Approximation of generalized frequency response functions via vector fitting

Antonio Carlucci, Ion Victor Gosea, and Stefano Grivet-Talocia

Approximation of generalized frequency response functions via vector fitting

Abstract: This paper describes a method for data-driven approximation of nonlinear systems, using the notion of generalized frequency response functions as provided by the Volterra series theory. Using rational approximation, frequency-domain input–output data are fitted to a bilinear model structure. The proposed algorithm performs optimization of model coefficients through a greedy approach that relies on linear least-squares solves only, thus ensuring speed and scalability.

Keywords: Volterra series, bilinear systems, vector fitting algorithm, generalized frequency response functions, least-squares fitting, Burgers' equation

MSC 2020: 41A20, 93C10, 93C80

1 Introduction and background

The Volterra series theory for nonlinear systems [21, 20] is a modeling approach based on expanding a given time-invariant input–output map $y(t) = G[u(t)]$ using a series of generalized time-domain convolutions as

$$y(t) = \sum_{m=1}^{\infty} \int_0^t h_m(\tau_1, \ldots, \tau_m) u(t-\tau_1) \cdots u(t-\tau_m) \mathrm{d}\tau_1 \ldots \mathrm{d}\tau_m, \tag{11.1}$$

where $u(t)$ is the system's input (vanishing for $t < 0$), while $y(t)$ is the output. This paper focuses on single-input, single-output (SISO) systems, that is, $u(t), y(t) \in \mathbb{R}$ are scalars. In (11.1), the output is decomposed as a sum of contributions $y_m(t)$ from *homogeneous* subsystems, whose response is given as a multidimensional convolution with the one-sided kernel $h_m(\tau_1, \ldots, \tau_m)$. This series expansion is typically used to model mildly nonlinear systems since it has local validity around an operating point (in this case, $u(t) = 0$) and it is ensured to be convergent if the amplitude of $u(t)$ is sufficiently small (see [20, 5] for details). Equation (11.1) is not enough to uniquely define the kernels h_m, since different choices can describe the same input–output relation. Requiring that the h_m's are symmetric, that is, invariant under arbitrary permutations of the arguments, leads to

Antonio Carlucci, **Stefano Grivet-Talocia,** Department of Electronics and Telecommunications, Politecnico di Torino, 24 Corso Duca degli Abruzzi, 10129 Turin, Italy, e-mails: antonio.carlucci@polito.it, stefano.grivet@polito.it

Ion Victor Gosea, Max Planck Institute for Dynamics of Complex Technical Systems, CSC group, Sandtorstraße 1, 39106 Magdeburg, Germany, e-mail: gosea@mpi-magdeburg.mpg.de

https://doi.org/10.1515/9783111376776-011

a unique definition that is particularly convenient because they can be easily obtained analytically and measured from simulations. This formalism is valuable from a modeling standpoint because it generalizes the well-known convolution formula of linear systems, and allows defining generalized frequency response functions (GFRFs) through the multivariate Laplace transform [10]

$$H_m(s_1,\dots,s_m) = \int_0^\infty \cdots \int_0^\infty h_m(\tau_1,\dots,\tau_m) e^{-\sum_{i=1}^m s_i\tau_i} \mathrm{d}\tau_1 \dots \mathrm{d}\tau_m \tag{11.2}$$

where $h_m(\tau_1,\dots,\tau_m)$ is the symmetric kernel and $H_m(s_1,\dots,s_m)$ is also called the degree-m symmetric transfer function.

The objective of this paper is to derive a dynamical model starting from sampled data of symmetric transfer functions $H_m(s_1,\dots,s_m)$. To this aim, a particularly convenient choice of the model structure is that of bilinear systems. These can be represented with the following state-space form:

