Nearly Optimal Control Laws for Nonlinear Systems with Saturating Actuators Using a Neural Network HJB Approach

Murad Abu-Khalaf, *Member, IEEE* and Frank L. Lewis, *Fellow, IEEE*

***Abstract*—We consider the use of nonlinear approximating networks to obtain nearly optimal solutions to constrained control problems. The method is based on least-squares successive approximation solution of the Generalized Hamilton-Jacobi-Bellman (GHJB) equation which appears in optimization problems. Successive approximation using the GHJB has not yet been rigorously applied for saturated controls. The proposed method successively solves the GHJB equation on a well-defined region of attraction making use of a suitable nonquadratic functional that allows working with smooth saturated controls. A neural network is used to approximate the GHJB solution. It is shown that the result is a closed-loop control based on a neural net that has been tuned a priori off-line. As the order of the network is increased, and as the algorithm is run on more points in the well-defined region of attraction, the network converges to the exact solution of the inherently nonlinear HJB equation associated with the saturating control inputs.**

***Index Terms*—Constrained control, nonquadratic performance functionals, optimal control, saturation.**

I. Introduction

The control of systems with saturating actuators has been the focus of many researchers for many years. Several methods for deriving control laws considering the saturation phenomena are found in [21], [23], [5]. However, most of these methods do not consider finding optimal control laws for general nonlinear systems. In this paper, we study this problem through the framework of the Hamilton-Jacobi-Bellman (HJB) equation resulting from optimal control theory [11]. The solution of the HJB equation is a challenging problem due to its inherently nonlinear nature. For linear systems, this equation results in the well-known Riccati equation used to derive a linear state feedback control. But even when the system is linear, the saturated control requirement makes the required control nonlinear, and makes the solution of the HJB equation more challenging.

In the nonlinear case, the HJB equation generally cannot be solved. There has been a great deal of effort to attack this problem. Approximate HJB solution has been confronted using many techniques by Saridis [22], Beard [2], [3], [4], Lendaris [20], Bertsekas and Tsitsiklis, [6], Munos [19], Lewis and Kim [9], Balakrishnan, [13], Lyshevski [14], [15], [16], [17], [18], Huang [8] and others.

Here, we focus on HJB solution using the so-called generalized HJB equation (GHJB) [4], [22]. In [22], Saridis et al. developed a successive approximation method that improves a given initial stabilizing control. This method reduces to the well-known Kleinman iterative method for solving the Riccati equation for linear systems [10]. However, for nonlinear systems, it is unclear how to solve the GHJB equation. Therefore, successful application of the GHJB was limited until the novel work of Beard [2], [3], [4]. He uses a Galerkin spectral approximation method to find approximate but close solutions to the GHJB at each iteration. The framework in which the algorithm is presented in Beard's work requires the computation of a large number of integrals and is also not suitable to handle explicit constraints on the controls, which is what we are interested in. In [15], [16], Lyshevski proposed a generalized nonquadratic functional to derive a smooth saturated control structure based on the HJB equation. But it remains difficult to solve the final nonlinear HJB equation.

We employ the method of weighted residuals and use it along with the successive approximation technique to get a least-squares solution to the HJB employing a nonquadratic functional for the control input. Thus the nearly optimal saturated control input is found. A neural network, [12], is used to approximate the GHJB solution at each successive iteration. It is shown that the result is a closed-loop control based on a neural net that has been tuned a priori off-line. A preliminary report of this work appears in [1].

II. Background in Optimal Control and Constrained Input Systems

Consider an affine in the control nonlinear dynamical system of the form

$$\dot{x} = f(x) + g(x)u(x) \tag{1}$$

where $x \in \Re^n$, $u : \Re^n \to \Re^m$, $f : \Re^n \to \Re^n$ and

Research supported by grants NSF ECS-0140490 and ARO DAAD19-02-1-0366.

M. Abu-Khalaf is with the Automation and Robotics Research Institute, The University of Texas at Arlington, Fort worth, TX 76118 USA. (E-mail: abukhalaf@arri.uta.edu).

F. L. Lewis is with the Automation and Robotics Research Institute, The University of Texas at Arlington, Fort worth, TX 76118 USA. (E-mail: flewis@arri.uta.edu).

$g:\Re^n \to \Re^n \times \Re^m$. Assume that $f+gu$ is Lipschitz continuous on a set Ω in $\Re^n$ containing the origin, and that the system (1) is controllable in the sense that there exists a continuous control on Ω that asymptotically stabilizes the system.

It is desired to find a control function $u:\Re^n \to \Re^m$, which minimizes a generalized nonquadratic functional

$$V=\int_0^\infty \left[Q(x)+W(u)\right]dt \tag{2}$$

where $Q(x)$ is positive definite monotonically increasing function on Ω, and thus satisfies the observability condition. $W(u)$ is a positive definite integrand function. For unbounded control inputs, a common choice for $W(u)$ is

$$W(u)=u^T Ru \tag{3}$$

where $R\in\Re^m\times\Re^m$. Note that the control u must not only stabilize the system on Ω, but also make the integral finite. Such controls are defined to be admissible [3].

