8. Appendix

Proof of Theorem 1

Theorem. *If the dynamics are control affine (Equation 5), the reward is separable w.r.t. to state and action (Equation 6) and the action cost g_c is positive definite and strictly convex, the continuous time optimal policy π^k w.r.t. V^k is described by*

$$\pi^k(\boldsymbol{x}) = \nabla \tilde{g}_c \left(\boldsymbol{B}(\boldsymbol{x})^T \nabla_x V^k \right) \tag{12}$$

with the convex conjugate $\tilde{g}$ of g and the Jacobian of V w.r.t. the system state $\nabla_x V$.

Proof. This proof follows the derivation of Lutter et. al. (2019). This prior work derived the optimal policy π^* using the Hamilton Jacobi Bellman differential equation (HJB) and generalized the special case described by Doya (2000). The value iteration update (Equation 8) is defined as

$$V_{\text{tar}}(\boldsymbol{x}_t) = \max_{\boldsymbol{u}} \; r(\boldsymbol{x}_t, \boldsymbol{u}) + \gamma V(f(\boldsymbol{x}_t, \boldsymbol{u});\, \psi_k).$$

Substituting the assumptions and using the Taylor expansion of the Value function, i.e.,

$$V(f(\boldsymbol{x}_t, \boldsymbol{u})) = V(\boldsymbol{x}_t) + \nabla_x V^T f_c(\boldsymbol{x}_t, \boldsymbol{u})\, \Delta t + O(\Delta t, \boldsymbol{x}_t, \boldsymbol{u})\, \Delta t,$$

this update can be rewritten - omitting all functional dependencies for brevity - as

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

with the higher order Terms $O(\Delta t, \boldsymbol{x}_t, \boldsymbol{u})$. Therefore, the optimal action is defined as

$$\boldsymbol{u}_t^* = \arg\max_{\boldsymbol{u}} \gamma \nabla_x V^T \; (\boldsymbol{a} + \boldsymbol{B}\boldsymbol{u}) + \gamma \; O(\Delta t, \boldsymbol{x}_t, \boldsymbol{u}) - g_c(\boldsymbol{u}).$$

In the continuous time limit, the higher order terms $O(\Delta t, \boldsymbol{x}_t, \boldsymbol{u})$ disappear as these depend on Δt, i.e., i.e., $\lim_{\Delta t \to 0} O(\Delta t, \boldsymbol{x}_t, \boldsymbol{u}) = 0$. The action is also independent of the discounting as $\lim_{\Delta t \to 0} \gamma = 1$. Therefore, the continuous time optimal action is defined as

$$\boldsymbol{u}_t^* = \arg\max_{\boldsymbol{u}} \; \nabla_x V^T \boldsymbol{B}(\boldsymbol{x}) \boldsymbol{u}_t - g_c(\boldsymbol{u}).$$

This optimization can be solved analytically as g_c is strictly convex and hence $\nabla g_c(\boldsymbol{u}) = \boldsymbol{w}$ is invertible, i.e., $\boldsymbol{u} = [\nabla g_c]^{-1}(\boldsymbol{w}) \coloneqq \nabla \tilde{g}_c(\boldsymbol{w})$ with the convex conjugate $\tilde{g}$. The optimal action is described by

$$\boldsymbol{B}(\boldsymbol{x})^T \nabla_x V^* - \nabla g_c(\boldsymbol{u}) \coloneqq 0 \;\; \Rightarrow \;\; \boldsymbol{u}^* = \nabla \tilde{g}_c \left(\boldsymbol{B}(\boldsymbol{x})^T \nabla_x V^k \right).$$

Therefore, the value function update can be rewritten by substituting the optimal action, i.e.,

$$\begin{aligned} V_{\text{tar}}(\boldsymbol{x}_t) &= r\left(\boldsymbol{x}_t, \nabla \tilde{g}\left(\boldsymbol{B}(\boldsymbol{x}_t)^T \nabla_x V^k\right)\right) + \gamma V^k(\boldsymbol{x}_{t+1};\, \psi_k) \\ \text{with} \quad \boldsymbol{x}_{t+1} &= f\left(\boldsymbol{x}_t, \nabla \tilde{g}(\boldsymbol{B}(\boldsymbol{x}_t)^T \nabla_x V^k)\right). \end{aligned}$$

□

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 3) are manufactured by Quanser (2018). For simulation, we use the equations of motion and physical parameters of the supplier. Both systems have very different 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.

