Adaptive Linear Quadratic Control Using Policy Iteration

Steven J. Bradtke,
B. Erik Ydstie, and
Andrew G. Barto

CMPSCI Technical Report 94-49

June 1994

NOTE: This paper is available by anonymous ftp from the site **ftp.cs.umass.edu** in the directory **pub/techrept/techreport/1994**.

Adaptive linear quadratic control using policy iteration

Steven J. Bradtke
Computer Science Department
University of Massachusetts
Amherst, MA 01003
bradtke@cs.umass.edu

B. Erik Ydstie
Department of Chemical Engineeering
Carnegie Mellon University
Pittsburgh, PA 15213
ydstie@andrew.cmu.edu

Andrew G. Barto
Computer Science Department
University of Massachusetts
Amherst, MA 01003
barto@cs.umass.edu

Abstract

In this paper we present stability and convergence results for Dynamic Programming-based reinforcement learning applied to Linear Quadratic Regulation (LQR). The specific algorithm we analyze is based on Q-learning and it is proven to converge to the optimal controller provided that the underlying system is controllable and a particular signal vector is persistently excited. The performance of the algorithm is illustrated by applying it to a model of a flexible beam.

This work was supported by the Air Force Office of Scientific Research, Bolling AFB, under Grant AFOSR F49620-93-1-0269 and by the National Science Foundation under Grant ECS-9214866 to Prof. Andrew Barto.

1 Introduction

In many practical applications a stabilizing feedback control for the system may be known. In this paper we discuss the problem of how to improve this controller and, under certain circumstances, make it converge to the optimal. The approach we take can be classified as direct optimal adaptive control and it is motivated by recent research on reinforcement learning which uses the principles of Dynamic Programming (DP). DP-based reinforcement learning algorithms include Sutton's Temporal Differences methods [9], Watkins' Q-learning [12], and Werbos' Heuristic Dynamic Programming [14]. Our approach is closely related to Q-learning. We apply the method to the Linear Qudratic Regulator (LQR) problem and we show that it converges to the optimal cost if the system is controllable and a particular signal vector is persistently excited. This is one of the first convergence results for DP-based reinforcement learning algorithms for a continuous state space. Previous results are limited to discrete time, finite-state systems, with either lookup-tables or linear function approximators. Watkins and Dayan [13] show that the Q-learning algorithm converges, under appropriate conditions, to the optimal Q-function for finite-state Markovian decision tasks, where the Q-function is represented by a lookup-table. Tsitsiklis [11] and Jaakkola, Jordan, and Singh [6] describe Q-learning as a form of stochastic approximation. Sutton [9] and Dayan [2] show that the linear TD(λ) learning rule, when applied to Markovian decision tasks where the states are representated by a linearly independent set of feature vectors, converges in the mean to V_{U}, the value function for a given control policy U.

Despite the paucity of theoretical results, applications of DP based algorithms with a continuous state representation have shown promise. For example, Tesauro [10] describes a system using TD(λ) that learns to play championship level backgammon, which can be viewed as a Markovian decision task, entirely through self-play. It uses a multilayer perceptron trained using backpropagation as a function approximator. Sofge and White [8] describe a system that learns to improve process control with continuous state and action spaces. Neither of these applications, nor many similar applications that have been described, have a firm theoretical grounding as yet. They do, however, produce good results experimentally. This paper takes a first step to provide a theoretical grounding for problems with continuous state and action spaces.

2 Problem Statement

Consider the discrete-time, multivariable system

$$x_{t+1} = f(x_t, u_t) = Ax_t + Bu_t \tag{1}$$

with feedback control

$$u_t = Ux_t$$

Here A, B, and U are matrices of dimensions $n \times n$, $n \times m$, and $m \times n$ respectively and U is chosen so that the matrix $A + BU$ has all of its eigenvalues strictly within the unit circle.

Associated with this system we assign a one step cost:

$$c_t = c(x_t, u_t) = x_t' E x_t + u_t' F u_t \tag{2}$$

where E is a symmetric positive semidefinite matrix of dimensions $n \times n$ and F is a symmetric positive definite matrix of dimensions $m \times m$. The *total cost* of a state x_t under the control policy U, $V_{\mathrm{U}}(x_t)$, is defined as the discounted sum of all costs that will be incurred by using U from time t onward, *i.e.*, $V_{\mathrm{U}}(x_t) = \sum_{i=0}^{\infty} \gamma^i c_{t+i}$, where $0 \leq \gamma \leq 1$ is the discount factor. This definition implies the recurrence relation

$$V_{\mathrm{U}}(x_t) = c(x_t, Ux_t) + \gamma V_{\mathrm{U}}(x_{t+1}). \tag{3}$$

V_{U} is a quadratic function [1] and therefore can be expressed as

$$V_{\mathrm{U}}(x_t) = x_t' K_{\mathrm{U}} x_t, \tag{4}$$

where K_{U} is the $n \times n$ *cost matrix* for policy U. U^* denotes the policy which is optimal in the sense that the total dicounted cost of every state is minimized. K^* represents the cost matrix associated with U^*.

It is a simple matter to derive U^* [1] *if accurate models of the system and cost function are available.* The problem we address is how to define an adaptive policy that converges to U^* *without* access to such models.

3 Q-functions and Policy improvement

Denardo [3] and Watkins [12] defined the Q-function for a stable control policy U as

$$Q_{\mathrm{U}}(x, u) = c(x, u) + \gamma V_{\mathrm{U}}(f(x, u)). \tag{5}$$

The value $Q_{\mathrm{U}}(x, u)$ is the sum of the one step cost incurred by taking action u from state x, plus the total cost that would accrue if the fixed policy U were followed from the state $f(x, u)$ and all subsequent states. u need not be the action specified by the given control policy for the state x. $Q_{\mathrm{U}}(x, u)$ is defined for all states x and *all* admissible control signals u. The function Q_{U} can also be defined recursively as

$$Q_{\mathrm{U}}(x_t, u_t) = c(x_t, u_t) + \gamma Q_{\mathrm{U}}(x_{t+1}, Ux_{t+1}), \tag{6}$$

by noting that

$$Q_{\mathrm{U}}(x, Ux) = c(x, Ux) + \gamma V_{\mathrm{U}}(f(x, Ux)) = V_{\mathrm{U}}(x). \tag{7}$$

