51st IEEE Conference on Decision and Control
December 10-13, 2012. Maui, Hawaii, USA

Piecewise Affine Direct Virtual Sensors with Reduced Complexity

Matteo Rubagotti, Tomaso Poggi, Alberto Bemporad and Marco Storace

Abstract— **In this paper, a piecewise-affine direct virtual sensor is proposed for the estimation of unmeasured outputs of nonlinear systems whose dynamical model is unknown. In order to overcome the lack of a model, the virtual sensor is designed directly from measured inputs and outputs. The proposed approach generalizes a previous contribution, allowing one to design lower-complexity estimators. Indeed, the reduced-complexity approach strongly reduces the effect of the so-called "curse of dimensionality", and can be applied to relatively high-order systems, while enjoying all the convergence and optimality properties of the original approach.**

I. Introduction

The estimation of unmeasurable variables of a dynamical system using available measurements and information on the system dynamics is a widely studied problem in control theory. If the system is nonlinear, it is usually impossible to rely on optimal solutions (such as the Kalman filter for linear systems), and approximate solutions must be sought, such as extended Kalman filters, unscented Kalman filters, ensemble Kalman filters, particle filters, and moving horizon estimation. These methods require a model of the system to be applied. However, in many practical applications reliable models are not available, and a problem of *filter design from data* must be solved. The standard *two-step procedure* to address this problem is the following:

1) obtain a model by using system identification;
2) design an observer based on the resulting model.

In this way, the overall performance is usually far from optimal, and alternative strategies were recently proposed. In particular, in [1] a direct (one-step) procedure for designing an optimal filter was proposed, which is applicable to nonlinear systems, and is proven to be the minimum variance estimator among the selected class of approximating filters. We refer to the observer obtained using the direct procedure as the *direct virtual sensor* (DVS). The DVS is a function of past measured inputs and outputs, and possibly of past estimates: no model of the system is required, only observability of the variable to be estimated is assumed as a necessary condition. The method consists of choosing a suitable set of basis functions, that leads to satisfying the assumptions required to apply the theoretical results in [2]. Apart from [1], where a stochastic framework is considered, a different approach to DVS design for nonlinear systems can be found in [3], where a set membership approach is exploited. The DVS has also been applied to relevant automotive case studies in [4]–[7].

In [8], piecewise-affine simplicial (PWAS) functions were proposed for DVS design. The main motivation was that PWAS functions can be implemented very efficiently in digital circuits (e.g. field-programmable gate arrays, FPGAs [9], or application specific integrated circuits, ASICs), thus providing fast response times, low cost, and low power consumption (at least for ASICs). The DVS in PWAS form was tested on both simulation and experimental data in [8], leading to an estimation accuracy of the same order of magnitude as in [1]. Moreover, the implementation of the DVS on a low-cost commercial FPGA led to latency times[1] smaller than 100 ns.

The main drawback of the approach presented in [8], henceforth referred to as Standard DVS (S-DVS), is that a single PWAS function is used to obtain the estimate. If a relatively large number of inputs or measurable outputs is available, or if a large number of past data are used, the exponential increase of the complexity ("curse of dimensionality", [10]) makes the approach impractical. In this paper, we propose a generalization of the approach of [8], that leads to a complexity reduction. The resulting Reduced-Complexity DVS (RC-DVS) is expressed as the sum of lower-dimensional PWAS functions instead of using a single higher-dimensional PWAS function. Moreover, past values of the estimated variable can be employed, which was not considered in [8].

A discrete-time version of Lorenz's system, whose parameters are set to make the dynamics chaotic, is chosen as a benchmark to compare our results with those of [8]. Note that the same benchmark was also used in [1].

The paper is organized as follows: Section II introduces the required system theoretical properties, and Section III describes the structure and actual implementation of the PWAS virtual sensor. Section IV deals with the convergence properties of the proposed RC-DVS, whereas the issues related to its practical implementation are discussed in Section

M. Rubagotti is with Nazarbayev University, 53 Kabanbay Batyr Ave, 010000 Astana, Kazakhstan. Email: matteo.rubagotti@nu.edu.kz.

T. Poggi is with the RF group, ESS-Bilbao, Paseo Landabarri, 2, 48940 Leioa, Spain. E-mail: tpoggi@essbilbao.org.

A. Bemporad is with IMT Institute for Advanced Studies, Piazza S. Ponziano 6, 55100 Lucca, Italy. E-mail: alberto.bemporad@imtlucca.it.

M. Storace is with the Biophysical and Electronic Engineering Department, University of Genoa, Via Opera Pia 11a, 16145 Genova, Italy. E-mail: marco.storace@unige.it.

This work was partially supported by the European Commission through project MOBY-DIC "Model-based synthesis of digital electronic circuits for embedded control" (FP7-INFSO-ICT-248858), http://www.mobydic-project.eu.

[1] The latency of a circuit is the time needed to process an input and provide the corresponding output.

