Chapter 19

The Adaptive Dynamic Programming Theorem

John J. Murray, Chadwick J. Cox,* and Richard E. Saeks*

Abstract: The centerpiece of the theory of dynamic programming is the Hamilton-Jacobi-Bellman (HJB) equation, which can be used to solve for the optimal cost functional V^o for a nonlinear optimal control problem, while one can solve a second partial differential equation for the corresponding optimal control law k^o. Although the direct solution of the HJB equation is computationally untenable, the HJB equation and the relationship between V^o and k^o serves as the basis for the adaptive dynamic programming algorithm. Here, one starts with an initial cost functional and stabilizing control law pair (V_0, k_0) and constructs a sequence of cost functional/control law pairs (V_i, k_i) in real time, which are stepwise stable and converge to the optimal cost functional/control law pair, for a prescribed nonlinear optimal control problem with unknown input affine state dynamics.

19.1 Introduction

Unlike the many soft computing applications where it suffices to achieve a "good approximation most of the time," a control system must be stable all of the time. As such, if one desires to learn a control law in real-time, a fusion of soft computing techniques (to learn the appropriate control law) with hard computing techniques (to maintain the stability constraint and guarantee convergence) is required. To implement this fused/hard computing approach to control, an adaptive dynamic program-

*This research was performed in part on the National Science Foundation SBIR contracts DMI-9660604, DMI-9860370, and DMI-9983287.

ming algorithm, which uses soft computing techniques to learn the optimal cost (or return) functional for a stabilizable nonlinear system with unknown dynamics, and hard computing techniques to verify the stability and convergence of the algorithm was developed in [8], where

- the underlying fusion of soft and hard computing concepts was described,
- the adaptive dynamic programming algorithm was formulated,
- a global convergence theorem for the algorithm with a limited sketch of the proof was introduced, and
- several examples of its application in flight control were presented.

The purpose of this chapter is to provide a detailed proof of the adaptive dynamic programming theorem.

The centerpiece of dynamic programming is the Hamilton-Jacobi-Bellman (HJB) equation [2, 3, 7], which one solves for the optimal cost functional $V^o(x_0, t_0)$. This equation characterizes the cost to drive the initial state x_0 at time t_0 to a prescribed final state using the optimal control. Given the optimal cost functional, one may then solve a second partial differential equation (derived from the HJB equation) for the corresponding optimal control law $k^o(x, t_0)$, yielding an optimal cost functional/optimal control law pair (V^o, k^o).

Although direct solution of the HJB equation is computationally untenable (the so-called "curse of dimensionality"), the HJB equation and the relationship between V^o and the corresponding control law k^o, derived therefrom, serves as the basis of the adaptive dynamic programming algorithm [8]. In this algorithm we start with an initial cost functional/control law pair (V_0, k_0), where k_0 is a stabilizing control law for the plant, and construct a sequence of cost functional/control law pairs (V_i, k_i) in real time, which converge to the optimal cost functional/control law pair (V^o, k^o) as follows.

- Given (V_i, k_i); $i = 0, 1, 2, \cdots$; run the system using control law k_i from an array of initial conditions x_0, covering the entire state space (or that portion of the state space where one expects to operate the system);
- Record the state $x_i(x_0, \cdot)$ and control trajectories $u_i(x_0, \cdot)$ for each initial condition;
- Given this data, define V_{i+1} to be the cost to take the initial state x_0 at time t_0 to the final state, using control law k_i;
- Take k_{i+1} to be the corresponding control law derived from V_{i+1} via the HJB equation;
- Iterate the process until it converges.

In Sections 19.2 and 19.3, we will show that (with the appropriate technical assumptions) this process is

- globally convergent to
- the optimal cost functional V^o and is
- stepwise stable; i.e., k_i is a stabilizing controller at every iteration with Lyapunov function V_i.

Since stability is an asymptotic property, technically it is sufficient that k_i be stabilizing in the limit. In practice, however, if one is going to run the system for any length of time with control law k_i, it is necessary that k_i be a stabilizing controller at each step of the iterative process. As such, for this class of adaptive control problems we "raise the bar," requiring stepwise stability, i.e., stability at each iteration of the adaptive process, rather than simply requiring stability in the limit. This is achieved by showing that V_i is a Lyapunov function for the feedback system with controller k_i, generalizing the classical result [7] that V^o is a Lyapunov function for the feedback system with controller k^o.

An analysis of the above algorithm (see Sections 19.2 and 19.3 for additional details) will reveal that a priori knowledge of the state dynamics is not required to implement the algorithm. Moreover, the requirement that the input mapping be known (to compute k_{i+1} from V_{i+1}) can be circumvented by the precompensator technique described in [8] and [9]. As such the above-described adaptive dynamic programming algorithm can be applied to plants with completely unknown dynamics.

19.2 Adaptive Dynamic Programming Algorithm

In the formulation of the adaptive dynamic programming algorithm and theorem, we use the following notation for the state and state trajectories associated with the plant. The variable x denotes a generic state while x_0 denotes an initial state, t denotes a generic time, and t_0 denotes an initial time. We use the notation $x(x_0, \cdot)$ for the state trajectory produced by the plant (with an appropriate control) starting at initial state x_0 (at some implied initial time), and the notation $u(x_0, \cdot)$ for the corresponding control. Finally, the state reached by a state trajectory at time t is denoted by $x = x(x_0, t)$, while the value of the corresponding control at time t is denoted by $u = u(x_0, t)$.

