proof. It suffices to find a bound to the quantity $\left\|\nabla^2_{\boldsymbol{\theta}} J_{\mathrm{D}}(\boldsymbol{\theta})\right\|_2$, for a generic $\boldsymbol{\theta}\in\Theta$. Notice that in the following we use the notation $\tau_{0:l}$ to refer to a trajectory of length l. Recalling that:

$$J_{\mathrm{D}}(\boldsymbol{\theta}) = \mathop{\mathbb{E}}_{\tau\sim p_{\mathrm{D}}(\cdot|\boldsymbol{\theta})}\left[\sum_{t=0}^{T-1}\gamma^t r(s_t,\mu_{\boldsymbol{\theta}}(s_t))\right],$$

we have what follows:

$$\begin{aligned}
\nabla^2_{\boldsymbol{\theta}} J_{\mathrm{D}}(\boldsymbol{\theta}) &= \nabla^2_{\boldsymbol{\theta}} \mathop{\mathbb{E}}_{\tau\sim p_{\mathrm{D}}(\cdot|\boldsymbol{\theta})}\left[\sum_{t=0}^{T-1}\gamma^t r(s_t,\mu_{\boldsymbol{\theta}}(s_t))\right] \\
&= \nabla^2_{\boldsymbol{\theta}} \int_\tau p_{\mathrm{D}}(\tau,\boldsymbol{\theta}) \sum_{t=0}^{T-1}\gamma^t r(s_t,\mu_{\boldsymbol{\theta}}(s_t))\mathrm{d}\tau \\
&= \sum_{t=0}^{T-1}\nabla^2_{\boldsymbol{\theta}} \int_{\tau_{0:t}} p_{\mathrm{D}}(\tau_{0:t},\boldsymbol{\theta})\gamma^t r(s_t,\mu_{\boldsymbol{\theta}}(s_t))\mathrm{d}\tau_{0:t} \\
&= \sum_{t=0}^{T-1}\nabla_{\boldsymbol{\theta}} \int_{\tau_{0:t}} p_{\mathrm{D}}(\tau_{0:t},\boldsymbol{\theta})\left(\nabla_{\boldsymbol{\theta}}\log p_{\mathrm{D}}(\tau_{0:t},\boldsymbol{\theta})\gamma^t r(s_t,\mu_{\boldsymbol{\theta}}(s_t)) + \gamma^t\nabla_{\boldsymbol{\theta}} r(s_t,\mu_{\boldsymbol{\theta}}(s_t))\right)\mathrm{d}\tau_{0:t} \\
&= \sum_{t=0}^{T-1}\mathop{\mathbb{E}}_{\tau_{0:t}\sim p_{\mathrm{D}}(\cdot|\boldsymbol{\theta})}\Bigg[\nabla_{\boldsymbol{\theta}}\log p_{\mathrm{D}}(\tau_{0:t},\boldsymbol{\theta})\left(\nabla_{\boldsymbol{\theta}}\log p_{\mathrm{D}}(\tau_{0:t},\boldsymbol{\theta})\gamma^t r(s_t,\mu_{\boldsymbol{\theta}}(s_t)) + \gamma^t\nabla_{\boldsymbol{\theta}} r(s_t,\mu_{\boldsymbol{\theta}}(s_t))\right) \\
&\qquad + \nabla^2_{\boldsymbol{\theta}}\log p_{\mathrm{D}}(\tau_{0:t},\boldsymbol{\theta})\gamma^t r(s_t,\mu_{\boldsymbol{\theta}}(s_t)) + \nabla_{\boldsymbol{\theta}}\log p_{\mathrm{D}}(\tau_{0:t},\boldsymbol{\theta})\gamma^t\nabla_{\boldsymbol{\theta}} r(s_t,\mu_{\boldsymbol{\theta}}(s_t)) + \gamma^t\nabla^2_{\boldsymbol{\theta}} r(s_t,\mu_{\boldsymbol{\theta}}(s_t))\Bigg].
\end{aligned}$$

Now that we have characterized $\nabla^2_{\boldsymbol{\theta}} J_{\mathrm{D}}(\boldsymbol{\theta})$, we can consider its norm by applying the assumptions in the statement, obtaining the following result:

$$\begin{aligned}
&\left\|\nabla^2_{\boldsymbol{\theta}} J_{\mathrm{D}}(\boldsymbol{\theta})\right\| \\
&\leqslant \sum_{t=0}^{T-1}\mathop{\mathbb{E}}_{\tau_{0:t}\sim p_{\mathrm{D}}(\cdot|\boldsymbol{\theta})}\Bigg[\left\|\nabla_{\boldsymbol{\theta}}\log p_{\mathrm{D}}(\tau_{0:t},\boldsymbol{\theta})\right\|_2\left(\left\|\nabla_{\boldsymbol{\theta}}\log p_{\mathrm{D}}(\tau_{0:t},\boldsymbol{\theta})\right\|_2\gamma^t\left|r(s_t,\mu_{\boldsymbol{\theta}}(s_t))\right| + \gamma^t\left\|\nabla_{\boldsymbol{\theta}} r(s_t,\mu_{\boldsymbol{\theta}}(s_t))\right\|_2\right) \\
&\qquad + \left\|\nabla^2_{\boldsymbol{\theta}}\log p_{\mathrm{D}}(\tau_{0:t},\boldsymbol{\theta})\right\|_2\gamma^t\left|r(s_t,\mu_{\boldsymbol{\theta}}(s_t))\right| + \left\|\nabla_{\boldsymbol{\theta}}\log p_{\mathrm{D}}(\tau_{0:t},\boldsymbol{\theta})\right\|_2\gamma^t\left\|\nabla_{\boldsymbol{\theta}} r(s_t,\mu_{\boldsymbol{\theta}}(s_t))\right\|_2 + \gamma^t\left\|\nabla^2_{\boldsymbol{\theta}} r(s_t,\mu_{\boldsymbol{\theta}}(s_t))\right\|_2\Bigg] \\
&\leqslant \sum_{t=0}^{T-1} L_\mu^2 L_p^2 R_{\max} t^2\gamma^t + (2L_\mu^2 L_p L_r + L_{2,\mu}L_{2,p}R_{\max})t\gamma^t + L_{2,\mu}L_{2,r}\gamma^t \\
&\leqslant L_\mu^2 L_p^2 R_{\max}\gamma\frac{1+\gamma-T^2\gamma^{T-1}+(2(T-1)^2+2(T-1)-1)\gamma^T-(T-1)^2\gamma^{T+1}}{(1-\gamma)^3} \\
&\qquad + (2L_\mu^2 L_p L_r + L_{2,\mu}L_{2,p}R_{\max})\gamma\frac{1-T\gamma^{T-1}+(T-1)\gamma^T}{(1-\gamma)^2} + L_{2,\mu}L_{2,r}\frac{1-\gamma^T}{1-\gamma} \\
&\leqslant L_\mu^2 L_p^2 R_{\max}\gamma\frac{1+\gamma-\gamma^T}{(1-\gamma)^3} + (2L_\mu^2 L_p L_r + L_{2,\mu}L_{2,p}R_{\max})\gamma\frac{1-\gamma^T}{(1-\gamma)^2} + L_{2,\mu}L_{2,r}\frac{1-\gamma^T}{1-\gamma} \\
&\leqslant \frac{\gamma(1+\gamma)L_\mu^2 L_p^2 R_{\max}}{(1-\gamma)^3} + \frac{\gamma(2L_\mu^2 L_p L_r + L_{2,\mu}L_{2,p}R_{\max})}{(1-\gamma)^2} + \frac{L_{2,\mu}L_{2,r}}{1-\gamma}.
\end{aligned}$$

□

Lemma E.3. *Let $\pi_{\boldsymbol{\theta}}$ be a white noise-based policy. Under Assumption 4.3, 4.4, and 4.5 it holds that for every $s\in\mathcal{S}$:*

(i) $\mathbb{E}_{\mathbf{a}\sim\pi_{\boldsymbol{\theta}}(\mathbf{a}|\mathbf{s})}[\|\nabla_{\boldsymbol{\theta}}\log\pi_{\boldsymbol{\theta}}(\mathbf{a}|\mathbf{s})\|_2^2] \leqslant c d_{\mathcal{A}}\sigma_A^{-2}L_\mu^2$;

(ii) $\mathbb{E}_{\mathbf{a}\sim\pi_{\boldsymbol{\theta}}(\mathbf{a}|\mathbf{s})}[\|\nabla^2_{\boldsymbol{\theta}}\log\pi_{\boldsymbol{\theta}}(\mathbf{a}|\mathbf{s})\|_2] \leqslant c\sigma_A^{-2}L_\mu^2 + c\sqrt{d_{\mathcal{A}}}\sigma_A^{-1}L_{2,\mu}$.

Proof. Since $\pi_{\boldsymbol{\theta}}$ is a white noise-based policy, we have that $\pi_{\boldsymbol{\theta}}(\mathbf{a}|\mathbf{s}) = \phi(\mathbf{a} - \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s}))$. Consequently, we have:

$$\nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(\mathbf{a}|\mathbf{s}) = \nabla_{\boldsymbol{\theta}} \log \phi(\mathbf{a} - \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s})) = -\nabla_{\boldsymbol{\theta}} \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s}) \nabla_{\boldsymbol{\epsilon}} \log \phi(\boldsymbol{\epsilon})|_{\boldsymbol{\epsilon} = \mathbf{a} - \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s})}, \tag{97}$$

$$\nabla^2_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(\mathbf{a}|\mathbf{s}) = \nabla^2_{\boldsymbol{\theta}} \log \phi(\mathbf{a} - \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s})) = \nabla_{\boldsymbol{\theta}} \boldsymbol{\mu}_{\boldsymbol{\theta}} \nabla^2_{\boldsymbol{\epsilon}} \log \phi(\boldsymbol{\epsilon})|_{\boldsymbol{\epsilon} = \mathbf{a} - \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s})} \nabla_{\boldsymbol{\theta}} \boldsymbol{\mu}_{\boldsymbol{\theta}}^\top - \nabla^2_{\boldsymbol{\theta}} \boldsymbol{\mu}_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\epsilon}} \log \phi(\boldsymbol{\epsilon})|_{\boldsymbol{\epsilon} = \mathbf{a} - \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s})}. \tag{98}$$

Thus, recalling that $\mathbf{a} - \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s}) \sim \Phi_{d_{\mathcal{A}}}$ and using the Lipschitzinity and smoothness of $\boldsymbol{\mu}_{\boldsymbol{\theta}}$, we have:

$$\mathop{\mathbb{E}}_{\mathbf{a} \sim \pi_{\boldsymbol{\theta}}(\mathbf{a}|\mathbf{s})} [\|\nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(\mathbf{a}|\mathbf{s})\|_2^2] = \mathop{\mathbb{E}}_{\mathbf{a} \sim \pi_{\boldsymbol{\theta}}(\mathbf{a}|\mathbf{s})} [\|\nabla_{\boldsymbol{\theta}} \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s}) \nabla_{\boldsymbol{\epsilon}} \log \phi(\boldsymbol{\epsilon})|_{\boldsymbol{\epsilon} = \mathbf{a} - \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s})}\|_2^2] \tag{99}$$

$$\leqslant L_\mu^2 \mathop{\mathbb{E}}_{\boldsymbol{\epsilon} \sim \Phi_{d_\Theta}} [\| - \nabla_{\boldsymbol{\epsilon}} \log \phi(\boldsymbol{\epsilon})\|_2^2] \leqslant c d_\Theta \sigma_{\mathrm{A}}^2 L_\mu^2, \tag{100}$$

$$\mathop{\mathbb{E}}_{\mathbf{a} \sim \pi_{\boldsymbol{\theta}}(\mathbf{a}|\mathbf{s})} [\|\nabla^2_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(\mathbf{a}|\mathbf{s})\|_2] = \mathop{\mathbb{E}}_{\mathbf{a} \sim \pi_{\boldsymbol{\theta}}(\mathbf{a}|\mathbf{s})} [\|\nabla_{\boldsymbol{\theta}} \boldsymbol{\mu}_{\boldsymbol{\theta}} \nabla^2_{\boldsymbol{\epsilon}} \log \phi(\boldsymbol{\epsilon})|_{\boldsymbol{\epsilon} = \mathbf{a} - \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s})} \nabla_{\boldsymbol{\theta}} \boldsymbol{\mu}_{\boldsymbol{\theta}}^\top - \nabla^2_{\boldsymbol{\theta}} \boldsymbol{\mu}_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\epsilon}} \log \phi(\boldsymbol{\epsilon})|_{\boldsymbol{\epsilon} = \mathbf{a} - \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s})}\|_2] \tag{101}$$

$$\leqslant L_\mu^2 \mathop{\mathbb{E}}_{\boldsymbol{\epsilon} \sim \Phi_{d_{\mathcal{A}}}} [\|\nabla^2_{\boldsymbol{\epsilon}} \log \phi(\boldsymbol{\epsilon})\|_2] + L_{2,\mu} \mathop{\mathbb{E}}_{\boldsymbol{\epsilon} \sim \Phi_{d_{\mathcal{A}}}} [\|\nabla_{\boldsymbol{\epsilon}} \log \phi(\boldsymbol{\epsilon})\|_2] \tag{102}$$

$$\leqslant c \sigma_{\mathrm{A}}^{-2} L_\mu^2 + c \sqrt{d_{\mathcal{A}}} \sigma_{\mathrm{A}}^{-1} L_{2,\mu}. \tag{103}$$

□

Lemma E.4. *Let $\nu_{\boldsymbol{\theta}}$ be a white noise-based hyperpolicy. Under Assumption 4.5, it holds that:*

(i) $\mathbb{E}_{\boldsymbol{\theta}' \sim \nu_{\boldsymbol{\theta}}} [\|\nabla_{\boldsymbol{\theta}} \log \nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}')\|_2^2] \leqslant c d_\Theta \sigma_P^{-2}$;
(ii) $\mathbb{E}_{\boldsymbol{\theta}' \sim \nu_{\boldsymbol{\theta}}} [\|\nabla^2_{\boldsymbol{\theta}} \log \nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}')\|_2] \leqslant c \sigma_P^{-2}$.

