Design of Asymptotic Estimators: An Approach Based on Neural Networks and Nonlinear Programming

Angelo Alessandri, *Senior Member, IEEE*, Cristiano Cervellera, and Marcello Sanguineti

***Abstract*—A methodology to design state estimators for a class of nonlinear continuous-time dynamic systems that is based on neural networks and nonlinear programming is proposed. The estimator has the structure of a Luenberger observer with a linear gain and a parameterized (in general, nonlinear) function, whose argument is an innovation term representing the difference between the current measurement and its prediction. The problem of the estimator design consists in finding the values of the gain and of the parameters that guarantee the asymptotic stability of the estimation error. Toward this end, if a neural network is used to take on this function, the parameters (i.e., the neural weights) are chosen, together with the gain, by constraining the derivative of a quadratic Lyapunov function for the estimation error to be negative definite on a given compact set. It is proved that it is sufficient to impose the negative definiteness of such a derivative only on a suitably dense grid of sampling points. The gain is determined by solving a Lyapunov equation. The neural weights are searched for via nonlinear programming by minimizing a cost penalizing grid-point constraints that are not satisfied. Techniques based on low-discrepancy sequences are applied to deal with a small number of sampling points, and, hence, to reduce the computational burden required to optimize the parameters. Numerical results are reported and comparisons with those obtained by the extended Kalman filter are made.**

***Index Terms*—Feedforward neural networks, Lyapunov function, offline optimization, penalty function, quasi-random sequences, state observer.**

I. Introduction

STATE ESTIMATORS play a basic role in many applications, whenever one has to make a decision on the basis of partial information, e.g., when one deals with a dynamic system and has to perform a control action having at one's disposal incomplete measurements of the state variables. For linear systems, the estimation problem was solved by the Luenberger observer [1]. After its appearance, various methods have been developed to construct state estimators for nonlinear dynamic systems. A Lyapunov function is usually sought to guarantee the convergence of the estimation error to zero. However, no general methodology is available to find such Lyapunov functions for observers of dynamic systems with nonlinearities in unmeasurable state variables.

The first convergence results on state observers for certain nonlinear dynamic systems were presented in [2] and [3]. Later on, different geometric approaches were proposed (see, e.g., [4] and [5]), and other methods were developed to estimate the state variables with asymptotically stable error dynamics [6]–[8]. In many applications, the so-called extended Kalman filter (EKF) usually yields satisfactory results, though only local convergence properties are known [9]. In recent years, the most successful state estimator for nonlinear systems has become the high-gain observer [7], [10], particularly for the purpose of output feedback control, thanks to its global convergence properties (see, e.g., [11]).

In this paper, we address the state estimation problem for a class of nonlinear continuous-time dynamic systems using an estimator designed by adjusting the parameters of a suitably parameterized innovation function via an ad-hoc developed nonlinear programming algorithm. In order to guarantee the asymptotic convergence of the estimation error via a quadratic Lyapunov function, we search for an innovation term that is made up of two contributions: 1) a linear gain and 2) a parameterized input/output mapping that can be represented by certain families of neural networks [12]. Such networks exhibit, besides the so-called "universal approximation property" [13]–[16], powerful approximation capabilities, like a "small" number of parameters required to guarantee a fixed approximation accuracy, especially in high-dimensional settings [17]–[20].

The design parameters (i.e., the linear gain and the parameters of the input/output mapping) are chosen such as to constrain the derivative of the Lyapunov function to be negative on a grid of points (obtained by a deterministic sampling rule) in the Cartesian product of the compact sets to which the state vector and the estimation error belong. The linear gain results from the solution of a Lyapunov equation. Once the gain has been found, the neural weights are searched for by minimizing a cost function that penalizes the unsatisfied constraints according to a penalty-function approach [21]. Under suitable assumptions on the choice of the sampling points and the smoothness of the Lyapunov function, the negative definiteness of the derivative is guaranteed also for points not belonging to the grid. Moreover, we show how this property is related to a scattering measure of the sequence of sampling points, called *dispersion*, which quantifies "how uniformly" the points are spread in the space. The resulting observer provides an asymptotically stable estimation error under any initial condition on a compact set.

Manuscript received May 4, 2005; revised May 31, 2006.

A. Alessandri is with the Department of Production Engineering, Thermoenergetics, and Mathematical Models (DIPTEM), University of Genoa, Genova 16129, Italy (e-mail: alessandri@diptem.unige.it).

C. Cervellera is with the Institute of Intelligent Systems for Automation, National Research Council of Italy (ISSIA-CNR), Genova 16149, Italy (e-mail: cervellera@ge.issia.cnr.it).

M. Sanguineti is with the Department of Communications, Computer and System Sciences (DIST), University of Genoa, Genova 16145, Italy (e-mail: marcello@dist.unige.it).

Digital Object Identifier 10.1109/TNN.2006.883015

As to the sampling technique, a straightforward approach based on a uniform sampling of each component of the state-error product space might be employed, but this would lead to a "structural" exponential growth of the number of discretization points and thus make the resulting algorithm inapplicable in high-dimensional contexts. Among the possible alternatives, Monte Carlo sampling—i.e., the discretization of the work space by a sequence of independently and identically distributed (i.i.d.) points from a uniform distribution—cannot be employed, due to the probabilistic nature of the bounds associated with it (i.e., they are subject to a confidence interval), whereas the stability of the estimation error requires deterministic bounds. In addition, it is known [22] that pure Monte Carlo sampling is subject to the formation of "clusters" of points (especially in high-dimensional settings) that undermine the uniformity of the discretization when the algorithm is implemented.

Therefore, to ensure the negative definiteness of the derivative of the Lyapunov function, we use a sampling technique that allows a deterministic coverage of the space. Toward this end, we exploit a family of deterministic sequences, commonly called *quasi-random sequences* (or quasi-Monte Carlo sequences), which aim at covering the space while minimizing a measure, called *discrepancy*, strictly related to the aforementioned dispersion index. Methods based on such sequences, generally employed in the fields of statistics and numerical analysis [23], exhibit two advantages over Monte Carlo methods: 1) they are characterized by better bounds on the dispersion, and 2) their bounds are deterministic and, hence, are not subject to a confidence interval [22]. Furthermore, it is often reported by practitioners that for such sequences the formation of clusters is less problematic than for Monte Carlo methods [22]. The use of quasi-random sequences for function learning by neural networks was first investigated in [24]. In this paper, we employ such sequences to sample the state and error spaces efficiently, and show how their use actually leads to the convergence of the proposed estimation algorithm.

As regards the selection of the design parameters, it is obtained by means of a nonlinear-programming algorithm running *offline* [25], [26]. This is a major advantage over other neural approaches to estimation proposed in the literature (see, e.g., [27]–[31]), which rely on the online adaptation of the neural weights and so may involve a large number of computations in real-time applications.

Finally, as to the comparisons with the previously mentioned classical methods for state estimation, we consider the EKF and the high-gain observer. The main drawback of the former is the little knowledge of its convergence properties, which are basically local [9]. On the other hand, the high-gain observer is characterized by an estimation error that is globally asymptotically stable but relies on a suitable state-space change of coordinates; moreover, it requires a global Lipschitz assumption about the function describing the nonlinear contribution to the dynamics. By contrast, our approach does not need an explicit global Lipschitz hypothesis. We only assume that the trajectory of the system belongs to a compact set. Under suitable conditions, we guarantee the convergence of the estimation error for any value of the initial error on such a compact set.

The paper is organized as follows. Section II is devoted to the description of the basic assumptions about the system equations and the structure of the estimator. The stability of the estimation error is considered in Section III. Section IV describes a class of parameterized nonlinear mappings that include feedforward neural networks. Such networks can be fruitfully used for the design of the estimator, thanks to their approximation properties. In Section V, a design procedure is outlined. Simulation results for observers designed by using certain one-hidden-layer feedforward neural networks are presented in Section VI, and the performances of our estimator are compared with those of the EKF. Brief remarks are made in Section VII, where some conclusions are also drawn.

II. System Description and Structure of the Observer

We consider dynamic and measurement equations of the form

$$\begin{cases} \dot{x} = Ax + f(x) \\ y = Cx \end{cases}, \qquad t \geq 0 \tag{1}$$

where $x(t) \in X \subset \mathbb{R}^n$ is the state vector, X is a connected, compact set, and $y(t) \in \mathbb{R}^m$ is the measurement vector. The matrices $A \in \mathbb{R}^{n \times n}$ and $C \in \mathbb{R}^{m \times n}$ as well as the function $f : X \to \mathbb{R}^n$ are supposed to be known.

We make the following assumption.

Assumption 1: The function $f : X \to \mathbb{R}^n$ is locally Lipschitz in X. Moreover, the solution of the differential equation in (1) is well defined $\forall t \geq 0$. □

The first part of Assumption 1 guarantees the existence and uniqueness of a local solution to the differential equation describing the dynamics in (1) (see, e.g., [11]). The second part of Assumption 1 ensures the existence of a regime for the dynamics of the estimation error and guarantees the possibility of estimating its asymptotic behavior.

For reasons that will be clarified in the proof of Theorem 1, we also make the following assumption.

Assumption 2: The pair (A, C) is observable. □

Assumption 2 expresses the fact that we focus on observable representations (1) or equivalently on systems that are diffeomorphic to (1). In [32], necessary and sufficient conditions are given to ensure the existence of a diffeomorphism that transforms quite a general nonlinear system into the form (1).

