Proceedings of the American Control Conference
Arlington, VA June 25-27, 2001

# Semiglobal nonlinear stabilization via Approximate Policy Iteration

C. I. Boussios, M. A. Dahleh, and J. N. Tsitsiklis
Massachusetts Institute of Technology
Cambridge, MA 02139

**Abstract**

We consider the problem of semiglobal nonlinear stabilization. Based on a given unstable dynamic system and a region of acceptable operation within which the state is desired to be confined, we define an appropriate alternative dynamic system. We define an optimal control problem for the alternative (redefined) system which is amenable to solution via Approximate Policy Iteration (a computational design procedure for suboptimal control design). We show that the optimal controller for the alternative problem is stabilizing for the original system, provided that the latter is stabilizable. It follows that the suboptimal controllers designed via approximate policy iteration are stabilizing for the original system, for sufficiently small approximation errors.[1]

## 1 Introduction

Instability is the most serious problem facing a control engineer. In practice, a system becomes unstable when a trajectory of the system starting from a non-equilibrium point grows out of some finite bounds. Exceeding such bounds is unacceptable for a device. The bounds may represent some kind of worst case performance specification limits or, even more importantly, some safety limits (e.g. an aircraft angle of attack exceeding the stall angle threshold, thus entering a situation from which there is not enough actuator power to recover, with possibly disastrous consequences).

Often, the mathematical models used to describe the dynamics of a device do not recognize these limitations. They are based on the system's dynamic behavior inside the limits of acceptable operation, without accounting for the state magnitude physical limitations. Instability for these models amounts to their trajectories starting from a non-equilibrium point growing unbounded. Such models may have several advantages. A potential advantage is linearity. There exists a variety of control design tools for linear models. If a system can be modeled by a linear model with a high degree of accuracy inside the region of acceptable operation, then good controllers can be designed via linear control design tools which ensure that the dynamic variables of the system will never exceed the safety limitations. In that case, not including the limitations in the model, makes sure that the design task is relatively straightforward, and the designed controllers are such that the closed loop system does indeed operate desirably, within the specification or safety limits. Even for systems that we model as nonlinear, the model is simpler and "less nonlinear" by not including these limitations, thus, making the use of analytical approaches for the control design problem easier.

It turns out, however, that taking into account the limits of acceptable operation may be beneficial for nonlinear control design in many cases. Control over a *selected* bounded region around the origin is known as *semiglobal* control [6, 4]. Computational nonlinear control design methods, where the feedback control law has to be computed at each point of the state space, are necessarily semiglobal [5].

In this article, we assume an unstable discrete-time nonlinear dynamic system. Given the region of interest for stable control design, we formulate an appropriate optimal control problem. It is shown that, if the system is stabilizable, the optimal controller, or policy[2] is stabilizing. Although this optimal control problem is amenable to solution via Approximate Policy Iteration. This is a computationally feasible relaxed version of Policy Iteration, a dynamic programming algorithm [1] which is computationally intractable for Euclidean state spaces. These are iterative methods that produce a sequence of controllers based on off-line simulation of the dynamic system. Policy Iteration (exact) produces monotonically improving and converging to the optimal [2] controller sequences. For small enough approximation errors, Approximate Policy Iteration produces a sequence of improving controllers [3].

[1]This research was partially supported by the AFOSR under contract F49620-95-1-0219 and the NSF under contracts DMI-9625489 and ECS-9612558

[2]The terms **policy** and **controller** are used interchangeably as synonyms throughout the article, both defined as a mapping from the state to the control input. A policy or controller is denoted by the letter $\mu$.