978-1-4673-2066-5/12/$31.00 ©2012 IEEE

V. Simulation examples are presented in Section VI, and conclusions are drawn in Section VII.

II. Preliminaries

We focus our attention on a nonlinear discrete-time dynamical system $\mathcal{S}$

$$\mathcal{S}:\left\{\begin{array}{rcl} x(t+1) & = & g(x(t),u(t)) \\ y(t) & = & h_y(x(t)) \\ z(t) & = & h_z(x(t)) \end{array}\right. \tag{1}$$

where the state vector is $x \in \mathbb{R}^{n_x}$, the input vector is $u \in \mathbb{R}^{n_u}$, the vector of measurable outputs is $y \in \mathbb{R}^{n_y}$, and t represents the discrete-time instant. Vector $z \in \mathbb{R}^{n_z}$ collects a set of variables to be estimated. We assume that only during training experiments $z(t)$ can be measured by a *real sensor* at time instants $t = 0, ..., T$. These measurements (the training set) are used to design the virtual sensor, which will operate without measuring $z(t)$. The functions $g(\,\cdot\,,\,\cdot\,) : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$, $h_y(\,\cdot\,) : \mathbb{R}^{n_x} \to \mathbb{R}^{n_y}$, and $h_z(\,\cdot\,) : \mathbb{R}^{n_x} \to \mathbb{R}^{n_z}$ are assumed unknown, whereas $n_z = 1$ is assumed for simplicity, without loss of generality, since the case $n_z > 1$ can be solved by n_z scalar problems in a component-wise fashion.

The possibility of estimating $z(t)$ is related to the concept of observability. Indeed, as stated in [3], the observability of the system implies that z can be uniquely determined using a finite number $0 \le M_u \le n_x$ of samples of u, a finite number $1 \le M_y \le n_x$ of samples of y, and a finite number $0 \le M_z \le n_x$ of samples of z. In particular, if $\mathcal{S}$ is *observable*, there exists a function f_z such that $z(t) = f_z(U(t), Y(t), Z(t))$, with

$$\begin{aligned} U(t) &\triangleq \left[\; u(t-M_u+1)' \;\; u(t-M_u+2)' \;\; \cdots \;\; u(t)' \;\right]' \\ Y(t) &\triangleq \left[\; y(t-M_y+1)' \;\; y(t-M_y+2)' \;\; \cdots \;\; y(t)' \;\right]' \\ Z(t) &\triangleq \left[\; z(t-M_z+1)' \;\; z(t-M_z+2)' \;\; \cdots \;\; z(t-1)' \;\right]' \end{aligned}$$

where $'$ denotes transposition. Two cases can be distinguished:

- if $\mathcal{S}$ is *fully observable*, the function f_z can be defined such that $z(t) = f_z(U(t), Y(t))$, i.e., the past values of z are not needed, since the whole state x can be reconstructed from $Y(t)$ and $U(t)$, and z is a static function of x; this was the case considered in [8];
- if $\mathcal{S}$ is *partially observable*, it is not possible to reconstruct all x, and past values of z are needed to reconstruct $z(t)$.

Vector $U(t)$ is empty when the system is autonomous, while $Z(t)$ is empty when the system is fully observable.

III. The proposed direct virtual sensor

A. General formulation of the DVS

Assuming that system $\mathcal{S}$ is (partially or fully) observable, the DVS is a function providing the estimate $\hat{z}(t)$ of z at time t. Since the actual variables y, u and z at past time instants are not available, the noisy measurements of them are assumed to be $\tilde{u}(t) = u(t) + \eta_u(t)$, $\tilde{y}(t) = y(t) + \eta_y(t)$, and $\tilde{z}(t) = z(t) + \eta_z(t)$, where η_u, η_y, and η_z are unknown stochastic variables. For given values of M_u, M_y, and M_z, the inputs of the DVS will be noisy sequences of measurements of y and u, and a vector of past values of $\hat{z}$, namely[2]

$$\begin{aligned} \tilde{U}(t) &\triangleq \left[\; \tilde{u}(t-M_u+1)' \;\; \tilde{u}(t-M_u+2)' \;\; \cdots \;\; \tilde{u}(t)' \;\right]' \\ \tilde{Y}(t) &\triangleq \left[\; \tilde{y}(t-M_y+1)' \;\; \tilde{y}(t-M_y+2)' \;\; \cdots \;\; \tilde{y}(t)' \;\right]' \\ \hat{Z}(t) &\triangleq \left[\; \hat{z}(t-M_z)' \;\; \hat{z}(t-M_z+1)' \;\; \cdots \;\; \hat{z}(t-1)' \;\right]' \end{aligned}$$

Remark 1: If a model of the system is available (though not directly used to obtain the DVS), it is possible to check the observability of the system and to determine suitable values for M_u, M_y, M_z. If a model is not available, observability is simply assumed a priori, and the values of M_u, M_y, M_z are considered as tuning parameters. □

