# Appendix: Control Regularization for Reduced Variance Reinforcement Learning

## A. Proof of Lemma 1

**Lemma 1.** *The policy $\overline{u}_k(s)$ in Equation (6) is the solution to the following regularized optimization problem,*

$$\overline{u}_k(s) = \arg\min_u \left\|u(s) - \overline{u}_{\theta_k}\right\|^2 + \lambda||u(s) - u_{prior}(s)||^2, \quad \forall s \in S, \tag{16}$$

*which can be equivalently expressed as the constrained optimization problem:*

$$\begin{aligned}\overline{u}_k(s) = \arg\min_u \ & \left\|u(s) - \overline{u}_{\theta_k}\right\|^2 \\ \text{s.t.} \ & ||u(s) - u_{prior}(s)||^2 \leq \tilde{\mu}(\lambda) \quad \forall s \in S,\end{aligned} \tag{17}$$

*where $\tilde{\mu}$ constrains the policy search. Assuming convergence of the RL algorithm, $\overline{u}_k(s)$ converges to the solution,*

$$\begin{aligned}\overline{u}_k(s) = \arg\min_u \ & \left\|u(s) - \arg\max_{u_\theta} \mathbb{E}_{\tau\sim u}\Big[r(s,a)\Big]\right\|^2 \\ & + \lambda||u(s) - u_{prior}(s)||^2, \quad \forall s \in S \quad \text{as} \quad k \to \infty\end{aligned} \tag{18}$$

*Proof.*

**Equivalence between (6) and (16) :** Let $\pi_{\theta_k}(a|s)$ be a Gaussian distributed policy with mean $\overline{u}_{\theta_k}(s)$: $\pi_{\theta_k}(a|s) \sim \mathcal{N}(\overline{u}_{\theta_k}(s), \Sigma)$. Thus, $\Sigma$ describes exploration noise. From the mixed policy definition (6), we can obtain the following Gaussian distribution describing the mixed policy:

$$\begin{aligned}\pi_k(a|s) &= \mathcal{N}(\frac{1}{1+\lambda}\overline{u}_{\theta_k} + \frac{1}{1+\lambda}u_{prior}, \Sigma) \\ &= \frac{1}{c_N}\mathcal{N}\Big(\overline{u}_{\theta_k}(s), (1+\lambda)\Sigma\Big) \cdot \mathcal{N}\Big(u_{prior}(s), \frac{1+\lambda}{\lambda}\Sigma\Big),\end{aligned} \tag{19}$$

where the second equality follows based on the properties of products of Gaussians. Let us define $\|u_1 - u_2\|_\Sigma = (u_1 - u_2)^T\Sigma^{-1}(u_1 - u_2)$, and let $|\Sigma|$ be the determinant of $|\Sigma|$. Then, distribution (19) can be rewritten as the product,

$$\begin{aligned}\mathbb{P}(X(s)) = &-c_1 \exp(-\frac{1}{2(1+\lambda)}\|X(s) - \overline{u}_{\theta_k}(s)\|_\Sigma) \times \\ &- c_1\lambda^{\frac{k}{2}} \exp(-\frac{\lambda}{2(1+\lambda)}\|X(s) - u_{prior}(s)\|_\Sigma) \\ c_1 = &\frac{1}{c_N\sqrt{(2\pi)^k(1+\lambda)^k|\Sigma|}}\end{aligned} \tag{20}$$

where $X(s)$ is a random variable with $\mathbb{P}(X(s))$ representing the probability of taking action $X$ from state $s$ under policy (6). Further simplifying this PDF, we obtain:

$$\begin{aligned}\mathbb{P}(X(s)) &= c_2 \exp\Big(-\|X(s) - \overline{u}_{\theta_k}(s)\|_\Sigma \\ &\qquad - \lambda\|X(s) - u_{prior}(s)\|_\Sigma\Big) \\ c_2 &= \frac{\lambda^{\frac{k}{2}}}{c_N(2\pi)^k(1+\lambda)^k|\Sigma|}\end{aligned} \tag{21}$$

Since the probability $\mathbb{P}(X(s))$ is maximized when the argument of the exponential in Equation (21) is minimized, then the maximum probability policy can be expressed as the solution to the following regularized optimization problem,

$$\begin{aligned}\overline{u}_k(s) = \arg\min_u(s) \ & \|u(s) - \overline{u}_{\theta_k}(s)\|_\Sigma + \\ & \lambda\|u(s) - u_{prior}(s)\|_\Sigma, \quad \forall s \in S.\end{aligned} \tag{22}$$

Therefore the mixed policy $\overline{u}_k(s)$ from Equation (6) is the solution to Problem (16) .

