Adaptive Variants of Optimal Feedback Policies

Brett T. Lopez BTLOPEZ@UCLA.EDU
Verifiable and Control-Theoretic Robotics Laboratory, University of California, Los Angeles, CA

Jean-Jacques Slotine JJS@MIT.EDU
Nonlinear Systems Laboratory, Massachusetts Institute of Technology, MA
Google AI

Editors: R. Firoozi, N. Mehr, E. Yel, R. Antonova, J. Bohg, M. Schwager, M. Kochenderfer

Abstract

The stable combination of optimal feedback policies with online learning is studied in a new control-theoretic framework for uncertain nonlinear systems. The framework can be systematically used in transfer learning and sim-to-real applications, where an optimal policy learned for a nominal system needs to remain effective in the presence of significant variations in parameters. Given unknown parameters within a bounded range, the resulting adaptive control laws guarantee convergence of the closed-loop system to the state of zero cost. Online adjustment of the learning rate is used as a key stability mechanism, and preserves certainty equivalence when designing optimal policies without assuming uncertainty to be within the control range. The approach is illustrated on the familiar mountain car problem, where it yields near-optimal performance despite the presence of parametric model uncertainty.

Keywords: Optimal Control - Online Learning - Reinforcement Learning - Adaptive Control

1. Introduction

Autonomous decision-making and control have become ubiquitous in many safety-critical systems. This trend highlights the importance of developing principled algorithms that possess performance guarantees even in the face of uncertainty. Due to its versatility and generality, optimal control (Kirk, 2004; Bertsekas, 2012; Bryson and Ho, 2018; Sutton and Barto, 2018) is the primary framework for representing and solving difficult decision-making and control problems. In its purest form, optimal control entails computing a feedback policy that minimizes a cost function given a dynamical model and set of constraints. While knowing a model is not strictly necessary, e.g., model-free reinforcement learning (RL), any optimal policy will implicitly depend on the underlying dynamics of the system making it susceptible to model uncertainties. In practice, sensitivity to model perturbations can at best yield suboptimal performance, or at worst result in a catastrophic failure.

Online learning is an effective strategy that reduces sensitivity to model uncertainty while yielding a high-performance, non-conservative feedback policy. The first common strategy — known as indirect learning (or system identification) — generates an explicit model of the underlying dynamics that is used to synthesize a feedback policy. This approach is generally used in settings that require some form of prediction such as planning or games in addition to optimal control (Sutton and Barto, 2018; Bertsekas, 2022). Indirect methods, however, require sufficiently rich data to obtain an accurate enough model suitable for control leading to the exploration-exploitation tradeoff.

The second strategy — referred to as direct learning[1] — embraces the philosophy of learning just enough about the system to achieve the desired behavior. Direct methods have a rich history in the controls community, e.g., model reference adaptive control, but have not been fully utilized in optimal control aside from model-free RL Sutton et al. (1992) and adaptive dynamic programming (Murray et al., 2002; Vrabie and Lewis, 2009; Lewis and Vrabie, 2009). Unfortunately, model-free RL requires extensive offline training and suffers from limited robustness while adaptive dynamic programming needs to iteratively estimate the true cost-to-go online. A few other approaches have been proposed, e.g., (Agarwal et al., 2019a; Kumar et al., 2021), but the complexities and subtleties of combining online learning with nonlinear control often limit their applicability as they rely on linear systems theory or employ ad hoc techniques that do not generalize.

Contributions. We develop a new adaptive optimal control framework that utilizes the certainty equivalence principle and online adjustment of the learning rate to guarantee closed-loop stability of near-optimal policies for nonlinear systems with parametric uncertainties. The approach consists of combining online learning with optimal value functions and policies computed offline for a family of dynamical systems. While typically the stability of such a combination cannot be ensured, we can guarantee that the closed-loop system will converge to the state of zero cost by adjusting the learning rate online (Lopez and Slotine, 2020). As a result, this work is the first to successfully combine Lyapunov-based learning with optimal control for nonlinear systems. Two learning algorithms are derived and shown to closely resemble the optimal policy for the well-known mountain car problem despite uncertainties in the dynamical model.

Scope. This work will consider deterministic, time-invariant, continuous-time optimal control of nonlinear systems with parametric uncertainties. Extension to other classes of problems is possible and is future work. More broadly, this work may find uses in model-free reinforcement learning, differential games, sim-to-real, transfer learning, underactuated robotics, or optimal prediction.

Notation. Let $\mathbb{R}_+$ and $\mathbb{R}_{>0}$ denote the set of positive and strictly positive reals, respectively. The shorthand notation for a function T parameterized by a vector a with vector argument s will be $T_a(s) \triangleq T(s;a)$. The partial differentiation with respect to variable $x \in \mathbb{R}^n$ of function $M(x,y)$ will be $\nabla_x M(x,y) = \partial M/\partial x \in \mathbb{R}^n$. The subscript for ∇ will be omitted when it is clear which variable the differentiation is with respect to. The desired terminal state will be denoted as x_d.

2. Optimal Control Review

Consider the deterministic, infinite-horizon optimal control problems of the form

$$\begin{aligned} V^*(x(t)) &= \min_{\pi \in \Pi} \int_t^\infty \ell(x(\tau), \pi(x(\tau)))\, d\tau \\ \dot{x} &= F_\theta(x, \pi(x)) \end{aligned} \tag{1}$$

with state $x \in \mathbb{R}^n$, feedback policy $\pi : \mathbb{R}^n \to \mathbb{R}^m$, partially known dynamics $F_\theta : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}^n$ parameterized by unknown coefficients $\theta \in \mathbb{R}^p$, stage cost $\ell : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}_+$ where $\ell(x, \pi(x)) = 0 \iff x = x_d$, and control constraint set Π. The optimal value function $V^* : \mathbb{R}^n \to \mathbb{R}_+$ is the cost-to-go from the initial state $x(t)$ if the optimal policy $\pi^*(x)$ is executed indefinitely.