For an LQR problem the Q function can be computed explicitly. We have

$$\begin{aligned}
Q_{\mathrm{U}}(x, u) &= c(x, u) + \gamma V_{\mathrm{U}}(f(x, u)) \\
&= x'Ex + u'Fu + \gamma (Ax + Bu)' K_{\mathrm{U}} (Ax + Bu) \\
&= x'(E + \gamma A' K_{\mathrm{U}} A)x + u'(F + \gamma B' K_{\mathrm{U}} B)u + \gamma x' A' K_{\mathrm{U}} B u + \gamma u' B' K_{\mathrm{U}} A x \\
&= \left[\begin{array}{c} x, u \end{array}\right]' \left[\begin{array}{cc} E + \gamma A' K_{\mathrm{U}} A & \gamma A' K_{\mathrm{U}} B \\ \gamma B' K_{\mathrm{U}} A & F + \gamma B' K_{\mathrm{U}} B \end{array}\right] \left[\begin{array}{c} x, u \end{array}\right] \\
&= \left[\begin{array}{c} x, u \end{array}\right] \left[\begin{array}{cc} H_{\mathrm{U}(11)} & H_{\mathrm{U}(12)} \\ H_{\mathrm{U}(21)} & H_{\mathrm{U}(22)} \end{array}\right] \left[\begin{array}{c} x, u \end{array}\right]' &(8) \\
&= \left[\begin{array}{c} x, u \end{array}\right]' H_{\mathrm{U}} \left[\begin{array}{c} x, u \end{array}\right], &(9)
\end{aligned}$$

where $[x, u]$ is the column vector concatenation of x and u and H_{U} is a symmetric positive definite matrix of dimensions $(n+m) \times (n+m)$. The submatrix $H_{\mathrm{U}(22)}$ is symmetric positive definite.

Given the policy U_k and the value function V_{U}, we can we find an improved policy, U_{k+1}, by following Howard [5] in defining U_{k+1} as

$$U_{k+1}x = \operatorname*{argmin}_u \left[c(x,u) + \gamma V_{\mathrm{U}}(f(x,u))\right].$$

But equation (7) tells us that this can be rewritten as

$$U_{k+1}x = \operatorname*{argmin}_u Q_{\mathrm{U}}(x,u).$$

We can find the minimizing u by taking the partial derivative of $Q_{\mathrm{U}}(x,u)$ with respect to u, setting that to zero, and solving for u. Taking the derivative we get

$$\frac{\partial Q_{\mathrm{U}}(x,u)}{\partial u} = 2(F + \gamma B' K_{\mathrm{U}} B)u + 2\gamma B' K_{\mathrm{U}} A x.$$

Setting that to zero and solving for u yields

$$u = \underbrace{-\gamma \left(F + \gamma B' K_{\mathrm{U}_k} B\right)^{-1} B' K_{\mathrm{U}_k} A}_{U_{k+1}} x.$$

Since the new policy U_{k+1} does not depend on x, it is the minimizing policy for all x. Using (8), U_{k+1} can be written as

$$U_{k+1} = -H^{-1}_{\mathrm{U}_k(22)} H_{\mathrm{U}_k(21)}.$$

The feedback policy U_{k+1} is per definition a stabilizing policy – it has no higher cost than U_k. A new Q function can then be assigned to this policy and the policy improvement procedure can be repeated *ad infinitum.*

Earlier work by Kleinman [7] and Bertsekas [1] showed that policy iteration will converge for LQR problems. However, the algorithms described Kleinman and Bertsekas required exact knowledge of the system model (equation 1) and the one-step cost function (equation 2). The analysis presented in this paper shows how policy iteration can be performed *without* that knowledge. Knowledge of the sequence of functions Q_{U_k} is sufficient.

4 Direct Estimation of Q-functions

We now show how the function Q_{U} can be directly estimated using recursive least squares (RLS). It is not necessary to identify either the system model or the one-step cost function separately. First, define the "overbar" function for vectors so that $\overline{x}$ is the vector whose elements are all of the quadratic basis functions over the elements of x, *i.e.,*

$$\overline{x} = \left[x_1^2, \ldots, x_1 x_n, x_2^2, \ldots, x_2 x_n, \ldots, x_{n-1}^2, x_{n-1}x_n, x_n^2\right]'.$$

Next, define the function Θ for square matrices. $\Theta(K)$ is the vector whose elements are the n diagonal entries of K and the $n(n+1)/2 - n$ distinct sums $(K_{ij} + K_{ji})$. The elements of $\overline{x}$ and $\Theta(K)$ are ordered so that $x'Kx = \overline{x}'\Theta(K)$. The original matrix K can be retrieved from $\Theta(K)$ if K is symmetric. If K is not symmetric, then we retrieve the symmetric matrix $\frac{1}{2}(K + K')$, which defines the same quadratic function as K. We can now write

$$Q_{\mathrm{U}}(x,u) = \left[\begin{array}{c} x, u \end{array}\right]' H_{\mathrm{U}} \left[\begin{array}{c} x, u \end{array}\right] = \overline{\left[\begin{array}{c} x, u \end{array}\right]}' \Theta(H_{\mathrm{U}}).$$

Finally, we rearrange equation (6) to yield

$$\begin{aligned} c(x_t, u_t) &= Q_{\mathrm{U}}(x_t, u_t) - \gamma Q_{\mathrm{U}}(x_{t+1}, Ux_{t+1}) \\ &= \left[\begin{array}{c} x_t, u_t \end{array}\right]' H_{\mathrm{U}} \left[\begin{array}{c} x_t, u_t \end{array}\right] - \gamma \left[\begin{array}{c} x_{t+1}, Ux_{t+1} \end{array}\right]' H_{\mathrm{U}} \left[\begin{array}{c} x_{t+1}, Ux_{t+1} \end{array}\right] \\ &= \overline{\left[\begin{array}{c} x_t, u_t \end{array}\right]}' \Theta(H_{\mathrm{U}}) - \gamma \overline{[x_{t+1}, Ux_{t+1}]}' \Theta(H_{\mathrm{U}}) \\ &= \phi_t' \theta_{\mathrm{U}}, \end{aligned}$$

where $\phi_t = \left\{\overline{[x_t, u_t]} - \gamma \overline{[x_{t+1}, Ux_{t+1}]}\right\}$, and $\theta_{\mathrm{U}} = \Theta(H_{\mathrm{U}})$.