**Convergence of (16) to (18):** Note that $\overline{u}_{\theta_k}$ and $\pi_{\theta_k}$ are parameterized by the same $\theta_k$ and represent the iterative solution to the optimization problem $\arg\max_\theta \mathbb{E}_{\tau\sim u_k}\Big[r(\tau)\Big]$ at the latest policy iteration. Thus, assuming convergence of the RL algorithm, we can rewrite problem (22) as follows,

$$\begin{aligned}\overline{u}_k = \arg\min_u \ & \left\|u(s) - \arg\max_{u_{\theta_k}} \mathbb{E}_{\tau\sim u_k}\Big[r(s,a)\Big]\right\|^2 \\ & + \lambda||u(s) - u_{prior}(s)||^2, \quad \forall s \in S.\end{aligned} \tag{23}$$

**Equivalence between (16) and (17) :** Finally, we want to show that the solutions for regularized problem (16) and the constrained optimization problem (17) are equivalent.

First, note that Problem (16) is the dual to Problem (17), where $\lambda$ is the dual variable. Clearly problem (16) is convex in $u$. Furthermore, Slater's condition holds, since there is always a feasible point (e.g. trivially $u(s) = u_{prior}(s)$). Therefore strong duality holds. This means that $\exists\lambda \geq 0$ such that the solution to Problem (17) must also be optimal for Problem (16).

To show the other direction, fix $\lambda > 0$ and define $R(u) = \|u(s) - \overline{u}_{\theta_k}(s)\|^2$ and $C(u) = ||u(s) - u_{prior}(s)||^2$ for all $s \in S$. Let us denote $u^*$ as the optimal solution for Problem (16) with $C(u^*) = \tau > \tilde{\mu}$ (note we can choose $\tilde{\mu}$). However supposed $u^*$ is *not* optimal for Problem (17). Then there exists $\tilde{u}$ such that $R(u^*) < R(\tilde{u})$ and $C(\tilde{u}) \leq \tilde{\mu}$. Denote the difference in the two rewards by $R(\tilde{u}) - R(u^*) = R_{diff}$. Thus the following relations hold,

$$R(\tilde{u}) + \lambda C(\tilde{u}) < R(u^*) + \lambda C(u^*) + R_{diff} + \lambda\big[\tilde{\mu} - \tau\big]. \tag{24}$$

This leads to the conditional statement,

$$\begin{aligned} &R_{diff} + \lambda\big[\tilde{\mu} - \tau\big] \geq 0 \\ &\quad \Rightarrow \quad R(\tilde{u}) + \lambda C(\tilde{u}) < R(u^*) + \lambda C(u^*). \end{aligned} \tag{25}$$

For fixed $\lambda$, there always exists $\tilde{\mu} > 0$ such that the condition $R_{diff} + \lambda\big[\tilde{\mu} - \tau\big] \geq 0$ holds. However, this leads to a contradiction, since we assumed that $u^*$ is optimal for Problem (16). We can conclude then that $\exists \tilde{\mu}$ such that the solution to Problem (16) must be optimal for Problem (17). Therefore, Problems (16) and (17) have equivalent solutions.

□

## B. Proof of Theorem 1

**Theorem 1.** *Consider the mixed policy (5) where $\pi_{\theta_k}$ is an RL controller learned through policy gradients, and denote the (potentially local) optimal policy to be $\pi_{opt}$. The variance (4) of the mixed policy arising from the policy gradient is reduced by a factor $(\frac{1}{1+\lambda})^2$ when compared to the RL policy with no control prior.*

*However, the mixed policy may introduce bias proportional to the sub-optimality of the control prior. More formally, if we let $D_{sub} = D_{TV}(\pi_{opt}, \pi_{prior})$, then the policy bias (i.e. $D_{TV}(\pi_k, \pi_{opt})$) is bounded as follows:*

$$\begin{aligned} &D_{TV}(\pi_k, \pi_{opt}) \geq D_{sub} - \frac{1}{1+\lambda} D_{TV}(\pi_{\theta_k}, \pi_{prior}) \\ &D_{TV}(\pi_k, \pi_{opt}) \leq \frac{\lambda}{1+\lambda} D_{sub} \qquad \text{as } k \to \infty \end{aligned} \tag{26}$$

*where $D_{TV}(\cdot, \cdot)$ represents the total variation distance between two probability measures (i.e. policies). Thus, if $D_{sub}$ and $\lambda$ are large, this will introduce policy bias.*

*Proof.* Let us define the stochastic action (i.e. random variable) $\mathcal{A}^{act}_{k+1} \sim \pi_{\theta_{k+1}}(a|s)$. Then recall from Equation (4) that assuming a fixed, Gaussian distributed policy, $\pi_{\theta_k}(a|s)$,

$$\text{var}_\theta[\mathcal{A}^{act}_{k+1}|s] \approx \alpha^2 \frac{d\pi_{\theta_k}}{d\theta} \text{var}_\theta[\nabla_\theta J(\theta_k)] \frac{d\pi_{\theta_k}}{d\theta}^T. \tag{27}$$