Definition 2.1: ***Admissible Controls***

Let $\Psi(\Omega)$ denote the set of admissible controls. A control $u:\Re^n\to\Re^m$ is defined to be admissible with respect to the state penalty function $Q(x)$ on Ω, denoted $u\in\Psi(\Omega)$, if:

1. u is continuous on Ω,
2. $u(0)=0$,
3. u stabilizes (1) on Ω,
4. $\int_0^\infty [Q(x)+W(u)]dt<\infty$, $\forall x\in\Omega$.

■

Differentiating V, the value function, along the system trajectories, we obtain what is known as the GHJB equation,

$$GHJB(V,u)\overset{\Delta}{=}\frac{\partial V}{\partial x}^T\left(f+gu\right)+Q+u^TRu=0, \tag{4}$$
$$V(0)=0.$$

Note that the GHJB equation becomes the well-known HJB equation on substitution of the optimal control

$$u^*(x)=-\frac{1}{2}R^{-1}g^T(x)\frac{\partial V^*(x)}{\partial x} \tag{5}$$

where $V^*(x)$ is the unique optimal solution to the Hamilton-Jacobi-Bellman (HJB) equation

$$HJB(V^*)\overset{\Delta}{=}\frac{\partial V^*}{\partial x}^T f+Q-\frac{1}{4}\frac{\partial V^*}{\partial x}^T gR^{-1}g^T\frac{\partial V^*}{\partial x}=0, \tag{6}$$
$$V^*(0)=0.$$

It is shown in [15] that the value function obtained from (6) serves as a Lyapunov function on Ω. It is important to note that the GHJB is linear in the value function derivative, while the HJB is nonlinear in the value function derivative. Solving the GHJB requires solving linear partial differential equations, while the HJB equation solution involves nonlinear partial differential equations, which may be impossible to solve. This is the reason for introducing the successive approximation technique using GHJB, which was based on a sound proof in [22]. In the successive approximation method, one solves (4) for $V(x)$ given a stabilizing control $u(x)$, then finds an improved control based on $V(x)$ using

$$u=-\frac{1}{2}R^{-1}g^T\frac{\partial V}{\partial x}. \tag{7}$$

Saridis [22] shows that if the initial control $u^{(0)}\in\Psi(\Omega)$, then repetitive application of (4), (7) is a contraction map, and that the sequence of solutions $V^{(i)}$ converges to the optimal HJB solution $V^*(x)$. This assumes one can find an exact solution to (4) at each step.

Although the GHJB equation is in theory easier to solve than the HJB equation, there is no general closed-form solution available to this equation. In [2], [3], Beard used Galerkin's spectral method to get an approximate solution $V(x)$ in (4) at each iteration. He proves convergence in the overall run. This technique does not set the GHJB equation to zero at each iteration, but to a residual error instead.

The Galerkin approximation requires the evaluation of numerous integrals. Moreover, in its current format, the successive approximation algorithm is unable to deal with saturated controls.

To confront bounded controls, Lyshevski [15], [16] introduced a generalized nonquadratic functional

$$W(u)=2\int\left(\phi^{-1}(u)\right)^T Rdu \tag{8}$$

where $\phi(\cdot):\Re^c\to\Re^m$ is a continuous one-to-one, bounded, real-analytic integrable function of class C^p $(p\ge 1)$ with $\phi(0)=0$, e.g. $\phi(\cdot)\to\tanh(\cdot)$. R is positive definite and assumed to be symmetric for simplicity of analysis. This does not restrict the design criteria on the control input vector, because the number of coefficients that we can choose independently in the symmetric design matrix R is equal to the number of quadratic terms possible from the control input vector. These two numbers are equal, that is $\frac{m^2-m}{2}+m=m+\binom{m}{2}$. Note that $W(u)$ is positive definite if $\phi^{-1}(u)$ has the same sign as u and R is positive definite.

For saturated controls, GHJB design equations (4), (7) are replaced with

$$\frac{\partial V}{\partial x}^T\left(f+g\cdot u\right)+Q+2\int\left(\phi^{-1}(u)\right)^T Rdu=0, \tag{9}$$
$$V(0)=0.$$

$$u(x)=-\phi\left(\frac{1}{2}R^{-1}g^T(x)\frac{\partial V(x)}{\partial x}\right). \tag{10}$$

Note that equation (10) guarantees that $u(x)$ is bounded.

If we substitute (10) into (9) we obtain the HJB equation for bounded controls. The positive definite solution of this equation is the stabilizing value function and its corresponding optimal control. Existence and uniqueness of the value function has been shown in [18]. This HJB equation cannot generally be solved. There is no current method for rigorously confronting this type of equation to find the value function for the system. Moreover, current solutions are not well defined over a specific region in the state space.

