Robotics: Science and Systems 2021
Held Virtually, July 12–16, 2021

Robust Value Iteration for Continuous Control Tasks

Michael Lutter[1,2], Shie Mannor[1,3], Jan Peters[2], Dieter Fox[1,4], Animesh Garg[1,5]
[1]Nvidia, [2]TU Darmstadt, [3] Technion, [4] University of Washington, [5]University of Toronto & Vector Institute

***Abstract*—When transferring a control policy from simulation to a physical system, the policy needs to be robust to variations in the dynamics to perform well. Commonly, the optimal policy overfits to the approximate model and the corresponding state-distribution, often resulting in failure to trasnfer underlying distributional shifts. In this paper, we present Robust Fitted Value Iteration, which uses dynamic programming to compute the optimal value function on the compact state domain and incorporates adversarial perturbations of the system dynamics. The adversarial perturbations encourage a optimal policy that is robust to changes in the dynamics. Utilizing the continuous-time perspective of reinforcement learning, we derive the optimal perturbations for the states, actions, observations and model parameters in closed-form. Notably, the resulting algorithm does not require discretization of states or actions. Therefore, the optimal adversarial perturbations can be efficiently incorporated in the min-max value function update. We apply the resulting algorithm to the physical Furuta pendulum and cartpole. By changing the masses of the systems we evaluate the quantitative and qualitative performance across different model parameters. We show that robust value iteration is more robust compared to deep reinforcement learning algorithm and the non-robust version of the algorithm. Videos of the experiments are shown at https://sites.google.com/view/rfvi**

I. Introduction

To avoid the laborious and potentially dangerous training of control policies on the physical system, Simulation to reality transfer (sim2real) learns a policy in simulation and evaluates the policy on the physical system. When transferred to the real world, the policy should solve the task and obtain a comparable reward to the simulation. Therefore, the goal of sim2real is to learn a policy that is robust to changes in the dynamics and successfully bridges the simulation-reality gap. Naive reinforcement learning (RL) methods usually do not succeed for sim2real as the resulting policies overfit to the approximate simulation model. Therefore, the resulting policies are not robust to changes in the dynamics and fail to solve the task in the real world. In contrast, sim2real methods extending RL with domain randomization [1–4] or adversarial disturbances [5–8] have shown the successful transfer to the physical world [9].

In this paper, we focus on adversarial disturbances to bridge the simulation-reality gap. In this paradigm the RL problem is formulated as a two player zero-sum game [6]. The optimal policy wants to maximize the reward while the adversary wants to minimize the reward. For control tasks, the policy controls the system input $\boldsymbol{u}$ and the adversary $\boldsymbol{\xi}$ controls the dynamics. For example, the adversary can change the state, action, observation, model parameters or all of them within limits. Therefore, this problem formulation optimizes the worst-case performance and not the expected reward as standard RL. The resulting policy is robust to changes in the dynamics as the planning in simulation uses the worst-case approximation which includes the physical system [10].

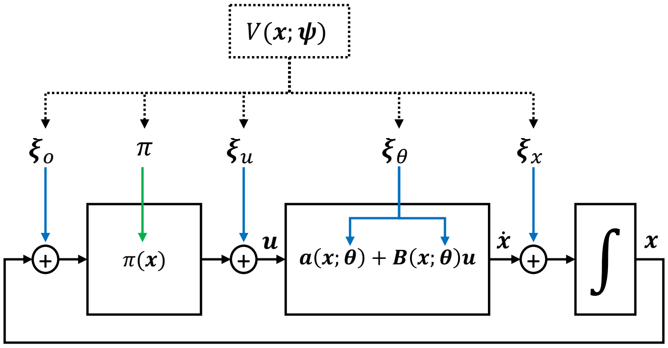
$V(\boldsymbol{x};\boldsymbol{\psi})$
$\boldsymbol{\xi}_o$
π
$\boldsymbol{\xi}_u$
$\boldsymbol{\xi}_\theta$
$\boldsymbol{\xi}_x$
$\pi(\boldsymbol{x})$
$\boldsymbol{u}$
$\boldsymbol{a}(\boldsymbol{x};\boldsymbol{\theta}) + \boldsymbol{B}(\boldsymbol{x};\boldsymbol{\theta})\boldsymbol{u}$
$\dot{\boldsymbol{x}}$
$\int$
$\boldsymbol{x}$

Figure 1. The control chart of robust fitted value iteration (rFVI) for continuous states, actions and time. The deterministic optimal policy and deterministic adversaries, which add an bias to the system dynamics, only depend on shared value function. While the optimal policy performs hill-climbing the adversaries perform steepest descent following the value function gradient.

In this adversarial RL setting,

(1) we show that the optimal action and optimal adversary can be directly computed using the value function estimate if the reward is separable into state and action reward and the continuous-time dynamics are non-linear and control-affine. We derive the solution for the state, action, observation and model bias analytically. Therefore, this paper extends the existing analytic solutions from continuous-time RL [11–14].

(2) we propose robust fitted value iteration (rFVI). This algorithm solves the adversarial RL problem with continuous states, actions and disturbances by leveraging the analytic expressions and value iteration. Using this approach, the continuous states and actions do not need to be discretized as in classical methods [15–17] or require multiple optimizations as the modern actor-critic approaches [7, 18].

(3) we provide an in-depth evaluation of rFVI on the physical system. We benchmark this algorithm on the real-world Furuta pendulum and cartpole. To test the robustness, we perturb the model parameters by applying additional weights. The performance is compared to standard deep RL algorithms with and without domain randomization.

Therefore the contributions of this paper are the derivation of the analytic adversarial actions, the introduction of robust fitted value iteration and the extensive evaluation on multiple physical systems. In the evaluation we focus on two under-actuated systems to provide an in-depth qualitative and quan-

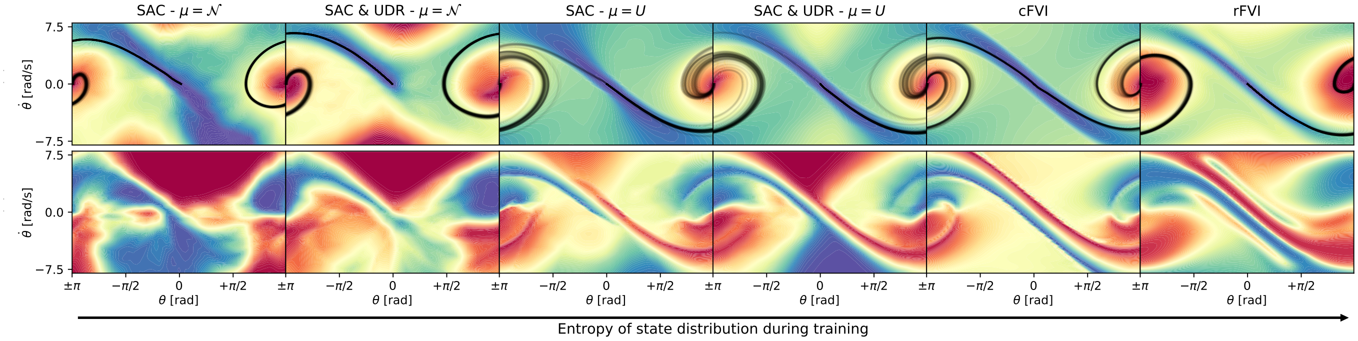

Figure 2. The optimal Value function V^* and policy π^* of rFVI, cFVI and four different variations of SAC. All policies achieve nearly identical reward on the nominal dynamics model. The variations of SAC demonstrate the change of the policy when increasing the entropy of the state distribution during training. The entropy is increased by enlarging the initial state distribution μ and using domain randomization. For SAC and $\mu = \mathcal{N}(\pm\pi, \sigma)$ the optimal policy is only valid on the optimal trajectory. For SAC UDR and $\mu = \mathcal{U}(-\pi, +pi)$, the policy is applicable on the complete state domain. rFVI and cFVI perform value iteration on the compact state domain and naturally obtain an optimal policy applicable on the complete state-domain. rFVI adapts V^* and π^* to have a smaller ridge leading up to the upright pendulum and exerts higher actions when deviating from the optimal trajectory.

titative analysis of the various algorithms in the physical world.

In the following we introduce the continuous-time formulation of adversarial RL (Section II). Section III derives the optimal adversaries and introduces robust fitted value iteration. Finally, we describe the experimental setup and report the performance of the algorithms on the physical system in Section IV.

II. Problem Statement

The infinite horizon continuous-time RL optimization with the adversary $\boldsymbol{\xi}$ is described by

$$\pi^*(\boldsymbol{x}) = \arg\max_{\pi} \inf_{\boldsymbol{\xi}\in\Omega} \int_0^\infty \exp(-\rho t)\;\; r_c(\boldsymbol{x}_t, \boldsymbol{u}_t)\, dt, \tag{1}$$

$$V^*(\boldsymbol{x}) = \max_{\boldsymbol{u}} \inf_{\boldsymbol{\xi}\in\Omega} \int_0^\infty \exp(-\rho t)\, r_c(\boldsymbol{x}_t, \boldsymbol{u}_t)\, dt, \tag{2}$$

$$\boldsymbol{x}(t) = \boldsymbol{x}_0 + \int_0^t f_c(\boldsymbol{x}_\tau, \boldsymbol{u}_\tau,\, \boldsymbol{\xi}_\tau)\, d\tau, \tag{3}$$

with the state $\boldsymbol{x}$, action $\boldsymbol{u}$, admissible set Ω, discounting constant $\rho \in (0, \infty]$, reward r_c, dynamics f_c, optimal value function V^* and policy π^* [6, 13]. The order of the optimizations can be switched as the optimal actions and disturbance remain identical [6]. The adversary $\boldsymbol{\xi}$ must be constrained to be in the set of admissible disturbances Ω as otherwise the adversary is too powerful and would prevent the policy from learning the task.

The deterministic continuous-time dynamics f_c is assumed to be non-linear w.r.t. the system state $\boldsymbol{x}$ but affine w.r.t. the action $\boldsymbol{u}$. Such dynamics are described by

$$\dot{\boldsymbol{x}} = \boldsymbol{a}(\boldsymbol{x}; \boldsymbol{\theta}) + \boldsymbol{B}(\boldsymbol{x}; \boldsymbol{\theta})\boldsymbol{u}, \tag{4}$$

with the non-linear drift $\boldsymbol{a}$, the non-linear control matrix $\boldsymbol{B}$ and the system parameters $\boldsymbol{\theta}$. We assume that the approximate parameters $\hat{\boldsymbol{\theta}}$ and the approximate equation of motions $\hat{\boldsymbol{a}}$, $\hat{\boldsymbol{B}}$ are known. For most rigid-body systems this assumptions is feasible as the equations of motions can be derived analytically and the system parameters can be measured. The resulting model is only approximate due to the idealized assumption of rigid bodies, measurement error of the system parameters and neglecting the actuator dynamics.

The optimal policy and adversary is modeled as a stationary and Markovian policy that applies a state-dependent disturbance. In this case, the worst-case action is deterministic if the dynamics are deterministic. If the adversary would be stochastic, the optimal policies are non-stationary and non-Markovian [19]. This assumption is used in most of the existing literature on adversarial RL [8, 13, 19, 20]. We consider four different adversaries that alter the state [20–22], action [7, 13, 18, 23–25], observation [8, 19] and model parameters [8]. The different adversaries address potential causes of the simulation gap. The state adversary $\boldsymbol{\xi}_x$ incorporates unmodeled physical phenomena in the simulation. The action adversary $\boldsymbol{\xi}_u$ addresses the non-ideal actuators. The observation adversary $\boldsymbol{\xi}_o$ introduces the non-ideal observations caused by sensors. The model adversary $\boldsymbol{\xi}_\theta$ introduces a bias to the system parameters. All adversaries could be subsumed via a single adversary with large admissible set. However, the resulting dynamics would not capture underlying structure of the simulation gap [8] and the optimal policy would be too conservative [26]. Therefore, we disambiguate between the different adversaries to capture this structure. Mathematically the models are described by

$$\begin{aligned}
\text{State } \boldsymbol{\xi}_x :& \quad \dot{\boldsymbol{x}} = \boldsymbol{a}(\boldsymbol{x}; \boldsymbol{\theta}) + \boldsymbol{B}(\boldsymbol{x}; \boldsymbol{\theta})\boldsymbol{u} + \boldsymbol{\xi}_{\boldsymbol{x}}, && (5)\\
\text{Action } \boldsymbol{\xi}_u :& \quad \dot{\boldsymbol{x}} = \boldsymbol{a}(\boldsymbol{x}; \boldsymbol{\theta}) + \boldsymbol{B}(\boldsymbol{x}; \boldsymbol{\theta})\; (\boldsymbol{u} + \boldsymbol{\xi}_{\boldsymbol{u}}), && (6)\\
\text{Observation } \boldsymbol{\xi}_o :& \quad \dot{\boldsymbol{x}} = \boldsymbol{a}(\boldsymbol{x} + \boldsymbol{\xi}_{\boldsymbol{o}}; \boldsymbol{\theta}) + \boldsymbol{B}(\boldsymbol{x} + \boldsymbol{\xi}_{\boldsymbol{o}}; \boldsymbol{\theta})\; \boldsymbol{u}, && (7)\\
\text{Model } \boldsymbol{\xi}_\theta :& \quad \dot{\boldsymbol{x}} = \boldsymbol{a}(\boldsymbol{x}; \boldsymbol{\theta} + \boldsymbol{\xi}_\theta) + \boldsymbol{B}(\boldsymbol{x}; \boldsymbol{\theta} + \boldsymbol{\xi}_\theta)\; \boldsymbol{u}. && (8)
\end{aligned}$$

Instead of disturbing the observation, Equation 7 disturbs the simulation state of the drift and control matrix. This disturbance is identical to changing the observed system state.

The deterministic perturbation is in contrast to the standard RL approaches, which describe $\boldsymbol{\xi}_i$ as a stochastic variable. However, this difference is due to the worst-case perspective, where one always selects the worst sample at each state. This worst case sample is deterministic if the policies are

Table I
The optimal actions $\boldsymbol{u}^k$ and adversarial actions ξ^k for the state-, action-, model- and observation bias with the admissible set Ω.

