Nonlinear stabilization by receding-horizon neural regulators

T. PARISINI†, M. SANGUINETI‡ and R. ZOPPOLI‡

A receding-horizon (RH) optimal control scheme for a discrete-time nonlinear dynamic system is presented. A non-quadratic cost function is considered and constraints are imposed on both the state and control vectors. A stabilizing regulator is derived by adding a proper terminal penalty function to the process cost. The control vector is generated by means of a feedback control law computed off-line instead of computing it on-line, as is done for existing RH regulators. The off-line computation is performed by approximating the RH regulator by a multilayer feedforward neural network. Bounds to this approximation are established. Algorithms are presented to determine some essential parameters for the design of the neural regulator, i.e. the parameters characterizing the terminal cost function and the number of neural units in the network implementing the regulator. Simulation results show the effectiveness of the proposed approach.

1. Introduction

In the past few years, neural networks have been widely used as nonlinear approximations to solve nonlinear functional optimization problems such as those associated with the areas of optimal control, state estimation, parameter estimation, etc. Their employment is substantially motivated by the fact that the solution of any such problems is given by an unknown function (e.g. a feedback control law, a state or parameter estimator, etc.) that cannot be obtained analytically, due to the very general assumptions under which the problem has been stated. The unknown function is then given a fixed structure (i.e. the one of a multilayer feedforward neural network, or of a radial basis function, etc.) in which a large but finite number of parameters must be optimized. Therefore, it is possible to approximate the original infinite-dimension optimization problem by a finite-dimension nonlinear programming one.

Such an approach is anything but new in control theory; it recalls, for example, the so-called specific optimal control proposed by Einsenberg and Sage (1966). The choice of feedforward networks instead of traditional nonlinear approximators, like polynomial or trigonometric expansions, has been suggested by the fact that neural approximators do not involve the so-called phenomenon of the curse of dimensionality, in that the number of parameters to be tuned grows only linearly and not exponentially with the dimension of the argument vector of the functions to be approximated, provided that these functions belong to classes with suitable smoothness characteristics. Note, however, that most nonlinear approximators share this property (Girosi et al. 1995, Parisini and Zoppoli 1995). The approxima-
tion described above has been accepted for many of the aforesaid problems, in the cases where the control or estimation process lasted for a finite number of temporal stages (we refer to discrete-time dynamic systems). Of course, things become much more complicated if the control or estimation horizon goes to infinity, thus involving asymptotic stability issues.

The purpose of this paper is to establish to what extent stabilization and neural approximation are compatible concepts. An analysis will be made with reference to a receding-horizon (RH) neural regulator for nonlinear dynamic systems. The RH control scheme can be described as follows. When the controlled plant is in the state $x_t$ at stage $t$, a finite-horizon (FH) $N$-stage optimal control problem is solved, thus the sequence of optimal control vectors $u_{t}^{FH}, \ldots, u_{t+N-1}^{FH}$ is derived, and the first control of this sequence becomes the control action $u_{t}^{RH}$ generated by the RH regulator at stage $t$ (i.e. $u_{t}^{RH} = u_{t}^{FH}$). This procedure is repeated stage after stage and a feedback control law is obtained, as the control vector $u_{t}^{FH}$ depends on the current state $x_t$.

Stabilizing properties of RH regulators for nonlinear systems were derived by Chen and Shaw (1982), Mayne and Michalska (1990), and Michalska and Mayne (1993) for continuous-time systems and by Keerthi and Gilbert (1988) for discrete-time systems.

In this paper we further develop the design procedure of RH neural regulators (Parisini and Zoppoli 1995) that stabilize nonlinear systems while minimizing a certain cost function (in general, non-quadratic). Constraints may be imposed on both the state and control vectors. For the solution of optimal control problems by neural approximators over an $N$-stage finite horizon, we refer to Zoppoli and Parisini (1992) and Parisini and Zoppoli (1994). The main contributions of the present work can be summarized as follows.

(i) The RH stabilizing optimal regulator is derived without imposing either the exact constraint $x_{t+N} = 0$ (Keerthi and Gilbert 1988, Mayne and Michalska 1990), or the condition of reaching a neighbourhood $W$ of the origin, where the RH nonlinear regulator switches to a linear stabilizing one (Michalska and Mayne 1993). Instead, in our approach, the attractiveness is imposed by means of a suitable penalty function.

(ii) In the works by Keerthi and Gilbert and by Mayne and Michalska, the control vectors were assumed to be generated by an on-line computation, which was implemented as soon as a certain state $x_t$ was reached at stage $t$. However, if the dynamics of the controlled plant is not sufficiently slow, as compared with the speed of the regulator’s computing system, the on-line computation of the control vectors turns out to be unfeasible. Then we propose to compute off-line the RH closed-loop optimal control law $u_{t}^{RH} = \gamma_{RH}^{0}(x_t)$ that enables the RH regulator to generate instantaneously the control vector $u_{t}^{RH}$ for any vector $x_t$ belonging to the set $X$ of admissible states.

(iii) Within our general non-LQ context, deriving analytically the function $\gamma_{RH}^{0}(x_t)$ is practically an impossible task. At least in principle, this function could be determined in the backward phase of a dynamic programming procedure. However, due to the well-known computational drawbacks of dynamic programming, we do not resort to this algorithm and present an approximate approach to the problem. This approach consists in approximating $\gamma_{RH}^{0}(x_t)$ by a control function of
the form $\gamma_{RH}(x_t, w)$, where $\gamma_{RH}$ is the input/output mapping of a multilayer feedforward neural network and $w$ is a vector of parameters to be tuned.

(iv) Special attention is given to the computational aspects involved in the design of RH regulators, namely, the determination of the penalty function attracting the system state to the origin and the neural network implementing the control law.

The paper is organized as follows: in §2 the RH optimal control problem is stated, and in §3 the stabilizing properties of the resulting RH optimal regulator are established. In §4 we address the problem of deriving a neural approximation for the RH optimal regulator and we present approximation bounds. The algorithms to compute the parameters needed for the design of the neural regulator are described in §5. Simulation results are reported in §6.

2. The receding-horizon optimal control law

Let us consider the discrete-time dynamic system (in general, nonlinear)

$$x_{t+1} = f(x_t, u_t), \quad t = 0, 1, \ldots$$

where $x_t \in X \subset \mathbb{R}^n$ and $u_t \in U \subset \mathbb{R}^m$ are the state and control vectors, respectively. We assume that the constraint regions $X$ and $U$ belong to the class $\mathcal{Z}$ of compact sets containing the origin as an internal point. In the following, we define $u_t := \text{col}(u_t, \ldots, u_t)$ for finite values of the integer $\tau \geq t$. Let us assume that $f \in \mathcal{C}^1[\mathbb{R}^n \times \mathbb{R}^m, \mathbb{R}^n]$ with $f(0, 0) = 0$, and that $h \in \mathcal{C}^1[\mathbb{R}^n \times \mathbb{R}^m, \mathbb{R}^+]$ with $h(0, 0) = 0$. We now define the following finite-horizon cost function

$$J_{FH}(x_t, u_t, t+1, N, a, P) = \sum_{i=t}^{t+N-1} h(x_i, u_i) + a\|x_{t+N}\|^2_P, \quad t \geq 0$$

where $\|x\|^2_P = x^T P x, a \in \mathbb{R}$ is a positive scaler, $P \in \mathbb{R}^{n \times n}$ is a positive-definite symmetric matrix, and $N$ is a positive integer denoting the length of the control horizon. Then we can state the following.

**Problem 1:** At each time instant $t \geq 0$, find the RH optimal control law $u_{RH}^0 = \gamma_{RH}^0(x_t) \in U$, where $u_{RH}^0$ is the first vector of the control sequence, $u_{FH}^0, \ldots, u_{FH}^{N-1}$ (i.e. $u_{RH}^0 = u_{FH}^0$), that minimizes the cost (2) for the state $x_t \in X$.