[1] Model-based and model-free RL are synonymous with indirect and direct, respectively.

The initial state can be represented as a function of time, i.e., $x(t)$, because the optimal control problem (1) possess an important time-invariant property where the initial time can be arbitrarily denoted as t. This property will be useful in analyzing the stability of the proposed approach. Note that a value function can also be defined for a suboptimal stable policy $\pi(x)$ and satisfies $V^*(x) \leq V^\pi(x) < \infty$ where the last inequality follows from $\pi(x)$ being a stabilizing controller[2]. The following assumptions are made about the dynamics in (1).

Assumption 1 *The dynamics $F_\theta(\cdot)$ can be decomposed into known and unknown dynamics with $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ representing the known part.*

Assumption 2 *The uncertain part of the nonlinear dynamics in (1) can be expressed as a linear combination of known basis functions $\Delta : \mathbb{R}^n \to \mathbb{R}^{p\times n}$ and unknown parameters $\theta \in \mathbb{R}^p$ that belong to a closed convex set Θ.*

Remark 1 *Assumption 2 is not overly restrictive as systems with non-parametric or nonlinearly parameterized uncertainties can be converted into a linear weighting of handpicked or learned basis functions (O'Connell et al., 2021; Richards et al., 2021)*

A common formulation for optimal control of dynamical systems with bounded parametric uncertainties is the following minimax optimization

$$\begin{aligned} V^*(x(t)) &= \min_{\pi\in\Pi}\max_{\theta\in\Theta} \int_t^\infty \ell(x(\tau),\pi(x(\tau)))\,d\tau \\ \dot{x} &= f(x,\pi(x)) - \Delta(x)^\top\theta. \end{aligned} \tag{2}$$

The optimal value function for (2) can be obtained by solving the Hamilton-Jacobi-Isaacs equation

$$\min_{\pi\in\Pi}\max_{\theta\in\Theta}\left\{\ell(x,\pi(x)) + \nabla_x V^*(x)^\top\left(f(x,\pi(x)) - \Delta(x)^\top\theta\right)\right\} = 0. \tag{3}$$

Conceptually, solving (3) is equivalent to computing a *single policy for all possible dynamical models*, i.e., a robust control strategy, which will inherently perform worse than its optimal counterpart if θ where known. We instead propose a certainty equivalence approach that uses a *family* of value functions $V^*_\theta(x)$ and policies $\pi^*_\theta(x)$, parameterized by the unknown model parameter vector θ, that satisfy the Hamilton-Jacobi-Bellman (HJB) equation

$$\min_{\pi_\theta\in\Pi}\left\{\ell(x,\pi_\theta(x)) + \nabla_x V^*_\theta(x)^\top\left(f(x,\pi_\theta(x)) - \Delta(x)^\top\theta\right)\right\} = 0, \quad \text{for each } \theta\in\Theta. \tag{4}$$

The difference between (3) and (4) is quite significant from a theoretical and practical point of view. The policy generated by solving (3) is the best policy given the worst possible realization of the model uncertainty; in essence a "one-size-fits-all" approach. Alternatively, the policy satisfying (4) will be optimal if θ is known. From the perspective of online learning, (4) allows one to employ the certainty equivalence principle to obtain a near-optimal policy with the current model estimate. The challenge then becomes designing the learning algorithm since combining a stable estimator and feedback policy does not necessarily yield a stable closed-loop if the system is nonlinear (Krstic et al., 1995). Moreover, the model-parameterized value functions present additional challenges as learning transients can also lead to unstable control. Next, two online learning algorithms will be proposed that can be stably combined with model-parameterized value functions and policies.

[2] A non-finite value function would indicate either $x \nrightarrow x_d$ or $x \to x_d$ but "slow enough" that infinite cost is accumulated. Imposing $V^\pi(x)$ be finite eliminates both scenarios.

3. Theory

3.1. Preliminaries

Before presenting the main theorems, we first recall a simple but fundamental result due to (Kalman and Bertram, 1960, p. 387), see also (Luenberger, 1979, p. 425-427): an optimal value function $V^*(x)$ is also a control Lyapunov function. This fact is central to our approach.

Proposition 2 *An optimal value function $V^*(x)$ is also a control Lyapunov function.*

Proof Recall $V^*(x(t)) = \int_t^\infty \ell(x(\tau), \pi(x(\tau)))\, d\tau > 0$ for all $x \neq x_d$ and $V^*(x_d) = 0$. Since $\dot{V}^*(x) = -\ell(x, \pi^*(x)) < 0$ for all $x \neq x_d$, then $V^*(x)$ is a control Lyapunov function. ■

Proposition 2 can be extended to suboptimal policies $\pi(x)$, e.g., policies which approximate $\pi^*(x)$, as long as one can show that the associated value function $V^\pi(x)$ is finite over the operating domain. Hence, once a valid value function $V^\pi(x)$ is known, whether optimal or not, one can conclude the closed-loop system converges to x_d with policy $\pi(x)$.

The proposed approach requires the following two differentiabilty assumptions.