For the purposes of this chapter, we consider a stabilizable time-invariant input affine plant of the form

$$\dot{x} = f(x, u) \equiv a(x) + b(x)u;\ x(t_0) = x_0 \tag{19.2.1}$$

with input quadratic performance measure

$$J = \int_{t_0}^{\infty} l(x((x_0, \lambda), u(x_0, \lambda)))d\lambda$$

$$\equiv \int_{t_0}^{\infty} [q(x(x_0, \lambda)) + u^T(x_0, \lambda)r(x(x_0, \lambda))u(x_0, \lambda)]d\lambda. \tag{19.2.2}$$

Here $a(x)$, $b(x)$, $q(x)$, and $r(x)$ are C^∞ matrix-valued functions of the state that satisfy

- $a(0) = 0$, producing a singularity at $(x, u) = (0, 0)$;
- the eigenvalues of $da(0)/dx$ have negative real parts, i.e., the linearization of the uncontrolled plant at zero is exponentially stable;

- $q(x) > 0$, $x \neq 0$; $q(0) = 0$;
- $q(x)$ has a positive-definite Hessian at $x = 0$, $d^2q(0)/dx^2 > 0$, i.e., any nonzero state is penalized independently of the direction from which it approaches 0; and
- $r(x) > 0$ for all x.

The goal of the adaptive dynamic programming algorithm is to adaptively construct an optimal control $u^o(x_0, \cdot)$, which takes an arbitrary initial state x_0 at t_0 to the singularity at $(0, 0)$, while minimizing the performance measure J.

Since the plant and performance measure are time-invariant, the optimal cost functional and optimal control law are independent of the initial time t_0, which we may, without loss of generality, take to be 0; i.e., $V^o(x_0, t_0) \equiv V^o(x_0)$ and $k^o(x, t_0) \equiv k^o(x)$. Even though the optimal cost functional is defined in terms of the initial state, it is a generic function of the state $V^o(x)$ and is used in this form in the HJB equation and throughout the chapter. Finally, we adopt the notation

$$F^o(x) \equiv a(x) + b(x)k^o(x)$$

for the optimal closed-loop feedback system. Using this notation, the HJB equation then takes the form

$$\frac{dV^o(x)}{dx} F^o(x) = -l(x, k^o(x)) = -q(x) - k^{oT}(x)r(x)k^o(x) \tag{19.2.3}$$

in the time-invariant case [7].

Differentiating the HJB equation (19.2.3) with respect to $u^o = k^o(x)$ now yields

$$\frac{dV^o(x)}{dx} b(x) = -2k^{oT}(x)r(x) \tag{19.2.4}$$

or equivalently

$$u = k^o(x) = \frac{1}{2} r^{-1}(x) b^T(x) \left[\frac{dV^o(x)}{dx} \right]^T \tag{19.2.5}$$

which is the desired relationship between the optimal control law and the optimal cost functional. Note that an input quadratic performance measure is required to obtain the explicit form for k^o in terms of V^o of (19.2.5), although a similar implicit relationship can be derived in the general case. (See [9] for a derivation of this result.)

Given the above preparation, we may now formulate the desired adaptive dynamic programming algorithm as follows.

Adaptive Dynamic Programming Algorithm.

(1) Initialize the algorithm with a stabilizing cost functional and control law pair (V_0, k_0), where $V_0(x)$ is a C^∞ function, $V_0(x) > 0$, $x \neq 0$; $V_0(0) = 0$, with a positive-definite Hessian at $x = 0$, $d^2V_0(0)/dx^2 > 0$; and $k_0(x)$ is the C^∞ control law,

$$u = k_0(x) = -\frac{1}{2} r^{-1}(x) b^T(x) \left[dV_0(x)/dx \right]^T.$$

(2) For $i = 0, 1, 2, \cdots$, run the system with control law k_i from an array of initial conditions x_0 at $t_0 = 0$, recording the resultant state trajectories $x_i(x_0, \cdot)$ and control inputs $u_i(x_0, \cdot) = k_i(x_i(x_0, \cdot))$.

(3) For $i = 0, 1, 2, \cdots$, let

$$V_{i+1}(x_0) \equiv \int_0^\infty l(x_i(x_0, \lambda), u_i(x_0, \lambda))d\lambda$$

$$u = k_{i+1}(x) = -\frac{1}{2}r^{-1}(x)b^T(x)\left[\frac{dV_{i+1}(x)}{dx}\right]^T$$

where, as above, we have defined V_{i+1} in terms of initial states but use it generically.

(4) Go to (2).

Since the state dynamics matrix $a(x)$ does not appear in the above algorithm, one can implement the algorithm for a system with unknown $a(x)$. Moreover, one can circumvent the requirement that $b(x)$ be known in Step 3 by augmenting the plant with a known precompensator at the cost of increasing its dimensionality, as shown in [8] and [9]. As such, the adaptive dynamic programming algorithm can be applied to plants with completely unknown dynamics.

In the following, we adopt the notation F_i for the closed-loop system defined by the plant and control law k_i:

$$\dot{x} = F_i(x) \equiv a(x) + b(x)k_i(x)$$

$$= a(x) - \frac{1}{2}b(x)r^{-1}(x)b^T(x)\left[\frac{dV_i(x)}{dx}\right]^T. \tag{19.2.6}$$

To initialize the adaptive dynamic programming algorithm for a stable plant, one may take

$$V_0(x) = \epsilon x^T x$$

and

$$k_0(x) = -\epsilon r^{-1}(x)b^T(x)x$$

which will stabilize the plant for sufficiently small ϵ (although in practice we often take $k_0(x) = 0$). Similarly, for a stabilizable plant, one can "prestabilize" the plant with any desired stabilizing control law such that $d^2V_0(x)/dx^2 > 0$ and the eigenvalues of $dF_0(0)/dx$ have negative real parts and then initialize the adaptive dynamic programming algorithm with the above cost functional/control law pair. Moreover, since the state trajectory going through any point in state space is unique, and the plant and controller are time invariant, one can treat every point on a given state trajectory as a new initial state when evaluating $V_{i+1}(x_0)$, by shifting the time scale analytically without rerunning the system.