In Section 2, we pose the stabilization problem. We then introduce a redefinition of the system dynamics and pose a discounted optimal control problem. In Section 3 we show that the optimal controller for the redefined problem is stabilizing. In Sections 4 and 5 we describe Policy Iteration, and Approximate Policy Iteration. Finally, in Sections 6 and 7 we discuss implementation and an application of the algorithm

## 2 Problem Formulation

We are given a model

$$x_{t+1} = f(x_t) + G(x_t)u, \quad x_0 \in \mathbf{R}^m, u \in U \subset \mathbf{R}^m \quad (1)$$

of a nonlinear system in discrete time, where U is a bounded region that includes 0. We assume that there exists no known controller which stabilizes the above model. The discrete time model may be derived from a continuous time model of the form $\dot{x} = f_c(x) + G_c(x)u$. The requirement to specify a bounded region $U$ that $u$ belongs to, is needed for the theoretical guarantees of the approach to be developed in the sequel of this chapter. We are free to select $U$ as large as we desire, but bounded. This is by all means a realistic assumption from a practical point of view, too, as in all real life applications such bounds exist. The origin 0 is an unstable equilibrium point of the open loop system. The trajectories of (1) with $u = 0$ starting from almost any nonzero state $x_0$ at time 0 grow unbounded:

$$\lim_{t\to\infty} \|x_t\| = \infty, \qquad \text{for } \textit{almost all} \quad x_0 \neq 0 \quad (2)$$

The *training region* TR, which includes the origin, is specified by the control engineer. It is the region for which a stable controller is desirable; that is, a controller $\mu(x)$ such that any trajectory of the closed loop system

$$x_{t+1} = f(x_t) + G(x_t)\mu(x_t), \qquad x_0 \in \text{TR} \quad (3)$$

starting off from a point $x_0$ in TR will remain bounded, or converge to the origin. The control engineer also specifies the *region of acceptable operation* RAO, such that TR $\subset$ RAO. Both regions TR and RAO are compact, that is, closed and bounded. For example, they may be of the form

$$\begin{aligned} \text{TR} &= \{x : \|x\| \leq R_{TR}\} \\ \text{RAO} &= \{x : \|x\| \leq R_{RAO}\} \\ 0 &< R_{TR} \leq R_{RAO} \end{aligned} \quad (4)$$

In that case, the surface $\{x : \|x\| = R_{TR}\}$ ($\{x : \|x\| = R_{RAO}\}$) is the boundary of region TR (region RAO). See Figure 1. We now introduce the following *model* in order to describe the underlying physical process:

$$x_{t+1} = \begin{cases} f(x_t) + G(x_t)u & \text{if } x_t \in RAO \\ x_t & \text{otherwise} \end{cases} \quad (5)$$

As is argued in the remainder of the paper, adopting a model of this form allows us to define an optimal control problem that is amenable to solution via Approximate Dynamic Programming. Furthermore, it is shown that the resulting policies have desirable stabilizing properties on the original model (1)

Let $u = \mu(x)$ be any controller (policy). Consider a state $x_0 \in$ TR. Then, we denote by $x^{\mu}_{x_0}(t)$ the trajectory of (5) controlled by $\mu$ which starts off at $x_0$ at time $t = 0$. Thus, $x^{\mu}_{x_0}(0) = x_0$, and, for all $t$, $x^{\mu}_{x_0}(t+1)$ is given by

$$\begin{cases} f(x^{\mu}_{x_0}(t)) + G(x^{\mu}_{x_0}(t))\mu(x^{\mu}_{x_0}(t)), & \text{if } x^{\mu}_{x_0}(t) \in RAO \\ x^{\mu}_{x_0}(t), & \text{otherwise} \end{cases} \quad (6)$$

Notice that, in (6), we changed the notation in order to avoid subscript jamming: the dependence on the discrete time index appears in a parenthesis following the state vector, rather than subscripted as in (1).

**Definition 2.1** *Given a controller (policy) $\mu$ and a value of the state $x_0 \in$ TR, the instant of destabilization $N_\mu(x_0)$ with respect to RAO is the instant at which the trajectory $x^{\mu}_{x_0}$ leaves RAO. For example, if RAO is of the type RAO $= \{x : \|x\| \leq R_{RAO}\}$, then*

$$N_\mu(x_0) = \min\{t \in \mathbf{Z}^+ : \|x^{\mu}_{x_0}(t)\| > R_{RAO}\} \quad (7)$$