Baselines The control performance is compared to Deep Deterministic Policy Gradients (DDPG) (Lillicrap et al., 2015), Soft Actor Critic (SAC) (Haarnoja et al., 2018) and Proximal Policy Optimization (PPO) (Schulman et al., 2017). The baselines are augmented with uniform domain randomization (UDR) (Muratore et al., 2018). For the experiments the open-source implementations of MushroomRL (D'Eramo et al., 2020) are used. We compare two different initial state distributions (μ). First, the initial pendulum angle θ_0 is sampled uniformly, i.e. $\theta_0 \sim \mathcal{U}(-\pi, +\pi)$. Second, the initial angle is sampled from a Gaussian distribution with the pendulum facing downwards, i.e., $\theta_0 \sim \mathcal{N}(\pm\pi, \sigma)$. The uniform sampling avoids the exploration problem and generates a larger state distribution of the optimal policy.

Reward Function The desired state for all tasks is the upward pointing pendulum at $\boldsymbol{x}_{\text{des}} = \boldsymbol{0}$. 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

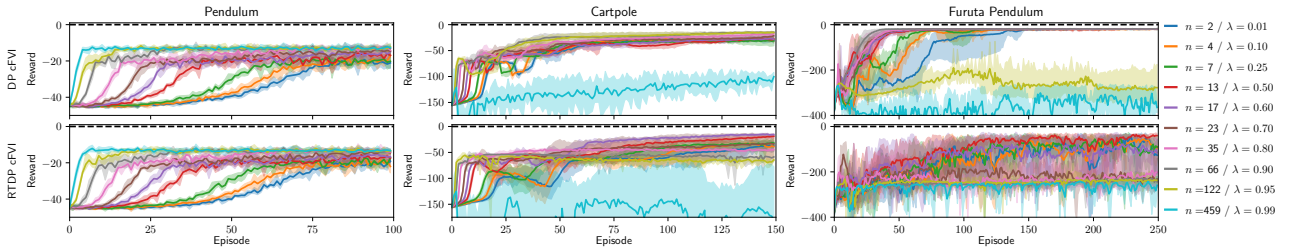

Figure 8. The learning curves averaged over 5 seeds for the n-step value function target. The shaded area displays the *min/max* range between seeds. The step count is selected such that $\lambda^n := 10^{-4}$. Increasing the horizon of the value function target increases the convergence rate to the optimal value function. For very long horizons the learning diverges as it over fits to the current value function approximation. Furthermore, the performance of the optimal policy also increases with roll out length.

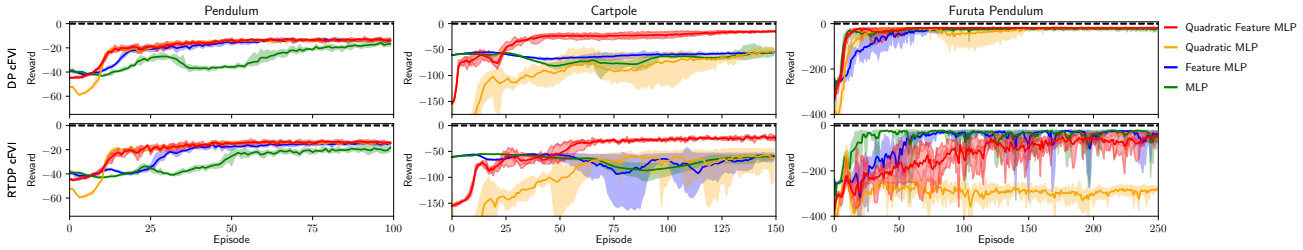

Figure 9. The learning curves averaged over 5 seeds for the different model architectures. The shaded area displays the *min/max* range between seeds. All network architectures are capable of learning the value function and policy for most of the tasks. The locally quadratic network architecture increases learning speed compared to the baselines. The structured architecture acts as an inductive bias that shapes the exploration. The global maximum of the locally quadratic value function is guaranteed at $\boldsymbol{x}_{\text{des}}$ and hence the initial policy performs hill-climbing towards this point.

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

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

Evaluation The rewards are evaluated using 100 roll outs in simulation and 15 roll outs on the physical system. If not noted otherwise, each roll out lasts $15s$ and starts with the pendulum downward . This duration is much longer than the required time to swing up. The pendulum is considered balancing, if the pendulum angle is below $\pm 5°$ degree for every sample of the last second.