The adaptive dynamic programming algorithm is characterized by the following theorem.

Theorem 19.2.1 (Adaptive Dynamic Programming Theorem). *Let the sequence of cost functional/control law pairs $(V_i, k_i),\ i = 0, 1, 2, \cdots$ be defined by and satisfy the conditions of the adaptive dynamic programming algorithm. Then,*

(i) *$V_{i+1}(x)$ and $k_{i+1}(x)$ exist, where $V_{i+1}(x)$ and $k_{i+1}(x)$ are C^∞ functions with $V_{i+1}(x) > 0$, $x \neq 0$; $V_{i+1}(0) = 0$; $d^2V_{i+1}(0)/dx^2 > 0$; $i = 0, 1, 2, \cdots$.*

(ii) *The control law k_{i+1} stabilizes the plant with Lyapunov function $V_{i+1}(x)$ for all $i = 0, 1, 2, \cdots$, and the eigenvalues of $dF_{i+1}(0)/dx$ have negative real parts.*

(iii) *The sequence of cost functionals V_{i+1} converge to the optimal cost functional V^o.*

Note that in (ii), the existence of the Lyapunov function $V_{i+1}(x)$ together with the eigenvalue condition on $dF_{i+1}(0)/dx$ implies that the closed-loop system $F_{i+1}(x)$ is exponentially stable [6] rather than asymptotically stable, as implied by the existence of the Lyapunov function alone.

19.3 Proof of the Adaptive Dynamic Programming Theorem

The proof of the adaptive dynamic programming theorem is divided into four steps.

(1) *Show that $V_{i+1}(x)$ and $k_{i+1}(x)$ exist and are C^∞ functions with $V_{i+1}(x) > 0$, $x \neq 0$; $V_{i+1}(0) = 0$; $i = 0, 1, 2, \cdots$.*

By construction $V_{i+1}(x) > 0, x \neq 0; V_{i+1}(0)$, while the existence and smoothness of $k_{i+1}(x)$ follows from that of $V_{i+1}(x)$ since $b(x)$ and $r(x)$ are C^∞ functions and $r^{-1}(x)$ exists.

As such, it suffices to show that $V_{i+1}(x)$ exists and is a C^∞ function. Since $V_{i+1}(x)$ is defined by the state trajectories generated by the ith control law $k_i(x)$, we begin by characterizing the properties of the state trajectories $x_i(x_0, \cdot)$. In particular, since the control law and the plant are defined by C^∞ functions, the state trajectories are also C^∞ functions of both x_0 and t [5]. Furthermore, since $k_i(x)$ is a stabilizing controller and the eigenvalues of $dF_i(0)/dx$ have negative real parts, the state trajectories $x_i(x_0, \cdot)$ converge to zero exponentially [6].

In addition to showing that the state trajectories $x_i(x_0, \cdot)$ are exponentially stable, we would also like to show that the partial derivatives of the state trajectories with respect to the initial condition $\partial^n x_i(x_0, \cdot)/\partial x_0^n$ are also exponentially stable. To this end we observe that $\partial x_i(x_0, \cdot)/\partial x_0$ satisfies the differential equation

$$\frac{\partial}{\partial t}\left[\frac{\partial x_i(x_0,\cdot)}{\partial x_0}\right] = \frac{\partial \dot{x}_i(x_0,\cdot)}{\partial x_0} = \frac{\partial F_i(x_i(x_0,\cdot))}{\partial x_0}$$

$$= \left[\frac{dF_i(x_i(x_0,\cdot))}{dx}\right]\left[\frac{\partial x_i(x_0,\cdot)}{\partial x_0}\right], \quad \frac{\partial x_i(x_0,0)}{\partial x_0} = 1. \tag{19.3.1}$$

Since $x_i(x_0, \cdot)$ is asymptotic to zero, (19.3.1) reduces to the linear time-invariant differential equation

$$\frac{\partial}{\partial t}\left[\frac{\partial x_i(x_0,\cdot)}{\partial x_0}\right] = \left[\frac{dF_i(0)}{dx}\right]\left[\frac{\partial x_i(x_0,\cdot)}{\partial x_0}\right],\ \ \frac{\partial x_i(x_0,0)}{\partial x_0} = 1 \tag{19.3.2}$$

for large t. As such, the partial derivative of the state trajectory with respect to the initial condition $\partial x_i(x_0, \cdot)/\partial x_0$ is exponentially stable since the eigenvalues of $dF_i(0)/dx$ have negative real parts.

Applying the above argument inductively, we assume that $x_i(x_0, \cdot)$ and

$$\frac{\partial^j x_i(x_0,\cdot)}{\partial x_0^j},\ \ j = 1, 2, \cdots n-1$$

are exponentially stable and observe that $\partial^n x_i(x_0, \cdot)/\partial x_0^n$ satisfies a differential equation of the form

$$\frac{\partial}{\partial t}\left[\frac{\partial^n x_i(x_0,\cdot)}{\partial x_0^n}\right] = \left[\frac{dF_i(x_i(x_0,\cdot)}{dx}\right]\left[\frac{\partial^n x_i(x_0,\cdot)}{\partial x_0^n}\right] + D(t),\ \frac{\partial x_i(x_0,\cdot)}{\partial x_0^n} = 0 \tag{19.3.3}$$

