Using data-driven discrete-time models and the unscented Kalman filter to estimate unobserved variables of nonlinear systems

Luis Antonio Aguirre, Bruno Otávio S. Teixeira, and Leonardo Antônio B. Tôrres
Programa de Pós-Graduação em Engenharia Elétrica, Laboratory of Modeling, Analysis and Control of Nonlinear Systems—MACSIN, Universidade Federal de Minas Gerais, Avenida Antônio Carlos 6627, 31270-901 Belo Horizonte, Minas Gerais, Brazil

(Received 20 January 2005; revised manuscript received 2 May 2005; published 31 August 2005)

This paper addresses the problem of state estimation for nonlinear systems by means of the unscented Kalman filter (UKF). Compared to the traditional extended Kalman filter, the UKF does not require the local linearization of the system equations used in the propagation stage. Important results using the UKF have been reported recently but in every case the system equations used by the filter were considered known. Not only that, such models are usually considered to be differential equations, which requires that numerical integration be performed during the propagation phase of the filter. In this paper the dynamical equations of the system are taken to be difference equations—thus avoiding numerical integration—and are built from data without prior knowledge. The identified models are subsequently implemented in the filter in order to accomplish state estimation. The paper discusses the impact of not knowing the exact equations and using data-driven models in the context of state and joint state-and-parameter estimation. The procedure is illustrated by means of examples that use simulated and measured data.

DOI: 10.1103/PhysRevE.72.026226 PACS number(s): 05.45.Tp, 07.05.Kf

I. INTRODUCTION

The problem of parameter estimation using the method of least squares has been known since the days of Gauss. Given a set of equations that suposedly described the movement of the planets, it was desired to find parameter values such that the set of equations and parameters fitted the available data in the best way possible, according to some criterion [1]. If some assumptions are made, the least-squares estimator provides a solution to the problem above.

The state estimation problem is somewhat the opposite. Instead of estimating the parameters from a set of independent variables, state estimation will provide the independent variables (states) given the parameters and a measured signal. A solution to this problem, in the linear case, is provided by the Kalman filter (KF) [2].

In the case of nonlinear systems, the Kalman filter is inadequate. A way to overcome such a shortcoming is to linearize the system equations at each iteration and simply run the standard KF. This procedure is known as the extended KF (EKF). Intuitive as it may be, this procedure is prone to a number of numerical difficulties and it is known to fail for strongly nonlinear systems, that is, systems that will not be well approximated by a linear one.

The challenges of estimating states and eventually some parameters of nonlinear systems has received great attention recently [3]. In particular, alternatives to the EKF have been sought, such as the unscented KF (UKF) proposed in the mid-1990s (see [4] and references therein) and which has already attracted attention in the context of nonlinear dynamics [5,6]. As shown in the cited references, the UKF is a very promising and powerful tool and seems superior to the EKF in various respects.

Given that the UKF has great potential for real applications, this paper will address the UKF in a context that is somewhat more realistic than previously studied. In the former works, the dynamical equations—that governed the systems for which the states and eventually parameters were to be estimated—were assumed known. In order to handle such differential equations in the context of the filter, numerical integration was necessary. In this paper the models are not assumed known. Rather, the starting point is a set of noisy data from which a set of difference equations are built and which are used to implement the UKF. As a by-product, because the models are discrete time, no integration is required. In real-time applications this feature is most welcome. In the literature, one simulated (no noise added) example was found that used a feedforward neural network and it refers to the Mackey-Glass chaotic time series [8].

In this paper the model-bulding and filtering stages are illustrated using two examples: the simulated Lorenz system and an electronic oscillator from which data were actually recorded.

This paper is organized as follows. The problem of state estimation using the KF, EKF, and UKF is briefly reviewed in Sec. II. Model building from data is reviewed in Sec. III. Section IV reports the results obtained using the Lorenz system and an electronic oscillator. Finally, Sec. V discusses the main points of the paper.

II. STATE ESTIMATION

The state estimation problem for the system

$$\mathbf{x}(k) = A\mathbf{x}(k-1) + B\mathbf{u}(k-1) + \mathbf{w}(k-1),$$

$$\mathbf{y}(k) = C\mathbf{x}(k) + \mathbf{r}(k),$$

where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times p}$, and $C \in \mathbb{R}^{m\times n}$ are constant matrices, can be described as follows. Suppose that the only known data are the initial conditions $\mathbf{x}(0) \in \mathbb{R}^n$, the measurements $\mathbf{y}(k) \in \mathbb{R}^m$, and the control inputs $\mathbf{u}(k) \in \mathbb{R}^p$. Process noise $\mathbf{w}(k) \in \mathbb{R}^n$ and measurement noise $\mathbf{r}(k) \in \mathbb{R}^m$ are assumed white, Gaussian, and mutually independent with covariance matrices Q and R, respectively[1]. If there are no exogenous signals, then simply $\mathbf{u}(k)=\mathbf{0}\ \forall\ k$. It is desired to obtain an estimate for the state vector $\mathbf{x}(k)$, $k=1, 2, \ldots$.

The standard solution to this problem is the classical Kalman filter [2]

$$\hat{\mathbf{x}}(k|k-1) = A\hat{\mathbf{x}}(k-1|k-1) + B\mathbf{u}(k-1),$$

$$\hat{\mathbf{y}}(k|k-1) = C\hat{\mathbf{x}}(k|k-1),$$

$$P(k|k-1) = AP(k-1|k-1)A^{\mathrm{T}} + Q(k),$$

$$P_{yy}(k|k-1) = CP(k|k-1)C^{\mathrm{T}} + R(k),$$

$$P_{xy}(k|k-1) = P(k|k-1)C^{\mathrm{T}},$$

$$K(k) = P_{xy}(k|k-1)P_{yy}^{-1}(k|k-1), \tag{1}$$

$$\hat{\mathbf{x}}(k|k) = \hat{\mathbf{x}}(k|k-1) + K(k)[\mathbf{y}(k) - C\hat{\mathbf{x}}(k|k-1)],$$

$$P(k|k) = P(k|k-1) - K(k)P_{yy}(k|k-1)K^{\mathrm{T}}(k), \tag{2}$$

where K is the Kalman gain matrix, the carets indicate the mean (conditional expectations) of the corresponding density function, and P is the covariance matrix of the vector of estimation errors. Moreover, the notation $z(k|k-1)$ indicates the value of the quantity z at time k computed based on information available up to time $k-1$. Likewise, $z(k|k)$ indicates the value of z computed at time k using information available up to and including time k.

The first equation in (1) shows how the state at time $k-1$ is propagated to time k and the second equation shows how the propagated state maps onto the output. Similarly, the third equation shows how the estimation error vector covariance matrix propagates from time $k-1$ to time k. A key remark here is to notice that such propagations are made using the linear dynamical model available for the system represented by matrices (A, B, C).

On the other hand, the two equations in (2) show how the current values, that is, at time k, of the state vector and its covariance matrix can be updated after new information—contained in the measured value $\mathbf{y}(k)$—becomes available.

A. The nonlinear case

If the system is nonlinear and described by

$$\mathbf{x}(k) = f(\mathbf{x}(k-1), \mathbf{u}(k-1), \mathbf{w}(k-1)),$$

$$\mathbf{y}(k) = h(\mathbf{x}(k), \mathbf{r}(k)), \tag{3}$$

the best would be to propagate the entire probability density function and then take expectations. This is impractical, but the state vector of the system can be comfortably propagated using the model (3). Having proceeded thus, it is straightforward to use the first equation in (2) to update the state vector when the new measurement becomes available.

Unfortunately, the equations that describe the propagation of P, P_{xy}, and P_{yy}, together with the linear state propagation equations, as stated in Eq. (1), are valid only for linear systems. It is interesting to notice that the other equations remain valid even in the nonlinear case. The traditional approach to propagate the covariance matrices in the nonlinear case has been to linearize the model (3) at each step and then apply the Kalman filter equations. This results in the well-known extended Kalman filter

$$\hat{\mathbf{x}}(k|k-1) = f(\hat{\mathbf{x}}(k-1|k-1), \mathbf{u}(k-1)),$$

$$\hat{\mathbf{y}}(k|k-1) = h(\hat{\mathbf{x}}(k|k-1)),$$

$$P(k|k-1) = \frac{\partial f}{\partial x_i}P(k-1|k-1)\left(\frac{\partial f}{\partial x_i}\right)^T + Q(k),$$

$$P_{yy}(k|k-1) = \frac{\partial h}{\partial x_i}P(k|k-1)\left(\frac{\partial h}{\partial x_i}\right)^T + R(k),$$

$$P_{xy}(k|k-1) = P(k|k-1)\left(\frac{\partial h}{\partial x_i}\right)^T,$$

$$K(k) = P_{xy}(k|k-1)P_{yy}^{-1}(k|k-1), \tag{4}$$

$$\hat{\mathbf{x}}(k|k) = \hat{\mathbf{x}}(k|k-1) + K(k)[\mathbf{y}(k) - h(\hat{\mathbf{x}}(k|k-1))],$$

$$P(k|k) = P(k|k-1) - K(k)P_{yy}(k|k-1)K^T(k). \tag{5}$$

It is vital to realize that in the EKF the nonlinear model can and should be used to propagate the state and output vectors [first two equations in (4)] and to update the state vector. However, the remaining equations of the filter must be computed using linearizations that, depending on the application, are prone to a number of numerical problems.

