Parameter estimation via artificial data generation with the "two-stage" approach

Simone Garatti and Sergio Bittanti

***Abstract*— In this paper, we consider one of the most classical estimation problem, that of identifying an unknown parameter in a given model from measurements of input/output data. We present a new method named the *two-stage* approach which provides efficient estimates. The method is based on the preliminary generation of artificial data, and it is fully non-Bayesian. In this way, it is possible to avoid the well known difficulties encountered when resorting to Kalman filtering techniques in parameter estimation.**

I. Introduction

Consider the problem of estimating the parameters of a given plant from observed data, [2], [3], [5], [11], [15]. To be precise, suppose that data are generated by a dynamical system (continuous time or discrete time, linear or nonlinear, noise free or subject to disturbances) depending on a certain parameter vector $\theta \in \mathbb{R}^q$. The system is denoted by $P(\theta)$ as in Figure 1. We suppose that, while a mathematical model

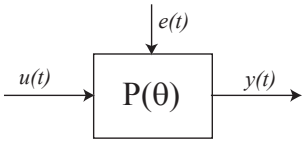

Fig. 1. The data generating system.

for $P(\theta)$ is available, the current value of parameter θ is unknown (white-box identification, [3]). For simplicity, in the sequel we will focus on discrete time models only.

In order to estimate θ from data, we assume that the system behavior is observed for a certain time interval over which a number N of input and output observations $\bar{D}^N = \{\bar{y}(1), \bar{u}(1), \ldots, \bar{y}(N), \bar{u}(N)\}$ are collected. The issue is then how to exploit the information contained in the data in order to obtain a fair estimate of the uncertain parameter θ.

In the situation described above, a common approach consists in resorting to Extended Kalman Filter (EKF) or to its Unscented (UKF) version, [1], [5], [6], [7], [8], [9], [12], [14], [17], [18], [19]. In these KF methods, parameter θ is seen as a state variable by introducing an additional state equation of the type: $\theta(k+1) = \theta(k)$, or, better, of the type: $\theta(k+1) = \theta(k) + w(k)$, where $w(k)$ is white noise with zero mean value and variance matrix W, introduced for increasing the reactivity of the algorithm. In this way, the estimation problem is reformulated as a state prediction problem. As is well known, even if $P(\theta)$ is described by a linear model, this is a nonlinear estimation problem due to the introduction of the additional state equation.

Both EKF and UKF are based on the application of the classical Bayes linear estimator formula. The difference relies in the updating rule of the filter gain. Such a gain is computed on the basis of the output variance matrix and of output-state the cross-variance by means of the Bayes estimator formula. In EKF these ingredients are (recursively) evaluated via system linearization by computing the Jacobians of the state and output equations around the current state estimate. In UKF, instead, the variance matrices are evaluated by means of their sample counterparts. These counterparts are computed by letting the system equations evolve for one step starting from a cloud of a few points suitably generated around the current state estimate. In this way no system linearization is required and the estimated variance matrices suffer from a lower degree of approximation.

Apart from the way in which the variance matrices are computed, both EKF and UKF are based on the same rationale, that is the iterated re-computation of the linear Bayes estimator around the current state estimate. This introduces a sort of adaptation of the filters which should hopefully cope with the system nonlinearities so as to guarantee the convergence of the state estimate to the true state value.

The question of convergence, however, is a well known critical issue of EKF. Indeed, there are celebrated (yet simple) examples showing the possible divergence/nonconvergence of EKF depending on the initialization of the filter (see e.g. [10]). Despite a number of results on local convergence ([4], [10], [13], [16]), the global convergence of EKF is still an open problem. As for UKF, to the best knowledge of the authors, the analysis of convergence is not yet so developed due to its relatively recent introduction.