III. SUCCESSIVE APPROXIMATION OF HJB FOR SATURATED CONTROLS

Successive approximation using the GHJB has not yet been rigorously applied for bounded controls. In this section, we will show that the successive approximation technique can be used for constrained controls when certain restrictions on the control input are met. Then, having the successive approximation theory well set, in the next section we will introduce a neural network approximation of the value function, and employ the successive solutions method in a least-squares sense over a mesh with certain size on Ω. This is far simpler than the Galerkin approximation appearing in [2], [3].

The successive approximation technique is now applied to the new set of equations (9), (10). The following Lemma shows how equation (10) can be used to improve the control law. It will be required that the control bound $\phi(\cdot)$ is monotonically non-decreasing.

Lemma 3.1: ***Improved Saturated Control Law***

If $u^{(i)} \in \Psi(\Omega)$, and $V^{(i)}$ satisfies the equation $GHJB(V^{(i)},u^{(i)})=0$ with the boundary condition $V^{(i)}(0)=0$, then the new control derived as

$$u^{(i+1)}(x) = -\phi\left(\frac{1}{2}R^{-1}g^T(x)\frac{\partial V^{(i)}(x)}{\partial x}\right) \tag{11}$$

is an admissible control for the system on Ω. Moreover, if the control bound $\phi(\cdot)$ is monotonically non-decreasing and $V^{(i+1)}$ is the unique positive definite function satisfying the equation $GHJB(V^{(i+1)},u^{(i+1)})=0$, with the boundary condition $V^{(i+1)}(0)=0$, then $V^{(i+1)}(x) \le V^{(i)}(x) \quad \forall x \in \Omega$.

Proof:

Admissibility: Since $V^{(i)}$ is continuously differentiable, the continuity assumption on g implies that $u^{(i+1)}$ is continuous. Since $V^{(i)}$ is positive definite it attains a minimum at the origin, and thus, $\partial V^{(i)}(x)/\partial x$ must vanish. This implies that $u^{(i+1)}(0)=0$.

Taking the derivative of $V^{(i)}$ along the system $(f,g,u^{(i+1)})$ trajectory we have,

$$\dot V^{(i)}(x,u^{(i+1)}) = \frac{\partial V^{(i)}(x)}{\partial x}^T f + \frac{\partial V^{(i)}(x)}{\partial x}^T g \quad u^{(i+1)} \tag{12}$$

But

$$\begin{aligned}\frac{\partial V^{(i)}(x)}{\partial x}^T f = -\frac{\partial V^{(i)}(x)}{\partial x}^T g \quad u^{(i)} - \\ Q(x) - 2\int_0^{u^{(i)}} \left(\phi^{-1}(u)\right)^T R du.\end{aligned} \tag{13}$$

This becomes

$$\begin{aligned}\dot V^{(i)}(x,u^{(i+1)}) = -\frac{\partial V^{(i)}(x)}{\partial x}^T g \quad u^{(i)} + \frac{\partial V^{(i)}(x)}{\partial x}^T g \quad u^{(i+1)} - \\ Q(x) - 2\int_0^{u^{(i)}} \left(\phi^{-1}(u)\right)^T R \quad du.\end{aligned} \tag{14}$$

Making use of the fact that $\frac{\partial V^{(i)}(x)}{\partial x}^T g(x) = -2\phi^{-1}\left(u^{(i+1)}\right)^T R$, we get

$$\begin{aligned}&\dot V^{(i)}(x,u^{(i+1)}) = -Q(x) + \\ &2\left\{\phi^{-1}\left(u^{(i+1)}\right)^T R(u^{(i)}-u^{(i+1)}) - \int_0^{u^{(i)}}\left(\phi^{-1}(u)\right)^T R du\right\}.\end{aligned} \tag{15}$$

The second term in the previous equation is negative when ϕ^{-1} and thus ϕ is monotonically non-decreasing. To see this, note that the design matrix R is symmetric positive definite, this means we can rewrite it as

$$R = \Lambda\Sigma\Lambda \tag{16}$$

where Σ is a triangular matrix with its values being the singular values of R and Λ is an orthogonal symmetric matrix. Substituting (16) in (15) we get,

$$\begin{aligned}&\dot V^{(i)}(x,u^{(i+1)}) = -Q(x) + \\ &2\left[\phi^{-1}\left(u^{(i+1)}\right)^T \Lambda\Sigma\Lambda(u^{(i)}-u^{(i+1)}) - \int_0^{u^{(i)}}\left(\phi^{-1}(u)\right)^T \Lambda\Sigma\Lambda du\right].\end{aligned} \tag{17}$$