	State Perturbation	Action Perturbation	Model Perturbation	Observation Perturbation
Dynamics $f_c(\boldsymbol{x}, \boldsymbol{u}, \xi)$	$\boldsymbol{a}(\boldsymbol{x}) + \boldsymbol{B}(\boldsymbol{x})\boldsymbol{u} + \xi$	$\boldsymbol{a}(\boldsymbol{x}) + \boldsymbol{B}(\boldsymbol{x})(\boldsymbol{u} + \xi)$	$\boldsymbol{a}(\theta + \xi) + \boldsymbol{B}(\theta + \xi)\boldsymbol{u}$	$\boldsymbol{a}(\boldsymbol{x} + \xi) + \boldsymbol{B}(\boldsymbol{x} + \xi)\boldsymbol{u}$
Optimal Action $\boldsymbol{u}^k$	$\nabla\tilde{g}(\boldsymbol{B}(\boldsymbol{x})^T \nabla_x V^k)$	$\nabla\tilde{g}(\boldsymbol{B}(\boldsymbol{x})^T \nabla_x V^k)$	$\nabla\tilde{g}(\boldsymbol{B}(\boldsymbol{x})^T \nabla_x V^k)$	$\nabla\tilde{g}(\boldsymbol{B}(\boldsymbol{x})^T \nabla_x V^k)$
Optimal Disturbance ξ^k	$-h_\Omega\left(\nabla_x V^k\right)$	$-h_\Omega\left(\boldsymbol{B}^T \nabla_x V^k\right)$	$-h_\Omega\left(\left(\frac{\partial \boldsymbol{a}}{\partial \boldsymbol{x}} + \frac{\partial \boldsymbol{B}}{\partial \theta}\boldsymbol{u}^k\right)^T \nabla_x V^k\right)$	$-h_\Omega\left(\left(\frac{\partial \boldsymbol{a}}{\partial \boldsymbol{x}} + \frac{\partial \boldsymbol{B}}{\partial \boldsymbol{x}}\boldsymbol{u}^k\right)^T \nabla_x V^k\right)$

stationary and Markovian. The filtering approaches of state-estimation are not applicable to this problem formulation as these approaches cannot infer a state-dependent bias.

The reward is separable into state reward q_c and action reward g_c described by

$$r_c(\boldsymbol{x}, \boldsymbol{u}) = q_c(\boldsymbol{x}) - g_c(\boldsymbol{u}). \tag{9}$$

The action cost is non-linear, positive definite and strictly convex. The assumptions on the action cost are not limiting as these resulting properties are desirable. The convexity of the action cost enforces that the optimal action is unique. The positive definiteness of g_c penalizes non-zero actions, which prevents the bang-bang controller to be optimal.

III. Robust Fitted Value Iteration

In the following, we summarize continuous fitted value iteration (cFVI) [27]. Afterwards, we derive the analytic solutions for the optimal adversary and present robust fitted value iteration (rFVI). This algorithm solves the adversarial RL problem and obtains a robust optimal policy. In contrast, cFVI only solves the deterministic RL problem and obtains an optimal policy that overfits to the approximate dynamics model.

A. Preliminaries - Continuous Fitted Value Iteration

Continuous fitted value iteration (cFVI) [27] extends the classical value iteration approach to compute the optimal value function for continuous action and state spaces. By showing that the optimal policy can be computed analytically, the value iteration update can be computed efficiently. For non-linear control-affine dynamics (Equation 4) and separable reward (Equation 9), the optimal action is described by

$$\pi^k(\boldsymbol{x}) = \nabla\tilde{g}_c\left(\boldsymbol{B}(\boldsymbol{x})^T \nabla_x V^k\right) \tag{10}$$

with current value function $\nabla_x V^k$ and the convex conjugate of the action cost $\tilde{g}_c$ [11, 12, 14]. The convex conjugate is defined as $\nabla\tilde{g}(\boldsymbol{w}) = [\nabla g(\boldsymbol{w})]^{-1}$. For a quadratic action cost, $\nabla\tilde{g}$ is a linear transformation. For barrier shaped action cost, $\nabla\tilde{g}$ rescales and limits the action range. This solution is intuitive as the optimal policy performs hill climbing on the value function manifold and action cost determines the step size.

Substituting the analytic policy into the value iteration update, the classical algorithm can be extended to continuous actions, i.e.,

$$\begin{aligned} V_{\text{tar}}^{k+1} &= \max_{\boldsymbol{u}} \, r(\boldsymbol{x}, \boldsymbol{u}) + \gamma V^k(f(\boldsymbol{x}_t, \boldsymbol{u})) & (11) \\ &= r\left(\boldsymbol{x}_t, \pi^k(\boldsymbol{x}_t)\right) + \gamma V^k\left(f\left(\boldsymbol{x}_t, \pi^k(\boldsymbol{x}_t)\right)\right). & (12) \end{aligned}$$

Therefore, the closed-form policy enables the efficient computation of the target. Combined with function approximation for the value function [28–31], classical value iteration can be extended to continuous state and action spaces without discretization. The fitting of the value function is described by,

$$\psi_{k+1} = \arg\min_{\psi} \sum_{\boldsymbol{x}} \|V_{\text{tar}}^{k+1}(\boldsymbol{x}) - V(\boldsymbol{x};\, \psi)\|_p^p, \tag{13}$$

with $\|\cdot\|_p$ being the ℓ_p norm. Iterating between computing V_{tar} and fitting the value function, learns the optimal value function V^* and policy π^*.

B. Deriving the Optimal Disturbances

For the adversarial RL formulation, the value function target contains a max-min optimization described by

$$V_{\text{tar}}^{k+1}(\boldsymbol{x}) = \max_{\boldsymbol{u}} \inf_{\boldsymbol{\xi} \in \Omega} r(\boldsymbol{x}, \boldsymbol{u}) + \gamma V^k\big(f(\boldsymbol{x}, \boldsymbol{u}, \, \boldsymbol{\xi})\big). \tag{14}$$

To efficiently obtain the value function update, the optimal action and the optimal disturbance need to be computed in closed form. We show that this optimization problem can be solved analytically for the described dynamics and disturbance models (Section II). Therefore, the adversarial RL problem can be solved by value iteration.

The resulting optimal actions $\boldsymbol{u}^*$ and disturbances $\boldsymbol{\xi}_i^*$ have a coherent intuitive interpretation. The optimal actions perform steepest ascent by following the gradient of the value function $\nabla_x V$. The optimal perturbations perform steepest descent by following the negative gradient of the value function. The step-size of policy and adversary is determined by the action cost g or the admissible set Ω. The optimal policy and the optimal adversary is described by

$$\boldsymbol{u}^k = \nabla\tilde{g}\left(\frac{\partial f_c(.)}{\partial \boldsymbol{u}}^T \nabla_x V^k\right), \boldsymbol{\xi}_i^k = -h_\Omega\left(\frac{\partial f_c(.)}{\partial \boldsymbol{\xi}_i}^T \nabla_x V^k\right). \tag{15}$$

In the following we abbreviate $[\partial f_c(.)/\partial \boldsymbol{y}]^T \nabla_x V$, as $\boldsymbol{z}_y$. For the adversarial policy, h_Ω rescales $\boldsymbol{z}_\xi$ to be on the boundary of the admissible set. If the admissible set bounds the signal energy to be smaller than α, the disturbance is rescaled to have the length α. Therefore, the adversary is described by

$$\Omega_E = \{\boldsymbol{\xi} \in \mathbb{R}^n \mid \|\boldsymbol{\xi}\|_2 \leq \alpha\} \quad \Rightarrow \quad h_E(\boldsymbol{z}_\xi) = \alpha\, \frac{\boldsymbol{z}_\xi}{\|\boldsymbol{z}_\xi\|_2}. \tag{16}$$

If the amplitude of the disturbance is bounded, the disturbance performs bang-bang control. In this case the adversarial policy is described by

$$\begin{aligned} \Omega_A = \{\boldsymbol{\xi} \in \mathbb{R}^n \mid \boldsymbol{\nu}_{\min} \leq \boldsymbol{\xi} \leq \boldsymbol{\nu}_{\max}\} \\ \Rightarrow \quad h_A(\boldsymbol{z}_\xi) = \boldsymbol{\Delta}\,\mathrm{sign}\left(\boldsymbol{z}_\xi\right) + \boldsymbol{\mu}, \end{aligned} \tag{17}$$

with $\boldsymbol{\mu} = (\boldsymbol{\nu}_{\max} + \boldsymbol{\nu}_{\min})/2$ and $\boldsymbol{\Delta} = (\boldsymbol{\nu}_{\max} - \boldsymbol{\nu}_{\min})/2$.

The following theorems derive Equation 15, 16 and 17 for the optimal policy and the different disturbances. Theorem 1 describes the state adversary, Theorem 2 describes action adversary, Theorem 3 the observation adversary and Theorem 4 describes the model adversary. Following the theorems, we provide sketches of the proofs for the state and model disturbance. The remaining proofs are analogous. The complete proofs for all theorems are provided in the appendix. All solutions are summarized in Table I.

Theorem 1. *For the adversarial state disturbance (Equation 5) with bounded signal energy (Equation 16), the optimal continuous-time policy π and state disturbance $\boldsymbol{\xi}_x$ is described by*

$$\pi(\boldsymbol{x}) = \nabla\tilde{g}\left(\boldsymbol{B}(\boldsymbol{x})^T \nabla_x V\right), \qquad \boldsymbol{\xi}_x = -\alpha \frac{\nabla_x V}{\|\nabla_x V\|_2}.$$

Theorem 2. *For the adversarial action disturbance (Equation 6) with bounded signal energy (Equation 16), the optimal continuous-time policy π and action disturbance $\boldsymbol{\xi}_u$ is described by*

$$\pi(\boldsymbol{x}) = \nabla\tilde{g}\left(\boldsymbol{B}(\boldsymbol{x})^T \nabla_x V\right), \quad \boldsymbol{\xi}_u = -\alpha \frac{\boldsymbol{B}(\boldsymbol{x})^T \nabla_x V}{\|\boldsymbol{B}(\boldsymbol{x})^T \nabla_x V\|_2}.$$

Proof Sketch Theorem 1 For the admissible set Ω_E, Equation 14 can be written with the explicit constraint. This optimization is described by

$$V_{\text{tar}} = \max_{\boldsymbol{u}} \min_{\boldsymbol{\xi}_x} r(\boldsymbol{x}, \boldsymbol{u}) + \gamma V\big(f(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{\xi}_x)\big) \quad \text{s.t.} \quad \boldsymbol{\xi}_x^T \boldsymbol{\xi}_x \leq \alpha^2.$$

Substituting the Taylor expansion for $V(\boldsymbol{x}_{t+1})$, the dynamics model and the reward, the optimization is described by

$$V_{\text{tar}} = \max_{\boldsymbol{u}} \min_{\boldsymbol{\xi}_x} r + \gamma V + \gamma \nabla_x V^T f_c \Delta t + \gamma O(\boldsymbol{x}, \boldsymbol{u}, \Delta t)\Delta t$$

with the higher order terms $O(\boldsymbol{x}, \boldsymbol{u}, \Delta t)$. In the continuous-time limit, the higher-order terms and the discounting disappear, i.e., $\lim_{\Delta t \to 0} O(\boldsymbol{x}, \boldsymbol{u}, \Delta t)$=0 and $\lim_{\Delta t \to 0} \exp(-\rho \Delta t)$=1. Therefore, the optimal action is described by

$$\boldsymbol{u}_t = \arg\max_{\boldsymbol{u}} \nabla_x V^T \boldsymbol{B}\, \boldsymbol{u} - g_c(\boldsymbol{u}) \quad \Rightarrow \quad \boldsymbol{u}_t = \nabla \tilde{g}_c\big(\boldsymbol{B}^T \nabla_x V\big).$$

The optimal state disturbance is described by

$$\boldsymbol{\xi}_x^* = \arg\min_{\boldsymbol{\xi}_x} \nabla_x V^T \boldsymbol{\xi}_x \quad \text{s.t.} \quad \frac{1}{2}\left[\boldsymbol{\xi}_x^T \boldsymbol{\xi}_x - \alpha^2\right] \leq 0.$$

Algorithm 1 Robust Fitted Value Iteration (rFVI)

```
Input: Model $f_c(\boldsymbol{x}, \boldsymbol{u})$, Dataset $\mathcal{D}$ & Admissible Set $\Omega_\xi$
Result: Value Function $V^*(\boldsymbol{x};\, \psi^*)$
while not converged do
    // Compute Value Target for $\boldsymbol{x} \in \mathcal{D}$:
    $\boldsymbol{x}_\tau = \boldsymbol{x}_i + \int_0^\tau f_c(\boldsymbol{x}_t, \boldsymbol{u}_t, \boldsymbol{\xi}_t^x, \boldsymbol{\xi}_t^u, \boldsymbol{\xi}_t^o, \boldsymbol{\xi}_t^\theta) dt$
    $R_t = \int_0^t \exp(-\rho\tau)\, r_c(\boldsymbol{x}_\tau, \boldsymbol{u}_\tau) d\tau + \exp(-\rho t) V^k(\boldsymbol{x}_t)$
    $V_{\text{tar}}(\boldsymbol{x}_i) = \int_0^T \beta\, \exp(-\beta t)\, R_t\, dt + \exp(-\beta T) R_T$

    // Fit Value Function:
    $\psi_{k+1} = \arg\min_\psi \sum_{\boldsymbol{x} \in \mathcal{D}} \|V_{\text{tar}}(\boldsymbol{x}) - V(\boldsymbol{x}; \psi)\|^p$

    if RTDP rFVI then
        // Add samples from $\pi^{k+1}$ to FIFO buffer $\mathcal{D}$
        $\mathcal{D}^{k+1} = h(\mathcal{D}^k, \{\boldsymbol{x}_0^{k+1} \ldots \boldsymbol{x}_N^{k+1}\})$
    end if
end while
```

This constrained optimization can be solved using the Karush-Kuhn-Tucker (KKT) conditions. The resulting optimal adversarial state perturbation is described by

$$\boldsymbol{\xi}_x = -\alpha \frac{\nabla_x V}{\|\nabla_x V\|_2}.$$

□

Theorem 3. *For the adversarial model disturbance (Equation 8) with element-wise bounded amplitude (Equation 17), smooth drift and control matrix (i.e., $\boldsymbol{a}, \boldsymbol{B} \in C^1$) and $\boldsymbol{B}(\boldsymbol{\theta} + \boldsymbol{\xi}_\theta) \approx \boldsymbol{B}(\boldsymbol{\theta})$, the optimal continuous-time policy π and model disturbance $\boldsymbol{\xi}_\theta$ is described by*

$$\pi(\boldsymbol{x}) = \nabla\tilde{g}\left(\boldsymbol{B}(\boldsymbol{x})^T \nabla_x V\right), \qquad \boldsymbol{\xi}_\theta = -\boldsymbol{\Delta}_\nu\, \mathrm{sign}\left(\boldsymbol{z}_\theta\right) + \boldsymbol{\mu}_\nu$$

$$\textit{with} \;\; \boldsymbol{z}_\theta = \left(\frac{\partial \boldsymbol{a}(\boldsymbol{x};\, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}} + \frac{\partial \boldsymbol{B}(\boldsymbol{x};\, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \pi(\boldsymbol{x})\right)^T \nabla_x V,$$

parameter mean $\boldsymbol{\mu}_\nu = (\boldsymbol{\nu}_{max} + \boldsymbol{\nu}_{min})/2$ and parameter range $\boldsymbol{\Delta}_\nu = (\boldsymbol{\nu}_{max} - \boldsymbol{\nu}_{min})/2$.