The proposed Luenberger-like observer for system (1) has the following form:

$$\dot{\hat{x}} = A\hat{x} + f(\hat{x}) + L(y - C\hat{x}) + \gamma(y - C\hat{x}, w), \qquad t \geq 0 \tag{2}$$

where $\hat{x}(t) \in \mathbb{R}^n$ is the estimate of the vector $x(t)$ at time t, $L \in \mathbb{R}^{n \times m}$ is a gain matrix, and $\gamma : \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}^n$ is a parameterized function that depends on a p-dimensional vector of parameters, which we denote by $w \in \mathbb{R}^p$. In Section IV, we shall choose functions γ (e.g., the input/output mappings of certain neural networks) that exhibit properties to be exploited in the design of the estimator. Let $Z \subset \mathbb{R}^m$ be the compact set to which $z \triangleq y - C\hat{x}$ belongs. As to the admissible functions γ, we assume the following.

Assumption 3: For every $w \in \mathbb{R}^p$, the function $\gamma(\cdot, w) : Z \to \mathbb{R}^n$ is locally Lipschitz in Z and such that $\gamma(0, w) = 0$. Moreover, the solution of the differential equation (2) is well defined $\forall t \geq 0$. □

The Lipschitz condition in Assumption 3 is a sufficient requirement for having a unique local solution to the differential (2) describing the observer. The condition $\gamma(0, w) = 0$, $\forall w \in \mathbb{R}^p$, guarantees that, if there exists $T \geq 0$ such that $\hat{x}(T) = x(T)$, then $\hat{x}(t) = x(t)$ for every $t \geq T$. Moreover, Assumption 3 ensures that the origin is an equilibrium point for the dynamic equation of the estimation error. It is worth noting that the function γ can be of a more general type [33], e.g., a function $\gamma : Z \times Z \times \mathbb{R}^p \to \mathbb{R}^n$ such that $\gamma(z_1, z_2, w) = 0$, $\forall z_1 = z_2 \in Z$ and $\forall w \in \mathbb{R}^p$. Similarly to what has been remarked after Assumption 1, in the last part of Assumption 3, we assume that the solution of the differential equation in (2) is well defined to admit the existence of a regime for the dynamics of the estimation error, and thus to study its asymptotic behavior.

III. Stability of the Estimation Error

Once 1) the structure of the innovation function γ is fixed, 2) the parameters, represented by the elements of the matrix L and the components of the vector $w \in \mathbb{R}^p$, are chosen, and 3) the measurement vector $y(t)$ is known, the evolution of the estimated state vector $\hat{x}(t)$ is completely determined by (2) and the initial conditions. The choice of the matrix L and the parameter vector w must ensure the asymptotic stability of the *estimation error*

$$e(t) \triangleq x(t) - \hat{x}(t).$$

Toward this end, we shall construct a Lyapunov function whose time derivative is negative when computed on the trajectories of the estimation error. Thanks to the smoothness of the function f (see Assumption 1), the structure (2) chosen for the estimator, and the hypotheses on γ (see Assumption 3), we shall obtain the negative definiteness of the derivative of the Lyapunov function on the whole set $X \times \{E \setminus \{0\}\}$ by imposing it only on a finite grid of points, provided that such a grid is suitably chosen. Theorem 1 formalizes the aforesaid.

Theorem 1: Consider the state observer (2) for the system (1) and suppose that Assumptions 1, 2, and 3 hold. Let $E \subset \mathbb{R}^n$ be a connected, compact set to which the estimation error belongs; moreover, let $S \triangleq X \times \{E \setminus \{0\}\}$ and, given a positive integer M, denote by $S_M \subset S$ a finite set of points $s_i \triangleq \mathrm{col}(x_i, e_i)$, $i = 1, 2, \ldots, M$, corresponding to the first M points of a sequence $\{s_i\}$. Let $V(e) = e^{\mathrm{T}} P e$ be a Lyapunov function, where $P \in \mathbb{R}^{n \times n}$ is the symmetric, positive-definite matrix that solves the matrix equation

$$(A - LC)^{\mathrm{T}} P + P(A - LC) = -Q \tag{3}$$

where $L \in \mathbb{R}^{n \times m}$ is a gain matrix and $Q \in \mathbb{R}^{n \times n}$ is symmetric and positive definite. Moreover, let $\dot{V} \triangleq dV/dt$ and suppose there exists a vector $w^* \in \mathbb{R}^p$ such that

$$\epsilon_{\dot{V}} \triangleq -\max_{\bar{s} \in S_M} \dot{V}(\bar{s}, w^*) > 0. \tag{4}$$

Finally, suppose the set S_M is such that

$$\theta(S_M) < \frac{\epsilon_{\dot{V}}}{\lambda_{\dot{V}}} \tag{5}$$

where $\lambda_{\dot{V}}$ is the Lipschitz constant[1] of the function $\dot{V}(s, w)$ with respect to s for $w = w^*$, i.e., $|\dot{V}(s, w^*) - \dot{V}(s', w^*)| \leq \lambda_{\dot{V}} \|s - s'\|$, $\forall s, s' \in S$, and

$$\theta(S_M) \triangleq \sup_{s \in S} \min_{\bar{s} \in S_M} \|s - \bar{s}\|. \tag{6}$$

Then, for any $e(0) \in E$, the estimation error $e(t)$ of the estimator (2) converges asymptotically to zero. □

Proof: From (1) and (2), we obtain the dynamics of the estimation error

$$\dot{e} = (A - LC)e + f(x) - f(\hat{x}) - \gamma(Ce, w). \tag{7}$$

Under Assumption 3, $e = 0$ is an equilibrium point of (7). The derivative of the Lyapunov function $V = e^{\mathrm{T}} P e$ is given by

$$\dot{V} = e^{\mathrm{T}} \left[(A - LC)^{\mathrm{T}} P + P(A - LC) \right] e + 2 \left[f(x) - f(\hat{x}) - \gamma(Ce, w) \right]^{\mathrm{T}} Pe.$$

Under Assumption 2, for any gain matrix L such that $A - LC$ is Hurwitz and for any symmetric positive-definite matrix Q, there exists a unique symmetric positive-definite matrix P that solves the Lyapunov equation (3) (see, e.g., [11, Th. 4.1, p. 136]). Therefore, in order to guarantee the asymptotic stability of the estimation error, we obtain $\dot{V} < 0$ along the trajectories of both the state and the estimation error under the constraints

$$2 \left[f(x) - f(x - e) - \gamma_\nu(Ce, w) \right]^{\mathrm{T}} Pe - e^{\mathrm{T}} Qe < 0 \qquad \forall x \in X \qquad \forall e \in E \setminus \{0\}. \tag{8}$$

Instead, we will prove the asymptotic convergence of the estimation error by i) imposing that $\dot{V}(s_i, w)$ be negative only at the points of S_M, ii) exploiting the regularity of $\dot{V}$, and iii) choosing a sampling rule that guarantees a suitably dense covering S_M of S. Toward this end, consider the derivative of the Lyapunov function at points $s_i \in S_M$

$$\dot{V}(s_i, w) = 2 \left[f(x_i) - f(x_i - e_i) - \gamma(Ce_i, w) \right]^{\mathrm{T}} Pe_i - e_i^{\mathrm{T}} Qe_i, \qquad i = 1, 2, \ldots, M. \tag{9}$$

In order to ensure that inequality (8) holds for $w = w^*$ (i.e., $\dot{V}(s, w^*) < 0$ for any $s \in S$), we first note that the function $\dot{V}(\cdot, w^*)$ is locally Lipschitz in S, due to the local Lipschitz continuity of the functions f and γ. For $s \in S$, let $\tilde{s} \triangleq \arg\min_{\bar{s} \in S_M} \|s - \bar{s}\|$ be the sample point in S_M closest to s. The inequality $|\dot{V}(s, w^*) - \dot{V}(\tilde{s}, w^*)| \leq \lambda_{\dot{V}} \|s - \tilde{s}\|$, $\forall s \in S$, provides

$$-\lambda_{\dot{V}} \|s - \tilde{s}\| \leq \dot{V}(s, w^*) - \dot{V}(\tilde{s}, w^*) \leq \lambda_{\dot{V}} \|s - \tilde{s}\| \qquad \forall s \in S. \tag{10}$$

[1]The constant $\lambda_{\dot{V}}$ depends on the number ν of basis functions in the parameterized innovation function, while $\epsilon_{\dot{V}}$ depends on ν and on the number M of sample points; however, to simplify the notation, we do not write these dependences.