Proof. Since $\nu_{\boldsymbol{\theta}}$ is a white noise-based hyperpolicy, we have that $\nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}') = \phi(\boldsymbol{\theta}' - \boldsymbol{\theta})$. Consequently, we have:

$$\nabla_{\boldsymbol{\theta}} \log \nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}') = \nabla_{\boldsymbol{\theta}} \log \phi(\boldsymbol{\theta}' - \boldsymbol{\theta}) = -\nabla_{\boldsymbol{\epsilon}} \log \phi(\boldsymbol{\epsilon})|_{\boldsymbol{\epsilon} = \boldsymbol{\theta}' - \boldsymbol{\theta}}, \tag{104}$$

$$\nabla^2_{\boldsymbol{\theta}} \log \nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}') = \nabla^2_{\boldsymbol{\theta}} \log \phi(\boldsymbol{\theta}' - \boldsymbol{\theta}) = \nabla^2_{\boldsymbol{\epsilon}} \log \phi(\boldsymbol{\epsilon})|_{\boldsymbol{\epsilon} = \boldsymbol{\theta}' - \boldsymbol{\theta}}. \tag{105}$$

Thus, recalling that $\boldsymbol{\theta}' - \boldsymbol{\theta} \sim \Phi_{d_\Theta}$

$$\mathop{\mathbb{E}}_{\boldsymbol{\theta}' \sim \nu_{\boldsymbol{\theta}}} [\|\nabla_{\boldsymbol{\theta}} \log \nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}')\|_2^2] = \mathop{\mathbb{E}}_{\boldsymbol{\theta}' \sim \nu_{\boldsymbol{\theta}}} [\|\nabla_{\boldsymbol{\epsilon}} \log \phi(\boldsymbol{\epsilon})|_{\boldsymbol{\epsilon} = \boldsymbol{\theta}' - \boldsymbol{\theta}}\|_2^2] = \mathop{\mathbb{E}}_{\boldsymbol{\epsilon} \sim \Phi_{d_\Theta}} [\|\nabla_{\boldsymbol{\epsilon}} \log \phi(\boldsymbol{\epsilon})\|_2^2] \leqslant c d_\Theta \sigma_{\mathrm{P}}^2, \tag{106}$$

$$\mathop{\mathbb{E}}_{\boldsymbol{\theta}' \sim \nu_{\boldsymbol{\theta}}} [\|\nabla^2_{\boldsymbol{\theta}} \log \nu_{\boldsymbol{\theta}}(\boldsymbol{\theta}')\|_2] = \mathop{\mathbb{E}}_{\boldsymbol{\theta}' \sim \nu_{\boldsymbol{\theta}}} [\|\nabla^2_{\boldsymbol{\epsilon}} \log \phi(\boldsymbol{\epsilon})|_{\boldsymbol{\epsilon} = \boldsymbol{\theta}' - \boldsymbol{\theta}}\|_2] = \mathop{\mathbb{E}}_{\boldsymbol{\epsilon} \sim \Phi_{d_\Theta}} [\|\nabla^2_{\boldsymbol{\epsilon}} \log \phi(\boldsymbol{\epsilon})\|_2] \leqslant c \sigma_{\mathrm{P}}^2. \tag{107}$$

□

F. General Convergence Analysis under Weak Gradient Domination

In this section, we provide the theoretical guarantees on the convergence to the global optimum of a generic stochastic first-order optimization algorithm $\mathfrak{A}$ (e.g., policy gradient employing either AB or PB exploration). Let $\boldsymbol{\theta}$ be the parameter vector optimized by $\mathfrak{A}$, and let $\Theta = \mathbb{R}^{d_\Theta}$ be the parameter space. The objective function that $\mathfrak{A}$ aims at optimizing is $J : \Theta \to \mathbb{R}$, which is a generic function taking as argument a parameter vector $\boldsymbol{\theta} \in \Theta$ and mapping it into a real value. Examples of objective functions of this kind are J_{D}, J_{A}, or J_{P}, which are all defined in Section 2. The algorithm $\mathfrak{A}$ is run for K iterations and it updates directly the parameter vector $\boldsymbol{\theta} \in \Theta$. At the k-th iteration, the update is:

$$\boldsymbol{\theta}_{k+1} \leftarrow \boldsymbol{\theta}_k + \zeta_k \widehat{\nabla}_{\boldsymbol{\theta}} J(\boldsymbol{\theta}_k),$$

where ζ_k is the step size, $\boldsymbol{\theta}_k$ is the parameter configuration at the k-th iteration, and $\widehat{\nabla}_{\boldsymbol{\theta}} J(\boldsymbol{\theta}_k)$ is an *unbiased* estimate of $\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}_k)$ computed from a batch $\mathcal{D}_k$ of N *samples*. In the following, we refer to N as *batch size*. Examples of unbiased gradient estimators are the ones employed by GPOMDP and PGPE, which can be found in Section 2. For GPOMDP, samples are trajectories; for PGPE, parameter-trajectory pairs. In what follows, we refer to the optimal parameter configuration as $\boldsymbol{\theta}^* \in \arg\max_{\boldsymbol{\theta} \in \Theta} J(\boldsymbol{\theta})$. For the sake of simplicity, we will shorten $J(\boldsymbol{\theta}^*)$ as J^*. Given an optimality threshold $\delta \geqslant 0$, we are interested in assessing the *last-iterate* convergence guarantees:

$$J^* - \mathbb{E}[J(\boldsymbol{\theta}_K)] \leqslant \delta,$$

where the expectation is taken over the stochasticity of the learning process.

Theorem F.1. *Under Assumptions 6.1, 6.2, and 6.3, running the Algorithm $\mathfrak{A}$ for $K > 0$ iterations with a batch size of $N > 0$ trajectories in each iteration with the constant learning rate ζ fulfilling:*

$$\zeta \leqslant \min \left\{ \frac{1}{L_2}, \frac{1}{\mu \max\{0, J^* - J(\boldsymbol{\theta}_0) - \beta\}}, \left(\frac{N}{L_2 V \mu} \right)^{1/3} \right\}$$

where $\mu = \frac{1}{\alpha^2}$*. Then, it holds that:*

$$J^* - \mathbb{E}[J(\boldsymbol{\theta}_K)] \leqslant \beta + \left(1 - \frac{1}{2}\sqrt{\frac{\mu\zeta^3 L_2 V}{N}}\right)^K \max\{0, J^* - J(\boldsymbol{\theta}_0) - \beta\} + \sqrt{\frac{L_2 V \zeta}{\mu N}}.$$

In particular, for sufficiently small $\epsilon > 0$*, setting* $\zeta = \frac{\epsilon^2 \mu N}{4 L_2 V}$*, the following total number of samples is sufficient to ensure that* $J(\boldsymbol{\theta}^*) - \mathbb{E}[J(\boldsymbol{\theta}_K)] \leqslant \beta + \epsilon$*:*

$$KN \geqslant \frac{16 L_2 V}{\epsilon^3 \mu^2} \log \frac{\max\{0, J^* - J(\boldsymbol{\theta}_0) - \beta\}}{\epsilon}. \tag{108}$$

Proof. Before starting the proof, we need a preliminary result that immediately follows from Assumption 6.1, by rearranging:

$$\frac{1}{\alpha}\max\{0, J(\boldsymbol{\theta}^*) - \beta - J(\boldsymbol{\theta})\} \leqslant \|\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta})\|_2, \tag{109}$$

and we will use the notation $\widetilde{J}(\boldsymbol{\theta}^*) := J(\boldsymbol{\theta}^*) - \beta$ and $\mu = \alpha^{-2}$. Note that $\widetilde{J}(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta})$ can be negative. Considering a $k \in [\![K]\!]$, it follows that:

$$\begin{aligned}
\widetilde{J}(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta}_{k+1}) &= \widetilde{J}(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta}_k) - (J(\boldsymbol{\theta}_{k+1}) - J(\boldsymbol{\theta}_k)) \\
&\leqslant \widetilde{J}(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta}_k) - \langle \boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}_k, \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}_k)\rangle + \frac{L_2}{2}\|\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}_k\|_2^2 \\
&\leqslant \widetilde{J}(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta}_k) - \zeta_k \left\langle \widehat{\nabla}_{\boldsymbol{\theta}} J(\boldsymbol{\theta}_k), \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}_k)\right\rangle + \frac{L_2}{2}\zeta_k^2 \|\widehat{\nabla}_{\boldsymbol{\theta}} J(\boldsymbol{\theta}_k)\|_2^2,
\end{aligned}$$

where the first inequality follows by applying the Taylor expansion with Lagrange remainder and exploiting Assumption 6.2, and the last inequality follows from the fact that the parameter update is $\boldsymbol{\theta}_{k+1} \leftarrow \boldsymbol{\theta}_k + \zeta_k \widehat{\nabla}_{\boldsymbol{\theta}} J(\boldsymbol{\theta}_k)$.

In the following, we use the shorthand notation $\mathbb{E}_k[\cdot]$ to denote the conditional expectation w.r.t. the history up to the k-th iteration *not* included. More formally, let $\mathcal{F}_k = \sigma(\boldsymbol{\theta}_0, \mathcal{D}_0, \mathcal{D}_1, \dots, \mathcal{D}_k)$ be the σ-algebra encoding all the stochasticity up to iteration k included. Note that all the stochasticity comes from the samples (except from the initial parameter $\boldsymbol{\theta}_0$, which may be randomly initialized), and that $\boldsymbol{\theta}_k$ is $\mathcal{F}_{k-1}$-measurable, that is, deterministically determined by the realization of the samples collected in the first $k-1$ iterations. Then, $\mathbb{E}_k[\cdot] := \mathbb{E}[\cdot|\mathcal{F}_{k-1}]$. We will make use of the basic facts $\mathbb{E}[\cdot] = \mathbb{E}[\mathbb{E}_k[\cdot]]$ and $\mathbb{E}_k[X] = X$ for $\mathcal{F}_{k-1}$-measurable X. The variance of $\widehat{\nabla} J(\boldsymbol{\theta}_k)$ must be always understood as conditional on $\mathcal{F}_{k-1}$. Now, for any $k \in [\![K]\!]$:

$$\begin{aligned}
\mathop{\mathbb{E}}_k\left[\widetilde{J}(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta}_{k+1})\right] &\leqslant \mathop{\mathbb{E}}_k\left[\widetilde{J}(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta}_k) - \zeta_k \left\langle \widehat{\nabla}_{\boldsymbol{\theta}} J(\boldsymbol{\theta}_k), \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}_k)\right\rangle + \frac{L_2}{2}\zeta_k^2 \|\widehat{\nabla}_{\boldsymbol{\theta}} J(\boldsymbol{\theta}_k)\|_2^2\right] \\
&\leqslant \widetilde{J}(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta}_k) - \zeta_k \|\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}_k)\|_2^2 + \frac{L_2}{2}\zeta_k^2 \mathop{\mathbb{E}}_k\left[\|\widehat{\nabla}_{\boldsymbol{\theta}} J(\boldsymbol{\theta}_k)\|_2^2\right] \\
&\leqslant \widetilde{J}(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta}_k) - \zeta_k\left(1 - \frac{L_2}{2}\zeta_k\right)\|\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}_k)\|_2^2 + \frac{L_2}{2}\zeta_k^2 \mathbb{V}\mathrm{ar}\left[\widehat{\nabla}_{\boldsymbol{\theta}} J(\boldsymbol{\theta}_k)\right] \\
&\leqslant \widetilde{J}(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta}_k) - \zeta_k\left(1 - \frac{L_2}{2}\zeta_k\right)\|\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}_k)\|_2^2 + \frac{L_2 V}{2N}\zeta_k^2,
\end{aligned}$$

where the third inequality follows from the fact that $\widehat{\nabla}_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$ is an unbiased estimator and from the definition of $\mathbb{V}\mathrm{ar}[\widehat{\nabla} J(\boldsymbol{\theta})]$, and the last inequality is by Assumption 6.3. Now, selecting a step size $\zeta_k \leqslant \frac{1}{L_2}$, we have that $1 - \frac{L_2}{2}\zeta_k \geqslant \frac{1}{2}$, we can use the bound derived in Equation (109):

$$\mathop{\mathbb{E}}_k\left[\widetilde{J}(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta}_{k+1})\right] \leqslant \widetilde{J}(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta}_k) - \frac{\mu\zeta_k}{2}\max\left\{0, \widetilde{J}(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta}_k)\right\}^2 + \frac{L_2 V}{2N}\zeta_k^2.$$

The next step is to consider the total expectation over both the terms of the inequality and observe that

$$\mathbb{E}\left[\max\left\{0, \widetilde{J}(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta}_k)\right\}^2\right] \geqslant \mathbb{E}\left[\max\left\{0, \widetilde{J}(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta}_k)\right\}\right]^2 \geqslant \max\left\{0, \mathbb{E}\left[\widetilde{J}(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta}_k)\right]\right\}^2,$$

having applied Jensen's inequality twice, being both the square and the $\max$ convex functions. In particular, we define $r_k := \mathbb{E}[J(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta}_k)]$. We can then rewrite the previous inequality as follows:

$$r_{k+1} \leqslant r_k - \frac{\mu\zeta_k}{2}\max\{0, r_k\}^2 + \frac{L_2 V}{2N}\zeta_k^2.$$

To study the recurrence, we define the helper sequence:

$$\begin{cases} \rho_0 = r_0 \\ \rho_{k+1} = \rho_k - \frac{\mu\zeta_k}{2}\max\{0,\rho_k\}^2 + \frac{L_2 V}{2N}\zeta_k^2 \quad \text{if } k \geqslant 0 \end{cases}. \tag{110}$$

We now show that under a suitable condition on the step size ζ_k, the sequence ρ_k upper bounds the sequence r_k.