Theorem 4. *For the adversarial observation disturbance (Equation 7) with bounded signal energy (Equation 16), smooth drift and control matrix (i.e., $\boldsymbol{a}, \boldsymbol{B} \in C^1$) and $\boldsymbol{B}(\boldsymbol{x} + \boldsymbol{\xi}_o) \approx \boldsymbol{B}(\boldsymbol{x})$, the optimal continuous-time policy π and observation disturbance $\boldsymbol{\xi}_o$ is described by*

$$\pi(\boldsymbol{x}) = \nabla\tilde{g}\left(\boldsymbol{B}(\boldsymbol{x})^T \nabla_x V\right), \qquad \boldsymbol{\xi}_o = -\alpha \frac{\boldsymbol{z}_o}{\|\boldsymbol{z}_o\|_2}$$

$$\textit{with} \;\; \boldsymbol{z}_o = \left(\frac{\partial \boldsymbol{a}(\boldsymbol{x};\, \boldsymbol{\theta})}{\partial \boldsymbol{x}} + \frac{\partial \boldsymbol{B}(\boldsymbol{x};\, \boldsymbol{\theta})}{\partial \boldsymbol{x}} \pi(\boldsymbol{x})\right)^T \nabla_x V.$$

Proof Sketch Theorem 3 Equation 14 can be written as

$$V_{\text{tar}} = \max_{\boldsymbol{u}} \min_{\boldsymbol{\xi}_\theta} r(\boldsymbol{x}, \boldsymbol{u}) + \gamma V\big(f(.)\big) \quad \text{s.t.} \quad (\boldsymbol{\xi}_\theta - \boldsymbol{\mu}_\nu)^2 \leq \boldsymbol{\Delta}_\nu^2$$

by replacing the admissible set Ω_A with an explicit constraint. In the following we abbreviate $\boldsymbol{B}(\boldsymbol{x}; \boldsymbol{\theta} + \boldsymbol{\xi}_\theta)$ as $\boldsymbol{B}_\xi$ and

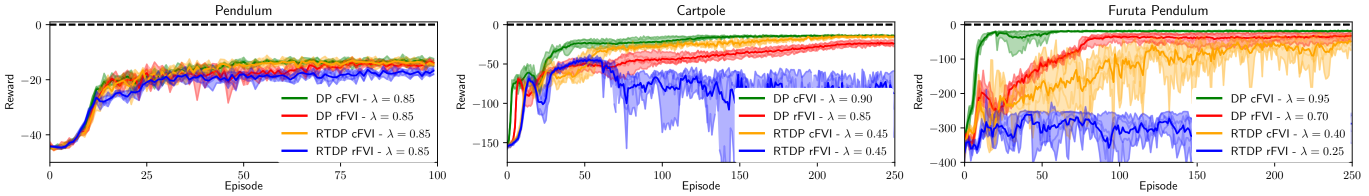

Figure 3. The learning curves for DP rFVI, DP cFVI, RTDP cFVI and RTDP rFVI averaged over 5 seeds. The shaded area displays the *min/max* range between seeds. DP rFVI learns slower compared to DP cFVI on the carpole and Furuta pendulum as the adversary prevents learning. RTDP rFVI does not learn the task as the adversary is too strong for the online variant of rFVI despite using the identical admissible set as the offline variant DP rFVI.

$\boldsymbol{a}(\boldsymbol{x};\boldsymbol{\theta}+\boldsymbol{\xi}_\theta)$ as $\boldsymbol{a}_\xi$. Substituting the Taylor expansion for $V(\boldsymbol{x}_{t+1})$, the dynamics models and reward yields

$$\frac{V_{\text{tar}} - \gamma V}{\Delta t} = q_c + \max_{\boldsymbol{u}} \min_{\boldsymbol{\xi}} \left[\gamma \nabla_x V^T \left(\boldsymbol{a}_\xi + \boldsymbol{B}_\xi \boldsymbol{u}\right) + \gamma O(.) - g_c\right]$$

In the continuous-time limit the optimal action and disturbance is determined by

$$\boldsymbol{u}^*, \boldsymbol{\xi}_\theta^* = \max_{\boldsymbol{u}} \min_{\boldsymbol{\xi}} \left[\nabla_x V^T \left(\boldsymbol{a}_\xi + \boldsymbol{B}_\xi \boldsymbol{u}\right) - g_c(\boldsymbol{u})\right].$$

This nested max-min optimization can be solved by first solving the inner optimization w.r.t. to $\boldsymbol{u}$ and substituting this solution into the outer maximization. The Lagrangian for the optimal model disturbance is described by

$$\boldsymbol{\xi}^* = \arg\min_{\boldsymbol{\xi}} \; \nabla_x V^T \left(\boldsymbol{a}_\xi + \boldsymbol{B}_\xi \boldsymbol{u}\right) + \frac{1}{2}\boldsymbol{\lambda}^T \left((\boldsymbol{\xi}_\theta - \boldsymbol{\mu}_\nu)^2 - \boldsymbol{\Delta}_\nu^2\right).$$

Using the KKT conditions this optimization can be solved. The stationarity condition yields

$$\boldsymbol{z}_\theta + \boldsymbol{\lambda}^T (\boldsymbol{\xi}_\theta - \boldsymbol{\mu}_\nu) := 0 \qquad \Rightarrow \qquad \boldsymbol{\xi}_\theta^* = -\boldsymbol{z}_\theta \oslash \boldsymbol{\lambda} + \boldsymbol{\mu}_\nu$$

with the elementwise division $\oslash$. Using the primal feasibility and the complementary slackness, the optimal $\boldsymbol{\lambda}^*$ can be computed. The resulting optimal model disturbance is described by

$$\boldsymbol{\xi}_\theta^*(\boldsymbol{u}) = -\boldsymbol{\Delta}_\nu \operatorname{sign}\left(\boldsymbol{z}_\theta(\boldsymbol{u})\right) + \boldsymbol{\mu}_\nu$$

as $\boldsymbol{z}_\theta \oslash \|\boldsymbol{z}_\theta\|_1 = \operatorname{sign}(\boldsymbol{z}_\theta)$. The action can be computed by

$$\boldsymbol{u}^* = \arg\max_{\boldsymbol{u}} \nabla_x V^T \left[\boldsymbol{a}(\boldsymbol{\xi}_\theta^*(\boldsymbol{u})) + \boldsymbol{B}\left(\boldsymbol{\xi}_\theta^*(\boldsymbol{u})\right)\boldsymbol{u}\right] - g_c(\boldsymbol{u}).$$

Due to the envelope theorem [32], the extrema is described by

$$\boldsymbol{B}(\boldsymbol{x};\boldsymbol{\theta} + \boldsymbol{\xi}_\theta^*(\boldsymbol{u}))^T \nabla_x V - g_c(\boldsymbol{u}) := 0.$$

This expression cannot be solved without approximation as $\boldsymbol{B}$ does not necessarily be invertible w.r.t. $\boldsymbol{\theta}$. Approximating $\boldsymbol{B}(\boldsymbol{x};\boldsymbol{\theta}+\boldsymbol{\xi}^*(\boldsymbol{u})) \approx \boldsymbol{B}(\boldsymbol{x};\boldsymbol{\theta})$, lets one solve for $\boldsymbol{u}$. In this case the optimal action $\boldsymbol{u}^*$ is described by $\boldsymbol{u}^*{=}\nabla\tilde{g}(\boldsymbol{B}(\boldsymbol{x};\boldsymbol{\theta})^T \nabla_x V)$. This approximation implies that neither agent or the adversary can react to the action of the other and must choose simultaneously. This assumption is common in prior works [5]. □

C. Algorithm

Using the theoretical insights from the previous section, robust fitted value iteration can be derived. Instead of computing the value function target using only the optimal action as in cFVI, rFVI includes the four adversaries to learn a optimal policy that is robust to changes in the dynamics. Therefore, the value function target is computed using

$$V_{\text{tar}}^{k+1} = r\left(\boldsymbol{x},\, \boldsymbol{u}^k\right) + \gamma V^k \left(f\left(\boldsymbol{x},\, \boldsymbol{u}^k,\, \boldsymbol{\xi}_x^k,\, \boldsymbol{\xi}_u^k,\, \boldsymbol{\xi}_o^k,\, \boldsymbol{\xi}_\theta^k\right)\right) \quad (18)$$

where actions $\boldsymbol{u}^k$ and disturbances $\boldsymbol{\xi}_i$ are determined according to Table I. Iterating between computing the target and fitting the value function network (Equation 13) enables the learning of the robust optimal value function and robust optimal policy.

N-Step Value Function Target The learning can be accelerated by using the exponentially weighted n-step value target instead of the 1-step target, as shown by the classical eligibility traces [17], generalized advantage estimation [33, 34] or model based value-expansion [35]. In the continuous limit this target is described by

$$\begin{aligned} V_{\text{tar}}(\boldsymbol{x}_0) &= \int_0^T \beta \, \exp(-\beta t) \, R_t \, dt + \exp(-\beta T) R_T, \\ R_t &= \int_0^t \exp(-\rho\tau) \, r_c(\boldsymbol{x}_\tau, \boldsymbol{u}_\tau) d\tau + \exp(-\rho t) V^k(\boldsymbol{x}_t), \\ \boldsymbol{x}_t &= \int_0^t f_c\left(\boldsymbol{x},\, \boldsymbol{u}^k,\, \boldsymbol{\xi}_x^k,\, \boldsymbol{\xi}_u^k,\, \boldsymbol{\xi}_o^k,\, \boldsymbol{\xi}_\theta^k\right) \, d\tau + \boldsymbol{x}_0, \end{aligned}$$

where β is the exponential decay factor. In practice we treat β as the hyperparameter and select T such that the weight of the R_T is $\exp\left(-\beta T\right) := 10^{-4}$.

Admissible Set For the state, action and observation adversary the signal energy is bounded. We limit the energy of $\boldsymbol{\xi}_x$, $\boldsymbol{\xi}_u$ and $\boldsymbol{\xi}_o$ as the non-adversarial disturbances are commonly modeled as multivariate Gaussian distribution. Therefore, the average energy is determined by the noise covariance matrix. For the model parameters $\boldsymbol{\theta}$ a common practice is to assume that the approximate model parameters have an model error of up to ±15% [1, 2]. Hence, we bound the amplitude of each component. To not overfit to the deterministic worst case system of V^k and enable the discovery of good actions, the amplitude of the adversarial actions of ξ_x, ξ_u, ξ_o is modulated using a Wiener process. This random process allows a continuous-time formulation that is agnostic to the sampling frequency.

Offline and Online rFVI The proposed approach is off-policy as the samples in the replay memory do not need to originate from the current policy π_k. Therefore, the dataset can either consist of a fixed dataset or be updated within each iteration. In the offline dynamic programming case, the dataset contains samples from the compact state domain $\mathcal{X}$. We refer to the

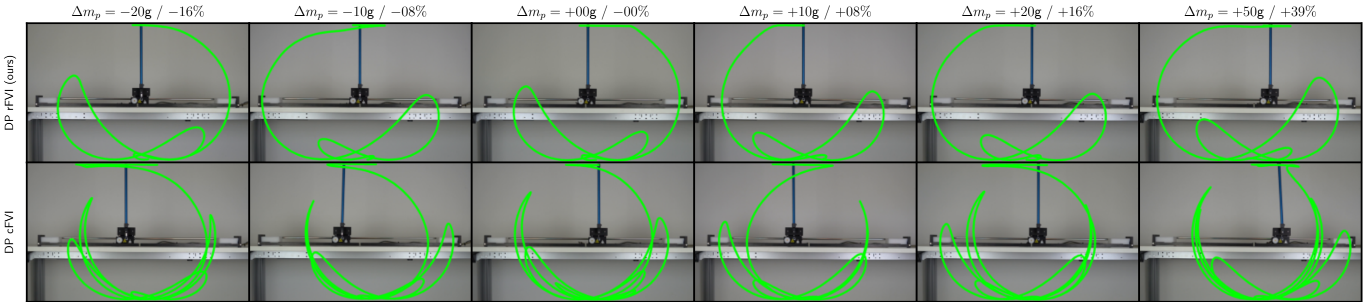
$\Delta m_p = -20\text{g} \,/\, -16\%$
$\Delta m_p = -10\text{g} \,/\, -08\%$
$\Delta m_p = +00\text{g} \,/\, -00\%$
$\Delta m_p = +10\text{g} \,/\, +08\%$
$\Delta m_p = +20\text{g} \,/\, +16\%$
$\Delta m_p = +50\text{g} \,/\, +39\%$
DP rFVI (ours)
DP cFVI

Figure 4. The tracked trajectories of DP rFVI and DP cFVI on the physical cartpole with varied pendulum masses. DP rFVI is capable to perform the swing-up for the varying pendulum mass. The qualitative performance does not change if the weight is added or reduced. DP cFVI can swing up and balance all varied pendulums but it requires more pre-swings for all configurations. During balancing the cart is not centered and the cart oscillates around the center for DP cFVI. The pendulum must significantly deviate from the target position before the DP cFVI policy breaks the stiction of the linear actuator. In contrast, the higher actions of DP rFVI break the stiction and balance the pendulum with the cart centered.

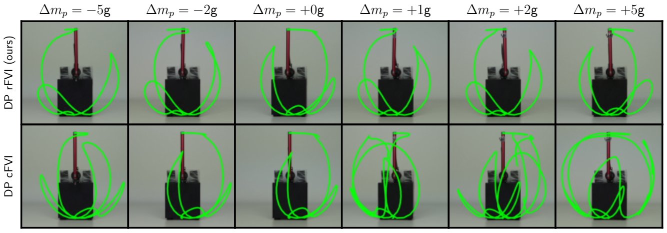
$\Delta m_p = -5\text{g}$
$\Delta m_p = -2\text{g}$
$\Delta m_p = +0\text{g}$
$\Delta m_p = +1\text{g}$
$\Delta m_p = +2\text{g}$
$\Delta m_p = +5\text{g}$
DP rFVI (ours)
DP cFVI

Figure 5. The tracked trajectories for DP rFVI and DP cFVI on the Furuta pendulum for different pendulum weights. The trajectories of rFVI do not significantly change when the pendulum mass altered. For DP cFVI the trajectories start to change when an additional weight is added. For these system dynamics, DP cFVI requires some failed swing-ups until the policy can balance the pendulum.

offline variant as DP rFVI. In the online case, the replay memory is updated with samples generated by the current policy π_k. Every iteration the states of the previous n-rollouts are added to the data and replace the oldest samples. This online update of state distribution performs real-time dynamic programming (RTDP) [36]. We refer to the online variant as RTDP rFVI. The pseudo code of DP cFVI and RTDP rFVI is summarized in Algorithm 1.