B. The unscented Kalman filter

To see the key contribution of the UKF, consider the vectors of random variables $\mathbf{x}$ and $\mathbf{y}$ that are related by a nonlinear function as $\mathbf{y}=f(\mathbf{x})$. If a sufficiently large number of realizations of $\mathbf{x}$ and $\mathbf{y}$ are available, then the sample means $\bar{\mathbf{x}}$ and $\bar{\mathbf{y}}$ and covariances P, P_{yy}, and P_{xy} can be readily computed regardless of the function f.

Based on this remark, a solution to the problem in the case of nonlinear systems would be to actually propagate a large set of state vectors and subsequently to numerically compute means and covariances.[2]

The problem with this approach, however, is the great number of such vectors that would be required in order to

[1] It is important to emphasize that Q and R play the roles of lower bounds for P (and consequently for P_{xy}) and P_{yy}, respectively.

[2] In this case $\mathbf{x}$ would be state vectors prior to propagation, and $\mathbf{y}$ would be the corresponding images after propagation using the system dynamical model f.

have a reliable estimate of means and covariances. Moreover, because the (approximately) propagated probability density function will most likely not be Gaussian, other moments besides the mean and covariance would be necessary to fully specify it.

Such difficulties have been partially circumvented by the use of the unscented transformation (UT) [4]. The UT reduces the potentially large number of state vectors to a small representative group, which have been named sigma points. The main idea is to determine a small number of sigma points (actually vectors)—$\mathcal{X}_i$, $i=0,1,\ldots,2n_{\mathrm{a}}$, where n_{a} is the dimension of the augmented state vector—that by construction has the same mean and covariance as $\mathbf{x}$, that is,

$$\mathrm{mean}\{\mathcal{X}_i\} = \bar{\mathbf{x}} \quad \text{and} \quad \mathrm{cov}\{\mathcal{X}_i\} = P,$$

and then to propagate the sigma points using the model $\mathcal{Y}_i = f(\mathcal{X}_i)$. Subsequently, the mean and covariance of the new set of points $\mathcal{Y}_i$ should be numerically determined. The values thus computed become working estimates for the mean and covariance of the random vector $\mathbf{y}$, respectively, that is,

$$\mathrm{mean}\{\mathcal{Y}_i\} \approx \bar{\mathbf{y}} \quad \text{and} \quad \mathrm{cov}\{\mathcal{Y}_i\} \approx P_{yy}.$$

Recall that, in the case of the UKF, $n_{\mathrm{a}}=2n+m$ is the dimension of the augmented state vector which is composed by the concatenation of the original state, process, and measurement noise variables; thus

$$\mathbf{x}^{\mathrm{a}}(k) = [\mathbf{x}^{\mathrm{T}}(k)\mathbf{w}^{\mathrm{T}}(k)\mathbf{r}^{\mathrm{T}}(k)]^{\mathrm{T}}$$

and $\mathbf{x}^{\mathrm{a}}(k) \in \mathbb{R}^{2n+m}$.[3] Consequently, the covariance matrix of the vector of estimation errors must refer to this augmented state vector and henceforth it will be referred to as $P^{\mathrm{a}} \in \mathbb{R}^{(2n+m)\times(2n+m)}$, that is,

$$P^{\mathrm{a}} = \begin{bmatrix} P & 0 & 0 \\ 0 & Q & 0 \\ 0 & 0 & R \end{bmatrix}.$$

The sigma points can be chosen as

$$\mathcal{X}_0^{\mathrm{a}}(k-1|k-1) = \hat{\mathbf{x}}^{\mathrm{a}}(k-1|k-1),$$

$$\mathcal{X}_i^{\mathrm{a}}(k-1|k-1) = \hat{\mathbf{x}}^{\mathrm{a}}(k-1|k-1) + [\sqrt{(n_{\mathrm{a}}+\lambda)P^{\mathrm{a}}(k-1|k-1)}]_i,$$

$$\mathcal{X}_{i+n_{\mathrm{a}}}^{\mathrm{a}}(k-1|k-1) = \hat{\mathbf{x}}^{\mathrm{a}}(k-1|k-1) - [\sqrt{(n_{\mathrm{a}}+\lambda)P^{\mathrm{a}}(k-1|k-1)}]_i, \qquad (6)$$

with associated weights given by

$$w_0^{(m)} = \frac{\lambda}{n_{\mathrm{a}}+\lambda},$$

$$w_0^{(c)} = \frac{\lambda}{n_{\mathrm{a}}+\lambda} + 1 - \alpha^2 + \beta,$$

$$w_i^{(m)} = w_i^{(c)} = \frac{1}{2(n_{\mathrm{a}}+\lambda)}, \qquad (7)$$

where $i=1,\ldots,n_{\mathrm{a}}$ and $[\sqrt{(\cdot)}]_i$ is either the ith row or column of the matrix square root.[4] For the sake of simplicity the following choices are made: $\lambda=\alpha^2(\kappa+n_{\mathrm{a}})-n_{\mathrm{a}}=0$ [4,5] $\alpha=1$, $\kappa=0$, and $\beta=2$ [8]. Choosing the sigma points as indicated in (6) guarantees exact matching of the first two moments (mean and covariance) and, because the state distribution was assumed symmetrical, the third moment is also exactly matched, since it is zero. Other schemes to choose sigma points and the respective implications for the filter have been discussed in [4]. Therefore, the unscented Kalman filter equations can be expressed as [4,8]

$$\hat{\mathbf{x}}(k|k-1) = \sum_{i=0}^{2n_{\mathrm{a}}} w_i^{(m)}\mathcal{X}_i^x(k|k-1) \quad \text{where } \mathcal{X}_i^x(k|k-1) = f(\mathcal{X}_i^x(k-1|k-1),\mathbf{u}(k-1),\mathcal{X}_i^w(k-1|k-1)),$$

$$\hat{\mathbf{y}}(k|k-1) = \sum_{i=0}^{2n_{\mathrm{a}}} w_i^{(m)}\mathcal{Y}_i(k|k-1) \quad \text{where } \mathcal{Y}_i(k|k-1) = h(\mathcal{X}_i^x(k|k-1),\mathcal{X}_i^r(k-1|k-1)),$$

$$P(k|k-1) = \sum_{i=0}^{2n_{\mathrm{a}}} w_i^{(c)}[\mathcal{X}_i^x(k|k-1) - \hat{\mathbf{x}}_i(k|k-1)][\mathcal{X}_i^x(k|k-1) - \hat{\mathbf{x}}_i(k|k-1)]^{\mathrm{T}},$$

$$P_{yy}(k|k-1) = \sum_{i=0}^{2n_{\mathrm{a}}} w_i^{(c)}[\mathcal{Y}_i(k|k-1) - \hat{\mathbf{y}}_i(k|k-1)][\mathcal{Y}_i(k|k-1) - \hat{\mathbf{y}}_i(k|k-1)]^{\mathrm{T}},$$

$$P_{xy}(k|k-1) = \sum_{i=0}^{2n_{\mathrm{a}}} w_i^{(c)}[\mathcal{X}_i^x(k|k-1) - \hat{\mathbf{x}}_i(k|k-1)][\mathcal{Y}_i(k|k-1) - \hat{\mathbf{y}}_i(k|k-1)]^{\mathrm{T}},$$

$$K(k) = P_{xy}(k|k-1)P_{yy}^{-1}(k|k-1), \qquad (8)$$

with $i=0,\ldots,2n_{\mathrm{a}}$ and where $\mathcal{X}_i^a(k-1|k-1)$ are given by (6) and $\mathcal{X}^a=[(\mathcal{X}^x)^{\mathrm{T}}\ (\mathcal{X}^w)^{\mathrm{T}}\ (\mathcal{X}^r)^{\mathrm{T}}]^{\mathrm{T}}$. The updating equations are

$$\hat{\mathbf{x}}(k|k) = \hat{\mathbf{x}}(k|k-1) + K(k)[\mathbf{y}(k) - h(\hat{\mathbf{x}}(k|k-1))],$$

$$P(k|k) = P(k|k-1) - K(k)P_{yy}(k|k-1)K^{\mathrm{T}}(k). \qquad (9)$$

In closing this section, it is important to notice that the updating equations of the three filter algorithms are the same,

[3] In the particular case of purely additive process and measurement noise, both the state vector and the estimation error covariance matrix need not be augmented since an alternative form of the algorithm can be used which reduces computational costs and increases numerical robustness [8].