**Optimal Control Problem:** We state an optimal control problem for (5). The optimization objective is finding a policy $\mu^*$ which minimizes the cost function

$$\sum_{t=0}^{N_\mu(x_0)} \alpha^t \left(x^{\mu}_{x_0}(t)^T Q x^{\mu}_{x_0}(t) + \mu(x^{\mu}_{x_0}(t))^T R \mu(x^{\mu}_{x_0}(t))\right) + M \cdot \Big( \sum_{N_\mu(x_0)+1}^{\infty} \alpha^t \left(x^{\mu}_{x_0}(t)^T Q x^{\mu}_{x_0}(t) + \mu(x^{\mu}_{x_0}(t))^T R \mu(x^{\mu}_{x_0}(t))\right) \Big) \quad (8)$$

for every $x_0 \in$ TR, where $\alpha \in (0,1)$ is a cost *discount factor*, $M \geq 1$ is the *unacceptable operation weight factor*, $Q$ and $R$ are symmetric, positive definite state and control weight factors respectively.

**Remark:** Selecting $M$ to be large, is motivated by our desire to heavily penalize the unstable part of a trajectory so as to make the generated policies pay a high price for going out of the bounds imposed by RAO. As expected, it turns out to be an instrumental element of the stability proofs that follow in the next section (part II of Proposition 3.1).

We denote the cost function associated with a certain policy $\mu$, discount factor $\alpha$ and unacceptable operation weight factor $M$, by $J^{(\alpha,M)}_\mu$. It turns out that the cost function $J^{(\alpha,M)}_\mu$ is finite for every policy $\mu$:

**Proposition 2.1** *The cost function $J^{(\alpha,M)}_\mu : TR \to \mathbf{R}$ is finite, for trajectories $x^{\mu}_{x_0}(t)$ of the type (6).*

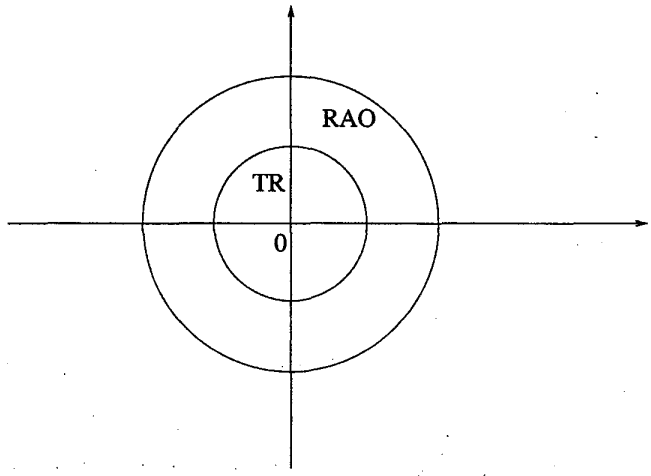


**Figure 1:** The regions RAO and TR

**Proof:**
Since RAO is a bounded region, it follows that $\|x^{\mu}_{x_0}(t)\|$ is upper bounded by a bound $x_{max}$, for all policy $\mu$ and initial conditions $x_0$. Similarly, since U is a bounded region all admissible controls $\mu(x)$ for all $x$ are upper bounded by a bound $u_{max}$. Since $\alpha \in (0,1)$ and $M \geq 1$ by choice, it can be easily shown that, for any $x_0$ and $\mu$, $J^{(\alpha,M)}_{\mu}(x_0)$ is upper bounded by

$$M(\lambda_{max}(Q)x^2_{max} + \lambda_{max}(R)u^2_{max})\frac{1}{1-\alpha}$$

where $\lambda_{max}(Q)$ ($\lambda_{max}(R)$) is the maximum eigenvalue of the positive definite matrix $Q$ ($R$). ■