where $D(t)$ is a polynomial in $x_i(x_0, \cdot)$ and the trajectories of the lower derivatives $\partial^j x_i(x_0, \cdot)/\partial x_0^j$, $j = 1, 2, \cdots, n-1$. By the inductive hypothesis $x_i(x_0, \cdot)$ and $\partial^j x_i(x_0, \cdot)/\partial x_0^j$, $j = 1, 2, \cdots, n-1$ are all exponentially convergent to zero and, therefore, so is $D(t)$. As such, (19.3.3) reduces to the linear time-invariant differential equation

$$\frac{\partial}{\partial t}\left[\frac{\partial^n x_i(x_0,\cdot)}{\partial x_0^n}\right] = \left[\frac{dF_i(0)}{\partial dx}\right]\left[\frac{\partial^n x_i(x_0,\cdot)}{\partial x_0^n}\right],\ \frac{\partial^n x_i(x_0,\cdot)}{\partial x_0^n} = 0 \tag{19.3.4}$$

for large t, implying that nth partial derivative of the state trajectory $\partial x_i(x_0, \cdot)/\partial x_0$ with respect to the initial condition is exponentially stable, since the eigenvalues of $dF_i(0)/dx$ have negative real parts. As such, $x_i(x_0, \cdot)$ and $\partial^n x_i(x_0, \cdot)/\partial x_0^n$; $n = 1, 2, \cdots$ are exponentially convergent to zero.

See [4] for an alternative proof that the derivatives of the state trajectories with respect to the initial condition are exponentially convergent to zero directly in terms of (19.3.1) and (19.3.3).

To verify the existence of $V_{i+1}(x)$, we express $l(x_i(x_0, \cdot), u_i(x_0, \cdot))$ in the form

$$l(x_i(x_0,\cdot), u_i(x_0,\cdot)) = q(x) + k_i^T(x) r(x) k_i(x)$$

$$= q(x) + \frac{1}{4}\left[\frac{dV_i(x)}{dx}\right] b(x) r^{-1}(x) b^T(x) \left[\frac{dV_i(x)}{dx}\right]^T \equiv l_i(x_i(x_0,\cdot)) \tag{19.3.5}$$

where x denotes $x_i(x_0, \cdot)$ and the notation $l_i(x_i(x_0, \cdot))$ is used to simplify the expression and emphasize that $l(x_i(x_0, \cdot), u_i(x_0, \cdot))$ is a function of the state trajectory. Now, expanding $q(x)$ as a power series around $x = 0$ and recognizing that $q(0) = 0$

and $dq(0)/dx = 0$, since $x = 0$ is a minimum of the positive-definite function $q(x)$, we obtain

$$q(x) = q(0)+\frac{dq(0)}{dx}x+x^T\frac{d^2q(0)}{dx^2}x+o(\|x\|^3) = x^T\frac{d^2q(0)}{dx^2}x+o(\|x\|^3). \quad (19.3.6)$$

As such, there exists K_1 such that $q(x) < K_1\|x\|^2$ for small x. Similarly, upon expanding $dV_i(x)/dx$ in a power series around $x = 0$, and recognizing $dV_i(0)/dx = 0$ since $x = 0$ is a minimum of V_i, we obtain

$$\frac{dV_i(x)}{dx} = \frac{dV_i(0)}{dx} + \frac{d^2V_i(x)}{dx^2}x + o(\|x\|^2) = \frac{d^2V_i(x)}{dx^2}x + o(\|x\|^2). \quad (19.3.7)$$

As such, there exists K_2 such that $dV_i(x)/dx < K_2\|x\|$ for small x. Finally, since $b(x)r(x)^{-1}b^T(x)$ is continuous at zero, there exists K_3 such that $b(x)r^{-1}(x)b^T(x) < K_3$ for small x. Substituting the inequalities $q(x) < K_1\|x\|^2$, $dV_i(x)/dx < K_2\|x\|$, and $b(x)r^{-1}(x)b^T(x) < K_3$ into (19.3.5) therefore yields

$$\begin{aligned} l(x_i(x_0,\cdot),u_i(x_0,\cdot)) &< K_1\|x_i(x_0,\cdot)\|^2 + K_3K_2^2\|x_i(x_0,\cdot)\|^2 \\ &= [K_1 + K_3K_2^2]\|x_i(x_0,\cdot)\|^2 \equiv K\|x_i(x_0,\cdot)\|^2. \end{aligned} \quad (19.3.8)$$

As such,

$$V_{i+1}(x_0) \equiv \int_0^\infty l(x_i(x_0,\lambda),u_i(x_0,\lambda))d\lambda \quad (19.3.9)$$

exists and is continuous in x_0, since the state trajectory $x_i(x_0,\cdot)$ is exponentially convergent to zero.

Finally, to verify that $V_{i+1}(x)$ is a C^∞ function, it suffices to show that trajectories $d^nl_i(x_i(x_0,\cdot))/dx_0^n$ are integrable, in which case one can interchange the derivative and integral operators obtaining

$$\frac{d^nV_{i+1}(x_0)}{dx_0^n} = \int_0^\infty \frac{d^nl_i(x_i(x_0,\cdot))}{dx_0^n}d\lambda. \quad (19.3.10)$$

Now,

$$\frac{dl_i(x_i(x_0,\cdot))}{dx_0} = \frac{dl_i(x_i(x_0,\cdot))}{dx}\frac{dx_i(x_0,\cdot)}{dx_0} \quad (19.3.11)$$

while $d^nl_i(x_i(x_0,\cdot))/dx_0^n$ is a sum of products composed of factors of the form $d^jl_i(x_i(x_0,\cdot))/dx^j$ and $d^kx_i(x_0,\cdot)/dx_0^k$, where every term has at least one factor of the latter type. Since the ith closed-loop system is stable each state trajectory $x_i(x_0,\cdot)$ is contained in a compact set and since $l_i(x_i(x_0,\cdot))$ is a C^∞ function, the derivatives $d^jl_i(x_i(x_0,\cdot))/dx^j$ are bounded on the state trajectory $x_i(x_0,\cdot)$, while we have already shown that the derivatives of the state trajectories with respect to the initial conditions $d^kx_i(x_0,\cdot)/dx_0^k$ converge to zero exponentially. As such, $d^nl_i(x_i(x_0,\cdot))/dx_0^n$ converges to zero exponentially and is therefore integrable, validating (19.3.10) and verifying that $V_{i+1}(x)$ is a C^∞ function.