One point raised in this paper is that both EKF and UKF may be poorly performing for plant parameter estimation, especially when the system dynamics is subject to substantial variations depending on the value of θ in the feasible region. Indeed, in such conditions, there is no guarantee that the estimate returned by the EKF and UKF converges to the correct parameter vector. Furthermore, the returned estimate turns out to be extremely sensitive to the algorithm initialization (in particular to the tuning of the covariance matrix of θ). This is a consequence of the fact that a parameter estimation problem is forced into a

Paper supported by the MIUR national project "Identification and adaptive control of industrial systems" and by CNR - IEEIT

S. Bittanti and S. Garatti are with the Dipartimento di Elettronica ed Informazione, Politecnico di Milano, p.zza L. da Vinci 32, 20133 Milano, Italy. E-mail: {bittanti,sgaratti}@elet.polimi.it

Bayesian setting.
In this paper, we propose a new estimation method, named the *two-stage* approach, which is fully developed in a non-Bayesian framework. With such method, good estimates of the uncertain parameter are obtained in a wide variety of situations. Its basic rationale is to reconstruct off-line the relationship linking the data to parameter θ through simulation trials via an intermediary step leading to a set of *artificial data*. Thus, the procedure develops in two phases: the first one transfers the information contained in the original data into the artificial data, while the second one enables establishing the link between these last data and parameter θ. The two-stage method does not suffer from any convergence issues and its range of applicability looks much wider than that of other approaches.

The paper is organized as follows. EKF and UKF are discussed in Section II, where by means of simple examples we spot out their flaws. Our new non-Bayesian approach is introduced in Section III, and then tested in Section IV by simulation.

II. A DISCUSSION ON EKF AND UKF

In this section, the drawbacks of EKF and UKF are spotted out by means of two simple examples.

Example 1. Consider the problem of estimating an unknown parameter θ from the observation $d = \arctan(3\theta)$. This problem can be trivially solved by inversion and does not require implementing neither EKF nor UKF. However, the link between data and the unknown is so elementary that the behavior of EKF and UKF is well explicated in this example.
Rewrite the problem as a state estimation problem according to the usual rationale of seeing the parameter as a state variable:

$$\begin{aligned} x(k+1) &= x(k) \\ x(0) &= \theta \\ y(k) &= \arctan(3x(k)). \end{aligned}$$

Estimating θ from d is equivalent to estimate $x(k+1)$ given output measurements up to time k.
The EKF update equations (see e.g. [14]) of the state estimate in this case are as follows where the 1-step ahead prediction of $x(k)$ is denoted by $\widehat{x}(k)$:

$$\begin{aligned} \widehat{x}(k+1) &= \widehat{x}(k) + \frac{1+\left(3\widehat{x}(k)\right)^2}{3}\left(y(k) - \arctan\left(3\widehat{x}(k)\right)\right) \\ \widehat{x}(0) &= \overline{x}. \end{aligned} \qquad (1)$$

Here $\overline{x}$ is the mean of the initial state, namely, the initial guess for the value of θ. Note that, due to the peculiarity of the example, the filter gain does not depend on the variance $P(k)$ of the estimation error at time k, so that no Riccati equation for the update of $P(k)$ is needed for implementing the EKF algorithm.
We will now study the evolution of EKF by a graphical representation. In Figure 2 the function $y = \arctan(3x)$ is plotted. For a certain value of d, the corresponding parameter θ is obviously the abscissa of the function for $y = d$. The sequence of estimates obtained by applying (1) is easily interpreted. Indeed, by noting that $\frac{3}{1+(3x)^2}$ is the derivative of $\arctan(3x)$, it is easy to recognize that $\widehat{x}(k+1)$ is nothing but the projection of the observed datum $y(k)$ on the tangent of $\arctan(3x)$ at the previous estimate value $\widehat{x}(k)$ (linearization around the previous estimate). As apparent

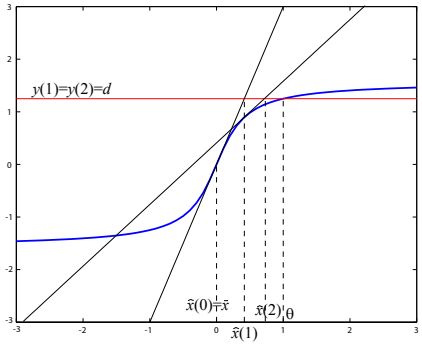

Fig. 2. EKF behavior.