Finiteness of the cost function for any controller/policy $\mu$ is the motivation behind modeling the unstable system in the form (5). It allows the use of policy iteration for solving the optimal control problem (8) (look Section 4 below). However, at this point it is important to notice that the optimal controller $\mu^*$ may not be stable for any $M$ and $\alpha$, in the sense of forcing the state to converge to the origin or simply remain close to it. We demonstrate this with the following example:
**Example:**
Consider the dynamic system

$$x_{t+1} = \begin{cases} 2x_t + u & \text{if } |x_t| \leq 2 \\ x_t & \text{otherwise} \end{cases}, \quad x, u \in \mathbf{R} \quad (9)$$

with training region $|x| \leq 2$. We consider an optimal control problem of the type (8) with $M = 1$, $\alpha \in (0,1)$, $Q = 1$ and $R = 1$. It is easy to realize that a policy $\mu$ cannot be stable unless it satisfies:

$$\mu(x) = \begin{cases} \leq -2x_t & \text{if } x_t \in [0,2] \\ \geq 2x_t & \text{if } x_t \in [-2,0) \end{cases} \quad (10)$$

Let $\mu_{st}$ be a stable policy. Consider the cost function:

$$\begin{aligned} J^{(\alpha,1)}_{\mu_{st}}(x_0) &= \sum_{t=0}^{\infty} \alpha^t \left(x^{\mu_{st}}_{x_0}(t)^2 + \mu_{st}(x^{\mu_{st}}_{x_0}(t))^2\right) \\ &\geq x_0^2 + \mu_{st}(x_0)^2 = 5x_0^2, \end{aligned} \quad (11)$$

for any discount factor $\alpha \in (0,1)$. Furthermore, consider the policy $\mu_n(x) = 0$ for all $x \in [-2,2]$ which is not stabilizing. Then,

$$\begin{aligned} J^{(\alpha,1)}_{\mu_n}(x_0) &= \sum_{t=0}^{\infty} \alpha^t \left(x^{\mu_n}_{x_0}(t)^2 + \mu_n(x^{\mu_n}_{x_0}(t))^2\right) \\ &\leq \sum_{t=0}^{\infty} \alpha^t \cdot 2^2 = \frac{4}{1-\alpha} \end{aligned} \quad (12)$$

If $\alpha < 0.2$ and for all $|x_0| \geq 1$, then $J^{(\alpha,1)}_{\mu_n}(x_0) < 5 \leq J^{(\alpha,1)}_{\mu_{st}}(x_0)$. Thus, $\mu_n$ induces a lower cost than $\mu_{st}$ for $|x_0| \geq 1$. It follows that a stable policy $\mu_{st}$ cannot be optimal for $\alpha < 0.2$. Therefore, *in general, the solution of the optimal control problem (5), (8) need not be a stable policy.* The reason for this is the discount factor $\alpha$. If there was no discount in the cost, then the optimal policy would be stable. At the same time, the role of the discount factor is instrumental for the purpose of using the policy iteration algorithm for unstable systems.

In the next section, we argue that for any stabilizable system there are ranges of $\alpha$ (close to one, but smaller) and $M$ (large enough), assuring that the optimal policy for the problem (5), (8) *is indeed a stable policy.*

## 3 Discounted Problem and Stability of the Optimal Policy

We define notions of stability, exponential stability and an $\epsilon$-attractive policy for a system of the type (5):

**Definition 3.1** *A policy $\mu_s$ : RAO $\rightarrow \mathbf{R}$ is called an* **RAO-safe** *policy if, for any $x_0 \in$ TR, the trajectory $x^{\mu_s}_{x_0}(t)$ belongs to RAO for every $t \geq 0$.*

**Definition 3.2** *A RAO-safe policy $\mu_{es}$ : RAO $\rightarrow \mathbf{R}$ for system (5) is called an* **exponentially stable** *pol-*

*icy if there exist a positive $C$ and a $q \in (0,1)$ such that, for all $x_0 \in TR$,*

$$x_{x_0}^{\mu_{es}}(t)^T Q x_{x_0}^{\mu_{es}}(t) + \mu_{es}(x_{x_0}^{\mu_{es}}(t))^T R \mu_{es}(x_{x_0}^{\mu_{es}}(t)) \leq C x_0^T Q x_0 q^t, \quad (13)$$

**Definition 3.3** *The $Q$-**closed ball** $B_\epsilon$ with center at the origin of $\mathbf{R}^n$ and radius $\epsilon > 0$ is defined to be the set of all $x \in \mathbf{R}^n$ such that $x^T Q x \leq \epsilon$.*