Lemma F.2. *If $\zeta_k \leqslant \frac{1}{\mu\rho_k}$ for every $k \geqslant 0$, then, $r_k \leqslant \rho_k$ for every $k \geqslant 0$.*

Proof of Lemma F.2. By induction on k. For $k=0$, the statement holds since $\rho_0 = r_0$. Suppose the statement holds for every $j \leqslant k$, we prove that it holds for $k+1$:

$$\rho_{k+1} = \rho_k - \frac{\mu\zeta_k}{2}\max\{0,\rho_k\}^2 + \frac{L_2 V}{2N}\zeta_k^2 \tag{111}$$

$$\geqslant r_k - \frac{\mu\zeta_k}{2}\max\{0,r_k\}^2 + \frac{L_2 V}{2N}\zeta_k^2 \tag{112}$$

$$\geqslant r_{k+1}. \tag{113}$$

where the first inequality holds by the inductive hypothesis and by observing that the function $f(x) = x - \frac{\mu\zeta_k}{2}\max\{0,x\}^2$ is non-decreasing in x when $\zeta_k \leqslant 1/(\mu x)$. Indeed, if $x < 0$, then $f(x) = x$, which is non-decreasing; if $x \geqslant 0$, we have $f(x) = x - \frac{\mu\zeta_k}{2}x^2$, that can be proved to be non-decreasing in the interval $[0, 1/(\mu\zeta_k)]$ simply by studying the sign of the derivative. The requirement $\zeta_k \leqslant 1/(\mu\rho_k)$ ensures that ρ_k falls in the non-decreasing region, and so does r_k by the inductive hypothesis. □

Thus, from now on, we study the properties of the sequence ρ_k and enforce the learning rate to be constant, $\zeta_k := \zeta$ for every $k \geqslant 0$. Let us note that, if ρ_k is convergent, than it converges to the fixed-point $\overline{\rho}$ computed as follows:

$$\overline{\rho} = \overline{\rho} - \frac{\mu\zeta}{2}\max\{0,\overline{\rho}\}^2 + \frac{L_2 V}{2N}\zeta^2 \Longrightarrow \overline{\rho} = \sqrt{\frac{L_2 V \zeta}{\mu N}}, \tag{114}$$

having retained the positive solution of the second-order equation only, since the negative one never attains the maximum $\max\{0,\overline{\rho}\}$. Let us now study the monotonicity properties of the sequence ρ_k.

Lemma F.3. *The following statements hold:*

- *If $r_0 > \overline{\rho}$ and $\zeta \leqslant \frac{1}{\mu r_0}$, then for every $k \geqslant 0$ it holds that: $\overline{\rho} \leqslant \rho_{k+1} \leqslant \rho_k$.*
- *If $r_0 < \overline{\rho}$ and $\zeta \leqslant \frac{1}{\mu\overline{\rho}}$, then for every $k \geqslant 0$ it holds that: $\overline{\rho} \geqslant \rho_{k+1} \geqslant \rho_k$.*

Before proving the lemma, let us comment on it. We have stated that if we initialize the sequence with $\rho_0 = r_0$ above the fixed-point $\overline{\rho}$, the sequence is non-increasing and remains in the interval $[\overline{\rho}, r_0]$. Symmetrically, if we initialize $\rho_0 = r_0$ (possibly negative) below the fixed-point $\overline{\rho}$, the sequence is non-decreasing and remains in the interval $[r_0, \overline{\rho}]$. These properties hold under specific conditions on the learning rate.

Proof of Lemma F.3. We first prove the first statement, by induction on k. The inductive hypothesis is "$\rho_{k+1} \leqslant \rho_k$ *and* $\rho_{k+1} \geqslant \overline{\rho}$ ". For $k=0$, for the first inequality, we have:

$$\rho_1 = \rho_0 - \frac{\zeta\mu}{2}\rho_0^2 + \frac{L_2 V}{2N}\zeta^2 \leqslant \rho_0 - \frac{\zeta\mu}{2}\overline{\rho}^2 + \frac{L_2 V}{2N}\zeta^2 = \rho_0, \tag{115}$$

having exploited the fact that $\rho_0 > \overline{\rho} > 0$ and the definition of $\overline{\rho}$. For the second inequality, we have:

$$\rho_1 = \rho_0 - \frac{\zeta\mu}{2}\rho_0^2 + \frac{L_2 V}{2N}\zeta^2 \geqslant \overline{\rho} - \frac{\zeta\mu}{2}\overline{\rho}^2 + \frac{L_2 V}{2N}\zeta^2 = \overline{\rho}, \tag{116}$$

recalling that the function $x - \frac{\zeta\mu}{2}x^2$ is non-decreasing in x for $x \leqslant \rho_0$ since $\zeta \leqslant 1/(\mu\rho_0)$, and by definition of $\overline{\rho}$. Suppose now that the statement holds for every $j < k$. First of all, we observe that, under this inductive hypothesis, $\rho_k \leqslant \rho_0$ and, consequently, the condition $\zeta \leqslant 1/(\mu\rho_0)$ entails $\zeta \leqslant 1/(\mu\rho_k)$. Thus, for the first inequality, we have:

$$\rho_{k+1} = \rho_k - \frac{\zeta\mu}{2}\rho_k^2 + \frac{L_2 V}{2N}\zeta^2 \leqslant \rho_k - \frac{\zeta\mu}{2}\overline{\rho}^2 + \frac{L_2 V}{2N}\zeta^2 = \rho_k, \tag{117}$$

having used the inductive hypothesis and the definition of $\overline{\rho}$. For the second inequality, we have:

$$\rho_{k+1} = \rho_k - \frac{\zeta\mu}{2}\rho_k^2 + \frac{L_2 V}{2N}\zeta^2 \geqslant \overline{\rho} - \frac{\zeta\mu}{2}\overline{\rho}^2 + \frac{L_2 V}{2N}\zeta^2 = \overline{\rho}, \tag{118}$$

having used the inductive hypothesis and recalled that the function $x - \frac{\zeta\mu}{2}x^2$ is non-decreasing in x for $x \leqslant \rho_k$ since $\zeta \leqslant 1/(\mu\rho_k)$.

For the second statement, we observe that if $\rho_0 = r_0 < 0$, we have:

$$\rho_k = \rho_0 + k\frac{L_2 V}{2N}\zeta^2, \tag{119}$$

for all $k \leqslant k^*$, where k^* is the minimum k in which $\rho_0 + k\frac{L_2 V}{2N}\zeta^2 \geqslant 0$. From that point on, we can proceed in an analogous way as for the first statement, simply switching the signs of the inequalities and recalling that the largest value of ρ_k is bounded by $\overline{\rho}$ in this case. □

We now focus on the first case of the previous lemma in which $r_0 > \overline{\rho}$, as the second one, as we shall see later, is irrelevant for the convergence rate. We now want to show that the sequence ρ_k actually converges to $\overline{\rho}$ and characterize its convergence rate. To this end, we introduce a new auxiliary sequence:

$$\begin{cases} \eta_0 = \rho_0 \\ \eta_{k+1} = \left(1 - \frac{\mu\zeta\overline{\rho}}{2}\right)\eta_k + \frac{L_2 V}{2N}\zeta^2 \quad \text{if } k \geqslant 0 \end{cases}. \tag{120}$$

We show that the sequence η_k upper bounds ρ_k when $\rho_0 = r_0 \geqslant \overline{\rho}$.

Lemma F.4. *If $r_0 > \overline{\rho}$ and $\zeta \leqslant \frac{1}{\mu r_0}$, then, for every $k \geqslant 0$, it holds that $\eta_k \geqslant \rho_k$.*

Proof. By induction on k. For $k = 0$, we have $\eta_0 = \rho_0$, so, the statement holds. Suppose the statement holds for every $j \leqslant k$, we prove it for $k+1$:

$$\eta_{k+1} = \left(1 - \frac{\mu\zeta\overline{\rho}}{2}\right)\eta_k + \frac{L_2 V}{2N}\zeta^2 \tag{121}$$

$$\geqslant \left(1 - \frac{\mu\zeta\rho_k}{2}\right)\eta_k + \frac{L_2 V}{2N}\zeta^2 \tag{122}$$

$$\geqslant \left(1 - \frac{\mu\zeta\rho_k}{2}\right)\rho_k + \frac{L_2 V}{2N}\zeta^2 \tag{123}$$

$$= \rho_k - \frac{\zeta\mu}{2}\max\{0, \rho_k\}^2 + \frac{L_2 V}{2N}\zeta^2 = \rho_{k+1}. \tag{124}$$

having exploited that $\rho_k \geqslant \overline{\rho}$ (by Lemma F.3) in the second line; using the inductive hypothesis in the third line, exploiting the fact that $1 - \frac{\mu\zeta\rho_k}{2} \geqslant 0$ whenever $\zeta \leqslant 2/(\mu\rho_k)$, which is entailed by the requirement $\zeta \leqslant 1/(\mu\rho_0)$; and by recalling that $\rho_k > 0$ since $\overline{\rho} > 0$ in the last line. □

Thus, we conclude by studying the convergence rate of the sequence η_k. This can be easily obtained by unrolling the recursion:

$$\eta_{k+1} = \left(1 - \frac{\mu\zeta\overline{\rho}}{2}\right)^{k+1}\eta_0 + \frac{L_2 V\zeta^2}{2N}\sum_{j=0}^{k}\left(1 - \frac{\mu\zeta\overline{\rho}}{2}\right)^j \tag{125}$$

$$\leqslant \left(1 - \frac{\mu\zeta\overline{\rho}}{2}\right)^{k+1}\eta_0 + \frac{L_2 V\zeta^2}{2N}\sum_{j=0}^{+\infty}\left(1 - \frac{\mu\zeta\overline{\rho}}{2}\right)^j \tag{126}$$

$$= \left(1 - \frac{\mu\zeta\overline{\rho}}{2}\right)^{k+1}\eta_0 + \frac{L_2 V\zeta}{N\mu\overline{\rho}} \tag{127}$$

$$= \left(1 - \frac{1}{2}\sqrt{\frac{\mu\zeta^3 L_2 V}{N}}\right)^{k+1}\eta_0 + \sqrt{\frac{L_2 V\zeta}{\mu N}}. \tag{128}$$

Putting all the conditions on the step size ζ together, we must set:

$$\zeta = \min\left\{\frac{1}{L_2}, \frac{1}{\mu\max\{0, r_0\}}, \left(\frac{N}{L_2 V\mu}\right)^{1/3}\right\}. \tag{129}$$

Thus, we have:

$$J(\boldsymbol{\theta}^*) - \mathbb{E}[J(\boldsymbol{\theta}_K)] \leqslant \beta + \left(1 - \frac{1}{2}\sqrt{\frac{\mu\zeta^3 L_2 V}{N}}\right)^K \max\{0, J(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta}_0) - \beta\} + \sqrt{\frac{L_2 V \zeta}{\mu N}}. \tag{130}$$

We derive the number of iterations (setting $K \leftarrow k+1$):

$$\left(1 - \frac{1}{2}\sqrt{\frac{\mu\zeta^3 L_2 V}{N}}\right)^K \eta_0 \leqslant \frac{\epsilon}{2} \Longrightarrow K \leqslant \frac{\log \frac{2\eta_0}{\epsilon}}{\log \frac{1}{1 - \frac{1}{2}\sqrt{\frac{\mu\zeta^3 L_2 V}{N}}}} \leqslant \sqrt{\frac{4N}{\mu\zeta^3 L_2 V}} \log \frac{2\eta_0}{\epsilon}, \tag{131}$$

having exploited the inequality $\log \frac{1}{1-x} \geqslant x$. Furthermore, let us observe that:

$$\overline{\rho} = \sqrt{\frac{L_2 V \zeta}{\mu N}} \leqslant \frac{\epsilon}{2} \Longrightarrow \zeta \leqslant \frac{\epsilon^2 \mu N}{4 L_2 V}. \tag{132}$$

