# Parametric Rational Model Order Reduction for Nonlinear Systems Subject to Periodic Inputs

Giovanni Conni[1,2,a)], Frank Naets[2,3] and Karl Meerbergen[1]

[1]*Department of Computer Science, KU Leuven, Celestijnenlaan 200A, Leuven, 3001, Belgium.*
[2]*Department of Mechanical Engineering, KU Leuven, Celestijnenlaan 300, Leuven, 3001, Belgium*
[3]*E2E lab, Flanders Make@KU Leuven, Leuven, Belgium*

[a)]Corresponding author: giovanni.conni@kuleuven.be

**Abstract.** With the increase of computationally expensive models over the past decades, the research on Model Order Reduction (MOR) methods has been necessary and fruitful. In particular, efficient and accurate methods to represent the dynamics of Linear Time Invariant (LTI) systems have been developed and thoroughly explored. However, the generalization of these methods to nonlinear systems has proven to be challenging. In this paper, a different point of view is employed to tackle this problem. By limiting the focus of the analysis to a family of nonlinear systems with periodic inputs, a method is developed to approximate the system's output through a parametric LTI system. This is done by treating the input frequency and amplitude as parameters, and defining an approximation method in such a way. Numerical results on a synthetic nonlinear system output are presented and discussed.

## INTRODUCTION

As computational power increased in the past decades, more accurate, and more complex models have been defined to describe physical systems. The increase of complexity and computational costs made necessary the use of MOR methods. Among these algorithms, of particular interest are the so called black-box MOR methods, which only require input-output data. In this paper we will focus on methods based on the AAA algorithm, which has been proven to be accurate and efficient, and allows for high degree of automatization. These methods have been thoroughly explored in the realm of LTI systems, both in the time domain and frequency domain. However, their extension to nonlinear systems has proven to be complicated because of the absence of a straight-forward representation of the input-output relationship, which in LTI systems is defined by the transfer function. Differently, most nonlinear systems can be described in the frequency domain by a series of functions, which increases the complexity of their approximation. The first steps towards the applications of these MOR methods to nonlinear systems have been going in this direction, usually by approximating the first terms of the series. However, the extension to an arbitrary number of terms, necessary to describe more accurately nonlinear dynamics, has not been defined yet.
In this paper, we will try to tackle the problem from another point of view. We will focus on a particular case of nonlinear systems, which allow for a Volterra series expansion, subject to periodic inputs described by two different parameters. Then, we showcase a way to assemble a parametric LTI system which can approximate the nonlinear system response to this family of inputs. While this setting may look restrictive, this method may be applied to mechanics and vibroacustics models, and can be extended to more complex settings.

## METHODOLOGY

The relationship between transforms of nonlinear systems' inputs and outputs can be represented as a functional series, see [1]. Other than the popular Volterra series, another possibility is to use the so called Nonlinear Output Frequency Response Functions (NOFRFs) $G_i(\imath\omega)$. These are defined as $Y(\imath\omega) = \sum_{n=1}^{\infty} G_n(\imath\omega)U_n(\imath\omega)$, where $Y(\imath\omega) = \mathcal{F}\{y(t)\}$ and $U_n(\imath\omega) = \mathcal{F}\{u^n(t)\}$. We are interested in periodic input functions which can be represented by two parameters: an

amplitude $A$ and a forcing frequency $\omega_F$, i.e. $Y(\imath\omega; A, \omega_F) = \sum_{n=1}^{\infty} G_n(\imath\omega) U_n(\imath\omega; A, \omega_F)$. It is important to point out that the poles of $Y$ can be divided in the system natural poles, which are included in $G_n$, and the forcing harmonics, generated by $U_n$, which depend on $\omega_F$ and not on $A$. This detail is at the core of the MOR we are going to define.

The idea of this paper is to employ parametric MOR techniques to represent the nonlinear system response when $\omega_F$ and $A$ are changed. We consider rational-interpolation-based MOR techniques, and in particular AAA, due to its flexibility and its straight-forward representation in a state-space form [2]. Moreover, the method determines automatically the approximation order based on two user-set stopping criterion: the minimum error `tol` and the maximum iterations number `mmax`. In this way, we can represent the nonlinear system as a parametric LTI system, greatly improving the computational cost of simulations. In particular, the proposed method combines two AAA variations which represent two different ways of approximating parametric functions: set-valued AAA (svAAA) [3] and parametric AAA (pAAA) [4]. This combination is therefore named psvAAA.