**Definition 3.4** *A RAO-safe policy $\mu_{ba}$ is called $\epsilon$-**attractive** if every trajectory $x_{x_0}^{\mu_{ba}}(t)$ which starts off at $x_0 \in TR$ goes inside the $Q$-closed ball $B_\epsilon$, that is, for every $x_0 \in TR$ there exists an instant $t_{x_0}$ such that $x_{x_0}^{\mu_{ba}}(t_{x_0}) \in B_\epsilon$.*

**Remark:** In contrast to other definitions of attractivity Definition 3.4 only requires that the trajectory enters the $Q-$closed ball $B_\epsilon$ once, allowing the possibility to subsequently exit $B_\epsilon$ Although a weaker notion of attractivity, it still represents a notion of a trajectory approaching the origin.

We define stabilizability for (5) and then proceed to show that the optimal policy is stabilizing or even $\epsilon$-attractive for appropriate values of $\alpha$ and $M$:

**Definition 3.5** *The system (5) is* **stabilizable** *if there exists a RAO-safe policy $\mu_s$. The system (5) is called* **exponentially stabilizable** *if there exists an exponentially stable policy $\mu_{es}$.*

**Proposition 3.1** *I. Assume that a system of the form (5) is stabilizable. Then, for any given discount factor $\alpha$, there exists a real $M(\alpha)$ such that the solution of the optimal control problem (8) is stable for discount factor $\alpha$ and unacceptable operation weight factor $M \geq M(\alpha)$. Then,*

$$M(\alpha) = \frac{b}{\alpha^2}, \quad (14)$$

*for some constant $b > 1$.*

*II. Assume that (5) is exponentially stabilizable and let $\mu_{es}$ be an exponentially stabilizing policy. Then, for every $\epsilon > 0$, there exists a number $\alpha_\epsilon \in (0,1)$ such that the solution of the optimal control problem (8) is an $\epsilon$-attractive policy for any discount factor $\alpha \in [\alpha_\epsilon, 1)$ and any unacceptable operation weight factor $M \geq 1$.*

**Proof:** Omitted due to length constraints. The reader is referred to Chapter 5 of [3].

**Remark:** The nonlinear system can be linearized around the origin and an asymptotically stable linear controller $\mu_{LTI}$ can be designed for the linearized system. The nonlinear closed loop system under that linear controller is then locally asymptotically stable, that is, there exists a neighborhood of the origin such that all trajectories starting off in that neighborhood converge to the origin. The number $\epsilon$ can be small enough such that $B_\epsilon$ is subset of that neighborhood. Assume that a nonlinear policy $\mu_{ba}$ forces all trajectories of the closed loop system under $\mu_{ba}$ to go inside $B_\epsilon$. Then, a controller that switches from $\mu_{ba}$ to $\mu_{LTI}$ as soon as a trajectory reaches $B_\epsilon$, is asymptotically stabilizing. This paradigm motivates part II of Proposition 3.1.

## 4 Policy Iteration

As an introduction to the design method, we describe the policy iteration algorithm for the discrete time optimal control problem (5), (8). This is an *exact* algorithm. Approximations are introduced in the approximate policy iteration algorithm, which we present in the next section. To apply policy iteration, it is necessary that the following assumption is satisfied:

**Assumption 4.1** *An initial policy $\mu_0$ is available such that $J_{\mu_0}^{(\alpha,M)}(x_0)$ is finite for every finite $x_0 \in \mathbf{R}^n$.*

According to Proposition 2.1, this assumption is always satisfied, for any policy $\mu_0$, for the optimal control problem considered in this work. Therefore, policy iteration is a feasible approach for our problem. The policy iteration algorithm generates a sequence of policies $\mu_1, \mu_2, \cdots$ that provably satisfy [1]:

$$J_{\mu_0}^{(\alpha,M)}(x) \geq J_{\mu_1}^{(\alpha,M)}(x) \geq J_{\mu_2}^{(\alpha,M)}(x) \geq \cdots,$$

for every $x \in \mathbf{R}^n$. Starting from the $k$-th policy $\mu_k$, the following two step process results in $\mu_{k+1}$:
Policy Evaluation: Compute $J_{\mu_k}^{(\alpha,M)}(x)$ for every $x$;
Policy Improvement: Obtain $\mu_{k+1}(x)$ as the minimizing value of:

$$\min_{u \in \mathbf{R}^m} \left\{ x^T Q x + u^T R u + \alpha J_{\mu_k}^{(\alpha,M)} \left( f(x) + G(x) u \right) \right\}$$