$$\dot{x}(t) = Ax(t) + Nx(t)u(t) + Bu(t), \quad y(t) = Cx(t). \tag{11.3}$$

Such systems can be also viewed as time-varying linear systems, by rewriting the state equation as $\dot{x}(t) = \overline{A}(t)x(t) + Bu(t)$ with $\overline{A}(t) = A + Nu(t)$. Therefore, stability and other relevant properties can be deduced from the already existing theory for linear systems, and hence, the bilinear structure has proven very useful and tractable for modeling. It is also sufficiently general since a broad class of systems with analytic nonlinearities can be embedded into a bilinear representation, for example, through the Carleman linearization approach [7] as in [6]. Additionally, by imposing specific boundary conditions for PDEs as in [3], bilinear models can be derived after semidiscretization in the spatial domain. For bilinear systems, several model reduction (MOR) techniques have been described as an immediate extension of their linear counterpart. Intrusive methods for MOR of bilinear systems, mostly based on moment matching (rational Krylov approaches) or balancing, were proposed in [6, 3] (see also the references therein), together with optimality-enforced methods and matching of infinite series [23, 2, 13]. Then data-driven approaches were also proposed, starting with a direct extension of the Loewner framework to bilinear systems in [1], and with more recent work in [16] that uses time-domain data to infer values of (symmetric) transfer functions; the latter was extended to quadratic systems in [17]. Subspace identification techniques for bilinear systems were proposed in [12], while the problem of structure-preserving MOR was treated in [4]. This paper proposes a novel approach toward data-driven approximation of GFRFs based on rational fitting. We provide an extension of the vector fitting algorithm to the nonlinear setting, featuring a greedy strategy whereby a sequence of linear least-squares problems is solved to optimize the coefficients of a conveniently chosen bilinear model structure.

This paper is organized as follows. The notation and the model structure are provided in Section 2. The core ideas for fitting GFRFs are introduced in Section 2.1, followed

by a discussion on the relation between the fitting error and the input–output response in Section 2.2. Then algorithmic tools for rational approximation based on Vector Fitting (VF) are reviewed and adapted in Section 2.3. Two numerical experiments are studied and the results are reported in Section 3, before concluding with Section 4.

2 Formulation

In order to introduce the problem data, let us use the shorthand $\mathbf{s}_m \triangleq (s_1, \dots, s_m)$ to denote a point in the m-dimensional frequency space. The starting point for model inference is thus a data set $\mathcal{H} = \{\mathcal{H}_m\}_{m=1}^{M}$ with evaluations of degree-m transfer functions at an arbitrary set of points

$$\mathcal{H}_m = \{(\mathbf{s}_m^{(k)}, \check{H}_m^{(k)}),\ k = 1, \dots, K_m\}, \quad m = 1, \dots, M. \tag{11.4}$$

This data is fitted with a predefined model structure corresponding to a particular bilinear dynamical system G with the following form:

$$G: \begin{cases} \dot{x}_1(t) = A_1 x_1(t) + B_1 u(t), \\ \dot{x}_m(t) = A_m x_m(t) + N_m x_{m-1}(t) u(t), \quad 2 \le m \le M, \\ y(t) = \sum_{m=1}^{M} C_m x_m(t). \end{cases} \tag{11.5}$$

This was introduced in [20] as a way to construct bilinear realizations of Volterra transfer functions. Its structure is such that all kernels beyond the first M are zero, implying that G is a *polynomial* system of degree M (i. e., the sum in (11.1) contains M terms). In addition, state equations corresponding to every individual m in (11.5) correspond to the mth degree *homogeneous subsystem* of G [20], that is, the mth term in (11.1). The system (11.5) can be interpreted as a cascade of linear systems with a multiplicative nonlinearity in between. Denoting with $H_m^{(k)}$ the symmetric transfer functions of G evaluated at $\mathbf{s}_m^{(k)}$, the model coefficients should be optimized so that H_m is close to $\check{H}_m$, that is, $H_m^{(k)} \approx \check{H}_m^{(k)}$.

It is to be mentioned that G in (11.5) could be recast into a standard bilinear system format as in (11.3). The way to do this is as suggested by the realization procedure in [20], that is, by putting together the bilinear system matrices (A, B, C, N), as follows:

$$A = \operatorname{blkdiag}\{A_1, \dots, A_M\}, \quad B = \operatorname{col}\{B_1, 0, \dots, 0\},$$
$$N = \begin{pmatrix} 0 & & & \\ N_2 & 0 & & \\ & N_3 & 0 & \\ & & \ddots & \ddots \end{pmatrix}, \quad C = \begin{pmatrix} C_1 & \cdots & C_M \end{pmatrix}. \tag{11.6}$$

Different from [16], where the matrix N of the fitted bilinear system is computed to match the realization (A, B, C) computed with the Loewner framework in the first step, here we construct individual submodels (A_m, N_m, C_m) at each step $m \geq 2$. If necessary, these submodels may be assembled into a standard bilinear system, by following (11.6).