Recursive Least Squares (RLS) can now be used to estimate θ_{U}. The recurrence relations for RLS are given by

$$\hat{\theta}_k(i) = \hat{\theta}_k(i-1) + \frac{P_k(i-1)\phi_t(c_t - \phi_t'\hat{\theta}_k(i-1))}{1 + \phi_t' P_k(i-1)\phi_t} \tag{10a}$$

$$P_k(i) = P_k(i-1) - \frac{P_k(i-1)\phi_t\phi_t' P_k(i-1)}{1 + \phi_t' P_k(i-1)\phi_t} \tag{10b}$$

$$P_k(0) = P_0. \tag{10c}$$

$P_0 = \beta I$ for some large positive constant β. $\theta_k = \Theta(H_{\mathrm{U}_k})$ is the true parameter vector for the function Q_{U_k}. $\hat{\theta}_k(i)$ is the i^{th} estimate of θ_k. The subscript t and the index i are both incremented at each time step. The reason for the distinction between t and i will be made clear in the next section.

Goodwin and Sin [4] show that this algorithm converges to the true parameters if θ_k is fixed and ϕ_t satisfies the persistent excitation condition

$$\epsilon_0 I \leq \frac{1}{N}\sum_{i=1}^{N} \phi_{t-i}\phi_{t-i}' \leq \bar{\epsilon}_0 I \qquad \text{for all} \qquad t \geq N_0 \text{ and } N \geq N_0 \tag{11}$$

where $\epsilon_0 \leq \bar{\epsilon}_0$, and N_0 is a positive number. But, it takes the algorithm an infinitely long time to converge to the true parameters.

5 Adaptive Policy Iteration for LQR

The policy improvement process based on Q-functions and the ability to directly estimate H_{U} (Section 4) are the two key elements of the adaptive policy iteration algorithm that is the focus of this paper. Figure 1 gives an outline of the algorithm.

Each policy iteration step consists of two phases: estimation of the Q-function for the current controller, and policy improvement based on that estimate. Consider the k^{th} policy iteration step. U_k is the current controller. $\theta_k = \Theta(H_{U_k})$, the true parameter vector for the function Q_{U_k}. $\hat{\theta}_k = \hat{\theta}_k(N)$ is the estimate of θ_k at the end of the parameter estimation interval. Each estimation interval is N time-steps long. The RLS algorithm is initialized at the start of the k^{th} estimation interval by setting $P_k(0) = P_0$ and initializing the parameter estimates for the k^{th} estimation interval to the final parameter estimates from the previous interval, *i.e.*, $\hat{\theta}_k(0) = \hat{\theta}_{k-1}(N)$. The index i used in equations (10) counts the number of time steps since the beginning of the estimation interval. After identifying the parameters $\Theta(H_{U_k})$ for N timesteps, one policy improvement step is taken based on the estimate $\hat{\theta}_k$. This produces the new controller U_{k+1}, and a new policy iteration step is begun.

Initialize parameters $\hat{\theta}_1(0)$.
$t = 0$, $k = 1$.
do forever {
    Initialize $P_k(0) = P_0$.
    for i = 1 to N {

    - $u_t = U_k x_t + e_t$, where e_t is the "exploration" component of the control signal.
    - Apply u_t to the system, resulting in state x_{t+1}.
    - Update the estimates of the Q-function parameters, $\hat{\theta}_k(i)$ using RLS (equations 10).
    - $t = t + 1$.

    }
    Find the symmetric matrix $\hat{H}_k$ that corresponds to the parameter vector $\hat{\theta}_k$.
    Perform policy improvement based on $\hat{H}_k$: $U_{k+1} = -\hat{H}_{k(22)}^{-1} \hat{H}_{k(21)}$.
    Initialize parameters $\hat{\theta}_{k+1}(0) = \hat{\theta}_k$.
    $k = k + 1$
}

Figure 1: The Q-function based policy iteration algorithm. It starts with the system in some initial state x_0 and with some stabilizing controller U_0. k keeps track of the number of policy iteration steps. t keeps track of the total number of time steps. i counts the number of time steps since the last change of policy. When $i = N$, one policy improvement step is executed.

Since the k^{th} policy improvement step is based on an *estimate* of $\Theta(H_{U_k})$, it is not clear *a priori* that the sequence U_k will converge to the optimal policy U^*, or even that each of the U_k's is guaranteed to be stabilizing. The convergence proofs of Kleinman [7] and Bertsekas [1] require exact knowledge of the system and take no account of estimation error. Theorem 1 establishes that the adaptive policy iteration algorithm presented above does indeed converge, under certain conditions, to the optimal controller.

Theorem 1: (Convergence of adaptive policy iteration). *Suppose that* $\{A, B\}$ *is a* *controllable pair, that U_0 is a stabilizing control, and that the vector $\phi(t)$ is persistently excited according to inequality (11). Then there exists an estimation interval $N < \infty$ so that the adaptive policy iteration mechanism described above generates a sequence $\{U_k, k = 1, 2, 3, ...\}$ of stabilizing controls, converging so that*

$$\lim_{k\to\infty} \|U_k - U^*\| = 0,$$

where U^* is the optimal feedback control matrix.

Proof: In order to prove this we need a few intermediate results concerning the policy iteration scheme and RLS estimation. The results are summarized below and the proofs are given in the appendix. First, define the function

$$\sigma(U_k) = \mathrm{trace}(K_{U_k}). \tag{12}$$

Lemma 1. *If $\{A, B\}$ is controllable, U_1 stabilizing with associated cost matrix K_1 and U_2 is the result of one policy improvement step from U_1, i.e. $U_2 = -\gamma(F + \gamma B'K_1B)^{-1}B'K_1A$, then*

$$\Delta\|U_1 - U_2\|^2 \le \sigma(U_1) - \sigma(U_2) \le \delta\|U_1 - U_2\|^2,$$

where

$$0 < \Delta = \underline{\sigma}(F) \le \delta = \mathrm{trace}\,(F + \gamma B'K_1B)\|\sum_{i=0}^{\infty}\gamma^{(i/2)}(A + BU_2)^i\|^2,$$

and $\underline{\sigma}(\cdot)$ denotes the minimum singular value of a matrix.

Lemma 2. *If ϕ_t is persistently excited as given by inequaliy (11) and $N \ge N_0$, then we have*

$$\|\theta_k - \hat{\theta}_k\| \le \epsilon_N(\|\theta_k - \theta_{k-1}\| + \|\theta_{k-1} - \hat{\theta}_{k-1}\|), \qquad \text{where } \epsilon_N = \frac{1}{\epsilon_0 N p_0}$$

and p_0 is the minimum singular value of P_0.