Assumption 3 *The stage cost $\ell(\cdot)$ is continuously differentiable.*

Assumption 4 *The optimal value function $V_\theta^*(x)$ and policy $\pi_\theta^*(x)$ are continuously differentiable.*

Assumption 3 is easy to ensure by appropriate selection of $\ell(\cdot)$. Assumption 4 may seem more restrictive as it excludes discontinuous optimal policies, e.g., bang-bang control. However, an optimal policy is often itself the result of as differentiable computing pipeline, or otherwise can be smoothed by appropriate selection of a continuously differentiable function. For instance, if the optimal policy is restricted to take discrete values, e.g., $\pi^*(x) \in \Pi = \{\pi^*(x) : \{-1, 0\, 1\},\ \forall x\}$, then functions like the saturation function, logistic function, etc. can be used to make $\pi^*(x)$ continuously differentiable with negligible performance degradation given a reasonable choice of parameters. Similar smoothing can be applied to the optimal value function.

We will make use of the Bregman divergence operator in our online learning algorithm to impose physical consistency (Wensing et al., 2017; Lee et al., 2018) or sparsity based on past trajectories (Ghai et al., 2020; Boffi and Slotine, 2021) of the parameter estimates.

Definition 3 (Bregman Divergence) *Let $\psi(\cdot)$ be a strictly convex, continuously differentiable function on a closed convex set. The* Bregman divergence *associated with $\psi(\cdot)$ is given by*

$$\mathrm{d}_\psi(y \parallel x) = \psi(y) - \psi(x) - (y - x)^\top \nabla\psi(x), \tag{5}$$

with a time-derivative that satisfies $\dot{\mathrm{d}}_\psi(y \parallel x) = (x - y)^\top \nabla^2\psi(x)\, \dot{x}$.

3.2. Direct Online Learning

We now state the first main technical result of this work.

Theorem 4 *Let $\upsilon(\cdot)$ be any strictly-increasing, strictly-positive scalar function and $\psi(\cdot)$ be a continuously differentiable, strictly convex function on a closed convex set. If a value function $V^*_\theta(x)$ and policy $\pi^*_\theta(x)$ can be computed for each $\theta \in \Theta$ then the closed-loop system asymptotically converges to the state of zero cost x_d with the policy $\pi^*_{\hat\theta}(x)$ and learning algorithm*

$$\dot{\hat\theta} = -\gamma\, \upsilon(\rho)[\nabla^2\psi(\hat\theta)]^{-1}\Delta(x)\nabla_x V^*_{\hat\theta}(x), \tag{6a}$$

$$\dot\rho = -\frac{\upsilon(\rho)}{\nabla\upsilon(\rho)}\sum_{i=1}^{p}\left[\frac{1}{V^*_{\hat\theta}(x)+\eta}\,\nabla_{\hat\theta_i}V^*_{\hat\theta}(x)\right]\dot{\hat\theta}_i, \tag{6b}$$

where $\gamma \in \mathbb{R}_{>0}$ is the learning rate and $\eta \in \mathbb{R}_{>0}$.

Proof Consider the Lyapunov-like function

$$V_c(t) = \upsilon(\rho)(V^*_{\hat\theta}(x)+\eta) + \tfrac{1}{\gamma}\mathrm{d}_\psi(\theta \parallel \hat\theta), \tag{7}$$

where $0 < \eta < \infty$ and $\tilde\theta \triangleq \hat\theta - \theta$. Differentiating (7) along the unknown dynamics,

$$\begin{aligned}
\dot V_c(t) &= \upsilon(\rho)\dot V^*_{\hat\theta}(x) + \dot\rho\,\nabla\upsilon(\rho)(V^*_{\hat\theta}(x)+\eta) + \tfrac{1}{\gamma}\tilde\theta^\top\nabla^2\psi(\hat\theta)\,\dot{\hat\theta}\\
&= \upsilon(\rho)\Big[\nabla_x V^*_{\hat\theta}(x)^\top\left(f(x,\pi^*_{\hat\theta}(x)) - \Delta(x)^\top\theta\right) + \sum_{i=1}^{p}\nabla_{\hat\theta_i}V^*_{\hat\theta}(x)\,\dot{\hat\theta}_i\Big]\\
&\quad + \dot\rho\,\nabla\upsilon(\rho)(V^*_{\hat\theta}(x)+\eta) + \tfrac{1}{\gamma}\tilde\theta^\top\nabla^2\psi(\hat\theta)\,\dot{\hat\theta}.
\end{aligned}$$

Note the presence of the sign-indefinite terms $\sum_{i=1}^{p}\nabla_{\hat\theta_i}V^*_{\hat\theta}(x)\,\frac{d}{dt}\hat\theta_i$ which appear because the value function is parameterized by $\hat\theta$. Using (4) and the definition $\theta = \hat\theta - \tilde\theta$,

$$\begin{aligned}
\dot V_c(t) &= \upsilon(\rho)\nabla_x V^*_{\hat\theta}(x)^\top\left(f(x,\pi^*_{\hat\theta}(x)) - \Delta(x)^\top\hat\theta\right) + \upsilon(\rho)\nabla_x V^*_{\hat\theta}(x)^\top\Delta(x)^\top\tilde\theta\\
&\quad + \sum_{i=1}^{p}\nabla_{\hat\theta_i}V^*_{\hat\theta}(x)\dot{\hat\theta}_i + \dot\rho\,\nabla\upsilon(\rho)(V^*_{\hat\theta}(x)+\eta) + \tfrac{1}{\gamma}\tilde\theta^\top\nabla^2\psi(\hat\theta)\,\dot{\hat\theta}\\
&= -\upsilon(\rho)\ell(x,\pi^*_{\hat\theta}(x)) + \upsilon(\rho)\nabla_x V^*_{\hat\theta}(x)^\top\Delta(x)^\top\tilde\theta + \sum_{i=1}^{p}\nabla_{\hat\theta_i}V^*_{\hat\theta}(x)\dot{\hat\theta}_i\\
&\quad + \dot\rho\,\nabla\upsilon(\rho)(V^*_{\hat\theta}(x)+\eta) + \tfrac{1}{\gamma}\tilde\theta^\top\nabla^2\psi(\hat\theta)\,\dot{\hat\theta}.
\end{aligned}$$