At every $x$, determining $\mu_{k+1}(x)$ amounts to a minimization problem over $u$. Let's take a closer look at the Policy Improvement step. Assume that at time $t = 0$ the value of the state is $x$, and that we have the option to freely choose a control input $u_0$ at $t = 0$, but for all $t \geq 1$ we are forced to use $u_t = \mu_k(x_t)$. Then, the best way to take advantage of the option to freely select $u_0$ is as the minimizing value of

$$\min_{u \in \mathbf{R}^m} \left\{ x^T Q x + \mu(x)^T R \mu(x) + J_{\mu_k}^{(\alpha,M)} (f(x) + G(x) u) \right\}$$

The trajectory with $u_0$ as above at $t = 0$ and $u_t = \mu_k(x_t)$ thereafter, is of improved cost in comparison to the one resulting from $u_t = \mu_k(x_t)$ for all $t \geq 0$. Applying the same strategy to select an control at every $x$, results in the improved controller $\mu_{k+1}$.

Clearly, implementation of policy iteration is practically impossible, since both steps of the algorithm involve a computation over every $x \in \mathbf{R}^n$, and the policy improvement step requires the solution of a generally nonconvex optimization problem. This motivates the approximate policy iteration algorithm.

## 5 Approximate Policy Iteration

Approximate policy iteration is again a two-step algorithm. The second step is now called "policy update", since there are no guarantees that the new policy is an improvement. Given a RAO-unsafe policy $\mu$, we select $\alpha$ and $M$. The corresponding cost function is denoted by $J_\mu^{(\alpha,M)}$. Assume that we are at the $(k+1)$-th step of the iteration, the stabilizing controller $\mu_k$ is available and we want to compute the $(k+1)$-th controller:
Step 1: Approximate Policy Evaluation amounts to determining a function $\tilde{J}_{\mu_k}$ as an approximator of $J_{\mu_k}^{(\alpha,M)}$. We postpone the details of performing this task until section 6.1. For now, we assume that such an approximation is available.
Step 2: Policy Update Ideally, the updated policy should be the policy $\mu_{min}(x)$ which solves

$$\min_{u \in U} \left\{ x^T Q x + u^T R u + \alpha \tilde{J}_\mu(f(x) + G(x)u) \right\}, \tag{15}$$

for all $x \in RAO$. However, updating the policy in this manner calls for the solution of an optimization problem in $u$ for every $x$, which is in general hard. In Section 6, we develop a simplification of the policy update step that allows to obtain an updated policy (denoted by $\bar{\mu}$) in closed form, given $\tilde{J}_{\mu_k}$. The simplification introduces an additional approximation error (the update step approximation error). We assume that both approximation errors are bounded for $x \in RAO$. The errors and their bounds are expressed as

$$\max_{x \in RAO} |\tilde{J}_\mu - J_\mu^{(\alpha,M)}| \leq \epsilon, \tag{16}$$

$$\begin{aligned} \max_{x \in RAO} \Big| \Big\{ & x^T Q x + \mu_{min}(x)^T R \mu_{min}(x) \\ & + \alpha \tilde{J}_\mu(f(x) + G(x)\mu_{min}(x)) \Big\} - \Big\{ x^T Q x \\ & + \bar{\mu}(x)^T R \bar{\mu}(x) + \alpha \tilde{J}_\mu(f(x) + G(x)\bar{\mu}(x)) \Big\} \Big| \leq \delta. \end{aligned} \tag{17}$$

Assume that starting from an initial policy $\mu_0$ a sequence $\mu_0, \mu_1, \cdots$ is generated as described, and that the error bounds $\epsilon$ and $\delta$ are uniform over all iterations.