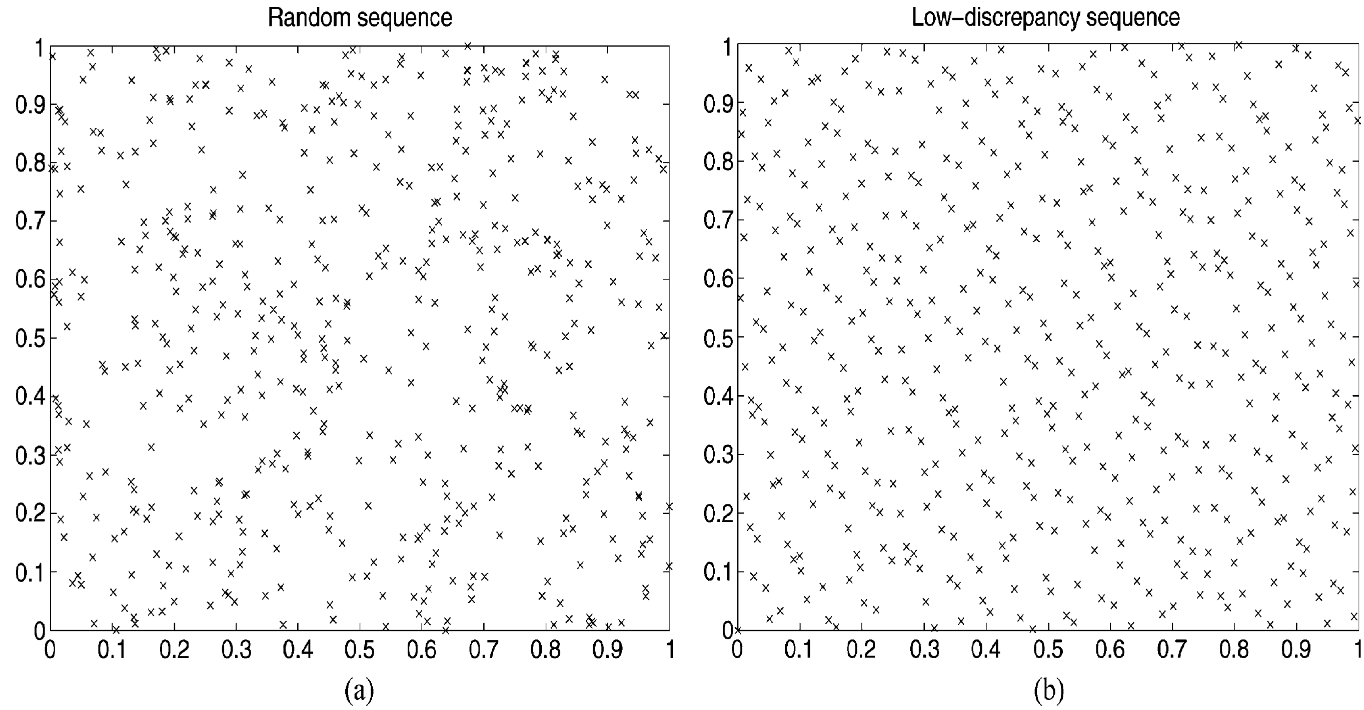


Fig. 1. Comparison between random and low-discrepancy samplings of the 2-D unit cube.

Using (4), the second inequality in (10) gives

$$\begin{aligned}\dot{V}(s,w^*) &\le \lambda_{\dot{V}}\|s-\tilde{s}\| + \dot{V}(\tilde{s},w^*)\\ &\le \lambda_{\dot{V}}\sup_{s\in S}\|s-\tilde{s}\| - \epsilon_{\dot{V}} \qquad \forall s\in S. \end{aligned}\tag{11}$$

Thus, by the definitions of $\tilde{s}$ and $\theta(S_M)$ [see (6)], we have $\theta(S_M)=\sup_{s\in S}\|s-\tilde{s}\|$ and (11) yields

$$\dot{V}(s,w^*) \le \lambda_{\dot{V}}\theta(S_M) - \epsilon_{\dot{V}}.$$

If (5) holds, then $\dot{V}(s,w^*)<0$, $\forall s\in S$ and so the inequality (8) holds; hence, using [11, Th. 4.1, p. 114], the estimation error converges asymptotically to zero. □

Condition (5) deserves some comments. The quantity $\theta(S_M)$ is called the *dispersion* of the sequence of M points of the set S_M [23] and is a measure of the uniformity of the distribution of the points of S_M in S. Roughly speaking, a small value of $\theta(S_M)$ guarantees that the points of S_M are spread on S "in a uniform way," i.e., without leaving regions of the space "undersampled," and by placing the points "close enough" to one another.

To satisfy the low-dispersion condition (5), one can use random sampling with a uniform distribution, provided that the sample size M is large enough [23]. However, the best dispersion properties, in the sense described previously, belong to special *deterministic* sequences called *low-discrepancy sequences*, commonly employed in the fields of number-theoretic methods, statistics, and quasi-Monte Carlo integration. Examples of such sequences are the (t,n)*-sequences*, the *Halton sequence*, and the *Hammersley sequence* [22], [23].

Low-discrepancy sequences share an important feature [23]: They attain *deterministically* a rate of convergence for the dispersion of order

$$O(M^{-1/2n}) \le \theta(S_M) \le O(\sqrt{2n}M^{-1/2n}).$$

Using such sequences, we are guaranteed that, for every $\epsilon_{\dot{V}}$ and $\lambda_{\dot{V}}$, there exists a suitable number $\tilde{M}$ of points such that condition (5) is satisfied, i.e., $\theta(S_M)<\epsilon_{\dot{V}}/\lambda_{\dot{V}}$ for every $M>\tilde{M}$.

For what concerns the actual number of points needed to fulfill the condition (5), there exist nonasymptotic bounds on discrepancy and dispersion [23]. Yet, such bounds are often very conservative, and in practice one finds that satisfactory results can be obtained by using a "reasonably small" number of discretization points, as is the case with the experimental results of Section VI.

As compared with random sampling, a discretization based on low-discrepancy sequences suffers less from the formation of clusters of points in particular regions of the space [22] (such formation undermines the sampling uniformity). Fig. 1 shows the comparison between a sampling of the two-dimensional (2-D) unit cube by a sequence of 500 points i.i.d. according to the uniform distribution and by a sampling of the same cube obtained via a low-discrepancy sequence (in this case, the *Sobol' sequence* [34]). It can be clearly seen how the space is better covered by the second sequence, as well as how the largest empty spaces among the points appear in the first sampling scheme.

To determine the parameter vector $w^*\in\mathbb{R}^p$ that satisfies (4), we exploit a penalty method [21, p. 397]. For a fixed $\Delta>0$, we define the cost function

$$J_M(w) = \sum_{s_i\in S_M}\left\{\max\left[0,\dot{V}(s_i,w)+\Delta\right]\right\}^2 \tag{12}$$

and we introduce the nonlinear programming problem that consists in finding $w^\circ\in\mathbb{R}^p$ such that

$$w^\circ \in \arg\min_{w\in\mathbb{R}^p} J_M(w). \tag{13}$$

By its definition, $J_M(w) \geq 0$ for every $w \in \mathbb{R}^p$ and, by inspection of the proof of Theorem 1, the conditions (4) and (5) imply that there exists $w^* \in \mathbb{R}^p$ such that $\dot{V}(s, w^*) < 0, \forall s \in S$. Hence, by taking w^* as in (4) and $\Delta = \min_{s_i \in S_M} |\dot{V}(s_i, w^*)|$, one has $J_M(w^*) = 0$, and so $w^* \in \arg\min_{w \in \mathbb{R}^p} J_M(w)$. On the other hand, if $\bar{w} \in \arg\min_{w \in \mathbb{R}^p} J_M(w)$, then $\dot{V}(s_i, \bar{w}) < 0, \forall s_i \in S_M$; hence, in Theorem 1 one can take $w^* = \bar{w}$.

In Section IV, we will consider certain parameterized innovation functions, including some widely used neural networks, the choice of which allows one to satisfy the condition (4), hence (on the basis of the discussion reported previously) to guarantee that $\arg\min_{w \in \mathbb{R}^p} J_M(w)$ is nonempty.

Note that there may be different values of w° that result from the minimization in (13), i.e., the solution of the minimization problem may be nonunique. In particular, for commonly used feedforward networks, there exist transformations of the parameter vector w leaving the input/output mapping unchanged [35], so the minimum point of $J_M(\cdot)$ is not unique, i.e., in general, $\arg\min_{w \in \mathbb{R}^p} J_M(w)$ is not a singleton.

At this point, the design of the observer is reduced to the solution of the optimization problem (13). Its solution can be obtained via nonlinear programming techniques, which allow one to find *offline* the values of the parameter vector w°. This is an important advantage of our approach over other neural approaches to estimation proposed in the literature (see, e.g., [28]–[31]), which are based on the online optimization of parameters, possibly involving large computational efforts in real-time applications.

IV. Class of One-Hidden-Layer Networks in the Structure of the Observer

So far, we have considered the problem of designing the estimator (2) for the system (1) by searching for a matrix $L \in \mathbb{R}^{n \times m}$ and a vector $w \in \mathbb{R}^p$ associated with a suitable Lyapunov function for the estimation error. In order to attain this objective in a convenient way, we further restrict the class (2) of observers by considering functions γ that not only satisfy Assumption 3 but also take on a special structure. For a positive integer ν, they are expressed as linear combinations of ν functions with a fixed structure and dependent on a vector $w_\nu \in \mathbb{R}^p$ of parameters. By $\mathcal{C}(K, \mathbb{R}^n)$, we denote the space of continuous n-valued functions on a compact set $K \subset \mathbb{R}^m$, equipped with the supremum norm. For every positive integer ν, we define the following class of functions.

Definition 1: Let $K \subset \mathbb{R}^m$ be compact, ν and $l_i, i = 1, \ldots, \nu$ be positive integers, $\varphi_i : K \times \mathbb{R}^{l_i} \to \mathbb{R}$, and $\mathcal{N}(\nu) = \sum_{i=1}^{\nu} l_i + n\nu$. We call one-hidden-layer networks the functions belonging to the set

$$A_\nu \triangleq \left\{ \gamma_\nu : K \times \mathbb{R}^{\mathcal{N}(\nu)} \to \mathbb{R}^n \text{ such that} \right.$$

$$1)\ \gamma_{\nu j}\left(\xi, \omega_{\nu_j}\right) = \sum_{i=1}^{\nu} c_{ij} \varphi_i(\xi, \kappa_i), \qquad \exists\, \bar{c} \in \mathbb{R}^+$$
$$\text{such that } |c_{ij}| \leq \bar{c},\ \kappa_i \in \mathbb{R}^{l_i}, \qquad i = 1, \ldots, \nu,$$
$$j = 1, \ldots, n,\ \omega_{\nu j} \triangleq \text{col}(c_{ij}, \kappa_i : i = 1, \ldots, \nu);$$

$$2) \text{ the functions } \varphi_i(\cdot, \kappa_i) \text{ are Lipschitz, i.e.,}$$
$$\forall i = 1, \ldots, \nu\ \exists\, \lambda_i \in \mathbb{R}^+ \text{ such that}$$
$$\forall \kappa_i \in \mathbb{R}^{l_i},\ |\varphi_i(\xi, \kappa_i) - \varphi_i(\xi', \kappa_i)| \leq \lambda_i |\xi - \xi'|;$$
$$3)\ \varphi_i(0, \kappa_i) = 0,\ i = 1, \ldots, \nu \qquad \forall \kappa_i \in \mathbb{R}^{l_i};$$
$$4) \bigcup_{\nu=1}^{\infty} A_\nu \text{ is dense in } C(K, \mathbb{R}^m) \text{ with respect to}$$
$$\left. \text{the supremum norm.} \right\} \qquad \square$$

To avoid burdening the notation, we do not write explicitly the dependence of A_ν on the integers m, l_i, and n, the reals $\bar{c}$, $\tilde{c}$, and λ_i, the set K, and the functions φ_i.