Considering a parametric function of the form $f(s; p, q)$, the psvAAA method looks for a rational approximation of the form

$$f_{psv}(s; p, q) = \frac{\sum_{i=1}^{m_\sigma} \sum_{j=1}^{m_\pi} w_{i,j} \tilde{f}_{i,j}(q) / (s-\sigma_i)(p-\pi_j)}{\sum_{i=1}^{m_\sigma} \sum_{j=1}^{m_\pi} w_{i,j} / (s-\sigma_i)(p-\pi_j)}, \tag{1}$$

where $m_\sigma$ and $m_\pi$ are the approximation orders in the $s$ and $p$ terms respectively, $\sigma_i$ and $\pi_j$ are the interpolation frequencies and parameter values respectively, $w_{i,j}$ are the barycentric weights, and $\tilde{f}_{i,j}(q) = \tilde{f}(\sigma_i; \pi_j, q)$ are a collection of functions, only dependent on the parameter $q$, which interpolate the original function in $q_k$, i.e. $\tilde{f}_{i,j}(q_k) = f(\sigma_i; \pi_j, q_k)$. Note that, except the dependency of $\tilde{f}_{i,j}$ from $q$, this form is the same as pAAA. The algorithm is automatic, and only requires the user to give the sampled data $[s_i]_{i=1}^N$, $[p_i]_{i=1}^P$, $[q_i]_{i=1}^Q$, the function values $f_{i,j,k} = f(s_i; p_j, q_k)$, and the two stopping criterions, `tol` and `mmax`. At the $m$-th iteration, the algorithm selects a new frequency-parameter pair in a greedy manner by solving $(\sigma_m, \pi_m) = \arg\max_{s_i, p_j} \|f(s_i; p_j, q_k) - f_{psv}(s_i; p_j, q_k)\|_\infty$ for $k = 1, \ldots, P$. Then, $w_{i,j}$ are found by minimizing the linearized $L^2$ approximation error

$$\begin{aligned} \left\| f(s; p, q) - f_{psv}(s; p, q) \right\|_2 &\approx \left\| \left( \sum_{i=1}^{m_\sigma} \sum_{j=1}^{m_\pi} w_{i,j} / (s-\sigma_i)(p-\pi_j) \right) f(s; p, q) - \sum_{i=1}^{m_\sigma} \sum_{j=1}^{m_\pi} w_{i,j} \tilde{f}_{i,j}(q) / (s-\sigma_i)(p-\pi_j) \right\|_2 \\ &\approx \left\| \left[ \sum_{i=1}^{m_\sigma} \sum_{j=1}^{m_\pi} \left( f(s;p,q) - \tilde{f}_{i,j}(q) \right) / (s-\sigma_i)(p-\pi_j) \right] w_{i,j} \right\|_2, \end{aligned} \tag{2}$$

which is evaluated in the sampled data. Therefore, no definition of the functions $\tilde{f}_{i,j}(q)$ is necessary in this step. The least squares problem in Equation 2 can be rewritten in matrix form, and the weights $w_{i,j}$ can be found as the minimum singular vector of said matrix. Once the weights have been recovered, they are plugged into the $f_{psv}$ definition of Equation 1, and if neither of the stopping conditions is met, the procedure is repeated. Once the iterations are stopped, the functions $\tilde{f}_{i,j}(q)$ are defined by interpolation of $f(\sigma_i; \pi_j, q)$ in $q_k$. In the following barycentric interpolation is employed, but different methods choices may be used.

The core difference between psvAAA and the version of pAAA with 2 parameters, described in [4], is that in the former the weights are minimizing the error for every parameter value $q_k$. This means that the weights, and therefore the poles of $f_{psv}$, which are the zeros of the denominator in Equation 1, are independent from $q$. This characteristic corresponds to the problem we introduced at the beginning of this section, as the poles of the function $Y(\imath\omega; A, \omega_F)$ depend only on $\omega_F$ and not on $A$. As will be shown in the next section, this distinction improves the approximation accuracy, and prevents the generation of spurious poles.