Thus, recalling that $\rho_0 = \eta_0 = r_0$, we have that: (i) when $r_0 < \overline{\rho}$, we have that $r_k \leqslant \rho_k \leqslant \overline{\rho} \leqslant \epsilon/2$; (ii) when $r_0 \geqslant \overline{\rho}$, we have $r_k \leqslant \rho_k \leqslant \eta_k \leqslant \epsilon$. Thus, for sufficiently small ϵ, we plug $\zeta = \frac{\epsilon^2 \mu N}{4 L_2 V}$ in Equation (131) to obtain the following upper bound on the sample complexity:

$$KN \leqslant \frac{16 L_2 V}{\epsilon^3 \mu^2} \log \frac{\max\{0, J(\boldsymbol{\theta}^*) - J(\boldsymbol{\theta}_0) - \beta\}}{\epsilon}, \tag{133}$$

which guarantees $J(\boldsymbol{\theta}^*) - \mathbb{E}[J(\boldsymbol{\theta}_K)] \leqslant \beta + \epsilon$. □

Theorem 6.1. *Consider an algorithm running the update rule of Equation (13). Under Assumptions 6.1, 6.2, and 6.3, with a suitable constant step size, to guarantee $J_\dagger^* - \mathbb{E}[J_\dagger(\boldsymbol{\theta}_K)] \leqslant \epsilon + \beta$ the sample complexity is at most:*

$$NK = \frac{16 \alpha^4 L_{2,\dagger} V_\dagger}{\epsilon^3} \log \frac{\max\{0, J_\dagger^* - J_\dagger(\boldsymbol{\theta}_0) - \beta\}}{\epsilon}. \tag{15}$$

Proof. Directly follows from the second statement of Theorem F.1. □

G. Specifications of the Algorithms

PGPE. In this section we report the algorithm PGPE as it is reported in its original paper (Sehnke et al., 2010). In particular, we show the pseudo-code (Algorithm 1) of its original basic version, that is also the one we analyzed throughout this work, even if several variants are available.

Algorithm 1 PGPE.

Input : Number of iterations K, batch size N, initial parameter vector $\boldsymbol{\theta}_0$, environment $\mathcal{M}$, deterministic policy $\mu_{\boldsymbol{\theta}}$, hyperpolicy $\nu_{\boldsymbol{\theta}}$, step size schedule $(\zeta_k)_{k=0}^{K-1}$, exploration parameter σ_{P}.

Initialize $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta}_0$

for $i \in [\![K]\!]$ **do**

  Set the hyperpolicy parameters: $\nu_{\boldsymbol{\theta}}$

  for $l \in [\![N]\!]$ **do**

    Sample a parameter configuration $\boldsymbol{\rho}_l \sim \nu_{\boldsymbol{\theta}}$ according to the exploration parameter σ_{P}

    Collect a trajectory τ_l by acting in $\mathcal{M}$ with $\mu_{\boldsymbol{\rho}_l}$

    Compute the cumulative discounted reward $R(\tau_l)$

  end

  Compute the gradient estimator: $\widehat{\nabla}_{\boldsymbol{\theta}} J_{\text{P}}(\boldsymbol{\theta}) \leftarrow \frac{1}{N} \sum_{j=0}^{N-1} \nabla_{\boldsymbol{\theta}} \log \nu_{\boldsymbol{\theta}}(\boldsymbol{\rho}_j) R(\tau_j)$

  Update the hyperpolicy parameter vector: $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \zeta_i \widehat{\nabla}_{\boldsymbol{\theta}} J_{\text{P}}(\boldsymbol{\theta})$

end

Return $\boldsymbol{\theta}$.

Notice that, the original version of PGPE by Sehnke et al. (2010) considers to collect M trajectories for each parameter configuration $\boldsymbol{\rho}$ sampled from the hyperpolicy $\nu_{\boldsymbol{\theta}}$. In the pseudo-code (as well as in the paper) we consider $M = 1$ (i.e., we collect a single trajectory) in order to make GPOMDP and PGPE testing the same number of trajectories in each iteration, given an equal batch size N. In the original paper also other variants of PGPE are considered, that we have not considered in or work. For instance, the one with symmetric sampling, or the one employing a baseline while sampling. Moreover, it would be possible to learn a proper exploration amount σ_{P} while learning the hyperpolicy parameters, however we decided to keep σ_{P} fixed, for reasons remarked in Appendix C.

GPOMDP. As done for PGPE, here we report the algorithm GPOMDP in its original version (Baxter & Bartlett, 2001; Peters & Schaal, 2006). We show the pseudo-code (Algorithm 2) of such original basic version, that is also the one we analyzed throughout this work.

Algorithm 2 GPOMDP.

Input : Number of iterations K, batch size N, initial parameter vector $\boldsymbol{\theta}_0$, environment $\mathcal{M}$, stochastic policy $\pi_{\boldsymbol{\theta}}$ (with exploration parameter σ_{A}), step size schedule $(\zeta_k)_{k=0}^{K-1}$, , horizon T, discount factor γ.

Initialize $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta}_0$

for $i \in [\![K]\!]$ **do**

  Set the stochastic policy parameters: $\pi_{\boldsymbol{\theta}}$

  for $l \in [\![N]\!]$ **do**

    Initialize trajectory τ_l as an empty tuple

    for $t \in [\![T]\!]$ **do**

      Observe state s_t

      Play action $\mathbf{a}_t \sim \pi_{\boldsymbol{\theta}}(\cdot|s_t)$

      Observe reward r_t

      Add to τ_l the tuple (s_t, a_t, r_t)

    end

  end

  Compute the gradient estimator: $\widehat{\nabla}_{\boldsymbol{\theta}} J_{\text{A}}(\boldsymbol{\theta}) \leftarrow \frac{1}{N} \sum_{i=1}^{N} \sum_{t=0}^{T-1} \left(\sum_{k=0}^{t} \nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(\mathbf{a}_{\tau_i,k} | \mathbf{s}_{\tau_i,k}) \right) \gamma^t r(\mathbf{s}_{\tau_i,t}, \mathbf{a}_{\tau_i,t})$

  Update the policy parameter vector: $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \zeta_i \widehat{\nabla}_{\boldsymbol{\theta}} J_{\text{A}}(\boldsymbol{\theta})$

end

Return $\boldsymbol{\theta}$.

In the original paper, it is available a variant of GPOMDP which employs baselines while sampling, but in our work we do not consider this approach, as for PGPE. Also in this case, we decided to employ a fixed value for σ_{A}, even if it would be possible to adapt it at runtime (Appendix C).

H. Additional Experimental Results

In this section, we present additional experimental results for what concerns the comparison of GPOMDP and PGPE, and the sensitivity analysis on the exploration parameters, respectively σ_{A} and σ_{P}.

H.1. Learning Curves of the Variance Study of Section 9.

Setting. We show the results gained by learning in three environments of increasing complexity taken from the MuJoCo (Todorov et al., 2012) suite: *Swimmer-v4*, *Hopper-v4*, and *HalfCheetah-v4*. Details on the environmental parameters are shown in Table 3. In order to facilitate the exploration, thus highlighting the results of the sensitivity study on the exploration parameters, we added an action clipping to the environments.[10] The target *deterministic policy* $\boldsymbol{\mu_\theta}$ is linear in the state, while the *hyperpolicy* $\boldsymbol{\nu_\theta}$ employed by PGPE is Gaussian with a parameterized mean, and the *stochastic policy* $\boldsymbol{\pi_\theta}$ employed by GPOMDP is Gaussian with a mean linear in the state. Both PGPE and GPOMDP were run for $K = 2000$ iterations, generating $N = 100$ trajectories per iteration. We conducted a sensitivity analysis on the exploration parameters, using $\{0.01, 0.1, 1, 10, 100\}$ as values for σ_{P}^2 and σ_{A}^2. We employed Adam (Kingma & Ba, 2014) to set the step size with initial values 0.1 for PGPE and 0.01 for GPOMDP. The latter does not support a larger step size due to the higher variance of the employed estimator w.r.t. the one used by PGPE.

Environment	T	γ	$d_{\mathcal{S}}$	$d_{\mathcal{A}}$	d_Θ
Swimmer	200	1	8	2	16
Hopper	100	1	11	3	33
HalfCheetah	100	1	17	6	102

Table 3. Parameters of the environments.

Here we show the learning curves of J_{P} and J_{A} (and the associated empirical J_{D}) obtained in the same setting of Section 9, which is also summarized in Table 3. In particular, Figures 3 and 4 show the learning curves associated with the *HalfCheetah-v4* environment, Figures 5 and 6 show the ones for the *Hopper-v4* environment, while Figures 7 and 8 show the ones for the *Swimmer-v4* environment. In all the environments, it is possible to notice that, for increasing values of the exploration parameters σ_{P} and σ_{A}, the learning curves J_{P} and J_{A} (optimized respectively by PGPE and GPOMDP) differ increasingly with the associated empirical deterministic one J_{D} (reported in right-hand side column in the plots). This is due to the fact that small values of σ_{P} and σ_{A} lead to a lower amount of exploration. Poorly exploratory $\boldsymbol{\nu_\theta}$ and $\boldsymbol{\pi_\theta}$ make the algorithms test actions that are very similar to the ones that target deterministic policy $\boldsymbol{\mu_\theta}$ would suggest. Conversely, large values of σ_{P} and σ_{A} lead to a higher amount of exploration, thus J_{P} and J_{A} tend to show a higher offset w.r.t. to the associated empirical J_{D}.

HalfCheetah. In Figures 3 and 4, it is possible to see the learning curves of J_{P} and J_{A} (and the associated empirical J_{D}) seen by PGPE and GPOMDP while learning on *HalfCheetah-v4*. Note that, in this case, the optimal value for σ_{P}^2 is 1, while the one for σ_{A}^2 is 10. With $T = 100$, PGPE seems to struggle a bit more in finding a good deterministic policy w.r.t. GPOMDP. This can be explained by the fact that the parameter dimensionality d_Θ is the highest throughout the three presented environments.

Hopper. In Figures 5 and 6, it is possible to see the learning curves of J_{P} and J_{A} (and the associated empirical J_{D}) seen by PGPE and GPOMDP while learning on *Swimmer-v4*. Also in this case, the optimal value for σ_{P}^2 is 1, while the one for σ_{A}^2 is 10. As for *HalfCheetah*, with $T = 100$, PGPE seems to struggle a bit more in finding a good deterministic policy w.r.t. GPOMDP, even if this is the intermediate difficulty environment for what concerns the parameter dimensionality d_Θ.

Swimmer. In Figures 7 and 8, it is possible to see the learning curves of J_{P} and J_{A} (and the associated empirical J_{D}) seen by PGPE and GPOMDP while learning on *Swimmer-v4*. Note that, in this case, the optimal value for σ_{P}^2 is 10, while the one for σ_{A}^2 is 1. Here we employed an horizon $T = 200$. Indeed, as also commented in Section 9, GPOMDP struggles more than PGPE in finding a good deterministic policy.

[10] When the policy draws an action the environment performs a clip of the action *before* the reward is computed.