If one uses a one-hidden-layer network as a parameterized innovation function in (2), then the observer takes on the form

$$\dot{\hat{x}} = A\hat{x} + f(\hat{x}) + L(y - C\hat{x}) + \gamma_\nu(y - C\hat{x}, w_\nu), \qquad t \geq 0 \quad (14)$$

where $\gamma_\nu \in A_\nu$.

The definition of A_ν characterizes one-hidden-layer networks as a particular kind of nonlinear approximators called *variable-basis functions* [17], [18], [36]. For $i = 1, \ldots, \nu$, the mappings $\varphi_i(\cdot, \cdot)$ play the role of "mother functions," and the coefficients c_{ij} and the components of the vectors κ_i are "free" parameters, lumped together in the vector $w_\nu \triangleq \text{col}(w_{\nu j}, j = 1, \ldots, n) \in \mathbb{R}^{\mathcal{N}(\nu)}$. In practice, as the parameters κ_i inside the basis functions allow high flexibility, variable-basis functions are typically obtained by changing such parameters in a unique mother function $\varphi(\cdot, \cdot)$, i.e., one usually takes $\varphi_i(\cdot, \cdot) = \varphi(\cdot, \cdot)$ for every $i = 1, \ldots, \nu$. The properties of variable-basis approximation and their application to optimization problems have been studied in [17]–[19], [36]–[38].

Note that items 2) and 3) in Definition 1 take into account the requirements of Assumption 3 (the functions γ_ν are Lipschitz, as they are the sums of ν Lipschitz functions). Then, we have Proposition 1.

Proposition 1: For any positive integer ν, one-hidden-layer networks A_ν are admissible functions γ in the observer (2) for the system (1). $\square$

Once a type of one-hidden-layer network has been chosen, i.e., once a mother function $\varphi(\cdot, \cdot)$ has been selected, the Lyapunov function for the observer (14) depends on the values of the entries of the matrix L and of the components of the vector w_ν. As to the function φ, it is suitable to make a choice generating sets A_ν that have as large as possible a closure in the supremum norm on K. Loosely speaking, the larger such a closure is, the wider the choice at our disposal for a Lyapunov function becomes. We will use one-hidden-layer networks that are dense in the space of continuous functions on compact sets, i.e., in the neural-network parlance, that exhibit the *universal approximation property* [39]. Well-known examples of one-hidden-layer networks are feedforward neural networks of the perceptron type, with at most ν hidden units and bounded parameters, and radial basis function (RBF) networks with at most ν hidden units and bounded input weights and variances. The following are examples of sets A_ν obtained by basis functions corresponding to computational units widely used in neurocomputing.

1) Feedforward neural networks of the perceptron type, with at most ν hidden units and bounded parameters

$$\mathcal{P}_\nu \triangleq \Big\{ \gamma_\nu : K \times \mathbb{R}^{\nu(m+n+1)-m-1} \to \mathbb{R}^n \text{such that} \\ \gamma_{\nu j}(\xi,\omega_{\nu j}) = \sum_{i=1}^{\nu-1} c_{ij}\varphi(\xi^{\mathrm{T}}\alpha_i+\beta_i)+\eta_j, \text{ where} \\ \omega_{\nu j} \triangleq \mathrm{col}(c_{ij},\alpha_i,\beta_i, i=1,\ldots,\nu-1,\eta_j), \\ j=1,\ldots,n; \text{ and} \\ \varphi:\mathbb{R}\to\mathbb{R} \text{ is bounded, Lipschitz, and nonpolynomial;} \\ c_{ij},\beta_i \in \mathbb{R}, \alpha_i \in \mathbb{R}^m, \ \exists\, c\in\mathbb{R}^+ \text{ such that} \\ \max_{i,j}\{|c_{ij}|\} \le c \Big\}. \tag{15}$$

2) Radial-basis-functions with at most ν hidden units and bounded parameters

$$\mathcal{R}_\nu \triangleq \Big\{ \gamma_\nu : K \times \mathbb{R}^{\nu(m+n+1)-m-1} \to \mathbb{R}^n \text{ such that} \\ \gamma_{\nu j}(\xi,\omega_{\nu j}) = \sum_{i=1}^{\nu-1} c_{ij}\varphi\left(\frac{\|\xi-\tau_i\|}{b_i}\right)+\eta_j, \text{ where} \\ \omega_{\nu j} \triangleq \mathrm{col}(c_{ij},\tau_i,b_i, i=1,\ldots,\nu-1,\eta_j), \\ j=1,\ldots,n; \text{ and} \\ \varphi:\mathbb{R}\to\mathbb{R} \text{ is bounded, Lipschitz, and radial;} \\ c_{ij},\ b_i \in \mathbb{R},\ b_i>0,\ \tau_i \in \mathbb{R}^m, \ \exists\, c\in\mathbb{R}^+ \text{ such that} \\ \max_{i,j}\{|c_{ij}|\} \le c, \Big\}. \tag{16}$$

To simplify the notation, we have omitted the dependences of $\mathcal{P}_\nu$ and $\mathcal{R}_\nu$ on c, the integers m and n, the set K, and the function φ.

Note that the definitions of P_ν and R_ν entail a summation of $\nu-1$ parameterized functions with the same structure φ, plus a constant that plays the role of the νth function, required to account for a "bias." In the definition of A_ν, the explicit introduction of the bias term is not necessary, as the summation is over ν possibly different functions φ_i (so the biases can be "modeled" by one of such functions).

The proofs of the fact that the functions in $\mathcal{P}_\nu$ and $\mathcal{R}_\nu$ are provided with the density property in the space $\mathcal{C}(K,\mathbb{R}^n)$, where $K\subset\mathbb{R}^m$ is compact, can be found, for example, in [13] and [14] for feedforward neural networks of the perceptron type and in [15] and [16] for RBF networks.

In general, the Lipschitz constants of the functions in $\mathcal{P}_\nu$ and $\mathcal{R}_\nu$ depend on the number ν of basis functions, and generally grow with it. A sufficient condition to have Lipschitz constants independent of ν lies in imposing for some $u>0$ also the overall condition $\max_j\{\sum_{i=1}^{\nu-1}|c_{ij}|+\eta_j\}\le u$ on the coefficients of the linear combination of parameterized basis functions; unfortunately, in such a case, the density requirement cannot be guaranteed [14].

It is worth noting that the procedure used in our algorithm to optimize the parameters (i.e., the vector of the weights of the neural network) is quite different from standard neural network training, where the goal is to minimize the distance of the network output from given target values. This procedure, used also in [40], is sometimes called *distal training* and applies ad-hoc techniques, which are often modified versions of standard minimization algorithms (for such algorithms, see, e.g., [41], [42], and the references therein). Our approach is somehow dual to the one reported in [40] to design closed-loop neural controllers, where the constraints are satisfied by minimizing a quadratic penalty function via a specialized version of the Levenberg–Marquardt algorithm.

We conclude this section with a remark on another desirable property of certain one-hidden-layer networks. When the innovation function depends on a large number m of measurements, a very large number ν of basis functions might be required by a one-hidden-layer network to guarantee the desired estimation accuracy ε. If, for a fixed value of ε, the growth of ν as a function of m is very fast (e.g., an exponential growth), then the computational requirements might become unacceptable for large values of m. This behavior is an instance of the so-called *curse of dimensionality* [43]. To guarantee the possibility of finding an observer implemented with a "small" number ν of basis functions also for estimation problems with large m, we can resort to special families A_ν of parameterized functions [19]. Such families exhibit the important and highly desirable property that, under not too restrictive conditions, the number ν of basis functions required to guarantee a fixed approximation accuracy has to grow at most polynomially with the number m of components of the measurement vector (see [17]–[20] for details). Feedforward neural networks with suitable choices of the basis functions show this property, and will be used in the numerical simulations reported in Section VI.

V. Design Procedure

On the basis of Theorem 1 and of the choices of elements of low-discrepancy sequences as discretization points and neural networks as parameterized functions, discussed in Sections III and IV, respectively, we propose the following procedure for the design of the observer.