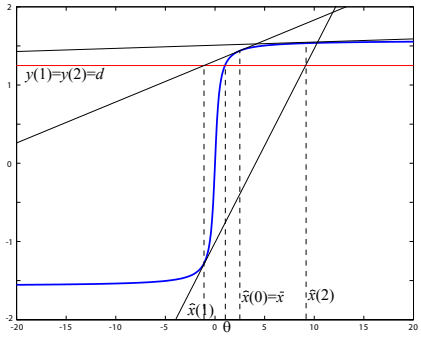

Fig. 3. EKF behavior.

from Figure 2 the sequence of estimates starting from $\overline{x} = 0$ converges to the actual value of parameter $\theta = 1$.
The adaptation introduced by the re-computation of the linearization at each step is fundamental for the convergence of the algorithm. However, even in such a simple example, convergence is not always guaranteed. For instance, the sequence of estimates diverges when the initial guess is $\overline{x} = 2$. See Figure 3.
Turn now to UKF. The estimate equations in this case are too complex to be reported here; in particular, note that when resorting to UKF a Riccati equation is necessary even in this simple example (the interested reader is referred again to [14]). Yet, the interpretation of UKF is again simple and intuitive.
Similarly to EKF, the estimate is a projection of the current observed output $y(k) = d$. However, the approximating straight lines to be considered are no more the tangents. Rather, they are obtained by interpolating $\arctan(3x)$ over an interval centered around $\widehat{x}(k)$ and whose size is determined

on the basis of the value taken by the estimation error variance $P(k)$ at the previous step, obtained by the auxiliary Riccati equation. Figure 4 graphically shows the first step

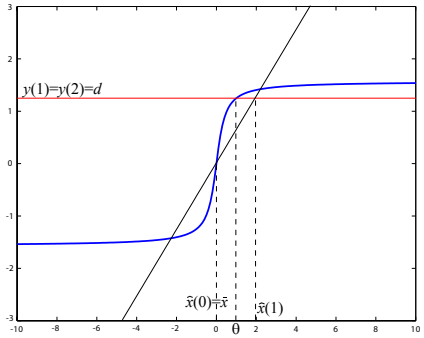

Fig. 4. EKF behavior.

of the UKF algorithm when $\theta = 1$, $\overline{x} = 0$, and the Riccati equation is initialized with $P(0) = \overline{P} = 5$.
From these considerations, it is apparent that an adaptation mechanism is also present in UKF, concerning the full position of the interpolating line. Also in this case, however, there is no guarantee that such adaptation leads to convergent estimates. Indeed, the case of Figure 5 ($\theta = 1$, $\overline{x} = 2$, $\overline{P} = 5$)

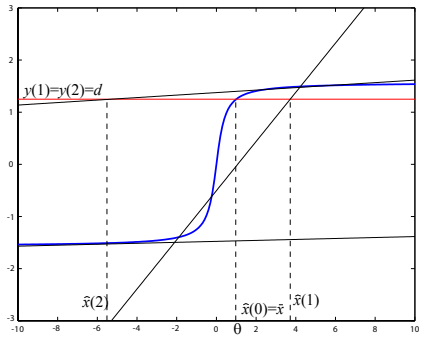

Fig. 5. EKF behavior.

leads to divergence.

Example 2. We considered the following data-generation mechanism:

$$\begin{aligned} x_1(k+1) &= a \cdot x_1(k) + v_{11}(k) && \text{(2a)} \\ x_2(k+1) &= x_1(k) + a^2 \cdot x_2(k) + v_{12}(k) && \text{(2b)} \\ y(k) &= a \cdot x_1(k) + x_2(k) + v_2(k), && \text{(2c)} \end{aligned}$$

where a is an unknown real parameter in the range $[-0.9, 0.9]$ and $v_{11} \sim WGN(0,1)$, $v_{12} \sim WGN(0,1)$, and $v_2 \sim WGN(0, 0.01)$ (WGN = White Gaussian Noise) are mutually uncorrelated noise signals. The objective was to retrieve the value of parameter a from $N = 1000$ observations of the output variable y.
In order to apply both EKF and UKF, system (2) was rewritten as:

$$\begin{aligned} x_1(k+1) &= x_3(k) \cdot x_1(k) + v_{11}(k) \\ x_2(k+1) &= x_1(k) + x_3(k)^2 \cdot x_2(k) + v_{12}(k) \\ x_3(k+1) &= x_3(k) + w(k) \\ y(k) &= x_3(k) \cdot x_1(k) + x_2(k) + v_2(k), \end{aligned}$$

where x_3 is an additional state variable representing parameter a. Herein, we will report the simulation results obtained by taking as $w(k)$ a $WGN(0, 10^{-6})$.
In all our experiments, system (2) was initialized with $x_1(0) = 0 = x_2(0)$. We then extracted 1000 values for the parameter a uniformly in the interval $[-0.9, 0.9]$. For each extracted value, a was estimated as the 1-step ahead prediction of x_3 when 1000 values of the output y were observed, i.e. $\widehat{a} = \widehat{x}_3(1001|1000)$. In such a computation, both EKF