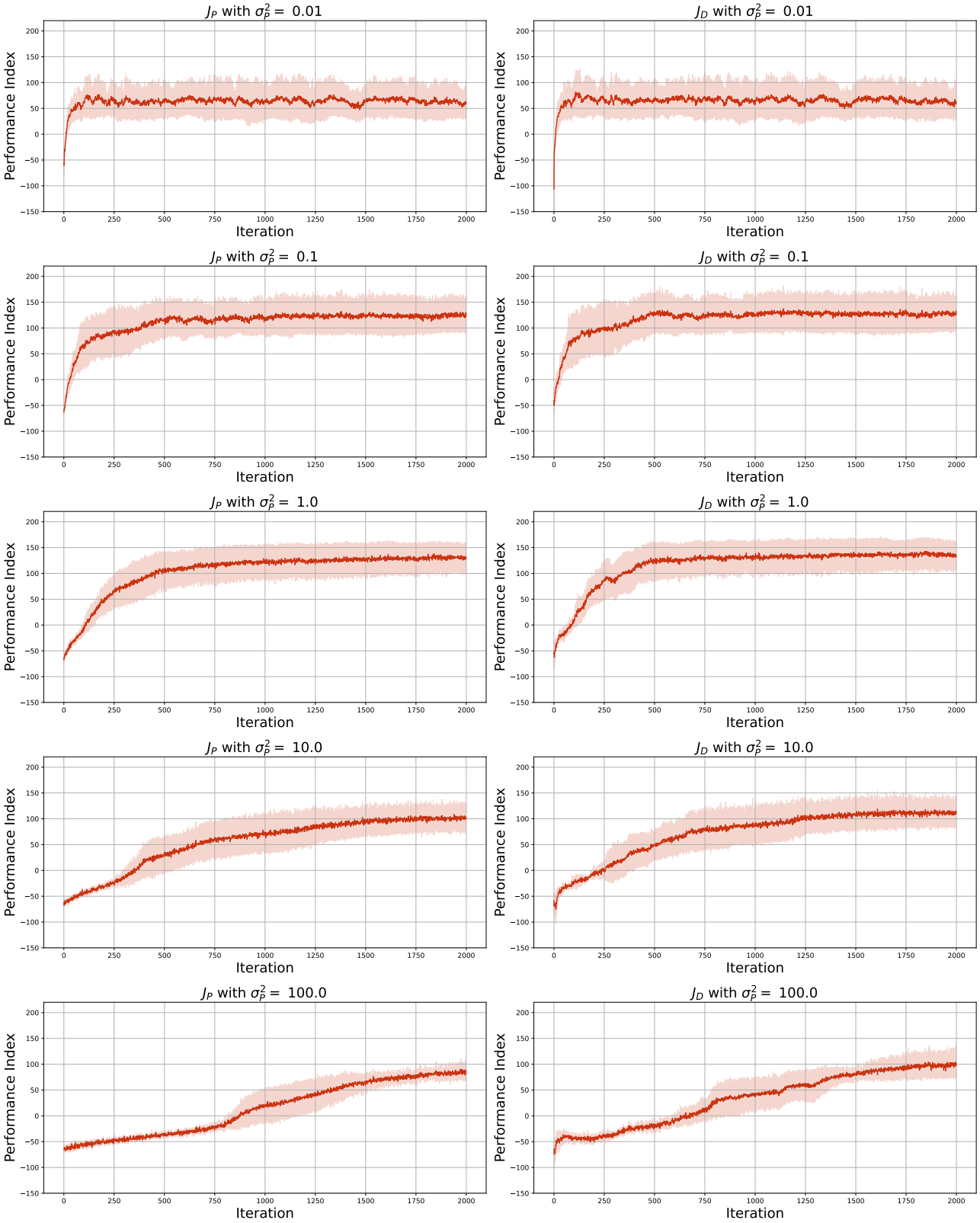


Figure 3. J_P and J_D learning curves (5 runs, mean $\pm 95\%$ C.I.) for PGPE on *Half Cheetah-v4*.

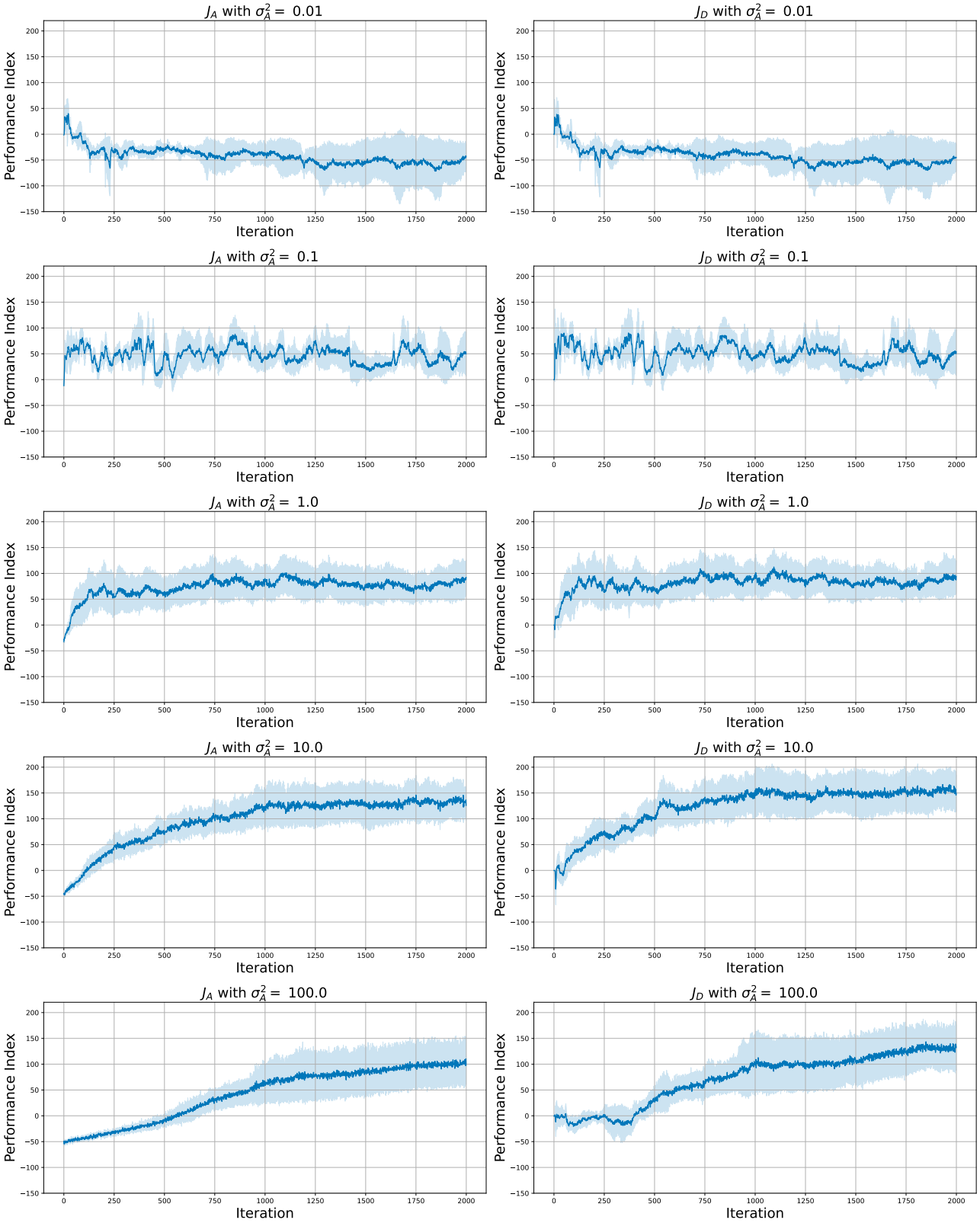


Figure 4. J_A and J_D learning curves (5 runs, mean $\pm 95\%$ C.I.) for GPOMDP on *Half Cheetah-v4*.

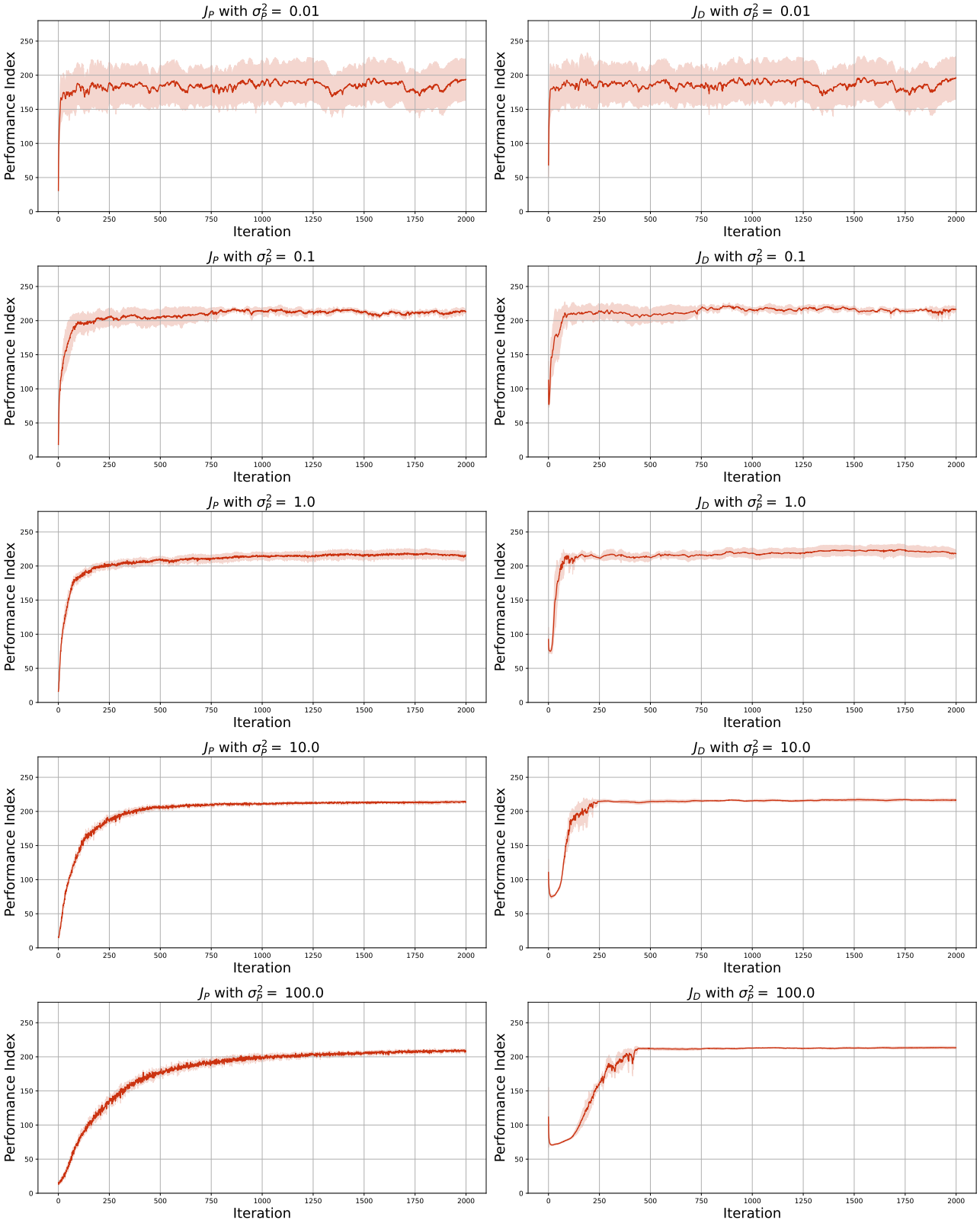


Figure 5. J_P and J_D learning curves (5 runs, mean $\pm 95\%$ C.I.) for PGPE on *Hopper-v4*.

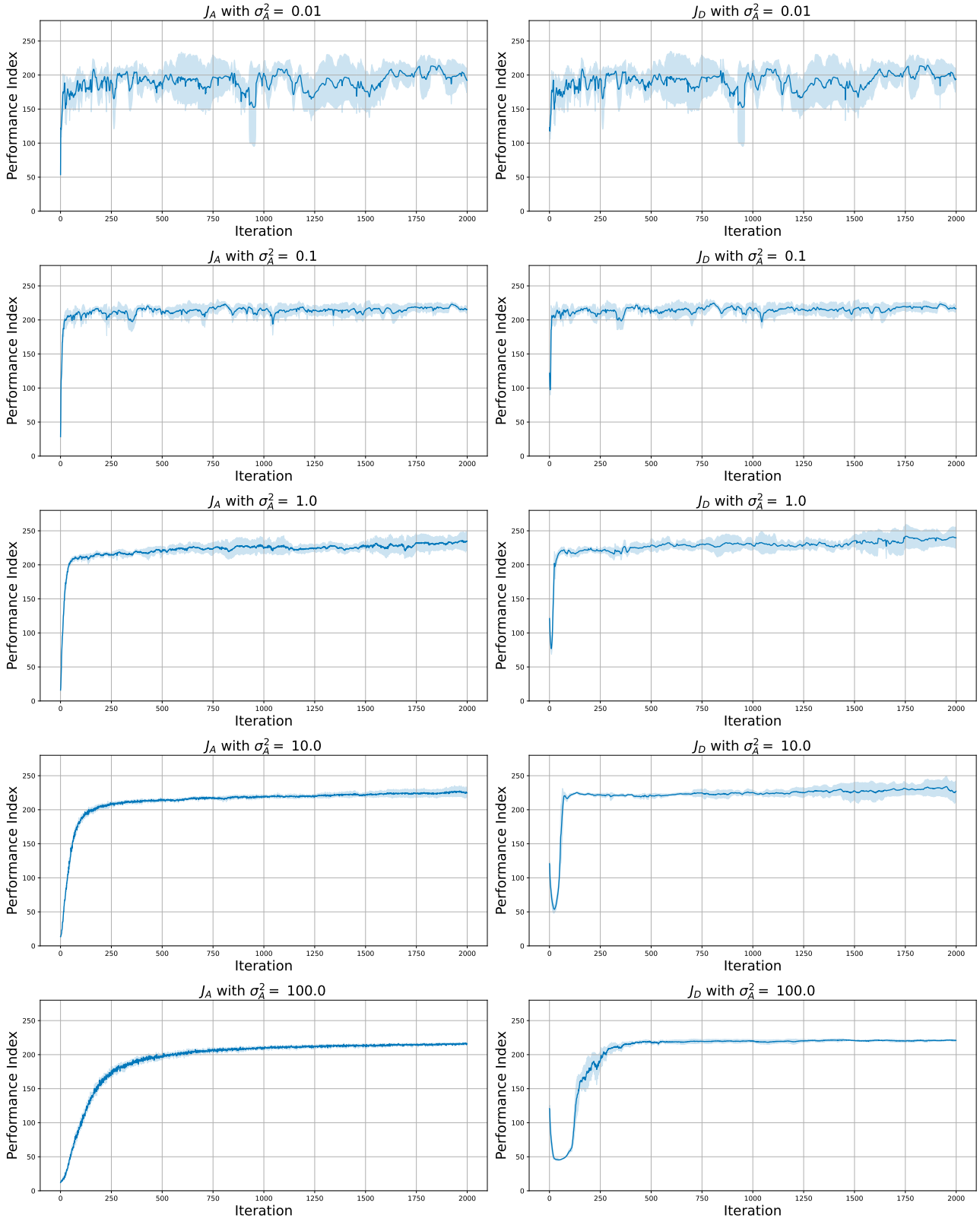


Figure 6. J_A and J_D learning curves (5 runs, mean $\pm 95\%$ C.I.) for GPOMDP on *Hopper-v4*.