For the sake of compactness, henceforth the input of the DVS is referred to as

$$\Xi(t) \triangleq \left[\; \tilde{U}'(t) \quad \tilde{Y}'(t) \quad \hat{Z}'(t) \;\right]' \in \mathbb{R}^{n_\xi}$$

where $n_\xi \triangleq M_u n_u + M_y n_y + M_z$. Assume to split vector Ξ into $\nu \in \mathbb{N}$ subsets Ξ^1, $\Xi^2, ..., \Xi^\nu$, such that all elements of Ξ are included in one and only one of these subsets. Each of the Ξ^j, $j = 1, ..., \nu$, has dimension equal to N_j, such that $1 \le N_j \le n_\xi$, and $N_1 + N_2 + ... + N_\nu = n_\xi$. The N_j elements of each Ξ^j are denoted as $\xi_{j,1}, \xi_{j,2}, ..., \xi_{j,N_j}$. The proposed DVS is referred to as $\mathcal{V}_\alpha(w)$ and is defined as follows:

$$\hat{z}(t) = f_\alpha(\Xi(t); w) = \sum_{j=1}^{\nu} \sum_{k=1}^{N_j} w_{j,k} \alpha_{j,k}(\Xi^j(t)) \tag{2}$$

where $f_\alpha : \mathbb{R}^{n_\xi} \to \mathbb{R}$ (for fixed w), and $\{\alpha_{j,k}\}$ is a basis of PWAS functions that is described in Section III-B. Also,

$$w \triangleq \left[\, w_{1,1} \cdots w_{1,N_1} \; w_{2,1} \cdots w_{2,N_2} \cdots w_{\nu,1} \cdots w_{\nu,N_\nu} \,\right]'$$

with $w \in D_w \subset \mathbb{R}^{n_\xi}$, D_w being a convex compact set. The vector of parameters w is obtained by solving the least-squares problem

$$w^* = \arg\min_w \left\{ \sum_{t=M}^{T-1} \left[\tilde{z}(t+1) - f_\alpha(\Xi(t); w)\right]^2 \right\} \tag{3}$$

where $M = \max(M_u, M_y, M_z)$, $M \ll T$. Notice that, since (2) is linear with respect to the weights w and since the cost function is quadratic and defined over the convex set D_w, the optimization problem (3) is convex.

Remark 2: We generalize the approach of [8] in two directions. First, the past values of $\hat{z}$ were not considered in [8], where $\Xi(t) \triangleq \left[\; \tilde{U}'(t) \quad \tilde{Y}'(t) \;\right]'$, so the approach was not applicable to systems that are not fully observable. Moreover, past values of $\hat{z}$ can be used for estimation even when the system is fully observable, in order to increase the performance of the DVS. Second, in [8] there was no partitioning of Ξ, and then the PWAS S-DVS was defined over a domain of dimension n_ξ. This might cause serious implementation problems, since the number of coefficients

[2] Vector $\tilde{U}(t)$ is empty when the system is autonomous, $\hat{Z}(t)$ is empty when the the past values of $\hat{z}$ are not used for the estimation.

in (2) increases exponentially with n_ξ. The possibility of splitting the domain into subspaces can lead to huge practical advantages, as will be highlighted in Section VI. □

Remark 3: Another practical problem is related to the initialization phase of the DVS. If the DVS starts receiving measurements at time $t = 0$, the needed past values of $\tilde{u}$ and $\tilde{y}$ will be available at time $t_{uy} \triangleq \max(M_u, M_y)$. Therefore, if $M_z = 0$, the DVS will start providing its output at time t_{uy}. In case $M_z > 0$, past values of $\hat{z}$ would be needed, and then the DVS will be able to generate its output at time $t = M - 1$, using an initial guess for the past values of $\hat{z}$. In practical applications, the initial guess can be related to an a-priori knowledge of the initial condition. For example, when estimating the sideslip angle in a vehicle [6], since the DVS starts working when the engine is turned on, it is perfectly reasonable to assume that the vehicle is not moving, which implies that the sideslip angle is equal to zero. □

B. Digital implementation of the DVS

To implement the RC-DVS (2) on a digital circuit, we consider a class of continuous and regular PWAS basis functions, defined over regular partitions of hyper-rectangular domains

$$S_j = \left\{ \Xi^j \in \mathbb{R}^{N_j} : \underline{\xi}_{j,i} \le \xi_{j,i} \le \bar{\xi}_{j,i},\ j = 1, ..., \nu,\ i = 1, ..., N_j \right\} \tag{4}$$

The circuits proposed in [9] can evaluate PWAS functions defined over this kind of domains. Each of the S_j is partitioned into a set of regular *simplices*, and the functions that can be obtained by combining the elements of this basis are in turn PWAS functions. For details on simplices and PWAS functions, the reader is referred to [8] and the references therein.

If the algorithm proposed in [11] is used to define the simplicial partitioning, the numbers of vertices and simplices are equal to $N_j = \prod_{i=1}^{N_j}(p_{j,i} + 1)$ and $L = N_j! \prod_{i=1}^{N_j} p_{j,i}$, respectively, where $p_{j,i}$ is the number of non-overlapping subintervals of equal length into which each interval $\left[\underline{\xi}_{j,i}, \bar{\xi}_{j,i}\right]$ is partitioned. The union of all the simplicial partitions of S_j is equal to S_j itself, and the interiors of the simplicial partitions are disjoint.