(2) *Show that the iterative HJB equation*

$$\frac{dV_{i+1}(x)}{dx}F_i(x) = -l(x, k_i(x))$$

is satisfied and that $d^2V_{i+1}(0)/dx^2 > 0$; $i = 0, 1, 2, \cdots$.

To verify the iterative HJB equation we compute $dV_{i+1}(x_i(x_0, t))/dt$ via the chain rule, obtaining

$$\begin{aligned}\frac{dV_{i+1}(x_i(x_0,t))}{dt} &= \frac{dV_{i+1}(x_i(x_0,t))}{dx}\frac{dx_i(x_0,t)}{dt} \\ &= \frac{dV_{i+1}(x_i(x_0,t))}{dx}F_i(x_i(x_0,t)) \qquad (19.3.12)\end{aligned}$$

and by directly differentiating the integral

$$V_{i+1}(x_i(x_0,t)) = \int_0^\infty [l(x_i(x_i(x_0,t),\lambda), u_i(x_i(x_0,t),\lambda))]d\lambda. \qquad (19.3.13)$$

Since there is a unique state trajectory passing through the state $x_i(x_0, t)$, the trajectory $x_i(x_i(x_0, t), \cdot)$ must coincide with the tail, after time t, of the trajectory $x_i(x_0, \cdot)$ starting at x_0 at $t_0 = 0$. Translating this trajectory in time to start at $t_0 = 0$ yields the relationship

$$x_i((x_i(x_0,t),\lambda)) = x_i(x_0, \lambda + t),\ \lambda \geq 0 \qquad (19.3.14)$$

and similarly for the corresponding control. Substituting this expression into (19.3.13) and invoking the change of variable $\gamma = \lambda + t$ now yields

$$\begin{aligned}V_{i+1}(x_i(x_0,t)) &= \int_0^\infty l(x_i(x_0,\lambda+t), u_i(x_0,\lambda+t))d\lambda \\ &= \int_t^\infty l(x_i(x_0,\gamma), u_i(x_0,\gamma))d\gamma. \qquad (19.3.15)\end{aligned}$$

Now,

$$\begin{aligned}&\frac{dV_{i+1}(x_i(x_0,t))}{dt} = \frac{d}{dt}\int_t^\infty l(x_i(x_0,\gamma), u_i(x_0,\gamma))d\gamma \\ &= l(x_i(x_0,\gamma), u_i(x_0,\gamma))\Big|_t^\infty = -l(x_i(x_0,t), u_i(x_0,t))\end{aligned} \qquad (19.3.16)$$

since $l(x_i(x_0, \cdot), u_i(x_0, \cdot)) \equiv l_i(x_i(x_0, \cdot))$ is asymptotic to zero (see (1) above).

Finally, the iterative HJB equation follows by equating the two expressions for $dV_{i+1}(x_i(x_0, t))/dt$ of (19.3.12) and (19.3.16).

To show that

$$\frac{d^2V_{i+1}(0)}{dx^2} > 0,$$

we note that $dV_i(0)/dx = 0$ since zero is a minimum of $V_i(x)$ and, similarly $dV_{i+1}(0)/dx = 0$, while

$$F_i(0) = a(0) - \frac{1}{2}b(0)r^{-1}(0)b^T(0)\left[dV_i(0)/dx\right]^T = 0$$

since $a(0) = 0$. As such, taking the second derivative on both sides of the iterative HJB equation, evaluating it at $x = 0$, and deleting those terms that contain $dV_i(0)/dx$, $dV_{i+1}(0)/dx$, or $F_i(0)$ as a factor yields

$$2\frac{d^2V_{i+1}(0)}{dx^2}\frac{dF_i(0)}{dx} = -\left[\frac{d^2q(0)}{dx^2} + \frac{1}{2}\left[\frac{d^2V_i(0)}{dx^2}\right](b(x)r^{-1}(x)b^T(x))\times \left[\frac{d^2V_i(0)}{dx^2}\right]^T\right]. \tag{19.3.17}$$

Since the right-hand side of (19.3.17) is symmetric, so is the left-hand side. As such, one can replace one of the two terms

$$\frac{d^2V_{i+1}(0)}{dx^2}\frac{dF_i(0)}{dx}$$

on the left-hand side of (19.3.17) by its transpose yielding the linear Lyapunov equation [1]

$$\left[\frac{dF_i(0)}{dx}\right]^T\frac{d^2V_{i+1}(0)}{dx^2} + \frac{d^2V_{i+1}(0)}{dx^2}\left[\frac{dF_i(0)}{dx}\right] = -\left[\frac{d^2q(0)}{dx^2} + \frac{1}{2}\left[\frac{d^2V_{i+1}(0)}{dx^2}\right](b(x)r^{-1}(x)b^T(x))\left[\frac{d^2V_{i+1}(0)}{dx^2}\right]^T\right] \tag{19.3.18}$$

where we have used the fact that $d^2V_{i+1}(0)/dx^2$ is symmetric in deriving (19.3.18). Moreover, since the eigenvalues of $dF_i(0)/dx$ have negative real parts, while

$$\frac{d^2q(0)}{dx^2} > 0 \text{ and } \left[\frac{d^2V_i(0)}{dx^2}\right]b(x)r^{-1}(x)b^T(x)\left[\frac{d^2V_i(0)}{dx^2}\right]^T \geq 0,$$

the unique symmetric solution of (19.3.18) is positive-definite [1]. As such,

$$\frac{d^2V_{i+1}(0)}{dx^2} > 0,$$

as required.