Define a scalar "Lyapunov" function candidate

$$s_k = \sigma(U_{k-1}) + \|\theta_{k-2} - \hat{\theta}_{k-2}\| \tag{13}$$

and suppose that

$$s_i \le \bar{s}_0 < \infty \qquad \text{for all } 0 \le i \le k \tag{14}$$

for some upper bound $\bar{s}_0$. From this it follows that U_{k-1} is stabilizing in the sense that

$$\sigma(U_{k-1}) \le \bar{s}_0 \tag{15}$$

and that the parameter estimation error is bounded so that

$$\|\theta_{k-2} - \hat{\theta}_{k-2}\| \le \bar{s}_0. \tag{16}$$

It also follows that that the control resulting from a policy update using accurate parameters, U_k^*, is stabilizing and that $\sigma(U_k^*) \le \bar{s}_0$. From continuity of the optimal policy update it then follows that for every $\delta > 0$ there exists $\epsilon_\delta > 0$ so that

$$|\sigma(U) - \sigma(U_k^*)| \le \delta\|U_k^* - U\| \qquad \text{for all } \|U_k^* - U\| \le \epsilon_\delta. \tag{17}$$

This implies that control laws in a sufficiently small neighborhood around the optimal are stabilizing as well.

We will show that $s_{k+1} \leq s_k$ provided that the estimation interval N, is chosen to be long enough.

Define

$$v_k = \|\theta_{k-1} - \hat{\theta}_{k-1}\|,$$

and we get from Lemma 2 that for all k

$$v_k \leq \epsilon_N(v_{k-1} + \|\theta_{k-1} - \theta_{k-2}\|), \tag{18}$$

where $\lim_{N\to\infty} \epsilon_N = 0$. Now from the inductive hypothesis (assumption (14)) we have

$$v_{k-1} \leq \bar{s}_0 \qquad \text{and} \qquad \|\theta_{k-2} - \theta_{k-3}\| \leq \kappa_1, \tag{19}$$

where κ_1 is a constant. By application of (18) we then get

$$v_k \leq \epsilon_N(\bar{s}_0 + \kappa_1). \tag{20}$$

It follows that $v_k = \|\theta_{k-1} - \hat{\theta}_{k-1}\|$ can be made arbitrarily small be choosing the estimation interval N long enough.

U_k^* is defined to be the result from applying one step of policy iteration using accurate parameter values, *i.e.*

$$U_k^* = -H_{\text{k-1}(22)}^{-1} H_{\text{k-1}(21)}, \tag{21}$$

whereas U_k is the feedback law which results from applying the estimated parameters, *i.e.*

$$U_k = -\hat{H}_{\text{k-1}(22)}^{-1} \hat{H}_{\text{k-1}(21)}. \tag{22}$$

The matrix inverse is guaranteed to exist when the estimation interval is long enough. From equations (21) and (22) we now have

$$U_k - U_k^* = -\hat{H}_{\text{k-1}(22)}^{-1} \hat{H}_{\text{k-1}(21)} + H_{\text{k-1}(22)}^{-1} H_{\text{k-1}(21)}.$$

Hence

$$\begin{aligned} U_k - U_k^* &= H_{\text{k-1}(22)}^{-1}(H_{\text{k-1}(21)} - \hat{H}_{\text{k-1}(21)}) + (H_{\text{k-1}(22)}^{-1} - \hat{H}_{\text{k-1}(21)}^{-1})\hat{H}_{\text{k-1}(21)} \\ &= H_{\text{k-1}(22)}^{-1}((H_{\text{k-1}(21)} - \hat{H}_{\text{k-1}(21)}) + (\hat{H}_{\text{k-1}(22)} - H_{\text{k-1}(22)})\hat{H}_{\text{k-1}(22)}^{-1} \hat{H}_{\text{k-1}(21)}). \end{aligned}$$

From the definition of θ we have

$$\|\hat{H}_{\text{k-1}(22)} - H_{\text{k-1}(22)}\| \leq \|\theta_{k-1} - \hat{\theta}_{k-1}\| \qquad \text{and} \qquad \|\hat{H}_{\text{k-1}(22)}\| \leq \|\hat{\theta}_{k-1}\|.$$

It follows that we have

$$\|U_k - U_k^*\| \leq \bar{\kappa}_0(1 + \|\hat{\theta}_{k-1}\|) \cdot \|\theta_{k-1} - \hat{\theta}_{k-1}\|,$$

where $\bar{\kappa}_0$ is a positive constant, provided that N is sufficiently large. Since the estimated parameters are bounded it follows that there exists another constant κ_0 so that

$$\|U_k - U_k^*\| \leq \kappa_0\|\theta_{k-1} - \hat{\theta}_{k-1}\| = \kappa_0 v_k. \tag{23}$$

It follows from equation (20) that we have

$$\|U_k - U_k^*\| \le \epsilon_N \kappa_0(\bar{s}_0 + \kappa_1). \tag{24}$$

It then follows from (17) that

$$|\sigma(U_k) - \sigma(U_k^*)| \le \delta \|U_k^* - U_k\| \qquad \text{for all } N \text{ such that } \epsilon_N \kappa_0(\bar{s}_0 + \kappa_1) \le \epsilon_\delta.$$

This implies that U_k is stabilizing if N is large enough and that there exists an integer N_1 and an associated constant $\bar{\delta}$, so that

$$|\sigma(U_k) - \sigma(U_{k-1})| \le \bar{\delta} \|U_k - U_{k-1}\| \qquad \text{for all } N \ge N_1.$$

In other words, if the estimation interval is long enough, then the difference between two consecutive costs is bounded by the difference between two consecutive controls. We use the definition of the parameter estimation vector to write this as

$$\|\theta_{k-1} - \theta_{k-2}\| \le \delta_1 \|U_k - U_{k-1}\|^2 \qquad \text{for all } N \ge N_1, \tag{25}$$

where δ_1 is a constant. We now re-write (25) as

$$\|\theta_{k-1} - \theta_{k-2}\| \le 2\delta_1(\|U_k^* - U_{k-1}\|^2 + \|U_k^* - U_k\|^2).$$

From inequality (23) and the definition of v_k, we then get

$$\|\theta_{k-1} - \theta_{k-2}\| \le 2\delta_1(w_k^2 + \kappa_0 v_k), \tag{26}$$

where

$$w_k = \|U_k^* - U_{k-1}\|.$$

By combining equations (18) and (26) we then get

$$v_k \le \epsilon_N(v_{k-1} + 2\delta_1(w_k^2 + \kappa_0 v_k)),$$

which we re-write as

$$v_k \le \epsilon_N \mu_N (v_{k-1} + 2\delta_1 w_k^2), \tag{27}$$

where

$$\mu_N = (1 - 2\delta_1 \kappa_0 \epsilon_N)^{-1}.$$

According to the assumption we can choose N large enough so that $0 < \mu_N < \infty$. This gives a recursion for v_k. The critical point to notice is that v_k has a strong stability property when the estimation interval is long. The parameter $\epsilon_N \mu$ is then small since ϵ_N converges uniformly to 0 and μ towards 1.