To derive the results given in the following, we assume the solution of Problem 1 and, consequently, the optimal state trajectory $x_{FH}^0$ to exist and be unique. The statement of Problem 1 does not impose any particular way of computing the control vector $u_{RH}^0$ as a function of $x_t$. Actually, we have two possibilities to determine the optimal control law numerically:

1. **On-line computation.** Problem 1 is an open-loop optimal control problem and may be regarded as a nonlinear programming one. The main advantage of this approach (adopted in the works of Keerthi and Gilbert and by Mayne and Michalska) is that many well-established nonlinear programming techniques are available to solve Problem 1.

2. **Off-line computation.** This approach implies that the control law $\gamma_{RH}^0(x_t)$ must be computed a priori and stored in the regulator’s memory. Clearly, the off-line computation has advantages and disadvantages that are opposite the ones of the on-line approach. No on-line computational effort is requested.
from the regulator, but a very large amount of computer memory may be required to store the closed-loop control law (as regards the neural control functions, we must store the weight parameters in the memory).

3. Stabilizing properties of the receding-horizon regulator

Let us make the following assumptions.

**Assumption 1:** The linear system $x_{t+1} = Ax_t + Bu_t$, obtained via the linearization of the system (1) in the neighbourhood of the origin, i.e.

$$A \triangleq \frac{\partial f}{\partial x_t} \bigg|_{x_t=0,u_t=0} \quad \text{and} \quad B \triangleq \frac{\partial f}{\partial u_t} \bigg|_{x_t=0,u_t=0}$$

is stabilizable.

**Assumption 2:** The transition cost function $h(x,u)$ depends on both $x$ and $u$, and there exist two strictly increasing functions $r,s \in \mathbb{R}^+[\mathbb{R}^+]$ with $r(0) = s(0) = 0$, such that

$$r(\|(x,u)\|) \leq h(x,u) \leq s(\|(x,u)\|)^2, \quad \forall x \in X, \forall u \in U$$

where $(x,u) \triangleq \text{col}(x,u)$.

**Assumption 3:** There exists a compact set $X_0 \subseteq X$, $X_0 \in \mathbb{Z}$, with the property that there exists a control horizon $M \geq 1$ such that there exists a sequence of admissible control vectors $\{u_i \in U, i = t, \ldots, t + M - 1\}$ that yield an admissible state trajectory $x_i \in X, i = t, \ldots, t + M$ ending in the origin of the state space (i.e. $x_{t+M} = 0$) for any initial state $x_t \in X_0$.

**Assumption 4:** For any integer $N \geq 1$ and for any $x_t \in X$, the optimal FH feedback control functions $\gamma^0_{FH}(x_t,i), i = t, \ldots, t + N - 1$, which minimize the cost (2), are continuous with respect to $x_i$.

Note that Assumption 3 substantially concerns the controllability of the non-linear system (1). Let us now denote by

$$J^0_{FH}(x_t,N,a,P) = J_{FH}(x_t,u^0_{t+N-1},N,a,P) = \sum_{i=t}^{t+N-1} h(x^0_{i},u^0_{i},P) + h_F(x^0_{t+N})$$

the cost corresponding to the optimal $N$-stage trajectory starting from $x_t$, i.e. $x^0_{t+N} = x_t$ (to simplify the notation, we let $h_F(x) \triangleq a \|x\|_P^2$, without any ambiguity whenever $a$ and $P$ need not be rendered explicit). Then the following theorem can be proved.

**Theorem 1:** If Assumptions 1–4 are verified, there exist a positive scalar $\widetilde{a}$ and a positive-definite symmetric matrix $P \in \mathbb{R}^{n \times n}$ such that, for any $N \geq M$ and any $a \in \mathbb{R}, a \geq \widetilde{a}$, the following properties hold:

(a) the RH control law stabilizes asymptotically the origin, which is an equilibrium point of the resulting closed-loop system

(b) there exists a positive scalar $\beta$ such that the set

$$\mathcal{W}(N,a,P) \in \mathbb{Z}, \quad \mathcal{W}(N,a,P) \triangleq \{x \in X : J^0_{FH}(x,N,a,P) \leq \beta\}$$
is an invariant subset of $X_0$ and a domain of attraction for the origin, i.e. for any $x_t \in W(N,a,P)$, the state trajectory generated by the RH regulator remains entirely contained in $W(N,a,P)$ and converges to the origin.

Theorem 1 is a simplified and more constructive version of a similar theorem presented by Parisini and Zoppoli (1995). In the following, we sketch its proof, which is based partly on some facts established by Parisini and Zoppoli (1995) and partly on a new approach aimed at determining the scalar $\tilde{a}$ and the matrix $P$ explicitly, thus making the design of the RH regulator implementable (such a determination will be presented in § 5). Theorem 1 relies essentially on the demonstration that the cost $J_{FH}^0(x_t,N,a,P)$ is a Lyapunov function in $X_0$ for the system (1) driven by the RH regulator. This property is easy to assess (Parisini and Zoppoli 1995), once the following lemma has been proved (note that this lemma is slightly different from the one presented by Parisini and Zoppoli 1995).

**Lemma:** There exist a positive-definite symmetric matrix $P \in \mathbb{R}^{n \times n}$ and a positive scalar $\tilde{a}$ such that

$$J_{FH}^0(x_t,N,a,P) \geq J_{FH}^0(x_t,N+1,a,P), \quad \forall x_t \in X_0, \forall N \geq M, \forall a \geq \tilde{a}$$

**Proof:** To prove this lemma, we need the following two facts.

**Fact 1:** If Assumption (1) is verified, there exists a matrix $K \in \mathbb{R}^{m \times n}$ such that the origin is an asymptotically stable equilibrium point of the closed-loop system $x_{t+1} = f(x_t, Kx_t)$. Moreover, for such a system, there exists a compact set $W(K,P) \subseteq X$, $W(K,P) \in \mathbb{R}^n$, which is an invariant set and a domain of attraction for the origin ($P \in \mathbb{R}^{n \times n}$ is a suitable positive-definite symmetric matrix with the property that $x^T P x$ is a Lyapunov function for $x_{t+1} = f(x_t, Kx_t)$ in $W(K,P)$).

**Fact 2:** Consider the matrices $K$ and $P$, as established in Fact 1. Then there exists a positive scalar $\hat{a}$ such that

$$h_{F}(x_t) \geq h(x_t, Kx_t) + h_F(x_{t+1}), \quad \forall x_t \in W(K,P), \forall a \geq \hat{a}$$

**Proof:** The proof of Fact 1 is a discrete-time version of the one given by Michalska and Mayne (1993), and Fact 2 has been proved by Parisini and Zoppoli (1995). In section 5 an alternative proof of Fact 2 will be presented that is not only an existence proof, but also allows one to practically determine the above scalar $\hat{a}$.