[4] If $B=A^{\mathrm{T}}A$, then the rows of A should be used. Conversely, if $B=AA^{\mathrm{T}}$, the columns of A should be used to compose the sigma points [4].

that is, Eqs. (2), (5), and (9). On the other hand, the state vector propagation is carried out using the linear model for the KF, using the Jacobian—that is, it is locally approximate only—for the EKF and using the *full nonlinear model* in the case of the UKF.[5] For details on Kalman filtering in general, the reader is referred to [7] and for the UKF see [4,8] and references therein. Recently, the UKF has been analyzed in the light of Bayesian theory [6] and it has been classified as a σ-point Kalman filter algorithm [9]. The latter expression refers to a group of nonlinear filters that make implicit the use of a deterministic sampling approach and a statistical linearization technique called weighted statistical linear regression so as to obtain the optimal terms used in (9).

C. Simultaneous state and parameter estimation

An important characteristic of the EKF and UKF algorithms is that they can be readily used to jointly estimate the system states and parameters. Once the algorithms are able to deal with nonlinear dynamical evolution, it is easy to consider a parameter as a "virtual" state that has to be estimated. Often it is expected that the system parameters do not vary or, if they do, the variation is much slower than that of the system state. This can be mathematically represented by rewriting Eq. (3) as

$$\mathbf{x}(k) = f(\mathbf{x}(k-1), \boldsymbol{\theta}(k-1), \mathbf{u}(k-1), \mathbf{w}(k-1)),$$

$$\boldsymbol{\theta}(k) = \boldsymbol{\theta}(k-1) + \mathbf{v}(k-1),$$

$$\mathbf{y}(k) = h(\mathbf{x}(k), \boldsymbol{\theta}(k), \mathbf{r}(k)), \tag{10}$$

where $\boldsymbol{\theta}(k) \in \mathbb{R}^{n_\theta}$ is the vector of parameters to be estimated with covariance matrix $P_\theta \in \mathbb{R}^{n_\theta \times n_\theta}$ and $\mathbf{v}(k) \in \mathbb{R}^{n_\theta}$ corresponds to a Gaussian, zero-mean, white noise term, with covariance matrix $Q_\theta \in \mathbb{R}^{n_\theta \times n_\theta}$, that accounts for the uncertainty in the estimated parameters.[6] Indeed, Eqs. (10) describe an extended system, whose *extended state vector* is $\mathbf{x}_{\text{ext}}(k) = [\mathbf{x}(k)^{\text{T}}\ \boldsymbol{\theta}(k)^{\text{T}}]^{\text{T}}$, $\mathbf{x}_{\text{ext}}(k) \in \mathbb{R}^{n+n_\theta}$.

In the UKF algorithm, in order to take into account the effect of process and measurement noise, together with uncertainty in the parameter estimates, by considering them as unknown perturbations to the nominal dynamical equations [5,8], the augmented extended state vector for the extended system (10) is

$$\mathbf{x}^{\text{a}}_{\text{ext}}(k) = [\mathbf{x}(k)^{\text{T}}\ \boldsymbol{\theta}(k)^{\text{T}}\ \mathbf{w}(k)^{\text{T}}\ \mathbf{v}(k)^{\text{T}}\ \mathbf{r}(k)^{\text{T}}]^{\text{T}},$$

and $\mathbf{x}^{\text{a}}_{\text{ext}}(k) \in \mathbb{R}^{n_{\text{a}}}$ where $n_{\text{a}} = 2n + 2n_\theta + m$.

It is important to note that, for jointly estimating states and parameters using Eqs. (10), the matrix P^{a} in Eq. (6) must be changed to $P^{\text{a}}_{\text{ext}}$; and the matrix P in Eqs. (8) and (9) must be changed to P_{ext}. These extended covariance matrices are defined as

$$P^{\text{a}}_{\text{ext}} = \begin{bmatrix} P & 0 & 0 & 0 & 0 \\ 0 & P_\theta & 0 & 0 & 0 \\ 0 & 0 & Q & 0 & 0 \\ 0 & 0 & 0 & Q_\theta & 0 \\ 0 & 0 & 0 & 0 & R \end{bmatrix}, \quad P_{\text{ext}} = \begin{bmatrix} P & 0 \\ 0 & P_\theta \end{bmatrix},$$

where $P \in \mathbb{R}^{n \times n}$ is the covariance matrix of the vector of state estimation errors, already defined in (1). This distinction is necessary so that the noise means and covariance matrices will not change during the iterations of the UKF algorithm, and only P_{ext} will evolve dynamically. This behavior is consistent with the assumption that the noise signals are independent from $\mathbf{x}(k)$, $\boldsymbol{\theta}(k)$, and $\mathbf{y}(k)$.

III. NONLINEAR MODEL BUILDING FROM DATA

A. The nonlinear representation

In this paper, the system will be modeled using a multivariable nonlinear autoregressive moving average (NARMA) model of the form [10]

$$\mathbf{y}(k) = f^{\ell}[\mathbf{y}(k-1), \mathbf{e}(k), \ldots, \mathbf{e}(k-n_e)], \tag{11}$$

where

$$\mathbf{y}(k) = \begin{bmatrix} y_1(k) \\ y_2(k) \\ \vdots \\ y_m(k) \end{bmatrix}, \quad \mathbf{e}(k) = \begin{bmatrix} e_1(k) \\ e_2(k) \\ \vdots \\ e_m(k) \end{bmatrix}.$$

In Eq. (11) n_e is the maximum lag considered for the noise terms, and $\mathbf{y}(k)$ is a vector of m measurements taken at time $t = kT_{\text{s}}$ obtained by sampling the continuous data $\mathbf{y}(t)$ with sampling time T_{s}. $\mathbf{e}(k)$ accounts for uncertainties, possible noise, unmodeled dynamics, and so on. For each signal, $f^{\ell}[\cdot]$ is some nonlinear function of $\mathbf{y}(k)$ and $\mathbf{e}(k)$. In this paper, $f^{\ell}[\cdot]$ is taken to be polynomial with maximum degree ℓ, but, of course, many other representations are available in the literature.

It is pointed out that the standard least-squares estimator cannot handle noise in the independent variables without incurring bias. The use of noise terms in addition to the extended least-squares estimator is effective in handling such a situation and avoiding bias. This problem has been recently reviewed in [3].

Equation (11) represents an autonomous multivariable system with m outputs and no inputs. In both examples to be discussed in the present work $m = 3$, but the extension for $m > 3$ is rather obvious. Therefore, the models take the following general form;

$$y_1(k) = \hat{f}^{\ell}_1(y_1(k-1), y_2(k-1), y_3(k-1), \xi_1(k-1), \xi_1(k-2), \ldots, \xi_1(k-n_{\xi_1})) + \xi_1(k),$$

[5] In fact, not the state vector but the sigma points are propagated in the case of the UKF.

[6] An estimate of Q_θ can be obtained as a by-product of the model-building process, as described in Sec. III. Different approaches, namely, fixed or time-varying Q_θ, can be used to tune the EKF or UKF convergence rate and parameters tracking performance [8]. For the sake of simplicity, in the present work Q_θ will be taken as a constant matrix.

$$y_2(k) = \hat{f}_2^{\ell}(y_1(k-1), y_2(k-1), y_3(k-1), \xi_2(k-1), \xi_2(k-2), \ldots, \xi_2(k-n_{\xi_2})) + \xi_2(k),$$

$$y_3(k) = \hat{f}_3^{\ell}(y_1(k-1), y_2(k-1), y_3(k-1), \xi_3(k-1), \xi_3(k-2), \ldots, \xi_3(k-n_{\xi_3})) + \xi_3(k), \qquad (12)$$

where $\xi_i(k)$, $i=1$, 2, 3, are the residuals, also known as the one-step-ahead prediction errors.

The model (12) clearly has three outputs. However, it is interesting to notice that for a given equation, the other outputs appear as exogenous inputs. For this reason, the modeling procedure can be dealt with as three multi-input single-output (MISO) problems that *together must compose an autonomous model* of the underlying dynamics. Consequently, in what follows the formula relates to the MISO case.

B. Parameter estimation

Having defined which representation to use, the identification of a model consists roughly of two steps, namely, structure selection and parameter estimation.

Any equation of (12) can be expanded into polynomials and, if taken along the available data, can be expressed in matrix form as $\mathbf{y}=\Psi\hat{\boldsymbol{\theta}}+\boldsymbol{\xi}$, where $\Psi \in \mathbb{R}^{N\times n_\theta}$ contains both independent variables, one-step-lagged dependent variable, and residuals; $\hat{\boldsymbol{\theta}} \in \mathbb{R}^{n_\theta}$, the caret indicates estimated values, $\mathbf{y}=\{y(k)\}_{k=1}^{N}$, and $\boldsymbol{\xi}=\{\xi(k)\}_{k=1}^{N}$ is the vector of residuals.