2.1 Fitting of symmetric transfer functions

The symmetric transfer functions of (11.5) can be expressed recursively. Starting from $m = 1$, let us define the auxiliary functions

$$X_1(s_1) = (s_1 I - A_1)^{-1} B_1, \tag{11.7}$$

$$X_m(\mathbf{s}_m) = \left[(s_1 + \cdots + s_m)I - A_m\right]^{-1} N_m Q_m(\mathbf{s}_m), \quad m \geq 2, \tag{11.8}$$

$$Q_m(\mathbf{s}_m) = \sum_{i=1}^{M} X_{m-1}(\mathbf{s}_m \backslash s_i), \tag{11.9}$$

with $\mathbf{s}_m \backslash s_i \triangleq (s_1, \ldots, s_{i-1}, s_{i+1}, \ldots, s_m)$. The symmetric transfer functions become $H_m(\mathbf{s}_m) = C_m X_m(\mathbf{s}_m)$, with the explicit formulation below:

$$H_m(\mathbf{s}_m) = \begin{cases} C_1(s_1 I - A_1)^{-1} B_1, & m = 1, \\ \frac{1}{m!} C_m[(s_1 + \cdots + s_m)I - A_m]^{-1} N_m Q_m(\mathbf{s}_m), & m \geq 2. \end{cases} \tag{11.10}$$

Expressions (11.9)–(11.10) can be easily derived for the assumed system structure (11.5) by applying the so-called growing exponential approach; see [20].

In principle, the approximation problem we need to solve is to find model parameters so that the following cost function is minimized:

$$J = \sum_{m=1}^{M} \sum_{k=1}^{K_m} \left| H_m(s_m^{(k)}) - \check{H}_m(s_m^{(k)}) \right|^2. \tag{11.11}$$

Our approach is to greedily optimize individual terms of the summation corresponding to each m. Starting from $m = 1$, we consider the approximation

$$C_1(s_1^{(k)} I - A_1)^{-1} B_1 \approx \check{H}_1(s_1^{(k)}), \quad k = 1, \ldots, K_1, \tag{11.12}$$

that is equivalent to fitting a univariate rational function to given data, a problem known as *rational fitting* and efficiently solvable using existing algorithms, such as AAA [19] or vector fitting [15].

As for $m > 1$, the recursive expression in (11.10) clearly shows the main idea of our formulation, based on observing that the mth degree transfer function is the product of a univariate transfer function

$$F_m(s) = \frac{1}{m!} C_m (sI - A_m)^{-1} N_m, \tag{11.13}$$

evaluated at $s = s_1 + \cdots + s_m$ with the function $Q_m(\mathbf{s}_m)$, which only depends on lower-degree subsystems. This implies that, if the system coefficients $A_1, \ldots, A_m$ and $B_1, N_2, \ldots, N_m$ are known up to index m, the quantity $Q_m(\mathbf{s}_m)$ is entirely determined. Therefore, the approximation problem is reduced to optimizing the coefficients of $F_m(s)$ for each higher-degree index $m = 2, \ldots, M$ so that

$$F_m(s_1^{(k)} + \cdots + s_m^{(k)})Q(\mathbf{s}_m^{(k)}) \approx \check{H}_m(\mathbf{s}_m^{(k)}), \quad k = 1, \ldots, K_m, \tag{11.14}$$

which can be tackled again with rational fitting algorithms. The greedy aspect of this method lies in considering $Q(\mathbf{s}_m^{(k)})$ a given constant as resulting from optimization of previous terms of the summation (11.11), rather than jointly optimizing all system parameters in (11.5) simultaneously.

2.2 Cost function and model error

The motivation for using J in (11.11) as a measure of model mismatch is that it works as a proxy for the H_2 norm of the error system. In particular, using the notation $\check{h}_m(\tau_1, \ldots, \tau_n)$ for the *true* Volterra kernels of the original system and $h_m(\tau_1, \ldots, \tau_n)$ for the fitted degree-M model, we can define the error $e_m(\tau_1, \ldots, \tau_n) = h_m(\tau_1, \ldots, \tau_n) - \check{h}_m(\tau_1, \ldots, \tau_n)$. Following the definition in [23, 13], the H_2 norm of the error system E would be

$$\|E\|_{H_2} = \sqrt{\sum_{m=1}^{\infty} \|e_m\|_2^2}, \quad \|e_m\|_2^2 = \int_0^{\infty} \cdots \int_0^{\infty} e_m(\tau_1, \ldots, \tau_m)^2 d\tau_1 \cdots d\tau_m. \tag{11.15}$$

