

Gradient-Based Policy Iteration for Linear Quadratic Regulation with Finite-Time Guarantees

Vijayanand Digge and Gianluca Bianchin

ICTEAM institute, UCLouvain, {vijayanand.digge, gianluca.bianchin}@uclouvain.be

1 Introduction

Reinforcement learning (RL) provides a general framework for sequential decision making in uncertain environments. In continuous-state control, RL methods have attracted significant attention, with linear quadratic regulation (LQR) serving as a canonical benchmark that enables rigorous theoretical analysis. Classical policy iteration schemes for LQR rely on least-squares policy evaluation, which requires repeated matrix inversions and becomes computationally demanding in higher dimensions. This motivates gradient-based policy iteration methods that improve scalability while retaining data efficiency and admitting explicit finite-time performance guarantees.

In this work, we propose and analyze an online implementation of policy iteration for the LQR problem. Instead of explicitly solving the least-squares policy evaluation problem, we perform a finite number of gradient descent iterations on the quadratic Bellman error and update the policy via greedy minimization of the resulting approximate Q -function. Building on the finite-time policy iteration analysis for LQR [1], we provide explicit finite-time bounds on both the number of policy improvements and the number of gradient steps required to obtain a controller whose LQR cost is within a prescribed tolerance of the optimal cost.

2 Problem Formulation

Consider the linear discrete-time system

$$x(k+1) = Ax(k) + Bu(k) \quad (1)$$

where, $x(k) \in \mathbb{R}^n$ is the state, $u(k) \in \mathbb{R}^m$ is the input to the system. The state feedback controller is given by $u(k) = -Kx(k)$. We consider LQR control problem described by the minimization of the infinite-horizon cost:

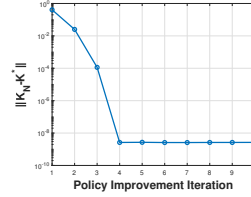
$$J(x(k), u(k)) = \sum_{k=0}^{\infty} x(k)^\top Qx(k) + u(k)^\top Ru(k) \quad (2)$$

where $Q, R \succ 0$.

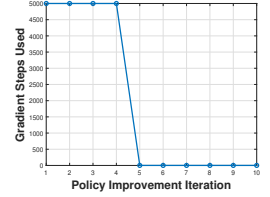
Problem: Compute an optimal LQR feedback gain K^* via policy iteration, and characterize the finite-time and sample complexity of the resulting algorithm when policy evaluation is performed using gradient-based methods.

3 Methodology and Results

The proposed method follows a policy iteration structure [1] with an inexact policy evaluation step. Given a stabilizing controller K_t at policy iteration t , policy evaluation is performed by approximating the corresponding state-action value function using a linear parametrization of the



(a) Policy convergence



(b) Adaptive gradient steps

Figure 1: This figure illustrates the simulation of gradient-based policy iteration. Figure 1a depicts convergence of the estimated policy K_N toward the optimal LQR gain K^* , illustrated by the evolution of the error norm $\|K_N - K^*\|$ over policy iteration steps N . Figure 1b represents adaptive gradient steps over policy iteration steps N .

Q -function. The associated Bellman error induces a convex quadratic loss whose minimizer coincides with the least-squares policy evaluation solution. The approximate Q -function obtained after T gradient descent steps is then used in a greedy policy improvement step, yielding an updated stabilizing controller K_{t+1} . After N policy improvement iterations, the algorithm returns the controller K_N .

Main Result: Let K_0 be an initial stabilizing controller. For any accuracy $\delta > 0$, there exist integers T and N , scaling logarithmically with $1/\delta$, such that if at least T gradient descent steps are performed at each policy evaluation, then the controller K_N obtained after N policy improvement iterations satisfies

$$\|K_N - K^*\| \leq \delta. \quad (3)$$

The bounds depend explicitly on the cost matrices, the initial controller, and the condition number of the sampled data covariance.

Simulation Results: We consider the example described in [2]. Figure 1a illustrates the convergence behavior of the policy across successive iterations. As observed, the policy quickly converges to the optimal LQR gain within a few iterations. Figure 1b shows the number of gradient descent steps required per policy iteration.

Future work will consider measurement noise in the data, leading to high-probability finite-sample guarantees.

References

- [1] K. Krauth, S. Tu, and B. Recht, "Finite-time analysis of approximate policy iteration for the linear quadratic regulator," *Advances in Neural Information Processing Systems*, vol. 32, 2019.
- [2] D. Lee, "Convergence of dynamic programming on the semidefinite cone for discrete-time infinite-horizon lqr," *IEEE Transactions on Automatic Control*, vol. 67, no. 10, pp. 5661–5668, 2022.