Having decided which regressors to use, the n_θ-dimensional parameter vector $\boldsymbol{\theta}$ can be estimated by minimizing the quadratic cost function $J_{\mathrm{ELS}}(\hat{\boldsymbol{\theta}})=\boldsymbol{\xi}^{\mathrm{T}}\boldsymbol{\xi}$ using orthogonal *extended* least squares [11], which effectively overcomes two major difficulties, namely, (i) numerical ill conditioning and (ii) structure selection. This amounts to selecting the columns of the regressor matrix Ψ. A criterion for structure selection that has proved helpful in many situations involving both real and simulated data is the *error reduction ratio* (ERR) [10]. This criterion is a welcome by-product of the procedure outlined above. Complementary procedures have also been proposed in the literature [12,13].

C. Structure selection

The regressors of model (12) may contain any combination of lagged outputs. In the present case, the number of such combinations is determined by the values of ℓ and n_{ξ_i}. Some kind of mechanism is called for in order to automatically choose the best n_θ regressors to compose the model. This problem is often referred to as model structure selection and must be judiciously accomplished regardless of the mathematical representation being used.

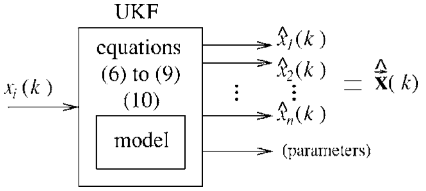

FIG. 1. Schematic representation of the aim of the UKF. A model is needed to provide f and h in Eqs. (8) and (9). Apart from the full state vector, it is possible to estimate parameters also, in which case Eq. (10) is required.

An important remark concerning structure selection is in order. In certain mathematical representations for which the base functions are of the same kind—for instance, radial basis function (RBF) models—or in the case of networks for which the activation functions could be the same, the problem of structure selection is basically one of deciding on the *size* of the model, that is, the "optimal" number of base functions or nodes in the hidden layer. This problem is somewhat easier to solve and has been addressed in different ways, such as the use of the description length for both RBF [14] and neural network models [15]; error reduction ratio, regularization combined with the predicted residual sum of squares statistic for RBF models [16], and multiobjective optimization for neural networks [17]. All such criteria are based on either statistics or information criteria but are not directly connected to the resulting dynamics. Recently, some guidelines have been suggested to use dynamical information (symmetry) in order to assist the model-building process for RBF and neural network models [18].

In the case of other model representations, as the one used in this paper, for which the basis functions are different, then it does not suffice to decide on the size of the model, but also, and primarily, it is necessary to decide *which type* of basis functions are required. When this is the case, it has been shown that a wrong type of basis functions (called term cluster) can have a very strong effect on the *dynamics* of the model, without necessarily affecting the *statistics* [12,19,20]. All the aforementioned representations have been compared on a set of real data in a recent paper [21].

D. Models for UKF implementation

It should be noticed that the dynamical model needed to implement the UKF will be used to perform two tasks, as seen in Eqs. (8) and (9). First, the model is used to predict the entire state vector from the single measured variable that drives the filter plus the other components of the state vector that must be totally estimated by the filter, as illustrated in Fig. 1.[7] In the second place, the model is used to propagate the sigma points as an approximate way to estimate how mean and covariance are affected by the system.

Having realized the different roles of the model within the filter, a number of remarks are in order.

From a *dynamical* point of view, the model should provide good predictions of the full state vector at time k, that is, $\hat{\mathbf{x}}(k)$, given the scalar signal that drives the filter $x_i(k)$ at time k and the estimates of the state variables provided by the filter at time $k-1$, that is, $\hat{x}_j(k-1)$, $j=1,\ldots,n$. Because the model does not need to produce free-run predictions, the

[7] In the examples investigated in this paper only one variable was used to drive the filter. This role can also be performed by any generic nonlinear function(s) of the state variable(s).

requirements on its dynamics are less exacting as, for instance, when free runs are used to reconstruct attractor properties. Therefore, it is unnecessary to require that the model dynamics be topologically equivalent to the original dynamics. Moreover the dynamical inaccuracy of the model is somewhat compensated by the dynamical information provided by the measured variable that drives the filter.

From a *static* point of view, the model is used to propagate the sigma points. In this respect the dynamical features are not important but the *nonlinearity* of the model should be adequate. A poor approximation of the nonlinearity would have the effect of mapping the sigma points incorrectly and this could eventually alter significantly the covariance matrix and, in turn, the filter stability. Therefore if the system cannot be correctly approximated by a given model representation, a different one should be tried.

IV. NUMERICAL EXAMPLES

A. The Lorenz system

This example will consider the well-known Lorenz system [22] which has also been studied in the context of state and joint parameter estimation in [5]

$$\dot{x} = -\lambda_1 x + \lambda_1 y,$$

$$\dot{y} = \lambda_2 x - y - xz,$$

$$\dot{z} = -\lambda_3 z + xy, \tag{13}$$

with $\lambda_1=12$, $\lambda_2=40$, and $\lambda_3=4$. The set of equations (13) was integrated with $\delta t=0.01$ using a fourth-order Runge-Kutta algorithm with initial condition $x(0)=y(0)=z(0)=0.1$. After integration, three independent and Gaussian distributed realizations of white noise were added to each state variable. The noise distributions were zero mean and had a standard deviation equal to 10% of the respective state variable. The resulting data are shown in Fig. 2(b). These data are the starting point of this example.

1. Global modeling

One thousand observations of each state variable (obtained by decimating the integrated data by a factor 2, to yield data sampled at $T_s=20$ ms) were used in the model-building stage. These data are shown in Fig. 2(b). The following model was obtained as indicated in Sec. III:

$$x(k) = +0.861\,921\,218\,252x(k-1) + 0.234\,404\,677\,244y(k-1) - 0.752\,478\,985\,861 \times 10^{-3}y(k-1)z(k-1) - 0.183\,251\,890\,379 \times 10^{-2}x(k-1)z(k-1),$$

$$y(k) = 1.159\,618\,833\,82y(k-1) - 0.572\,159\,400\,169 \times 10^{-2}y(k-1)z(k-1) - 0.142\,963\,657\,294 \times 10^{-1}x(k-1)z(k-1) + 0.577\,685\,873\,218x(k-1),$$

$$z(k) = +0.962\,354\,730\,753z(k-1) + 0.553\,656\,929\,609 \times 10^{-2}y^2(k-1) + 0.940\,412\,257\,934 \times 10^{-2}x(k-1)y(k-1) - 0.116\,546\,062\,702 \times 10^{-2}z^2(k-1) + 0.399\,604\,189\,148 \times 10^{-2}x^2(k-1), \tag{14}$$

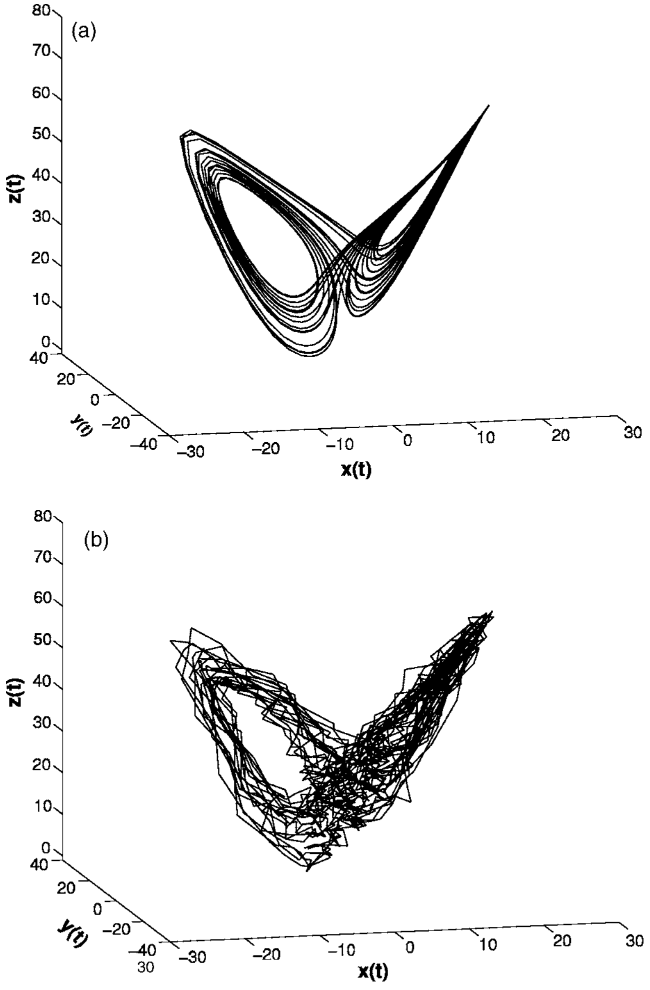

FIG. 2. Lorenz attractor obtained by numerical integration of Eq. (13) with $\lambda_1=12$, $\lambda_2=40$, and $\lambda_3=4$ and initial conditions $x(0)=y(0)=z(0)=0.1$. (a) Noise-free and (b) with 10% Gaussian white noise added to each component.

where only the deterministic part of the model is shown. The stochastic part, used during parameter estimation, in order to avoid bias is not shown nor used in the filter. The attractor to which Eq. (14) settles under free-run simulation is shown in Fig. 3(a).