Considering the magnitude of the output error $e(t) = \check{y}(t) - y(t)$,

$$|e(t)| \leq \sqrt{\sum_{m=1}^{M} \|e_m\|_2^2} \sqrt{\sum_{m=1}^{M} \|u\|_2^{2m}} + \left| \sum_{m=M+1}^{\infty} \check{y}_m(t) \right|. \tag{11.16}$$

This estimate shows that the error contains two components, the first one depends on the modeling error of the first M kernels, while the second is the unmodeled nonlinearity of degree higher than M. The connection with the cost function J is made through Plancherel's theorem to express $\|e_m\|_2^2$ in terms of the H_2-norm of $E_m(\mathbf{s}_m) = H_m(\mathbf{s}_m) - \check{H}_m(\mathbf{s}_m)$,

$$\sum_{m=1}^{M} \|e_m\|_2^2 = \sum_{m=1}^{M} \frac{1}{(2\pi)^m} \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \left| E_m(j\omega_1, \ldots, j\omega_m) \right|^2 d\omega_1 \ldots d\omega_m, \tag{11.17}$$

and J can be viewed as a discrete analogue of the frequency-domain integrals.

As for the choice of M, Equation (11.16) suggests that it should be chosen as the smallest degree for which the error due to the higher-degree responses ($m \geq M+1$) are cumu-

latively negligible compared to the modeling error $\sum_{m=1}^{M} \|e_m\|^2$ arising from the fitting procedure.

2.3 Weighted vector fitting for GFRFs

This section completes the formulation by describing the main enabling tool to solve the approximation problems (11.14) and (11.12), that is, the weighted vector fitting algorithm. In particular, (11.12) can be tackled with the standard formulation of VF [15, 11, 14]. On the other hand, (11.14) requires incorporating the arbitrary but known weighting vectors $Q(\mathbf{s}_m^{(k)})$ using a modified formulation such as [8], here reviewed for the specific problem at hand. First of all, consider problem (11.14) for a particular m, restated as

$$\text{minimize} \sum_{k=1}^{K} \left| F_m(s_1^{(k)} + \cdots + s_m^{(k)}) Q(\mathbf{s}_m^{(k)}) - \check{H}_m^{(k)} \right|^2, \tag{11.18}$$

where $F_m(s)$ is unknown and represented in pole-residue form as

$$F_m(s) = \sum_{i=1}^{\nu} \frac{R_i}{s - p_i}, \tag{11.19}$$

with the index m being omitted in both the R_i's and also the p_i's. Note that this pole-residue expansion is another way of parameterizing $F_m(s)$ through coefficients R_i and p_i, instead of the coefficients C_m, A_m, N_m used in (11.13). However, it is always possible to write $F_m(s)$ in the form (11.13) starting from the pole-residue (11.19), through a *realization* procedure (see [14]).

Going back to the optimization problem in (11.18), a key observation is that if p_i are given constants, (11.18) is an easily solvable linear least-squares (LS) problem in the variables R_i. However, if p_i are to be optimized, this is no longer true and the problem becomes much harder. For this reason, VF goes through a preliminary pole identification phase to find poles p_i through an iterated solution of a simplified problem. In this stage, a double barycentric form is used to represent $F_m(s)$ as

$$F_m(s) = \frac{N(s)}{d(s)}, \quad N(s) = \sum_{i=1}^{\nu} \frac{N_i}{s - q_i}, \quad d(s) = 1 + \sum_{i=1}^{\nu} \frac{d_i}{s - q_i}, \tag{11.20}$$

using a set of auxiliary poles $\{q_i\}_{i=1}^{\nu}$. Starting from an initial guess of q_i, this is updated through several iterations where the following modified cost function is minimized:

$$\min \sum_{k=1}^{K} \left| N(s_1^{(k)} + \cdots + s_m^{(k)}) Q(\mathbf{s}_m^{(k)}) - \check{H}_m^{(k)} d(s_1^{(k)} + \cdots + s_m^{(k)}) \right|^2. \tag{11.21}$$

Solving (11.21) at a given iteration yields coefficients d_i of $d(s)$, that are used to update the auxiliary poles for the next iteration through the rule $q_i \leftarrow \text{zeros}\{d(s)\}$. In this way,

(11.21) is optimized repeatedly with an updated choice of q_i each time. Upon convergence or after a maximum number of iterations, the q_i's are chosen to be the model poles p_i in (11.19). Finally, (11.18) and (11.19) with fixed p_i are solved via linear LS to find R_i.