Now, to prove the lemma, consider the FH optimal control problem

$$\text{minimize } \sum_{i=t}^{t+N-1} h(x_i, u_i), \quad \text{subject to } x_{t+N} = 0$$

with $N \geq M$ and $\forall x_t \in X_0$ (constraints on $x_t$ and $u_t$ are understood). By writing $x_{t+N}$ as a function of $x_t$ and of the control vectors, (3) may be rewritten

$$\text{minimize } J(x_t, u), \quad \text{subject to } F(x_t, u) = 0$$

where $u \perp u_{t+1}$ and the functions $F$ and $J$ have clear meanings. In virtue of Assumption 3, the problem (4) has an optimal solution for any $x_t \in X_0$ (assume this solution to be unique). We now recall that the FH optimal control problem introduced in Problem 1 is given by

$$\text{minimize } J(x_t, u) + d \| F(x_t, u) \|^2_P$$

(5)
Clearly, \( \|F(x_t, u_k)\|_p^2 \) has the characteristics of a penalty function. Let \( \{a_k, k = 1, 2, \ldots\} \) be a sequence tending to infinity, with \( a_{k+1} > a_k \) for each \( k \). Then, the penalty-functions method generates the sequence of optimal solutions \( \{\hat{u}_k, k = 1, 2, \ldots\} \) of the problem (5) (we assume that this problem has a unique optimal solution for each \( k \)). It is now easy to show that
\[
\lim_{k \to \infty} \|F(x_t, u_k)\|_p = 0
\]
hence
\[
\lim_{k \to \infty} x_{i+N}^k = 0
\]
(\( x_{i+N}^k \) is the final state of the trajectory generated by \( u_k \)). It follows that there exists a sufficiently large integer \( k^* \) (and then a scalar \( a_{k^*} \)) such that the optimal solution \( u_{k^*} \) drives \( x_{i+N}^{k^*} \) into the subset \( W(K, P) \) (see Fact 1). As \( a_{k^*} \) depends on \( x_t \), we let
\[
a^* = \max_{x_t \in \mathcal{X}_0} a_{k^*}(x_t)
\]
(6)
We can conclude that the optimal solution of the problem (5) (i.e. \( u_{t+1}^{FH}, \ldots, u_{t+N-1}^{FH} \)) makes
\[
x_{t+N}^{FH} \in W(K, P), \quad \forall N \geq M, \forall a \geq a^*, \forall x_t \in \mathcal{X}_0
\]
We now add to the optimal trajectory a further stage generated by the control \( Kx_{t+N}^{FH} \). As \( x_{t+N} \in W(K, P) \), we can use Fact 2, thus obtaining
\[
J_{FH}(x_t, u_{t+N}, N+1, a, P) = J_{FH}^0(x_t, N, a, P) - h_{FH}(x_{t+N}^{FH}) + h_{FH}(x_{t+N}, Kx_{t+N}^{FH}) + h_{FH}(x_{t+N+1}^{FH})
\]
\[
\leq J_{FH}^0(x_t, N, a, P), \quad \forall a \geq \tilde{a} = \max \{a, a^*\}
\]
Then, thanks to the optimality of \( J_{FH}^0(\cdot) \), we have
\[
J_{FH}(x_t, N + 1, a, P) \leq J_{FH}(x_t, u_{t+N}, N+1, a, P) \leq J_{FH}(x_t, N, a, P),
\]
\( \forall x_t \in \mathcal{X}_0, \forall N \geq M, \forall a \geq \tilde{a} \)
which proves the lemma.

It is now worth noting that, in deriving (both on-line and off-line) the optimal RH control law \( u_{t}^{RH} = \gamma_{RH}^0(x_t) \), computational errors may affect the vector \( u_{t}^{RH} \) and possibly lead to a closed-loop instability of the origin. Analogously, as will be shown in the next section, errors on \( u_{t}^{RH} \) may be induced by the neural network that approximate the RH regulator. Therefore, we need to address the sensitivity of the state trajectory when errors affect the control law \( \gamma_{RH}^0 \). This is done by means of the following theorem, which characterizes the properties of the RH regulator when suboptimal control vectors \( \hat{u}_{t}^{RH} \in U, i \geq t \), are used in the RH control scheme, instead of the optimal ones \( u_{t}^{RH} \) solving Problem 1. Let us denote by \( \hat{x}_{i}^{RH}, i > t \), the state vectors belonging to the suboptimal RH trajectory starting from \( x_t \).

**Theorem 2:** If Assumptions 1–4 are verified, there exist a positive scalar \( \tilde{a} \) and a positive-definite symmetric matrix \( P \in \mathbb{R}^{m \times n} \) such that, for any \( N \geq M \) and for any \( a \geq \tilde{a} \), the following properties hold:

(a) There exist suitable scalars \( \hat{\delta}, \tilde{\delta} \in \mathbb{R}_+ \) such that, if
\[
\|u_t^{RH} - \hat{u}_t^{RH}\| \leq \hat{\delta}, i \geq t, \quad \hat{x}_i^{RH} \in \mathcal{W}(N, a, P), \quad \forall t > i, \forall x_t \in \mathcal{W}(N, a, P)
\]
Among various possible approximating functions, multilayer feedforward neural

As the proof of Theorem 2 is a direct consequence of the regularity assumptions on the state equation, we do not report it here (for computational details, see Parisini and Zoppoli (1993)). However, analogously to what previously done for Fact 2, an alternative proof of Theorem 2 will be presented in section 5, aimed at practically determining the scalars $\tilde{\alpha}_i, i \geq t$. Theorem 2 has the following meaning: provided that the errors on the control vectors are suitably bounded, $w(N, a, P)$ still remains an invariant set. Moreover, the RH regulator can drive the state into each desired neighbourhood $w_d$ of the origin in a finite time ($w_d$ is an invariant set, too). Finally, using the hybrid control scheme described by Michalska and Mayne (1993), $w(N, a, P)$ becomes a domain of attraction for the origin.

4. The neural approximation for the receding-horizon regulator

As stated above, we are mainly interested in computing the RH control law $u_t^{RH} = \gamma_{RH}(x_t)$ off-line. Then we need to derive a priori an FH closed-loop optimal control law

$$u^0_i = \gamma_{FH}(x_t, i), \quad t \geq 0, i = t, \ldots, t + N - 1$$

that minimizes the cost (2) for any $x_t \in X$. Because of the time invariances of the dynamic system (1) and of the cost function (2), from now on we consider the control functions

$$u^{FH} = \gamma_{FH}(x_t, i - t), \quad t \geq 0, i = t, \ldots, t + N - 1$$

instead of $u^{FH} = \gamma_{FH}(x_t, i)$, and state the following.

Problem 2: Find the FH optimal feedback control law

$$\{u^i_{FH} = \gamma_{FH}(x_t, i - t) \in U, t \geq 0, i = t, \ldots, t + N - 1\}$$

that minimizes the cost (2) for any $x_t \in X$.

Then, once the solution of Problem 2 has been achieved, we consider only the first optimal control function (i.e. the one corresponding to $i = t$), and write

$$u_t^{RH} = \gamma_{RH}(x_t) = \gamma_{FH}(x_t, 0), \quad \forall x_t \in X, t \geq 0$$

(7)

As mentioned above, we do not use dynamic programming to solve Problem 2, as this algorithm exhibits well-known computational drawbacks. As we have the possibility of computing (off-line) any number of open-loop optimal control sequences $u_t^{FH}, \ldots, u_t^{FH+N-1}$ (see Problem 1) for different vectors $x_t \in X$, we propose to approximate the function $\gamma_{RH}(x_t) = \gamma_{FH}(x_t, 0)$ by means of a function $\hat{\gamma}_{RH}(x_t, w)$, to which we assign a given structure. $w$ is a vector of parameters to be optimized. A possible way to derive the approximate RH neural control law consists in finding a vector $w^0$ that minimizes the integrated square approximation error

$$E(w) = \int_X \| \gamma_{RH}(x_t) - \hat{\gamma}_{RH}(x_t, w) \|^2 \, dx_t$$

(8)

Among various possible approximating functions, multilayer feedforward neural
networks may be chosen. The reason for this choice will be given at the end of this section. A feedforward network is composed of \( L \) layers, and in the generic layer \( s \) neural units are active. The input/output mapping of the \( q \)th neural unit of the \( s \)th layer is given by

\[
y_q(s) = g \left[ \sum_{p=1}^{n_{s-1}} w_{pq}(s) y_p(s-1) + w_{0q}(s) \right], \quad s = 1, \ldots, L, q = 1, \ldots, n_s
\]

where \( y_q(s) \) is the output variable of the neural unit, and \( w_{pq}(s) \) and \( w_{0q}(s) \) are the weight and bias coefficients, respectively. We use the activation function \( g(x) = \tanh(x) \). All these coefficients are the components of the vector \( w \); the variables \( y_q(0) \) are the components of \( x_t \), and the variables \( y_q(L) \) are the components of \( u_t \).