Based on the mixed policy definition (5), we obtain the following relation between the variance of $\pi_k$ and $\pi_{\theta_k}$ (the mixed policy and RL policy, respectively),

$$\begin{aligned} \text{var}_\theta[\pi_{k+1}] &= \text{var}_\theta\Big[\frac{1}{1+\lambda}\mathcal{A}^{act}_{k+1} + \frac{\lambda}{1+\lambda}u_{prior}|s\Big] \\ &= \frac{1}{(1+\lambda)^2}\text{var}_\theta[\mathcal{A}^{act}_{k+1}|s] \\ &= \frac{\alpha^2}{(1+\lambda)^2}\frac{d\pi_{\theta_k}}{d\theta}\text{var}_\theta[\nabla_\theta J(\theta_k)]\frac{d\pi_{\theta_k}}{d\theta}^T. \end{aligned} \tag{28}$$

Compared to the variance (4), we achieve a variance reduction when utilizing the same learning rate $\alpha$. Taking the same policy gradient from (4), var$[\nabla_\theta J(\theta_k)]$, then the variance is reduced by a factor of $(\frac{1}{1+\lambda})^2$ by introducing policy mixing.

Lower variance comes at a price – potential introduction of bias into policy. Let us define the policy bias as $D_{TV}(\pi_k, \pi_{opt})$, and let us denote $D_{sub} = D_{TV}(\pi_{opt}, \pi_{prior})$. Since total variational distance, $D_{TV}$ is a metric, we can use the triangle inequality to obtain:

$$D_{TV}(\pi_k, \pi_{opt}) \geq D_{TV}(\pi_{prior}, \pi_{opt}) - D_{TV}(\pi_{prior}, \pi_k). \tag{29}$$

We can further break down the term $D_{TV}(\pi_{prior}, \pi_k)$:

$$\begin{aligned} &D_{TV}(\pi_{prior}, \pi_k) \\ &\quad = \sup_{(s,a)\in S\text{x}A}\Big|\pi_{prior} - \frac{1}{1+\lambda}\pi_{\theta_k} - \frac{\lambda}{1+\lambda}\pi_{prior}\Big| \\ &\quad = \frac{1}{1+\lambda}\sup_{(s,a)\in S\text{x}A}|\pi_{\theta_k} - \pi_{prior}| \\ &\quad = \frac{1}{1+\lambda}D_{TV}(\pi_{\theta_k}, \pi_{prior}). \end{aligned} \tag{30}$$

This holds for all $k \in \mathbb{N}$. From (29) and (30), we can obtain the lower bound in (26),

$$D_{TV}(\pi_k, \pi_{opt}) \geq D_{sub} - \frac{1}{1+\lambda}D_{TV}(\pi_{\theta_k}, \pi_{prior})$$

To obtain the upper bound, let the policy gradient algorithm with *no* control prior achieve asymptotic convergence to the (locally) optimal policy $\pi_{opt}$ (as proven for certain classes of function approximators in (Sutton et al., 1999)). Denote this policy as $\pi^{(p)}_{\theta_k}$, such that $\pi^{(p)}_{\theta_k} \to \pi_{opt}$ as $k \to \infty$. In this case, we can derive the total variation distance between the mixed policy (5) and the optimal policy as follows,

$$\begin{aligned} &D_{TV}(\pi_{opt}, \pi_k^{(p)}) \\ &\quad = \sup_{(s,a)\in S\text{x}A} |\pi_{opt} - \frac{1}{1+\lambda}\pi_{\theta_k}^{(p)} - \frac{\lambda}{1+\lambda}\pi_{prior}| \\ &\quad = \frac{\lambda}{1+\lambda} \sup_{(s,a)\in S\text{x}A} |\pi_{opt} - \pi_{prior}| \quad \text{as } k \to \infty \\ &\quad = \frac{\lambda}{1+\lambda} D_{TV}(\pi_{opt}, \pi_{prior}) \quad \text{as } k \to \infty \\ &\quad = \frac{\lambda}{1+\lambda} D_{sub} \quad \text{as } k \to \infty. \end{aligned} \tag{31}$$

Note that this represents an *upper bound* on the bias, since it assumes that $\pi_{\theta_k}^{(p)}$ is uninfluenced by $\pi_{prior}$ during learning. It shows that $\pi_{\theta_k}^{(p)}$ is a feasible policy, but not necessarily optimal when accounting for regularization with $\pi_{prior}$. Therefore, we can obtain the upper bound:

$$\begin{aligned} D_{TV}(\pi_{opt}, \pi_k) &\leq D_{TV}(\pi_{opt}, \pi_k^{(p)}) \\ &= \frac{\lambda}{1+\lambda} D_{sub} \quad \text{as } k \to \infty. \end{aligned} \tag{32}$$