Convergence to a given set of poles q_i corresponds to the condition $d(s) \to 1$, detected either by checking whether the norm of coefficients $\sum_{i=1}^{\nu} d_i^2$ falls below a predefined tolerance, or by looking at the maximum deviation $|d(s) - 1|_\infty$. Convergence of VF is discussed in [18], where it is shown that the algorithm might not converge in a strong sense. Nonetheless, VF still produces good solutions to the rational approximation problem, for which a globally optimal solution is hard to obtain due to the strongly nonconvex nature of the related optimization problem. In fact, thanks to its robustness and reliability, VF has become the method of choice in design automation tools and flows available on the market, in particular for electronic system design applications.

The proposed algorithm for symmetric transfer function fitting is summarized as the pseudocode in Algorithm 11.1.

Algorithm 11.1 Approximation of GFRFs.

Require: M, K_m, ν and the data set $\mathcal{H}$ as defined in (11.4)
  Fit rational function $H_1(s)$ in pole-residue form to data $\check{H}_1(s_1^{(k)})$ using VF
  Turn pole-residue form of $H_1(s)$ into a realization (A_1, B_1, C_1)
  Compute $Q_2(\mathbf{s}_2^{(k)})$ for $k = 1, \ldots, K_2$, as defined in (11.9)
  for $m = 2, \ldots, M$ **do**
    Fit $F_m(s)$ in pole-residue form by solving (11.18) using weighted VF
    Turn pole-residue form of $F_m(s)$ into a realization (A_m, N_m, C_m)
    Compute $Q_{m+1}(\mathbf{s}_{m+1}^{(k)})$ for $k = 1, \ldots, K_{m+1}$ as defined in (11.9)
  end for

3 Numerical examples and discussion

3.1 Viscous Burgers' equation

This section reports on the example of the viscous Burgers' equation

$$\frac{\partial}{\partial t} v(x,t) - \nu \frac{\partial^2}{\partial x^2} v(x,t) + v(x,t) \frac{\partial}{\partial x} v(x,t) = 0, \tag{11.22}$$

describing the velocity $v(x,t)$ of a fluid in one spatial coordinate. The viscosity parameter was set to $\nu = 0.02$. By following [6], the PDE was semi-discretized in space through a finite difference scheme with 200 points. The resulting nonlinear dynamical system describing the temporal evolution was formulated as a quadratic-bilinear system of size $q = 200$.

A data-driven bilinear model with degree $M = 2$ could be constructed by sampling the first- and second-degree transfer functions $H_1(s_1)$ and $H_2(s_1, s_2)$. In particular, $H_1(s_1)$ was sampled on the imaginary axis at $K_1 = 200$ points, log-spaced and corresponding to the frequency band $[10^{-5}, 800]$ Hz. The function $H_2(s_1, s_2)$ was sampled at $K_2 = 4964$ points, with both s_1 and s_2 purely imaginary and in the frequency interval $[10^{-5}, 400]$ Hz. Running the proposed algorithm with $\nu = 8$ poles, 40 iterations, and $M = 2$ to approximate H_1 and H_2 leads to a fitted bilinear model with overall order 13. Frequency-domain accuracy is reported in Figure 11.1, where the left panel shows the relative error with respect to both s_1 and s_2, and the right panel directly compares the responses.

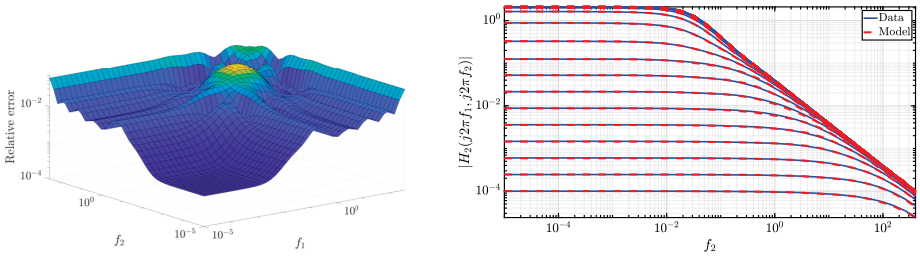

Figure 11.1: Error analysis for the example in Section 3.1. *Left panel*: Relative error on H_2 for both s_1 and s_2 on the imaginary axis. *Right panel*: Model-data comparison for $H_2(j2\pi f_1, j2\pi f_2)$ with respect to f_2 and for several fixed values of f_1.