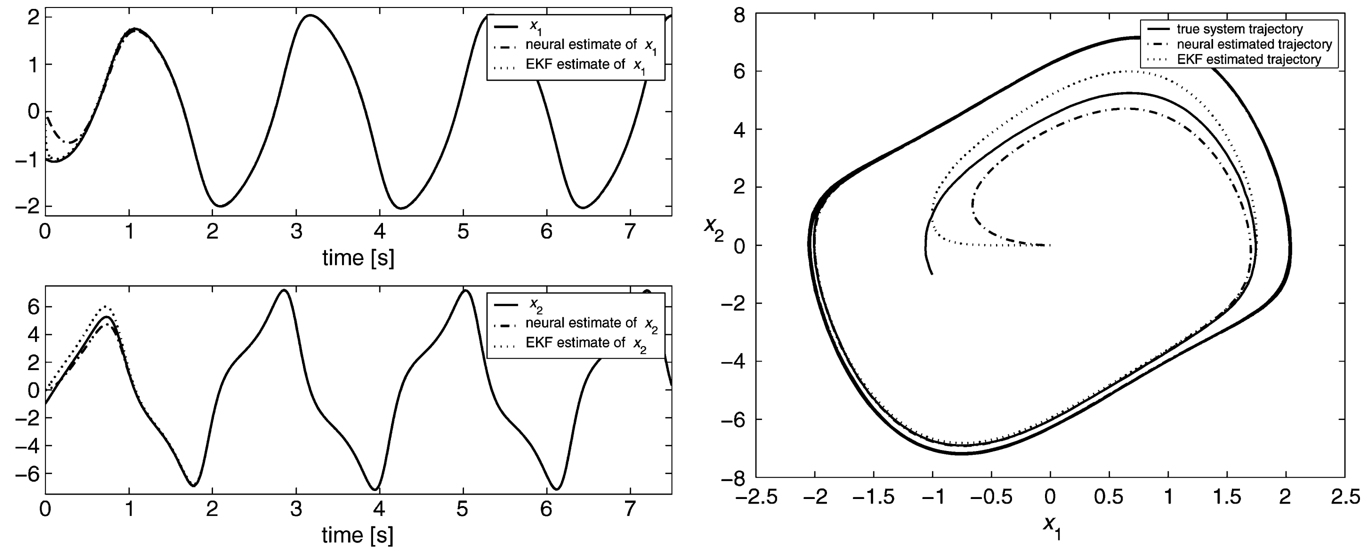
Given: $X\subset\mathbb{R}^n$ compact, $A\in\mathbb{R}^{n\times n}$, $C\in\mathbb{R}^{m\times n}$, $f:X\to\mathbb{R}^n$.

1) Find a gain matrix L such that $A-LC$ is Hurwitz and take a symmetric, positive definite matrix Q. Let P be the unique symmetric, positive-definite matrix solving the Lyapunov equation

$$(A-LC)^{\mathrm{T}}P+P(A-LC)=-Q$$

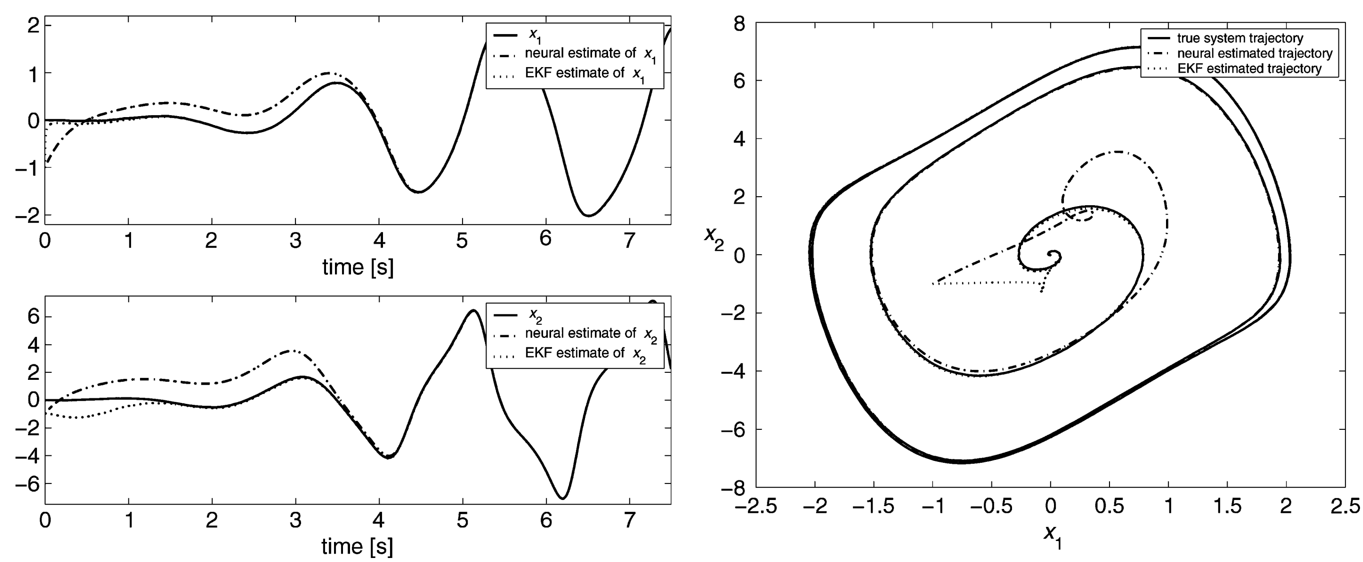
and define the quadratic Lyapunov function $V=e^{\mathrm{T}}Pe$.

2) Choose a connected, compact set $E\subset\mathbb{R}^n$, positive integers M and ν, and $\Delta>0$.
3) Choose a basis function φ and construct the set A_ν of admissible innovation functions.
4) Generate a low-discrepancy sequence of sample points s_i, $i=1,\ldots,M$, belonging to $S_M\subset X\times\{E\setminus\{0\}\}$.



(a) (b)

Fig. 2. Numerical results for $p_0 = 0.01$, $q = 0.001$, $r = 0.01$, $x(0) = [-1 \ -1]^{\mathrm{T}}$, and $\hat{x}(0) = [0 \ 0]^{\mathrm{T}}$.



(a) (b)

Fig. 3. Numerical results for $p_0 = 0.01$, $q = 0.001$, $r = 0.01$, $x(0) = [0 \ 0]^{\mathrm{T}}$, and $\hat{x}(0) = [-1 \ -1]^{\mathrm{T}}$.

5) Let

$$J_M(w) = \sum_{s_i \in S_M} \left\{ \max\left[0, \dot{V}(s_i, w) + \Delta\right] \right\}^2$$

and

$$w^{\circ}_{\nu,M} \in \underset{w \in \mathbb{R}^p}{\operatorname{argmin}} \, J_M(w).$$

6) If $J_M(w^{\circ}_{\nu,M}) > 0$, then decrease $\Delta > 0$, increase ν, and go to step 5).

7) Let $\lambda_{\dot{V}} \in \mathbb{R}^+$ be such that

$$\left| \dot{V}\left(s, w^{\circ}_{\nu,M}\right) - \dot{V}\left(s', w^{\circ}_{\nu,M}\right) \right| \leq \lambda_{\dot{V}} \|s - s'\|.$$

Let $\epsilon_{\dot{V}} = -\max_{\bar{s} \in S_M} \dot{V}(\bar{s}, w^{\circ}_{\nu,M})$ and $\theta(S_M) = \sup_{s \in S} \min_{\bar{s} \in S_M} \|s - \bar{s}\|$.

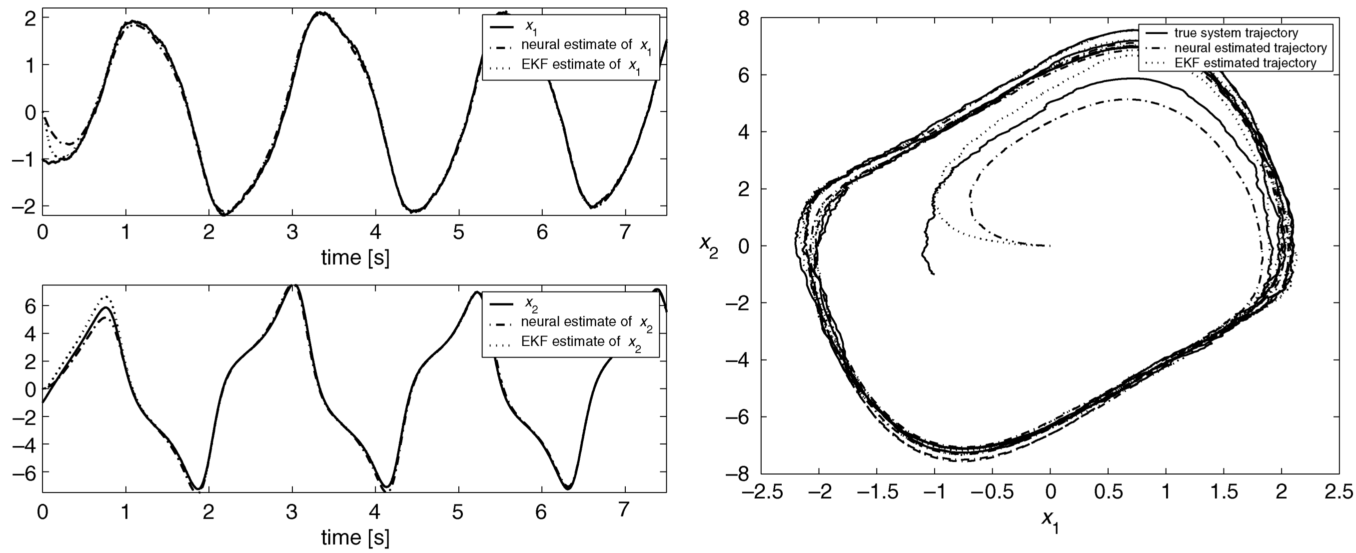
8) If

$$\theta(S_M) < \frac{\epsilon_{\dot{V}}}{\lambda_{\dot{V}}}$$

then stop, else increase M and go to step 4).