Different types of continuous basis functions can be defined; we use here the so-called α-basis [12]. Each function $\alpha_{j,k}(\Xi^j)$ in (2) is a PWAS hyper-pyramid, which takes the value 1 at the vertex $v_{j,k}$ (i.e., the $k - th$ vertex of the $j - th$ domain) and 0 at all the other vertices $v_{j,q},\ q \neq k$. Every element of the α-basis has a local nature, is affine over each simplex, and moreover $0 \le \alpha_{j,k}(\Xi^j) \le 1,\ \forall\ \Xi^j \in S^j$. For the use of other bases to represent PWAS functions the reader is referred to [13], [14].

Analogously to [8], the implementation of the proposed DVS on a digital circuit consists of two blocks: a bank of registers to store the past values of $\tilde{u}$, $\tilde{y}$, and $\hat{z}$ (i.e., $\Xi(t)$), and an arithmetic unit to calculate the value of the PWAS function f_α. We implement the PWAS function on FPGA using linear interpolators: The value of the j-th component of (2) is obtained by linearly interpolating $N_j + 1$ values, i.e., the values assumed by the function at the vertices of the corresponding simplex. To solve the point location problem, an algorithm based on Kuhn's lemmas [11] is used, which is optimal with respect to the number of inputs [15].

IV. Convergence analysis of the estimation error

Consider a standard two-step procedure to obtain a minimum variance filter $\mathcal{K}(\theta)$, estimating a state-space model of (1) from a set of available measurements, and then designing $\mathcal{K}(\theta)$ based on this model. Generally speaking, the filter $\mathcal{K}(\theta)$ will be based on the set of parameters $\theta \in D_\theta$ (D_θ being a compact set), and designed relying on a class of models $\mathcal{M}(\theta)$ of (1). In particular, we consider the model $\mathcal{M}(\theta^*)$, obtained using a prediction error method from a set of measured data, and the corresponding filter realization $\mathcal{K}(\theta^*)$. It is possible to represent the estimate given by $\mathcal{K}(\theta^*)$ in regression form as

$$\hat{z}_{\mathcal{K}}(t+1) = f_{\mathcal{K}}(\Xi(t); \theta^*). \tag{5}$$

The following theorem describes the properties of the proposed RC-DVS (2) in comparison with (5).

Theorem 1: Let system (1) be (partially or fully) observable. Consider a minimum variance filter $\mathcal{K}(\theta^*)$ in (5), and the virtual sensor $\mathcal{V}_\alpha(w^*)$ in (2), whose parameter vector w^* is obtained from (3). Let $\hat{z}_{\mathcal{V}}$ be the value of the estimate obtained with a RC-DVS $\mathcal{V}_\alpha(w^*)$ in (2). Then, denoting expected values by $E[\cdot]$, the following results hold with probability 1 as $T \to \infty$:

$i)$ The vector of parameters defined in (3) guarantees the minimization of the variance of the estimation error among all the virtual sensors with the same structure, i.e., $\mathcal{V}_\alpha(w^*) = \arg \min\limits_{\mathcal{V}_\alpha(w)} E\left[(z(t) - \hat{z}_{\mathcal{V}})^2\right]$;

$ii)$ If there exists w such that $\mathcal{K}(\theta^*) = \mathcal{V}(w)$ (i.e., it is possible to express the two-step observer in regression form as a particular realization of the virtual sensor), one obtains that $E\left[(z(t) - \hat{z}_{\mathcal{K}}(t))^2\right] \ge E\left[(z(t) - \hat{z}_{\mathcal{V}}(t))^2\right]$, i.e. the performance of the RC-DVS is better than or equal to that of (5);

$iii)$ If there exists $\theta^o \in D_\theta$ such that $\mathcal{S} = \mathcal{M}(\theta^o)$ (i.e., there exists a set of parameters of the two-step observer that describes exactly the system), and there exists a vector w such that $\mathcal{K}(\theta^o) = \mathcal{V}(w)$, then $\mathcal{V}(w^*)$ is a minimum variance filter.

For $ii)$ and $iii)$ to be applied, it is necessary that $\mathcal{K}(\theta^*)$ has fading memory. □

Proof: See the Appendix. ■

In conclusion, the proposed RC-DVS retains all the positive features of the general DVS framework of [1], [3].

V. Hints on implementation issues

When designing the proposed RC-DVS, some practical issues must be taken into account. These are briefly described in the following, and the reader is referred to [8] for a deeper analysis.

If problem (3) is solved directly, it is possible to obtain a solution that is sensitive to small changes in the data. A possible way to solve this problem relies on the so-called *Tikhonov regularization*, consisting of obtaining w by solving the *regularized least squares* (RLS) problem

$$\min_{w} \left\{ \sigma w' \Gamma w + \sum_{t=M}^{T-1} \left[\tilde{z}(t) - f_{\alpha}(\Xi(t); w) \right]^2 \right\} \tag{6}$$

where σ is the *Tikhonov regularization parameter*. The reader is referred to [16] for further details.