(3) *Show that $V_{i+1}(x)$ is a Lyapunov function for the closed-loop system F_{i+1} and that the eigenvalues of $dF_{i+1}(0)/dx$ have negative real parts, $i = 0, 1, 2, \cdots$.*

To show that k_{i+1} is a stabilizing control law for the plant, we show that $V_{i+1}(x)$ is a Lyapunov function for the closed-loop system, F_{i+1}, $i = 0, 1, 2, \cdots$. Since $V_{i+1}(x)$ is positive-definite it suffices to show that the derivative of $V_{i+1}(x)$ along the

state trajectories defined by the control law k_{i+1}, $dV_{i+1}(x_{i+1}(x_0,t))/dt$ is negative-definite. To this end we use the chain rule to compute

$$\frac{dV_{i+1}(x_{i+1}(x_0,t))}{dt} = \frac{d[V_{i+1}(x_{i+1}(x_0,t))]}{dx}\frac{dx_{i+1}(x_0,t)}{dt}$$

$$= \frac{d[V_{i+1}(x_{i+1}(x_0,t))]}{dx}F_{i+1}(x_{i+1}(x_0,t)). \tag{19.3.19}$$

Now, upon substituting

$$F_{i+1}(x_{i+1}) = a(x_{i+1}) - \frac{1}{2}b(x_{i+1})r^{-1}(x_{i+1})b^T(x_{i+1})\left[\frac{dV_{i+1}(x_{i+1})}{dx}\right]^T \tag{19.3.20}$$

(where we have used x_{i+1} as a shorthand notation for $x_{i+1}(x_0,t)$) into (19.3.19), we obtain

$$\frac{dV_{i+1}(x_{i+1}(x_0,t))}{dt} = \left[\frac{dV_{i+1}(x_{i+1})}{dx}\right]a(x_{i+1})$$

$$-\frac{1}{2}\left[\frac{dV_{i+1}(x_{i+1})}{dx}\right]b(x_{i+1})r^{-1}(x_{i+1})b^T(x_{i+1})\left[\frac{dV_{i+1}(x_{i+1})}{dx}\right]^T. \tag{19.3.21}$$

Similarly, we may substitute the equality

$$F_i(x_{i+1}) = a(x_{i+1}) - \frac{1}{2}b(x_{i+1})r^{-1}(x_{i+1})b^T(x_{i+1})\left[\frac{dV_i(x_{i+1})}{dx}\right]^T \tag{19.3.22}$$

into the iterative HJB equation obtaining

$$\left[\frac{dV_{i+1}(x_{i+1})}{dx}\right]a(x_{i+1}) = \frac{1}{2}\left[\frac{dV_{i+1}(x_{i+1})}{dx}\right]b(x_{i+1})r^{-1}(x_{i+1})b^T(x_{i+1})\times$$

$$\left[\frac{dV_i(x_{i+1})}{dx}\right]^T - l(x_{i+1},k_i(x_{i+1})). \tag{19.3.23}$$

Substituting (19.3.23) into (19.3.21) now yields

$$\frac{dV_{i+1}(x_{i+1}(x_0,t))}{dt} = \frac{1}{2}\left[\frac{dV_{i+1}(x_{i+1})}{dx}\right]b(x_{i+1})r^{-1}(x_{i+1})b^T(x_{i+1})\times$$

$$\left[\frac{dV_i(x_{i+1})}{dx}\right]^T - l(x_{i+1},k_i(x_{i+1}))$$

$$-\frac{1}{2}\left[\frac{dV_{i+1}(x_{i+1})}{dx}\right]b(x_{i+1})r^{-1}(x_{i+1})b^T(x_{i+1})\left[\frac{dV_{i+1}(x_{i+1})}{dx}\right]^T \tag{19.3.24}$$

while expressing $l(x_{i+1},k_i(x_{i+1}))$ in the form

$$l(x_{i+1},k_i(x_{i+1})) = q(x_{i+1}) + \frac{1}{4}\left[\frac{dV_i(x_{i+1})}{dx}\right]\times$$

$$b(x_{i+1})r^{-1}(x_{i+1})b^T(x_{i+1})\left[\frac{dV_i(x_{i+1})}{dx}\right]^T \tag{19.3.25}$$

and substituting this expression into (19.3.24) yields

$$\frac{dV_{i+1}(x_{i+1}(x_0,t))}{dt} = \frac{1}{2}\left[\frac{dV_{i+1}(x_{i+1})}{dx}\right]b(x_{i+1})r^{-1}(x_{i+1})b^T(x_{i+1})\left[\frac{dV_i(x_{i+1})}{dx}\right]^T$$

$$-q(x_{i+1}) - \frac{1}{4}\left[\frac{dV_i(x_{i+1})}{dx}\right]b(x_{i+1})r^{-1}(x_{i+1})b^T(x_{i+1})\left[\frac{dV_i(x_{i+1})}{dx}\right]^T$$

$$-\frac{1}{2}\left[\frac{dV_{i+1}(x_{i+1})}{dx}\right]b(x_{i+1})r^{-1}(x_{i+1})b^T(x_{i+1})\left[\frac{dV_{i+1}(x_{i+1})}{dx}\right]^T. \tag{19.3.26}$$