2. State estimation

In the following, model (14) was used as part of the UKF [see Eqs. (8) and (9)] to estimate the three components of the Lorenz attractor. It should be noticed that in this example the state vector is composed thus $\mathbf{x}=[x(k)\ y(k)\ z(k)]^{\mathrm{T}}$. Model (14) was used as an approximation to the function f. Because only the state $x(k)$ was chosen to drive the filter, function h was defined as the vector $C=[1\ 0\ 0]$.

The estimation result is shown in Fig. 3(b). An important point to notice is that *only* the noisy sampled sequence $x(k)$ was presented to the UKF to reconstruct the three components shown in Fig. 3(b). Therefore the three components are only needed during the model-building phase. From then on,

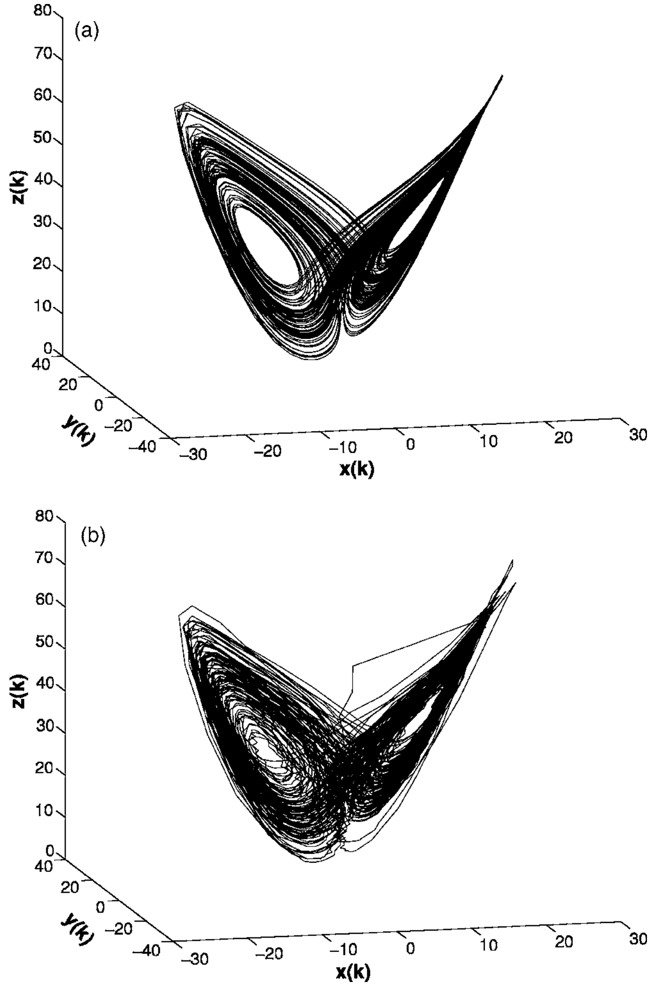

FIG. 3. Lorenz attractor obtained (a) by free-run simulation of Eq. (14) with initial conditions $x(0)=y(0)=z(0)=0.1$ [model (14) was obtained from the data shown in Fig. 2(b)] and (b) by the UKF with Eq. (14) from the single variable $x(k)$. The transient at the beginning was included in the plot to give an idea of how quickly the filter settles.

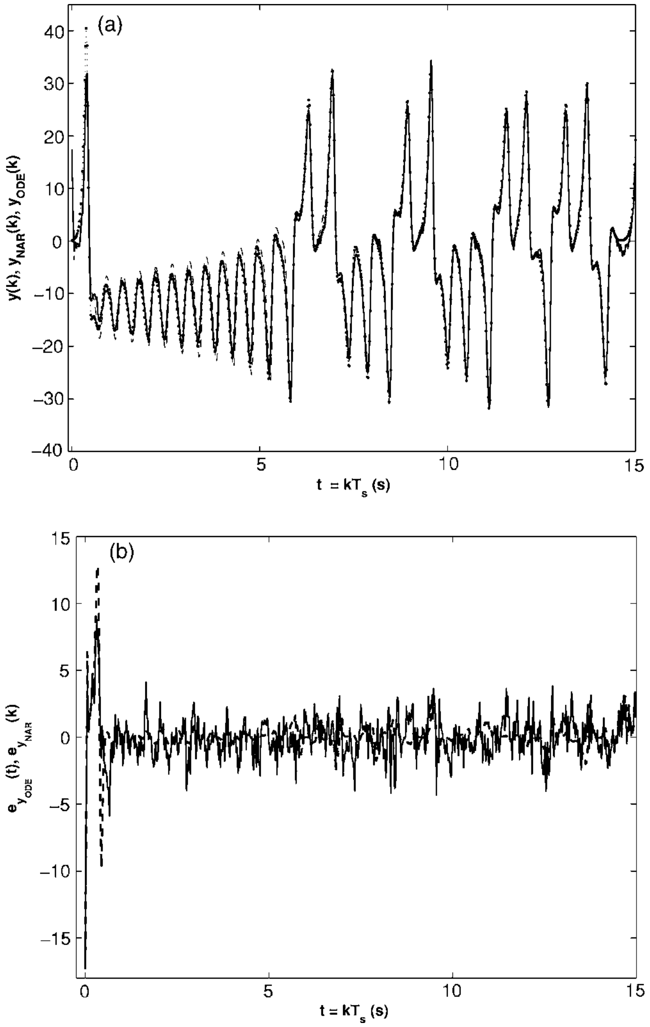

FIG. 4. (a) Window of data corresponding to the y component of the Lorenz attractor and (b) estimation error. In the plot (·) is the true (obtained by integration of the differential equations) value $y(t)$, $t=kT_s$, (—) and (- -) indicate, respectively, the state estimated by the UKF from the single variable $x(k)$ with Eq. (14), $\hat{y}_{\mathrm{NAR}}(k)$ [$e_{\mathrm{NAR}}(k)$ for plot (b)], and with Eq. (13), $\hat{y}_{\mathrm{ODE}}(k)$ [$e_{\mathrm{ODE}}(k)$ for plot (b)].

only one component is used to drive the filter, and consequently to estimate the full state vector.

Figure 4 shows the true state obtained by integration of the Lorenz equations (13), $y(t)$, $t=kT_s$, and the states $\hat{y}_{\mathrm{ODE}}(k)$ and $\hat{y}_{\mathrm{NAR}}(k)$ estimated by UKF using the theoretical (13) and identified (14) models, respectively.

Table I summarizes the UKF performance for various noise levels and for both additive and multiplicative noise. In the case of additive noise model (14) was used to propagate the sigma points. A counterpart model (not shown) was obtained from data with multiplicative noise. The table also includes the rms estimation error when the theoretical model (13) was used in the filter. It is vital to realize that the same identified model [see Eq. (14)] was used in all examples of additive noise and that the model obtained from data with multiplicative noise was always used in the examples that considered that type of noise. As stated before, such models were obtained from data sets composed of the three state components with 10% noise (additive and multiplicative). The higher noise levels indicated in the table correspond to noise added to the x component used to drive the filter, in the case of additive noise, or to noise included in the first differential equation in (13), in the case of multiplicative noise.

In view of Table I, it would be tempting to say that the UKF with the identified model is quite competitive especially at high noise levels. However, it should be noticed that in practice the theoretical ordinary differential equation (ODE) model will not be available in general and some type of identified model is required.

It is important to remark that, whenever the identified model (14) was used, or its counterpart in the case of multiplicative noise, the process noise covariance matrix was taken such that $Q \neq 0$. Otherwise, the small model imperfections added to the high sensitivity to initial conditions of chaotic systems led to bad tracking performance after the convergence of the filter. Hence Q was taken to be a diagonal matrix with elements equal to the variance of the one-step-

TABLE I. Normalized root mean square (rms) error of estimated states for the Lorenz system corrupted with different amounts of white Gaussian noise. Figures in romantype relate to additive noise whereas figures in italics relate to multiplicative noise. In every case the UKF was used. In the table, ODE indicates that the original model (13) was used to propagate the sigma points and NAR indicates that the identified model (14) was used instead. In every case, only the noise contaminated x component was used to drive the filter.

	Noise (%)		x(%)		y(%)		z(%)	
ODE	10	*10*	1.25	*0.11*	1.57	*0.95*	0.88	*0.70*
NAR	10	*10*	3.31	*0.26*	4.23	*1.94*	2.52	*1.84*
ODE	25	*25*	3.53	*0.28*	4.41	*2.40*	2.41	*1.91*
NAR	25	*25*	6.06	*1.08*	7.55	*3.38*	3.92	*2.77*
ODE	50	*50*	8.40	*1.16*	10.90	*4.59*	5.79	*3.73*
NAR	50	*50*	9.91	*2.95*	12.36	*5.93*	6.28	*4.29*

ahead prediction errors obtained by simulation of model (14). In a sense, matrix Q thus obtained quantifies model uncertainties.