IV. Experiments

In the following non-linear control experiments we want to answer the following questions:

Q1: Does rFVI learn a robust policy that can be successfully transferred to the physical systems with different model parameters?
Q2: How does the policy obtained by rFVI differ qualitatively compared to cFVI and the deep RL baselines?

To answer these questions, we apply the algorithms to perform the swing-up task of the under-actuated cartpole (Fig. 4) and Furuta pendulum (Fig. 5). Both systems are standard environments for benchmarking non-linear control policies. We focus only on these two systems to perform extensive robustness experiments on the actual physical systems. To test the robustness with respect to model parameters, we attach small weights to the passive pendulum.

A. Experimental Setup

Systems The physical cartpole and Furuta pendulum are manufactured by Quanser [37] and voltage controlled. For the approximate simulation model we use the rigid-body dynamics model with the parameters supplied by the manufacturer. If we add negative weights to the pendulum, we attach the weights to the opposite lever of the pendulum. This moves the center of mass of the pendulum closer to the rotary axis. Therefore, this shift reduces the downward force and is equivalent to a lower pendulum mass.

Baselines The performance is compared to the actor-critic deep RL methods: DDPG [38], SAC [39] and PPO [34]. The robustness evaluation is only performed for the best performing baselines on the nominal physical system. The performance of all baselines on the nominal system is summarized in Table IV (Appendix). The initial state distribution is abbreviated by {SAC, PPO, DDPG}-U for a uniform distribution of the pendulum angle and {SAC, PPO, DDPG}-N for a Gaussian distribution. The baselines with Gaussian initial state distribution did not achieve robust performance on the nominal system. If the baseline uses uniform domain randomization the acronym is appended with UDR.

Evaluation To evaluate rFVI and the baselines we separately compare the state and action reward as these algorithms optimize a different objectives. Hence, these algorithms trade-off state and action associated rewards differently. It is expected that the worst-case optimization uses higher actions to prevent deviation from the optimal trajectory. On the physical system, the performance is evaluated using the 25th, 50th and 75th reward percentile as the reward distribution is multi-modal.

B. Experimental Results

The learning curves averaged over 5 seeds of DP rFVI and RTDP rFVI are shown in Figure 3. The results in simulation are summarized in Table II. DP rFVI learns a policy that obtains slightly lower reward compared to cFVI and the deep RL baselines. This lower reward is expected as the worst-case optimization yields conservative policies [26]. DP rFVI exhibits low variance between seeds but learns slower than DP cFVI. This slower learning is caused by the adversary which counteracts the learning progress. RTDP rFVI does not learn to successfully swing-up the Furuta pendulum and cartpole. Despite using the same admissible set for both variants, the adversary is too powerful for RTDP rFVI and

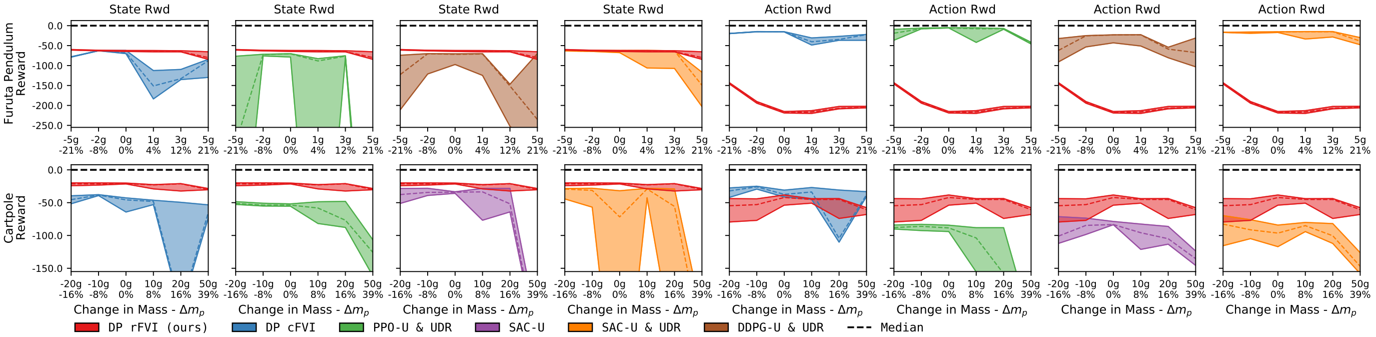

Figure 6. The 25th, 50th and 75th reward percentile for the physical Furuta pendulum and cartpole with varied pendulum weights. DP rFVI achieves higher state reward for real-world systems compared to the baselines. For the different weights the reward remains nearly constant. For the Furuta pendulum the action cost is significantly higher compared to the baselines as the DP rFVI causes a chattering during balancing due to the high actions and minor time delays in the control loop. If only the swing-up phase is considered the rewards are comparable.

Table II
The average rewards in simulation.

Algorithm	Sim Pendulum		Sim Cartpole		Sim Furuta Pendulum	
	State Rwd	Action Rwd	State Rwd	Action Rwd	State Rwd	Action Rwd
DP rFVI (ours)	−24.5 ± 00.1	**−08.3 ± 00.4**	**−15.5 ± 05.4**	−11.6 ± 02.0	−37.0 ± 13.2	−04.8 ± 02.7
DP cFVI	−22.3 ± 03.0	**−08.3 ± 03.2**	**−14.4 ± 02.8**	−09.9 ± 02.2	**−22.3 ± 02.5**	−05.9 ± 01.3
SAC-U	**−21.1 ± 05.2**	−09.4 ± 04.7	**−13.9 ± 02.4**	−10.4 ± 02.3	**−22.6 ± 02.9**	−05.9 ± 01.5
SAC-U & UDR	−22.6 ± 02.6	−08.9 ± 01.9	**−13.9 ± 02.6**	−10.3 ± 02.5	**−22.1 ± 02.6**	−06.1 ± 01.4
DDPG-U	**−20.9 ± 01.2**	−10.6 ± 01.3	−16.4 ± 04.7	−11.8 ± 04.9	**−24.6 ± 03.6**	−05.5 ± 01.9
DDPG-U & UDR	**−21.0 ± 04.8**	−11.5 ± 01.6	**−14.5 ± 03.5**	−12.8 ± 03.5	−26.1 ± 02.4	−06.2 ± 01.9
PPO-U	−24.5 ± 04.4	−09.0 ± 02.4	−82.2 ± 83.1	**−04.9 ± 01.1**	−34.9 ± 12.6	**−03.4 ± 01.1**
PPO-U & UDR	−24.5 ± 00.2	−11.1 ± 03.0	−55.9 ± 23.1	−09.8 ± 05.2	−42.9 ± 04.6	−06.1 ± 02.4

prevents learning. In this case policy does not discover the positive reward at the top as the adversary prevents balancing. Therefore, the policy is too pessimistic and converges to a bad local optima. The ablation study in the appendix shows, that RTDP rFVI learns a successful policy if the admissible sets are reduced. To overcome this problem one would need to bias the exploration to be optimistic.

The performance on the physical systems is summarized in Figure 4, 5 and 7.[1] Across the different parameters of both systems, rFVI achieves the highest state reward compared to cFVI and the deep RL baselines. The best performing trajectories between different configurations are nearly identical. Only for the cartpole the failure rates slightly increases when positive weights are added. In this case the pendulum is swung-up but cannot be stabilized, due to the backslash of the linear actuator. For cFVI and the deep RL baselines even the best trajectories deteriorate when weights are added to the pendulum. This is especially notable in Figure 7, where the deep RL baselines start to explore the complete state space when additional weights are added. The high state rewards of rFVI are obtained at the expense of higher action cost, which can be higher compared to some baselines on the physical system. To summarize rFVI obtains a robust policy that performs the swing-up with low state reward and more consistency than the baselines but uses higher actions.

Compared to the policies obtained by DP cFVI and the deep RL baselines, DP rFVI policy exerts higher actions. Therefore, DP rFVI achieves the robustness compared to the baselines by utilizing a stiffer feedback policy approaching bang-bang control. The higher actions are only observed on the physical system where the deviation from the optimal trajectory is inevitable. In simulation the action cost are comparable to the baselines. Therefore, the high actions originate from the feedback-term of the non-linear policy that tries to compensate the tracking error. The stiffer feedback policy is expected as traditional robust control approaches yield high feedback gains [40]. The stiffness of the rFVI policy is clearly visible in Figure 2. On the ridge leading up to the balancing point, the policy directly applies maximum action, if one deviates from the center of the ridge. In contrast, DP cFVI has a gradient that slightly increases the action when one deviates from the optimal trajectory.

For the cartpole the higher actions achieve much better performance. DP rFVI achieves stabilization of the cart in the center of the track (Figure 4). The higher actions break the stiction and overcome the backslash of the geared cogwheel actuator during the balancing. For DP cFVI and the baselines, the cart oscillates around the center. The pendulum angle has to deviate significantly until the actions become large and break the stiction. For the Furuta pendulum the high actions are more robust during the swing-up phase. However, during the balancing the lower link starts to chatter due to the high-frequency switching between high actions. This switching is caused by minor delays in the control loop and the very low friction of the pendulum. About 90% of the action cost of DP rFVI for the Furuta pendulum is incurred during the stabilization. DP cFVI incurs only 10% of the action cost during stabilization. If one would only consider the swing-up phase, the reward of DP rFVI is higher compared to the baselines. To summarize, high-stiffness feedback policies are robust to changes in dynamics. However, this robustness can also make the resulting policies more sensitive to other sources error not included in the specification, e.g., control delays.

Besides the performance evaluation of DP rFVI, the experiments show that DP cFVI achieves comparable performance than the deep RL baselines with domain randomization. Despite overfitting to a deterministic approximate model, DP cFVI is able to be robust against some variations in model parameters. This result suggests that for these two physical systems, preventing the distribution shift by solving

[1] Videos of the experiments at https://sites.google.com/view/rfvi.

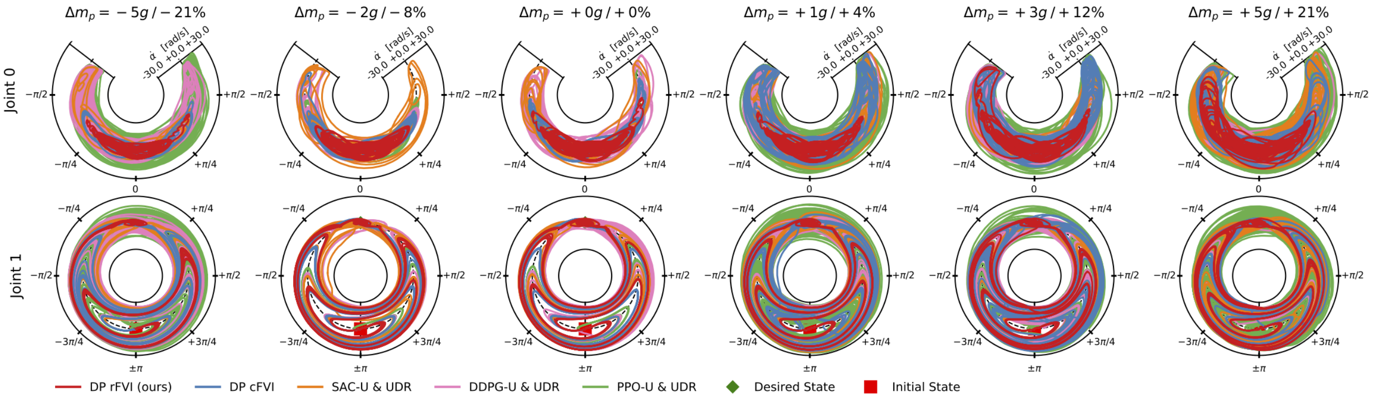

Figure 7. The roll-outs of DP rFVI, DP cFVI and the deep RL baselines with domain randomization the physical Furuta pendulum. The different columns correspond to different pendulum masses. The deviation from the dashed center line corresponds to the joint velocity. DP rFVI achieves a consistent swing-up for the different pendulum masses. In contrast to DP rFVI, the baselines start to deviate strongly from trajectories on the nominal system. When weights are added the baselines start to cover the complete state-space. A figure displaying the roll-outs per algorithm is provided in the appendix.

for the policy on the compact state domain obtains a policy with comparable robustness as uniform domain randomization. Furthermore, the deep RL performance increases with larger state distribution and using the maximum-entropy formulation on the physical system. Therefore, the experiments suggest that the state-distribution of the policy affects the policy robustness. This correlation should be investigated in-depth in future work.

V. Related Work

Robust Policies Learning robust policies has been approached by (1) changing the optimization objective that balances risk and reward [41–43], (2) introducing a adversary to optimize the worst-case performance [5–7, 44, 45] and (3) randomizing the dynamics model to be robust to the model parameters [2, 2–4, 9, 46]. In robotics, domain randomization is most widely used to achieve successful sim2real transfer. For example domain randomization was used for in-hand manipulation [46], ball-in-a-cup [2], locomotion [9], manipulation [3, 4].

In this paper we focus on the adversarial formulation. This approach has been extensively used for continuous control tasks [7, 8, 13, 18, 23]. For example, Pinto et. al. [7, 18] used a separate agent as adversary controlling an additive control input. This adversary maximized the negative reward using a standard actor-critic learning algorithm. The agent and adversary do not share any information. Therefore, an additional optimization is required to optimize the adversary. A different approach by Mandlekar et. al. [8] used auxiliary loss to maximize the policy actions. Both approaches are model-free. In contrast to these approaches, our approach derives the adversarial perturbations using analytic expressions derived directly from the Hamilton-Jacobi-Isaacs (HJI) equation. Therefore, our approach shares knowledge between the actor and adversary due to a shared value function and requires no additional optimization. However, our approach requires knowledge of the model to compute the actions and perturbations analytically. The approach is similar to Morimoto and Doya [13]. In contrast to this work, we extend the analytic solutions to state, action, observation and model disturbances, do not require a control-affine disturbance model, and use the constrained formulation rather than the penalized formulation.

Continuous-Time Reinforcement Learning Various approaches have been proposed to solve the Hamilton-Jacobi-Bellman (HJB) differential equation with the machine learning toolset. These methods can be divided into trajectory and state-space based methods. Trajectory based methods solve the stochastic HJB along a trajectory using path integral control [47–49] or forward-backward stochastic differential equations [50, 51]. State-space based methods solve the HJB globally to obtain a optimal non-linear controller applicable on the complete state domain. Classical approaches discretize the problem and solve the HJB or HJI using a PDE solver [5]. To overcome the curse of dimensionality of the grid based methods, machine learning methods have proposed to use various function approximators ranging from polynomial functions [52, 53], kernels [54] to deep networks [12, 14, 55, 56]. In this paper, we utilize the state-space based approach solve the HJI via dynamic programming. The value function is approximated using a deep network and optimal value function is solved via value iteration.