In the following section, the psvAAA method is applied to a synthetic test case, a simplified form of the output transform of a Duffing oscillator.

## NUMERICAL TESTS

To test the psvAAA method, we consider a synthetic transform output $Y(\imath\omega) = G_1(\imath\omega) U_1(\imath\omega) + G_2(\imath\omega) U_2(\imath\omega)$, with

$$G_1(\imath\omega) = \frac{1}{-m\omega^2 + 2c\imath\omega + k}, \quad G_2(\imath\omega) = -k_2 G_1^2(\imath\omega) G_1(2\imath\omega), \quad U_1(\imath\omega; A, \omega_F) = \frac{A/2}{\imath\omega - (-1 + \imath\omega_F)} + \frac{A/2}{\imath\omega - (-1 - \imath\omega_F)},$$

$$U_2(\imath\omega; A, \omega_F) = \frac{A^2/4}{\imath\omega - (-2 + 2\imath\omega_F)} + \frac{A^2/4}{\imath\omega - (-2 - 2\imath\omega_F)} + \frac{A^2/2}{\imath\omega + 2}.$$

The NOFRFs $G_n$ formulas have been found in [5] for the case of a Duffing oscillator under harmonic inputs. On the other hand, the input transforms $U_n$ are the results obtained for the input $u(t; A, \omega_F) = Ae^{-t}\cos(\omega_F t)\theta(t)$, with $\theta(t)$ the Heaviside function. The poles of $Y$ are $\mu = \left\{-c \pm \sqrt{c^2 - mk}, \left(-2c \pm \sqrt{4c^2 - 4mk}\right)/4, -1 \pm \imath\omega_F, -2 \pm 2\imath\omega_F, -2\right\}$, and as expected they do not depend on $A$ but only on $\omega_F$. Note that the NOFRFs series is truncated at order 2, which is often too low an order for nonlinear systems. Moreover, note that for a real nonlinear model the definition of $G_n$ is dependent on $U_n$, see [5]. Even though $Y(\imath\omega; A, \omega_F)$ is not physically meaningful, it gives important insights on the performance of these methods and it is the first step towards applications to more realistic nonlinear systems.

The tests are carried out in Matlab on $N = 100$ sample points along the imaginary axis in the interval $\omega \in \{0\} \cup [0.1, 100]$. The forcing frequency $\omega_F$ samples are $M = 20$ in the interval $[0.5\pi, 10\pi]$, while for $A$, $nF = 10$ samples in $[0.1, 100]$ are considered. The parameters of $G_i$ are selected as $m = 1$, $c = 1$, $k = 50$, and $k_2 = 10^3$. Note that for this application we are using a modified version of psvAAA which includes complex conjugate terms in order for the state-space form to be real-valued.