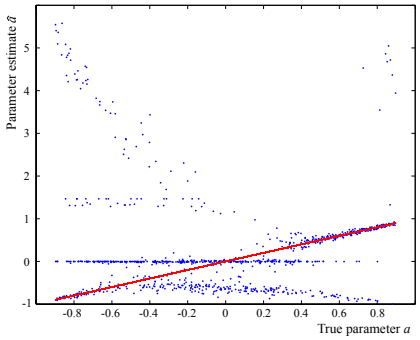

Fig. 6. Estimates of a via EKF (large initial variance).

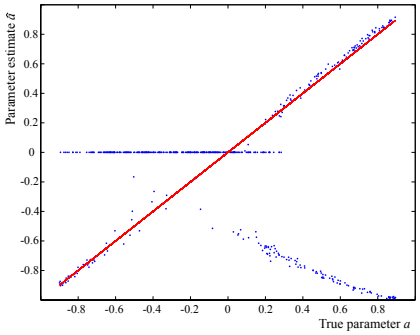

Fig. 7. Estimates of a via UKF (large initial variance).

and UKF were applied (the reader is referred to the literature for the update equations of EKF and UKF, see [1], [5], [6], [7], [9], [14]).
Figures 6-9 display the result obtained in different operating conditions by plotting the estimates versus the parameter actual values. In other words, for each point in the figure, the x-coordinate is the extracted value for a, while the y-coordinate is the corresponding estimate $\widehat{a}$ supplied by the used filter. Clearly a good estimator should return points concentrating around the bisector of the first and third quadrant.
As for the filter operating conditions, Figures 6 and 7 depict the results obtained when EKF and UKF, respectively, were

used with the following initialization: $\widehat{x}_1(0) = \widehat{x}_2(0) = 1$, $\widehat{x}_3(0) = -0.4$, and

$$P(0) = \begin{bmatrix} 10 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 2 \end{bmatrix} \tag{3}$$

($P(0)$ is the initial covariance of the estimation error).
Figures 8 and 9, instead, display the results obtained when

$$P(0) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 10^{-2} \end{bmatrix}. \tag{4}$$

As it appears, the behavior of EKF and UKF is quite

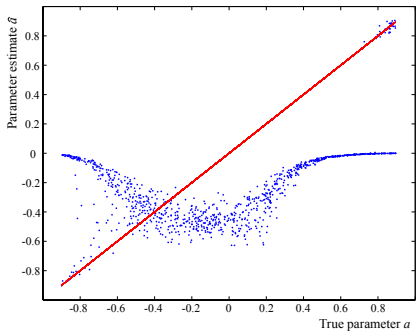

Fig. 8. Estimates of a via EKF (small initial variance).

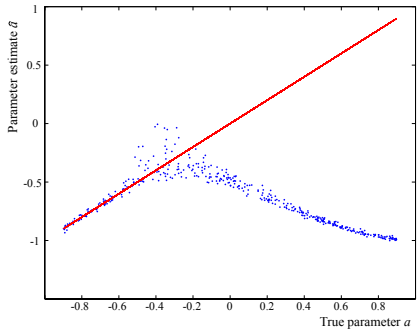

Fig. 9. Estimates of a via UKF (small initial variance).

different from the optimal expected one. Indeed, in many instances the estimate does not converge to the true value of parameter a.
As it can be seen from Figures 6 and 7, when $P(0)$ is "large" as in (3) (i.e. parameter a is assumed to be highly uncertain), $\widehat{a}$ can even converge to many different values, depending on the current realization of disturbances v_{11}, v_{12}, v_2. This was already pointed out in [10] for EKF. Somewhat more surprising is the fact that UKF suffers from the same drawback too.
When $P(0)$ is "small" as in (4), the estimates obtained via EKF (Figure 8) are still rather spread whereas UKF provides estimates which are more concentrated in a small region. However, as it is apparent from Figure 9 the estimation error can be quite large for positive values of a. As a matter of fact, selecting a "small" variance matrix means that the parameter a is assumed to be located near the initial guess with which the filter is initialized. As a result, the filters have an acceptable behavior only when the true parameter a is in a relatively small neighborhood of -0.4. When instead a is far from -0.4, neither EKF nor UKF guarantee the convergence of the estimate to the actual value of a.
Perhaps it is worth noticing that further simulations were performed by changing the initialization of $\widehat{x}_3(0)$ (precisely, to -0.8, 0.4, and 0.8), but such simulations are not reported here due to space limitations. The results, however, were similar to those previously presented, and the conclusions drawn above remain still valid.

