

# EVALUATING THE POTENTIAL OF VOLTERRA-PARAFAC IIR MODELS

Phillip M. S. Burt<sup>1</sup> and José Henrique M. Goulart<sup>2</sup>

Universidade de São Paulo – EPUSP – PTC, São Paulo-SP, Brazil  
E-mails: {phillip, jgoulart}@lcs.poli.usp.br

## ABSTRACT

The Volterra-PARAFAC (VP) nonlinear system model, which consists of a FIR filterbank followed by a memoryless nonlinearity, aims at offering a good compromise between accuracy and parametric complexity. Here, for an even better compromise, we propose a generalization with IIR filters (VPI model) and evaluate both models. For the evaluation, we consider the concrete case of two audio loudspeakers and initially compute reference Volterra kernels from their known physical state-space models, using an efficient procedure. Then, VP and VPI models are derived and their accuracy is tested. As shown, the VPI models have in this case only 15 to 26 % of the parametric complexity of VP models with the same accuracy, which points to a great potential for accurate and efficient nonlinear system modeling.

**Index Terms**— Volterra-PARAFAC model, nonlinear system modeling, IIR filters

## 1. INTRODUCTION

Several nonlinear approaches to system modeling and identification have recently been proposed [1, 2, 3, 4, 5]. As has been shown by many authors, nonlinear models [6, 7, 8] are able to provide good approximations in several contexts in which linear ones are not, due to a significantly nonlinear behavior of some system of interest. Examples of this include acoustic echo cancellation [1], satellite channel equalization [2] and physiological process modeling [5].

Because of the difficulty in representing all kinds of nonlinear physical systems in an unified and useful manner, a wide range of formalisms and structures appears in the literature, such as NARMAX models [9], block-oriented structures [10] and Volterra filters [1, 4, 11]. The widely adopted Volterra filter (VF) model, in particular, is based on a continuous-time functional expansion employed by Wiener in his pioneering work on nonlinear system modeling [5, 6]. The set of VFs is fairly general, in the sense that they can approximate with arbitrary precision a wide class of physical systems (more specifically, systems with fading memory) [8], and have also the advantage of being feedforward and thus inherently stable. On the other hand, it is well known that VFs

are quite inefficient in terms of their parametric complexity and therefore are not appropriate in many circumstances.

A promising alternative that has recently been proposed is the Volterra-PARAFAC (VP) model [3]. It consists of a memoryless nonlinearity preceded by a linear FIR (finite impulse response) filterbank, which can be obtained from a conventional VF by means of a tensor decomposition approximation. This strategy retains many of the good features of VFs and may be able to model nonlinear systems accurately, with a significant reduction in parametric complexity.

To evaluate this potential for accurate and efficient modeling of the VP and other alternative models, typical physical models (PM) of systems of interest can be considered [3, 12, 13]. In particular, a precise quantitative assessment about model efficiency and accuracy can be made with a set of *reference Volterra kernels* computed from the PM. This computation is not trivial but is feasible with the use of the algorithm we proposed in [12], where this approach for evaluating alternative nonlinear structures is also discussed<sup>3</sup>.

In the above context, and considering that low-order IIR (infinite impulse response) filters often provide good models of linear physical systems, a question that naturally arises is whether a structure similar to the VP but containing a low-order IIR filterbank—which would result in even higher complexity reduction – has the potential to provide accurate models of nonlinear systems. This is the main problem addressed in this paper, which is organized as follows.

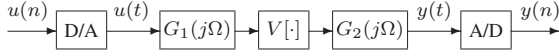
Section 2 describes the considered configuration and briefly reviews Volterra modeling and the method for computing reference kernels proposed in [12]. Section 3 reviews the Volterra-PARAFAC (VP) model and discusses its generalization using an IIR filterbank. Simulation results are presented in Section 4. Finally, Section 5 brings the concluding remarks and perspectives for future work.

## 2. REVIEW OF VOLTERRA MODELING

In the following, we consider the configuration shown in Fig. 1, which occurs in many applications, such as acoustic echo cancellation, active noise control and closed loop system linearization. The elements involved are an ideal

<sup>1</sup>This work was supported by FAPESP<sup>1</sup> and CNPq<sup>2</sup> (153549/2012-2).

<sup>3</sup>A more extensive article is in preparation.