To exploit the results given by Theorem 2, we have now to specify quantitatively the magnitude of the errors generated by the control law of the form

\[
\hat{u}^{\text{RH}}_t = \gamma^{(0)}_{\text{RH}}(x_t, w),
\]

when it approximates the control law (7). To this end, we need to specialize the above-described neural network. More precisely, we assume that the approximating neural function \( \hat{y}^{(0)}_{\text{RH}}(x_t, w) \) contains only one hidden layer (i.e. \( L = 2 \)) composed of \( \nu \) neural units, and that the output layer is composed of linear activation units. We denote such a function by \( \hat{y}^{(0)}_{\text{RH}}(x_t, w) \). From the approximation property of feedforward neural networks based on sigmoidal functions, as reported for instance by Hornik et al. (1989), it is easy to derive the following.

**Theorem 3:** Assume that, in the solution of Problem 2, the first control function \( \gamma^{(0)}_{\text{RH}}(x_t) = \gamma^{(0)}_{\text{FF}}(x_t, 0) \) (see (7)) exists and is unique, and that it is a \( \mathbb{C}[X, \mathbb{R}^m] \) function. Then, if Assumptions 1–4 are verified, there exists an RH neural regulator \( \hat{u}^{\text{RH}}_t = \hat{y}^{(0)}_{\text{RH}}(x_t, w), t \geq 0 \), for which the two properties in Theorem 2 hold true.

Theorem 3 allows us to assess the existence of an RH neural regulator able to drive the system state, in a finite time, into any desired neighbourhood \( \mathcal{W}_d \) of the origin. This is obtained by constraining the control vectors \( \hat{u}^{\text{RH}}_t \) to take their values from the admissible set \( \bar{U} = \{ u : u + \Delta u \in \bar{U}_s, ||\Delta u|| \leq \varepsilon \} \), where for a given \( \mathcal{W}_d \) and any \( x_t \in \mathcal{W}(N_s, \alpha_s, P) \), \( \varepsilon \) is such that \( \varepsilon \leq \delta_s, i \geq t \) (see the scalars in Theorem 2), and \( \bar{U} \in \mathbb{R}^\nu \). Then we need a guarantee that the control errors generated by \( \hat{u}^{\text{RH}}_t = \gamma^{(0)}_{\text{RH}}(x_t, w^0) \) are uniformly bounded on \( X \) by the scalar \( \varepsilon \). Such a guarantee may be obtained by solving the minimax problem stated below. First, we have to define an approximating network that differs slightly from the one considered to state Theorem 3. The new network is the parallel of \( m \) single-output neural networks of the type described above (i.e. containing a single hidden layer and linear output activation units). Each network generates one of the \( m \) components of the control vector \( \hat{u}^{\text{RH}}_t \). We denote by \( \gamma^{(0)}_{\text{RH}}(x_t, w_j) \) the input–output mapping of the \( j \)th of such networks, where \( w_j \) is the number of neural units in the hidden layer and \( w_j \) is the weight vector, and we denote by \( \gamma^{(0)}_{\text{RH}}(x_t) \) the \( j \)th component of the vector function \( \gamma^{(0)}_{\text{RH}} \). Now we can state the following.

**Problem 3:** For each function \( \gamma^{(0)}_{\text{RH}}(x_t, w_j) \), find the numbers \( v^*_1, \ldots, v^*_m \) of neural units such that

\[
\min_{w_j} \max_{x_t \in X} \left| \gamma^{(0)}_{\text{RH}}(x_t) - \gamma^{(0)}_{\text{RH}}(x_t, w_j) \right| \leq \frac{\varepsilon}{\sqrt{m}}, \quad j = 1, \ldots, m
\]

In section 5, we shall address the possibility of computing the scalar \( \varepsilon \). As to the numbers \( v^*_j \), rather a naive trial-and-error procedure for determining them is the
following: for each \( j \), increase \( \nu_j \) until the term on the left-hand side of (9) is less than or equal to \( \varepsilon / \sqrt{n} \). This procedure seems reasonable, as a bound (uniform on \( X \)) to 
\[
| \gamma_{RH}^0(x,t) - \hat{\gamma}_{RH}^{(uj)}(x_t,w_j) | \]
can be established that is proportional to \( 1 / \sqrt{\nu_j} \). We now briefly discuss the conditions under which this property holds true. Following Barron (1993), we assume that each of the optimal control functions \( \gamma_{RH}^0 \) to be approximated has a bound to the average of the norm of the frequency vector weighted by its Fourier transform (see (10) below). However, the functions \( \gamma_{RH}^0 \) have been defined on the compact set \( X \), not on the space \( \mathbb{R}^n \). Then, to compute the Fourier transforms, we need to extend the functions \( \gamma_{RH}^0(x_t) \) from \( X \) to \( \mathbb{R}^n \). To this end, we define the functions \( \tilde{\gamma}_{RH}^0 : \mathbb{R}^n \to \mathbb{R} \) that coincide with \( \gamma_{RH}^0(x_t) \) on \( X \). We also define the class of functions
\[
G_\text{c}_j = \left\{ \tilde{\gamma}_{RH}^0, \text{ such that } \int_{\mathbb{R}^n} |\omega|^1 |\gamma_j(\omega)| d\omega \leq c_j \right\}
\]
\[(10)\]
where 
\[
|\omega|^1 = \max_{\omega \in \mathbb{K}} |\omega| \]
\( \gamma_j(\omega) \) is the Fourier transform of \( \tilde{\gamma}_{RH}^0 \), and \( c_j \) is any finite positive scalar. We can now state the following.

**Theorem 4:** Assume that, in the solution of Problem 2, the first control function \( \gamma_{RH}^0(x_t) = \gamma_{FH}^0(x_t,0) \) (see (7)) is unique, and that \( \tilde{\gamma}_{RH}^0 \in G_\text{c}_j \) for some finite positive scalar \( \tilde{c}_j \), for each \( j \) with \( 1 \leq j \leq m \). Then, for each \( j \) with \( 1 \leq j \leq m \), for every probability measure \( \sigma \), and for each \( \nu_j \geq 1 \), there exist a weight vector \( w_j \) (i.e. a neural control function \( \hat{\gamma}_{RH}^{(uj)}(x_t,w_j) \)) and a positive scalar \( c_j^* \) such that
\[
\int_X \left| \gamma_{RH}^0(x_t) - \hat{\gamma}_{RH}^{(uj)}(x_t,w_j) \right|^2 \sigma[I(x_t)] \leq c_j^* \nu_j
\]
where \( c_j^* = (2\tilde{c}_j)^2 \). Moreover, there exists a positive scalar \( k_j \) such that
\[
\max_{x_t, \sigma} \left| \gamma_{RH}^0(x_t) - \hat{\gamma}_{RH}^{(uj)}(x_t,w_j) \right|^2 \leq k_j \nu_j c_j^*
\]

Theorem 4 is derived from two theorems by Barron (1992, 1993). It states that, for any control function \( \gamma_{RH}^0(x_t) \), the number of parameters required to achieve an \( \mathcal{L}_2 \) or an \( \mathcal{L}_\infty \) approximation error of order \( O(1/\nu_j) \) is \( O(\nu_j n) \), which grows linearly with the dimension \( n \) of the state vector. For a further discussion of this property of feedforward neural approximators, in comparison with traditional linear approximators and other classes of nonlinear ones, see Girosi et al. (1995) and Parisini and Zoppoli (1995).

5. **Determination of the penalty function and of the neural network implementing the RH regulator**

In this section by using some constructive aspects of the proofs of Theorems 1 and 2, we shall describe some computational procedures to derive the penalty function and the number of neural units of the network implementing the RH regulator.