VI. Conclusion and Future Work

We proposed robust fitted value iteration (rFVI). This algorithm solves the adversarial continuous-time reinforcement learning problem for continuous states, action and adversary via value iteration. To enable the efficient usage of value iteration, we presented analytic expressions for the adversarial disturbances for the state, action, observation and model adversary. Therefore, our derivations extend existing analytic expressions for continuous time RL from literature [11–14]. The non-linear control experiments using the physical cartpole and Furuta pendulum showed that rFVI is robust to variations in model parameters and obtains higher state-rewards compared to the deep RL baselines with uniform domain randomization. The robustness of rFVI is achieved by utilizing a stiffer feedback policy that exerts higher actions compared to the baselines.

In future work we plan to learn the admissible sets from data from the physical system. In domain randomization, the automatic tuning of the parameter distributions has been very successful [2, 4]. However, these approaches are not directly transferable as one would also need to estimate the admissible set of the action, state and observation and not only the system parameter distribution as in domain randomization.

Acknowledgements

M. Lutter was an intern at Nvidia during this project. A. Garg was partially supported by CIFAR AI Chair. We also want to thank Fabio Muratore, Joe Watson and the RSS reviewers for their feedback. Furthermore, we want to thank the open-source projects SimuRLacra [57], MushroomRL [58], NumPy [59] and PyTorch [60].

References

[1] F. Muratore, F. Treede, M. Gienger, and J. Peters, "Domain randomization for simulation-based policy optimization with transferability assessment," in *Conference on Robot Learning (CoRL)*, 2018.

[2] F. Muratore, C. Eilers, M. Gienger, and J. Peters, "Data-efficient domain randomization with bayesian optimization," *IEEE Robotics and Automation Letters (RAL)*, 2021.

[3] Y. Chebotar, A. Handa, V. Makoviychuk, M. Macklin, J. Issac, N. Ratliff, and D. Fox, "Closing the sim-to-real loop: Adapting simulation randomization with real world experience," 2019.

[4] F. Ramos, R. C. Possas, and D. Fox, "Bayessim: adaptive domain randomization via probabilistic inference for robotics simulators," 2019.

[5] S. Bansal, M. Chen, S. Herbert, and C. J. Tomlin, "Hamilton-Jacobi reachability: A brief overview and recent advances," 2017.

[6] R. Isaacs, *Differential games: a mathematical theory with applications to warfare and pursuit, control and optimization*. Courier Corporation, 1999.

[7] L. Pinto, J. Davidson, R. Sukthankar, and A. Gupta, "Robust adversarial reinforcement learning," in *International Conference on Machine Learning (ICML)*, 2017.

[8] A. Mandlekar, Y. Zhu, A. Garg, L. Fei-Fei, and S. Savarese, "Adversarially robust policy learning: Active construction of physically-plausible perturbations," in *International Conference on Intelligent Robots and Systems (IROS)*, 2017.

[9] Z. Xie, X. Da, M. van de Panne, B. Babich, and A. Garg, "Dynamics Randomization Revisited: A Case Study for Quadrupedal Locomotion," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021.

[10] J. Garcıa and F. Fernández, "A comprehensive survey on safe reinforcement learning," *Journal of Machine Learning Research*, 2015.

[11] S. E. Lyshevski, "Optimal control of nonlinear continuous-time systems: design of bounded controllers via generalized nonquadratic functionals," in *American Control Conference (ACC)*, vol. 1, pp. 205–209, IEEE, 1998.

[12] K. Doya, "Reinforcement learning in continuous time and space," *Neural computation*, vol. 12, no. 1, pp. 219–245, 2000.

[13] J. Morimoto and K. Doya, "Robust reinforcement learning," *Neural computation*, 2005.

[14] M. Lutter, B. Belousov, K. Listmann, D. Clever, and J. Peters, "HJB optimal feedback control with deep differential value functions and action constraints," in *Conference on Robot Learning (CoRL)*, 2019.

[15] R. Bellman, *Dynamic Programming*. USA: Princeton University Press, 1957.

[16] M. L. Puterman, *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 1994.

[17] R. S. Sutton, A. G. Barto, *et al.*, *Introduction to reinforcement learning*. MIT press Cambridge, 1998.

[18] L. Pinto, J. Davidson, and A. Gupta, "Supervision via competition: Robot adversaries for learning tasks," in *International Conference on Robotics and Automation (ICRA)*, 2017.

[19] H. Zhang, H. Chen, C. Xiao, B. Li, D. Boning, and C.-J. Hsieh, "Robust deep reinforcement learning against adversarial perturbations on observations," *arXiv preprint arXiv:2003.08938*, 2020.

[20] M. Heger, "Consideration of risk in reinforcement learning," in *Machine Learning Proceedings*, Elsevier, 1994.

[21] M. L. Littman, "Markov games as a framework for multi-agent reinforcement learning," in *Machine learning proceedings*, Elsevier, 1994.

[22] A. Nilim and L. El Ghaoui, "Robust control of markov decision processes with uncertain transition matrices," *Operations Research*, 2005.

[23] C. Tessler, Y. Efroni, and S. Mannor, "Action robust reinforcement learning and applications in continuous control," in *International Conference on Machine Learning (ICML)*, 2019.

[24] A. Pattanaik, Z. Tang, S. Liu, G. Bommannan, and G. Chowdhary, "Robust deep reinforcement learning with adversarial attacks," *arXiv preprint arXiv:1712.03632*, 2017.

[25] A. Gleave, M. Dennis, C. Wild, N. Kant, S. Levine, and S. Russell, "Adversarial policies: Attacking deep reinforcement learning," *arXiv preprint arXiv:1905.10615*, 2019.

[26] H. Xu and S. Mannor, "Robustness and generalization," *Machine learning*, 2012.

[27] M. Lutter, S. Mannor, J. Peters, D. Fox, and A. Garg, "Value Iteration in Continuous Actions, States and Time," in *International Conference on Machine Learning (ICML)*, 2021.

[28] D. Ernst, P. Geurts, and L. Wehenkel, "Tree-based batch mode reinforcement learning," *Journal of Machine Learning Research*, 2005.

[29] A. massoud Farahmand, M. Ghavamzadeh,

C. Szepesvári, and S. Mannor, "Regularized fitted q-iteration for planning in continuous-space markovian decision problems," in *American Control Conference (ACC)*, IEEE, 2009.

[30] M. Riedmiller, "Neural fitted q iteration–first experiences with a data efficient neural reinforcement learning method," in *European Conference on Machine Learning*, Springer, 2005.

[31] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, p. 529, 2015.

[32] M. Carter, *Foundations of mathematical economics*. MIT press, 2001.

[33] J. Schulman, S. Levine, P. Abbeel, M. I. Jordan, and P. Moritz, "Trust region policy optimization.," in *Icml*, vol. 37, pp. 1889–1897, 2015.

[34] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[35] V. Feinberg, A. Wan, I. Stoica, M. I. Jordan, J. E. Gonzalez, and S. Levine, "Model-based value estimation for efficient model-free reinforcement learning," *arXiv preprint arXiv:1803.00101*, 2018.

[36] A. G. Barto, S. J. Bradtke, and S. P. Singh, "Learning to act using real-time dynamic programming," *Artificial intelligence*, 1995.

[37] Quanser, "Quanser courseware and resources." https://www.quanser.com/solution/control-systems/, 2018.

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

[39] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *International Conference on Machine Learning*, 2018.

[40] K. J. Åström, L. Neumann, and P.-O. Gutman, "A comparison between robust and adaptive control of uncertain systems," *IFAC Proceedings Volumes*, 1987.

[41] V. S. Borkar, "A sensitivity formula for risk-sensitive cost and the actor–critic algorithm," *Systems & Control Letters*, 2001.

[42] Y. Chow, A. Tamar, S. Mannor, and M. Pavone, "Risk-sensitive and robust decision-making: a cvar optimization approach," *arXiv preprint arXiv:1506.02188*, 2015.

[43] H. Bharadhwaj, A. Kumar, N. Rhinehart, S. Levine, F. Shkurti, and A. Garg, "Conservative safety critics for exploration," in *International Conference on Learning Representations (ICLR)*, 2021.

[44] A. Tamar, H. Xu, and S. Mannor, "Scaling up robust mdps by reinforcement learning," *arXiv preprint arXiv:1306.6189*, 2013.

[45] J. Harrison*, A. Garg*, B. Ivanovic, Y. Zhu, S. Savarese, L. Fei-Fei, and M. Pavone (* equal contribution), "AdaPT: Zero-Shot Adaptive Policy Transfer for Stochastic Dynamical Systems," in *International Symposium on Robotics Research (ISRR)*, 2017.

[46] M. Andrychowicz, B. Baker, M. Chociej, R. Jozefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray, *et al.*, "Learning dexterous in-hand manipulation," *The International Journal of Robotics Research (IJRR)*, 2020.

[47] H. J. Kappen, "Linear theory for control of nonlinear stochastic systems," *Physical review letters*, 2005.

[48] E. Todorov, "Linearly-solvable markov decision problems," in *Advances in neural information processing systems*, 2007.

[49] E. Theodorou, J. Buchli, and S. Schaal, "Reinforcement learning of motor skills in high dimensions: A path integral approach," in *International Conference on Robotics and Automation (ICRA)*, IEEE, 2010.

[50] M. Pereira, Z. Wang, I. Exarchos, and E. Theodorou, "Learning deep stochastic optimal control policies using forward-backward sdes," in *Robotics: science and systems*, 2019.

[51] M. A. Pereira, Z. Wang, I. Exarchos, and E. A. Theodorou, "Safe optimal control using stochastic barrier functions and deep forward-backward sdes," *arXiv preprint arXiv:2009.01196*, 2020.

[52] X. Yang, D. Liu, and D. Wang, "Reinforcement learning for adaptive optimal control of unknown continuous-time nonlinear systems with input constraints," *International Journal of Control*, 2014.

[53] D. Liu, D. Wang, F.-Y. Wang, H. Li, and X. Yang, "Neural-network-based online hjb solution for optimal robust guaranteed cost control of continuous-time uncertain nonlinear systems," *IEEE transactions on cybernetics*, vol. 44, no. 12, pp. 2834–2847, 2014.

[54] P. Hennig, "Optimal reinforcement learning for gaussian systems," in *Advances in Neural Information Processing Systems*, 2011.

[55] Y. Tassa and T. Erez, "Least squares solutions of the HJB equation with neural network value-function approximators," *IEEE transactions on neural networks*, vol. 18, no. 4, pp. 1031–1041, 2007.

[56] J. Kim, J. Shin, and I. Yang, "Hamilton-jacobi deep q-learning for deterministic continuous-time systems with lipschitz continuous controls," *arXiv preprint arXiv:2010.14087*, 2020.

[57] F. Muratore, "Simurlacra - a framework for reinforcement learning from randomized simulations." https://github.com/famura/SimuRLacra, 2020.

[58] C. D'Eramo, D. Tateo, A. Bonarini, M. Restelli, and J. Peters, "Mushroomrl: Simplifying reinforcement learning research." https://github.com/MushroomRL/mushroom-rl, 2020.

[59] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río,

M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, “Array programming with NumPy,” *Nature*, 2020.

[60] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, “Pytorch: An imperative style, high-performance deep learning library,” in *Advances in Neural Information Processing Systems 32*, 2019.

[61] H. K. Khalil and J. W. Grizzle, *Nonlinear systems*, vol. 3. Prentice hall Upper Saddle River, NJ, 2002.

[62] F. Berkenkamp, M. Turchetta, A. Schoellig, and A. Krause, “Safe model-based reinforcement learning with stability guarantees,” in *Advances in neural information processing systems*, 2017.

[63] S. M. Richards, F. Berkenkamp, and A. Krause, “The lyapunov neural network: Adaptive stability certification for safe learning of dynamical systems,” *arXiv preprint arXiv:1808.00924*, 2018.

[64] J. Z. Kolter and G. Manek, “Learning stable deep dynamics models,” in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[65] Y.-C. Chang, N. Roohi, and S. Gao, “Neural lyapunov control,” in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Appendix

A. State Disturbance Proof

Theorem. *For the adversarial state disturbance (Equation 5) with bounded in signal energy (Equation 16), the optimal continuous time policy π^k and state disturbance $\boldsymbol{\xi}_x^k$ is described by*

$$\pi^k(\boldsymbol{x}) = \nabla\tilde{g}\left(\boldsymbol{B}(\boldsymbol{x})^T \nabla_x V^k\right) \qquad \boldsymbol{\xi}^k = -\alpha \frac{\nabla_x V^k}{\|\nabla_x V^k\|_2}.$$

Proof. Equation 14 can be formulated with the explicit constraint, i.e.,

$$\begin{aligned} V_{\text{tar}} = \max_{\boldsymbol{u}} \ \min_{\boldsymbol{\xi}_x} \ & r(\boldsymbol{x}_t, \boldsymbol{u}) + \gamma V\big(f(\boldsymbol{x}_t, \boldsymbol{u}, \ \boldsymbol{\xi}_x)\big) \\ & \text{with} \ \ \boldsymbol{\xi}_x^T \boldsymbol{\xi}_x - \alpha^2 \leq 0. \end{aligned}$$

Substituting the Taylor expansion, the dynamics model and the reward yields

$$\begin{aligned} V_{\text{tar}} &= \max_{\boldsymbol{u}} \ \min_{\boldsymbol{\xi}_x} \ r + \gamma V + \gamma \nabla_x V^T f_c \Delta t + \gamma O \Delta t \\ \frac{V_{\text{tar}} - \gamma V}{\Delta t} &= q_c + \max_{\boldsymbol{u}} \left[\nabla_x V^T \ (\boldsymbol{a} + \boldsymbol{B}\boldsymbol{u}) + \gamma O - g_c\right] \\ &\qquad + \min_{\boldsymbol{\xi}} \left[\nabla_x V^T \boldsymbol{\xi}_x\right] \end{aligned}$$

with the higher order terms $O(\boldsymbol{x}, \boldsymbol{u}, \Delta t)$. In the continuous time limit, the higher-order terms disappear, i.e., $\lim_{t \to 0} O = 0$. Therefore, the optimal action is described by

$$\begin{aligned} \boldsymbol{u}_t &= \arg\max_{\boldsymbol{u}} \ \nabla_x V^T \boldsymbol{B}(\boldsymbol{x}_t) \ \boldsymbol{u} - g_c(\boldsymbol{u}) \\ \Rightarrow \ \ \boldsymbol{u}_t &= \nabla \tilde{g}_c \left(\boldsymbol{B}(\boldsymbol{x}_t) \nabla_x V\right). \end{aligned}$$