The choice of the domains S_j for the PWAS function also requires some attention. If no a-priori information is available, the size of the sets S^j must be estimated before solving (6). If $\{\hat{S}_j\}$ is the set of hyper-rectangles that exactly contains all data, the choice $\{S_j\} = \{\hat{S}_j\}$ is not a good choice, because some trajectories could exit $\{\hat{S}_j\}$ in normal operating conditions. Following [8], the sets S_j are then computed as an expansion of the sets $\hat{S}_j$ with respect to their centers by a constant factor $\gamma > 1$, whose choice relies on heuristic criteria.

The size of w, which is equal to n_ξ, depends on how many simplices we use to obtain the partitions of the sets S_j. The value of n_ξ influences the complexity of the optimization problem (6) and, most important, the dimension of the memory required by the circuit implementation.

The parameters M_u, M_y and M_z are related in theory to the observability properties of the system. However, when a model of the system is not available a priori, they become design parameters. If the system is observable, large values of M_u, M_y and M_z lead to a better estimate, but also to large latency times and memory requirements of the digital circuit. M_u, M_y and M_z also influence the number of coefficients w of a DVS. In particular, for a S-DVS the size of w grows exponentially as these parameters increase. On the contrary, by using the proposed RC-DVS one can decide to increase the value of ν while keeping M_u, M_y and M_z fixed, in order to reduce the effect of the exponential increasing of circuit complexity, as shown in the case study of Section VI.

The distribution of the data in the time interval $[0, T]$, is also important, since all the main dynamic properties of the system (including transient responses) must lie within this time window (see, e.g., [17]).

VI. Simulation results

We test the performance of the RC-DVS on a simulation example, that permits a comparison with the S-DVS proposed in [8]. Consider the discrete-time Lorenz system

$$\begin{cases} x_1(t+1) = (1 - \tau s)x_1(t) + \tau s x_2(t) \\ x_2(t+1) = (1 - \tau)x_2(t) - \tau x_1(t)x_3(t) + \tau \rho x_1(t) \\ x_3(t+1) = (1 - \tau\beta)x_3(t) + \tau x_1(t)x_2(t) \\ \tilde{y}_1(t) = x_1(t)x_2(t) + \eta_{y_1}(t) \\ \tilde{y}_2(t) = x_2^2(t) + \eta_{y_2}(t) \\ \tilde{z}(t) = \sin(0.1 x_3(t)) + \eta_z(t) \end{cases} \tag{7}$$

where $\tau = 0.01$ is the sampling time, $s = 10$, $\beta = 8/3$ and $\rho = 28$ are fixed parameters, and $\eta_{y_1}(t)$, $\eta_{y_2}(t)$ and $\eta_z(t)$ are Gaussian processes with zero mean and standard deviations equal to 0.02, 0.02 and 0.01, respectively. With this set of parameters, system (7) exhibits a chaotic behavior. The S-DVS has already been tested on the same system in [8], and compared with the approach of [1]. Simulations were carried out using the Root Mean Square Estimation Error (RMSEE) calculated over a test set as a measure of the accuracy of the estimation

$$RMSEE = \sqrt{\frac{1}{T_s} \sum_{t=1}^{T_s} (\hat{z}(t) - \tilde{z}(t))^2} \tag{8}$$

where T_s is the number of samples in the test set. As a result, the values of the RMSEE for the two approaches were very close to each other.

Simulation	Method	n_ξ	M_z	M_y	RMSEE
A	S-DVS	256	0	2	0.201
	RC-DVS	32	0	2	0.225
B	S-DVS	4096	2	2	0.042
	RC-DVS	64	2	2	0.060
C	RC-DVS	4096	64	64	0.002

TABLE I

Parameters and simulation results for RC-DVS and S-DVS applied to Lorenz's system

In the following, a RC-DVS and a S-DVS are derived from a set of $T = 60000$ samples of $\tilde{z}(t)$ and $\tilde{y}(t)$. The parameters (M_y, M_z) of the two DVS have been varied in order to show the differences between the two methods. Note that the Lorenz system is autonomous ($n_u = 0$), so that M_u can be ignored.

In order to derive the RC-DVS, ν has been set to $\nu = \max\{M_y, M_z\}$, i.e., $\Xi(t)$ is divided into a number of subsets equal to the number of past samples used by the RC-DVS itself. As a consequence, the estimate $\hat{z}(t)$ is given by the sum of ν PWAS functions. We select a uniform partition with 3 subdivisions along each dimension and a zero-order Tikhonov regularization ($\sigma = 10^{-3}$). The remaining parameters used for the virtual sensors are reported in Table I. Table I also shows the RMSEE, calculated over $T_s = 3000$ samples, for RC-DVS and S-DVS.

Simulation A shows that the RMSEE of S-DVS is lower than the RMSEE of RC-DVS if the same value of M_y (i.e. past samples of the measurable output) is used. Nevetheless, the complexity in terms of coefficients is higher in the case of S-DVS (256 instead of 32).

In Simulation B we allowed the RC-DVS and the S-DVS to use past estimates $\hat{z}(t)$, i.e., we set $M_z = 2 > 0$. Figure 1 shows the transient response of the S-DVS and the RC-DVS virtual sensors. It is apparent that the performance is better at the cost of a higher complexity for both virtual sensors.