Applying the coordinate change $u = \Lambda^{-1}z$, equation (17) then becomes

$$\begin{aligned}&\dot V^{(i)}(x,u^{(i+1)}) = -Q(x) + \\ &2\left\{\begin{array}{l}\phi^{-1}\left(\Lambda^{-1}z^{(i+1)}\right)^T \Lambda\Sigma\Lambda(\Lambda^{-1}z^{(i)} - \Lambda^{-1}z^{(i+1)}) - \\ \int_0^{z^{(i)}}\left(\phi^{-1}(\Lambda^{-1}z)\right)^T \Lambda\Sigma\Lambda\Lambda^{-1}dz\end{array}\right\} \\ &= -Q(x) + 2\left\{\begin{array}{l}\phi^{-1}\left(\Lambda^{-1}z^{(i+1)}\right)^T \Lambda\Sigma(z^{(i)}-z^{(i+1)}) - \\ \int_0^{z^{(i)}}\left(\phi^{-1}(\Lambda^{-1}z)\right)^T \Lambda\Sigma dz\end{array}\right\} \\ &= -Q(x) + 2\left\{\pi^T\left(z^{(i+1)}\right)\Sigma(z^{(i)}-z^{(i+1)}) - \int_0^{z^{(i)}}\pi^T\left(z\right)\Sigma dz\right\}.\end{aligned} \tag{18}$$

where $\pi^T\left(z^{(i)}\right) = \phi^{-1}\left(\Lambda^{-1}z^{(i)}\right)^T \Lambda$.

Since Σ is a triangular matrix, we can now decouple the transformed input vector such that

$$\begin{aligned}&\dot V^{(i)}(x,u^{(i+1)}) = -Q(x) + \\ &2\left\{\pi^T\left(z^{(i+1)}\right)\Sigma(z^{(i)}-z^{(i+1)}) - \int_0^{z_k^{(i)}}\pi^T\left(z\right)\Sigma dz\right\} \\ &= -Q(x) + 2\sum_{k=1}^m \Sigma_{kk}\left\{\begin{array}{l}\pi^T\left(z_k^{(i+1)}\right)(z_k^{(i)}-z_k^{(i+1)}) - \\ \int_0^{z_k^{(i)}}\pi^T\left(z_k\right)dz_k\end{array}\right\}.\end{aligned} \tag{19}$$

Since the matrix R is positive definite, then we have the singular values Σ_{kk} being all positive. Also, from the geometrical meaning of $\pi^T\left(z_k^{(i+1)}\right)(z_k^{(i)}-z_k^{(i+1)}) - \int_0^{z_k^{(i)}}\pi^T\left(z_k\right)dz_k$, this term is always negative if $\pi^T\left(z_k\right)$ is monotonically non-decreasing. But since $\pi^T\left(z^{(i)}\right) = \phi^{-1}\left(\Lambda^{-1}z^{(i)}\right)^T\Lambda$, it is easy to show that ϕ^{-1} should be monotonically non-decreasing, and thus ϕ itself should be monotonically non-decreasing. This implies that $\dot V^{(i)}(x,u^{(i+1)}) \le 0$ and that $V^{(i)}(x)$ is a Lyapunov function for $u^{(i+1)}$ on Ω. Following Definition 2.1, $u^{(i+1)}$ is admissible on Ω.

To show the second part of Lemma 3.1, note that for performance along trajectories $(f, g, u^{(i+1)})$ $\forall x_0$, we can write,

$$\begin{aligned} V^{(i+1)} - V^{(i)} &= \int_0^\infty \left\{ Q(x(\tau, x_0, u^{(i+1)})) + \left\| u^{(i+1)}(\tau, x_0, u^{(i+1)}) \right\|_R^2 \right\} d\tau - \\ &\quad \int_0^\infty \left\{ Q(x(\tau, x_0, u^{(i+1)})) + \left\| u^{(i)}(\tau, x_0, u^{(i+1)}) \right\|_R^2 \right\} d\tau \\ &= -\int_0^\infty \left\{ \frac{d(V^{(i+1)} - V^{(i)})^T}{dx} \left[f + g u^{(i+1)} \right] \right\} d\tau. \end{aligned} \tag{20}$$

From $GHJB(V^{(i+1)}, u^{(i+1)}) = 0$, $GHJB(V^{(i)}, u^{(i)}) = 0$, we have

$$\frac{\partial V^{(i)}(x)}{\partial x}^T f = -\frac{\partial V^{(i)}(x)}{\partial x}^T g u^{(i)} - l(x) - 2\int_0^{u^{(i)}} \left(\phi^{-1}(u)\right)^T R du \tag{21}$$

$$\begin{aligned} \frac{\partial V^{(i+1)}(x)}{\partial x}^T f &= -\frac{\partial V^{(i+1)}(x)}{\partial x}^T g u^{(i+1)} - l(x) - \\ &\quad 2\int_0^{u^{(i+1)}} \left(\phi^{-1}(u)\right)^T R du. \end{aligned} \tag{22}$$