In the time domain, the responses of the full-order model are compared with the fitted one in Figure 11.2 (left panel). In this experiment, $t \in [0, 10]$ s and the input is an amplitude-modulated chirp $u(t) = 0.1\cos[2\pi f_0(t)t][1 - 1/2\cos(2\pi 2t)]$, with $f_0(t)$ sweeping from 1 mHz to 10 Hz, in order to test accuracy over a broad frequency range. In the right panel of Figure 11.2, we also show that the system nonlinearity is being modeled correctly because adding the degree-2 subsystem leads to a substantial error reduction compared to a simple degree-1 approximation (where only H_1 is fitted).

3.2 Nonlinear RC ladder

This example is an electrical network with nonlinear elements, originally introduced in [9]. In particular, it is an RC ladder circuit with diodes in parallel to resistors (see Figure 11.3). The input $u(t)$ is a current source, while the output $y(t)$ is the voltage at the first node.

Diodes have an exponential characteristic so that the original state equation contains nonpolynomial nonlinearities. Therefore, starting from a model with $q = 50$ states, Carleman bilinearization was used to obtain a lifted bilinear model with $q_{\text{bil}} = 50^2 + 50$.

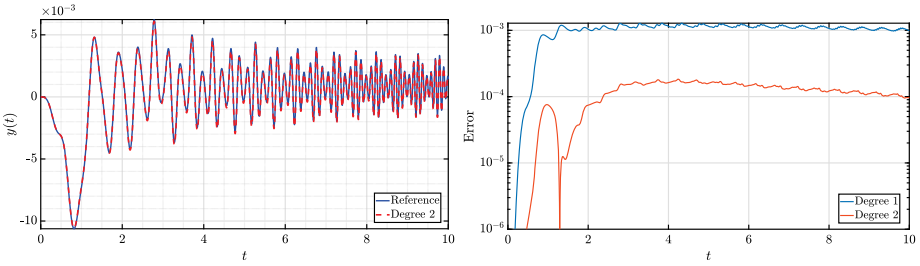

Figure 11.2: Time-domain solution of the example in Section 3.1. The *left panel* compares the full-order model response with that of the fitted model (of degree 2). The *right panel* reports the instantaneous error for two models with $M = 1, 2$, showing that the nonlinear ($M = 2$) model is more accurate than the linear approximation ($M = 1$).

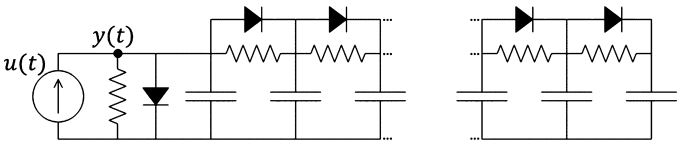

Figure 11.3: Nonlinear electrical network of the example in Section 3.2, picture adapted from [22].

The resulting system is considered the starting point for applying the proposed data-driven modeling scheme.

We sampled $H_1(s_1)$ at $K_1 = 101$ points on the imaginary axis $s_1 = j2\pi f_1$, with $f_1 \in [10^{-1}, 10^2]$ in addition to $s_1^{(1)} = 0$. Regarding higher-order transfer functions, we randomly selected $K_2 = 343$ and $K_3 = 101$ points to sample $H_2(s_1, s_2)$ and $H_3(s_1, s_2, s_3)$.

Using the proposed algorithm with $\nu = 6$ poles and 20 iterations leads to the accurate fitting of the first $M = 3$ frequency responses, as shown in Figure 11.4 for $m = 2$. In time-domain, the model was tested in a 2-s long simulation with amplitude-modulated chirp input $u(t) = 2 \cdot \sin[2\pi f_0(t)t + \pi/2][1 - \frac{1}{2}\cos(2\pi t)]$, with $f_0(t)$ varying linearly from 0.5 Hz to 50 Hz.