□

## C. Proof of Lemma 2

**Lemma 2.** *For any state $s$, satisfaction of the condition,*

$$\begin{aligned} 2s^T P\Big(d(s,a) + \frac{1}{1+\lambda}B_2 u_e\Big) < \\ s^T(C_1^T C_1 + \frac{1}{\gamma_k^2} P B_1 B_1^T P)s, \end{aligned}$$

*implies that $\dot{V}(s) < 0$.*

*Proof.* Recall that we are analyzing the Lyapunov function $V(s) = s^T P s$, where P is taken from the Algebraic Riccati Equation (50). Let us take the time derivative of the Lyapunov function as follows:

$$\begin{aligned} \dot{V}(s) &= \frac{dV}{ds}\dot{s} = 2s^T P\Big(As + B_2 a + d(s,a)\Big) \\ &= s^T(-C_1^T C_1 - \frac{1}{\gamma_k^2} P B_1 B_1^T P)s + 2s^T P d(s,a) + \\ &\qquad \frac{2}{1+\lambda} s^T P B_2 (u_{\theta_k} - u_{prior}) \\ &= s^T(-C_1^T C_1 - \frac{1}{\gamma_k^2} P B_1 B_1^T P)s + 2s^T P\Big(d(s,a) + \\ &\qquad \frac{1}{1+\lambda} B_2 u_e\Big). \end{aligned} \tag{33}$$

The second equality comes from the Algebraic Riccati Equation (50), which the dynamics satisfy by design of the $\mathcal{H}^\infty$ controller. From here, it follows directly that if,

$$\begin{aligned} 2s^T P\Big(d(s,a) + \frac{1}{1+\lambda}B_2 u_e\Big) < \\ s^T(C_1^T C_1 + \frac{1}{\gamma_k^2} P B_1 B_1^T P)s, \end{aligned}$$

then $\dot{V}(s) < 0$. □

## D. Proof of Theorem 2

**Theorem 2.** *Assume a stabilizing $H^\infty$ control prior within the set $\mathcal{C}$ for our dynamical system (14). Then asymptotic stability and forward invariance of the set $\mathcal{S}_{st} \subseteq \mathcal{C}$*

$$\begin{aligned} \mathcal{S}_{st} : \{s \in \mathbb{R}^n : \|s\|_2 \leq \frac{1}{\sigma_m(\gamma_k)}\Big(2\|P\|_2 C_D \\ + \frac{2}{1+\lambda}\|PB_2\|_2 C_\pi\Big)\ ,\ s \in \mathcal{C}\}. \end{aligned} \tag{34}$$

*is guaranteed under the mixed policy (5) for all $s \in \mathcal{C}$. The set $\mathcal{S}_{st}$ contracts as we (a) increase robustness of the control prior (increase $\sigma_m(\gamma_k)$), (b) decrease our dynamic uncertainty/nonlinearity $C_D$, or (c) increase weighting $\lambda$ on the control prior.*

*Proof.*

**Step (1):** Find a set in which Lemma 2 is satisfied.

Consider the condition in Lemma 2. Since the right hand side is positive (quadratic), we can consider a bound on the stability condition as follows,

$$\begin{aligned} |2s^T P d(s,a) + \frac{2}{1+\lambda} s^T P B_2 u_e| < \\ s^T(C_1^T C_1 + \frac{1}{\gamma_k^2} P B_1 B_1^T P)s. \end{aligned} \tag{35}$$

Clearly any set of $s$ that satisfy condition (35) also satisfy the condition in Lemma 2. To find such a set, we bound the terms in Condition (35) as follows,

$$\begin{aligned} &|2s^T P d(s,a) + \frac{2}{1+\lambda} s^T P B_2 u_e| \\ &\quad \leq\ 2|s^T P d(s,a)| + \frac{2}{1+\lambda}|s^T P B_2 u_e| \\ &\quad \leq\ 2\|s\|_2\|P\|_2 C_D + \frac{2}{1+\lambda}\|s\|_2\|PB_2\|_2 C_\pi, \end{aligned} \tag{36}$$

where the first inequality follows from the triangle inequality; the second inequality uses our bounds on the disturbance, $C_D$ and control input difference $C_\pi$, as well as the Cauchy-Schwarz inequality. Now consider the right hand side of Condition (35). Recall that $\sigma_m(\gamma_k) = \sigma_{min}(C_1^T C_1 + \frac{1}{\gamma_k^2} P B_1 B_1^T P)$, the minimum singular value. Then the following holds,

$$\sigma_m(\gamma_k)\|s\|_2^2 \leq s^T(C_1^T C_1 + \frac{1}{\gamma_k^2} P B_1 B_1^T P)s \qquad (37)$$