5.1. **Determination of the matrix \( P \)**

The closed-loop system \( x_{t+1} = f(x_t, K x_t) \) can be written in the form
where $\hat{A} \triangleq A + BK_x \phi$ is a suitable function such that $\phi(0) = 0$ and
\[
\lim_{\|x\| \to 0} \frac{\|\phi(x)\|}{\|x\|} = 0
\]
and $K^{-1}[U]$ represents the inverse image of the set $U$ via the linear operator $K$. It is shown below that $V(x) \triangleq x^T P x$ is a Lyapunov function for the system $x_{t+1} = f(x_t, Kx_t)$ in a suitable set $W(K, P)$. Then, once a matrix $K$ that asymptotically stabilizes the linearized system $x_{t+1} = \hat{A}x_t$ has been determined, $P$ can be computed as the solution of the Lyapunov equation
\[
P - \hat{A}^T P \hat{A} = Q
\]
for an arbitrary symmetric positive-definite matrix $Q$.

5.2. Determination of the scalar $\bar{\alpha}$

First, we need to determine the compact set $W(K, P)$. A little algebra yields
\[
\Delta V(x_t) \triangleq V(x_{t+1}) - V(x_t)
\]
\[
\leq - \left[ \lambda_{\min}(Q) - 2\|A^T P\| \|\phi(x_t)\| \right] - \lambda_{\max}(P) \left( \|\phi(x_t)\| \|x_t\| \right)^2 \|x_t\|^2
\]
(11)
where $\lambda_{\min}(Q)$ and $\lambda_{\max}(P)$ denote the minimum and maximum eigenvalues of the matrices $Q$ and $P$, respectively. The quantity within brackets is positive if
\[
\|\phi(x_t)\| \|x_t\| < \Lambda \triangleq \{ A \|A^T P\| + \|A^T P\|^2 + \lambda_{\min}(Q)\lambda_{\max}(P) \}^{1/2} / \lambda_{\max}(P)
\]
(12)
Clearly, the scalar $\Lambda$ defines implicitly the boundary $\partial A$ of the open set $A \triangleq \{ x_t \in X : \Delta V(x_t) < 0 \} \cup \{ 0 \}$. As $\Delta V(x_t) = 0$ at the boundary $\partial A$, we must determine the closed invariant set $W(K, P)$ such as not to include $\partial A$. Furthermore, to avoid numerical problems related to the nearness of $\partial A$ (these numerical problems will be pointed out later on in the determination of the scalar $\bar{\alpha}$), it is convenient to consider a proper subset of $A$ characterized by a suitable scalar $\bar{\alpha} < \Lambda$. Thus we impose the condition
\[
\|\phi(x_t)\| \|x_t\| \leq \bar{\alpha} \Lambda < \Lambda
\]
(13)
Now consider the inequalities
\[
\|\phi(x_t)\| \leq \sum_{i=1}^{n} \|\phi_i(x_t)\| \leq \bar{F}\|x_t\|^2 + G\|x_t\|^3
\]
(14)
where $\phi_i$ denotes the $i$th component of the vector $\phi$ and
\[
\bar{F} \triangleq \frac{1}{2} \sum_{i=1}^{n} \left[ \sum_{h,k \in \{1, \ldots, n\} \setminus \{i\}} \left| \frac{\partial^2 \bar{f}(x_t)}{\partial x_{th} \partial x_{ik}} \right|_{x_t=0} \right]
\]
\[
G \triangleq \frac{1}{6} \sum_{i=1}^{n} \max_{x_t \in \Gamma \setminus K^{-1}[U]} \left\{ \sum_{j,h,k \in \{1, \ldots, n\} \setminus \{i\}} \left| \frac{\partial^3 \bar{f}(x_t)}{\partial x_{ij} \partial x_{ih} \partial x_{ik}} \right| \right\}
\]
(x_k is the kth component of the vector x, and \( \tilde{f}_i \) is the ith component of the vector function \( \tilde{f}(x_i, Kx_i) \)). Now, so that (13) may be satisfied, it is sufficient that

\[
\|F\|_x + G\|x_i\|^2 \leq \lambda
\]

(15)

By using (13) and (14), we obtain immediately

\[
\Delta V(x_i) < 0, \quad \forall x_i \in X \cap K^{-1}[U] x_i \neq 0, x_i : \|x_i\| \leq \sqrt{\beta} \triangleq - \frac{F + (F^2 + 4G\lambda)^{1/2}}{2G}
\]

(16)

Note that, as

\[
\lim_{\|x\| \to 0} \frac{\|\phi(x)\|}{\|x\|} = 0
\]

there exists a suitably small positive scalar \( \beta \) such that the compact set

\[
W(K, P) \in \mathbb{R}^n, \quad W(K, P) = \{x \in \mathbb{R}^n : x^T P x \leq \beta\}
\]

is contained in \( \{x \in \mathbb{R}^n : \|x_i\| \leq \sqrt{\beta}\} \). Then

\[
\Delta V(x_i) < 0, \quad \forall x_i \in W(K, P) \setminus \{0\}
\]

More specifically, from (16) it follows that

\[
\beta = \lambda_{\text{min}}(P) \beta \Rightarrow W(K, P) = \{x_i \in \mathbb{R}^n : x^T P x_i \leq \lambda_{\text{min}}(P) \beta\}
\]

(17)

Hence \( W(K, P) \) is an invariant set and a domain of attraction for 0 as an equilibrium point of the linearized system \( x_{t+1} = A x_t \).

Once the compact set \( W(K, P) \) has been determined, we can prove Fact 2 (as mentioned earlier, the proof is different from the one given by Parisini and Zoppoli 1995 and will allow us to compute the scalar \( a \)). Obviously

\[
\frac{\partial h}{\partial x} \bigg|_{x=0,\mu=0} = 0 \quad \text{and} \quad \frac{\partial h}{\partial u} \bigg|_{x=0,\mu=0} = 0
\]

(see Assumption 2). We now determine a matrix \( R \in \mathbb{R}^{n \times n} \) such that

\[
h(x_i, Kx_i) \leq x_i^T R x_i, \quad \forall x_i \in W(K, P)
\]

(18)

By using Assumption 2 and (17), we have

\[
h(x_i, Kx_i) \leq s(||(x_i, Kx_i)||^2) \leq k_s(K, P)(1 + ||K||)^2\|x_i\|^2, \quad \forall x_i \in W(K, P)
\]

where

\[
K_s(K, P) \triangleq \max_{z \in W(K, P)} \left| \frac{ds(z)}{dz} \right|
\]

Then, so that (18) may be satisfied, it is sufficient to choose any positive-definite symmetric matrix \( R \) such that

\[
\lambda_{\text{min}}(R) \geq k_s(K, P)(1 + ||K||)^2
\]

For a generic terminal cost function \( h_F(x) = ax^T P x \), from (11) we obtain

\[
h_F(x_{t+1}) - h_F(x_t) = a \Delta V(x_i)
\]

\[
\leq - \left[ \lambda_{\text{min}}(Q) - 2A^T P \|\phi(x_i)\|_x - \lambda_{\text{max}}(P) \left( \frac{\|\phi(x_i)\|}{\|x_i\|} \right)^2 \right] \|x_i\|^2
\]
We now choose a scalar $\hat{a}$ such that the following inequality is satisfied:

$$\hat{a} \left[ \lambda_{\min}(Q) - 2\| A^T P \| \left\| \phi(x_t) \right\|_{x_t} - \lambda_{\max}(P) \left( \left\| \phi(x_t) \right\|_{x_t} \right)^2 \right] \geq \lambda_{\max}(R)$$  (19)

Then, for any $a \geq \hat{a}$ we have

$$h_F(x_{t+1}) - h_F(x_t) \leq - \lambda_{\max}(R) \left\| x_t \right\|^2 \leq - x_t^T R x_t \leq - h(x_t, K x_t), \quad \forall x_t \in W(K, P)$$

thus proving Fact 2. The scalar $\hat{a}$ for which (19) is satisfied can be easily computed by using (13). Then, from (19) it follows that we can choose the scalar $\hat{a}$ defined as

$$\hat{a} = \frac{\lambda_{\max}(R)}{\lambda_{\min}(Q) - 2\| A^T P \| \lambda_{\max}(P)}$$  (20)

It is worth noting that, as $\tilde{\lambda}$ approaches $\lambda$, $\hat{a}$ increases. This is consistent with the fact that we approach the region where $\Delta V(x_t)$ takes on small values (recall that we have $\Delta V(x_t) = 0$ at $\tilde{\lambda}$). Then, an increase in the attractiveness of the final cost $a \| x_{t+N} \|^2$ is needed.