**Fig. 1.** Configuration considered for the evaluation.

digital/analog converter, a reconstruction filter  $G_1(j\Omega)$ , the physical system of interest, which is denoted by the  $V[\cdot]$  operator and assumed to be time invariant, an anti-aliasing filter  $G_2(j\Omega)$  and a analog/digital converter.

## 2.1. Conventional Volterra modeling

Assuming that a VF of order  $P$  is used to model the resulting discrete-time structure of Fig. 1, its output,  $y(n)$ , is written as

$$y(n) = \sum_{p=1}^P y_p(n), \quad (1)$$

where  $y_p(n)$  is the  $p$ -th order component given by

$$y_p(n) = \sum_{k_1=0}^{N_p-1} \dots \sum_{k_p=k_{p-1}}^{N_p-1} v_p(k_1, \dots, k_p) \prod_{i=1}^p u(n - k_i), \quad (2)$$

in which  $u(n)$  is the input and  $v_p(k_1, \dots, k_p)$  is a causal Volterra kernel with memory extension (length) of  $N_p$  samples. It should be noted that the kernels involved in (2) are triangular [7], and therefore are allowed to have non-zero values only for time indexes  $k_1, \dots, k_p$  satisfying  $k_1 \leq \dots \leq k_p$ .

From a practical point of view, the model (1)–(2) has three desirable properties: it can approximate any fading memory system within arbitrary precision, as shown in [8]; it is by definition stable in the BIBO (bounded input, bounded output) sense, since (1) is a finite summation; and its output is clearly linear in the parameters  $v_p(k_1, \dots, k_p)$ , which permits, in principle, adapting them with standard algorithms like NLMS and RLS [4]. However, in many concrete situations, those advantages are overridden by the fact that a huge number of parameters is necessary to meet some prespecified precision level. Indeed, one can easily see that this number grows rapidly with  $N_2, \dots, N_P$  and  $P$ , rendering its use and adaptation unfeasible when a high order or a long memory is necessary.

## 2.2. Reference kernels computation

The method proposed in [12] permits assessing whether a VF can efficiently model a system of interest. Basically, reference kernels are computed from a known PM and used to construct some VFs with different structural characteristics. Then, by evaluating how accurately those VFs approximate the PM via a Monte Carlo experiment using typical inputs, one can estimate which values of  $P, N_1, \dots, N_P$  are sufficient to meet the desired precision. Note that, if the reference kernels were estimated from data produced by the PM, they would involve

uncertainties, while with the method described they incorporate only system characteristics. In the rest of this section, we succinctly review that method; for further details see [12].

The first step consists in applying the Carleman bilinearization (also called Carleman linearization [7]) to the PM, obtaining as a result a set of bilinear equations whose solution contains the exact components of orders 1 up to  $P$  of the Volterra series representation of the PM (which is, in general, of infinite order). For this step, [12] describes an efficient algorithm which determines each equation of that set recursively and does not involve redundant equations, unlike a direct implementation of the formulation of [7, Sec. 3.3]. Then, the first  $P$  kernels of that Volterra series representation can be calculated from the bilinear equations analytically in the time or in the frequency domains [7].

Since the reference kernels must represent the entire structure of Fig. 1, they must incorporate the effects of the filters  $G_1$  and  $G_2$ . Hence, for the second step, the simplest choice is to calculate the kernels of the PM in the frequency domain and then use the formulas for the interconnection of linear and nonlinear systems given in [7], through which the frequency domain kernels that model the input/output relation between the signals  $u(t)$  and  $y(t)$  of Fig. 1 may be expressed by

$$\tilde{H}_p(j\Omega_1, \dots, j\Omega_p) = \left( \prod_{q=1}^p G_1(j\Omega_q) \right) H_p(j\Omega_1, \dots, j\Omega_p) \times G_2(j\Omega_1 + \dots + j\Omega_p), \quad (3)$$

for  $1 \leq p \leq P$ , where  $H_p$  is the  $p$ -th order frequency domain kernel associated with the PM. Finally, assuming that a sampling frequency  $\Omega_s$  is appropriately chosen, the discrete-time reference  $p$ -th order kernel can be obtained by sampling  $\tilde{H}_p$  uniformly over the domain  $-\Omega_s/2 < \Omega_i \leq \Omega_s/2$  and then computing its  $p$ -dimensional IFFT of length  $N_p$  [12, 14].