As it appears,S both EKF and UKF suffer from a major drawback, the non-convergence of the estimate to the true parameter value. Non-convergence occurs especially when the parameter to be estimated is subject to a large uncertainty and is away from its initial guess (a condition, unfortunately, often encountered in application problems).
Furthermore, the behavior of EKF and UKF strongly depends on the choice of $\widehat{x}(0)$ and $P(0)$. In the Bayesian framework on which KF's rely on, these quantities should be selected as the mean and the variance matrix of the extended system state. However, in parameter estimation problems, θ is not a random variable in general, and the choice of its mean and variance is largely arbitrary and often obtained through heuristic trial and error procedures.
Summing up, the simulations above reported show that both EKF and UKF may be unsuitable for the estimation of an unknown parameter.

III. The two-stage approach

Given the discussion in Section II, a new approach to the problem of estimating an unknown parameter is introduced herein. This new method, called the *two-stage approach*, reconstructs through off-line intensive simulation trials the relationship between the input/output data and the unknown parameter. The method is fully developed in a non-Bayesian setting and it does not suffer from the convergence issue affecting EKF and UKF.
Suppose for a while that the actual system is fully accessible for experiments. In other words, it is possible to collect input/output data in a number of different experimental conditions each of which is associated with a certain chosen value of θ. In this way, one can collect N measurements $D_1^N = \{y^1(1), u^1(1), \ldots, y^1(N), u^1(N)\}$ for $\theta = \theta_1$, N measurements $D_2^N = \{y^2(1), u^2(1), \ldots, y^2(N), u^2(N)\}$ for $\theta = \theta_2$ and so on and so forth. By repeated experiments, one can work out a set of, say m, pairs $\{\theta_i, D_i^N\}$ as depicted in Table I.
Of course, one may object that performing such a number of experiments on the true system may be unaffordable in many practical situations. In many instances, the unknown parameter of a given plant is fixed and by no means modifiable; in other cases, θ may be modified but the repetition of experiments with different θ's is over-expensive. However, nowadays, any plant is usually accompanied by a very accurate simulator. Hence, the set of experiments can be

θ_1	$D_1^N = \{y^1(1),u^1(1),\ldots,y^1(N),u^1(N)\}$
θ_2	$D_2^N = \{y^2(1),u^2(1),\ldots,y^2(N),u^2(N)\}$
$\vdots$	$\vdots$
θ_m	$D_m^N = \{y^m(1),u^m(1),\ldots,y^m(N),u^m(N)\}$

TABLE I

THE OBSERVATIONS CHART AS THE STARTING POINT OF THE TWO-STAGE METHOD.

virtually performed by intensive simulation trials.

Summing up, our starting point is the set of observations $\{\theta_i, D_i^N\}$ of Table I, being it obtained by repeated experiments on the real plant or by simulations trials on the plant simulator.

The basic problem is then to find a (nonlinear) map $f : \mathbb{R}^{2N} \to \mathbb{R}^q$ capturing the relationship between observed θ's and sequences of *y*-*u* data. That is,

$$\widehat{f} \leftarrow \min_f \frac{1}{m}\sum_{i=1}^{m} \left\| \theta_i - f(y^i(1),u^i(1),\ldots,y^i(N),u^i(N)) \right\|^2. \quad (5)$$

Plainly, should $\widehat{f}$ be available, then the θ corresponding to the actual measurements $\bar{D}^N = \{\bar{y}(1),\bar{u}(1),\ldots,\bar{y}(N),\bar{u}(N)\}$ can be unveiled as $\widehat{\theta} = \widehat{f}(\bar{y}(1),\bar{u}(1),\ldots,\bar{y}(N),\bar{u}(N))$.