Finally, Simulation C shows the results obtained by increasing the value of M_y and M_z for the RC-DVS until the same number of coefficients of the S-DVS of Simulation B is

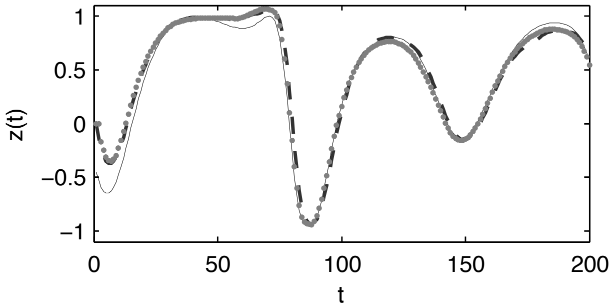

Fig. 1. Transient response of a virtual sensor: true data (solid black), S-DVS estimate (dashed gray) and RC-DVS estimate (dotted gray).

reached. In this case, the estimation error of the RC-DVS is lower than the error obtained with the S-DVS of Simulation B. Notice that in this case it is not possible to derive a practical realization of the S-DVS with $M_y = M_z = 64$, since the resulting PWAS function would be defined by more than 10^{115} coefficients.

VII. Conclusions

In this paper, a reduced-complexity PWAS direct virtual sensor was proposed to overcome the curse of dimensionality of the original approach in [8] while maintaining the same theoretical properties. Its practical implementation in low-cost digital circuits (FPGA) at very fast rates makes the approach very appealing for industrial applications, when unmeasurable variables of relatively low-order systems must be estimated with high sampling frequencies.

References

[1] M. Milanese, C. Novara, K. Hsu, and K. Poolla, "Filter design from data: direct vs. two-step approach," in *Proc. American Control Conference*, Minneapolis, MN, 2006.

[2] L. Ljung, "Convergence analysis of parametric identification methods," *IEEE Trans. Autom. Contr.*, vol. 23, no. 5, pp. 770–783, 1978.

[3] M. Milanese, C. Novara, K. Hsu, and K. Poolla, "The filter design from data (FD2) problem: nonlinear set membership approach," *Automatica*, vol. 45, no. 10, pp. 2350–2357, 2009.

[4] M. Canale, L. Fagiano, F. Ruiz, and M. Signorile, "A study on the use of virtual sensors in vehicle control," in *Proc. Conf. Decision and Control*, Cancun, Mexico, 2008.

[5] P. Borodani, "Virtual sensors: an original approach for dynamic variables estimation in automotive control systems," in *Proc. Int. Symposium on Advanced Vehicle Control*, Kobe, Japan, 2008.

[6] M. Canale, L. Fagiano, and C. Novara, "Vehicle side-slip angle estimation using a direct MH estimator," in *Proc. Int. Conf. Contr. Appl.*, Yokohama, Japan, 2010.

[7] C. Novara, F. Ruiz, and M. Milanese, "Direct identification of optimal SM-LPV filters and application to vehicle yaw rate estimation," *IEEE Trans. Contr. Sys.Tech.*, vol. 19, no. 1, pp. 5–17, 2011.

[8] T. Poggi, M. Rubagotti, A. Bemporad, and M. Storace, "High-speed piecewise affine virtual sensors," *IEEE Trans. Ind. Electr.*, vol. 56, no. 2, pp. 1228–1237, 2012.

[9] M. Storace and T. Poggi, "Digital architectures for the circuit implementation of PWL multi-variate functions: two FPGA implementations," *Int. J. Circuit Theory Appl.*, vol. 39, pp. 1–15, 2011.

[10] R. E. Bellman, *Nonlinear Programming*. Princeton University Press, 1957.

[11] H. Kuhn, "Some combinatorial lemmas in topology," *IBM J. of Res. Developm.*, vol. 4, no. 5, pp. 518–524, 1960.

[12] P. Julián, A. Desages, and B. D'Amico, "Orthonormal high level canonical PWL functions with applications to model reduction," *IEEE Trans. Circ. Syst. I*, vol. 47, no. 5, pp. 702–712, 2000.

[13] M. Storace, L. Repetto, and M. Parodi, "A method for the approximate synthesis of cellular non-linear networks – Part 1: Circuit definition," *Int. J. Circuit Theory Appl.*, vol. 31, no. 3, pp. 277–297, 2003.

[14] L. Repetto, M. Storace, and M. Parodi, "A method for the approximate synthesis of cellular non-linear networks – Part 2: Circuit reduction," *Int. J. Circuit Theory Appl.*, vol. 31, no. 3, pp. 299–313, 2003.

[15] R. Rovatti, M. Borgatti, and R. Guerrieri, "A geometric approach to maximum-speed n-dimensional continuous linear interpolation in rectangular grids," *IEEE Trans. Comput*, vol. 47, no. 8, pp. 894–899, 1998.

[16] R. Aster, B. Borchers, and C. Thurber, *Parameters Estimation and Inverse Problems*, 2004.

[17] O. De Feo and M. Storace, "Piecewise-linear identification of nonlinear dynamical systems in view of their circuit implementations," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 54, no. 7, pp. 1542–1554, July 2007.