The following result is due to Bertsekas and Tsitsiklis and reported in [2], p.42:

**Proposition 5.1** *The sequence of policies $\mu_k$ generated by approximate policy iteration satisfies*

$$\limsup_{k \to \infty} \max_{x \in RAO} (J_{\mu_k}^{(\alpha,M)}(x) - J_*^{(\alpha,M)}(x)) \leq \frac{\delta + 2\alpha\epsilon}{(1-a)^2},$$

*where $J_*^{(\alpha,M)}$ is the optimal cost function.*

On the basis of Proposition 5.1, the optimal policy is achievable provided that we have the ability of flawless function approximation and exactly solving the minimization problems (15). In the realistic case of imperfect approximations and minimizations, the policies that are generated are worse in performance than optimal by an amount that increases linearly with $\epsilon$ and $\delta$. Drawing from the proof of Proposition 3.1, we can make a continuity argument to conclude that there are thresholds $\epsilon_d$ and $\delta_d$ such that if not violated by our approximation errors, the suboptimal policies will be stable.

## 6 Implementation of Approximate Policy Iteration

**Definition 6.1 Directional Derivative** *The directional derivative $L_g J$ of a function $J : \mathbf{R}^n \to \mathbf{R}$ along the direction of the vector $g$, at a point $x \in \mathbf{R}^n$ is defined as*

$$L_g J(x) \triangleq \lim_{\delta \to 0} \frac{J(x + \delta g) - J(x)}{\delta}, \tag{18}$$

*provided that the limit exists. Consider a set of vectors, $g_1, \cdots, g_m$, forming a matrix $G = [g_1, \cdots, g_m]$. We denote by $L_G J(x)$ the column vector whose i-th element is $L_{g_i} J(x)$, that is, $L_G J(x) \triangleq [L_{g_1} J(x), \cdots, L_{g_m} J(x)]^T$.*

We consider the case where an unstable nonlinear system in continuous time is given:

$$\dot{x} = f_c(x) + g_c(x)u, \tag{19}$$

where $u$ is scalar (for simplicity), $f_c(x)$ and $g_c(x)$ are continuously differentiable. We do not know any stabilizing controller for (19), so we pick any continuously differentiable $\mu_c(x)$ to close the loop. We define TR and RAO, and use the discrete time model

$$x_{t+1} = \begin{cases} x_t + \delta(f_c(x_t) + g_c(x_t)\mu_c(x_t)) & \text{if } x_t \in RAO \\ x_t & \text{otherwise} \end{cases} \tag{20}$$

as a simulator of (19), for a small discretization interval $\delta$. We select $\alpha$ and $M$ and define the cost function

$J_{\mu_c}^{(\alpha,M)}$, such that $J_{\mu_c}^{(\alpha,M)}(x_0)$ is given by

$$\sum_{t=0}^{N_{\mu_c}(x_0)} \alpha^t \cdot \delta\left(x_{x_0}^{\mu_c}(t)^T Q x_{x_0}^{\mu_c}(t) + \mu_c(x_{x_0}^{\mu_c}(t))^T R \mu_c(x_{x_0}^{\mu_c}(t))\right)$$
$$+M \cdot \Big( \sum_{t=N_{\mu_c}(x_0)+1}^{\infty} \alpha^t \cdot \delta\left(x_{x_0}^{\mu_c}(t)^T Q x_{x_0}^{\mu_c}(t)\right.$$
$$\left. +\mu_c(x_{x_0}^{\mu_c}(t))^T R \mu_c(x_{x_0}^{\mu_c}(t))\right) \Big)$$

Let $\tilde{J}_{\mu_c}$ be a (smooth, by choice) approximate cost function. Then, the policy update rule (15) amounts to solving

$$\min_{u \in U} \left\{ \delta(x^T Q x + u^T R u) + \alpha \tilde{J}_{\mu_c}(x + \delta(f_c(x) + g_c(x)u)) \right\}, \tag{21}$$

for $x \in RAO$. In order to express the updated policy in a closed form, we introduce the first order Taylor approximation of $\tilde{J}_{\mu_c}(x + \delta(f_c(x) + g_c(x)u))$:

$$\tilde{J}_{\mu_c}(x) + \delta L_{f_c(x)} \tilde{J}_{\mu_c}(x) + \delta L_{g_c(x)} \tilde{J}_{\mu_c}(x) u \tag{22}$$