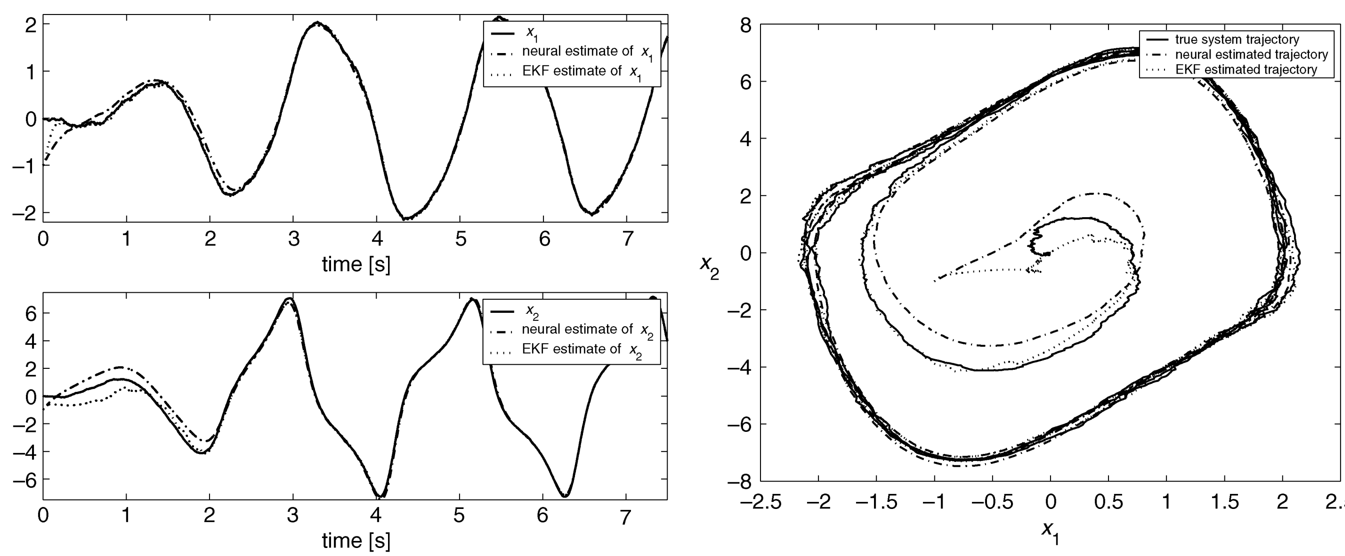
The selection of the connected, compact set $E \subset \mathbb{R}^n$ in step 2) can be made on the basis of some *a priori* knowledge of the magnitude of the estimation error.

The condition $J_M(w^{\circ}_{\nu,M}) > 0$ in step 6) occurs if the constraints $\dot{V}(s_i, w^{\circ}_{\nu,M}) < 0$, $\forall s_i \in S_M$ are not satisfied. Decreasing $\Delta > 0$ and increasing ν, it may be easier to fulfill such constraints. For a similar reason, one may increase M when the sampling points belonging to S_M are not so spread as to let $\theta(S_M) < \epsilon_{\dot{V}}/\lambda_{\dot{V}}$.



(a) (b)

Fig. 4. Numerical results for $p_0 = 0.01$, $q = 0.01$, $r = 0.1$, $x(0) = [-1\ \ -1]^{\mathrm{T}}$, and $\hat{x}(0) = [0\ 0]^{\mathrm{T}}$.



(a)

Fig. 5. Numerical results for $p_0 = 0.01$, $q = 0.01$, $r = 0.1$, $x(0) = [0\ 0]^{\mathrm{T}}$, and $\hat{x}(0) = [-1\ \ -1]^{\mathrm{T}}$.

VI. Numerical Results

Let us consider a Van der Pol equation with the vector $x = (x_1, x_2)^{\mathrm{T}}$ to be estimated by using only the measurements of x_1

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -9x_1 + 2\left(1 - x_1^2\right)x_2 \\ y &= x_1. \end{aligned} \tag{17}$$

As is well known, the dynamics of (17) is not stable to zero since it admits a limit cycle. Referring to the system model (1), from (17), we have

$$A = \begin{pmatrix} 0 & 1 \\ -9 & 0 \end{pmatrix} \quad f(x) = \begin{bmatrix} 0 \\ 2\left(1 - x_1^2\right)x_2 \end{bmatrix} \quad C = (1 \quad 0).$$

Note that the pair (A, C) is observable.

The design of the observer (14) for the system (17) needs first the selection of a type of neural network, according to the requisites in Section IV. Toward this end, we chose a one-hidden-layer sigmoidal feedforward neural network belonging to the class $\mathcal{P}_\nu$ defined in (15), with $\nu = 5$ neural units and output biases equal to zero. We took the gain matrix

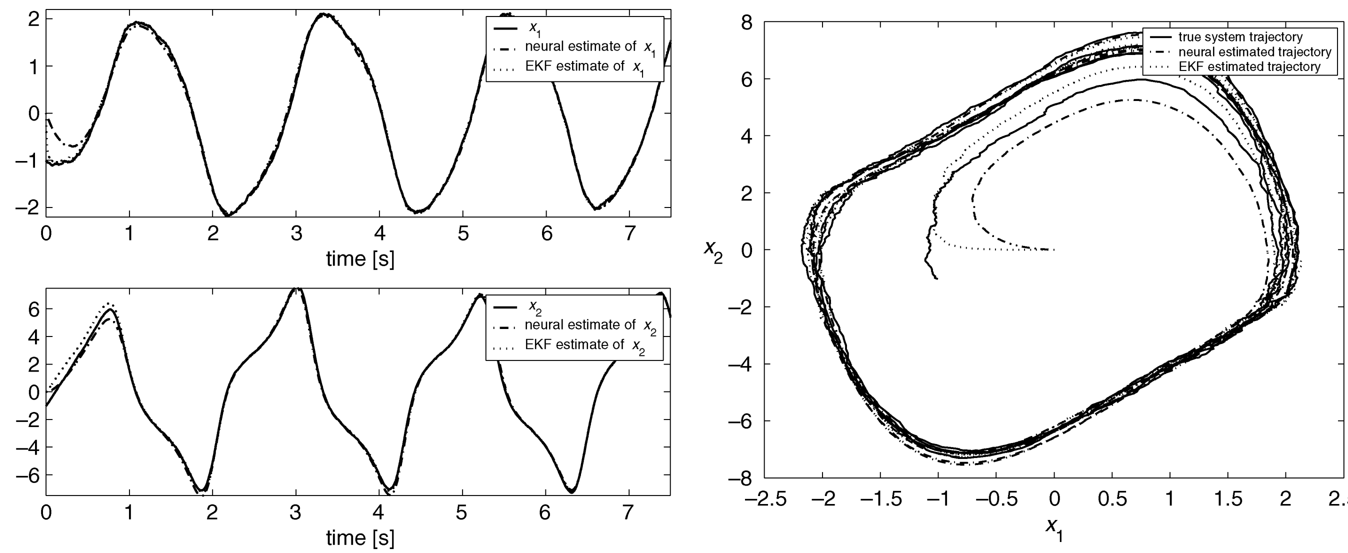
$$L = \begin{pmatrix} 0.03 \\ -0.06 \end{pmatrix}$$

and so the eigenvalues of $A - LC$ have negative real parts.

The solution of the Lyapunov (3) with Q equal to the identity gave

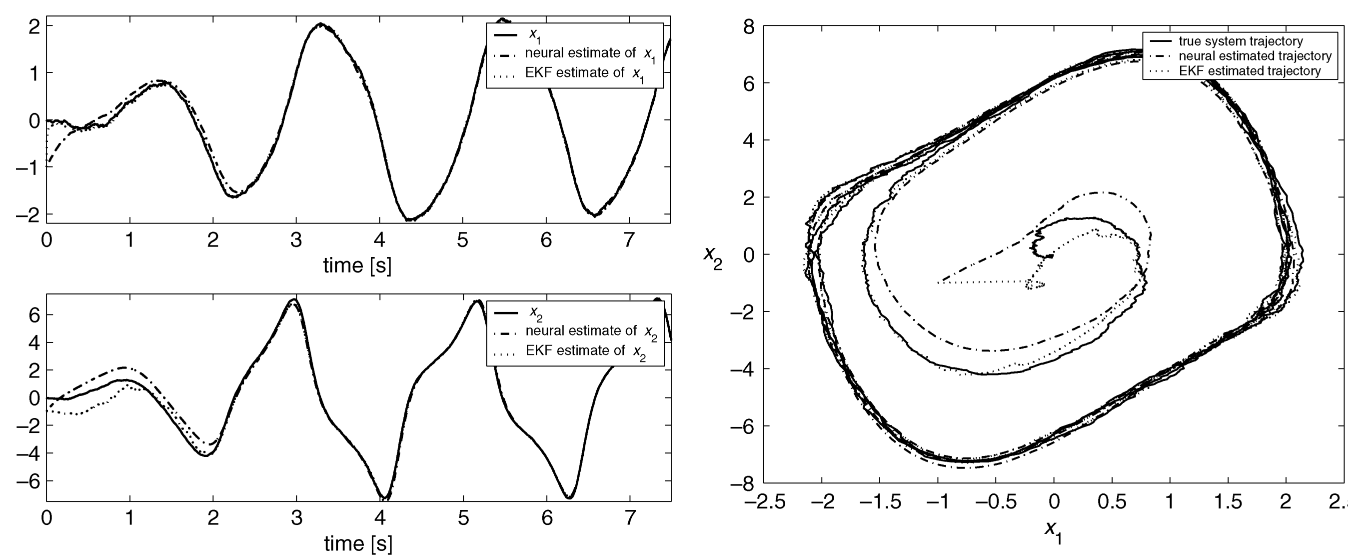
$$P = \begin{pmatrix} 15.6667 & -0.5000 \\ -0.5000 & 1.8829 \end{pmatrix}.$$

Once the matrix P was found, the weights of the neural network were determined by minimizing the cost (12) on a sufficiently "rich" grid on $X \times \{E \setminus \{0\}\}$, where $X = [-3, 3] \times [-7, 7]$ and



(a) (b)

Fig. 6. Numerical results for $p_0 = 0.1$, $q = 0.01$, $r = 0.1$, $x(0) = [-1\ -1]^{\mathrm{T}}$, and $\hat{x}(0) = [0\ 0]^{\mathrm{T}}$.