Extended Experimental Results

Ablation Study - N-step Value Targets

The learning curves for the ablation study highlighting the importance are shown in Figure 8. This figure contains in contrast to Figure 7 also RTDP cFVI. When increasing λ, which implicitly increases the n-step horizon (Section 3.3), the convergence rate to the optimal value function increases. This increased learning speed is expected as Equation 7 shows that the convergence rate depends on γ^n with $\gamma < 1$.

While the learning speed measured in iterations increases with λ, the computational complexity also increases. Longer horizons require to simulate n sequential steps increasing the required wall-clock time. Therefore, the computation time increases exponentially with increasing λ. For example the forward roll out in every iteration of the pendulum increases exponentially from 0.4s ($\lambda = 0.01$) to 56.4s ($\lambda = 0.99$). For the Furuta pendulum and the cartpole extremely long horizons of 100+ steps start to diverge as the value function target over fits to the untrained value function. For RTDP cFVI, the horizons must smaller, i.e., 10 - 20 steps. For longer horizons the predicted rollout overfits to the value function outside of the current state distribution, which prevents learning or leads to pre-mature convergence. This is very surprising as even for the true model long time-horizons can be counterproductive due to the local approximation of the value function. Therefore, DP cFVI works bests with $\lambda \in [0.85, 0.95]$ and RTDP cFVI with $\lambda \in [0.45 - 0.55]$.

Ablation Study - Model Architecture

To evaluate the impact of the locally quadratic architecture described by

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

where $\boldsymbol{L}$ is a lower triangular matrix with positive diagonal, we compare this architecture to a standard multi-layer perceptron with and without feature transformation. The learning

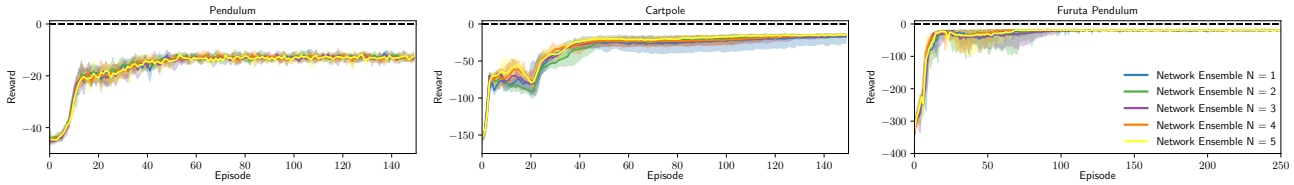

Figure 10. The learning curves averaged over 5 seeds with different model ensemble sizes N. The shaded area displays the *min/max* range between seeds. The performance of the optimal policy is not significantly affected by the model ensemble. For the cartpole and especially the Furuta pendulum, the larger model ensembles stabilize the training and achieve faster learning and exhibit less variations between seeds.

curves for the ablation study highlighting the importance of the network architecture are shown in Figure 9. The reward curves are averaged over 5 seeds and visualize the maximum range between seeds. For most systems all network architecture are able to learn a good policy. The locally quadratic value function is on average the best performing architecture. The structure acts as a inductive bias that shapes the exploration and leads to faster learning. The global maximum is guaranteed to be at $\boldsymbol{x}_{\text{des}}$ as $\boldsymbol{L}(\boldsymbol{x};\,\psi)\boldsymbol{L}(\boldsymbol{x};\,\psi)^T$ is positive definite. Therefore, the initial policy directly performs hill-climbing towards the balancing position. Only for the cartpole the other network architectures fail. For this system, these architectures learn a local optimal solution of balancing the pendulum downwards. This local optima the conservative solution as the cost associated with the cart position is comparatively high to avoid the cart limits on the physical system. Therefore, stabilizing the pendulum downwards is better compared to swinging the pendulum up and failing at the balancing. The locally quadratic network with the feature transform, learns the optimal policy for the cartpole. This architecture avoids the local solution as the network structure guides the exploration to be optimistic and the feature transform simplifies the value function learning to learn a successful balancing.

Ablation Study - Model Ensemble

The learning curves for different model ensemble sizes are shown in Figure 10. The model ensemble does not significantly affect the performance of the final policy but reduces the variance in learning speed between seeds. The variance between seeds also increases. The reduced variance for the model ensembles is caused by the smoothing of the network initialization. The mean across different initial weights lets the initialization be more conservative compared to a single network. For the comparatively small value function networks (i.e., 2-3 layers deep and 64 - 128 units wide), we prefer the network ensembles as the computation time does not increase when increasing the ensemble size. If the individual networks are batched and evaluated on the GPU the computation time does not increase. The network ensembles could also be evaluated at 500Hz for the real-time control experiments using an Intel i7 9900k.