Substituting (21), (22) in (20) we get

$$V^{(i+1)}(x_0) - V^{(i)}(x_0) = -2\int_0^\infty \left\{ \begin{array}{l} \left(\phi^{-1}(u^{(i+1)})\right)^T R(u^{(i+1)} - u^{(i)}) - \\ \int_{u^{(i)}}^{u^{(i+1)}} \left(\phi^{-1}(u)\right)^T R\, du \end{array} \right\} d\tau. \tag{23}$$

By decoupling the equation (24) using $R = \Lambda \Sigma \Lambda$, it can be shown that

$$V^{(i+1)}(x_0) - V^{(i)}(x_0) \le 0 \tag{25}$$

when $\phi(\cdot)$ is monotonically non-decreasing.

■

IV. Neural Network Least-Squares Approximate HJB Solution

Although equation (9) is linear in the value function when substituting (10) into (9) to improve the saturated control law, it is still difficult to solve for the cost function $V^{(i)}(x)$. Therefore, neural nets are now used to approximate the solution for the cost function $V^{(i)}(x)$ at each successive iteration i.

It is well known that neural networks can be used to approximate smooth functions on prescribed compact sets [12]. Since our analysis is restricted to a stability region, which is a compact set, neural networks are natural for our application. Therefore, to successively solve (9), (10) for bounded controls, we approximate $V^{(i)}(x)$ with a neural net

$$V_L^{(i)}(x) = \sum_{j=1}^L w_j^{(i)} \sigma_j(x) = W_L^{T(i)} \bar{\sigma}_L(x) \tag{26}$$

where the activation functions $\sigma_j(x): \Omega \to \Re$, are continuous, $\sigma_j(0) = 0$, span $\{\sigma_j\}_1^\infty \subseteq L_2(\Omega)$. The neural network weights are w_j and L is the number of hidden-layer neurons. Vectors $\bar{\sigma}_L(x) \equiv [\sigma_1(x)\, \sigma_2(x) \cdots \sigma_L(x)]^T$, $W_L \equiv [w_1\, w_2 \cdots w_L]^T$ are the vector activation function and the vector weight respectively. The neural network weights will be tuned to minimize the residual error in a least-squares sense over a set of points within the stability region Ω of the initial stabilizing control. Least-squares solution n attains the lowest possible residual error with respect to the neural network weights.

For the $GHJB(V, u) = 0$, the solution V is replaced with V_L having a residual error

$$GHJB\left(V_L(x) = \sum_{j=1}^L w_j^{(i)} \sigma_j(x), u\right) = e_L(x) \cdot \tag{27}$$

To find the least–squares solution, the method of weighted residuals is used [7]. The weights w_j are determined by projecting the residual error onto $d\, e_L(x)/d\, W_L$ and setting the result to zero $\forall x \in \Omega$, i.e.

$$\left\langle d\, e_L(x)/d\, W_L, e_L(x) \right\rangle = 0 \tag{28}$$

When expanded, equation (28) becomes,

$$\begin{aligned} &\left\langle \nabla\bar{\sigma}_L(f + gu), \nabla\bar{\sigma}_L(f + gu) \right\rangle W_L + \\ &\left\langle Q + 2\int \left(\phi^{-1}(u)\right)^T R du, \nabla\bar{\sigma}_L(f + gu) \right\rangle = 0 \end{aligned} \tag{29}$$

Expanding the derivative of the residual,

$$\begin{aligned} &\left\langle \nabla\bar{\sigma}_L(f + gu), \frac{d\sigma_j}{dx}(f + gu) \right\rangle w_j + \\ &\left\langle Q + 2\int \left(\phi^{-1}(u)\right)^T R du, \frac{d\sigma_j}{dx}(f + gu) \right\rangle = 0, \quad j = 1, \cdots, L. \end{aligned} \tag{30}$$

The following technical results are needed.

Lemma 4.1: if the set $\{\sigma_j\}_1^L$ is linearly independent and $u \in \Psi(\Omega)$, then the set

$$\left\{ \frac{d\sigma_j}{dx}(f + gu) \right\}_1^L \tag{31}$$

is also linearly independent.

Proof:

See [3].