Using the bounds in (36) and (37), we can say that Condition (35) is guaranteed to be satisfied if the following holds,

$$2\|s\|_2\|P\|_2 C_D + \frac{2}{1+\lambda}\|s\|_2\|PB_2\|_2 C_\pi < \sigma_m(\gamma_k)\|s\|_2^2 \qquad (38)$$

The set for which this condition (38) is satisfied can be described by,

$$\begin{aligned} \mathcal{C} \setminus \mathcal{S}_{st} : \{s \in \mathbb{R}^n : \|s\|_2 > \frac{1}{\sigma_m(\gamma_k)}\Big(2\|P\|_2 C_D \\ + \frac{2}{1+\lambda}\|PB_2\|_2 C_\pi\Big)\ ,\ s \in \mathcal{C}\}. \end{aligned} \qquad (39)$$

Recall that $\mathcal{C}$ is the set in which the stabilizing $\mathcal{H}^\infty$ controller exists. From Lemma 2, $\dot{V}(s) < 0$ for all $s \in \mathcal{C} \setminus \mathcal{S}_{st}$ described by the set (39).

**Step (2):** Establish stability and forward invariance of $\mathcal{S}_{st}$.

The Lyapunov function $V(s) = s^T P s$ decreases towards the origin, and we have established that the time derivative of the Lyapunov function is negative for $s$ in set (39). Therefore, any state $s$ described by the set (39) (intersected with $\mathcal{C}$) must move towards the origin (i.e. towards $\mathcal{S}_{st}$). This follows directly from the properties of Lyapunov functions. Therefore, the set $\mathcal{S}_{st}$ described in (34) must be asymptotically stable and forward invariant for all $s \in \mathcal{C}$.

□

## E. Description of Experiments

### E.1. Experimental Car-Following

In the original car-following experiments, a chain of 8 cars followed each other on an 8-mile segment of a single-lane public road. We obtain position (via GPS), velocity, and acceleration data from each of the cars. We cut this data into 4 sets of chains of 5 cars, in order to maximize the data available to learn from. We then cut this into 10 second "episodes" (100 data points each). We shuffle these training episodes randomly before each run and feed them to the algorithm, which learns the controller for the $4^{th}$ car in the chain.

The reward function we use in learning is described below:

$$r = -\dot{v}\min(0,a) - 100|G_1(s)| - 50G_2(s),$$

$$G_1(s) = \begin{cases} \frac{1}{s_{front}-s_{curr}} & \text{if } s_{front} - s_{curr} \leq 10 \\ \frac{1}{s_{curr}-s_{back}} & \text{if } s_{curr} - s_{back} \leq 10 \\ 0 & \text{otherwise} \end{cases} \qquad (40)$$

$$G_2(s) = \begin{cases} 1 & \text{if } s_{front} - s_{curr} \leq 2 \\ 1 & \text{if } s_{curr} - s_{back} \leq 2 \\ 0 & \text{otherwise} \end{cases}$$

where $s_{curr}$, $s_{front}$, and $s_{back}$ denote the position of the controlled car, the car in front of it, and the car behind it. Also, $a$ denotes the control action (i.e. acceleration/deceleration), and $\dot{v}$ denotes the velocity of the controlled car. Therefore, the first term represents the fuel efficiency of the controlled car, and the other terms encourage the car to maintain headway from the other cars and avoid collision.

The control prior we utilize is a simple bang-bang controller that (inefficiently) tries to keep us between the car and front and back. It is described by,

$$a = \begin{cases} 2.5 & \text{if } K_p\Delta s + K_d\Delta v > 0 \\ -5 & \text{if } K_p\Delta s + K_d\Delta v < 0 \\ 0 & \text{otherwise} \end{cases} \qquad (41)$$

$$\Delta s = s_{front} - 2s_{curr} - s_{back}$$
$$\Delta v = v_{front} - 2v_{curr} - v_{back}$$

where $v_{curr}$, $v_{front}$, and $v_{back}$ denote the velocity of the controlled car, the car in front of it, and the car behind it. We set the constants $K_p = 0.4$ and $K_d = 0.5$. Essentially, the control prior tries to maximize the distance from the car in front and behind, taking into account velocities as well as positions.

### E.2. TORCS Racecar Simulator

In its full generality TORCS provides a rich environment with input from up to 89 sensors, and optionally the 3D graphic from a chosen camera angle in the race. The controllers have to decide the values of up to 5 parameters during game play, which correspond to the acceleration, brake, clutch, gear and steering of the car. Apart from the immediate challenge of driving the car on the track, controllers also have to make race-level strategy decisions, like making pit-stops for fuel. A lower level of complexity is provided in the *Practice Mode* setting of TORCS. In this mode all race-level strategies are removed. Currently, so far as we know, state-of-the-art DRL models are capable of racing only in Practice Mode, and this is also the environment that we use. In this mode we consider the input from 29 sensors, and decide values for the acceleration, steering and brake actions.