We now develop the recursion for $\sigma(U_k)$. First we have

$$\sigma(U_k) - \sigma(U_{k-1}) = \sigma(U_k^*) - \sigma(U_{k-1}) + \sigma(U_k) - \sigma(U_k^*). \tag{28}$$

From equation (28) and Lemma 1, using (17) again, it follows that we can choose the update interval so that we have a constant δ_2 so that

$$\sigma(U_k) - \sigma(U_{k-1}) \le -\Delta \|U_k^* - U_{k-1}\|^2 + \delta_2 \|U_k^* - U_k\|^2.$$

Using equation (23) we then get

$$\begin{aligned}\sigma(U_k) - \sigma(U_{k-1}) &\leq -\Delta\|U_k^* - U_{k-1}\|^2 + \delta_2\kappa_0\|\theta_{k-1} - \hat{\theta}_{k-1}\| \\ &\leq -\Delta w_k^2 + \delta_2\kappa_0 v_k.\end{aligned}$$

By using equation (27) and the recursion for v_k we then have

$$\sigma(U_k) - \sigma(U_{k-1}) \leq -\Delta w_k^2 + \delta_1\kappa_0\epsilon_N\mu_N(v_{k-1} + 2\delta_2 w_k^2). \tag{29}$$

Equations (27) and (29) together define the system

$$\begin{bmatrix} v_k \\ \sigma(U_k) \end{bmatrix} = \begin{bmatrix} \epsilon_N\mu_N & 0 \\ \delta_2\kappa_0\epsilon_N\mu_N & 1 \end{bmatrix} \begin{bmatrix} v_{k-1} \\ \sigma(U_{k-1}) \end{bmatrix} + \begin{bmatrix} 2\epsilon_N\mu_N\delta_2 \\ -\Delta + 2\delta_2\kappa_0\epsilon_N\mu_N \end{bmatrix} w_k^2.$$

In order to study this system we defined the function

$$s_k = \sigma(U_{k-1}) + v_{k-1}.$$

From the above we then have

$$s_{k+1} = s_k + (-1 + \epsilon_N\mu_N(1 + \delta_2\kappa_0))v_{k-1} + (-\Delta + 2\epsilon_N\mu_N\delta_2(1 + \kappa_0))w_k^2.$$

It now suffices to choose N so that ϵ is small enough to give

$$\begin{aligned}1 - \epsilon_N\mu_N(1 + \delta_2\kappa_0) &= \epsilon_1 > 0 \\ \Delta - 2\epsilon_N\mu_N\delta_2(1 + \kappa_0) &= \epsilon_2 > 0.\end{aligned}$$

We then get

$$s_{k+1} = s_k - \epsilon_1 v_{k-1} - \epsilon_2 w_k^2 \leq s_k.$$

From this we conclude that $s_{k+1} \leq s_k$ and using induction we finally have

$$\epsilon_1 \sum_{k=1}^{\infty} v_k \leq \bar{s}_0 \qquad \text{and } \epsilon_2 \sum_{k=1}^{\infty} w_k^2 \leq \bar{s}_0.$$

The result now follows since U_0 is stabilizing.

6 Simulation results

Figure 2 demonstrates the performance of the adaptive policy iteration algorithm based on Q-functions. We used a random exploration signal generated from a normal distribution in order to induce persistent excitation. This has worked very well in practice. The demonstration system is a 20-dimensional discrete-time approximation of a Euler-Bernoulli flexible beam supported at both ends. There is one control point. The scalar control signal is the acceleration applied at that point. U_0 is an arbitrarily selected stabilizing controller for the system. x_0 is a random point in a neighborhood around $0 \in \mathcal{R}^{20}$. There are 231 parameters to be estimated for this system, so we set $N = 500$, approximately twice that.

Panel A of Figure 2 shows the norm of the difference between the current controller and the optimal controller. Panel B of Figure 2 shows the norm of the difference between the estimate of the Q-function parameters for the current controller and the Q-function parameters for the optimal controller. After only eight policy iteration steps the adaptive policy iteration algorithm has converged close enough to U^* and H^* that further improvements are limited by the machine precision. Although this demonstration is for a single-input system, the algorithm performs equally well on multi-input systems.

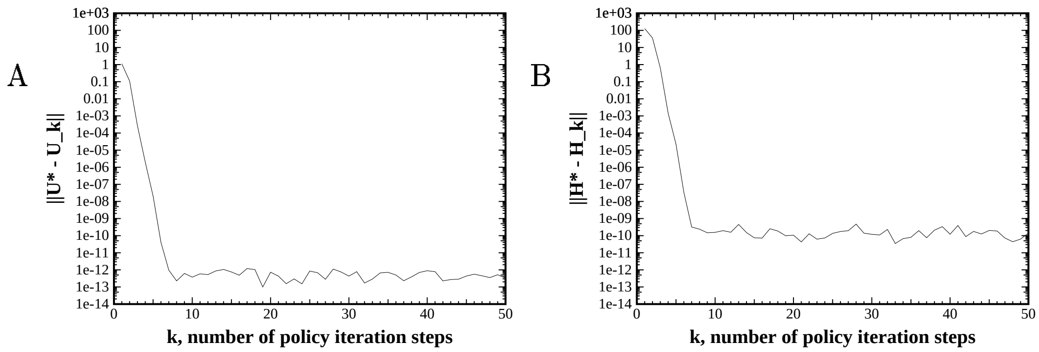

Figure 2: Performance of the adaptive policy iteration algorithm on a discretized beam system.