Using Eqs. (6a) and (6b) yields $\dot V_c(t) = -\upsilon(\rho)\ell(x,\pi^*_{\hat\theta}(x)) \leq 0$ which implies that $\tilde\theta$ and the product $\upsilon(\rho)(V^*_{\hat\theta}(x)+\eta)$ are bounded. Since $V^*_{\hat\theta}(x) > 0$ for all $x \neq x_d$ and $\upsilon(\rho) > 0$ uniformly, then both $V^*_{\hat\theta}(x)$ and $\upsilon(\rho)$ are bounded for all $x \neq x_d$. When $x = x_d$ the learning rate is zero, i.e., $\frac{d}{dt}\hat\theta = 0$, which implies $\dot\rho = 0$ from (6b) so $\upsilon(\rho)$ remains bounded. Hence, $V^*_{\hat\theta}(x)$ is bounded because η, $\upsilon(\rho)$, and $\upsilon(\rho)(V^*_{\hat\theta}(x)+\eta)$ are bounded. Furthermore, boundedness of $V^*_{\hat\theta}(x)$ implies that x is also bounded. Differentiating $\upsilon(\rho)\ell(x,\pi^*_{\hat\theta}(x))$ and utilizing (6b),

$$\frac{d}{dt}[\upsilon(\rho)\ell(x,\pi^*_{\hat\theta}(x))] = \upsilon(\rho)\dot\ell(x,\pi^*_{\hat\theta}(x)) - \upsilon(\rho)\ell(x,\pi^*_{\hat\theta}(x))\sum_{i=1}^{p}\left[\frac{1}{V^*_{\hat\theta}(x)+\eta}\,\nabla_{\hat\theta_i}V^*_{\hat\theta}(x)\right]\dot{\hat\theta}_i$$

which is bounded by smoothness of $V^*_{\hat{\theta}}(x)$, $\ell(x, \pi^*_{\hat{\theta}}(x))$ and boundedness of x, $\upsilon(\rho)$. Hence, $\upsilon(\rho)\ell_{\hat{\theta}}(x, \pi^*(x))$ is uniformly continuous. Noting $V_c(0)$ is initially bounded, integrating $\frac{d}{dt}V_c(t)$ yields $\int_0^\infty \upsilon(\rho(\tau))\ell_{\hat{\theta}}(x(\tau), \pi^*(x(\tau)))\, d\tau < \infty$ so by Barbalat's lemma $\upsilon(\rho)\ell_{\hat{\theta}}(x, \pi^*(x)) \to 0$. Since $\upsilon(\rho) > 0$ uniformly and $\ell_{\hat{\theta}}(x, \pi^*(x)) = 0 \iff x = x_d$ then $x \to x_d$ as $t \to +\infty$. ■

Theorem 4 is highly versatile. The direct learning algorithm (6) *guarantees* the system will converge to the state of zero cost despite nonlinear dynamics and without learning the true underlying model. This result is an immediate consequence of using direct adaptive control formulation. The intuition behind Theorem 4 is that the function $\upsilon(\rho)$ actively adjusts the learning rate online thereby leading to an *effective* learning rate $\gamma\,\upsilon(\rho)$. Conceptually, the online adjustment is needed to ensure the learning transients, i.e., $\frac{d}{dt}\hat{\theta}$, does not destabilize the closed-loop system. Inspecting (6), the effective learning rate is slowed when the learning transients is destabilizing while the converse is true when the learning transients is stabilizing. The online adjustment allows us to employ the certainty equivalence principle, i.e., select the optimal policy with the current parameter estimate, without concerns of instability.

The direct learning law has an interesting connection with Lagrange multipliers (or costates of (1)). Denoting $\lambda^* \in \mathbb{R}^n$ as the Lagrange multipliers for the optimal solution of (1), one can trivially show $\nabla_x V^*_\theta(x) = \lambda^*$. The adaptation law (6) can thus be interpreted as updating the parameters in the direction of lower cost-to-go along the vector field $\Delta(x)$. This relation provides a potentially attractive alternative to computing the gradient of $V^*(x)$ numerically, which may be cumbersome for high-dimensional systems. It may also lead to a new adaptive optimal control paradigm based on the Hamiltonian function $\mathcal{H}_\theta(x, \lambda) \triangleq \ell(x, \pi_\theta(x)) + \lambda^\top F_\theta(x, \pi_\theta(x))$ where

$$\dot{x} = \nabla_\lambda \mathcal{H}_\theta(x, \lambda)$$
$$\dot{\lambda} = -\nabla_x \mathcal{H}_\theta(x, \lambda)$$

which, in combination with the condition $\pi^*_\theta(x) = \arg\min_{\pi \in \Pi} \mathcal{H}_\theta(x, \lambda)$, can be easier to solve than the HJB equation as the above is just a set of ordinary differential equations. Future work will investigate this dual framework in more detail.

3.3. Hybrid Online Learning

The direct learning algorithm in (6) uses $\nabla_x V^*_{\hat{\theta}}(x)$ as an error signal to update the parameter estimates. While stable control and learning is the ultimate goal, better performance can be obtained by adding a state predictor to the adaptation law. A state predictor is simply a form of model estimation that, when combined with direct learning, yields smoother learning transients. We refer to the combination of direct learning and state prediction as *hybrid learning*.