We use (22) into (21) and obtain the alternative optimization problem:

$$\min_{u \in U} \Big\{ \delta(x^T Q x + u^T R u) + \alpha(\tilde{J}_{\mu_c}(x) +$$
$$+\delta L_{f_c(x)} \tilde{J}_{\mu_c}(x) + \delta L_{g_c(x)} \tilde{J}_{\mu_c}(x) u \ ) \Big\}$$
$$= \min_{u \in U} \left\{ \delta(u^T R u) + \delta \alpha L_{g_c(x)} \tilde{J}_{\mu_c}(x) u \right\},$$

with solution $-\frac{\alpha}{2} L_{g_c(x)} \tilde{J}_{\mu_c}(x)$ for every $x$. We use

$$\bar{\mu}(x) = -\frac{\alpha}{2} L_{g_c(x)} \tilde{J}_{\mu_c}(x) \tag{23}$$

as our updated policy. The policy $\bar{\mu}$ is smooth by virtue of smoothness of $\tilde{J}_{\mu_c}$. The approximate policy iteration procedure continues the same way.

### 6.1 Approximating the cost function

In [3], two alternative architectures for approximating the cost function are proposed. The first approximation architecture amounts to tuning the weights of a weighted linear combination of a selected "basis" function set. In this section, we present the second approach: **A grid-based approximation architecture** It follows from the implementation of $\mu_{k+1}$ (23) that it suffices to approximate the directional derivative of $J_{\mu_k}$ in order to generate the $(k+1)$-iterate. Based on that, a systematic way for generating an approximation of $L_{G(x)} J_{\mu_k}(x)$ is proposed in Chapter 6 of [3]. The directional derivative can be computed at each point of the state space via numerical integration of a set of differential equations given in [3], under the assumption of an underlying continuous time system and a discrete time representation of the form (19) and (20). We define a state space grid, and compute $L_G J_{\mu_k}$ at the vertices of the grid. Then, we interpolate between those values in the rest of the state space. This approximation strategy comes with the advantage of generality, and without the need for selecting basis functions.

## 7 Application

In [3], the method is successfully applied to the stabilization problem of a beam-and-ball problem, which is used as a benchmark problem in the nonlinear control community. The performance of the resulting closed loop system compares favorably with the best solutions for the problem that are reported in the literature. The details of this application is omitted due to size limitations. For a detailed account, the reader is referred to [3].

## 8 Conclusions

Feasibility of approximate policy iteration for stabilization of a large class of nonlinear systems that are linear in the control is established. By formulating an appropriate optimal control problem, it is shown that exact policy iteration leads to a stabilizing control design. It is then argued, by virtue of the achievable bounds on suboptimality, that approximate policy iteration may result in a stabilizing controller, provided that the approximation errors are sufficiently small.

## References


[1] D. P. Bertsekas. *Dynamic Programming and Optimal Control*, volume 1. Athena Scientific, Belmont, MA, 1995.

[2] D. P. Bertsekas. *Dynamic Programming and Optimal Control*, volume 2. Athena Scientific, Belmont, MA, 1995.

[3] C. I. Boussios. *An Approach for Nonlinear Control Design via Approximate Dynamic Programming.* PhD thesis, M.I.T., Cambridge, MA, 1998. Also M.I.T. Lab. for Information and Decision Systems Report No. LIDS-TH-2425.

[4] C.-P. Chao and P. M. Fitzsimons. Stabilization of a large class of nonlinear systems using conic sector bounds. *Automatica*, 33(5):945–953, May 1997.

[5] J. Kaloust, C. Ham, and Z. Qu. Nonlinear autopilot control design for a 2-DOF helicopter model. In *IEE Proceedings-Control Theory and Applications*, volume 144, pages 612–616, November 1997.

[6] A. Teel and L. Praly. Tools for semiglobal stabilization by partial state and output feedback. *SIAM J. on Control and Optimization*, 33(5):1443–1488, September 1995.