Finally, once the scalar $a^*$ has been derived from the maximization (6), $\tilde{a}$ can be derived from its definition, i.e. $\tilde{a} = \max(\hat{a}, a^*)$ (see the proof of the lemma in section 3).

5.3. Determination of the number of neural units

We assume the RH neural regulator to be implemented by the parallel of $m$ single-output networks, each containing a single hidden layer with $v_j$ neural units. We also assume the output layer to be composed of linear activation units (see the discussion before Theorem 3 and Problem 3; actually, what is said in the following for the determination of $\epsilon$ holds true for any kind of nonlinear approximator). Then, in virtue of Theorem 3, there exist sufficiently large numbers $v_1, \ldots, v_m$ such that the RH neural regulator makes the state vector enter the set $W(K, P)$ after a finite number of stages, for any initial state $x_i \in X$. This property holds true, provided that the errors $\| u_{t}^{RH} - \hat{u}_{t}^{RH} \|$, $i \geq t$, never exceed a proper scalar $\tilde{\epsilon}$. We first determine the value of such a scalar; this can be done by proving part (b) of Theorem 2, where $w_d = W(K, P)$.

Let us consider an arbitrary initial state $x_i \in X_0$ and the next two states $x_{t+1}^{RH} = f(x_t, u_t^{RH})$ and $\hat{x}_{t+1}^{RH} = f(x_t, \hat{u}_t^{RH})$. Consider also the function $\tilde{V}(x_t) \triangleq J_{x_{t+1}}^{\theta}(x_t, N, a, P)$. As mentioned in section 3 after Theorem 1, $\tilde{V}$ is a Lyapunov function in $X_0$ for the system (1) driven by the optimal RH regulator. Now, to develop a simple numerical technique to determine the scalar $\tilde{\epsilon}$, we make Assumption 4 a bit stronger, that is we assume that the optimal FH feedback control functions $\gamma_{t}^{\theta}(x_t, i), i = t, \ldots, t + N - 1$, are $C^1$ functions with respect to $x_t$, for any $x_t \in X$ and any finite integer $N \geq 1$. Hence, this new assumption, together with the regularity hypotheses on the system (1) and the cost (2), ensures that $\tilde{V}$ is a $C^1$ function. Then

$$\Delta \tilde{V}(x_t) \triangleq \tilde{V}(x_t^{RH}) - \tilde{V}(x_t) < 0, \forall x_t \in X_0$$

It is easy to see that, if the control error is bounded as

$$\| u_{t}^{RH} - \hat{u}_{t}^{RH} \| < \tilde{\epsilon}(x_t) \triangleq - \frac{\Delta \tilde{V}(x_t)}{K_{x_t}^{f}(x_t) K_{x_t}^{f}(x_t^{RH})}$$  (21)
where $K^f(x_t)$ and $K^V(x_{t+1}^{RH})$ are local Lipschitz constants of the functions $f$ and $\tilde{V}$, respectively, then
\[
\left| \tilde{V}(x_{t+1}^{RH}) - \tilde{V}(\hat{x}_{t+1}) \right| < -\Delta \tilde{V}(x_t), \quad \forall x_t \in X_0
\]
(21) yields
\[
\tilde{V}(\hat{x}_{t+1}) - \tilde{V}(x_t) < 0, \quad \forall x_t \in X_0
\]
Then, $\tilde{V}$ is a Lyapunov function also for the suboptimal RH trajectories, thus ending the proof of Part (b) of Theorem 2.

Let us now specify how to compute the Lipschitz constants. We can write
\[
\left\| f(x_t, u) - f(x_t, u_{t}^{RH}) \right\| \leq K^f(x_t) \left\| u - u_{t}^{RH} \right\|, \quad \forall u \in \mathcal{A}^f(x_t)
\]
where
\[
K^f(x_t) \triangleq \max_{u \in V(x_t)} \left\| \frac{\partial f(x_t, u)}{\partial u} \right\|
\]
and
\[
\mathcal{A}^f(x_t) \triangleq \left\{ u \in U : \left\| u - u_{t}^{RH} \right\| \leq \delta(x_t) \right\}
\]
Analogously
\[
\left\| \tilde{V}(x) - \tilde{V}(x_{t+1}^{RH}) \right\| \leq K^V(x_{t+1}^{RH}) \left\| x - x_{t+1}^{RH} \right\|, \quad \forall x \in \mathcal{A}^V(x_t)
\]
where
\[
K^V(x_{t+1}^{RH}) \triangleq \max_{x \in V(x_t)} \left\| \nabla \tilde{V}(x) \right\|
\]
and
\[
\mathcal{A}^V(x_t) \triangleq \left\{ x \in X : \left\| x - x_{t+1}^{RH} \right\| \leq K^f(x_t) \delta(x_t) \right\}
\]
As $\mathcal{A}^V(x_t)$ and $\mathcal{A}^f(x_t)$ depend on the scalar $\delta(x_t)$, which in turn depends on $\mathcal{A}^V(x_t)$ and $\mathcal{A}^f(x_t)$ through the constants $K^V(x_{t+1}^{RH})$ and $K^f(x_t)$ (see the maximizations (22) and (23)); some iterative procedure is needed to determine them. The following has proved to be good heuristics in the simulations performed, though its convergence is not ensured:

(a) initialize this procedure by setting
\[
K^f(x_t) = \left\| \frac{\partial f(x_t, u)}{\partial u} \right\|_{x_t, u_{t}^{RH}}
\]
\[
K^V(x_{t+1}^{RH}) = -\frac{\Delta \tilde{V}(x_t)}{\left\| x_{t+1}^{RH} - x_t \right\|}
\]
(b) determine $\tilde{\delta}(x_t)$
(c) determine $\mathcal{A}^V(x_t)$ and $\mathcal{A}^f(x_t)$

Then, one can compute $K^V(x_{t+1}^{RH})$ and $K^f(x_t)$ again by the maximizations (22) and (23), go to step (b), and repeat the procedure until convergence (if any) is reached, $\forall x_t \in X$. Then
\[
\epsilon = \min_{x_t \in X} \tilde{\delta}(x_t)
\]
Once $\epsilon$ has been determined, the trial-and-error procedure outlined after the statement of Problem 3 can be implemented to derive the numbers $\nu_1^*, \ldots, \nu_m^*$ of neural units.
Example 1: Consider the following nonlinear system
\[
\begin{aligned}
    x_{1,t+1} &= x_{1,t} + T \left[ x_{2,t} + \frac{1}{2} (1 + x_{1,t}) u_t \right] \\
    x_{2,t+1} &= x_{2,t} + T \left[ x_{1,t} + \frac{1}{2} (1 - 4 x_{2,t}) u_t \right]
\end{aligned}
\]
which is the discretized version of the undamped oscillator addressed by Mayne and Michalska (1990), where \( T = 0.05 \) is the sampling period. No state and control constraints are imposed, i.e. \( X = \mathbb{R}^2 \) and \( U = \mathbb{R} \). We derive the design parameters for the approximate RH neural control law by following the computational steps described in section 5. (We recall that Mayne and Michalska (1990) determined the RH control law by on-line computations.) The FH cost function is given by
\[
J_{FH} = \sum_{t=0}^{t+N-1} (u_t^2 + 0.1 \| x_i \|^2) + a \| x_N \|_p^2
\]
where \( N = 100 \).