Definition 5 (State Predictor) *The* state predictor *is* $\varepsilon_{\hat{\theta}}(x) \triangleq \frac{d}{dt}(x - x_{\hat{\theta}})$ *where* $\frac{d}{dt}x_{\hat{\theta}}$ *is the instantaneous state speed with the current parameter estimate, i.e.,* $\frac{d}{dt}x_{\hat{\theta}} = F_{\hat{\theta}}(x, \pi^*_{\hat{\theta}}(x))$.

The benefit of using Definition 5 as opposed to other predictors is the relation between $\varepsilon_{\hat{\theta}}(x)$ and $\tilde{\theta}$, namely $\varepsilon_{\hat{\theta}}(x) = \Delta(x)^\top \tilde{\theta}$ which can be arrived at using the definition of $\frac{d}{dt}x$ and $\frac{d}{dt}x_{\hat{\theta}}$. This identity will be important for showing stability of the hybrid learning algorithm.

Remark 6 *In practice* $\frac{d}{dt}x$ *if often not available. However, one can instead use the output of a first-order filter in place of* $\frac{d}{dt}x$. *For instance, if we let* $\frac{d}{dt}\hat{x} = \beta(x - \hat{x})$ *where* $\beta \in \mathbb{R}_{>0}$ *then it is*

straightforward to show the equivalence

$$\dot{\hat{x}} = \hat{f}(x, \pi^*(x)) - \hat{\Delta}(x)^\top \theta$$

where $\hat{f}(x, \pi^*(x))$ *and* $\hat{\Delta}(x)$ *are the filtered version of* $f(x, \pi^*(x))$ *and* $\Delta(x)$*, respectively. If identical filtering is done for* $\frac{d}{dt} x_{\hat{\theta}}$*, then one can let* $\varepsilon_{\hat{\theta}}(x) = \frac{d}{dt}(\hat{x} - \hat{x}_{\hat{\theta}})$ *which only depends on filtered quantities that are immediately available. Moreover, we still maintain the important property* $\varepsilon_{\hat{\theta}}(x) = \hat{\Delta}(x)^\top \tilde{\theta}$*. This modification is just a practical one as the proposed hybrid learning algorithm will still yield a stable adaptive policy whether the actual or filtered state velocity is used.*

We now present the second main technical result of this work.

Theorem 7 *Let* $\upsilon(\cdot)$ *be any strictly-increasing, strictly-positive scalar function and* $\psi(\cdot)$ *be a continuously differentiable, strictly convex function on a closed convex set. If a value function* $V^*_\theta(x)$ *and policy* $\pi^*_\theta(x)$ *can be computed for each* $\theta \in \Theta$ *then the closed-loop system asymptotically converges to the state of zero cost* x_d *with the policy* $\pi^*_{\hat{\theta}}(x)$ *and learning algorithm*

$$\dot{\hat{\theta}} = -\gamma\, \upsilon(\rho)[\nabla^2 \psi(\hat{\theta})]^{-1} \Delta(x) \nabla_x V^*_{\hat{\theta}}(x) - \alpha\, [\nabla^2 \psi(\hat{\theta})]^{-1} \Delta(x) \varepsilon_{\hat{\theta}}(x), \tag{8a}$$

$$\dot{\rho} = -\frac{\upsilon(\rho)}{\nabla \upsilon(\rho)} \sum_{i=1}^{p} \left[\frac{1}{V^*_{\hat{\theta}}(x) + \eta} \nabla_{\hat{\theta}_i} V^*_{\hat{\theta}}(x) \right] \dot{\hat{\theta}}_i, \tag{8b}$$

where $\varepsilon_{\hat{\theta}}(x)$ *is the state predictor,* $\gamma \in \mathbb{R}_{>0}$ *is the direct learning rate,* $\alpha \in \mathbb{R}_{>0}$ *is the prediction learning rate, and* $\eta \in \mathbb{R}_{>0}$*.*

Proof Using the same Lyapunov-like function (7), one can show

$$\dot{V}_c(t) = -\upsilon(\rho)\ell(x, \pi^*_{\hat{\theta}}(x)) - \tfrac{\alpha}{\gamma}\, \tilde{\theta}^\top \Delta(x) \varepsilon_{\hat{\theta}}(x) = -\upsilon(\rho)\ell(x, \pi^*_{\hat{\theta}}(x)) - \tfrac{\alpha}{\gamma} \varepsilon_{\hat{\theta}}(x)^\top \varepsilon_{\hat{\theta}}(x) \leq 0.$$

Furthermore, the uniform continuity of $\dot{V}_c(t)$ can be establishing employing similar boundedness and continuity arguments as in the proof of Theorem 4. Since the integral of $\dot{V}_c(t)$ is finite, then $x \to x_d$ as $t \to +\infty$ by Barbalat's lemma. ■

It is interesting to note that the state predictor learning rate does not have to be adjusted online to maintain stability. Conversely, the direct learning rate *must* be adjusted to cancel the effects of learning transients on the closed-loop system. This arises from the fundamentally different roles of direct learning and the state predictor. The purpose of direct learning is to guarantee that the system remains stable, while the state predictor adds a form of model estimation to improve the parameter estimation transients. One can show (Slotine and Li, 1987) that the state predictor acts as a damping term in the $\tilde{\theta}$ dynamics[3]

$$\dot{\tilde{\theta}} + \alpha\, [\nabla^2 \psi(\hat{\theta})]^{-1} \Delta(x) \Delta(x)^\top\, \tilde{\theta} = -\gamma\, \upsilon(\rho)[\nabla^2 \psi(\hat{\theta})]^{-1} \Delta(x) \nabla_x V^*_{\hat{\theta}}(x), \tag{9}$$

which is a stable first-order filter with the damping coefficient $\alpha\, \nabla^2 \psi(\hat{\theta}) \Delta(x) \Delta(x)^\top \geq 0$. Observe that if $\alpha\, \nabla^2 \psi(\hat{\theta}) \Delta(x) \Delta(x)^\top = 0$ then the right hand side of (9) must also be zero.