Finally, it is interesting to notice that in Table I the estimation error of the z variable is significantly less than for x and y. Also, the errors related to such variables are rather similar being the error of x slightly smaller than the error related to y. This is in good agreement with the estimates of the (normalized) observability indices $\delta_z = 1.00$, $\delta_y = 0.37$ and $\delta_x = 0.30$ [27]. In fact, such indices indicate that y is slightly more observable than x, but clearly show that z is significantly more observable than the other two. An important remark is that it is well known that such indices have a local character, rather than global. This is consistent with the fact that the performace of the filter depends quite strongly on the "local" performance of the model.

B. An electronic oscillator

This example considers an actual implementation of an electronic oscillator known as Chua's circuit [23]. In the gyrator-based implementation used, the frequency of the main peak in the spectral power density is around 1.6 Hz [24]. This enabled collecting the data with a sampling time of $T_s = 30$ ms.

1. Global modeling

As for the case of the Lorenz system, only 1000 observations of each state variable were used to build the model, according to the procedure briefly reviewed in Sec. III. The data used to build the model are shown in Fig. 5. The automatic structure selection scheme followed by an orthogonal extended least-squares estimation routine yielded the following model:

$$
\begin{aligned}
x(k) = {} & \mathit{1.210\,362\,032\,06}x(k-1) + 0.972\,0471\,837\,33 \\
& \times 10^{+2}z(k-1) + 0.598\,221\,283\,675y(k-1) \\
& - 0.171\,315\,924\,858 \times 10^{-1}x^3(k-1) \\
& + 0.398\,0977\,036\,75 \times 10^{+5}y(k-1)z^2(k-1) \\
& + 0.819\,593\,354\,856 \times 10^{+4}x(k-1)z^2(k-1) \\
& - 0.111\,072\,563\,509 \times 10^{+3}y^2(k-1)z(k-1),
\end{aligned}
$$

$$
\begin{aligned}
y(k) = {} & \underline{\mathit{0.895\,252\,009\,866}}y(k-1) - 0.223\,303\,6811\,19 \\
& \times 10^{+6}z^3(k-1) + 0.083\,714\,349\,294\,7x(k-1) \\
& + 0.129\,325\,180\,333 \times 10^{+3}z(k-1) \\
& - 0.677\,214\,817\,432 \times 10^{-3}x^3(k-1) \\
& + 0.181\,085\,268\,783 \times 10^{-3}x^2(k-1) \\
& - 0.204\,050\,446\,812 \times 10^{+1}x(k-1)y(k-1)z(k-1),
\end{aligned}
$$

$$
\begin{aligned}
z(k) = {} & \mathit{0.938\,265\,0571\,86}z(k-1) - 0.637\,485\,684\,220 \\
& \times 10^{-3}y(k-1) - 0.362\,134\,020\,185 \times 10^{-4}x(k-1) \\
& + 0.315\,363\,093\,575 \times 10^{-5}x^2(k-1) \\
& + 0.632\,868\,345\,441 \times 10^{-5}y^2(k-1) \\
& - 0.643\,318\,682\,312 \times 10^{-5}x(k-1)y(k-1) \\
& + 0.187\,619\,321\,537 \times 10^{-2}x(k-1)z(k-1). \qquad (15)
\end{aligned}
$$

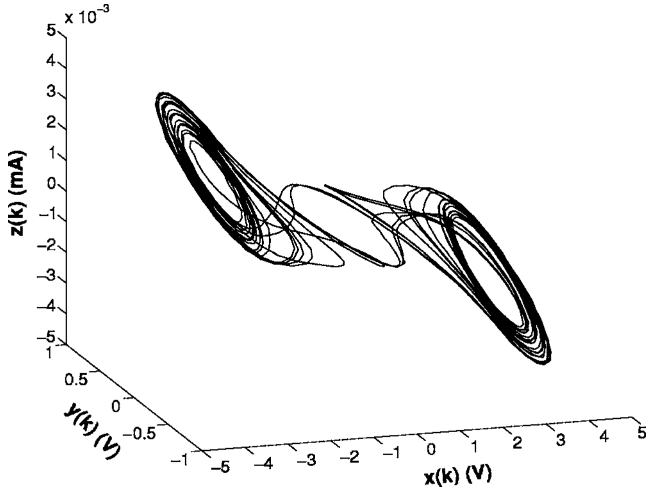

FIG. 5. The *measured* double scroll attractor of the implemented electronic oscillator. These are the data used to build model (15).

As before, only the deterministic part of the model is shown. It is important to notice that model (15) cannot produce a mathematically symmetrical attractor because it includes even parity terms [25]. When symmetry conditions were imposed during modeling, the new model did not settle into the double-scroll attractor and, for the reasons discussed in Sec. III D, state estimation results did not improve.

Besides the polynomial model (15) multilayer perceptron (MLP) neural networks were trained from the same set of data. Three networks were trained, one for each state variable. The input layer in each case was the same, namely, $[x(k-1), y(k-1), z(k-1)]^{\mathrm{T}}$. The activation functions of the nodes in the single hidden layer were hyperbolic tangent and the single output node was linear in all three networks. In this paper we discuss the results obtained by two set of net-

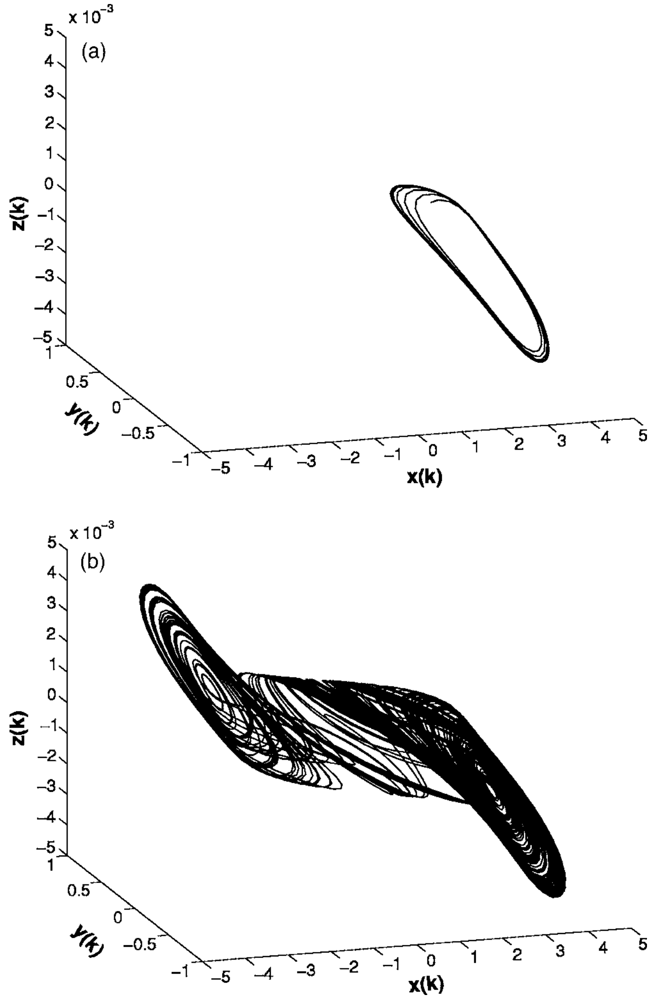

FIG. 6. Attractors produced by identified model (a) without and (b) with parameter perturbation, with initial conditions $x(0)=3.2$, $y(0)=0.37$ and -0.009.

works. The first set was composed of three fully connected networks with two nodes in the hidden layer and the second was composed of three fully connected networks with seven nodes in the hidden layer. Such neural models will be referred to as MLPs2 and MLPs7, respectively. Briefly, MLPs2 also settled to a limit cycle and MLPs7 showed a very long chaotic transient—consistent with the double-scroll attractor—before settling to a spiral strange attractor.

For the reasons discussed in Sec. III D, it was believed that all the estimated models—(15), MLPs2, and MLPs7—could be competitive in the implementation of the UKF. However, it was desired to verify if models with improved global dynamics would outperform locally optimal models. In order to investigate this issue slight changes were implemented in model (15) *a posteriori*, as described in the following paragraph.

The free run of model (15) settles to the attractor shown in Fig. 6(a). Such an attractor seems to be close to a genuine solution of the system in the sense that it could be the "stable version" of one of the unstable periodic orbits that compose the original attractor. When this is the case, it has been argued that the model can be perturbed (this can be done in different ways) in order to become chaotic [26]. In the present case this is achieved by slightly increasing the underlined parameter. For instance, if 0.924 274 41 is used, model (15) settles to the double-scroll attractor shown in Fig. 6(b).[8] In what follows we shall refer to this as the perturbed double-scroll model.