Appendix

Proof of Theorem 1: Analogously to [1] and [8], one needs to show that conditions S3, C1, and M1 in [2] hold, which leads to the fulfillment of i), ii), and iii). Condition S3 refers to the data set and is satified if we assume that system (1) is observable. Condition C1 refers to the choice of vector w and is fulfilled if the quadratic criterion in (3) is adopted. Condition M1 requires to check if the proposed DVS retains the following property: there exist two scalars $C > 0$ and λ, $0 < \lambda < 1$, such that

1) The estimate is limited at origin, namely
$$|f_\alpha(\Xi_0(t); w)| \leq C \tag{9}$$
for $\Xi_0(t) \triangleq \left[\ \tilde{U}_0'(t) \quad \tilde{Y}_0'(t) \quad \hat{Z}_0'(t)\ \right] = 0 \in \mathbb{R}^{n_\xi}$;
2) The virtual sensor (2) has exponential fading memory
$$\begin{aligned}|f_\alpha(\Xi_1(t); w) &- f_\alpha(\Xi_2(t); w)| \\ &\leq C \sum_{s=0}^{t} \lambda^{t-s} \Big[\|\tilde{u}_1(s) - \tilde{u}_2(s)\|_1 \\ &+ \|\tilde{y}_1(s) - \tilde{y}_2(s)\|_1 + \|\hat{z}_1(s) - \hat{z}_2(s)\|_1 \Big]\end{aligned} \tag{10}$$
for any $\Xi_1(t)$, $\Xi_2(t)$;
3) Function f_α is differentiable with respect to w for all $w \in D_w$ and the following exponential fading property is satisfied:
$$\begin{aligned}\|\nabla_w f_\alpha(\Xi_1(t); w) &- \nabla_w f_\alpha(\Xi_2(t); w)\|_1 \\ &\leq C \sum_{s=0}^{t} \lambda^{t-s} \Big[\|\tilde{u}_1(s) - \tilde{u}_2(s)\|_1 \\ &+ \|\tilde{y}_1(s) - \tilde{y}_2(s)\|_1 + \|\hat{z}_1(s) - \hat{z}_2(s)\|_1 \Big]\end{aligned} \tag{11}$$
for any $\Xi_1(t)$, $\Xi_2(t)$.

In the reminder of the proof we will prove that all three properties hold. Recalling that $0 \leq \alpha_k(\,\cdot\,) \leq 1$ holds for all k, it yields

$$\begin{aligned}|f_\alpha(\Xi_0(t); w)| &= \left| \sum_{j=1}^{\nu} \sum_{k=1}^{N_j} w_{j,k} \alpha_{j,k}(\Xi_0^j(t)) \right| \\ &\leq \sum_{j=1}^{\nu} \sum_{k=1}^{N_j} |w_{j,k}| \triangleq C_1 > 0\end{aligned}$$

which implies the fulfilment of (9).

Consider the left-hand side of (10):

$$\begin{aligned}
&|f_\alpha(\Xi_1(t);w) - f_\alpha(\Xi_2(t);w)| \\
&\quad \leq \left| \sum_{j=1}^{\nu} \sum_{k=1}^{N_j} w_{j,k} \left(\alpha_{j,k}(\Xi_1^j(t)) - \alpha_{j,k}(\Xi_2^j(t)) \right) \right| \\
&\quad \leq \sum_{j=1}^{\nu} \sum_{k=1}^{N_j} \left| w_{j,k} \left[\alpha_{j,k}(\Xi_1^j(t)) - \alpha_{j,k}(\Xi_2^j(t)) \right] \right| \\
&\quad = \sum_{j=1}^{\nu} \sum_{k=1}^{N_j} |w_{j,k}| \left| \alpha_{j,k}(\Xi_1^j(t)) - \alpha_{j,k}(\Xi_2^j(t)) \right| .
\end{aligned}$$

The structure of the α-basis chosen for the design of the virtual sensor implies that each basis function $\alpha_{j,k}$ is Lipschitz continuous with respect to the input $\Xi(t)$. More precisely, there exists $\beta > 0$ such that

$$|\alpha_{j,k}(\Xi_1^j(t)) - \alpha_{j,k}(\Xi_2^j(t))| \leq \beta \left\| \Xi_1^j(t) - \Xi_2^j(t) \right\|_1 \qquad (12)$$

for all $(j,k) \in (1,...,\nu \times 1,...,N_j)$. Then,

$$\begin{aligned}
&\sum_{j=1}^{\nu} \sum_{k=1}^{N_j} |w_{j,k}| \left| \alpha_{j,k}(\Xi_1^j(t)) - \alpha_{j,k}(\Xi_2^j(t)) \right| \\
&\qquad \leq \beta \sum_{j=1}^{\nu} \sum_{k=1}^{N_j} |w_{j,k}| \left\| \Xi_1^j(t) - \Xi_2^j(t) \right\|_1 \\
&\qquad \leq \beta C_1 \sum_{j=1}^{\nu} \sum_{k=1}^{N_j} \left\| \Xi_1^j(t) - \Xi_2^j(t) \right\|_1 \\
&\qquad = \beta C_1 \sum_{j=1}^{\nu} N_j \left\| \Xi_1^j(t) - \Xi_2^j(t) \right\|_1 \\
&\qquad \leq \beta C_1 n_\xi \sum_{j=1}^{\nu} \left\| \Xi_1^j(t) - \Xi_2^j(t) \right\|_1 \\
&\qquad = \beta C_1 n_\xi \left\| \Xi_1(t) - \Xi_2(t) \right\|_1 \qquad (13)
\end{aligned}$$