■

From Lemma 4.1, equation (30) can be rewritten, after defining $\nabla\bar{\sigma}_L(f + gu) \triangleq \bar{\theta}$ as,

$$\begin{aligned} &\left\langle \nabla\bar{\sigma}_L(f + gu), \theta_j \right\rangle w_j + \\ &\left\langle Q + 2\int \left(\phi^{-1}(u)\right)^T R du, \theta_j \right\rangle = 0, \quad j = 1, \cdots, L. \end{aligned} \tag{32}$$

Because of Lemma 4.1, the term $\left\langle \bar{\theta}, \bar{\theta} \right\rangle$ is of full rank, and thus is invertible. Therefore a unique solution for W_L exists. We can solve equation (32) for W_L as follows,

$$W_L = -\left\langle \bar{\theta}, \bar{\theta} \right\rangle^{-1} \left\langle Q + 2\int \left(\phi^{-1}(u)\right)^T R du, \bar{\theta} \right\rangle. \tag{33}$$

Introducing a mesh on Ω, with mesh size equal to Δx, we can rewrite some terms of (33) as follows:

$$X = \left[\bar{\theta}\Big|_{x1} \cdots \bar{\theta}\Big|_{x_p} \right]^T \tag{34}$$

$$Y = \left[Q + 2\int \left(\phi^{-1}(u)\right)^T R du\Big|_{x1} \cdots\cdots Q + 2\int \left(\phi^{-1}(u)\right)^T R du\Big|_{x_p} \right]^T \tag{35}$$

where p in x_p represents the number of points of the mesh.

Finally, we can calculate W_L as

$$W_L = -\left(X^T X\right)^{-1}\left(X^T Y\right) \tag{36}$$

An interesting observation is that equation (36) is the standard least-squares method of estimation for a mesh on Ω. Note that the mesh size Δ should be such that the number of points p is greater than or equal to the order of approximation L. This guarantees a full rank for $\left(X^T X\right)$.

V. Illustrative Example

We start by applying the algorithm obtained above for the linear system

$$\dot{x}_1 = 2x_1 + x_2 + x_3,$$
$$\dot{x}_2 = x_1 - x_2 + u_2,$$
$$\dot{x}_3 = x_3 + u_1.$$

It is desired to control the system with a control bounds $|u_1| \le 3, |u_2| \le 20$. This system when uncontrolled has eigenvalues with positive real parts. This systems is not asymptotically null controllable, therefore global asymptotic stabilization cannot be achieved, [23].

The following smooth function is used to approximate the value function of the system,

$$\begin{aligned} V_{21}(x_1, x_2, x_3) = {} & w_1 x_1^2 + w_2 x_2^2 + w_3 x_3^2 + w_4 x_1 x_2 + w_5 x_1 x_3 + \\ & w_6 x_2 x_3 + w_7 x_1^4 + w_8 x_2^4 + w_9 x_3^4 + w_{10} x_1^2 x_2^2 + w_{11} x_1^2 x_3^2 + \\ & w_{12} x_2^2 x_3^2 + w_{13} x_1^2 x_2 x_3 + w_{14} x_1 x_2^2 x_3 + w_{15} x_1 x_2 x_3^2 + \\ & w_{16} x_1^3 x_2 + w_{17} x_1^3 x_3 + w_{18} x_1 x_2^3 + w_{19} x_1 x_3^3 + w_{20} x_2 x_3^3 + \\ & w_{21} x_2^3 x_3 \end{aligned}$$

The number of neurons required is chosen to guarantee the uniform convergence of the algorithm. To initialize the algorithm, a stabilizing control is needed. It is very easy to find this using LQR for unconstrained controls. A stabilizing unconstrained state feedback control is found

$$u_1 = -8.31x_1 - 2.28x_2 - 4.66x_3,$$
$$u_2 = -8.57x_1 - 2.27x_2 - 2.28x_3,$$

However, when this controller is applied through saturated actuators, the stability region shrinks, and the control law is not optimal anymore.

Fig. 1 shows the performance of this controller assuming working with unsaturated actuators. Fig. 2 shows the performance when this control signal is bounded by $|u_1| \le 3, |u_2| \le 20$. Note how the bounds destroy the performance.