TABLE II. Normalized root mean square (rms) error of estimated states for the electronic system. The UKF was implemented with the identified model (15)—in several contexts (see text)—and with the neural models MLPs2 and MLPs7.

Situation investigated	x	y	z
(i) Model	$2.80\times10^{-7}\%$	2.58%	0.87%
(ii) Perturbed model	$3.75\times10^{-7}\%$	5.85%	2.86%
(iii) Joint parameter estimation	$3.70\times10^{-7}\%$	3.03%	1.24%
(iv) Model with $Q=0$	88.46%	85.46%	76.46%
(v) MLPs2	$2.57\times10^{-7}\%$	2.06%	1.18%
(vi) MLPs7	$9.33\times10^{-7}\%$	1.27%	0.86%

2. State estimation

The UKF with the identified model (15) was used to estimate the full state vector of the electronic circuit using only the x component to drive the filter. The following situations were tested: (i) using (15) as obtained from the modeling step; (ii) slightly perturbing the underlined parameter of (15) in order to approximate the resulting attractor to the original double scroll attractor; (iii) using (15) but jointly estimating the three parameters in italic (which correspond to the terms of each equation with higher ERR index; see Sec. III C); (iv) the same as in (i) but using a null process noise covariance matrix Q.

It is important to declare that in the aforementioned situations, except in (iv), Q was taken as a diagonal matrix with elements equal to the one-step-ahead prediction error variance of the respective model.[9]This is analogous to what was done in the example of the Lorenz system. In case (iii), where three parameters were also estimated, the elements of Q_θ were taken as the corresponding parameter variances obtained as a byproduct from the extended least squares estimator used during modeling step. The results are summarized in Table II.

The very low values of rms for the estimation errors of the x component are a consequence of low measurement noise in our experimental setup and due to the fact that the x variable is used to drive the filter.

An interesting remark that can be made based on the results of Table II is that although the perturbed model is, in a sense, closer to the original system (for instance, both have a positive Lyapunov exponent), the unperturbed model—

[8] Other parameter values were tested for perturbing model (15) in order to obtain the double-scroll attractor. Nevertheless, the greater the disturbance (i.e., the farther from the optimal set of parameters) the worse the performance of the UKF in estimating the states, although the model might display improved global features.

[9] Obtained by one-step-ahead simulation using each model.

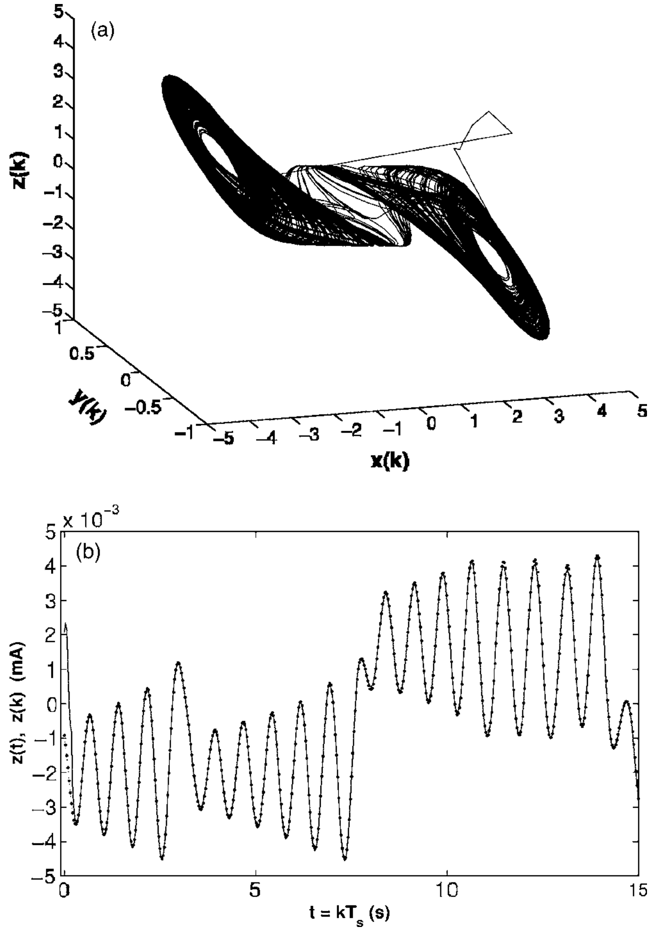

FIG. 7. The estimated double-scroll attractor of the implemented electronic oscillator. Estimation was performed by the UKF with model (15) and the x component driving the filter. (a) Full state vector; (b) detail of the z component (⋯) measured data, and (—) estimated component $\hat{z}(k)$. In the plot (a) the transient at the beginning was also included to give an idea of how quickly the filter settles.

which has a lower one-step-ahead prediction error—is better suited for the UKF.

When the UKF was implemented using (15) then, quite naturally, the UKF estimates also settled to the same attractor (possibly just a limit cycle) shown in Fig. 6(a). This explains the very high values of rms for situation (iv) shown in Table II. This problem can be circumvented by setting a lower bound for the covariance matrix P by properly setting the process noise covariance matrix Q. Otherwise, when convergence was achieved, the trace of matrix P would tend to zero. In so doing the filter remains more active and the most recent measurements gain more weight compared to the model and as a result the UKF settles to a double-scroll attractor [see Fig. 7(a)]. In other words, there are two important sources of information in the filter: on the one hand, there is the model that propagates the sigma points and, on the other hand, there are the measurements. Suitably defining Q and R is a way of weighting differently these two sources. So as to express lack of confidence in the model, the trace of Q should be increased. Similarly, the trace of R ought to be directly related to the degree of uncertainty in the measurements.

When parameters are estimated in addition to the states, the estimated parameters converge to the values obtained during the modeling stage as indicated in Fig. 8. Consequently situations (i) and (iii) have close performance indices in Table II.

Table III summarizes the UKF performance when extra white Gaussian noise is added to the x component in the case of polynomial and neural models. As can be seen, the UKF implemented with the neural model with seven hidden nodes is significantly more robust to noise than the filters implemented with the other models. At first sight this could seem inconsistent with the results shown in Table II where MLPs7 has the highest error for the x variable. It should be noticed however that this weakness of MLPs7 does not appear in Table III because the filters are driven by the x variable, that is, poor performance in predicting the driving signal is not as serious as poor performance in predicting the remainder of the state vector, in which case the predictions are all we have.

V. DISCUSSION AND CONCLUSIONS

This paper has addressed state estimation for nonlinear systems. Two examples were considered: the Lorenz system and an electronic oscillator. The former was simulated and the latter used data actually recorded from the implemented circuit. These two systems were also studied recently in the context of UKF state and joint parameter estimation [5,6].

The procedure followed in this paper is innovative in two ways. First, the UKF was implemented using discrete models. Consequently no numerical integration is required in the propagation stage of the filter. Second, the used models were built from data only. Therefore, in each example the starting point was simply and only three time series $\{x(k),y(k),z(k)\}_1^{1000}$. For the Lorenz example, such data were contaminated with 10% noise both additive and multiplicative. In the case of the actual oscillator, such data were measured and digitalized using a 12-bit analog-to-digital converter. In both examples considered in this paper, in the state estimation stage, only one variable $[x(k)]$ was used to drive the filter. Therefore the situation investigated in this paper corresponds to a context in which only a short test is needed to measure the complete state vector. From the data collected during such a test the model is built and from then on only one variable needs to be recorded in order to estimate the complete state vector. This also explains why noise levels different from the modeling data were investigated in the problem of state estimation.

As mentioned before, the corresponding filters were driven by the $x(k)$ component only. Attempts were made using components $y(k)$ and $z(k)$ to drive the filter. In the case of the Lorenz system, driving the filter with y is also fine. However, due to symmetry, driving the system with z yields estimates of x and y that can turn out to be the opposite of the actual variables. This is also true when the filter is driven by the y variable for the electronic oscillator. This is due to the fact that observing such systems from such variables results