Consider the right-hand side of (10), and take any λ, $0 < \lambda < 1$. Moreover, let $C_2 > 0$ be a constant to be determined. Recalling that $M = \max(M_u, M_y, M_z)$, it yields

$$\begin{aligned}
&C_2 \sum_{s=0}^{t} \lambda^{t-s} \Big[\left\| \tilde{u}_1(s) - \tilde{u}_2(s) \right\|_1 \\
&\qquad + \left\| \tilde{y}_1(s) - \tilde{y}_2(s) \right\|_1 + \left\| \hat{z}_1(s) - \hat{z}_2(s) \right\|_1 \Big] \\
&\qquad \geq C_2 \left(\sum_{s=t-M_u+1}^{t} \lambda^{t-s} \left\| \tilde{u}_1(s) - \tilde{u}_2(s) \right\|_1 \right. \\
&\qquad\quad + \sum_{s=t-M_y+1}^{t} \lambda^{t-s} \left\| \tilde{y}_1(s) - \tilde{y}_2(s) \right\|_1 \\
&\qquad\quad \left. + \sum_{s=t-M_z+1}^{t} \lambda^{t-s} \left\| \hat{z}_1(s) - \hat{z}_2(s) \right\|_1 \right) \\
&\qquad \geq C_2 \lambda^{M-1} \left(\sum_{s=t-M_u+1}^{t} \left\| \tilde{u}_1(s) - \tilde{u}_2(s) \right\|_1 \right. \\
&\qquad\quad + \sum_{s=t-M_y+1}^{t} \left\| \tilde{y}_1(s) - \tilde{y}_2(s) \right\|_1 \\
&\qquad\quad \left. + \sum_{s=t-M_z+1}^{t} \lambda^{t-s} \left\| \hat{z}_1(s) - \hat{z}_2(s) \right\|_1 \right) \\
&\qquad = C_2 \lambda^{M-1} \left\| \Xi_1(t) - \Xi_2(t) \right\|_1
\end{aligned}$$

If we define

$$C_2 \triangleq \lambda^{1-M} C_1 \beta n_\xi$$

we obtain

$$C_1 \beta n_\xi \left\| \Xi_1(t) - \Xi_2(t) \right\|_1 = C_2 \lambda^{M-1} \left\| \Xi_1(t) - \Xi_2(t) \right\|_1$$

which implies the fulfillment of (10).

Function f_α is differentiable with respect to w, and its gradient is

$$\begin{aligned}
\nabla_w f_\alpha(\Xi(t);w) &= \nabla_w \left(\sum_{k=1}^{N_1} w_{1,k} \alpha_{1,k}(\Xi^1(t)) \right. \\
&\qquad \left. + \ldots + \sum_{k=1}^{N_\nu} w_{1,k} \alpha_{\nu,k}(\Xi^\nu(t)) \right) \\
&= \left[\; V'_{\alpha_1}(\Xi^1(t)) \quad V'_{\alpha_2}(\Xi^2(t)) \cdots \quad V'_{\alpha_\nu}(\Xi^\nu(t)) \; \right]
\end{aligned}$$

where

$$V_{\alpha_j}(\Xi^j(t)) = \left[\alpha_{j,1}(\Xi^j(t)) \; \alpha_{j,2}(\Xi^j(t)) \; \cdots \; \alpha_{j,N_j}(\Xi^j(t)) \right]'$$

with $j = 1,...,\nu$. Considering the left-hand side of (11), from (12) we obtain

$$\begin{aligned}
&\left\| \nabla_w f_\alpha(\Xi_1(t);w) - \nabla_w f_\alpha(\Xi_2(t);w) \right\|_1 \\
&\qquad = \sum_{j=1}^{\nu} \sum_{k=1}^{N_j} \left| \alpha_{j,k}(\Xi_1^j(t)) - \alpha_{j,k}(\Xi_2^j(t)) \right| \\
&\qquad \leq \beta n_\xi \left\| \Xi_1(t) - \Xi_2(t) \right\|_1 = \beta n_\xi \left\| \Xi_1(t) - \Xi_2(t) \right\|_1
\end{aligned}$$

Noting that the right-hand side of (11) coincides with that of (10), by setting

$$C_3 = \lambda^{1-M} \beta n_\xi$$

we obtain

$$\beta n_\xi \left\| \Xi_1(t) - \Xi_2(t) \right\|_1 \leq C_3 \lambda^{M-1} \left\| \Xi_1(t) - \Xi_2(t) \right\|_1$$

which leads to the fulfilment of (11).

The existence of C_1, C_2, and C_3 implies that, for any choice of λ, $0 < \lambda < 1$, by choosing

$$C = \max(C_1, C_2, C_3)$$

conditions (9)-(11) are satisfied, which completes the proof. ■