The optimal state disturbance is described by

$$\boldsymbol{\xi}_x^* = \arg\min_{\boldsymbol{\xi}_x} \ \nabla_x V^T \boldsymbol{\xi}_x \quad \text{with} \quad \boldsymbol{\xi}_x^T \boldsymbol{\xi}_x - \alpha^2 \leq 0.$$

This constrained optimization can be solved using the Karush-Kuhn-Tucker (KKT) conditions, i.e.,

$$\nabla_x V + 2\lambda \, \boldsymbol{\xi}_x = 0 \quad \Rightarrow \quad \boldsymbol{\xi}_x^* = -\frac{1}{2\lambda} \nabla_x V$$

with the Lagrangian multiplier $\lambda \geq 0$. From primal feasibility and the complementary slackness condition of the KKT conditions follows that

$$\begin{aligned} & \frac{1}{4\lambda^2} \nabla_x V^T \nabla_x V - \alpha^2 \leq 0 \\ & \qquad \Rightarrow \lambda \geq \frac{1}{2\alpha} \sqrt{\nabla_x V^T \nabla_x V} \\ & \lambda \left(\frac{1}{4\lambda^2} \nabla_x V^T \nabla_x V - \alpha^2\right) = 0 \\ & \qquad \Rightarrow \lambda_0 = 0, \ \ \lambda_1 = \frac{1}{2\alpha} \sqrt{\nabla_x V^T \nabla_x V}. \end{aligned}$$

Therefore, the optimal adversarial state perturbation is described by

$$\boldsymbol{\xi}^k = -\alpha \frac{V_x^k}{\|V_x^k\|_2}.$$

This solution is intuitive as the adversary wants to minimize the reward and this disturbance performs steepest descent using the largest step-size. Therefore, the perturbation is always on the constraint. □

B. Action Disturbance Proof

Theorem. *For the adversarial action disturbance (Equation 5) with bounded in signal energy (Equation 16), the optimal continuous time policy π^k and action disturbance $\boldsymbol{\xi}_u^k$ is described by*

$$\pi^k(\boldsymbol{x}) = \nabla\tilde{g}\left(\boldsymbol{B}(\boldsymbol{x})^T \nabla_x V^k\right) \quad \boldsymbol{\xi}_u^k = -\alpha \frac{\boldsymbol{B}(\boldsymbol{x})^T \nabla_x V^k}{\|\boldsymbol{B}(\boldsymbol{x})^T \nabla_x V^k\|_2}.$$

Proof. Equation 14 can be formulated with the explicit constraint, i.e.,

$$V_{\text{tar}} = \max_{\boldsymbol{u}} \ \min_{\boldsymbol{\xi}_u} \ r(\boldsymbol{x}, \boldsymbol{u}) + \gamma V\big(f(\boldsymbol{x}, \boldsymbol{u}, \ \boldsymbol{\xi}_u)\big) \ \ \text{with} \ \ \boldsymbol{\xi}_u^T \boldsymbol{\xi}_u \leq \alpha^2.$$

Substituting the Taylor expansion, the dynamics model and the reward yields

$$\begin{aligned} V_{\text{tar}} &= \max_{\boldsymbol{u}} \ \min_{\boldsymbol{\xi}_u} \ r + \gamma V + \gamma \nabla_x V^T f_c \Delta t + \gamma O \Delta t \\ \frac{V_{\text{tar}} - \gamma V}{\Delta t} &= q_c + \max_{\boldsymbol{u}} \left[\nabla_x V^T \ (\boldsymbol{a} + \boldsymbol{B}\boldsymbol{u}) + \gamma O - g_c\right] \\ &\qquad + \min_{\boldsymbol{\xi}} \left[\nabla_x V^T \boldsymbol{B} \, \boldsymbol{\xi}_u\right] \end{aligned}$$

with the higher order terms $O(\boldsymbol{x}, \boldsymbol{u}, \Delta t)$. In the continuous time limit, the higher-order terms disappear, i.e., $\lim_{t \to 0} O = 0$. Therefore, the optimal action is described by

$$\begin{aligned} \boldsymbol{u}_t &= \arg\max_{\boldsymbol{u}} \ \nabla_x V^T \boldsymbol{B}(\boldsymbol{x}) \ \boldsymbol{u} - g_c(\boldsymbol{u}) \\ \Rightarrow \ \ \boldsymbol{u}_t &= \nabla \tilde{g}_c \left(\boldsymbol{B}(\boldsymbol{x}) \nabla_x V\right). \end{aligned}$$

The optimal state disturbance is described by

$$\boldsymbol{\xi}_u^* = \arg\min_{\boldsymbol{\xi}_u} \ \nabla_x V^T \boldsymbol{B}(\boldsymbol{x}) \, \boldsymbol{\xi}_u \quad \text{with} \quad \boldsymbol{\xi}_u^T \boldsymbol{\xi}_u \leq \alpha^2.$$

This constrained optimization can be solved using the Karush-Kuhn-Tucker (KKT) conditions, i.e.,

$$\boldsymbol{B}(\boldsymbol{x}_t)^T \nabla_x V + 2\lambda \, \boldsymbol{\xi}_u = 0 \quad \Rightarrow \quad \boldsymbol{\xi}_u^* = -\frac{1}{2\lambda} \boldsymbol{B}(\boldsymbol{x}_t)^T \nabla_x V$$

with the Lagrangian multiplier $\lambda \geq 0$. From primal feasibility and the complementary slackness condition of the KKT conditions follows that

$$\begin{aligned} & \frac{1}{4\lambda^2} \nabla_x V^T \boldsymbol{B}\boldsymbol{B}^T \nabla_x V - \alpha^2 \leq 0 \\ & \qquad \Rightarrow \lambda \geq \frac{1}{2\alpha} \sqrt{\nabla_x V^T \boldsymbol{B}\boldsymbol{B}^T \nabla_x V} \\ & \lambda \left(\frac{1}{4\lambda^2} \nabla_x V^T \boldsymbol{B}\boldsymbol{B}^T \nabla_x V - \alpha^2\right) = 0 \\ & \qquad \Rightarrow \lambda_0 = 0, \ \ \lambda_1 = \frac{1}{2\alpha} \sqrt{\nabla_x V^T \boldsymbol{B}\boldsymbol{B}^T \nabla_x V}. \end{aligned}$$

Therefore, the optimal adversarial state perturbation is described by

$$\boldsymbol{\xi}_u^k = -\alpha \frac{\boldsymbol{B}(\boldsymbol{x}_t)^T V_x^k}{\|\boldsymbol{B}(\boldsymbol{x}_t)^T V_x^k\|_2}.$$

□

C. Model Disturbance Proof

Theorem. *For the adversarial model disturbance (Equation 8) with element-wise bounded amplitude (Equation 17), smooth drift and control matrix (i.e., $\boldsymbol{a}, \boldsymbol{B} \in C^1$) and $\boldsymbol{B}(\theta + \boldsymbol{\xi}_\theta) \approx \boldsymbol{B}(\theta)$, the optimal continuous time policy π^k and model disturbance $\boldsymbol{\xi}_\theta^k$ is described by*

$$\pi^k(\boldsymbol{x}) = \nabla \tilde{g}\left(\boldsymbol{B}(\boldsymbol{x})^T \nabla_x V^k\right) \qquad \boldsymbol{\xi}_\theta^k = -\boldsymbol{\Delta}_\nu \operatorname{sign}\left(\boldsymbol{z}_\theta\right) + \boldsymbol{\mu}_\nu$$
$$\textit{with } \; \boldsymbol{z}_\theta = \left(\frac{\partial \boldsymbol{B}}{\partial \theta}\boldsymbol{u}^* + \frac{\partial \boldsymbol{a}}{\partial \theta}\right)^T V_x^*,$$

mean $\boldsymbol{\mu_\nu} = \left(\boldsymbol{\nu}_{max} + \boldsymbol{\nu}_{min}\right)/2$ and range $\boldsymbol{\Delta_\nu} = \left(\boldsymbol{\nu}_{max} - \boldsymbol{\nu}_{min}\right)/2$.

Proof. Equation 14 can be written with the explicit constraint instead of the infimum with the admissible set Ω_A, i.e.,

$$V_{\text{tar}} = \max_{\boldsymbol{u}} \min_{\boldsymbol{\xi}_\theta} r(\boldsymbol{x}, \boldsymbol{u}) + \gamma V\big(f(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{\xi}_\theta)\big)$$
$$\text{with } \frac{1}{2}\left((\boldsymbol{\xi}_\theta - \boldsymbol{\mu}_\nu)^2 - \boldsymbol{\Delta}_\nu^2\right) \leq \boldsymbol{0}.$$

Substituting the Taylor expansion for $V(\boldsymbol{x}_{t+1})$, the dynamics models and reward as well as abbreviating $\boldsymbol{B}(\boldsymbol{x}; \theta + \boldsymbol{\xi}_\theta)$ as $\boldsymbol{B}_\xi$ and $\boldsymbol{a}(\boldsymbol{x}; \theta + \boldsymbol{\xi}_\theta)$ as $\boldsymbol{a}_\xi$ yields

$$V_{\text{tar}} = \max_{\boldsymbol{u}} \; \min_{\boldsymbol{\xi}_\theta} \; r + \gamma V + \gamma \nabla_x V^T f_c \Delta t + \gamma O(.)\Delta t$$
$$\frac{V_{\text{tar}} - \gamma V}{\Delta t} = q_c + \max_{\boldsymbol{u}} \min_{\boldsymbol{\xi}} \left[\gamma \nabla_x V^T \left(\boldsymbol{a}_\xi + \boldsymbol{B}_\xi \boldsymbol{u}\right) + \gamma O(.) - g_c\right]$$

In the continuous time limit the optimal action and disturbance is determined by

$$\boldsymbol{u}^*, \boldsymbol{\xi}_\theta^* = \max_{\boldsymbol{u}} \min_{\boldsymbol{\xi}} \left[\nabla_x V^T \left(\boldsymbol{a}_\xi + \boldsymbol{B}_\xi \boldsymbol{u}\right) - g_c(\boldsymbol{u})\right].$$

This nested max-min optimization can be solved by first solving the inner optimization w.r.t. to $\boldsymbol{u}$ and substituting this solution into the outer maximization. The Lagrangian for the optimal model disturbance is described by

$$\boldsymbol{\xi}^* = \arg\min_{\boldsymbol{\xi}} \; \nabla_x V^T \left(\boldsymbol{a}_\xi + \boldsymbol{B}_\xi \boldsymbol{u}\right) + \frac{1}{2}\boldsymbol{\lambda}^T \left((\boldsymbol{\xi}_\theta - \boldsymbol{\mu}_\nu)^2 - \boldsymbol{\Delta}_\nu^2\right)$$

Using the KKT conditions this optimization can be solved. The stationarity condition yields

$$\boldsymbol{z}_\theta + \boldsymbol{\lambda}^T(\boldsymbol{\xi}_\theta - \boldsymbol{\mu}_\nu) = 0 \qquad \Rightarrow \qquad \boldsymbol{\xi}_\theta^* = -\boldsymbol{z}_\theta \oslash \boldsymbol{\lambda} + \boldsymbol{\mu}_\nu$$
$$\text{with } \; \boldsymbol{z}_\theta = \left[\frac{\partial \boldsymbol{a}}{\partial \theta} + \frac{\partial \boldsymbol{B}}{\partial \theta}\boldsymbol{u}\right]^T \nabla_x V$$

and the elementwise division $\oslash$. The primal feasibility and the complementary slackness yields

$$\frac{1}{2}\left(-\boldsymbol{z}_\theta^2 \oslash \boldsymbol{\lambda}^2 - \boldsymbol{\Delta}_\nu^2\right) \leq 0 \;\; \Rightarrow \;\; \boldsymbol{\lambda} \geq \|\boldsymbol{z}_\theta\|_1 \oslash \boldsymbol{\Delta}_\nu$$
$$\frac{1}{2}\boldsymbol{\lambda}^T\left(-\boldsymbol{z}_\theta^2 \oslash \boldsymbol{\lambda}^2 - \boldsymbol{\Delta}_\nu^2\right) = 0 \;\; \Rightarrow \;\; \boldsymbol{\lambda}_0 = \boldsymbol{0}, \;\; \boldsymbol{\lambda}_1 = \|\boldsymbol{z}_\theta\|_1 \oslash \boldsymbol{\Delta}_\nu.$$

Therefore, the optimal model disturbance is described by

$$\boldsymbol{\xi}_\theta^*(\boldsymbol{u}) = -\boldsymbol{\Delta}_\nu \operatorname{sign}\left(\boldsymbol{z}_\theta(\boldsymbol{u})\right) + \boldsymbol{\mu}_\nu$$

as $\boldsymbol{z}_\theta \oslash \|\boldsymbol{z}_\theta\|_1 = \operatorname{sign}(\boldsymbol{z}_\theta)$. Then the optimal action can be computed by

$$\boldsymbol{u}^* = \arg\max_{\boldsymbol{u}} \nabla_x V^T \left[\boldsymbol{a}(\boldsymbol{\xi}_\theta^*(\boldsymbol{u})) + \boldsymbol{B}\left(\boldsymbol{\xi}_\theta^*(\boldsymbol{u})\right)\boldsymbol{u}\right] - g_c(\boldsymbol{u}).$$

Due to the envelope theorem, the extrema is described by

$$\boldsymbol{B}(\boldsymbol{x}; \theta + \boldsymbol{\xi}^*(\boldsymbol{u}))^T \nabla_x V - g_c(\boldsymbol{u}) = 0.$$

This expression cannot be solved without approximation as $\boldsymbol{B}$ does not necessarily be invertible w.r.t. θ. Approximating $\boldsymbol{B}(\boldsymbol{x}; \theta + \boldsymbol{\xi}^*(\boldsymbol{u})) \approx \boldsymbol{B}(\boldsymbol{x}; \theta)$, lets one solve for $\boldsymbol{u}$. In this case the optimal action $\boldsymbol{u}^*$ is described by $\boldsymbol{u}^* = \nabla \tilde{g}(\boldsymbol{B}(\boldsymbol{x}; \theta)^T \nabla_x V)$. This approximation is feasible for two reasons. First of all, if the adversary can significantly alter the dynamics in each step, the system would not be controllable and the optimal policy would not be able to solve the task. Second, this approximation implies that neither agent or the adversary can react to the action of the other and must choose simultaneously. This assumption is common in prior works [5]. The order of the minimization and maximization is interchangeable. For both cases the optimal action as well as optimal model disturbance are identical and require the same approximation during the derivation. □