The control prior we utilize is a linear controller of the form:

$$K_p(\epsilon - o_i) + K_i \sum_{j=i-N}^{i} (\epsilon - o_j) + K_d(o_{i-1} - o_i) \quad (42)$$

Where $o_i$ is the most recent observation provided by the simulator for a chosen sensor, and $N$ is a predetermined constant. We have one controller for each of the actions, acceleation, steering and braking.

The pseudo-reward used during training is given by:

$$r_t = V\cos(\theta) - V\sin(\theta) - V|\texttt{trackPos}| \quad (43)$$

Here $V$ is the velocity of the car, $\theta$ is the angle the car makes with the track axis, and `trackPos` provides the position on the track relative to the track's center. This reward captures the aim of maximizing the longitudinal velocity, minimizing the transverse velocity, and penalizing the agent if it deviates significantly from the center of the track.

### E.3. CartPole Stabilization

The CartPole simulator is implemented in the OpenAI gym environment ('CartPole-v1'). The dynamics are the same as in the default, as described below,

$$\begin{aligned}
\theta_{t+1} &= x_t + \dot{x}\tau, \\
\dot{\theta}_{t+1} &= \dot{\theta}_t + \Big(\frac{Mg\sin\theta - F\cos\theta - ml\dot{\theta}^2\sin\theta\cos\theta}{\frac{4}{3}Ml - ml\cos^2\theta}\Big)\tau, \\
x_{t+1} &= x_t + \dot{x}\tau, \\
\dot{x}_{t+1} &= \dot{x}_t + \Big(\frac{F + ml\dot{\theta}^2\sin\theta - ml\ddot{\theta}\cos\theta}{M}\Big)\tau,
\end{aligned} \quad (44)$$

where the only modification we make is that the force on the cart can take on a continuous value, $F \in [-10, 10]$, rather than 2 discrete values, making the action space much larger. Since the control prior can already stabilize the CartPole, we also modify the reward to characterize *how well* the control stabilizes the pendulum. The reward function is stated below, and incentivizes the CartPole to keep the pole upright while minimizing movement in the x-direction:

$$r = -100|\theta| - 2x^2. \quad (45)$$

## F. Control Theoretic Stability Guarantees

This section in the Appendix goes over the same material in Section 5, but goes into more detail on the $\mathcal{H}^\infty$ problem definition. Consider the linear dynamical system described by:

$$\begin{aligned}
\dot{s} &= As + B_1 w + B_2 a \\
z &= C_1 s + D_{11} w + D_{12} a \\
y &= C_2 s + D_{21} w + D_{22} a
\end{aligned} \quad (46)$$

where $w \in \mathbb{R}^{m_1}$ is the disturbance vector, $u \in \mathbb{R}^{m_1}$ is the control input vector, $z \in \mathbb{R}^{p_1}$ is the error vector (controlled output), $y \in \mathbb{R}^{p_2}$ is the observation vector, and $s \in \mathbb{R}^n$ is the state vector. The system transfer function is denoted,

$$\begin{aligned}
P^s(s) &= \begin{pmatrix} P^s_{11} & P^s_{12} \\ P^s_{21} & P^s_{22} \end{pmatrix} \\
&= \begin{pmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{pmatrix} + \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} (sI - A)^{-1} \begin{bmatrix} B_1 & B_2 \end{bmatrix},
\end{aligned} \quad (47)$$

where $A, B_i, C_i, D_{ij}$ are defined by the system model (46). Let us make the following assumptions,

- The pairs $(A, B_2)$ and $(C_2, A)$ are stabilizable and observable, respectively.
- The algebraic Riccati equation $A^T P + PA + C_1^T C_1 + P(B_2 B_2^T - \frac{1}{\gamma_k^2} B_1 B_1^T)P = 0$ has positive-semidefinite solution $P$,
- The algebraic Riccati equation $AP_Y + P_Y A^T + B_1^T B_1 = P_Y(C_2 C_2^T - \frac{1}{\gamma_k^2} C_1 C_1^T)P_Y$ has positive-semidefinite solution $P_Y$,
- The matrix $\gamma I - P_Y P$ is positive definite.