Figure 3 demonstrates that the adaptive policy iteration algorithm can fail when the assumptions of the convergence theorem are violated. That is, when either persistent excitation is not maintained, or when the estimation interval, N, is too short. The demonstration system is the same discretized beam used above. Panel A shows the results of violating the persistent excitation assumption. As in the experiment described above, policy improvement steps were performed every 500 time steps. However, the exploratory signal was a constant zero, so ϕ_t was not persistently excited. The graph in panel A shows the size of $\|x_t\|_\infty$ growing rapidly to infinity after the first policy "improvement" step at time 500. The lack of persistent excitation prevented $\hat{H}_{U_1}$ from being an adequate approximation to H_{U_1}, causing the "improved" controller, U_2, to be destabilizing. Panel B shows the results of a too short estimation interval. In this experiment, policy improvement was performed every 100 time-steps instead of every 500 time-steps. Since there are 231 parameters to be estimated the estimator could not have formed a good approximation to all of them. The graph shows that the controller that resulted from the first policy "improvement" step was destabilizing in this situation also.

7 Conclusions

In this paper we take a first step toward extending the theory of DP-based reinforcement learning to domains with continuous state and action spaces, and to algorithms that use non-linear function approximators. We concentrate on the problem of Linear Quadratic Regulation. We describe a policy iteration algorithm for LQR problems that is proven to

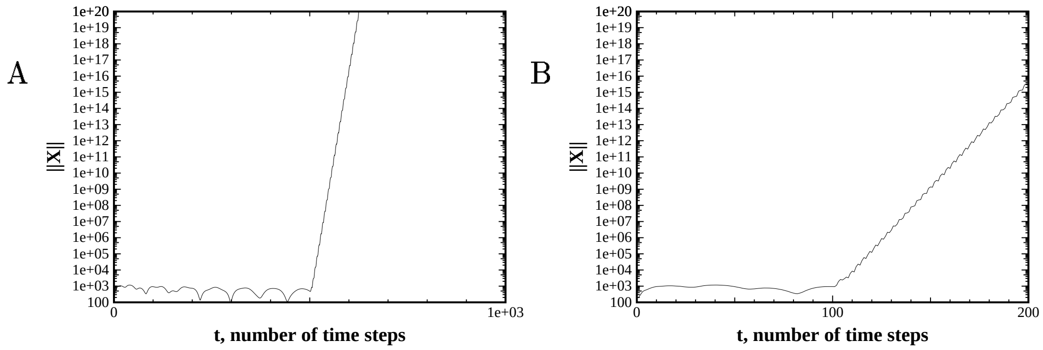

Figure 3: Performance of the adaptive policy iteration algorithm on a discretized beam system when either (A) the persistent excitation condition, or (B) the estimation interval conditions are violated.

converge to the optimal policy. In contrast to standard methods of policy iteration, it does not require a system model. It only requires a suitably accurate estimate of H_{U_k}. This is the first result of which we are aware showing convergence of a DP-based reinforcement learning algorithm in a domain with continuous states and actions.

The convergence proof for the policy iteration algorithm described in this paper requires exact matching between the form of the Q-function for LQR problems and the form of the function approximator used to learn that function. Future work will explore convergence of DP-based reinforcement learning algorithms when applied to non-linear systems for which the form of the Q-functions is unknown. It will be necessary in such cases to use more general function approximation techniques, such as multilayer perceptrons.

References

[1] D. P. Bertsekas. *Dynamic Programming: Deterministic and Stochastic Models.* Prentice Hall, Englewood Cliffs, NJ, 1987.

[2] P. Dayan. The convergence of TD(λ) for general λ. *Machine Learning*, 8:341–362, 1992.

[3] E. V. Denardo. Contraction mappings in the theory underlying dynamic programming. *SIAM Review*, 9(2):165–177, April 1967.

[4] G. C. Goodwin and K. S. Sin. *Adaptive filtering prediction and control.* Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1984.

[5] R. A. Howard. *Dynamic Programming and Markov Processes.* John Wiley & Sons, Inc., New York, 1960.

[6] T. Jaakkola, M. I. Jordan, and S. P. Singh. Stochastic convergence of iterative dp algorithms. In *Proceedings of the Conference on Neural Information Processing Systems — Natural and Synthetic*, 1993. Accepted.

[7] D. L. Kleinman. On an iterative technique for Riccati equation computations. *IEEE Transactions on Automatic Control*, pages 114–115, February 1968.

[8] D. A. Sofge and D. A. White. Neural network based process optimization and control. In *Proceedings of the 29th Conference on Decision and Control*, Honolulu, Hawaii, December 1990.

[9] R. S. Sutton. Learning to predict by the method of temporal differences. *Machine Learning*, 3:9–44, 1988.

[10] G. J. Tesauro. Practical issues in temporal difference learning. *Machine Learning*, 8(3/4):257–277, May 1992.

[11] J. N. Tsitsiklis. Asynchronous stochastic approximation and Q-learning. Technical Report LIDS-P-2172, Laboratory for Information and Decision Systems, MIT, Cambridge, MA, 1993.

[12] C. J. C. H. Watkins. *Learning from Delayed Rewards.* PhD thesis, Cambridge University, Cambridge, England, 1989.

[13] C. J. C. H. Watkins and P. Dayan. Q-learning. *Machine Learning*, 8(3/4):257–277, May 1992.

[14] P. J. Werbos. Building and understanding adaptive systems: A statistical/numerical approach to factory automation and brain research. *IEEE Transactions on Systems, Man, and Cybernetics*, 17(1):7–20, 1987.

A Proof of lemma 1

Our proof of Lemma 1 requires some preliminary definitions and three subsidiary lemmata A1 through A3.

Let U_1 be a stabilizing controller for this system, and let K_1 be the associated cost matrix. Let U_2 be the result of performing the policy improvement algorithm on U_1, *i.e.*,

$$U_2 = -\gamma(F + \gamma B'K_1B)^{-1}B'K_1A. \tag{30}$$

Let K_2 be the cost matrix associated with U_2. Define $A_1 = A + BU_1$, and $A_2 = A + BU_2$.

We know [1] that the cost matrix K_{U} for a given control matrix U satisfies the equations

$$K_{\text{U}} = E + U'FU + \gamma\,(A + BU)'\,K_{\text{U}}\,(A + BU)\,, \tag{31}$$

and

$$K_{\text{U}} = \sum_{i=0}^{\infty}\gamma^i(A + BU)^{i\prime}(E + U'FU)(A + BU)^i. \tag{32}$$

Lemma A1. *If $\{A, B\}$ is controllable, U_1 is stabilizing, and $U_2 = -\gamma(F + \gamma B'K_1B)^{-1}B'K_1A$, then*

$$K_1 - K_2 = \sum_{i=0}^{\infty}\gamma^i A_2^{i\prime}\left[(U_1 - U_2)'(F + \gamma B'K_1B)(U_1 - U_2)\right]A_2^i,$$

where $A_1 = A + BU_1$ and $A_2 = A + BU_2$.