Finally, upon completing the square, (19.3.26) reduces to

$$\frac{dV_{i+1}(x_{i+1}(x_0,t))}{dt} = -q(x_{i+1}) - \frac{1}{4}\left[\frac{d[V_{i+1}(x_{i+1}) - V_i(x_{i+1})]}{dx}\right]\times$$

$$b(x_{i+1})r^{-1}(x_{i+1})b^T(x_{i+1})\left[\frac{d[V_{i+1}(x_{i+1}) - V_i(x_{i+1})]}{dx}\right]^T$$

$$-\frac{1}{4}\left[\frac{dV_{i+1}(x_{i+1})}{dx}\right]b(x_{i+1})r^{-1}(x_{i+1})b^T(x_{i+1})\left[\frac{dV_{i+1}(x_{i+1})}{dx}\right]^T. \tag{19.3.27}$$

As such,

$$\frac{dV_{i+1}(x_{i+1}(x_0,t))}{dt} < 0 \text{ for } x_{i+1}(x_0,t) \neq 0$$

verifying that $V_{i+1}(x)$ is a Lyapunov function for F_{i+1} and that k_{i+1} is a stabilizing controller for the plant, as required.

To show that the eigenvalues of $dF_{i+1}(0)/dx$ have negative real parts, we note that $dV_{i+1}(0)/dx = 0$ since it is a minimum of $V_{i+1}(x)$, and similarly that

$$\frac{dV_i(0)}{dx} = 0,$$

while

$$F_{i+1}(0) = a(0) - \frac{1}{2}b(0)r^{-1}(0)b^T(0)\left[\frac{dV_{i+1}(0)}{dx}\right]^T = 0$$

since $a(0) = 0$. Now, substituting (19.3.19) for the left-hand side of (19.3.27), taking the second derivative on both sides of the resultant equation, evaluating it at $x = 0$, and deleting those terms that contain $dV_{i+1}(0)/dx$, $dV_i(0)/dx$, or $F_{i+1}(0)$ as a factor, yields

$$2\frac{d^2V_{i+1}(0)}{dx^2}\frac{dF_{i+1}(0)}{dx} = -\frac{d^2q(0)}{dx^2} - \frac{1}{2}\left[\frac{d^2V_{i+1}(0)}{dx^2}\right]b(0)r^{-1}(0)b^T(0)\left[\frac{d^2V_{i+1}(0)}{dx^2}\right]^T$$

$$-\frac{1}{4}\left[\frac{d^2[V_{i+1}(0)-V_i(0)]}{dx^2}\right]b(0)r^{-1}(0)b^T(0)\left[\frac{d^2[V_{i+1}(0)-V_i(0)]}{dx^2}\right]^T. \quad (19.3.28)$$

Now, since the right-hand side of (19.3.28) is symmetric so is the left-hand side and, as such, we may equate the left-hand side of (19.3.28) to its hermitian part. Moreover, since $-d^2q(0)/dx^2 < 0$ while the second and third terms on the right-hand side of (19.3.28) are negative semidefinite, the right-hand side of (19.3.28) reduces to a negative-definite symmetric matrix $-Q$. As such, (19.3.28) may be expressed in the form

$$\left[\frac{dF_{i+1}(0)}{dx}\right]^T\frac{d^2V_{i+1}(0)}{dx^2}+\frac{d^2V_{i+1}(0)}{dx^2}\left[\frac{dF_{i+1}(0)}{dx}\right]=-Q. \quad (19.3.29)$$

Finally, to verify that the eigenvalues of $dF_{i+1}(0)/dx$ have negative real parts we let λ be an arbitrary eigenvalue of $dF_{i+1}(0)/dx$ with eigenvector v. As such, $(dF_{i+1}(0)/dx)v = \lambda v$, while premultiplying this relationship by $v^*d^2V_{i+1}(0)/dx^2$ yields

$$v^*\frac{d^2V_{i+1}(0)}{dx^2}\frac{dF_{i+1}(0)}{dx}v=\lambda v^*\frac{d^2V_{i+1}(0)}{dx^2}v. \quad (19.3.30)$$

Now, upon taking the complex conjugate of (19.3.30) and adding it to (19.3.30), we obtain

$$v^*\left(\left[\frac{dF_{i+1}(0)}{dx}\right]^T\frac{d^2V_{i+1}(0)}{dx^2}+\frac{d^2V_{i+1}(0)}{dx^2}\left[\frac{dF_{i+1}(0)}{dx}\right]\right)v=2Re(\lambda)v^*\frac{d^2V_{i+1}(0)}{dx^2}v. \quad (19.3.31)$$

Finally, substituting (19.3.29) in (19.3.31) yields

$$-v^*Qv=2Re(\lambda)v^*\frac{d^2V_{i+1}(0)}{dx^2}v \quad (19.3.32)$$

from which it follows that $Re(\lambda) < 0$, since $d^2V_{i+1}(0)/dx^2 > 0$ (see part (2) of the proof), and $-v^*Qv < 0$.