[3] Recall θ is constant so $\frac{d}{dt}\tilde{\theta} = \frac{d}{dt}\hat{\theta}$.

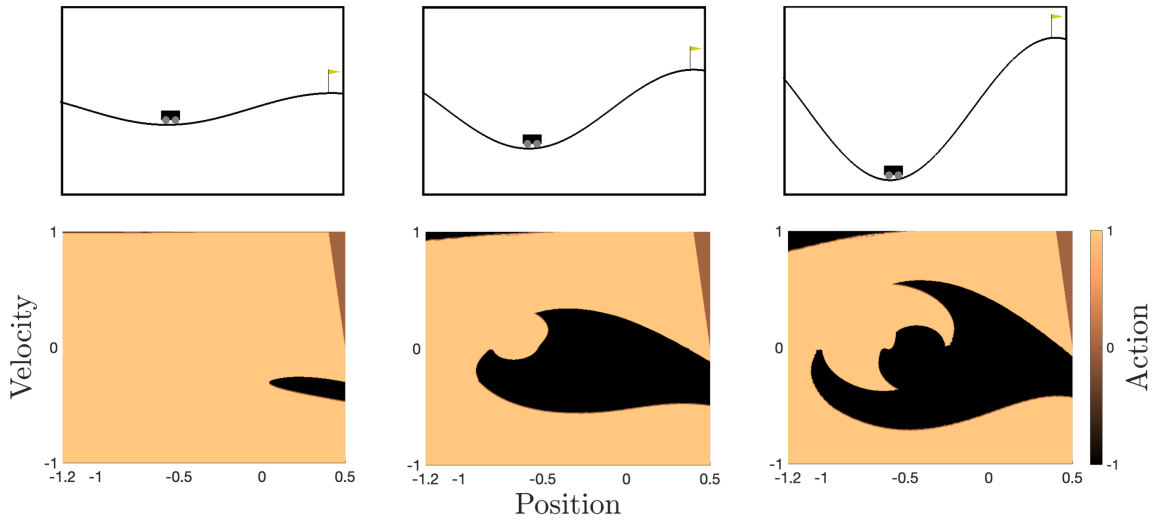


Figure 1: The proposed learning algorithms were tested in different mountain car environments (top). Each environment has its own unique optimal policy (bottom).

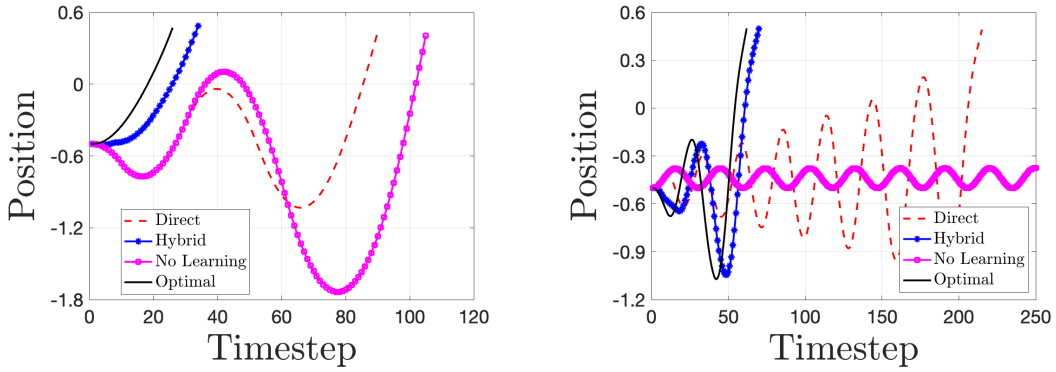
3.4. Incorporating State Constraints

Although not explicitly represented in the optimal control problem (1), the proposed framework can incorporate state constraints by constructing an augmented stage cost $\ell_a : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}_+$ that penalizes states near the constraint. For example, consider a constraint set $\mathcal{C}$ with boundary ∂C and interior $\operatorname{Int}\mathcal{C}$. Let the state constraint be of the form $g(x) \leq 0$ where $g(x) \to 0$ as $x \to \partial\mathcal{C}$. The augmented stage cost can then be defined to be $\ell_a(x, \pi(x)) \triangleq \ell(x, \pi(x)) + \sigma(g(x))$ where $\sigma(\cdot)$ is continuously differentiable, $\sigma(\cdot) \geq 0$ for all $x \in \operatorname{Int}\mathcal{C}$, and $\sigma(g(x)) \to \infty$ as $x \to \partial\mathcal{C}$. The value function $V^*(x(t)) = \int_t^\infty \ell_a(x(\tau), \pi(x(\tau)))\, d\tau$ is still a control Lyapunov function as it possess all the necessary properties (see Proposition 2). In contrast to control barrier functions Ames et al. (2016, 2019), using an augmented cost function yields less myopic control since the optimal policy has explicit knowledge of the constraints.