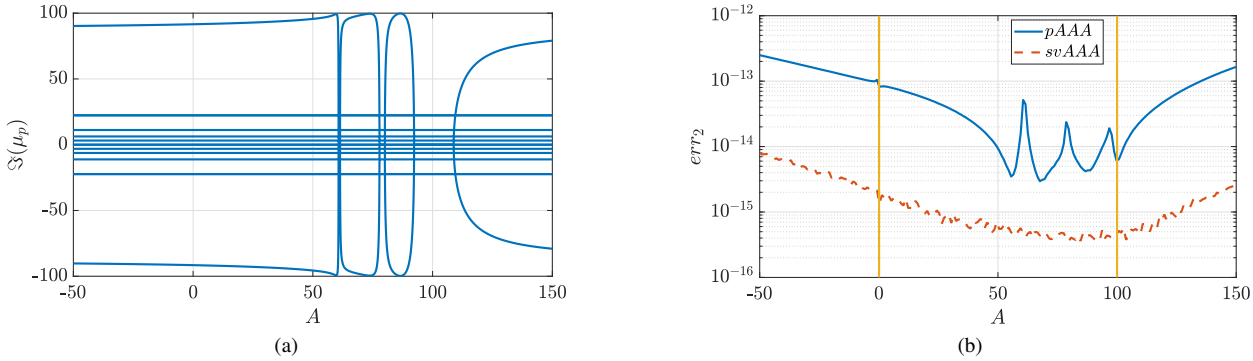


**FIGURE 1.** Comparison of pAAA and svAAA approximations of $Y(\imath\omega; A, \pi)$: (a) the approximation poles $\Im(\mu_p)$ vary with $A$, and include spurious poles; (b) normalized $L^2$ error $err_2(A) = {\|Y(\imath\omega;A,\pi) - Y_\cdot(\imath\omega;A)\|_2}/{\|Y(\imath\omega;A,\pi)\|_2}$ for both methods. The vertical yellow lines indicate the sampling set for $A$.

We first compare the accuracy of pAAA and svAAA for cases in which the system poles do not change with a parameter, by fixing $\omega_F = \pi$ and only considering $Y(\imath\omega; A, \pi)$. For pAAA we use `mmax` $= 100$ and `tol` $= 10^{-10}$. The algorithm reaches the desired tolerance after 11 iterations, with $m_\sigma = 14$ and $m_\pi = 6$. As can be seen in Figure 1.(a), the poles of the approximation do vary with $A$, differently from what is expected. For svAAA we set `mmax` $= m_\sigma$ and `tol` $= 0$, thus forcing the approximation order to be the same as pAAA. The normalized $L^2$ error is then compared in Figure 1.(b), including for parameter values outside the sampling set. As can be seen, $Y_{sv}$ is around two orders of magnitude more accurate than $Y_p$, and its poles are not dependent on the parameter $A$, as expected. This supports the idea of combining the two methods to approximate $Y(\imath\omega; A, \omega_F)$.

We now apply the psvAAA method to $Y(\imath\omega; A, \omega_F)$ with `tol` $= 10^{-8}$ and `mmax` $= 100$. The algorithm stops after 9 iterations, with $m_\sigma = 13$ and $m_\pi = 7$. The poles of $Y$, together with the ones of of $Y_{psv}$, are plotted in Figure 2 for two different values of $\omega_F$. The poles error $err_\mu(\omega_F) = \max_{i=1,\dots,9}\left(\min {|\mu_i(\omega_F) - \mu_{psv}(\omega_F)|}/{|\mu_i(\omega_F)|}\right)$ for these $\omega_F$ values are $err_\mu(0.5\pi) = 1.0523 \cdot 10^{-7}$ and $err_\mu(10\pi) = 1.1388 \cdot 10^{-8}$. The approximation $Y_{psv}$ is accurate for parameters outside the sampling interval too, as can be seen both in Figure 3.(a) and 3.(b). In particular, in Figure 3.(b) it is possible to see the sampling interval, represented by the white rectangle, and the overall accuracy of $Y_{psv}$.

## CONCLUSIONS

We have presented a combination of two MOR methods, pAAA and svAAA, which has proven to be accurate in representing the behavior of a synthetic nonlinear systems subject to a family of periodic inputs, with varying amplitude $A$ and forcing frequency $\omega_F$. The accuracy of this method is coming from its differentiation between $A$ and $\omega_F$ in their effects on the output.