Solving Problem (5) requires the preliminary choice of a suitable class of functions $\mathscr{F}$ within which performing optimization. This is indeed a critical issue, due to the high dimensionality of the problem (function f depends upon $2N$ variables, normally a very large number if compared to the number m of experiments). If $\mathscr{F}$ is a class of low-complexity functions, it is difficult to replicate the whole relationship linking D^N to θ (bias error). On the opposite, if $\mathscr{F}$ is a class of high-complexity functions, then the over-fitting issue arises (variance error). As is well known, the bias vs. variance error trade off in high dimensional problems is the bane of system identification theorists and practitioners, [11], [15].

In order to achieve a sensible compromise between bias and variance error, the two-stage approach is proposed. In this method, the selection of the family of functions $\mathscr{F}$ is split in two steps. The objective of the first step is to reduce the dimensionality of the estimation problem, leading to m new artificial observations each of which is constituted by a limited number n of samples ($n \ll N$). In the second step, the map between the artificial observations and parameter θ is identified. By combining the results of the two steps, the sought relationship between the original *y*-*u* data and θ is finally unveiled.

First step. The first step consists in a compression of the information conveyed by measured input/output sequences D_i^N in order to obtain data sequences $\widetilde{D}_i^n$ of reduced dimensionality. While in the original data D_i^N the information on the unknown parameter θ_i is scattered in a long sequence of N samples, in the new artificial data $\widetilde{D}_i^n$ such information is compressed in a very short sequence of n samples ($n \ll N$). This leads to a new artificial observations chart constituted by the pairs $\{\theta_i, \widetilde{D}_i^n\}$, $i = 1,\ldots,m$, see Table II.

θ_1	$\widetilde{D}_1^n = \{\alpha_1^1,\ldots,\alpha_n^1\}$
θ_2	$\widetilde{D}_2^n = \{\alpha_1^2,\ldots,\alpha_n^2\}$
$\vdots$	$\vdots$
θ_m	$\widetilde{D}_m^n = \{\alpha_1^m,\ldots,\alpha_n^m\}$

TABLE II

THE ARTIFICIAL OBSERVATIONS CHART.

The artificial data vector $\widetilde{D}_i^n$ can be obtained from the original D_i^N by resorting to a standard identification method. To be precise, one can fit a simple model to each sequence $D_i^N = \{y^i(1),u^i(1),\ldots,y^i(N),u^i(N)\}$. The parameters of this model, say $\alpha_1^i,\alpha_2^i,\ldots,\alpha_n^i$, are seen as the new *artificial data*, i.e. $\widetilde{D}_i^n = \{\alpha_1^i,\ldots,\alpha_n^i\}$.

Example 1: To be concrete, for each $i = 1,\ldots,m$, the data sequence $D_i^N = \{y^i(1),u^i(1),\ldots,y^i(N),u^i(N)\}$ can be concisely described by an ARX model:

$$\begin{aligned} y^i(t) \;=\; & \alpha_1^i y^i(t-1) + \cdots \alpha_{n_y}^i y^i(t-n_y) + \\ & \alpha_{n_y+1}^i u^i(t-1) + \cdots + \alpha_{n_y+n_u}^i u^i(t-n_u), \end{aligned}$$

with a total number of parameters $n = n_y + n_u$. The artificial data $\alpha_1^i,\ldots,\alpha_n^i$ can be computed through the least squares algorithm ([11], [15]):

$$\begin{bmatrix} \alpha_1^i \\ \vdots \\ \alpha_n^i \end{bmatrix} = \Big[\sum_{t=1}^{N} \varphi^i(t)\varphi^i(t)^T\Big]^{-1} \cdot \sum_{t=1}^{N} \varphi^i(t) y^i(t), \quad (6)$$

$\varphi^i(t) = [y^i(t-1)\cdots y^i(t-n_y)\; u^i(t-1)\cdots u^i(t-n_u)]^T$.

Remark 1: Notice that the artificial data $\widetilde{D}_i^n = \{\alpha_1^i,\ldots,\alpha_n^i\}$ have no physical meaning. They play a purely intermediary role in the process of enlightening the relationship between the unknown parameter and the original collected data. □

To summarize, the first step aims at finding a function $g : \mathbb{R}^{2N} \to \mathbb{R}^n$ in order to transform the original data D_i^N into the new artificial data $\widetilde{D}_i^n$. Function g is implicitly defined by the chosen identification algorithm.