4. Experiments: Mountain Car

Overview. The direct and hybrid learning algorithms were tested on a modified version of the benchmark mountain car problem. The goal is for an under-powered car to reach the top of a mountain with an unknown slope. The continuous-time optimal control problem is

$$\begin{aligned} V_\theta^*(p(t), v(t)) = \min_{\pi \in \Pi} \int_t^\infty \ell_a(p(\tau), v(\tau))\, d\tau \\ \dot{p} = v,\ \dot{v} = 0.1\,\pi_\theta(p, v) - \theta\cos(3\,p), \\ p \in [-1.2,\ 0.5]\,\mathrm{m},\ v \in [-1,\ 1]\,\mathrm{m/s}, \\ \pi_\theta(p, v) \in \{-1,\ 0,\ 1\}\,\mathrm{m/s^2},\ \theta \in [0.05,\ 0.4]\,\mathrm{m/s^2}. \end{aligned} \tag{10}$$



(*a*) Relatively flat mountain.

(*b*) Steep mountain.

Figure 2: Comparison of four policies for different mountain car environments.

The slope of the mountain is considered to be unknown, but varies continuously from relatively flat to steep (top row of Fig. 1). The corresponding optimal policies are significantly different depending on the slope of the mountain, as seen in the bottom row of Fig. 1. The policy for the relatively flat mountain is to predominately drive forward as the car has enough power to reach the top. However, as the mountain becomes steeper the policy changes to a switching function where the vehicle must accelerate forward and backward until it has enough momentum to reach the top. The dramatically different feedback policies further motivates the use online learning to ensure the closed-loop system is stable and performing well.

Parameters. All differential equations were discretized with a time step $dt = 0.001\,\mathrm{s}$. The direct and state predictor learning rates were $\gamma = 0.02$ and $\alpha = 200000$, respectively. Note the large magnitude difference is due to the relative sizes of the error signal used by each component. The stage cost was chosen to be $\ell(p) = 1 - \exp(-40|p - p_d|)$ which is continuously differentiable and satisfies $\ell(p_d) = 0$. The augmented stage cost $\ell_a(\cdot)$ utilized an L_1 penalty on states near the path constraints of (10). The learning rate scaling function was $\upsilon(\rho) = 0.9\,e^{\rho} + 0.1$ and $\eta = 100$. The parameter estimates were bounded by using the the Bregman divergence with the log function.

Methods. Optimal value functions and feedback policies were computed offline using value iteration (VI) for several different mountain slopes. The dynamics were discretized with a time step of $0.1\,\mathrm{s}$ so the discrete-time version of VI could be employed. Bilinear interpolation between states was utilized to improve accuracy and rate of convergence.

Results. The direct and hybrid learning algorithms were evaluated in different environments by comparing their performance to the true optimal policy and to a static (no learning) policy. Each policy (aside from the optimal) was initialized with model parameters that were furthest from that of the true model, i.e., the policy for the relatively flat mountain ($\theta = 0.05\,\mathrm{m/s^2}$) was used on the steep mountain ($\theta = 0.4\,\mathrm{m/s^2}$) and vice versa. Fig. 2 shows the closed-loop position for the different policies when deployed on the relatively flat (Fig. 2(*a*)) and steep (Fig. 2(*b*)) mountain. In both environments, the direct and hybrid algorithms outperform the static policy. The steep mountain test case (Fig. 2(*b*)) shows that the closed-loop system still converges to the state of zero cost with the direct and hybrid learning policies despite the initial policy being unstable. This experimentally confirms the stability guarantees that both algorithms possess. Quantitatively, the hybrid learning

Table 1: Closed-Loop Cost of Different Policies

Mountain Grade	No Learning	Direct	Hybrid	Optimal
Relatively Flat	20.9	17.9	**6.57**	5.01
Steep	∞	42.7	**13.7**	12.1

policy most closely resembles the optimal policy, yielding a similar closed-loop cost as shown in Table 1. The improved performance is an immediate benefit of incorporating state perdition in learning. Nonetheless, direct learning is necessary to ensure closed-loop stability so a combination of the two is ideal.

5. Discussion

This work proposed two online learning algorithms that can be combined with optimal feedback policies to improve closed-loop performance when the underlying dynamical model is not fully known. By taking a control-theoretic approach, we proved asymptotic convergence of the closed-loop system to the state of zero cost. We also empirically showed that combining direct learning with a state predictor can yield a near-optimal stable policy. The approach leverages the certainty equivalence principle when computing optimal policies through online adjustment of the learning rate. Unlike other existing approaches, stability is *guaranteed*. The proposed learning framework may have a profound impact on offline computation/training of optimal policies as it is highly parallelizable, i.e., generate N policies for N different dynamical models independently. More generally, it bypasses the "one-size-fits-all" strategy of finding a single policy for all possible models, simplifying the control synthesis problem.

Several future works are of interest. The first is a thorough regret analysis (Hazan et al., 2007; Agarwal et al., 2019b; Dean et al., 2018; Boffi et al., 2021) of the proposed approach. The empirical results presented here hint that the developed adaptive policies are nearly optimal, but a rigorous analysis is needed before any claims can be made. Moreover, regret may play a fundamental role in designing the state predictor or other modifications to the proposed learning law. Nonetheless, our approach possess strong stability characteristics that apply to both linear and nonlinear systems in its current form. Investigating the relationship between the choice of the stage cost, i.e., reward shaping (Ng et al., 1999), and robustness is also of interest. As shown in the Appendix of (Lopez and Slotine, 2021), an appropriate choice of the stage cost can guarantee converge to the state of zero cost even without online learning, although an adaptive robust strategy is, in our view, the most effective. More analysis on adaptive combinations of optimal policies is also of interest as several phenomenon encountered in nature are interconnected combinations of systems that minimize some cost function, e.g., energy. Lastly, additional empirical evaluation will be pursued, including investigating uses in RL, differential games, transfer learning, and model predictive control.