## 3. VOLTERRA-PARAFAC IIR MODEL

The high parametric complexity shown by a VF in some circumstances severely hinders its use and estimation, due to the implied computational cost. In addition, it also degrades its performance when operating in an adaptive fashion, since the steady-state misadjustment increases. Therefore, several strategies have been proposed for alleviating such complexity by noting that, in many cases—and especially for physical systems—the essential information of a Volterra kernel can be captured by a structure of much smaller algebraic dimension.

A particularly effective strategy, which is based on tensor algebra concepts, was proposed in [3] and leads to the Volterra-PARAFAC (VP) model. In order to review this model, some notation must be introduced. A  $p$ -th order tensor is a numerical array of geometric dimension  $p$ , being denoted by a capital calligraphic letter as, e.g.,  $\mathcal{T} \in \mathbb{C}^{M_1 \times \dots \times M_p}$ . An element of  $\mathcal{T}$  is written as  $[\mathcal{T}]_{m_1, \dots, m_p}$ , where  $1 \leq m_i \leq M_i$ .

When all  $M_i = M$ ,  $\mathcal{T}$  is called cubical [15] and its space is denoted by  $\mathbb{C}^{M^{(p)}}$ . A cubical tensor is said to be symmetric if  $[\mathcal{T}]_{m_1, \dots, m_p}$  is invariant under a permutation of  $m_1, \dots, m_p$ .

### 3.1. Volterra-PARAFAC model

The VP model relies on the PARAFAC (PARAllel FACtor) decomposition [15], through which any symmetric tensor  $\mathcal{T} \in \mathbb{C}^{M^{(p)}}$  may have its elements uniquely written as [3]

$$[\mathcal{T}]_{m_1, \dots, m_p} = \sum_{r=1}^{R_p} [\mathbf{a}_{p,r}]_{m_1} [\mathbf{a}_{p,r}]_{m_2} \dots [\mathbf{a}_{p,r}]_{m_p}, \quad (4)$$

where the vectors  $\mathbf{a}_{p,r} \in \mathbb{C}^M$  are called parallel factors and  $R_p$  is the symmetric rank of  $\mathcal{T}$  [3, 15]. Note that (4) is a generalization of the decomposition of a symmetric matrix into a sum of rank-1 matrices. To the extent of our knowledge, no algorithm capable of both determining  $R_p$  and computing the decomposition for  $p \geq 3$  is known, but there are practical alternatives, such as the simple ALS [16]. This algorithm can, in many cases, find a good approximation of the form (4) for a given  $R_p$  by seeking the minimization of the Frobenius norm of the error, although its convergence to a stationary point of such cost function is not guaranteed [16].

Now, by observing that any discrete-time finite Volterra kernel  $v_p$  with length  $N_p$  admits an equivalent symmetric form [3, 7] (in the sense that they produce identical components  $y_p(n)$  for a given  $u(n)$ ) which may be viewed as a symmetric tensor  $\mathcal{V}_p \in \mathbb{C}^{N_p^{(p)}}$ , we can rewrite (2) as

$$y_p(n) = \sum_{k_1=0}^{N_p-1} \dots \sum_{k_p=0}^{N_p-1} [\mathcal{V}_p]_{k_1+1, \dots, k_p+1} \prod_{i=1}^p u(n - k_i), \quad (5)$$

and then apply the decomposition (4), which yields [3]

$$y_p(n) = \sum_{r=1}^{R_p} [\mathbf{a}_{p,r}^T \mathbf{u}^{(p)}(n)]^p, \quad (6)$$

where  $\mathbf{u}^{(p)}(n) = [u(n) \dots u(n - N_p + 1)]^T$  and the vectors  $\mathbf{a}_{p,r}$  are the parallel factors of  $\mathcal{V}_p$ . Structurally, this relation can be represented as shown in Fig. 2. As we can see, the resulting structure is essentially a Wiener model [5] with a FIR filterbank followed by a memoryless nonlinearity with no cross-terms. Although the output it is no longer linear in the parameters, the interest in this structure comes from the fact that, for a Volterra kernel with small  $R_p$ , it is very compact: it has  $N_p R_p$  parameters, while a triangular  $p$ -th order Volterra kernel has  $(N_p + p - 1)! / [p!(N_p - 1)!]$  parameters [3].