(a) (b)

Fig. 7. Numerical results for $p_0 = 0.1$, $q = 0.01$, $r = 0.1$, $x(0) = [0\ 0]^{\mathrm{T}}$, and $\hat{x}(0) = [-1\ -1]^{\mathrm{T}}$.

$E = [-3, 1] \times [-6, 4]$. We verified that a discretization made up of 1500 points from a low-discrepancy sequence (specifically, a Sobol' sequence [34]), suitably scaled to fit the set $X \times E$, was enough to give satisfactory results with Δ equal to 0.07.

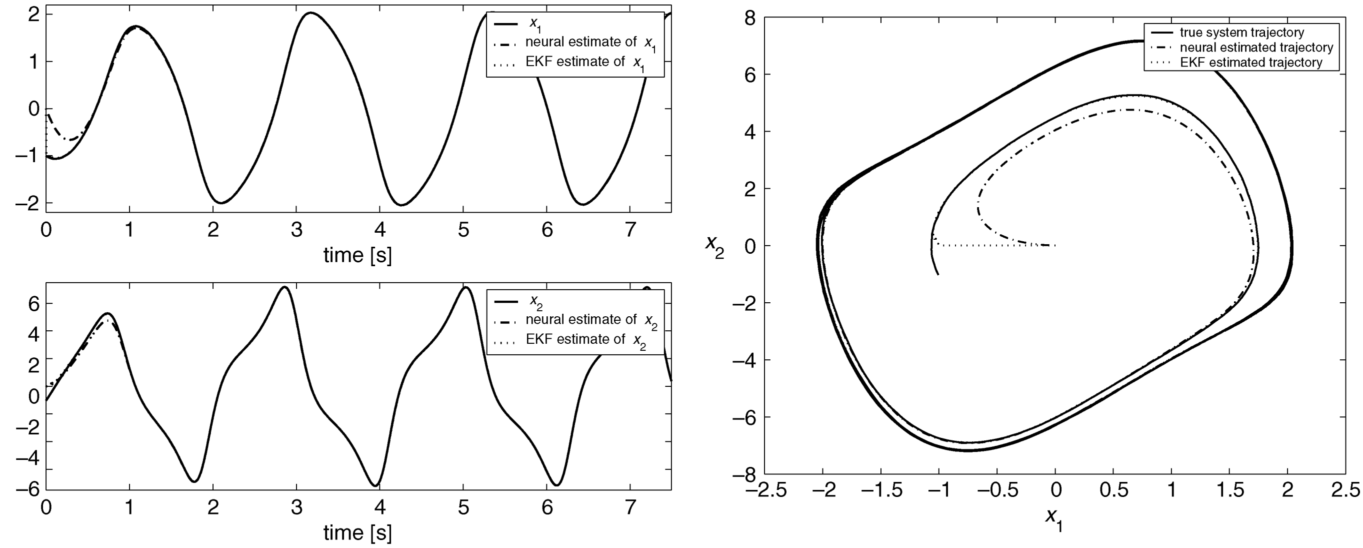
The minimization was performed by using a standard Matlab routine of the optimization toolbox [44].

For the purpose of a comparison with the EKF, we considered additive, truncated, Gaussian, zero-mean random disturbances acting on the dynamic and measurement equations in (17) with the covariance matrices of the initial state, the system noise, and the measurement noise equal to $p_0 I$, qI, and r, respectively.

The results obtained under different conditions and four different values of p_0, q, and r are shown in Figs. 2–9. It is worth noting that such information on the statistics of the initial state and of the disturbances is used only by the EKF, i.e., the parameters p_0, q, and r can be used to tune this filter.

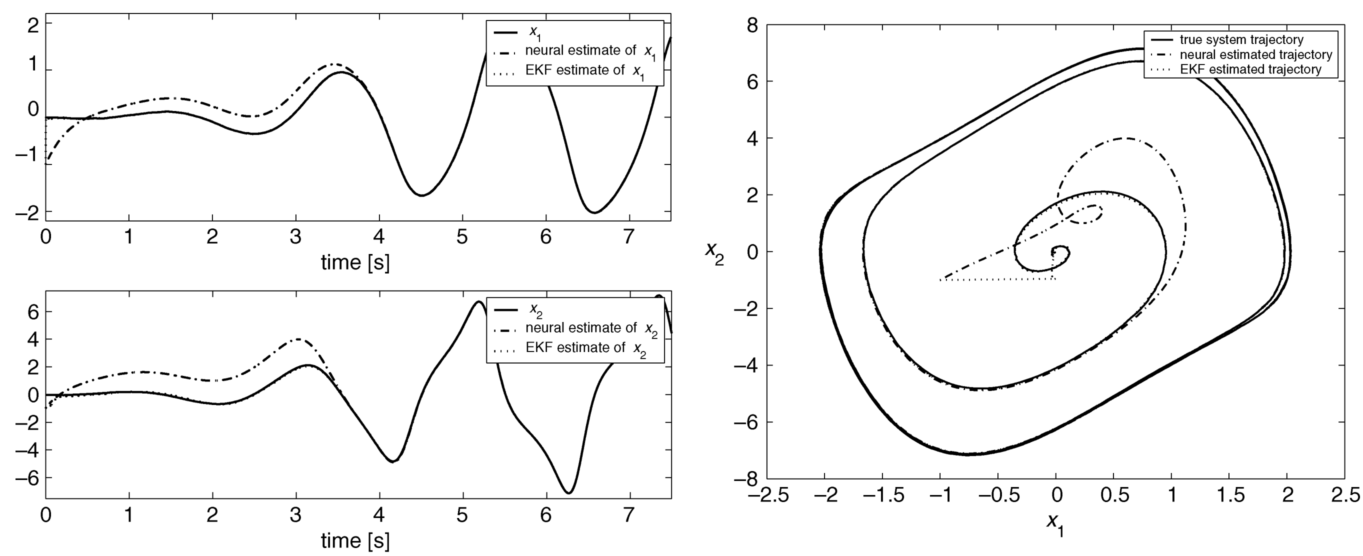
VII. Conclusion

We have presented a methodology based on neural networks and nonlinear programming to design state observers for a class of nonlinear continuous-time dynamic systems. The contribution of this work is three-fold. First, a structure of the estimator has been considered that depends on an innovation function made up of two terms, i.e., a linear gain and a parameterized nonlinear mapping that can be taken on by certain feedforward neural networks. The gain and the parameters (e.g., the neural weights) may be chosen such as to guarantee the asymptotic convergence of the estimation error to zero by finding a quadratic Lyapunov function. Second, such a Lyapunov function is searched for by constraining its derivative to be negative on a well-shaped sampling grid of points; this is accomplished by minimizing offline a cost function that penalizes the



(a) (b)

Fig. 8. Numerical results for $p_0 = 0.1$, $q = 0.001$, $r = 0.01$, $x(0) = [-1\ -1]^{\mathrm{T}}$, and $\hat{x}(0) = [0\ 0]^{\mathrm{T}}$.



(a) (b)

Fig. 9. Numerical results for $p_0 = 0.1$, $q = 0.001$, $r = 0.01$, $x(0) = [0\ 0]^{\mathrm{T}}$, and $\hat{x}(0) = [-1\ -1]^{\mathrm{T}}$.

constraints that are not satisfied. Third, as to the choice of the sampling points, we have exploited the so-called "low-discrepancy sequences," which ensure a deterministic convergence at favorable rates when the number of sampling points grows.

The proposed state estimator does not require a state-space coordinate change and a global Lipschitz assumption, as needed, for example, by the high-gain observer. As compared with the EKF, we have proved the asymptotic stability of the estimation error under any initial condition on a compact set, whereas, up to now, the error of the EKF has been demonstrated to be only locally stable. Simulation results on a simple numerical example confirm the effectiveness of our approach in comparison with the EKF.

REFERENCES

[1] D. G. Luenberger, "Observing the state of a linear system," *IEEE Trans. Mil. Electron.*, vol. ME-8, pp. 74–80, Apr. 1964.

[2] F. E. Thau, "Observing the state of non-linear dynamic systems," *Int. J. Control*, vol. 17, no. 3, pp. 471–479, 1973.

[3] S. R. Kou, D. L. Elliott, and T. J. Tarn, "Exponential observers for nonlinear dynamic systems," *Inf. Control*, vol. 29, pp. 204–216, 1975.

[4] A. J. Krener and A. Isidori, "Linearization by output injection and nonlinear observers," *Syst. Control Lett.*, vol. 3, pp. 47–52, 1983.

[5] A. J. Krener and W. Respondek, "Nonlinear observer with linearizable error dynamics," *SIAM J. Control Optim.*, vol. 23, no. 2, pp. 197–216, 1985.

[6] B. L. Walcott and S. H. Zak, "State observation of nonlinear uncertain dynamical systems," *IEEE Trans. Autom. Control*, vol. 32, no. 2, pp. 166–170, Feb. 1987.