D. Observation Disturbance Proof

Theorem. *For the adversarial observation disturbance (Equation 7) with bounded signal energy (Equation 16), smooth drift and control matrix (i.e., $\boldsymbol{a}, \boldsymbol{B} \in C^1$) and $\boldsymbol{B}(\boldsymbol{x} + \boldsymbol{\xi}_o) \approx \boldsymbol{B}(\boldsymbol{x})$, the optimal continuous time policy π^k and observation disturbance $\boldsymbol{\xi}_o^k$ is described by*

$$\pi^k(\boldsymbol{x}) = \nabla \tilde{g}\left(\boldsymbol{B}(\boldsymbol{x})^T \nabla_x V^k\right) \qquad \boldsymbol{\xi}_o^k = -\alpha \frac{\boldsymbol{z}_o}{\|\boldsymbol{z}_o\|_2}$$
$$\textit{with } \; \boldsymbol{z}_o = \left(\frac{\partial \boldsymbol{a}(\boldsymbol{x};\, \theta)}{\partial \boldsymbol{x}} + \frac{\partial \boldsymbol{B}(\boldsymbol{x};\, \theta)}{\partial \boldsymbol{x}}\boldsymbol{u}^*\right)^T V_x^*.$$

Proof. Equation 14 can be written with the explicit constraint instead of the infimum with the admissible set Ω_A, i.e.,

$$V_{\text{tar}} = \max_{\boldsymbol{u}} \min_{\boldsymbol{\xi}_o} r(\boldsymbol{x}, \boldsymbol{u}) + \gamma V\big(f(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{\xi}_o)\big)$$
$$\text{with } \frac{1}{2}\left((\boldsymbol{\xi}_o^T \boldsymbol{\xi}_o - \alpha^2\right) \leq \boldsymbol{0}.$$

Substituting the Taylor expansion for $V(\boldsymbol{x}_{t+1})$, the dynamics models and reward as well as abbreviating $\boldsymbol{B}(\boldsymbol{x} + \boldsymbol{\xi}_o; \theta)$ as $\boldsymbol{B}_\xi$ yields

$$V_{\text{tar}} = \max_{\boldsymbol{u}} \; \min_{\boldsymbol{\xi}_o} \; r + \gamma V + \gamma \nabla_x V^T f_c \Delta t + \gamma O(.)\Delta t$$
$$\frac{V_{\text{tar}} - \gamma V}{\Delta t} = q_c + \max_{\boldsymbol{u}} \min_{\boldsymbol{\xi}} \left[\gamma \nabla_x V^T \left(\boldsymbol{a}_\xi + \boldsymbol{B}_\xi \boldsymbol{u}\right) + \gamma O(.) - g_c\right]$$

In the continuous time limit the optimal action and disturbance is determined by

$$\boldsymbol{u}^*, \boldsymbol{\xi}_o^* = \max_{\boldsymbol{u}} \min_{\boldsymbol{\xi}} \left[\nabla_x V^T \left(\boldsymbol{a}_\xi + \boldsymbol{B}_\xi \boldsymbol{u}\right) - g_c(\boldsymbol{u})\right].$$

This nested max-min optimization can be solved by first solving the inner optimization w.r.t. to ξ and substituting this

Table III
Average rewards on the simulated and physical systems. The ranking describes the decrease in reward compared to the best result averaged on all systems. The initial state distribution during training is noted by μ. The dynamics are either deterministic model $\theta \sim \delta(\theta)$ or sampled using uniform domain randomization $\theta \sim \mathcal{U}(\theta)$. During evaluation the roll outs start with the pendulum pointing downwards.

			Simulated Pendulum		**Simulated Cartpole**		**Simulated Furuta Pendulum**		**Physical Cartpole**		**Physical Furuta Pendulum**		**Average**
Algorithm	μ	θ	Success [%]	Reward [$\mu \pm 2\sigma$]	Success [%]	Reward [$\mu \pm 2\sigma$]	Success [%]	Reward [$\mu \pm 2\sigma$]	Success [%]	Reward [$\mu \pm 2\sigma$]	Success [%]	Reward [$\mu \pm 2\sigma$]	**Ranking** [%]
DP rFVI (ours)	–	$\delta(\theta)$	100.0	−032.7 ± 000.3	100.0	−027.1 ± 004.8	100.0	−041.3 ± 010.8	100.0	**−074.1 ± 040.3**	100.0	−278.0 ± 034.3	−062.7
DP cFVI	–	$\delta(\theta)$	100.0	**−030.5 ± 000.8**	100.0	**−024.2 ± 002.1**	100.0	**−027.7 ± 001.6**	73.3	−143.7 ± 210.4	100.0	**−082.1 ± 007.6**	−019.2
RTDP cFVI (ours)	$\mathcal{U}$	$\delta(\theta)$	100.0	**−031.1 ± 001.4**	100.0	**−024.9 ± 001.6**	100.0	−040.1 ± 002.7	100.0	−101.1 ± 029.0	00.0	−1009.9 ± 004.5	−247.7
SAC	$\mathcal{N}$	$\mathcal{U}(\theta)$	100.0	**−031.1 ± 000.1**	100.0	−026.9 ± 003.2	100.0	−029.3 ± 001.5	00.0	−518.6 ± 028.1	86.7	−330.7 ± 799.0	−185.8
SAC & UDR	$\mathcal{N}$	$\delta(\theta))$	100.0	−032.9 ± 000.6	100.0	−029.7 ± 004.6	100.0	−032.0 ± 001.1	100.0	−394.8 ± 382.8	100.0	−181.4 ± 157.9	−120.8
SAC	$\mathcal{U}$	$\mathcal{U}(\theta)$	100.0	**−030.6 ± 001.4**	100.0	**−024.2 ± 001.4**	100.0	**−028.1 ± 002.0**	53.3	−144.5 ± 204.0	100.0	−350.8 ± 433.3	−086.5
SAC & UDR	$\mathcal{U}$	$\mathcal{U}(\theta)$	100.0	−031.4 ± 002.5	100.0	**−024.2 ± 001.3**	100.0	**−028.1 ± 001.3**	40.0	−296.4 ± 418.9	100.0	−092.3 ± 064.1	−063.8
DDPG	$\mathcal{N}$	$\mathcal{U}(\theta)$	100.0	**−031.1 ± 000.4**	98.0	−050.4 ± 285.6	100.0	−030.5 ± 003.5	06.7	−536.7 ± 262.7	46.7	−614.1 ± 597.8	−281.4
DDPG & UDR	$\mathcal{N}$	$\delta(\theta))$	100.0	−032.5 ± 000.5	100.0	−027.4 ± 002.3	100.0	−034.6 ± 009.8	00.0	−517.9 ± 117.6	86.7	−192.7 ± 404.8	−156.6
DDPG	$\mathcal{U}$	$\mathcal{U}(\theta)$	100.0	−031.5 ± 000.7	100.0	−028.2 ± 005.5	100.0	−030.0 ± 001.7	06.7	−459.4 ± 248.3	100.0	−146.6 ± 218.3	−126.0
DDPG & UDR	$\mathcal{U}$	$\mathcal{U}(\theta)$	100.0	−032.5 ± 003.6	100.0	−027.2 ± 001.0	100.0	−032.1 ± 001.5	00.0	−318.1 ± 063.4	100.0	−156.7 ± 246.4	−091.7
PPO	$\mathcal{N}$	$\mathcal{U}(\theta)$	100.0	−032.0 ± 000.2	100.0	−031.5 ± 007.2	100.0	−081.1 ± 018.3	00.0	−287.9 ± 068.8	33.3	−718.7 ± 456.1	−261.7
PPO & UDR	$\mathcal{N}$	$\delta(\theta))$	100.0	−032.3 ± 000.6	100.0	−084.0 ± 007.8	100.0	−040.9 ± 004.6	00.0	−435.4 ± 111.9	46.7	−935.7 ± 711.6	−370.0
PPO	$\mathcal{U}$	$\mathcal{U}(\theta)$	100.0	−033.4 ± 004.7	99.0	−039.7 ± 045.7	100.0	−038.2 ± 013.1	00.0	−183.8 ± 018.0	60.0	−755.3 ± 811.0	−219.4
PPO & UDR	$\mathcal{U}$	$\mathcal{U}(\theta)$	100.0	−035.6 ± 003.1	100.0	−044.8 ± 021.4	100.0	−048.5 ± 006.2	40.0	−143.8 ± 016.1	100.0	**−080.6 ± 010.8**	−054.4

solution into the outer maximization. The Lagrangian for the optimal model disturbance is described by

$$\boldsymbol{\xi}_o^* = \arg\min_{\boldsymbol{\xi}_o} \; \nabla_x V^T \left(\boldsymbol{a}(\boldsymbol{\xi}_o) + \boldsymbol{B}(\boldsymbol{\xi}_o)\boldsymbol{u}\right) + \frac{\lambda}{2}\left(\boldsymbol{\xi}_o^T \boldsymbol{\xi}_o - \alpha^2\right)$$

Using the KKT conditions this optimization can be solved. The stationarity condition yields

$$\boldsymbol{z}_o + \lambda\, \boldsymbol{\xi}_o = 0 \qquad \Rightarrow \qquad \boldsymbol{\xi}_o^* = -\frac{1}{\lambda}\boldsymbol{z}_o$$

$$\text{with } \; \boldsymbol{z}_o = \left[\frac{\partial \boldsymbol{a}(\boldsymbol{x};\, \theta)}{\partial \boldsymbol{x}} + \frac{\partial \boldsymbol{B}(\boldsymbol{x};\, \theta)}{\partial \boldsymbol{x}}\boldsymbol{u}^*\right]^T \nabla_x V.$$

The primal feasibility and the complementary slackness yield

$$\frac{1}{2}\left(\frac{1}{\lambda^2}\boldsymbol{z}_o^T \boldsymbol{z}_o - \alpha^2\right) \leq 0 \quad \Rightarrow \quad \lambda \geq \frac{1}{\alpha}\|\boldsymbol{z}_\theta\|_2$$

$$\frac{\lambda}{2}\left(\frac{1}{\lambda^2}\boldsymbol{z}_o^T \boldsymbol{z}_o - \alpha^2\right) = 0 \quad \Rightarrow \quad \lambda_0 = 0, \;\; \lambda_1 = \frac{1}{\alpha}\|\boldsymbol{z}_o\|_2.$$

Therefore, the optimal observation disturbance is described by

$$\boldsymbol{\xi}_o^*(\boldsymbol{u}) = -\alpha\frac{\boldsymbol{z}_o}{\|\boldsymbol{z}_o\|_2}.$$

Then the optimal action can be computed by

$$\boldsymbol{u}^* = \arg\max_{\boldsymbol{u}} \nabla_x V^T \left[\boldsymbol{a}(\boldsymbol{\xi}_o^*(\boldsymbol{u})) + \boldsymbol{B}\left(\boldsymbol{\xi}_o^*(\boldsymbol{u})\right)\boldsymbol{u}\right] - g_c(\boldsymbol{u}).$$

Due to the envelope theorem, the extrema is described by

$$\boldsymbol{B}(\boldsymbol{x}; \theta + \boldsymbol{\xi}_o^*(\boldsymbol{u}))^T \nabla_x V - g_c(\boldsymbol{u}) = 0.$$

This expression cannot be solved without approximation as $\boldsymbol{B}$ does not necessarily be invertible w.r.t. $\boldsymbol{x}$. Approximating $\boldsymbol{B}(\boldsymbol{x} + \boldsymbol{\xi}_o^*(\boldsymbol{u});\, \theta) \approx \boldsymbol{B}(\boldsymbol{x}; \theta)$, lets one solve for $\boldsymbol{u}$. In this case the optimal action $\boldsymbol{u}^*$ is described by $\boldsymbol{u}^* = \nabla\tilde{g}(\boldsymbol{B}(\boldsymbol{x};\theta)^T \nabla_x V)$. This approximation is feasible for two reasons. First of all, if the adversary can significantly alter the dynamics in each step, the system would not be controllable and the optimal policy would not be able to solve the task. Second, this approximation implies that neither agent or the adversary can react to the action of the other and must choose simultaneously. This assumption is common in prior works [5]. The order of the minimization and maximization is interchangeable. For both cases the optimal action as well as optimal model disturbance are identical and require the same approximation during the derivation. □

Value Function Representation

For the value function we are using a locally quadratic deep network as described in [27]. This architecture assumes that the state cost is a negative distance measure between $\boldsymbol{x}_t$ and the desired state $\boldsymbol{x}_{\text{des}}$. Hence, q_c is negative definite, i.e., $q(\boldsymbol{x}) < 0 \;\forall\; \boldsymbol{x} \neq \boldsymbol{x}_{\text{des}}$ and $q(\boldsymbol{x}_{\text{des}}) = 0$. These properties imply that V^* is a negative Lyapunov function, as V^* is negative definite, $V^*(\boldsymbol{x}_{\text{des}}) = 0$ and $\nabla_x V^*(\boldsymbol{x}_{\text{des}}) = \mathbf{0}$ [61]. With a deep network a similar representation can be achieved by

$$V(\boldsymbol{x};\, \psi) = -\left(\boldsymbol{x} - \boldsymbol{x}_{\text{des}}\right)^T \boldsymbol{L}(\boldsymbol{x};\, \psi)\boldsymbol{L}(\boldsymbol{x};\, \psi)^T \left(\boldsymbol{x} - \boldsymbol{x}_{\text{des}}\right)$$

with $\boldsymbol{L}$ being a lower triangular matrix with positive diagonal. This positive diagonal ensures that $\boldsymbol{L}\boldsymbol{L}^T$ is positive definite. Simply applying a ReLu activation to the last layer of a deep network is not sufficient as this would also zero the actions for the positive values and $\nabla_x V^*(\boldsymbol{x}_{\text{des}}) = \mathbf{0}$ cannot be guaranteed. The local quadratic representation guarantees that the gradient and hence, the action, is zero at the desired state. However, this representation can also not guarantee that the value function has only a single extrema at $\boldsymbol{x}_{\text{des}}$ as required by the Lyapunov theory. In practice, the local regularization of the quadratic structure to avoid high curvature approximations is sufficient as the global structure is defined by the value function target. $\boldsymbol{L}$ is the mean of a deep network ensemble with N independent parameters ψ_i. The ensemble mean smoothes the initial value function and is differentiable. Similar representations have been used by prior works in the safe reinforcement learning community [62–65].