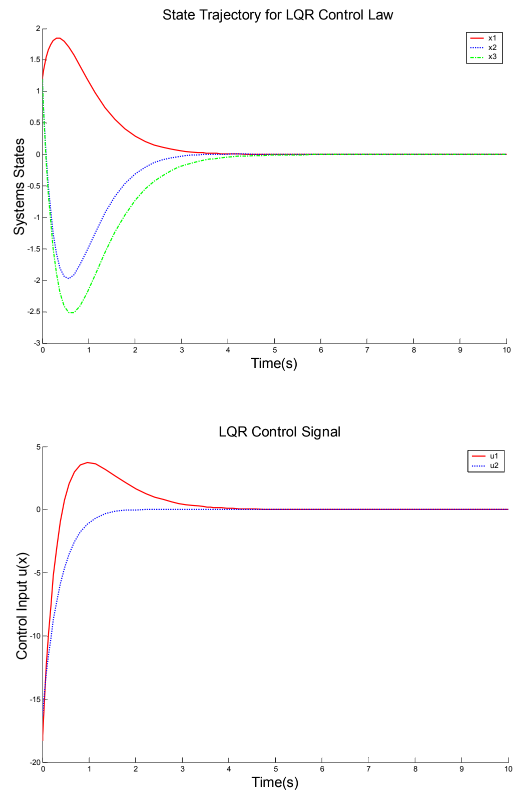


Fig. 1. LQR optimal unconstrained control

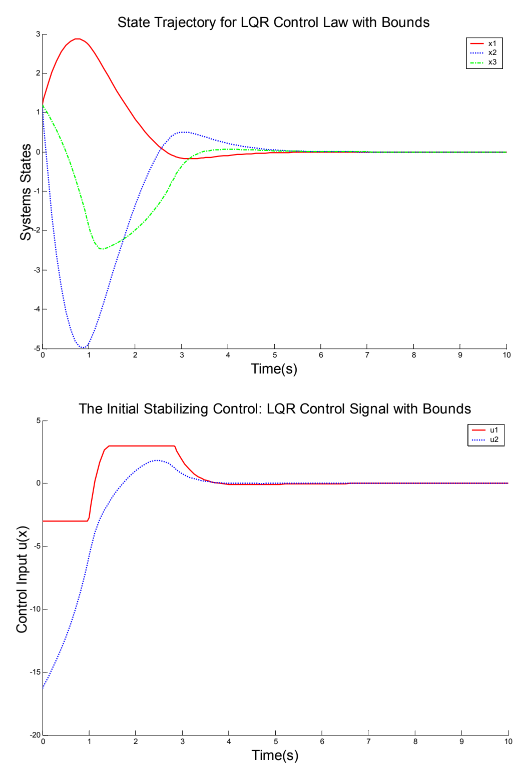
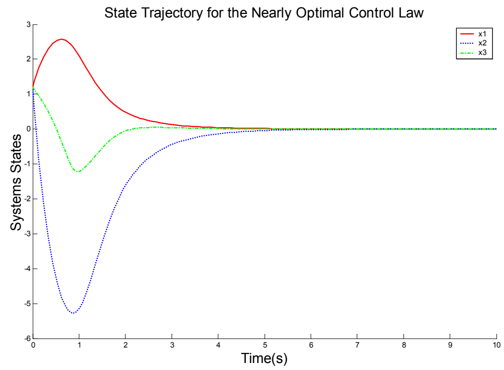


Fig. 2. LQR control with actuator saturation



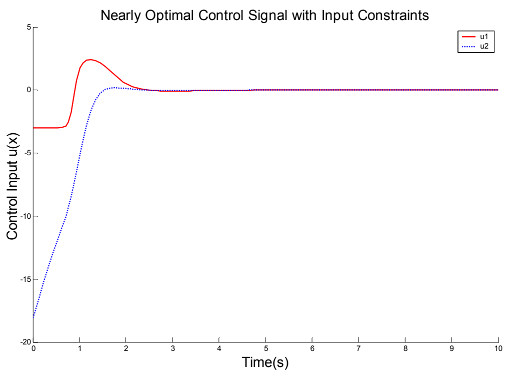


Fig. 3. Nearly optimal nonlinear neural control law considering actuator saturation

The nonquadratic cost performance is

$$\begin{aligned} W(u) &= 2\int \left(\phi^{-1}(u)\right)^T R\,du \\ &= 2\int \left(A\tanh^{-1}\left(u/A\right)\right)^T R\,du \\ &= 2\cdot \mathrm{A}\cdot \mathrm{R}\cdot \mathrm{u}\cdot \tanh^{-1}\left(u/A\right) + \mathrm{A}^2\cdot \mathrm{R}\cdot \ln\left(1-u^2/A^2\right) \end{aligned}$$

where A, is the saturation limit.

This nonquadratic cost performance is then used in the algorithm to calculate the optimal bounded control.

The neural net based nearly optimal saturated optimal control law is found to be,

$$u_1(x) = -3\tanh\left(\frac{1}{3}\begin{Bmatrix} 7.70x_1 + 2.44x_2 + 4.75x_3 + 2.45x_1^3 + \\ 2.27x_1^2x_2 + 3.73x_1x_2x_3 + 0.708x_1x_2^2 + \\ 5.78x_1^2x_3 + 4.78x_1x_3^2 + 0.08x_2^3 + \\ 0.57x_2^2x_3 + 1.56x_2x_3^2 + 1.39x_3^3 \end{Bmatrix}\right)$$

$$u_2(x) = -20\tanh\left(\frac{1}{20}\begin{Bmatrix} 9.78x_1 + 2.94x_2 + 2.44x_3 - \\ 0.21x_1^3 - 0.02x_1^2x_2 + 1.42x_1x_2x_3 + \\ 0.12x_1x_2^2 + 2.27x_1^2x_3 + 1.87x_1x_3^2 + \\ 0.02x_2^3 + 0.23x_2^2x_3 + 0.57x_2x_3^2 + 0.52x_3^3 \end{Bmatrix}\right)$$

This result is obtained after 20 successive iterations. The algorithm is run over the region $-1.2 \le x_1 \le 1.2$, $-1.2 \le x_2 \le 1.2$, $-1.2 \le x_3 \le 1.2$ with the design parameters $R = I_{2x2}$, $Q = I_{3x3}$. The performance is shown in fig 3.