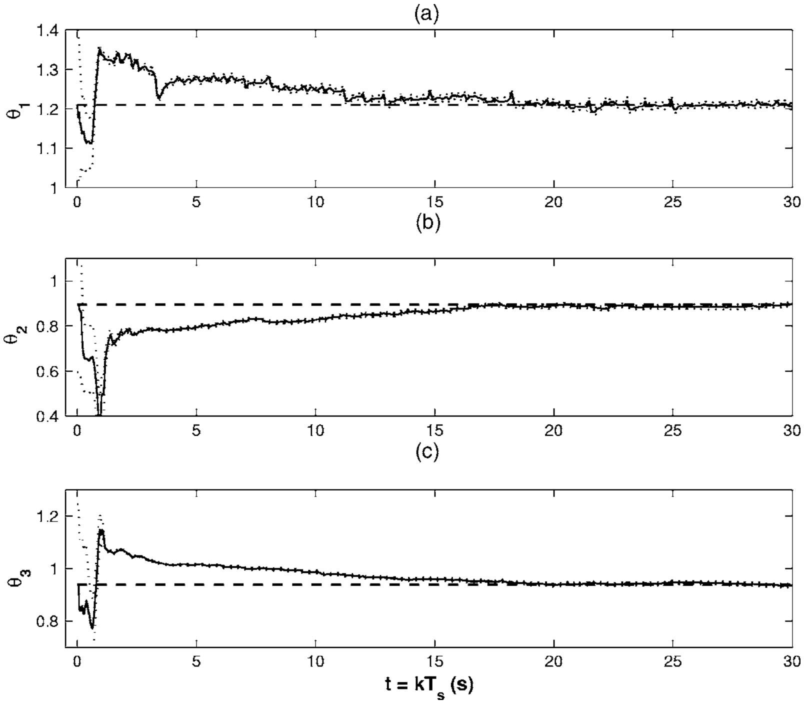

FIG. 8. The solid lines (—) show the results obtained from the joint estimation of the three parameters in italics in Eq. (15). The dotted lines (·) indicate plus and minus one standard deviation ($\pm\sqrt{\text{diag}\{P_\theta\}}$) of the parameters estimated by joint UKF. The dashed lines (- -) indicate the values for these parameters estimated by extended least squares during the modeling step.

in the superposition of fixed points [25]. Namely, from the z variable it is only possible to distinguish two fixed points of the Lorenz system and observing the double-scroll attractor from the y variable only one fixed point can be distinguished. Finally, driving the filter with the z variable, in the case of the electronic oscillator, no convergence was attained. The specific reasons for this remain somewhat unclear.

For the electronic oscillator not only state estimation but also joint state and parameter estimation were performed. It was noticed that the parameter values estimated by the filter were very close to those estimated by the extended least squares estimator, thus confirming a fact known from theory. Therefore, in both cases, the obtained parameters led to limit cycle attractors.

Although the model quality does have a direct bearing on the filter performance, it was noticed that setting a proper process noise covariance matrix Q was very important. For instance, when a model that settled to a limit cycle rather than to a chaotic was used in the filter, the filter too converged to a limit cycle, even when the driving signal was chaotic, if an unsuitable process noise covariance matrix was used. In fact, underestimating the trace of Q indicates to the filter that we are greatly confident in the model and consequently the filter settles to the "model-dictated" limit cycle. On the other hand, if a lower bound is imposed on the covariance matrix P (through setting a proper non-null process noise covariance matrix)—indicating some lack of confidence in the model—the filter settles to a chaotic attractor that very closely resembles the original one.

TABLE III. Normalized root mean square (rms) error of estimated states for the electronic oscillator. The UKF was implemented with the identified model (15) and with the neural models MLPs2 and MLPs7. The x component, which drives the filter, was corrupted with extra additive white Gaussian noise.

Model	Noise	x(%)	y(%)	z(%)
Polyn	10%	1.97	3.37	1.42
MLPs2	10%	3.48	5.63	2.55
MLPs7	10%	1.01	2.88	1.37
Polyn	25%	4.35	6.08	3.08
MLPs2	25%	8.20	12.66	5.54
MLPs7	25%	1.57	3.53	1.84
Polyn	50%	17.82	17.93	12.84
MLPs2	50%	25.56	27.20	18.13
MLPs7	50%	2.95	5.09	2.75

Several attempts were made using poor models and even linear models. As expected, the filter was unable to correctly estimate the states. This scenario did not change when rather high values of the lower bound for the covariance matrix were used.[10] Therefore it becomes clear that it is impractical to try to compensate for a very poor model by means of the covariance matrix.[11] On the other hand, it was noticed that to a certain extent uncertainties and imperfections in the model can be acommodated by the filter by setting a lower bound for the covariance matrix P (directly or through the previous definition of Q). An important point to make is that in a

[10] An interesting exception to this happened in the case of the electronic oscillator. In this case, it is known from the theory [24] that the second and third equations of the circuit are purely linear. Therefore when the filter was driven by the x variable, a purely linear model could be used with very good results. In this case, the rms errors associated with x, y, and z were, respectively, 3.04×10^{-7}%, 4.87%, and 0.81% (compare with Table II).

[11] There is an interesting interplay between the two sources of information, namely, the model and the measured variable(s). A decrease in the quality of one source should be somewhat compensated for with an increase in the quality of the other.

scenario of high noise the filter will naturally place a greater weight on the model. Therefore a more detailed study on the requirements on the model and its role in the implementation of the UKF should be carried out.

The former discussion seems to suggest that it would be nice to have not only dynamical models for nonlinear systems but also some measure of uncertainty which could be used to define a lower bound for the covariance matrix in the filter. In this paper the process noise covariance matrix was taken to be a diagonal matrix with elements equal to the variance of the one-step-ahead prediction errors obtained by simulation of the model which was used later in the state estimation stage. Also, regardless of the model, the covariance matrix plays a very critical role in the filter performance and guidelines to choose initial and limit values for this matrix should be devised.

The use of discrete models built from data in connection with the unscented Kalman filter to estimate states and parameters of nonlinear systems is viable. The paper considered the case in which the full state vector was available to build the model but only one variable was measured to drive the filter. Uncertainties and imperfections in the model can be compensated for, to some extent, by setting a suitable non-null process noise covariance matrix. Poor models usually result in poor filter performance or even in lack of convergence, regardles of the covariance matrix lower bound.

ACKNOWLEDGMENTS

The authors are grateful to CNPq and CAPES for financial support.

[1] H. W. Sorenson, IEEE Spectrum **7**, 63 (1970).
[2] R. E. Kalman, J. Basic Eng. **82**, 35 (1960).
[3] H. U. Voss, J. Timmer, and J. Kurths, Int. J. Bifurcation Chaos Appl. Sci. Eng. **14**, 1905 (2004).
[4] S. J. Julier and J. K. Uhlmann, Proc. IEEE **92**, 401 (2004).
[5] A. Sitz, U. Schwarz, J. Kurths, and H. U. Voss, Phys. Rev. E **66**, 016210 (2002).
[6] A. Sitz, U. Schwarz, and J. Kurths, Int. J. Bifurcation Chaos Appl. Sci. Eng. **14**, 2093 (2004).
[7] P. S. Maybeck, *Stochastic Models, Estimation and Control* (Academic Press, San Diego, 1979), Vol. 1.
[8] S. Haykin, *Kalman Filtering and Neural Networks* (Wiley Publishing, New York, 2001).
[9] R. van der Merwe, Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models, Ph.D. Thesis, OGI School of Science & Engineering, Oregon Health & Science University, April 2004. Available at: http://www.cse.ogi.edu/~rudmerwe/pubs/pdf/rvdmerwe_phd_thesis.pdf
[10] S. A. Billings, S. Chen, and M. J. Korenberg, Int. J. Control **49**, 2157 (1989).
[11] S. Chen, S. A. Billings, and W. Luo, Int. J. Control **50**, 1873 (1989).
[12] L. A. Aguirre and S. A. Billings, Int. J. Control **62**, 569 (1995).
[13] S. Lu, K. H. Ju, and K. H. Chon, IEEE Trans. Biomed. Eng. **48**, 1116 (2001).
[14] K. Judd and A. Mees, Physica D **120**, 273 (1998).
[15] M. Small and C. K. Tse, Phys. Rev. E **66**, 066701 (2002).
[16] S. Chen, X. Hong, C. J. Harris, and P. M. Sharkey, IEEE Trans. Syst. Sci. Cybern. **34**, 898 (2004).
[17] R. A. Teixeira, A. P. Braga, R. C. H. Takahashi, and R. R. Saldanha, Neurocomputing **35**, 189 (2000).
[18] L. A. Aguirre, R. A. M. Lopes, G. Amaral, and C. Letellier, Phys. Rev. E **69**, 026701 (2004).
[19] L. A. Aguirre and S. A. Billings, Physica D **80**, 26 (1995).
[20] E. M. A. M. Mendes and S. A. Billings, Int. J. Bifurcation Chaos Appl. Sci. Eng. **7**, 2593 (1997).
[21] U. Parlitz *et al.*, Chaos **14**, 420 (2004).
[22] E. Lorenz, J. Atmos. Sci. **20**, 130 (1963).
[23] T. Matsumoto, IEEE Trans. Circuits Syst. **31**, 1055 (1984).
[24] L. A. B. Tôrres and L. A. Aguirre, Electron. Lett. **36**, 1915 (2000).
[25] C. Letellier, G. Gouesbet, and N. Rulkov, Int. J. Bifurcation Chaos Appl. Sci. Eng. **6**, 2531 (1996).
[26] L. A. Aguirre, J. Maquet, and C. Letellier, Int. J. Bifurcation Chaos Appl. Sci. Eng. **12**, 135 (2002).
[27] C. Letellier and L. A. Aguirre, Chaos **12**, 549 (2002).