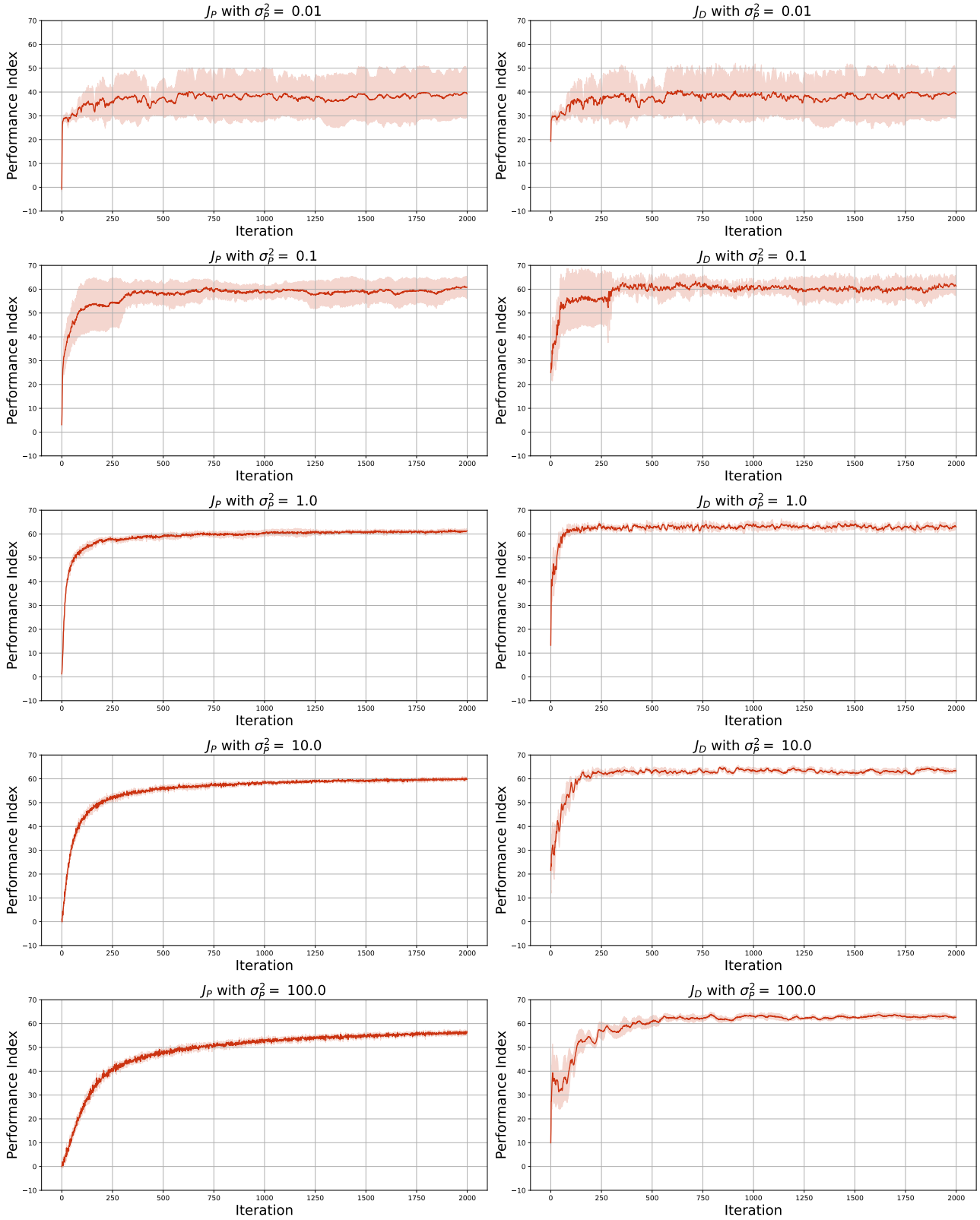


Figure 7. J_P and J_D learning curves (5 runs, mean $\pm 95\%$ C.I.) for PGPE on *Swimmer-v4*.

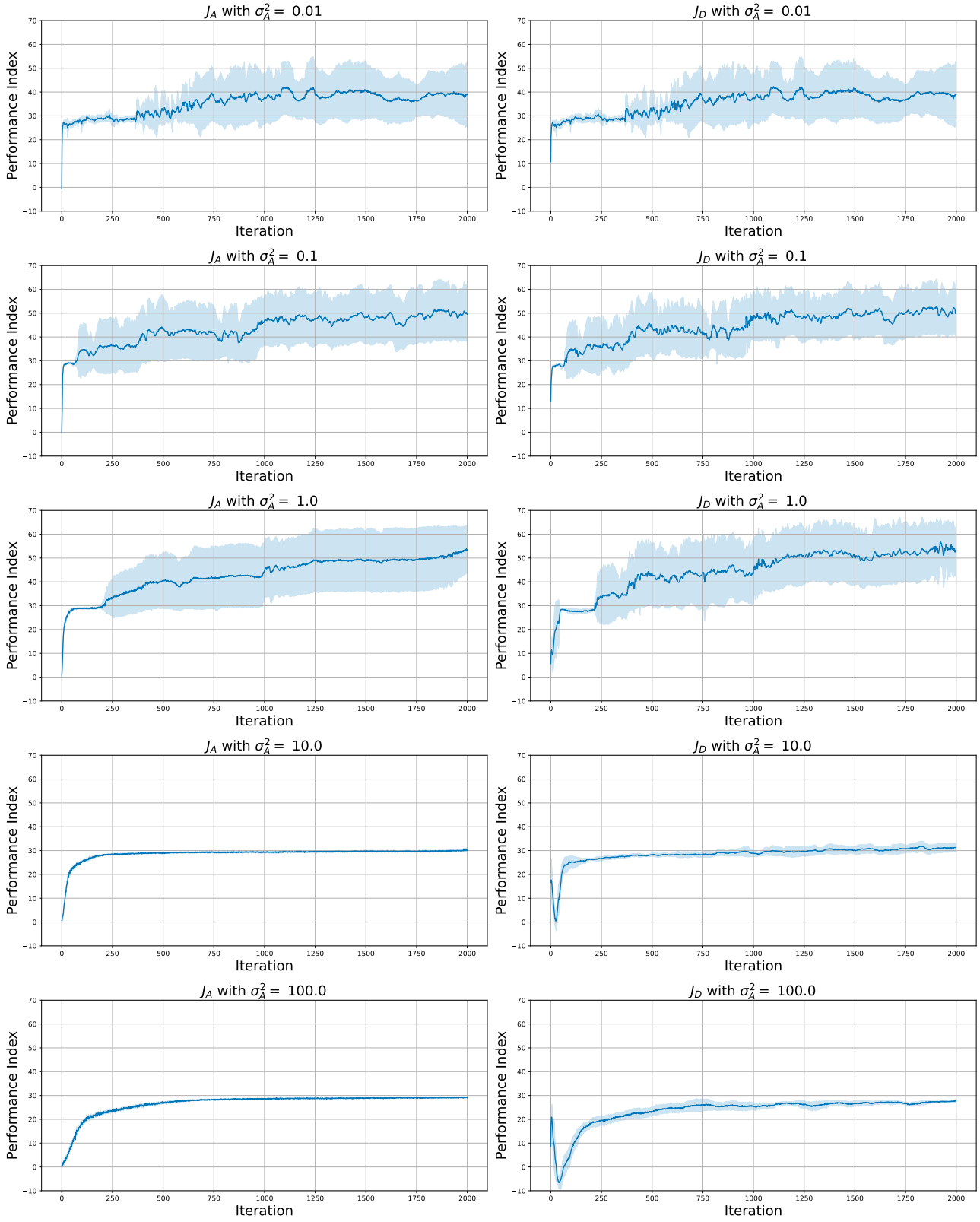


Figure 8. J_A and J_D learning curves (5 runs, mean $\pm 95\%$ C.I.) for GPOMDP on *Swimmer-v4*.

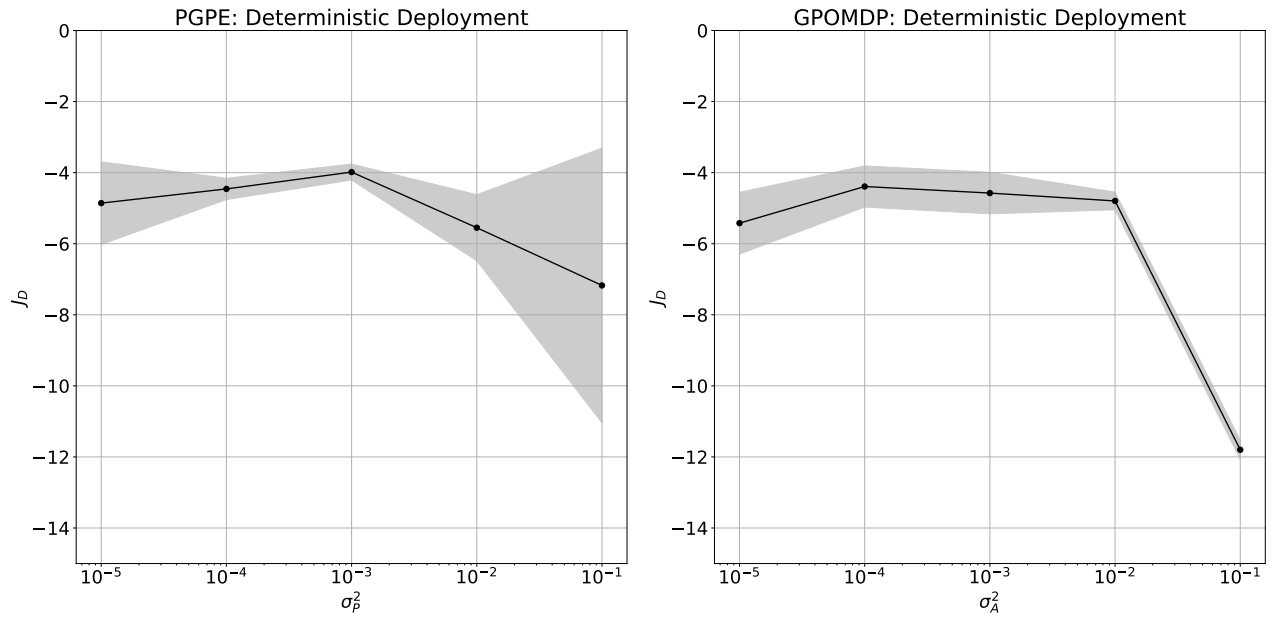


Figure 9. $J_{\mathrm{D}}(\boldsymbol{\theta}_K)$ associated with GPOMDP and PGPE in LQR ($T=50$) employing linear (hyper)policies and varying exploration amounts (5 runs, mean $\pm 95\%$ C.I.).

H.2. GPOMDP vs. PGPE: the case of LQR

In order to show more clearly the discussed trade-offs, here we present a numerical validation conduced on the Linear Quadratic Regulator (LQR, Kucera, 1992) environment, much smaller w.r.t. the ones offered by the MuJoCo suite.

Brief description of the environment Here we summarize the considered version of the LQR environment. Considering x_t and u_t as the state and action at time t, respectively, the state evolution is computed as: $x_{t+1} = Ax_t + Bu_t$. The reward at time t is computed as: $r_t = -x_t^\top Q x_t - u_t^\top R u_t$. The initial state of the environment is randomly sampled from the interval $[-3, 3]$ using a uniform distribution.

Setting Our objective is to control the LQR environment via a deterministic linear policy. For the presented results, we considered a number of iterations of $K = 3000$ for both PGPE and GPOMDP, with a batch size of $N = 100$ trajectories for each iteration, and a learning rate schedule governed by Adam, with initial step sizes of 0.01. We conducted 3 runs for each experiment, and the plots depict the mean $\pm 95\%$ confidence interval. Moreover, we considered a bi-dimensional LQR environment (i.e., $d_{\mathcal{S}} = 2$ and $d_{\mathcal{A}} = 2$), with unlimited state and action spaces (i.e., state and action ranges are $(-\infty, +\infty)$). Furthermore, the characteristic matrices of the LQR environment were selected as:

$$A = B = \begin{bmatrix} 0.9 & 0 \\ 0 & 0.9 \end{bmatrix}, \qquad Q = \begin{bmatrix} 0.9 & 0 \\ 0 & 0.1 \end{bmatrix}, \qquad \text{and} \qquad R = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.9 \end{bmatrix}. \tag{134}$$

Sensitivity w.r.t. $\sigma_\dagger^2$ Here we present a similar study to the one that has been discussed in the main paper. We tested both PGPE and GPOMDP on the previously described LQR with $T = 50$ and with the exploration amounts varying in $\sigma_\dagger^2 \in \{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$. As can be noticed in Figure 9, there are values for the exploration amounts σ_{P} and σ_{A} leading to higher performance values for the deployed deterministic policy. In particular, PGPE deploys its best version of $\mu_{\boldsymbol{\theta}}$ when setting $\sigma_{\mathrm{P}}^2 = 10^{-3}$, while the same happens with GPOMDP setting $\sigma_{\mathrm{A}} = 10^{-4}$.