Second step. Once the artificial observations chart in Table II is worked out, problem (5) becomes that of finding a map $h : \mathbb{R}^n \to \mathbb{R}^q$ which fits the set of artificial observations, i.e.

$$\widehat{h} \leftarrow \min_h \frac{1}{m}\sum_{i=1}^{m} \left\| \theta_i - h(\alpha_1^i,\ldots,\alpha_n^i) \right\|^2. \quad (7)$$

Function minimization in (7) is reminiscent of the original minimization problem in (5). However, being n small, the bias vs. variance error trade-off is not an issue and the new optimization problem can be tackled by suitably parameterizing h with a Neural Network ([2]) or with another class of linear or nonlinear functions.

Perhaps it is worth noticing that a computationally cheap (but possibly loose) parametrization of h is $h(\alpha_1^i,\ldots,\alpha_n^i) = c_1\alpha_1^i + \ldots + c_n\alpha_n^i$, i.e. h is linear in $\alpha_1^i,\ldots,\alpha_n^i$. In this way, the identification from the artificial m observations can be easily performed through the least squares algorithm, [11], [15], at a low computational

cost. Notice that, in spite of the linear dependence on $\alpha_1^i,\ldots,\alpha_n^i$, the final estimator of θ is nonlinear with respect to the system input/output observations. As a matter of fact, the artificial data $\alpha_1^i,\ldots,\alpha_n^i$ nonlinearly depend upon $y^i(1),u^i(1),\ldots,y^i(N),u^i(N)$ (see e.g. equation (6)). In some cases, such nonlinearity suffices for capturing the relationship between θ and $y(1),u(1),\ldots,y(N),u(N)$. In other cases, instead, using a nonlinear h may be advisable.

Use of the method. Once function g has been chosen and function $\widehat{h}$ has been identified, the link between the original data and the unknown θ is given by $\widehat{h}(g(\cdot))$. When an actual input/ouput sequence is observed, say $\bar{D}^N = \{\bar{y}(1),\bar{u}(1),\ldots,\bar{y}(N),\ \bar{u}(N)\}$, the corresponding unknown parameter is estimated by $\widehat{\theta} = \widehat{h}(g(\bar{D}^N)))$.
As previously discussed, viewing this data-θ function as the composition of g and $\widehat{h}$ is the key to transform a numerically intractable problem into an affordable one.

IV. Simulation example

The two-stage approach was applied to system (2). Precisely, $m = 500$ values for $\theta = a$ were extracted uniformly from the interval $[-0.9,0.9]$. Correspondingly 500 sequences of 1000 output values were collected.
For each sequence $y^i(1),\ldots,y^i(1000)$, an artificial data sequence was obtained by identifying through the least squares algorithm the coefficients $\alpha_1^i,\ldots,\alpha_5^i$ of an AR(5) model $(y^i(t) = \alpha_1^i y(t-1)+\cdots+\alpha_5^i y(t-5))$. The final estimator

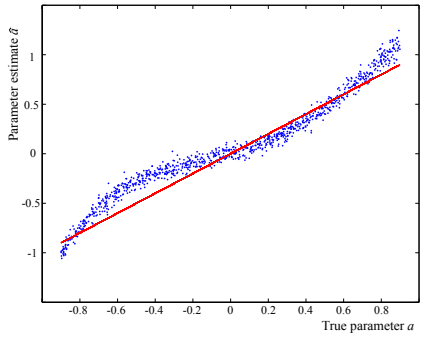

Fig. 10. Estimates of a (h linearly parameterized).

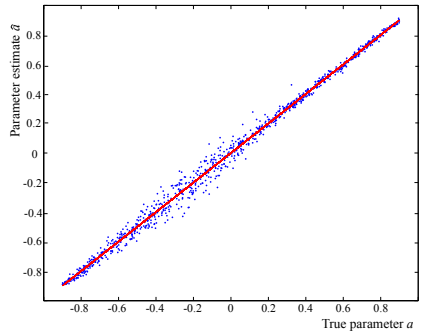

Fig. 11. Estimates of a (h parameterized via neural networks).