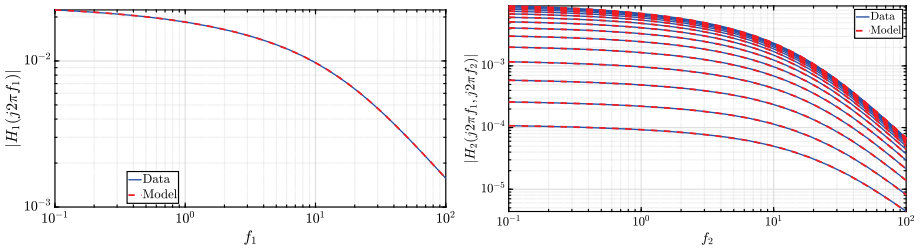

Figure 11.4: Error analysis for the example in Section 3.2. *Left panel*: Model-data comparison for the degree-1 transfer function. *Right panel*: Degree-2 transfer function with respect to f_2 for several fixed values of f_1.

Figure 11.5 compares the original and reduced model responses, showing that adding higher-order transfer functions decreases the error and that the degree-3 model is sufficient to produce a response that is indistinguishable from the full-order model for this case. Note that the error curves depicted in Figure 11.5 are related to the time-domain simulation used for testing that involves an oscillatory signal. The input has here been chosen to push the system into a regime of nonlinear operation where the contribution from the higher-order transfer functions can be appreciated and, in particular, models with $M = 1, 2$ are insufficient, as shown in Figure 11.6.

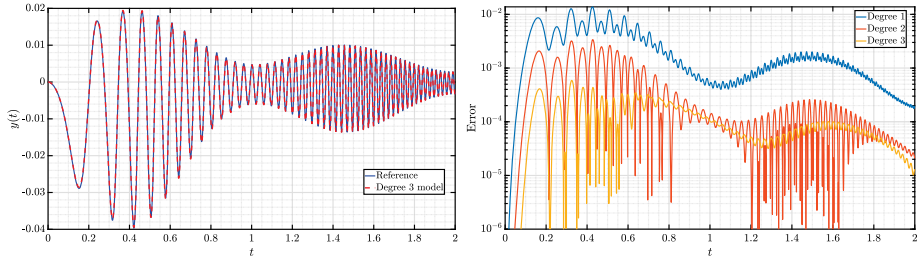

Figure 11.5: Time-domain solution for the example in Section 3.2. *Left panel*: Comparison of full-order and fitted model of degree $M = 3$. *Right panel*: Evolution of instantaneous error with respect to model degree.

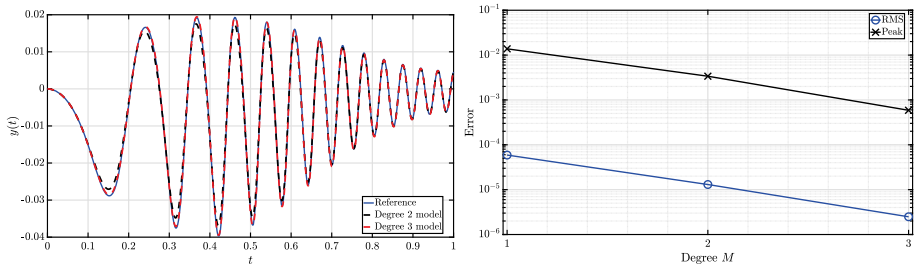

Figure 11.6: *Left panel*: Time-domain response of example in Section 3.2, highlighting the qualitative difference between models with $M = 2$ and $M = 3$. *Right panel*: Error in the time-domain solution of Figure 11.5, measured as root mean square (RMS) and peak error (max. in $t \in [0, 2]$).

4 Conclusions

To summarize, this work described a new algorithm for data-driven approximation of nonlinear input–output maps in the frequency domain, based on Volterra series. The method uses samples of the symmetric transfer functions, directly measurable from response evaluation, without requiring access to the first-principles description of the system to be modeled. Compared to previous work [16], this framework is flexible as it can

approximate GFRFs beyond the second degree. The entire procedure is fast and scalable as it consists of a sequence of linear least-squares optimization problems, thanks to a greedy approach that provides simplicity in exchange for exact optimality. Future work could address this issue, by investigating extensions of this algorithm in which transfer functions are fitted simultaneously rather than sequentially.

Bibliography

[1] A. C. Antoulas, I. V. Gosea, and A. C. Ionita. Model reduction of bilinear systems in the Loewner framework. *SIAM Journal on Scientific Computing*, 38(5):B889–B916, 2016.