Increasing T Here we present a study on the horizon length T, for which we tested values $T \in \{50, 100, 200\}$. We display in Figure 10 the resulting J_{D} associated with the learning processes of PGPE and GPOMDP. For each of the algorithms, we employed the best values of σ obtained from the previous experiment. Additionally, we depicted as a dashed line the estimated performance of the optimal policy for LQR, showing that both algorithms manage to achieve performance close to optimal on average. As can be observed, GPOMDP struggles more than PGPE in converging to the globally optimal

deterministic policy. In particular, the performance of the deterministic policy associated with PGPE appears not to change with increasing values of T.

H.3. GPOMDP vs. PGPE: the case of *Swimmer-v4*.

In this section, we conduce experiments to highlight the trade-off between parameter dimensionality d_Θ and trajectory length T. As emerges from the theoretical results shown in the main paper, GPOMDP should struggle in finding a good deterministic policy $\mu_{\boldsymbol{\theta}}$ from large values of T, while PGPE should struggle in the same task form large values of d_Θ. Notice that this behavior was already visible in the variance study conducted in Section 9 and Appendix H.1, where we added action clipping to environments. To better illustrate the trade-off at hand, we removed the action clipping to conduce the following experimental results, restoring the original version of the MuJoCo environments. Indeed, we remark that action clipping was introduced to facilitate the exploration, highlighting the outcomes of the variance study.

Setting. We consider two different target deterministic policies $\mu_{\boldsymbol{\theta}}$:

- *linear*: PGPE and GPOMDP are run for $K = 2000$, with $N = 100$, $d_\Theta = 16$ (parameters initialized to 0);
- *neural network* (two dense hidden layers with 32 neurons and with hyperbolic tangent activation functions): PGPE and GPOMDP are run for $K = 2000$, with $N = 100$, $d_\Theta = 1344$ (parameters initial values sampled by $\mathcal{N}(0,1)$).

For the learning rate schedule, we employed Adam with the same step sizes 0.1 for PGPE and 0.01 for GPOMDP (the reason is the same explained in Section 9). For all the experiments we fixed both σ_P and σ_A to 1.

Increasing T. Here we show the results of learning on *Swimmer-v4* with $T \in \{100, 200\}$ (and $\gamma = 1$). The target deterministic policy in this case is the linear one, thus $d_\Theta = 16$. Figures 11 and 12 show the learning curves of J_P and J_A, with their associated empirical J_D. For $T = 100$, PGPE and GPOMDP reach deterministic policies exhibiting similar values of $J_\text{D}(\boldsymbol{\theta}_K)$. For $T = 200$, instead, the algorithms reach deterministic policies showing an offset in the values of $J_\text{D}(\boldsymbol{\theta}_K)$ in favor of PGPE. As suggested by the theoretical results shown in the paper, the fact that GPOMDP struggles in reaching a good deterministic policy can be explained by the doubling of the horizon value.

Increasing d_Θ. Here we show the results of learning on *Swimmer-v4* with $T \in \{100, 200\}$ (and $\gamma = 1$), with two different target deterministic policies: the linear one ($d_\Theta = 16$) and the neural network one ($d_\Theta = 1344$).

Figures 13 and 15 show the learning curves of J_P, with their associated empirical J_D, for both the target policies, when learning with trajectories respectively of length 100 and 200. For both the values of the horizon, it is possible to notice that with a smaller value of d_Θ PGPE manages to find a better deterministic policy. Indeed, the found linear and neural network deterministic policies show an offset in $J_\text{D}(\boldsymbol{\theta}_K)$ in favor of the linear one. As suggested by the theoretical results shown in the paper, the fact that PGPE struggles in reaching a good deterministic policy can be explained by the heavily increased parameter dimensionality d_Θ.

Figures 14 and 16 show the learning curves of J_A, with their associated empirical J_D, for both the target policies, when learning with trajectories respectively of length 100 and 200. From Figure 14, even with the target neural network policy, for $T = 100$ GPOMDP is able however to find a deterministic policy with similar performances to the one found when the target deterministic policy is the linear one. Switching to $T = 200$ (Figure 16), it is possible to notice a severe offset between the learning curves of the empirical J_D associated to J_A, in favor of the case in which the target policy is the linear one. As done for the analysis on the increasing T, this can be explained by the fact that the horizon has been doubled, which is in line with the theoretical results shown throughout this work.

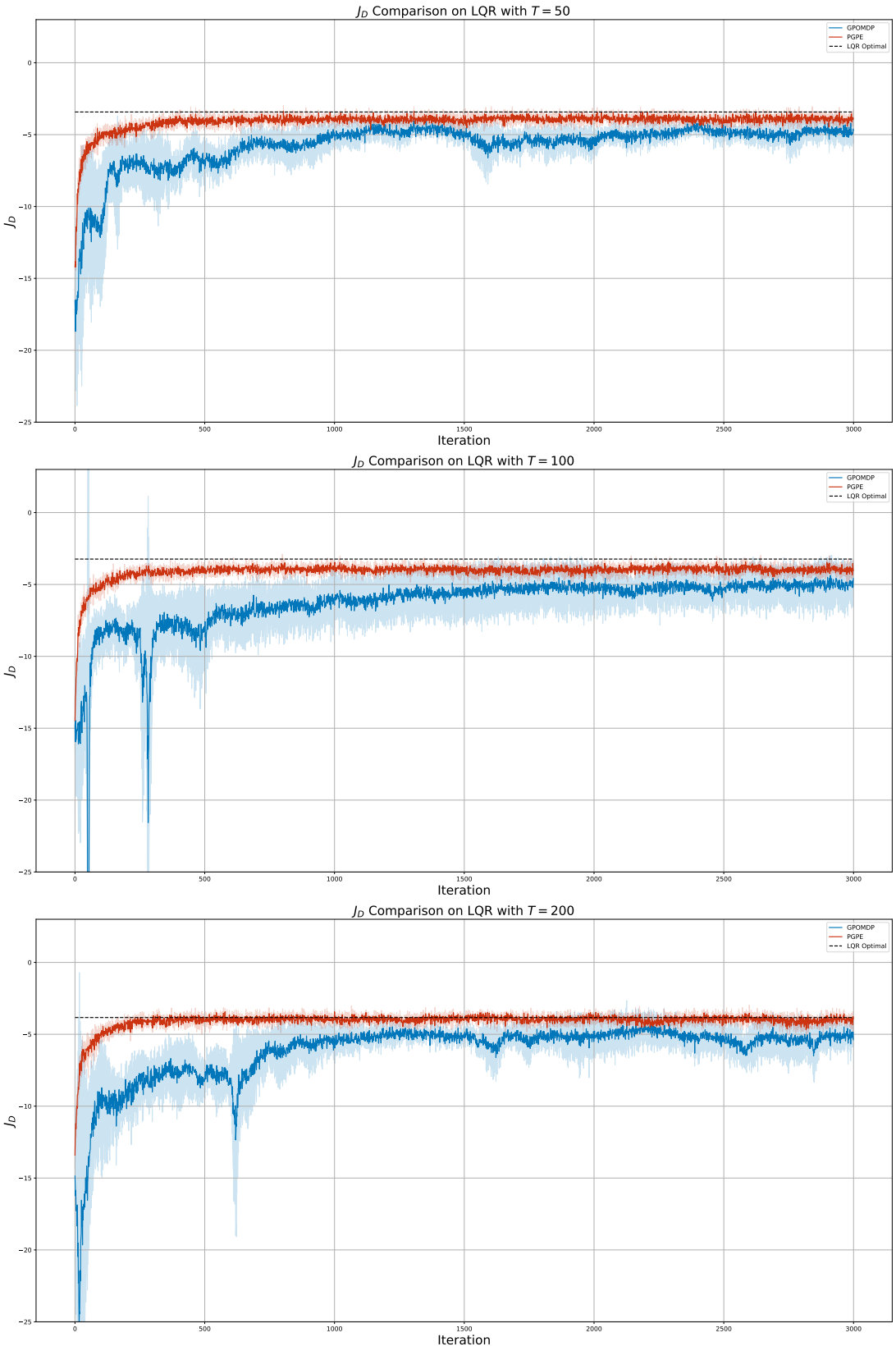
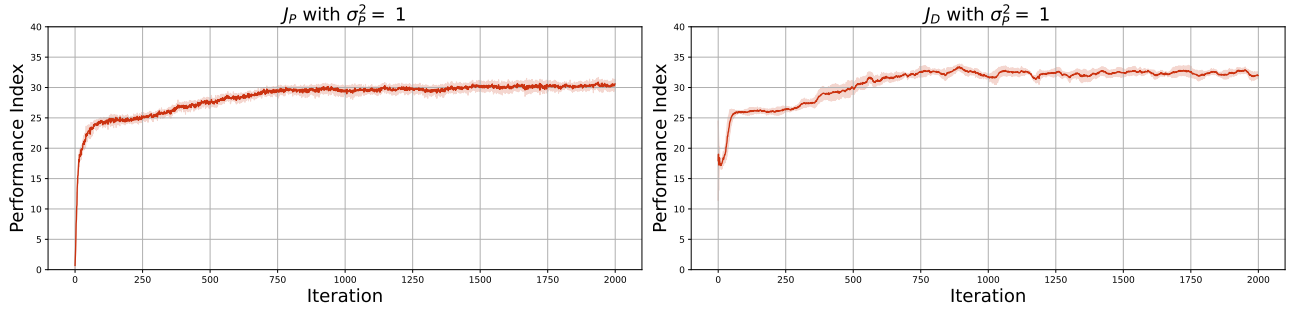
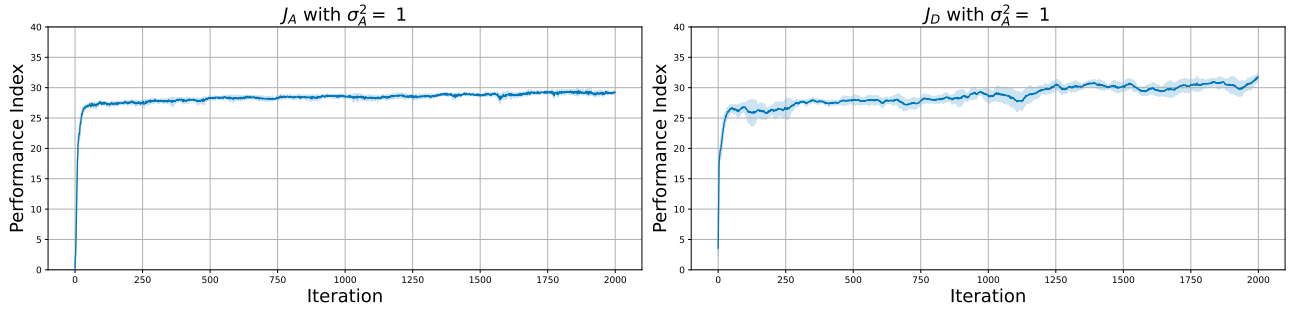


Figure 10. J_{D} associated with GPOMDP and PGPE in LQR employing linear (hyper)policies and $T \in \{50, 100, 200\}$ (5 runs, mean $\pm 95\%$ C.I.).

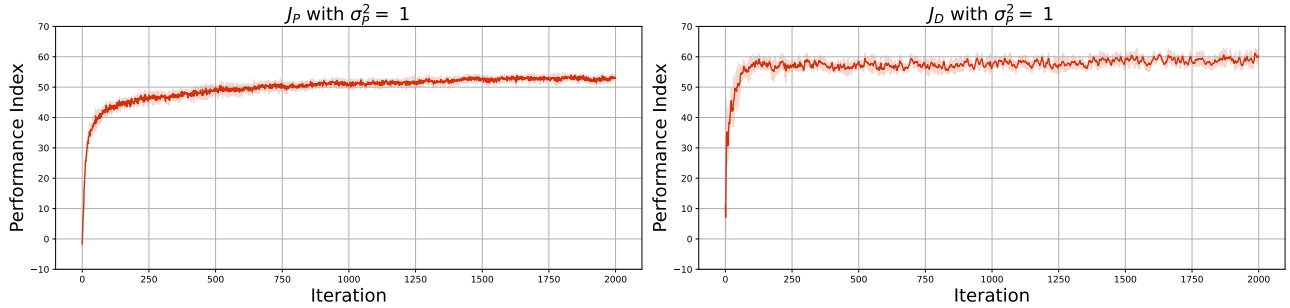


(a) PGPE.