$\widehat{h}(\alpha_1^i,\ldots,\alpha_5^i)$, instead, was computed by resorting, first, to a linear parametrization ($h = c_1\alpha_1^i+\ldots+c_5\alpha_5^i)$), with coefficients $c_1,\ldots,c_5$ estimated again by the least squares algorithm. As an alternative, $\widehat{h}$ was also derived by resorting to a neural network (to be precise, we considered an Elman neural network with 2 layers, 10 neurons in the first layer and one neuron in the second one; the network was trained with the 500 artificial observations by the usual back-propagation algorithm).
The performance of the obtained estimator was tested by validation. To this purpose, 1000 new values for $\theta = a$ were extracted in $(-0.9,0.9)$. For each of these θ's, a sequence of $N = 1000$ outputs was generated from equation (2). The previously obtained estimator was used for each of these sequences. The performance of the obtained estimates can be appreciate in Figure 10 (h linearly parameterized) and in Figure 11 (h parameterized via neural networks).
As can be seen, the two-stage estimator works much better than EKF and UKF.

References

[1] B.D.O. Anderson and J.B. Moore. *Optimal Filtering*. Prentice Hall, 1979.
[2] S. Bittanti and G. Picci, editors. *Identification, adpatation, learning – the science of learning models from data*. Springer-Verlag, Berlin, Gemrany, 1996.
[3] T. Bohlin. *Practical grey-box identification: theory and applications*. Springer-Verlag, London, UK, 2006.
[4] M. Boutayeb, H. Rafaralay, and M. Darouch. Convergence analysis of the extended kalman filter used as an observer for nonlinear deterministic discrete-time systems. *IEEE Transaction on Automatic Control*, 42(4):581–586, 1997.
[5] A. Gelb, Jr. J.F. Kasper, Jr. R.A. Nash, C.F. Price, and Jr. A.A. Sutherland. *Applied Optimal Estimation*. MIT press, 1974.
[6] M.S. Grewal and A.P. Andrews. *Kalman Filtering - theory and practice using MATLAB*. John Wiley & Sons, 2001.
[7] S.J. Julier and J.K. Uhlmann. Unscented filtering and nonlinear estimation. *Proceedingds of the IEEE*, 92(3):401–402, 2004.
[8] S.J. Julier, J.K. Uhlmann, and H.F. Durrant-Whyte. A new method for the nonlinear transformation of means andcovariances in filters and estimators. *IEEE Transaction on Automatic Control*, 45(3):477–482, 2000.
[9] T. Kailath, A.H. Sayed, and B. Hassabi. *Linear Estimation*. Prentice-Hall, 2000.
[10] L. Ljung. Asymptotic behavior of the extended kalman filter as a parameter estimator for linear systems. *IEEE Transaction on Automatic Control*, 24(1):36–50, 1979.
[11] L. Ljung. *System Identification: Theory for the User*. Prentice-Hall, Upper Saddle River, NJ, 1999.
[12] P.E. Morall and J.W. Grizzle. Observer design for nonlinear systems with discrete-time measurements. *IEEE Transaction on Automatic Control*, 40(3):395–404, 1995.
[13] K. Reif and R. Unbehauen. The extended kalman filter as an exponential observer for nonlinear systems. *IEEE Transaction on Signal Processing*, 47(8):2324–2328, 1999.
[14] D. Simon. *Optimal state estimation*. John Wiley & Sons, Hoboken, NJ, 2006.
[15] T. Söderström and P. Stoica. *System Identification*. Prentice-Hall, Englewood Cliffs, NJ, 1989.
[16] Y. Song and J.W. Grizzle. The extended kalman filter as a local asymptotic observer for nonlinear discrete-time systems. *J. Math. Systems Estim. Contr.*, 5(1):59–78, 1995.
[17] J.K. Su and E.W. Kamen. *Introduction to Optimal Estimation*. Springer, Englewood Cliffs, NJ, 1999.
[18] W. Sun, K.M. Nagpal, and P.P. Khargonekar. H_∞ control and filtering for sampled-datasystems. *IEEE Transaction on Automatic Control*, 38(8):1162–1175, 1993.
[19] E.A. Wan and R. van der Merwe. The unscented kalman filter. In S. Haykin, editor, *Kalman filtering and Neural Networks*, New York, NY, USA, 2001. John Wiley & Sons.