Under these assumptions, we are guaranteed existence of a stabilizing linear $\mathcal{H}^\infty$ controller, $u^{\mathcal{H}^\infty} = -Ks$ (Doyle et al., 1989). The closed-loop transfer function from disturbance, $w$, to controlled output, $z$, is:

$$T_{wz} = P^s_{11} + P^s_{12} K (I - P^s_{22} K)^{-1} P^s_{21}. \quad (48)$$

Let $\sigma(\cdot)$ denotes the maximum singular value of the argument, and recall that $\|T_{wz}\|_\infty := \sup_w \sigma(T_{wz}(jw))$. Then the $H_\infty$ controller solves the problem,

$$\min_K \sup_w \sigma(T_{wz}(jw)) = \gamma_k, \quad (49)$$

to give us controller $u^{\mathcal{H}^\infty} = -Ks$. This generates the maximally robust controller so that the *worst-case disturbance* is attenuated by factor $\gamma_k$ in the system before entering the controlled output. We can synthesize the $H_\infty$ controller using techniques described in (Doyle et al., 1989).

The $H_\infty$ controller is defined as $u^{\mathcal{H}_\infty} = -B_2^T Px$, where $P$ is a positive symmetric matrix satisfying the Algebraic Riccati equation,

$$A^T P + PA + C_1^T C_1 + \frac{1}{\gamma_k^2} PB_1 B_1^T P - PB_2 B_2^T P = 0, \quad (50)$$

where $(A, B_1, B_2, C_1)$ are defined in (46). The result is that the control law $u^{\mathcal{H}_\infty}$ stabilizes the system with disturbance attenuation $\|T_{wz}\|_\infty \leq \gamma_k$.

Since we are not dealing with a linear system, we need to consider a modification to the dynamics (46) that *linearizes* the dynamics about some equilibrium point and gathers together all non-linearities and disturbances,

$$\dot{s} = f_c(s, a) = As + B_2 a + d(s, a), \tag{51}$$

where $d(s, a)$ captures dynamic uncertainty/nonlinearity as well as disturbances. To keep this small, we could use feedback linearization based on our nominal nonlinear model (1), but this is outside the scope of this work.

Consider the Lyapunov function $V(s) = s^T P s$, where P is taken from Equation (50). We can analyze stability of the uncertain system (14) under the mixed policy (5) using Lyapunov analysis. We can utilize Lemma 2 in this analysis (see Appendix C) in order to compute a set $\mathcal{S}_{st}$ such that $\dot{V}(s) < 0$ in a region outside that set. Satisfaction of this condition guarantees forward invariance of that set (Khalil, 2000), as well as its asymptotic stability (from the region for which $\dot{V}(s) < 0$).

By bounding terms as described in Section 5, we can conservatively compute the set $\mathcal{S}_{st}$ for which $\dot{V}(s) < 0$, which is shown in Theorem 2. See Appendix D for the derivation of the set (i.e. proof of Theorem 2).

## G. PPO + TRPO Results

We also ran all experiments using Proximal Policy Optimization (PPO) or Trust Region Policy Optimization (TRPO) in place of DDPG. The results are shown in Figures 5 and 6. The trends mirror those seen in the main paper using DDPG. Low values of $\lambda$ exhibit significant deterioration of performance, because of the larger policy search space. High values of $\lambda$ also exhibit lower performance because they heavily constrain learning. Intermediate $\lambda$ allow for the best learning, with good performance and low variance. Furthermore, adaptive strategies for setting $\lambda$ allows us to better tune the reward-variance tradeoff.

Note that we do not show results for the TORCS Racecar. This is because we were not able to get the baseline PPO or TRPO agent to complete a lap throughout learning. The code for the PPO, TRPO, and DDPG agent for each environment can be found at `https://github.com/rcheng805/CORE-RL`.