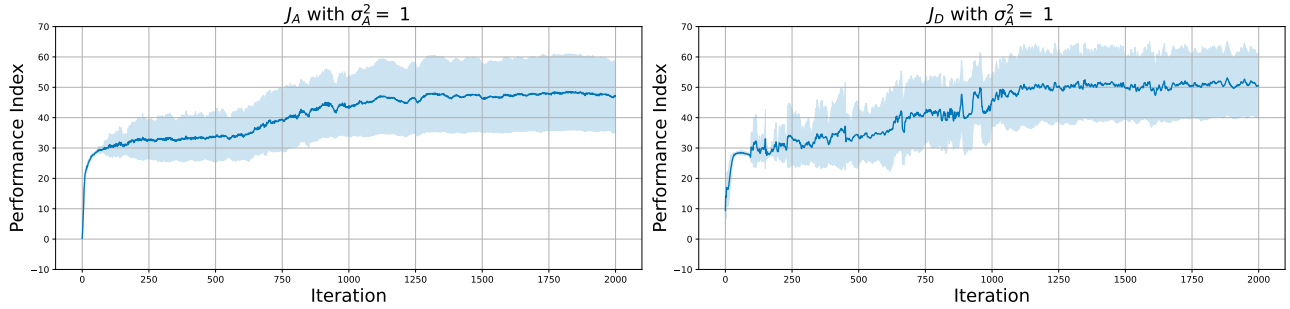


(b) GPOMDP.

Figure 11. PGPE and GPOMDP on *Swimmer-v4* with linear policy and $T = 100$ (5 runs, mean $\pm 95\%$ C.I.).

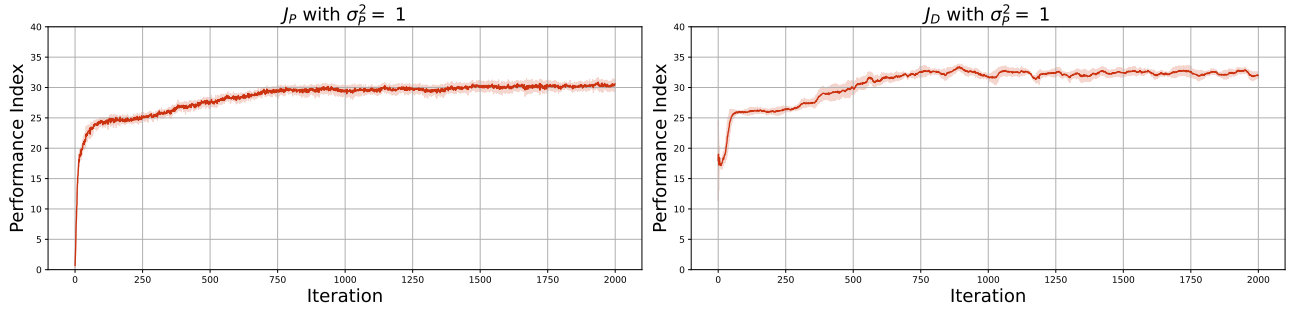


(a) PGPE.

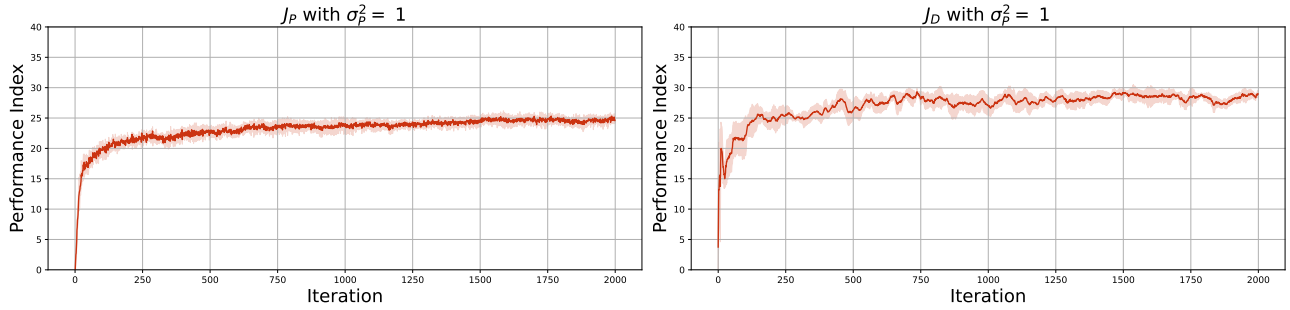


(b) GPOMDP.

Figure 12. PGPE and GPOMDP on *Swimmer-v4* with linear policy and $T = 200$ (5 runs, mean $\pm 95\%$ C.I.).

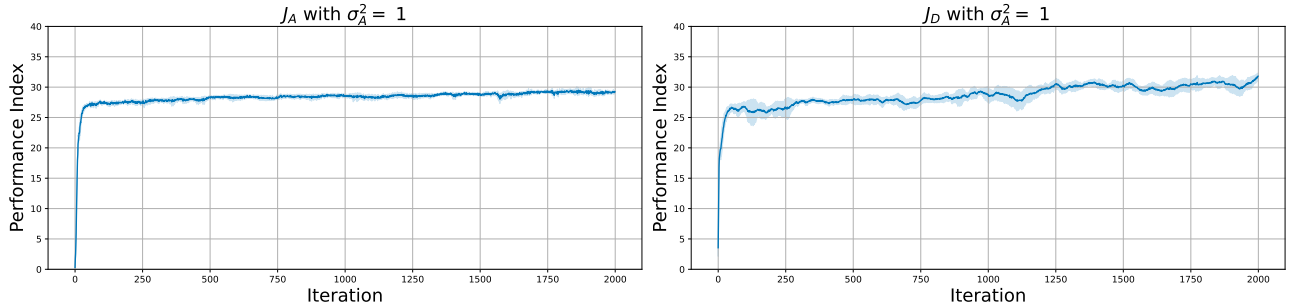


(a) Linear.

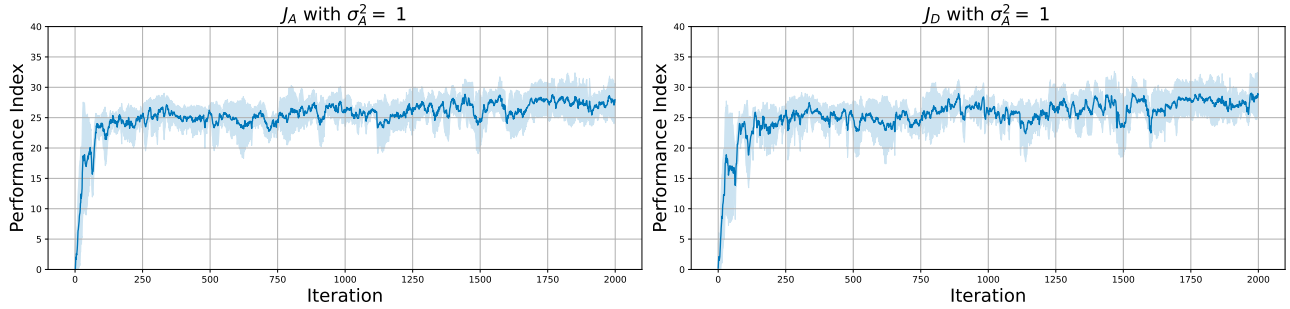


(b) Neural Network.

Figure 13. PGPE on *Swimmer-v4* with linear and neural network policies, and $T = 100$ (5 runs, mean $\pm 95\%$ C.I.).

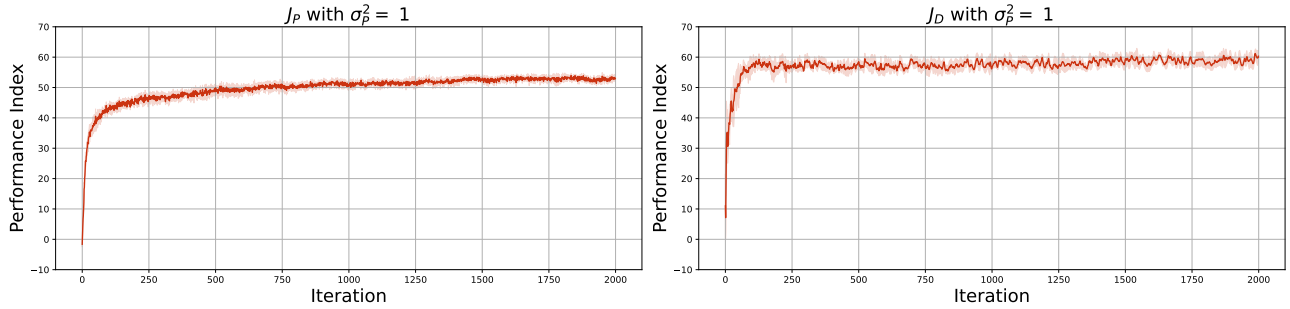


(a) Linear.

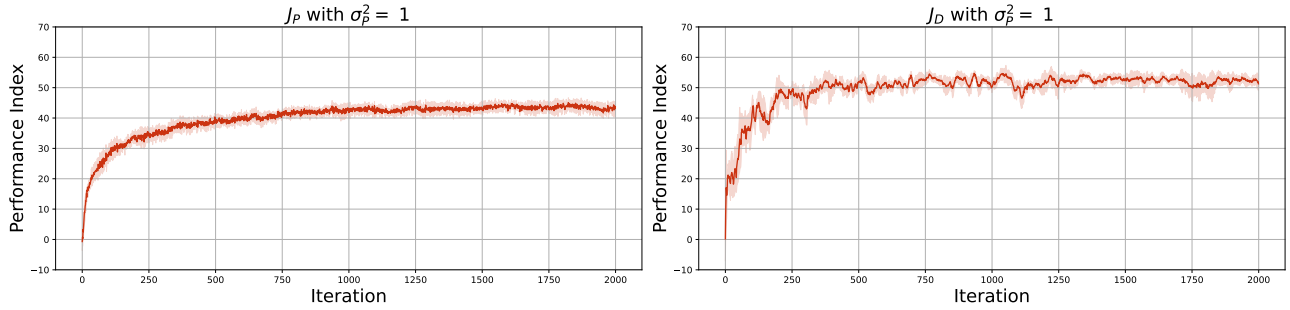


(b) Neural Network.

Figure 14. GPOMDP on *Swimmer-v4* with linear and neural network policies, and $T = 100$ (5 runs, mean $\pm 95\%$ C.I.).

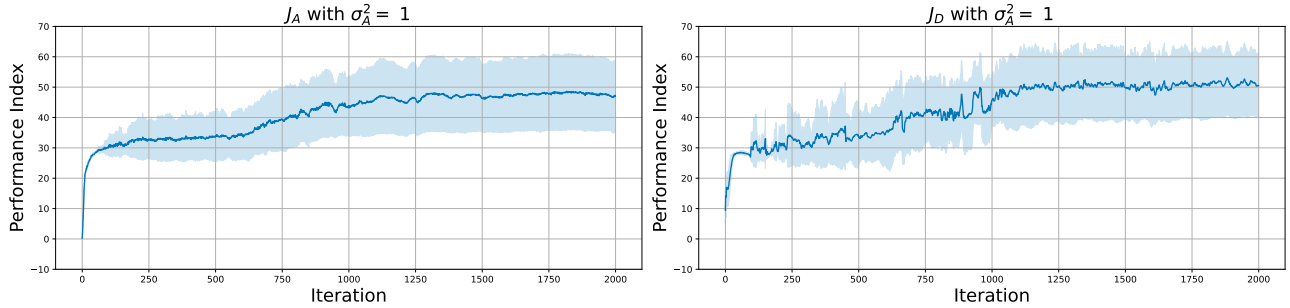


(a) Linear.

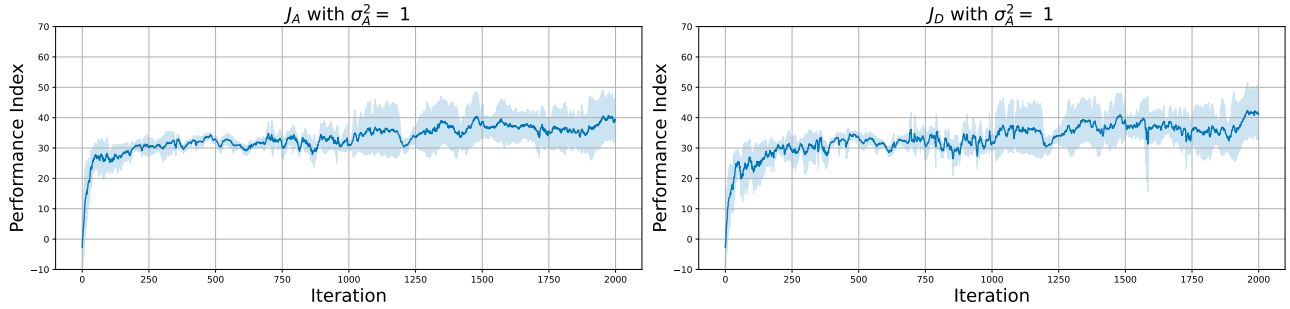


(b) Neural Network.

Figure 15. PGPE on *Swimmer-v4* with linear and neural network policies, and $T = 200$ (5 runs, mean $\pm 95\%$ C.I.).



(a) Linear.



(b) Neural Network.

Figure 16. GPOMDP on *Swimmer-v4* with linear and neural network policies, and $T = 200$ (5 runs, mean $\pm 95\%$ C.I.).