Proof: First, rewrite K_1 as:

$$\begin{aligned}
K_1 &= K_1 + \sum_{i=1}^{\infty}(\gamma^i A_2^{i\prime}K_1A_2^i) - \sum_{i=1}^{\infty}(\gamma^i A_2^{i\prime}K_1A_2^i)\\
&= \sum_{i=0}^{\infty}(\gamma^i A_2^{i\prime}K_1A_2^i) - \sum_{i=1}^{\infty}(\gamma^i A_2^{i\prime}K_1A_2^i)\\
&= \sum_{i=0}^{\infty}(\gamma^i A_2^{i\prime}K_1A_2^i) - \sum_{i=0}^{\infty}(\gamma^i A_2^{i\prime}[\gamma A_2'K_1A_2]A_2^i)\\
&= \sum_{i=0}^{\infty}\gamma^i A_2^{i\prime}[K_1 - \gamma A_2'K_1A_2]A_2^i
\end{aligned} \tag{33}$$

Combining equations (32) and (33) we get

$$K_1 - K_2 = \sum_{i=0}^{\infty}\gamma^i A_2^{i\prime}\left[K_1 - \gamma A_2'K_1A_2 - E - U_2'FU_2\right]A_2^i. \tag{34}$$

Let us define $D = [K_1 - \gamma A_2'K_1A_2 - E - U_2'FU_2]$.

Substituting from equation (31) into the definition of D, we get

$$\begin{aligned}
D &= E + U_1'FU_1 + \gamma A_1'K_1A_1 - \gamma A_2'K_1A_2 - E - U_2'FU_2\\
&= U_1'FU_1 + \gamma A_1'K_1A_1 - \gamma A_2'K_1A_2 - U_2'FU_2.
\end{aligned}$$

Expanding $A_2{}'K_1A_2$ yields

$$\begin{aligned}
D &= U_1{}'FU_1 + \gamma A_1{}'K_1A_1 \\
&\quad -\gamma A'K_1A - \gamma A'K_1BU_2 - \gamma U_2{}'B'K_1A - \gamma U_2{}'B'K_1BU_2 \\
&\quad -U_2{}'FU_2 \\
&= U_1{}'FU_1 + \gamma A_1{}'K_1A_1 - \gamma A'K_1A \\
&\quad -\gamma A'K_1BU_2 - \gamma U_2{}'B'K_1A \\
&\quad -U_2{}'(F + \gamma B'K_1B)U_2.
\end{aligned}$$

Using the definition of U_2 from equation (30) now gives

$$\begin{aligned}
D &= U_1{}'FU_1 + \gamma A_1{}'K_1A_1 - \gamma A'K_1A \\
&\quad +U_2{}'(F + \gamma B'K_1B)U_2 + U_2{}'(F + \gamma B'K_1B)U_2 \\
&\quad -U_2{}'(F + \gamma B'K_1B)U_2 \\
&= U_1{}'FU_1 + \gamma A_1{}'K_1A_1 - \gamma A'K_1A + U_2{}'(F + \gamma B'K_1B)U_2.
\end{aligned}$$

Finally, expanding $A_1{}'K_1A_1$ and again using the definition of U_2 leads to

$$\begin{aligned}
D &= U_1{}'FU_1 \\
&\quad +\gamma A'K_1A + \gamma A'K_1BU_1 + \gamma U_1{}'B'K_1A + \gamma U_1{}'B'K_1BU_1 \\
&\quad -\gamma A'K_1A \\
&\quad +U_2{}'(F + \gamma B'K_1B)U_2 \\
&= U_1{}'(F + \gamma B'K_1B)U_1 \\
&\quad +\gamma A'K_1BU_1 + \gamma U_1{}'B'K_1A \\
&\quad +U_2{}'(F + \gamma B'K_1B)U_2 \\
&= U_1{}'(F + \gamma B'K_1B)U_1 - U_2{}'(F + \gamma B'K_1B)U_1 \\
&\quad -U_1{}'(F + \gamma B'K_1B)U_2 + U_2{}'(F + \gamma B'K_1B)U_2 \\
&= (U_1 - U_2)'(F + \gamma B'K_1B)(U_1 - U_2).
\end{aligned}$$

Substitute this final expression for D back into equation (34) to get the desired result

$$K_1 - K_2 = \sum_{i=0}^{\infty} \gamma^i A_2^{i\,\prime} \left[(U_1 - U_2)'(F + \gamma B'K_1B)(U_1 - U_2)\right] A_2^i.$$

Lemma A2. *If $\{A, B\}$ is controllable, U_1 is stabilizing, and $U_2 = -\gamma(F + \gamma B'K_1B)^{-1}B'K_1A$, then*

$$\sigma(U_1) - \sigma(U_2) \geq \Delta\|U_1 - U_2\|^2.$$

where $0 < \Delta = \underline{\sigma}(F)$.

Proof: By Lemma A1, we know that

$$\begin{aligned}
K_1 - K_2 &= \sum_{i=0}^{\infty} \gamma^i A_2^{i\,\prime} \left[(U_1 - U_2)'(F + \gamma B'K_1B)(U_1 - U_2)\right] A_2^i \\
&\geq (U_1 - U_2)'F(U_1 - U_2),
\end{aligned}$$

since all of the summands are positive.

Taking the trace of both sides we get

$$\begin{aligned}\operatorname{trace}(K_1) - \operatorname{trace}(K_2) &= \operatorname{trace}(K_1 - K_2)\\ &\geq \operatorname{trace}((U_1 - U_2)'(F + \gamma B'K_1B)(U_1 - U_2))\\ &\geq \underline{\sigma}(F)\|U_1 - U_2\|^2.\end{aligned}$$

Noting that F is positive definite and substituting from the definitions of $\sigma(U_1)$ and $\sigma(U_2)$ gives us the final result

$$0 < \sigma(U_1) - \sigma(U_2) \geq \Delta\|U_1 - U_2\|^2.$$

Lemma A3. *If $\{A, B\}$ is controllable, U_1 is stabilizing, and $U_2 = -\gamma(F + \gamma B'K_1B)^{-1}B'K_1A$, then*

$$\sigma(U_1) - \sigma(U_2) \leq \delta\|U_1 - U_2\|^2.$$

where $0 < \delta = \operatorname{trace}(F + \gamma B'K_1B)\|G\|^2$, $G = \left(\sum_{i=0}^{\infty} \gamma^{(i/2)} A_2^i\right)$, $A_1 = A + BU_1$, and $A_2 = A + BU_2$.