6.1. Determination of the matrix \( P \)

We first need to determine a linear control law \( u_t = k^T x_t \) that asymptotically stabilizes the linearized system \( x_{t+1} = A x_t + b u_t \), where
\[
A = \begin{bmatrix} 1 & -0.05 \\ 0.05 & 1 \end{bmatrix}, \quad b = \begin{bmatrix} 0.025 \\ 0.025 \end{bmatrix}
\]
As a linear control law, we choose the gain \( k^T = \begin{bmatrix} 0.3799 \\ -0.4201 \end{bmatrix} \) for which the closed-loop poles are \( 0.99 \pm 0.0495i \). We now determine the matrix \( P \) by solving the Lyapunov equation \( P - \tilde{A}^T P \tilde{A} = Q \), where \( \tilde{A} = A + b k^T \). More specifically, we choose
\[
Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \text{which yields } P = \begin{bmatrix} 48.3043 & -2.53 \\ -2.53 & 70.8191 \end{bmatrix}
\]

6.2. Determination of the scalar \( \alpha \)

From (12) we obtain \( \Lambda = 7.059 \times 10^{-3} \). A reasonable choice of \( \tilde{\Lambda} \) turns out to be \( \Lambda = \Lambda - 4 \times 10^{-4} = 6.659 \times 10^{-3} \) (see the discussion before (13)). As \( F = 0.1 \) and \( G = 0, \) we have \( \beta = (\tilde{\Lambda} / F)^{1/2} \) (see (15)), hence \( \beta = \lambda_{\min}(P) \beta = 0.2129 \). As discussed in section 5, the choice of the matrix \( Q \) is arbitrary. However, for numerical reasons, it is convenient to make \( W(K, P) \) as wide as possible; then, the above gain \( k \) and \( Q \) have been chosen by a trial-and-error procedure to make \( \beta \) sufficiently large. As the transition cost function \( h(x, u) \) is quadratic, the matrix \( R \) can be obtained trivially as
\[
R = 0.1 I + k k^T = \begin{bmatrix} 0.2443 & 0.1596 \\ 0.1596 & 0.2765 \end{bmatrix}
\]
where \( I \) is the \( 2 \times 2 \) identity matrix. Then (20) yields \( \hat{\alpha} = 7.4014 \). We now must determine \( \alpha^* \) through the maximization (6). As it turned out \( \alpha^* < \hat{\alpha} \), we obtained \( \tilde{\alpha} = \hat{\alpha} = 7.4014 \).

6.3. Determination of the number of neural units and of the RH regulator

The neural RH regulator will be determined in the region
In section 4 we have described a trial-and-error procedure to determine the minimum number of neural units to achieve a control error uniformly bounded by \( e \). This guarantees that all approximate trajectories enter \( W(K, P) \). In the following, we drop the index \( j \) in \( ^\gamma_RH (x_t, w_j) \), as the control variable is scalar. More specifically, we recall that \( \nu \) has to be increased until the following inequality is satisfied:

\[
\min \max_{w} \max_{x_t \in A} | ^\gamma_RH (x_t) - ^\gamma_RH (x_t, w)| \leq e \tag{24}
\]

By using the iterative procedure outlined in section 5, we obtained \( e = 6 \times 10^{-4} \). However, a very large value of \( \nu \) is needed so that (24) may be satisfied. On the other hand, \( \tilde{\alpha}(x_t) \) is of the same order of magnitude as \( e \) only in the vicinity of the contour of \( W(K, P) \) and increases when we go away from \( W(K, P) \). This means that, in the present example, (24) is too conservative with respect to (21) (which, as stated in section 5, is sufficient so that any approximate trajectory may enter \( W(K, P) \)). Then we considered the following trial-and-error procedure: increase \( \nu \) until

\[
\min \max_{w} \max_{x_t \in A} | ^\gamma_RH (x_t) - ^\gamma_RH (x_t, w)|/\tilde{\alpha}(x_t) \leq 1 \tag{25}
\]

is satisfied. Condition (25) is a slight modification to (24), and is simple to interpret: the errors \(| ^\gamma_RH (x_t) - ^\gamma_RH (x_t, w)| \) are weighted by \( 1/\tilde{\alpha}(x_t) \); that is, the errors are more penalized at the points of \( A \) where \( \tilde{\alpha}(x_t) \) is small, and vice versa.

We found that, for \( \nu = 20 \)

\[
\max_{x_t \in A} | ^\gamma_RH (x_t) - ^\gamma_RH (x_t, w^*)|/\tilde{\alpha}(x_t) \leq 0.0481
\]

where \( w^* \) is the weight vector obtained via the minimax procedure defined in (25) for \( \nu = 20 \). Then (21) is satisfied with a considerable margin. In figures 1(a) and 1(b) the control functions \( ^\gamma_RH (x_t) \) and \( ^\gamma_RH (x_t, w^*) \) are plotted, respectively; the comparison between the shapes of these functions confirms the powerful approximation characteristics of the neural function \( ^\gamma_RH \). Figure 2(a) shows some state trajectories (starting from different initial states), under the actions of the optimal (dashed line) and approximate (solid line) regulators. These trajectories practically coincide. To appreciate the very small differences, in figure 2(b) a part of figure 2(a) is depicted in enlarged form, also showing the set \( W(K, P) \).

Example 2: In the present example, we address a problem of freeway traffic optimal control. The choice of this example is motivated by its intrinsic engineering importance and by the fact that it deals with a high-order strongly nonlinear dynamic system. We shall follow an RH neural approach that is rather different from the one described in this paper. More specifically, we do not enforce the final cost \( \alpha \| x_{t+1} \|_2^2 \), and we design the RH neural control law by minimizing the cost (8) instead of solving Problem 3. Therefore, stabilization is not ensured. In compensation, we have the possibility of comparing an RH neural approach with a traditional computational technique presented in the extensive literature on freeway traffic control.

We refer to the following model, originally proposed by Payne (1971) and then used in many works on freeway control (a comprehensive description of the model can be found in the work of Papageorgiou 1983):
The freeway is assumed to be divided into $K$ sections of length $\Delta_k$ (see figure 3). $T$ is the sample time interval. $v_{kt}$ is the mean traffic speed on section $k$ at time $t$. $\rho_{kt}$ is the
traffic density, and $l_{kt}$ is the number of queueing vehicles on the on-ramp of section $k \in I_r$ (the indexes of the sections with on/off ramps make up the set $I_r$). $r_{kt}$ is the on-ramp traffic volume that is imposed by monitoring the vehicle access via traffic lights, and $b_{kt}$ denotes speed limits set by means of variable message signs ($b_{kt} = 1$ means that there is no speed limit on section $k$). Then we can define the state and control vectors as follows:

$$x_t \triangleq \col(v_{kt}, k = 1, \ldots, K; p_{kt}, k = 1, \ldots, K; l_{kt}, k \in I_r)$$

$$u_t \triangleq \col(r_{kt}, k \in I_r; b_{kt}, k = 1, \ldots, K)$$

$d_{kt}$ is the demand flow for access to the on-ramp of section $k \in I_r$. 