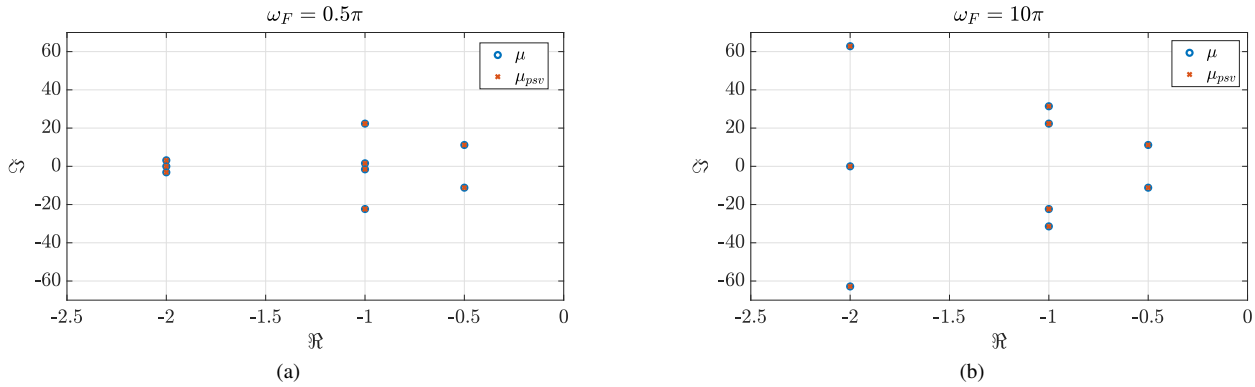


**FIGURE 2.** Comparison between the poles of $Y$, $\mu$, and of $Y_{psv}$, $\mu_{psv}$, for different values of $\omega_F$.

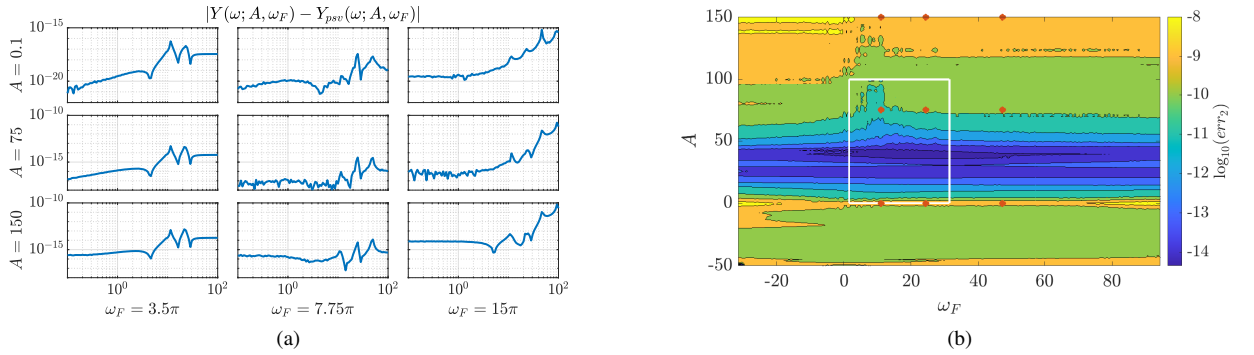


**FIGURE 3.** Accuracy results of the psvAAA approximation: (a) absolute error for different parameters choice: sampled values ($A = 0.1$, $\omega_F = 3.5\pi$); values in the sampling set but not sampled ($A = 75$, $\omega_F = 7.75\pi$); values outside the sampling set ($A = 150$, $\omega_F = 15\pi$); (b) normalized $L^2$ approximation error. The white rectangle represents the parameters sampling set, the red dots the values used in (a).

## ACKNOWLEDGMENTS

The authors want to thank Dr. Ion Victor Gosea for his help and support during the development of this paper.

## REFERENCES

[1] C. M. Cheng, Z. K. Peng, W. M. Zhang, and G. Meng "Volterra-series-based nonlinear system modeling and its engineering applications: A state-of-the-art review," , Mech. Syst. Signal Pr. **87**, 340–364 (2017).

[2] Y. Nakatsukasa, O. Sète, and L. N. Trefethen "The AAA algorithm for rational approximation," , SIAM J. Sci. Comput. **40**, A1494–A1522 (2018).

[3] P. Lietaert, K. Meerbergen, J. Pérez, and B. Vandereycken "Automatic rational approximation and linearization of nonlinear eigenvalue problems," , IMA J. Numer. Anal. **42**, 1087–1115 (2022).

[4] A. C. Rodriguez, L. Balicki, and S. Gugercin "The p-AAA algorithm for data-driven modeling of parametric dynamical systems," , SIAM J. Sci. Comput. **45**, A1332–A1358 (2023).

[5] Z. K. Peng, Z. Q. Lang, and S. A. Billings "Resonances and resonant frequencies for a class of nonlinear systems," , J. Sound Vib. **300**, 993–1014 (2007).