References

[1] M. Abu-Khalaf , F. L. Lewis, "Nearly Optimal HJB Solution for Constrained Input Systems Using a Neural Networks Least-Squares Approach," *Proc. IEEE Conference on Decision and Control,* , pp 943-948, December 2002.

[2] R. Beard, G. Saridis, J. Wen, "Approximate Solutions to the Time-Invariant Hamilton-Jacobi-Bellman Equation," *Journal of Optimization Theory and Application*, Vol 96, No. 3, March 1998, pp. 589-626.

[3] R. Beard, G. Saridis, J. Wen, "Galerkin Approximations of the Generalized Hamilton-Jacobi-Bellman Equation," *Automatica* 33:12, December, pp. 2159-2177, 1997.

[4] R. Beard, "Improving the Closed-Loop Performance of Nonlinear Systems," PhD thesis, Rensselaer Polytechnic Institute, Troy, NY 12180, 1995.

[5] D. S. Bernstein, "Optimal nonlinear, but continuous, feedback control of systems with saturating actuators," *International Journal of Control,* Vol 62, NO. 5, pp. 1209-1216.

[6] D. P. Bertsekas, J. N. Tsitsiklis, "*Neuro-Dynamic Programming,*" Athena Scientific, Belmont, MA, 1996.

[7] B. A. Finlayson, "The Method of Weighted Residuals and Variational Principles," Academic Press, New York, NY, 1972.

[8] J. Huang, C. F. Lin, "Numerical Approach to Computing Nonlinear H_∞ Control Laws," *Journal of Guidance, Control, and Dynamics,* vol. 18, no. 5, September-October 1995.

[9] Y. H. Kim, F. L. Lewis, D. Dawson, "Intelligent optimal control of robotic manipulators using neural networks," *Automatic 36*, 2000, pp. 1355 – 1364.

[10] D. Kleinman, "*On an iterative Technique for Riccati Equation Computations,*" *IEEE Trans. Automatic Control*, pp. 114-115, February 1968.

[11] F. L. Lewis, V. L. Syrmos, "*Optimal Control*," John Wiley & Sons, Inc. New York, NY, 1995.

[12] F. L. Lewis, S. Jagannathan, A. Yesildirek, "Neural Network Control of Robot Manipulators and Nonlinear Systems," Taylor & Francis, 1999.

[13] X. Liu, S. N. Balakrishnan, "Convergence Analysis of Adaptive Critic Based Optimal Control," *Proc. American Control Conference*, June. 2000, pp.1929 – 1933.

[14] S. E. Lyshevski, "*Control Systems Theory with Engineering Applications,*" Birkhauser, Boston, MA,2001.

[15] S. E. Lyshevski, A. U. Meyer, "Control System Analysis and Design Upon the Lyapunov Method," *Proc. American Control Conference*, June. 1995, pp. 3219 - 3223.

[16] S. E. Lyshevski, "Optimal Control of Nonlinear Continuous-Time Systems: Design of Bounded Controllers Via Generalized Nonquadratic Functionals," *Proc. American Control Conference*, June. 1998, pp.205 – 209.

[17] S. E. Lyshevski, "Role of Performance Functionals in Control Laws Design," *Proc. American Control Conference,* June. 2001, pp. 2400 - 2405.

[18] S. E. Lyshevski, "Constrained Optimization and Control of Nonlinear Systems: New Results in Optimal Control," *Proc. IEEE Conference on Decision and Control*, December. 1996, pp. 541 - 546.

[19] R. Munos, L. C. Baird, A. Moore, "Gradient Descent Approaches to Neural-Net-Based Solutions of the Hamilton-Jacobi-Bellman Equation," *International Joint Conference on Neural Networks IJCNN*, 1999, Vol 3, pp. 2152 -- 2157.

[20] J. Murray, C. Cox, R. Saeks and G. Lendaris, "Globally Convergent Approximate Dynamic Programming Applied to an Autolander," *Proc. American Control Conference*, June. 2001, pp.2901 – 2906.

[21] A. Saberi, Z. Lin, A. Teel, "Control of Linear Systems with Saturating Actuators," *IEEE Transactions on Automatic Control,* Vol 41, NO. 3, March 1996, pp. 368 -378.

[22] G. Saridis, C. S. Lee, "An Approximation Theory of optimal Control for Trainable Manipulators," *IEEE Trans. Systems, Man, Cybernetics*, Vol. 9, No. 3, pp. 152-159, March 1979.

[23] H. Sussmann, E. D. Sontag, Y. Yang, "A General Result on the Stabilization of Linear Systems Using Bounded Controls," *IEEE Trans. Automatic Control*, Vol. 39, No. 12, pp. 2411-2425, December 1994.