Figure 2. (a) State trajectories under the action of the optimal (dashed line) and the approximate (solid line) control laws; (b) the same state trajectories in a neighbourhood of the origin and the set $W(K, P)$ (ellipsoidal grey region).
Clearly, both the state and control variables must belong to suitable intervals: 
\[ 0 \leq v_{kt} \leq v_{kmax}, \quad 0 \leq \rho_{kt} \leq \rho_{kmax}, \quad \text{and} \quad 0 \leq l_{kt} \leq l_{kmax}. \]
As to the control variables, we assume 
\[ 0.7 \leq b_{kt} \leq 1 \]
and the more complex constraints
\[ \max \left[ r_{kmin}, d_{kt} - \frac{1}{\tau} (l_{kmax} - l_{kt}) \right] \leq r_{kt} \leq \min \left( r_{kmax}, d_{kt} + \frac{1}{\tau} l_{kt} \right) \]
As in Problem P3 of Papageorgiou (1983), the process cost is given by the total travel time spent on the freeway plus the total waiting time spent in the queues at the freeway entrances; that is
\[ J_{FH} = \tau \sum_{t=1}^{t=N} \sum_{k=1}^{K} \rho_{k}, \Delta_k + \sum_{k=1}^{K} l_{kt} \]  \hspace{1cm} (26)
The freeway stretch considered consists of \( K = 30 \) sections (each of 1 km length), \( N = 5 \) and \( \tau = 15 \) s. On-ramps and off-ramps are present for sections 3, 9, 15, 21 and 27. There are five control variables for speed limits and each of them acts on six successive sections. In particular, variable message signs are placed on sections 1, 7, 13, 19 and 25. It follows that \( \dim (\mathbf{x}_t) = 65 \) and \( \dim (\mathbf{u}_t) = 10 \). The model describing the dynamic behaviour of the freeway traffic is characterized by the following parameters: \( \alpha = 0.9, \quad V_f = 123 \text{ km/h}, \quad l = 4, m = 1.4, \quad \tau = 0.01 \text{ h}, \quad \nu = 21.6 \text{ km}^2/\text{h}, \quad \chi = 20 \text{ veh/km}, \quad \delta_{on} = 0.1, \quad \rho_{kmax} = 200 \text{ veh/km} \) \quad and \quad \( v_{kmax} = 200 \text{ km/h} \) for \( k = 1, \ldots, K, \quad l_{kmax} = 200 \text{ veh}, \quad r_{kmin} = 0 \text{ veh/h}, \quad r_{kmax} = 20000 \text{ veh/h}, \) \quad for \quad \) each on-ramp, \quad and \quad \( \gamma_k = 0.1 \) for each off-ramp (the values of these scalars were taken from table 3.1 of Papageorgiou 1983). The demands \( d_{kt} \) are assumed to be known and time-invariant. We set \( d_{3t} = 6000 \text{ veh/h}, d_{9t} = 1000 \text{ veh/h}, k = 9, 15, 21 \) and 27.

To evaluate the effectiveness of our neural approach, we compare it with the control scheme proposed by Messner and Papageorgiou (1992) to solve the above-described traffic control problem. Such a scheme essentially consists in deriving online the optimal control law (OLO control law) that solves Problem 1 (the period of the on-line optimization can be longer than one stage). Instead, following the general approach proposed in the paper, we derive off-line the RH optimal neural (RHON) control law \( \mathbf{u}_t^{\text{RH}} = \mathbf{\gamma}^{\text{RH}}(\mathbf{x}_t, \mathbf{w}^0) \) that minimizes the approximation error (8). To this end, we focus our attention on an algorithm of the gradient type, as when applied to neural networks it turns out to be simple and well-suited to distributed computation. We define the function
\[ D(\mathbf{w}, \mathbf{x}_t) = \| \mathbf{\gamma}^0(\mathbf{x}_t, 0) - \mathbf{\gamma}^0(\mathbf{x}_t, \mathbf{w}) \|^2 = \| \mathbf{\gamma}^{\text{RH}}(\mathbf{x}_t) - \mathbf{\gamma}^{\text{RH}}(\mathbf{x}_t, \mathbf{w}) \|^2 \]
and note that we are able to evaluate $g_{FH}(x_t, w)$ only pointwise; that is, by solving Problem 2 for specific values of $x_t \in X$. It follows that we are unable to compute the function $D(w, x_t)$, which leads us to compute the realization $\nabla w D(w_t, x_t)$, instead of the gradient $\nabla w E(w)$. We generate the sequence $\{x_t(k), k = 0, 1, \ldots\}$ randomly and use the following updating algorithm:

$$w(k + 1) = w(k) - \alpha(k) \nabla w D[w(k), x_t(k)] \quad k = 0, 1, \ldots \quad (27)$$

The probabilistic algorithm (27) is based on the concept of stochastic approximation. Sufficient conditions for the algorithm convergence can be found, for instance, in the work of Polyak and Tsypkin (1973). As to $\alpha(k)$, we take the step-size $\alpha(k) = c_1/(c_2 + k)$, $c_1, c_2 > 0$, which satisfies the aforesaid conditions for the algorithm convergence. We also add a momentum $\rho[w(k) - w(k - 1)]$ to (27), as is usually done in training neural networks ($\rho$ is a suitable positive constant). We set $c_1 = 1, c_2 = 10^5$ and $\rho = 0.9$. To derive the components of $\nabla w D[w(k), x_t(k)]$ the well-known backpropagation updating rule can be applied.

As the descent algorithm is able to solve only an unconstrained nonlinear programming problem (whereas deriving the OLO control law entails the solution of a constrained one), we remove the constraints on the control and state variables and add suitable penalty functions to the cost (26). Then, the costs $h(x_t, u_t)$ are defined like the terms in (26) plus the penalty functions. This is done for both the OLO control law proposed by Messner and Papageorgiou (1992) and our neural method, thus making the two approaches comparable. The RH optimal neural control law $\gamma_{RH}(x_t, w^0)$ is implemented by a neural network containing one hidden layer of 100 units. The gradient algorithm to optimize the neural control law converged after about $5 \times 10^5$ iterations. This large number of iterations may be decreased by using some of the acceleration techniques proposed in the literature. However, we recall that this is not a very critical point, as all the computations are performed off-line.

The traffic situation considered shows a severe congestion at time $t = 0$ on sections 10, 11 and 12 and consequently, low values of traffic density on the successive ones. More specifically, figure 4 shows the traffic density $\rho_{kt}$ and the mean traffic speed $v_{kt}$ on the freeway sections when the RHON control law is applied. When the freeway system is driven by the OLO regulator, the surfaces of $\rho_{kt}$ and $v_{kt}$ (as functions of the stage $t$ and of the section $k$) are nearly the same, so we do not specify them. To appreciate the very small difference between the state behaviours related to the two control Shemes, in figure 5 we show the bidimensional diagrams of $\rho_{kt}$ and $v_{kt}$ as functions of the section $k$, at the stage $t = 30$ (i.e. after 7.5 min).

It can be noted that the RHON control law, even though approximate, turns out to be very effective, as it yields performances almost identical with those obtained by using the OLO control law. More specifically, define the RH cost as

$$J_{RH}^0(\tilde{x}) \triangleq \sum_{t=0}^{N_T-1} h(x_t^{RH0}, u_t^{RH0})$$

where the state $\tilde{x}$ describes the traffic configuration at time $t = 0, N_T$ is the number of stages in which the RH control law was activated ($N_T = 60$, corresponding to 15 min
in the present case), and $u_t^{RH}$ is the control action generated by either the OLO or the RHON control law. Then the use of the RHON control law implies an increase of only 0.25% in the cost (28). Furthermore, it is worth noting that the optimal neural control law drives the system state into a neighbourhood of the traffic configuration (or state vector) characterized by a mean traffic speed that is quite close to the free speed $V_f = 123$ km/h$^{-1}$. Such a result may be interpreted as a stabilizing property of the neural regulator. This property was verified experimentally for a variety of initial conditions.

7. Conclusions

The RH neural control scheme proposed in the paper exhibits the interesting
property that it can be computed off-line. Instead, the RH regulators that have so far appeared in the literature generate the control vectors on-line by solving more or less difficult optimal control problems or nonlinear programming problems. Existence conditions on an RH stabilizing regulator and on neural networks implementing it have been established. It has been shown that the parameters needed for the design of the neural regulator can be determined partly analytically and partly by solving some suitable global optimization problems. As is well known, solving these problems to a given guaranteed accuracy is not an easy task, unless some regularity information on the function to be minimized (or maximized) is available and the dimensions of the unknown vectors (in our case, typically the dimension of the system state) are moderate.

Actually, more attention has been paid to the statement of the global optimization problems than to the development of solving algorithms that exploit the
peculiarities of the design problems. The results presented in the paper constitute only a first step toward a satisfactory off-line design of RH stabilizing neural regulators.

ACKNOWLEDGMENTS

The authors wish to thank Angela Di Febbraro for her helpful suggestions in stating the freeway optimal control problem, and Marco Baglietto and Simona Sacone for their assistance in preparing the simulation examples. This work was supported by the Italian Ministry for the University and Research.

REFERENCES