### 3.2. An IIR generalization

A direct generalization of the VP model is the *Volterra-PARAFAC IIR* (VPI) model, where an IIR filterbank is used

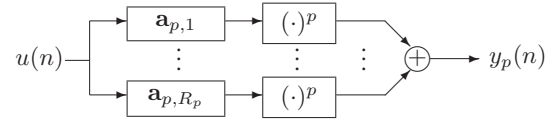


Fig. 2.  $p$ -th order subsystem of a Volterra-PARAFAC model.

instead of a FIR one. This is motivated by the expectation that a similarity exists with the linear system case, where a low-order IIR model provides, in many cases, an efficient discrete-time representation of the dynamics of some system. Our goal here is to assess if the VPI model has the potential to fulfill this expectation for nonlinear systems of interest.

To this end, a connection could be sought with the analytical discretization discussed in [5, Sec. 3.5]. However, this is a hard problem in general and, to the best of our knowledge, no simple and general method to tackle it exists.

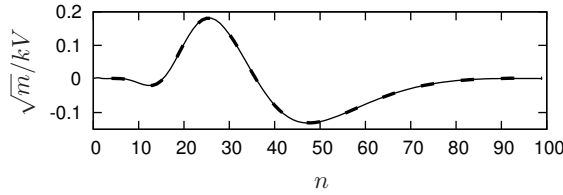
Our approach is rather to compute the Volterra kernels from the PM, as described in Section 2.2, and then obtain the IIR filters of the VPI model from the kernels. This second step, ideally, would require an extension of the ALS algorithm (for instance) to the IIR case, in order to jointly determine the filterbank. At least for an initial assessment, however, it is sufficient to obtain the IIR filters individually from the FIR filters of a VP model, by using a standard order reduction procedure [17], [18, p. 153].

## 4. PERFORMANCE EVALUATION

We consider now the application of the VP and VPI models to two loudspeakers for which accurate PMs have been previously derived in the literature. Note that coping with the nonlinear behavior of these devices can be crucial in applications such as acoustic echo cancellation and active noise control. Loudspeaker A is a small (82 mm in diameter) 4Ω bass automotive loudspeaker [19] and loudspeaker B is a larger (130 mm in diameter) 4Ω bass loudspeaker [20].

### 4.1. Model derivation

The basic procedure to obtain the models is as follows. 1) Using the method described in Sec. 2.2, with a sampling rate of 5 kHz and length  $N_p = 100$  for all  $p$ ,  $P = 4$  Volterra kernels are computed, providing an initial underlying VF model. 2) With this VF and given the number of factors  $R_p$ , the ALS algorithm is used to obtain an ancillary VP model. 3) The FIR filters of this model are approximated by IIR filters, providing the VPI model. As exemplified in Fig. 3 for  $p = 2$  and  $r = 2$ , if the FIR impulse responses  $\mathbf{a}_{p,r}$  are smooth they can frequently be approximated by IIR filters with orders  $M_{p,r} \ll N_p = 100$ . This is done by obtaining a balanced realization and then eliminating the state elements associated with the smaller Hankel singular values. Such approximation is favored when the FIR impulse responses have



**Fig. 3.** Example of impulse response of VP FIR factor (---) and order 8 IIR approximation (—).

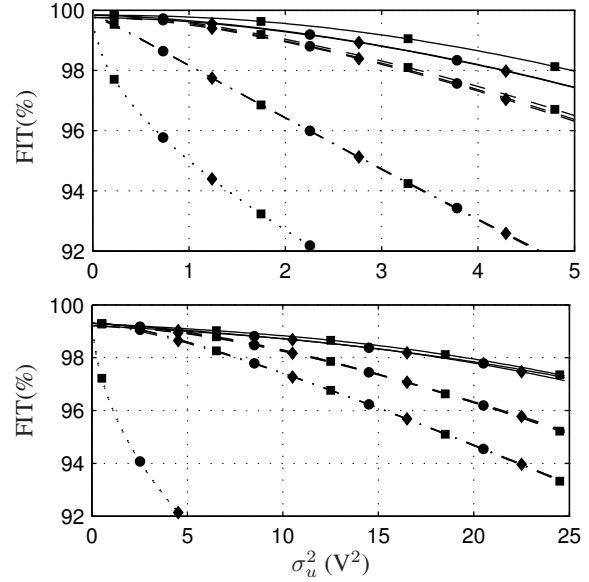
decayed to almost zero after  $N_p$  samples, as in Fig. 3. 4) This same behavior suggests that a more efficient VP model with  $N_p < 100$  is possible, which is obtained by applying the ALS algorithm to truncated versions of the Volterra kernels.