Proof: By Lemma A1, we know that

$$\begin{aligned}K_1 - K_2 &= \sum_{i=0}^{\infty} \gamma^i A_2^{i\prime}\left[(U_1 - U_2)'(F + \gamma B'K_1B)(U_1 - U_2)\right] A_2^i\\ &\leq \left(\sum_{i=0}^{\infty} \gamma^{(i/2)} A_2^{i\prime}\right)(U_1 - U_2)'(F + \gamma B'K_1B)(U_1 - U_2)\left(\sum_{i=0}^{\infty} \gamma^{(i/2)} A_2^{i}\right)\\ &= G'(U_1 - U_2)'(F + \gamma B'K_1B)(U_1 - U_2)G.\end{aligned}$$

Taking the trace of both sides we get

$$\begin{aligned}\operatorname{trace}(K_1) - \operatorname{trace}(K_2) &= \operatorname{trace}(K_1 - K_2)\\ &\leq \operatorname{trace}(G'(U_1 - U_2)'(F + \gamma B'K_1B)(U_1 - U_2)G)\\ &\leq \operatorname{trace}(F + \gamma B'K_1B)\|G\|^2\|U_1 - U_2\|^2.\end{aligned}$$

Noting that $(F + \gamma B'K_1B)$ is positive definite and substituting from the definitions of $\sigma(U_1)$ and $\sigma(U_2)$ gives us the final result

$$0 < \sigma(U_1) - \sigma(U_2) \leq \delta\|U_1 - U_2\|^2.$$

Lemma 1. *If $\{A, B\}$ is controllable, U_1 stabilizing with associated cost matrix K_1 and U_2 is the result of one policy improvement step from U_1, i.e. $U_2 = -\gamma(F + \gamma B'K_1B)^{-1}B'K_1A$, then*

$$\Delta\|U_1 - U_2\|^2 \leq \sigma(U_1) - \sigma(U_2) \leq \delta\|U_1 - U_2\|^2,$$

where

$$0 < \Delta = \underline{\sigma}(F) \leq \delta = \operatorname{trace}(F + \gamma B'K_1B)\|\sum_{i=0}^{\infty} \gamma^{(i/2)}(A + BU_2)^i\|^2,$$

and $\underline{\sigma}(\cdot)$ denotes the minimum singular value of a matrix.

Proof: Lemma 1 follows from Lemmata A2 and A3.

B Proof of lemma 2

Lemma 2. *If ϕ_t is persistently excited as given by inequality (11) and $N \geq N_0$, then we have*

$$\|\theta_k - \hat{\theta}_k\| \leq \epsilon_N(\|\theta_k - \theta_{k-1}\| + \|\theta_{k-1} - \hat{\theta}_{k-1}\|), \qquad \textit{where } \epsilon_N = \frac{1}{\epsilon_0 N p_0}$$

and p_0 is the minimum singular value of P_0.

Proof: Let us consider the k^{th} estimation interval. $\theta_k = \Theta(H_{\mathrm{U}_k})$, the true vector of parameters for the function Q_{U_k}. $\hat{\theta}_k = \hat{\theta}_k(N)$ is the estimate of θ_k at the end of the k^{th} estimation interval. The parameter estimates are initialized for the k^{th} estimation interval with the final values from the previous estimation interval, *i.e.*, $\hat{\theta}_k(0) = \hat{\theta}_{k-1}$. The RLS algorithm is initialized at the start of the k^{th} estimation interval by setting the inverse covariance matrix $P_k(0) = P_0$, and setting the initial parameter estimates to the final values from the previous interval, *i.e.*, $\hat{\theta}_k(0) = \hat{\theta}_{k-1}$. Define $\tilde{\theta}_k(i) = \hat{\theta}_k(i) - \theta_k$. Then following Goodwin and Sin [4] we have

$$\tilde{\theta}_k(i) = P_k(i)P_k(i-1)^{-1}\tilde{\theta}_k(i-1)$$

for all $i > 0$. Applying this relation recursively results in

$$\tilde{\theta}_k(i) = P_k(i)P_k(0)^{-1}\tilde{\theta}_k(0).$$

Taking the norms of both sides we have

$$\begin{aligned} \|\tilde{\theta}_k(i)\| &= \|P_k(i)P_k(0)^{-1}\tilde{\theta}_k(0)\| \\ &\leq \|P_k(i)\| \cdot \|P_k(0)^{-1}\| \cdot \|\tilde{\theta}_k(0)\|. \end{aligned} \tag{35}$$

Now, $P_k(i)^{-1} = P_k(0)^{-1} + \sum_{i=1}^{N} \phi_k(i)\phi_k(i)'$. Therefore,

$$\begin{aligned} \|P_k(i)^{-1}\| &= \|P_k(0)^{-1} + \sum_{i=1}^{N} \phi_k(i)\phi_k(i)'\| \\ &\geq \|\sum_{i=1}^{N} \phi_k(i)\phi_k(i)'\| \\ &\geq N\epsilon_0 I \end{aligned} \tag{36}$$

We also know that

$$\|P_k(0)^{-1}\| = \frac{1}{p_0}. \tag{37}$$

Substituting (36) and (37) into (35) and using the definition of ϵ_N yields

$$\begin{aligned} \|\theta_k - \hat{\theta}_k(i)\| &= \|\tilde{\theta}_k(i)\| \\ &\leq \epsilon_N\|\tilde{\theta}_k(0)\| \\ &= \epsilon_N\|\hat{\theta}_k(0) - \theta_k\| \\ &= \epsilon_N\|\hat{\theta}_{k-1} - \theta_k\| \\ &= \epsilon_N\|\hat{\theta}_{k-1} - \theta_k + \theta_{k-1} - \theta_{k-1}\| \\ &\leq \epsilon_N(\|\theta_{k-1} - \hat{\theta}_{k-1}\| + \|\theta_k - \theta_{k-1}\|), \end{aligned}$$

and we have the desired result.