(4) *Show that the sequence of cost functionals V_{i+1} is convergent.*

The key step in our convergence proof is to show that

$$\frac{d[V_{i+1}(x_i(x_0,t))-V_i(x_i(x_0,t))]}{dt}=\frac{d[V_{i+1}(x_i(x_0,t))]}{dt}-\frac{d[V_i(x_i(x_0,t))]}{dt} \quad (19.3.33)$$

is positive along the trajectories defined by the control law k_i. Substituting (19.3.5) into (19.3.16) yields

$$\frac{d[V_{i+1}(x_i(x_0,t))]}{dt}=-l(x_i,u_i)$$

$$=-q(x_i)-\frac{1}{4}\left[\frac{dV_i(x_i)}{dx}\right]b(x_i)r^{-1}(x_i)b^T(x_i)\left[\frac{dV_i(x_i)}{dx}\right]^T \quad (19.3.34)$$

(where we have used x_i as a shorthand notation for $x_i(x_0,t)$ and similarly for u_i) while one can obtain an expression for $[dV_i(x,t)/dt]\big|_{x_i(x_0,t)}$ from (19.3.27) by replacing the index $i+1$ by the index i

$$\frac{dV_i(x_i(x_0,t))}{dt} = -q(x_i) - \frac{1}{4}\left[\frac{d[V_i(x_i)-V_{i-1}(x_i)]}{dx}\right] b(x_i)r^{-1}(x_i)b^T(x_i)\times$$
$$\left[\frac{d[V_i(x_i)-V_{i-1}(x_i)]}{dx}\right]^T - \frac{1}{4}\left[\frac{dV_i(x_i)}{dx}\right] b(x_i)r^{-1}(x_i)b^T(x_i)\left[\frac{dV_i(x_i)}{dx}\right]^T \tag{19.3.35}$$

which is valid for $i = 1, 2, 3, \cdots$ after reindexing. Finally, substituting (19.3.34) and (19.3.35) into (19.3.33) yields

$$\frac{d[V_{i+1}(x_i(x_0,t))-V_i(x_i(x_0,t))]}{dx} = -q(x_i) - \frac{1}{4}\left[\frac{dV_i(x_i)}{dx}\right] b(x_i)r^{-1}(x_i)b^T(x_i)\times$$
$$\left[\frac{dV_i(x_i)}{dx}\right]^T q(x_i) + \frac{1}{4}\left[\frac{d[V_i(x_i)-V_{i-1}(x_i)]}{dx}\right] b(x_i)r^{-1}(x_i)b^T(x_i)\times$$
$$\left[\frac{d[V_i(x_i)-V_{i-1}(x_i)]}{dx}\right]^T + \frac{1}{4}\left[\frac{dV_i(x_i)}{dx}\right] b(x_i)r^{-1}(x_i)b^T(x_i)\left[\frac{dV_i(x_i)}{dx}\right]^T$$
$$= \frac{1}{4}\left[\frac{d[V_i(x_i)-V_{i-1}(x_i)]}{dx}\right] b(x_i)r^{-1}(x_i)b^T(x_i)\left[\frac{d[V_i(x_i)-V_{i-1}(x_i)]}{dx}\right]^T > 0 \tag{19.3.36}$$

for $i = 1, 2, 3, \cdots$.

Since F_i is asymptotically stable, its state trajectories $x_i(x,\cdot)$ converge to zero, and hence so does $V_{i+1}(x_i(x_0,\cdot)) - V_i(x_i(x_0,\cdot))$. Since

$$\frac{d[V_{i+1}(x)-V_i(x)}{dt} > 0$$

on these trajectories, however, this implies that

$$V_{i+1}(x_i(x_0,\cdot)) - V_i(x_i(x_0,\cdot)) < 0$$

on the trajectories of F_i, $i = 1, 2, 3, \cdots$. Since every point x in the state space lies along some trajectory of F_i, $x = x_i(x_0,t)$, however, this implies that $V_{i+1}(x) - V_i(x) < 0$ for all x in the state space, or equivalently, $V_{i+1}(x) < V_i(x)$ for all x; $i = 1, 2, 3, \cdots$. As such, $V_{i+1}(x)$, $i = 1, 2, 3, \cdots$ is a decreasing sequence of positive numbers, $i = 1, 2, 3, \cdots$, and is therefore convergent (as is the sequence, $V_{i+1}(x)$, $i = 0, 1, 2, \cdots$, since the behavior of the first entry of a sequence does not affect its convergence), completing the proof of the adaptive dynamic programming theorem.

Although an initial cost functional of the form $V_0(x) = \epsilon x^T x$ is technically required to initialize the algorithm for a stable plant (to guarantee that $d^2V_0(0)/dx^2 > 0$), a review of the proof will reveal that one can, in fact, initiate the adaptive dynamic programming algorithm for a stable system with $V_0(x) = 0$.

19.4 Conclusions

Our goal was to provide a detailed proof of the adaptive dynamic programming theorem. The reader is referred to [8] for a discussion of the techniques used to implement the theorem in a computationally efficient manner, and examples of its application to both linear and nonlinear systems.

Bibliography

[1] S. Barnett, *The Matrices of Control Theory*, Van Nostrand Reinhold, New York, 1971.

[2] R. E. Bellman, *Dynamic Programming*, Princeton University Press, Princeton, NJ, 1957.

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

[4] A. Devinatz and J. L. Kaplan, "Asymptotic estimates for solutions of linear systems of ordinary differential equations having multiple characteristics roots," *Indiana University Math J.*, vol. 22, p. 335, 1972.

[5] J. Dieudonne, *Foundations of Mathematical Analysis*, Academic Press, New York, 1960.

[6] A. Halanay and V. Rasvan, *Applications of Liapunov Methods in Stability*, Kluwer, Dordrecht, 1993.

[7] D. G. Luenberger, *Introduction to Dynamic Systems: Theory, Models, and Applications*, John Wiley, New York, 1979.

[8] J. Murray, C. Cox, G. Lendaris, and R. Saeks, "Adaptive dynamic programming," *IEEE Trans. Systems, Man and Cybernetics: Part C*, vol. 32, pp. 140–153, 2002.

[9] R. Saeks and C. Cox, "Adaptive critic control and functional link networks," *Proc. 1998 IEEE Conference on Systems, Man and Cybernetics*, San Diego, CA, pp. 1652–1657, 1998.