[7] J. P. Gauthier, H. Hammouri, and S. Othman, "A simple observer for nonlinear systems applications to bioreactors," *IEEE Trans. Autom. Control*, vol. 37, no. 6, pp. 875–880, Jun. 1992.

[8] G. Ciccarella, M. Dalla Mora, and A. Germani, "A Luenberger-like observer for nonlinear systems," *Int. J. Control*, vol. 57, no. 3, pp. 537–556, 1994.

[9] M. D. M. Boutayeb and H. Rafaralahy, "Convergence analysis of the extended Kalman filter used as an observer for nonlinear deterministic discrete-time systems," *IEEE Trans. Autom. Control*, vol. 42, no. 4, pp. 581–586, Apr. 1997.

[10] J. P. Gauthier and I. A. K. Kupka, "Observability and observers for nonlinear systems," *SIAM J. Control Optim.*, vol. 32, no. 4, pp. 975–994, 1994.
[11] H. K. Khalil, *Nonlinear Systems*. Upper Saddle River, NJ: Prentice Hall, 2002.
[12] A. Alessandri, C. Cervellera, A. F. Grassia, and M. Sanguineti, "Design of observers for continuous-time nonlinear systems using neural networks," in *Proc. Amer. Control Conf.*, Boston, MA, 2004, pp. 2433–2438.
[13] M. Leshno, V. Ya, A. Pinkus, and S. Schocken, "Multilayer feedforward networks with a nonpolynomial activation function can approximate any function," *Neural Netw.*, vol. 6, pp. 861–867, 1993.
[14] V. Kůrková, "Approximation of functions by perceptron networks with bounded number of hidden units," *Neural Netw.*, vol. 8, pp. 745–750, 1995.
[15] J. Park and I. W. Sandberg, "Universal approximation using radial-basis-function networks," *Neural Comput.*, vol. 3, pp. 246–257, 1991.
[16] F. Girosi, "Regularization theory, radial basis functions and networks," in *From Statistics to Neural Networks. Theory and Pattern Recognition Applications*, ser. F: Computer and Systems Sciences. New York: Springer-Verlag, 1994, pp. 166–187.
[17] V. Kurková and M. Sanguineti, "Comparison of worst case errors in linear and neural network approximation," *IEEE Trans. Inf. Theory*, vol. 48, no. 1, pp. 264–275, 2002.
[18] ——, "Error estimates for approximate optimization by the extended Ritz method," *SIAM J. Optim.*, vol. 15, pp. 461–487, 2005.
[19] R. Zoppoli, M. Sanguineti, and T. Parisini, "Approximating networks and extended Ritz method for the solution of functional optimization problems," *J. Optim. Theory Appl.*, vol. 112, no. 2, 2002.
[20] S. Giulini, M. Sanguineti, and R. Zoppoli, "Approximation schemes for functional optimization problems," *J. Optim. Theory Appl.*, to be published.
[21] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed. Belmont, MA: Athena Scientific, 2003.
[22] K.-T. Fang and Y. Wang, *Number-Theoretic Methods in Statistics*. London, U.K.: Chapman & Hall, 1994.
[23] H. Niederreiter, *Random Number Generation and Quasi-Monte Carlo Methods*. Philadelphia, PA: SIAM, 1992.
[24] C. Cervellera and M. Muselli, "Deterministic design for neural network learning: an approach based on discrepancy," *IEEE Trans. Neural Netw.*, vol. 15, no. 3, pp. 533–544, May 2004.
[25] A. Alessandri, M. Baglietto, T. Parisini, and R. Zoppoli, "A neural state estimator with bounded errors for nonlinear systems," *IEEE Trans. Autom. Control*, vol. 44, no. 11, pp. 2028–2042, Nov. 1999.
[26] A. Alessandri and M. Sanguineti, "Optimization of approximating networks for optimal fault diagnosis," *Optim. Methods Software*, vol. 20, no. 2–3, pp. 235–260, 2005.
[27] A. Alessandri, T. Parisini, and R. Zoppoli, "Neural approximators for finite-memory state estimation," *Int. J. Control*, vol. 67, no. 2, pp. 275–302, 1997.
[28] J. Y. Choi and J. A. Farrel, "Adaptive observer backstepping control using neural networks," *IEEE Trans. Neural Netw.*, vol. 12, no. 5, pp. 1103–1112, Sep. 2001.
[29] A. G. Parlos, S. K. Menon, and A. F. Atiya, "An algorithmic approach to adaptive state filtering using recurrent neural networks," *IEEE Trans. Neural Netw.*, vol. 12, no. 6, pp. 1411–1432, Nov. 2001.
[30] X. Zhang, M. M. Polycarpou, and T. Parisini, "Robust fault isolation of a class of nonlinear input-output systems," *Int. J. Control*, vol. 74, no. 133, pp. 1295–1310, 2001.
[31] ——, "A robust detection and isolation scheme for abrupt and incipient faults in nonlinear systems," *IEEE Trans. Autom. Control*, vol. 47, no. 4, pp. 576–593, Apr. 2002.
[32] H. Shim, Y. I. Son, and J. H. Seo, "Semi-global observer for multi-output nonlinear systems," *Syst. Control Lett.*, vol. 42, pp. 233–244, 2001.
[33] X. Xia and W. Gao, "On exponential observers for nonlinear systems," *Syst. Control Lett.*, vol. 11, pp. 319–325, 1988.
[34] I. M. Sobol', "The distribution of points in a cube and the approximate evaluation of integrals," *Zh. Vychisl. Mat. i Mat. Fiz.*, vol. 7, pp. 784–802, 1967.
[35] H. J. Sussmann, "Uniqueness of the weights for minimal feedforward nets with a given input-output map," *Neural Netw.*, vol. 5, no. 4, pp. 589–593, 1992.
[36] V. Kurková and M. Sanguineti, "Bounds on rates of variable-basis and neural-network approximation," *IEEE Trans. Inf. Theory*, vol. 47, no. 6, pp. 2659–2665, Sep. 2001.
[37] P. Kainen, V. Kurková, and M. Sanguineti, "Minimization of error functionals over variable-basis functions," *SIAM J. Optim.*, vol. 14, pp. 732–742, 2003.
[38] V. Kurková and M. Sanguineti, "Learning with generalization capability by kernel methods of bounded complexity," *J. Complexity*, vol. 21, no. 3, pp. 350–367, 2005.
[39] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Netw.*, vol. 2, pp. 359–366, 1989.
[40] S. Yu and A. M. Annaswamy, "Stable neural controllers for nonlinear dynamic systems," *Automatica*, vol. 34, no. 5, pp. 641–650, 1998.
[41] L. Grippo, "Convergent on-line algorithms for supervised learning in neural networks," *IEEE Trans. Neural Netw.*, vol. 11, no. 6, pp. 1284–1299, Nov. 2000.
[42] A. Alessandri, M. Sanguineti, and M. Maggiore, "Optimization-based learning with bounded error for feedforward neural networks," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 261–273, Mar. 2002.
[43] R. Bellman, *Dynamic Programming*. Princeton, NJ: Princeton Univ. Press, 1957.
[44] T. Coleman, M. A. Branch, and A. Grace, *Optimization Toolbox User's Guide*. Natick, MA: Math Works, 1999.

Angelo Alessandri (S'88–M'97–SM'02) was born in Genova, Italy, in 1967. He received the "Laurea" degree in electronic engineering in 1992 and the Ph.D. degree in electronic engineering and computer science in 1996, both from the University of Genoa, Genova, Italy.

In 1998, he was a Visiting Scientist at the Naval Postgraduate School, Monterey, CA. From 1996 to 2005, he was a Research Scientist at the National Research Council of Italy (CNR), Genova, Italy. He is currently an Assistant Professor at the Department of Production Engineering, Thermoenergetics, and Mathematical Models (DIPTEM), University of Genoa. His research interests include optimization, neural networks, linear and nonlinear estimation, and fault diagnosis.

Dr. Alessandri is an Associate Editor for the *IEEE Control Systems Society Conference* Editorial Board, for the *IFAC Journal Engineering Applications of Artificial Intelligence*, and for the IEEE TRANSACTIONS ON NEURAL NETWORKS.

Cristiano Cervellera received the M.Sc. degree in electronic engineering in 1998 and the Ph.D. degree in electronic engineering and computer science, in 2002, both from the University of Genoa, Genova, Italy,

Since 2002, he has been a Researcher at the Institute of Intelligent Systems for Automation, the National Research Council, Genova, Italy. His research interests include neural networks, optimal control of nonlinear systems, and number-theoretic methods for optimization and approximation.

Marcello Sanguineti was born in Chiavari, Genova, Italy, in 1968. He received the "Laurea" degree in electronic engineering in 1992 and the Ph.D. degree in electronic engineering and computer science in 1996, both from the University of Genoa, Genova, Italy.

Since 1997, he has spent several periods as a Visiting Scientist and then Iinvited Lecturer at the Institute of Computer Science, Czech Academy of Sciences, Prague, Czech Republic. He is currently an Assistant Professor at the Department of Communications, Computer and System Sciences (DIST), University of Genoa, Genova, Italy. He has been the Scientific Coordinator of a NATO Collaborative Linkage Grant and of a research project between University of Genoa, National Research Council of Italy, and Czech Academy of Sciences. Currently, he coordinates Project 22–Area MC6 between the Italian and Czech Ministries of Foreign Affairs. His main research interests are approximation schemes for functional optimization problems, nonlinear programming algorithms for data mining, optimal control, and neural networks for optimization.

Dr. Sanguineti is the Associate Editor of the *International Mathematical Forum* and founding member of the Interdepartmental Center for Research in Computational Learning, the University of Genoa.