#### 4.2. Comparison of VP and VPI models

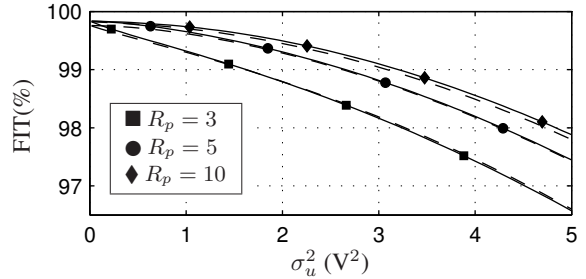
Vectors  $\mathbf{y}$  of 500 output samples of a PM that comprises the loudspeaker and filters  $G_1$  and  $G_2$  (see Fig. 1) are obtained by numerical integration, using as inputs zero mean Gaussian signals with 625 Hz bandwidth and variable power  $\sigma_u^2$ . The sample fit of a realization is defined as  $\text{FIT}(\hat{\mathbf{y}}) = 100 \times (1 - \|\mathbf{y} - \hat{\mathbf{y}}\| / \|\mathbf{y} - \mu_{\mathbf{y}}\|)$ , where  $\mu_{\mathbf{y}}$  is the sample mean and  $\hat{\mathbf{y}}$  is the output sample vector produced by a given model.

Initially, models VF with  $N_p = 100$  for all  $p$  and models VP and VPI with  $R_p = 5$  for all  $p$ , whose lengths and orders are given in Table 1, are compared. In Table 1 it can be seen that for loudspeaker A and  $R_p = 5$ , for instance, the VP model has lengths  $N_1 = 100$  and  $N_p = 80$  for  $p > 1$  and the VPI model has orders  $M_{1,1} = 7$ ,  $M_{2,1} = 6$ ,  $M_{2,2} = M_{2,3} = 7$ ,  $M_{2,4} = 8$ ,  $M_{2,5} = 9$  and  $M_{p,r} = 8$  for  $p > 2$  and all  $r$ . The average fit over 100 realizations of the input, as a function of input power  $\sigma_u^2$  and for model orders  $P = 1, \dots, 4$ , is shown in Fig. 4. As expected, as  $\sigma_u^2$  grows, higher order models are required to accurately represent the loudspeakers. As also expected, the larger loudspeaker B admits higher input power before its nonlinear behavior becomes important. The FIT of models VP and VPI are almost identical, but for  $P = 4$  the total parameter count of VPI is about only 0.20 times that of VP for both loudspeakers, as can be seen in Table 1.

Now, Fig. 5 shows a comparison of VP and VPI models



**Fig. 4.** Loudspeakers A (top) and B (bottom): FIT of VF (■), VP (●) and VPI (◆) for  $P = 1$  (···), 2 (— · —), 3 (— — —) and 4 (—).



**Fig. 5.** Loudspeaker A: FIT of VP (—) and VPI (---).

for loudspeaker A, with  $P = 4$  and  $R_p = 3, 5$  and 10. Almost identical FITs can be obtained in all cases, but, as can be seen in Table 1, their total parameter count relation rises from 0.15 to 0.26 with  $R_p$ , reflecting the accumulation of the approximation errors of the not jointly designed IIR filters.

#### 5. CONCLUDING REMARKS

In this contribution, we considered a generalized version of the VP model in which the filterbank is IIR and described a relatively simple procedure to evaluate the performance of the resulting model when a typical PM of some system of interest is known *a priori*. The simulation results presented show that such model may lead to an excellent compromise between precision and parametric complexity. As a continuation of this work, we intend to develop a procedure for a joint design of the IIR factors which compose the VPI model.

**Table 1.** Model lengths/orders, total parameter counts and parameter reduction factor ( $p = 1, 2, 3, 4; r = 1, 2, \dots, R_p$ ).

| Loud.  | $R_p^\dagger$ | VP           |            | VPI         |            | $\frac{\Sigma_I}{\Sigma_F}$ |
|--------|---------------|--------------|------------|-------------|------------|-----------------------------|
|        |               | $N_p$        | $\Sigma_F$ | $M_{p,r}$   | $\Sigma_I$ |                             |
| A [19] | 3             | 100/80/80/80 | 820        | 7/6/6/5     | 126        | 0.15                        |
|        | 5             | 100/80/80/80 | 1300       | 7/*/8/8     | 264        | 0.20                        |
|        | 10            | 100/80/80/80 | 2500       | 10/10/10/10 | 651        | 0.26                        |
| B [20] | 5             | 100/90/90/90 | 1450       | 8/‡/§/8     | 266        | 0.18                        |