Appendix

Lemma 8 (Barbalat) *If a function $g(t)$ is uniformly continuous and $\lim_{t\to\infty} \int_0^t g(\tau)\, d\tau < \infty$, then $\lim_{t\to\infty} g(t) = 0$.*

Acknowledgments: We thank Michael Everett for stimulating discussions and Kenny Chen for helping with the figures.

References

Naman Agarwal, Brian Bullins, Elad Hazan, Sham Kakade, and Karan Singh. Online control with adversarial disturbances. In *International Conference on Machine Learning*, pages 111–119. PMLR, 2019a.

Naman Agarwal, Elad Hazan, and Karan Singh. Logarithmic regret for online control. *Advances in Neural Information Processing Systems*, 32:10175–10184, 2019b.

Aaron D Ames, Xiangru Xu, Jessy W Grizzle, and Paulo Tabuada. Control barrier function based quadratic programs for safety critical systems. *IEEE Transactions on Automatic Control*, 62(8): 3861–3876, 2016.

Aaron D Ames, Samuel Coogan, Magnus Egerstedt, Gennaro Notomista, Koushil Sreenath, and Paulo Tabuada. Control barrier functions: Theory and applications. In *2019 18th European Control Conference (ECC)*, pages 3420–3431. IEEE, 2019.

Dimitri Bertsekas. *Dynamic programming and optimal control: Volume I*, volume 1. Athena scientific, 2012.

Dimitri Bertsekas. *Lessons from alphazero for optimal, model predictive, and adaptive control*. Athena Scientific, 2022.

Nicholas M Boffi and Jean-Jacques E Slotine. Implicit regularization and momentum algorithms in nonlinearly parameterized adaptive control and prediction. *Neural Computation*, 33(3):590–673, 2021.

Nicholas M Boffi, Stephen Tu, and Jean-Jacques E Slotine. Regret bounds for adaptive nonlinear control. In *Learning for Dynamics and Control*, pages 471–483. PMLR, 2021.

Arthur E Bryson and Yu-Chi Ho. *Applied optimal control: optimization, estimation, and control*. Routledge, 2018.

Sarah Dean, Horia Mania, Nikolai Matni, Benjamin Recht, and Stephen Tu. Regret bounds for robust adaptive control of the linear quadratic regulator. *arXiv preprint arXiv:1805.09388*, 2018.

Udaya Ghai, Elad Hazan, and Yoram Singer. Exponentiated gradient meets gradient descent. In *Algorithmic Learning Theory*, pages 386–407. PMLR, 2020.

Elad Hazan, Amit Agarwal, and Satyen Kale. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69(2-3):169–192, 2007.

Rudolf E Kalman and John E Bertram. Control system analysis and design via the "second method" of lyapunov: I—continuous-time systems. *Trans. ASME Basic Engineering, Ser. D*, 82:371–400, 1960.

Donald E Kirk. *Optimal control theory: an introduction*. Courier Corporation, 2004.

Miroslav Krstic, Petar V Kokotovic, and Ioannis Kanellakopoulos. *Nonlinear and adaptive control design*. John Wiley & Sons, 1995.

Ashish Kumar, Zipeng Fu, Deepak Pathak, and Jitendra Malik. Rma: Rapid motor adaptation for legged robots. *arXiv preprint arXiv:2107.04034*, 2021.

Taeyoon Lee, Jaewoon Kwon, and Frank C Park. A natural adaptive control law for robot manipulators. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1–9. IEEE, 2018.

Frank L Lewis and Draguna Vrabie. Reinforcement learning and adaptive dynamic programming for feedback control. *IEEE circuits and systems magazine*, 9(3):32–50, 2009.

Brett T Lopez and Jean-Jacques E Slotine. Universal adaptive control of nonlinear systems. *IEEE Control Systems Letters*, To Appear, 2020.

Brett T Lopez and Jean-Jacques E Slotine. Adaptive variants of optimal feedback policies. *arXiv:2104.02709*, 2021.

David G Luenberger. *Introduction to dynamic systems; theory, models, and applications*. John Wiley & Sons, 1979.

John J Murray, Chadwick J Cox, George G Lendaris, and Richard Saeks. Adaptive dynamic programming. *IEEE transactions on systems, man, and cybernetics, Part C (Applications and Reviews)*, 32(2):140–153, 2002.

Andrew Y Ng, Daishi Harada, and Stuart Russell. Policy invariance under reward transformations: Theory and application to reward shaping. In *Icml*, volume 99, pages 278–287, 1999.

Michael O'Connell, Guanya Shi, Xichen Shi, and Soon-Jo Chung. Meta-learning-based robust adaptive flight control under uncertain wind conditions. *arXiv:2103.01932*, 2021.

SM Richards, N Azizan, J-JE Slotine, and M Pavone. Adaptive-control-oriented meta-learning for nonlinear systems. In *Robotics science and systems*, 2021.

J-J E Slotine and Weiping Li. On the adaptive control of robot manipulators. *International Journal of Robotics Research*, 6(3), 1987.

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

Richard S Sutton, Andrew G Barto, and Ronald J Williams. Reinforcement learning is direct adaptive optimal control. *IEEE Control Systems Magazine*, 12(2):19–22, 1992.

Draguna Vrabie and Frank Lewis. Neural network approach to continuous-time direct adaptive optimal control for partially unknown nonlinear systems. *Neural Networks*, 22(3):237–246, 2009.

Patrick M Wensing, Sangbae Kim, and Jean-Jacques E Slotine. Linear matrix inequalities for physically consistent inertial parameter identification: A statistical perspective on the mass distribution. *IEEE Robotics and Automation Letters*, 3(1):60–67, 2017.