VII. Detailed Experimental Setup

Systems The performance of the algorithms is evaluated using the *swing-up* the torque-limited pendulum, cartpole and Furuta pendulum. The physical cartpole (Figure 4) and Furuta pendulum (Figure 5) are manufactured by Quanser [37]. For simulation, we use the equations of motion and physical parameters of the supplier. Both systems have very different

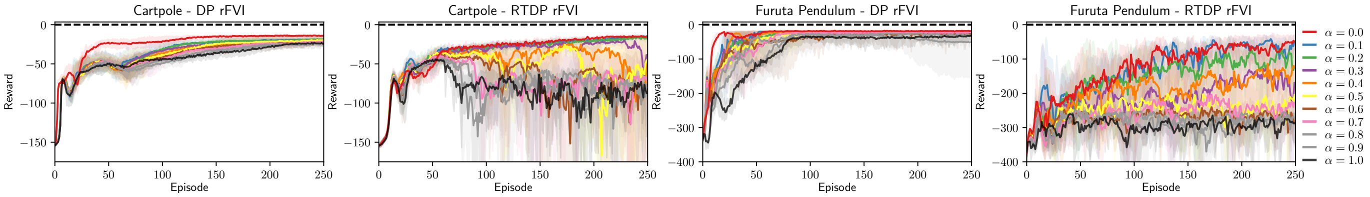

Figure 8. The learning curves for DP rFVI and RTDP rFVI with different adversary amplitudes averaged over 5 seeds. The shaded area displays the *min/max* range between seeds. The α corresponds of the percentage of the admissible set for all adversaries, i.e., with increasing α the adversary becomes more powerful. For DP rFVI the stronger adversaries do affect the final performance only marginally. For RTDP rFVI the adversaries become too powerful for small α and prevent learning of the optimal policy. This effect is especially distinct for the Furuta pendulum as this system is very sensible due to the low masses. Therefore, DP rFVI can learn a good optimal policy despite very strong adversaries.

characteristics. The Furuta pendulum consists of a small and light pendulum (24g, 12.9cm) with a strong direct-drive motor. Even minor differences in the action cause large changes in acceleration due to the large amplification of the mass-matrix inverse. Therefore, the main source of uncertainty for this system is the uncertainty of the model parameters. The cartpole has a longer and heavier pendulum (127g, 33.6cm). The cart is actuated by a geared cogwheel drive. Due to the larger masses the cartpole is not so sensitive to the model parameters. The main source of uncertainty for this system is the friction and the backlash of the linear actuator. The systems are simulated and observed with 500Hz. The control frequency varies between algorithm and is treated as hyperparameter.

Reward Function The desired state for all tasks is the upward pointing pendulum at $\boldsymbol{x}_{\text{des}} = \mathbf{0}$. The state reward is described by $q_c(\boldsymbol{x}) = -(\boldsymbol{z} - \boldsymbol{z}_{\text{des}})^T \boldsymbol{Q} (\boldsymbol{z} - \boldsymbol{z}_{\text{des}})$ with the positive definite matrix $\boldsymbol{Q}$ and the transformed state $\boldsymbol{z}$. For continuous joints the joint state is transformed to $z_i = \pi^2 \sin(x_i)$. The action cost is described by $g_c(\boldsymbol{u}) = -2\,\boldsymbol{\beta}\,\boldsymbol{u}_{\max}/\pi \log\cos(\pi\,\boldsymbol{u}/(2\,\boldsymbol{u}_{\max}))$ with the actuation limit $\boldsymbol{u}_{\max}$ and the positive constant $\boldsymbol{\beta}$. This barrier shaped cost bounds the optimal actions. The corresponding policy is shaped by $\nabla\tilde{g}(\boldsymbol{w}) = 2\,\boldsymbol{u}_{\max}/\pi\ \tan^{-1}(\boldsymbol{w}/\boldsymbol{\beta})$. For the experiments, the reward parameters are

$$\begin{aligned}
\text{Pendulum:}\quad & \boldsymbol{Q}_{\text{diag}} = [\ \ 1.0,\ 0.1]\,, && \beta = 0.5\\
\text{Cartpole:}\quad & \boldsymbol{Q}_{\text{diag}} = [25.0,\ 1.0,\ 0.5,\ 0.1]\,, && \beta = 0.1\\
\text{Furuta Pendulum:}\quad & \boldsymbol{Q}_{\text{diag}} = [\ \ 1.0,\ 5.0,\ 0.1,\ 0.1]\,, && \beta = 0.1
\end{aligned}$$

Additional Experimental Results

This section summarizes the additional experiments omitted in the main paper. In the following we perform an ablation study varying the admissible set and report the performance of additional baselines on the nominal physical system.

Ablation Study - Admissible Set

The ablation study highlighting the differences in learning curves for different admissible sets is shown in Figure 8. In this plot we vary the admissible set of each adversary from 0 to the 1, where $\alpha = 1$ corresponds to the admissible set used for the robustness experiments. With increasing admissible set the final performance of the adversary decreases marginally for DP rFVI. For RTDP rFVI the optimal policy does not learn the task for stronger adversaries. RTDP rFVI starts to fail for $\alpha > 0.3$ on the cartpole. On the Furuta pendulum, RTDP rFVI starts to fail at $\alpha > 0.1$. The Furuta pendulum starts to fail earlier as this system is much more sensitive and smaller actions cause large changes in dynamics compared to the cartpole. This ablation study shows that the dynamic programming variant can learn the optimal policy despite strong adversaries. In contrast the real-time dynamic programming variant fails to learn the optimal policy for comparable admissible set. This failure is caused by the missing positive feedback during exploration. The adversary prevents the policy from discovering the positive reward during exploration. Therefore, the policy converges to a too pessimistic policy. In contrast the dynamic programming variant does not rely on exploration and covers the compact state domain. Therefore, the optimal policy discovers the optimal policy despite the strong adversary.

A. Physical Experiments

The rewards of all baselines on the nominal physical system are reported in Table III. The robustness experiments were only performed for the best performing baselines. On the nominal system rFVI outperforms all baselines on the cartpole. The domain randomization baselines do not necessarily outperform the deterministic deep RL baselines as the main source of simulation gap is the backslash and stiction of the actuator which is not randomized. For the Furuta pendulum, DP rFVI has a lower reward compared to DP cFVI. However, this lower reward is only caused by the chattering during the balancing that causes very high actions costs. If one only considers the swing-up phase, DP rFVI outperforms both PPO-U UDR and DP cFVI. For the Furuta pendulum the deep RL & UDR baselines outperform the deep RL baselines without UDR. This is expected as the main uncertainty for the Furuta pendulum is caused by the uncertainty of the system parameters. In general a trend is observable that the algorithms with larger state domain during training achieve the better sim2real transfer.

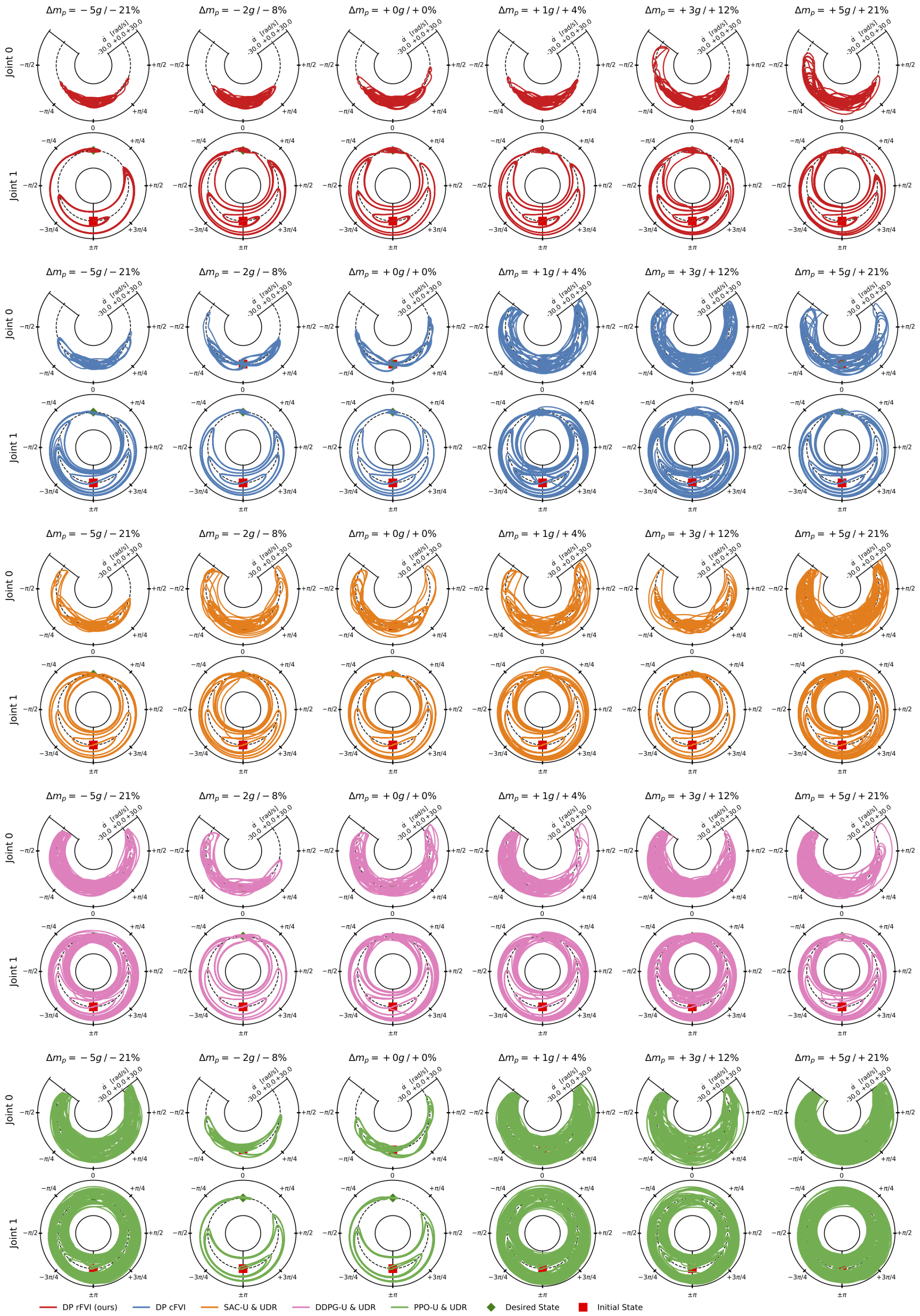

Figure 9. The roll-outs of DP rFVI, DP cFVI and the deep RL baselines with domain randomization the physical Furuta pendulum. The different columns correspond to different pendulum masses. The deviation from the dashed center line corresponds to the joint velocity. DP rFVI achieves a consistent swing-up for the different pendulum masses. In contrast to DP rFVI, the baselines start to deviate strongly from trajectories on the nominal system. When weights are added the baselines start to cover the complete state-space.

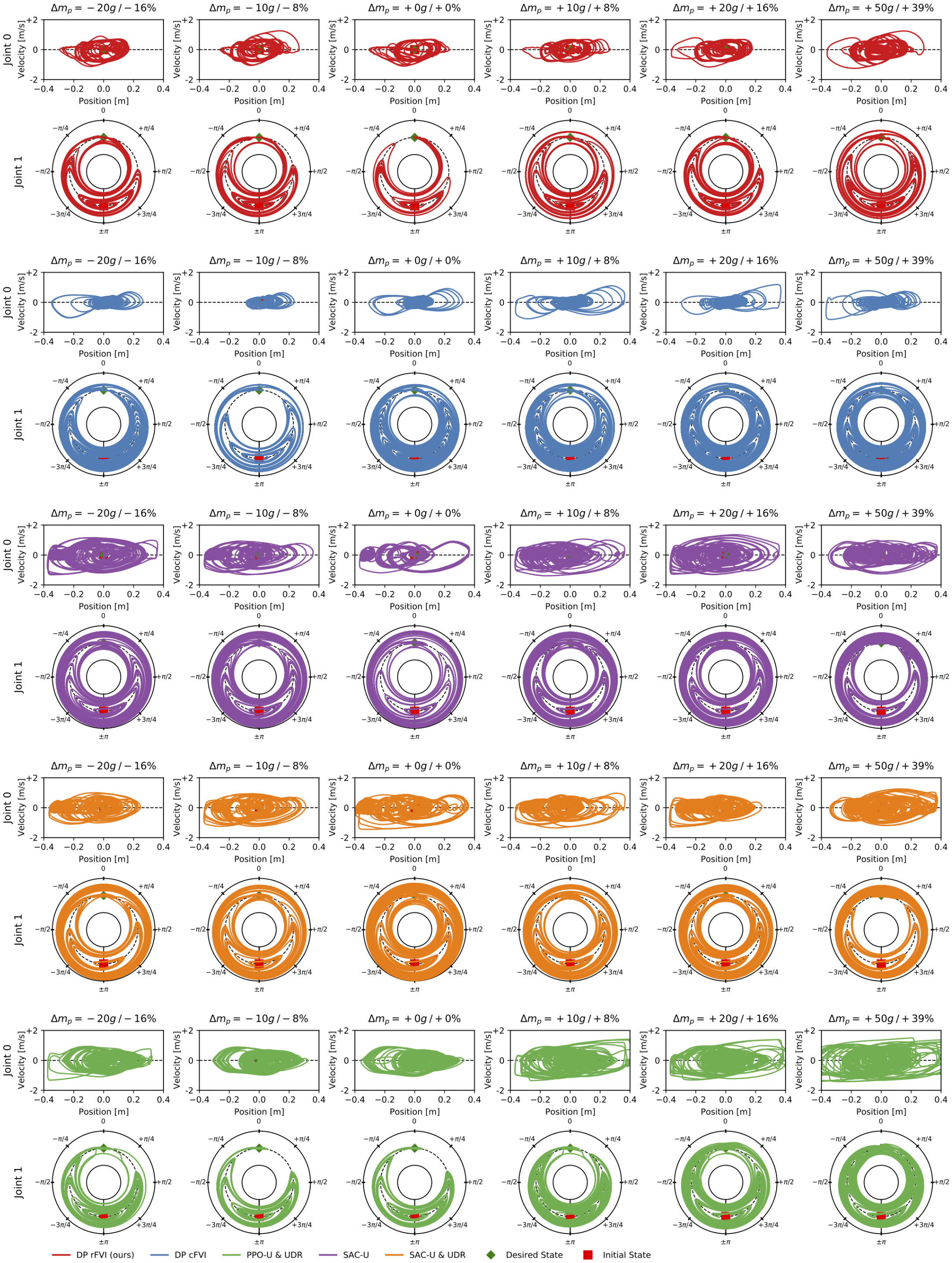

Figure 10. The roll-outs of DP rFVI, DP cFVI and the deep RL baselines with domain randomization the physical Cartpole. The different columns correspond to different pendulum masses. The deviation from the dashed center line corresponds to the joint velocity. DP rFVI achieves a consistent swing-up for the different pendulum masses but the failure rate slighlty increases when weights are added to the pendulum. In contrast to DP rFVI, the baselines start to deviate stronger from trajectories on the nominal system when the system dynamics are altered.