$^\dagger R_1 = 1 * M_{2,r} = \{6, 7, 8, 9\}$   $^\ddagger M_{2,r} = \{6, 6, 7, 9, 10\}$   $^\S M_{3,r} = \{7, 7, 7, 9, 9\}$   
 $\Sigma_F = \sum_{p=1}^P R_p N_p$   $\Sigma_I = \sum_{p=1}^P \sum_{r=1}^{R_p} (2M_{p,r} + 1)$

## 6. REFERENCES

- [1] L. A. Azpicueta-Ruiz, M. Zeller, A. R. Figueiras-Vidal, J. Arenas-García, and W. Kellermann, "Adaptive Combination of Volterra Kernels and its Application to Nonlinear Acoustic Echo Cancellation," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 19, no. 1, pp. 97–110, January 2011.
- [2] B. F. Beidas, "Intermodulation Distortion in Multi-carrier Satellite Systems: Analysis and Turbo Volterra Equalization," *IEEE Transactions on Communications*, vol. 59, no. 6, pp. 1580–1590, June 2011.
- [3] G. Favier, A. Y. Kibangou, and T. Bouilloc, "Nonlinear system modeling and identification using Volterra-PARAFAC models," *International Journal of Adaptive Control and Signal Processing*, vol. 26, no. 1, pp. 30–53, January 2012.
- [4] M. Zeller and W. Kellermann, "Fast and Robust Adaptation of DFT-Domain Volterra Filters in Diagonal Coordinates Using Iterated Coefficient Updates," *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1589–1604, March 2010.
- [5] V. Marmarelis, *Nonlinear Dynamic Modeling of Physiological Systems*, John Wiley & Sons, Hoboken, NJ, USA, 2004.
- [6] N. Wiener, *Nonlinear problems in random theory*, Massachusetts Institute of Technology, 1958.
- [7] W. J. Rugh, *Nonlinear System Theory: The Volterra/Wiener Approach*, Johns Hopkins University Press, Baltimore, MD, 1981.
- [8] S. Boyd and L. Chua, "Fading memory and the problem of approximating nonlinear operators with Volterra series," *IEEE Transactions on Circuits and Systems*, vol. CAS-32, no. 11, pp. 1150–1161, November 1985.
- [9] H.-K. Jang and K.-J. Kim, "Identification of Loudspeaker Nonlinearities Using the NARMAX Modeling Technique," *Journal of the Audio Engineering Society*, vol. 42, no. 1/2, pp. 50–59, February 1994.
- [10] A. Y. Kibangou and G. Favier, "Tensor analysis-based model structure determination and parameter estimation for block-oriented nonlinear systems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 3, pp. 514–525, June 2010.
- [11] V. J. Mathews and G. L. Sicuranza, *Polynomial signal processing*, Wiley, 2000.
- [12] J. H. M. Goulart and P. M. S. Burt, "Efficient kernel computation for Volterra filter structure evaluation," *IEEE Signal Processing Letters*, vol. 19, no. 3, pp. 135–138, March 2012.
- [13] G. D. Mitsis, M. G. Markakis, and V. Z. Marmarelis, "Nonlinear Modeling of the Dynamic Effects of Infused Insulin on Glucose: Comparison of Compartmental With Volterra Models," *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 10, pp. 2347–2358, October 2009.
- [14] H. Koepl and D. Schwingshackl, "Comparison of discrete-time approximations for continuous-time nonlinear systems," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, 2004 (ICASSP '04)*, May 2004, vol. 2, pp. II–881–II–884.
- [15] P. Comon, G. Golub, L.-H. Lim, and B. Mourrain, "Symmetric tensors and symmetric tensor rank," *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 3, pp. 1254–1279, 2008.
- [16] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.
- [17] B. Moore, "Principal component analysis in linear systems: Controllability, observability, and model reduction," *IEEE Transactions on Automatic Control*, vol. 26, no. 1, pp. 17–32, February 1981.
- [18] P. A. Regalia, *Adaptive IIR Filtering in Signal Processing and Control*, Marcel Dekker, 1995.
- [19] D. Franken, K. Meerkotter, and J. Waßmuth, "Passive parametric modeling of dynamic loudspeakers," *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 8, pp. 885–891, November 2001.
- [20] D. Franken, K. Meerkotter, and J. Waßmuth, "Observer-based feedback linearization of dynamic loudspeakers with Ac amplifiers," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 2, pp. 233–242, March 2005.