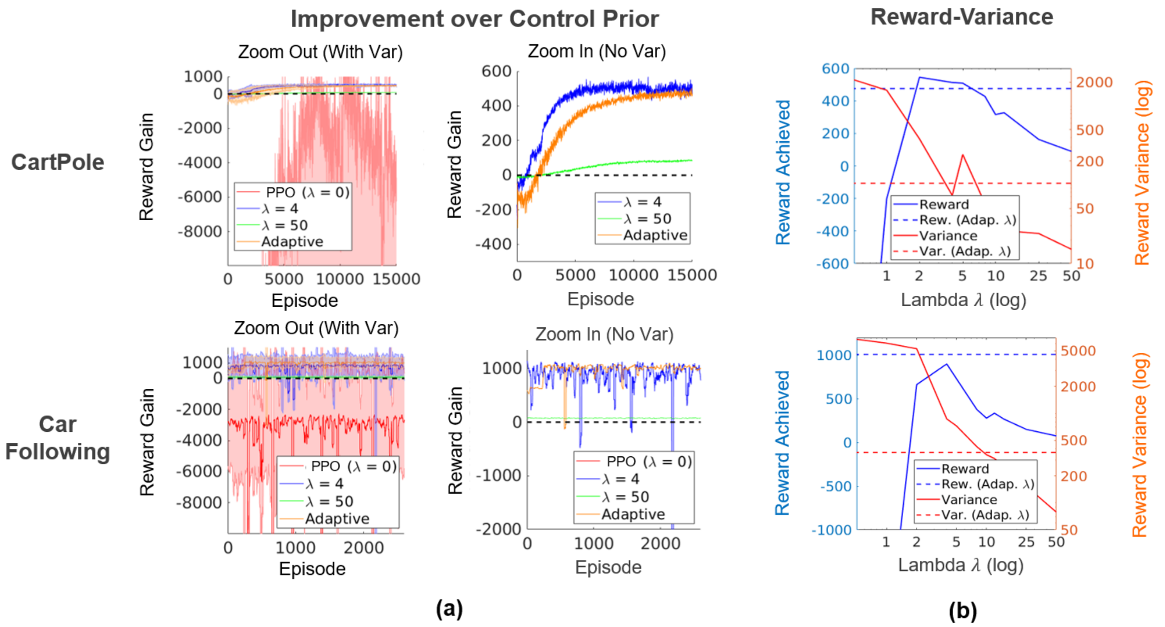

*Figure 5.* Learning results for CartPole and Car-Following Problems using PPO. (a) Reward improvement over control prior with different set values for $\lambda$ or an adaptive $\lambda$. The right plot is a zoomed-in version of the left plot without variance bars for clarity. Values above the dashed black line signify improvements over the control prior. (b) Performance and variance in the reward as a function of the regularization $\lambda$, across different runs of the algorithm using random initializations/seeds. Dashed lines show the performance (i.e. reward) and variance using the adaptive weighting strategy. Variance is measured for all episodes across all runs. Again, performance is baselined to the control prior, so any performance value above 0 denotes improvement over the control prior.

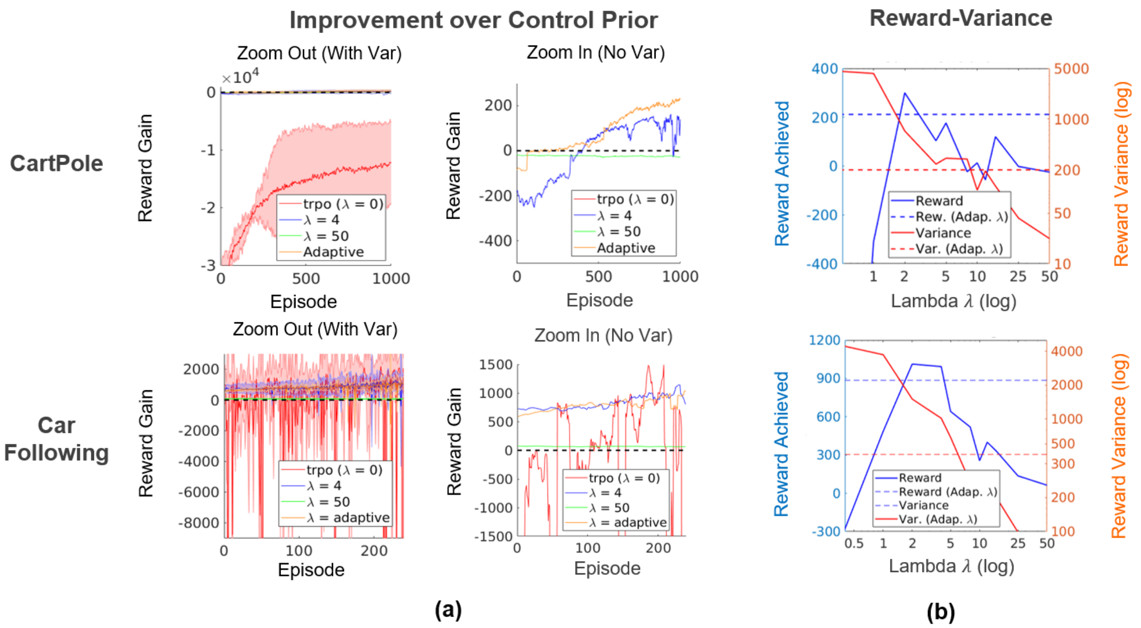

*Figure 6.* Learning results for CartPole and Car-Following Problems using TRPO. (a) Reward improvement over control prior with different set values for $\lambda$ or an adaptive $\lambda$. The right plot is a zoomed-in version of the left plot without variance bars for clarity. Values above the dashed black line signify improvements over the control prior. (b) Performance and variance in the reward as a function of the regularization $\lambda$, across different runs of the algorithm using random initializations/seeds. Dashed lines show the performance (i.e. reward) and variance using the adaptive weighting strategy. Variance is measured for all episodes across all runs. Again, performance is baselined to the control prior, so any performance value above 0 denotes improvement over the control prior.