[2] P. Benner and T. Breiten. Interpolation-based $\mathcal{H}_2$-model reduction of bilinear control systems. *SIAM Journal on Matrix Analysis and Applications*, 33(3):859–885, 2012.

[3] P. Benner and T. Damm. Lyapunov equations, energy functionals, and model order reduction of bilinear and stochastic systems. *SIAM Journal on Control and Optimization*, 49(2):686–711, 2011.

[4] P. Benner, S. Gugercin, and S. W. R. Werner. Structure-preserving interpolation of bilinear control systems. *Advances in Computational Mathematics*, 47(3):43, 2021.

[5] S. Boyd, L. O. Chua, and C. A. Desoer. Analytical foundations of Volterra series. *IMA Journal of Mathematical Control and Information*, 1(3):243–282, 1984.

[6] T. Breiten and T. Damm. Krylov subspace methods for model order reduction of bilinear control systems. *Systems & Control Letters*, 59(8):443–450, 2010.

[7] T. Carleman. Application de la théorie des équations intégrales linéaires aux systèmes d'équations différentielles non linéaires. *Acta Mathematica*, 59:63–87, 1932.

[8] A. Carlucci, T. Bradde, and S. Grivet-Talocia. Addressing load sensitivity of rational macromodels. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 13(10):1591–1602, 2023.

[9] Y. Chen. Model reduction for nonlinear systems. Master's thesis, Massachusetts Institute of Technology, 1999.

[10] J. Debnath and R. S. Dahiya. Theorems on multidimensional Laplace transform for solution of boundary value problems. *Computers & Mathematics with Applications*, 18(12):1033–1056, 1989.

[11] D. Deschrijver, M. Mrozowski, T. Dhaene, and D. De Zutter. Macromodeling of multiport systems using a fast implementation of the vector fitting method. *IEEE Microwave and Wireless Components Letters*, 18(6):383–385, 2008.

[12] W. Favoreel, B. De Moor, and P. Van Overschee. Subspace identification of bilinear systems subject to white inputs. *IEEE Transactions on Automatic Control*, 44(6):1157–1165, 1999.

[13] G. Flagg and S. Gugercin. Multipoint Volterra series interpolation and $\mathcal{H}_2$ optimal model reduction of bilinear systems. *SIAM Journal on Matrix Analysis and Applications*, 36(2):549–579, 2015.

[14] S. Grivet-Talocia and B. Gustavsen. *Passive Macromodeling: Theory and Applications*. Wiley Series in Microwave and Optical Engineering. Wiley, 2015.

[15] B. Gustavsen and A. Semlyen. Rational approximation of frequency domain responses by vector fitting. *IEEE Transactions on Power Delivery*, 14(3):1052–1061, 1999.

[16] D. S. Karachalios, I. V. Gosea, and A. C. Antoulas. On bilinear time-domain identification and reduction in the Loewner framework. In: *Model Reduction of Complex Dynamical Systems*. International Series of Numerical Mathematics, volume 171, pages 3–30. Birkhäuser, Cham, 2021.

[17] D. S. Karachalios, I. V. Gosea, L. Gkimisis, and A. C. Antoulas. Data-driven quadratic modeling in the loewner framework from input-output time-domain measurements. *SIAM Journal on Applied Dynamical Systems*, 24(1):457–500, 2025.

[18] S. Lefteriu and A. C. Antoulas. On the convergence of the vector-fitting algorithm. *IEEE Transactions on Microwave Theory and Techniques*, 61(4):1435–1443, 2013.

[19] Y. Nakatsukasa, O. Sète, and L. N. Trefethen. The AAA algorithm for rational approximation. *SIAM Journal on Scientific Computing*, 40(3):A1494–A1522, 2018.
[20] W. J. Rugh. *Nonlinear System Theory: The Volterra/Wiener Approach*. Johns Hopkins Series in Information Sciences and Systems. Johns Hopkins University Press, 1981.
[21] M. Schetzen. *The Volterra and Wiener Theories of Nonlinear Systems*. John Wiley & Sons Inc, 1980. ISBN-10: 0471044555.
[22] The MORwiki Community. Nonlinear RC Ladder. MORwiki—Model Order Reduction Wiki, 2018.
[23] L. Zhang and J. Lam. On H_2 model reduction of bilinear systems. *Automatica*, 38(2):205–216, 2002.