KU LEUVEN

TDAAA and parametric linearization of nonlinear systems with periodic input



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

[1] NUMA, CS Dpt., KU Leuven

[2]LMSD, Mech. Dpt., KU Luevenm

[3]Flanders Make@KU Leuven, Heverlee, Belgium

1 Outline

1. Time-domain MOR algorithms
   - TDVF
   - TDLF
2. TDAAA
3. Extension to nonlinear systems

KU LEUVEN

Vector Fitting algorithm

Input : $\vec{s}, \vec{H}, \vec{\mu}_0, \texttt{tol}, \texttt{itmax}$

Output: $\vec{H}_r, \vec{\mu}_r$

$\vec{\mu_r} \leftarrow \vec{\mu}_0; \quad it = 0$;

while $it < \texttt{itmax}$, $err > \texttt{tol}$ **do**

$\begin{bmatrix} \vec{c} \\ \vec{k} \end{bmatrix} \leftarrow \arg\min_{\vec{x}} \left\| \vec{H} + \begin{bmatrix} -1 & -\frac{1}{s-\vec{\mu}_r^T} & \frac{\vec{H}}{s-\vec{\mu}_r^T} \end{bmatrix} \vec{x} \right\|_2$;

$\vec{\mu}_r \leftarrow \texttt{eig}\left(\text{diag}(\vec{\mu}_r) - \vec{1}\vec{k}^T\right)$;

Calculate err; $\quad it = it + 1$;

end

KU LEUVEN

- For stability, any unstable pole $\boldsymbol{\mu}_r$ is flipped along the imaginary axis to make it stable
- Variants of the VF algorithm for noise stability and convergence have been defined
- If H is the transfer function of a dynamical system, VF can be used as a ROM technique
- Therefore H can be substituted by the ratio of the input and output transform, $H = {}^{Y}/_{U}$, and thus the Tie-Domain Vector Fitting is defined

All the systems considered are SISO, extension to MIMO systems are available.
First consider LTI systems of the form

$$\begin{cases} \dot{\vec{x}}(t) = A\vec{x}(t) + Bu(t) \\ y(t) = C\vec{x}(t) \end{cases}$$

KU LEUVEN

Time-Domain Vector Fitting (Gustavsen et al. 1999)

Input : $\vec{t}, \vec{u}, \vec{y}, \vec{\mu}_0$, `tol`,`itmax`

Output: $\vec{y}_r$,$\vec{\mu}_r$

$\vec{\mu}_r \leftarrow \vec{\mu}_0; \quad it = 0$;

while $it <$ `itmax`, $err >$ `tol` **do**

$$\begin{bmatrix} \vec{c} \\ \vec{k} \end{bmatrix} \leftarrow \arg\min_{\vec{x}} \left\| \vec{y} + \begin{bmatrix} -\vec{u} & -\vec{u}_{1:n} & \vec{y}_{1:n} \end{bmatrix} \vec{x} \right\|_2, \text{ with}$$

$$f_j = \int_0^t e^{\mu_j (t-\tau)} f(\tau) d\tau \ ;$$

$$\vec{\mu}_r \leftarrow \texttt{eig}\left(\text{diag}(\vec{\mu}_r) - \vec{1}\vec{k}^T \right) ;$$

Calculate err; $\quad it = it + 1$;

end

KU LEUVEN

1 Time-Domain Vector Fitting

	Rational form	Method	Stabilization	Extensions
TDVF	Pole-residue	L^2 minimization	Pole flipping	Relaxed VF (Gustavsen 2006), Fast VF (Deschrijver et al. 2008)

- The convolutions at each iteration are the main bottleneck
- The approximation order (fixed) and the starting poles affect the method's performance

- At each iteration any unstable pole $\vec{\mu}_r$ is flipped along the imaginary axis (Pole Flipping)
- Variants of the VF algorithm which improve stability to noise (Gustavsen 2006) and computational speed (ADD CITATION) have been developed
- The TDVF iteration is just the antitranform of the standard VF iteration
- Is based on the pole-residue form
- Approximation based on the L^2 error
- The TDVF method carries out all calculations in the time domain: the convolutions at each iteration can be calculated via filters
- The approximation order is fixed and the starting poles affect the method's performance

Loewner Framework (Antoulas et al. 1986)

Input : $\vec{s}, \vec{h}$

Output: E_r, A_r, B_r, C_r

1 Define $\vec{v}, \vec{u} : \vec{v} \cup \vec{w} = \vec{h}, \vec{v} \cap \vec{w} = \emptyset$,
$\vec{\sigma}, \vec{\rho} : \vec{\sigma} \cup \vec{\rho} = \vec{s}, \vec{\sigma} \cap \vec{\rho} = \emptyset$;

2 $[\mathbb{L}]_{ij} \leftarrow \frac{v_i - w_j}{\sigma_i - \rho_j}$, $[\mathbb{L}_s]_{ij} = \leftarrow \frac{\sigma_i v_i - \rho_j w_j}{\sigma_i - \rho_j}$;

3 $[Y, \Sigma_1, \sim] \leftarrow \texttt{svd}([\mathbb{L}\ \mathbb{L}_s])$, $[\sim, \Sigma_2, X] \leftarrow \texttt{svd}([\mathbb{L}\ \mathbb{L}_s]^T)$;

4 Choose approximation order r ;

5 $\{E_r, A_r, B_r, C_r\} = \{-Y_r^T \mathbb{L} X_r, -Y_r^T \mathbb{L}_s X_r, Y_r^T \vec{v}, \vec{w} X_r\}$

KU LEUVEN

Time-Domain Loewner Framework (Peherstorfer et al. 2017)

Input : $\vec{u}, \vec{y}$

Output: E_r, A_r, B_r, C_r

1 $\vec{h} \leftarrow \arg\min_{\vec{x}} \|F\vec{x} - \vec{y}\|_2^2$, with $[F]_{ij} = U_i e^{\mathrm{i}\frac{2\pi j}{N_t} i}$, U_i non-zero Fourier coefficients of $\vec{u}$;

2 Define $\vec{v}, \vec{u} : \vec{v} \cup \vec{w} = \vec{h}, \vec{v} \cap \vec{w} = \emptyset$, $\vec{\sigma}, \vec{\rho} : \vec{\sigma} \cup \vec{\rho} = \vec{s}, \vec{\sigma} \cap \vec{\rho} = \emptyset$;

3 $[\mathbb{L}]_{ij} \leftarrow \frac{v_i - w_j}{\sigma_i - \rho_j}$, $[\mathbb{L}_s]_{ij} \leftarrow \frac{\sigma_i v_i - \rho_j w_j}{\sigma_i - \rho_j}$;

4 $[Y, \Sigma_1, \sim] \leftarrow \mathtt{svd}([\mathbb{L}\ \mathbb{L}_s])$, $[\sim, \Sigma_2, X] \leftarrow \mathtt{svd}([\mathbb{L}\ \mathbb{L}_s]^T)$;

5 Choose approximation order r ;

6 $\{E_r, A_r, B_r, C_r\} = \{-Y_r^T \mathbb{L} X_r, -Y_r^T \mathbb{L}_s X_r, Y_r^T \vec{v}, \vec{w} X_r\}$

KU LEUVEN

1 Time-Domain Loewner Framework

	Rational form	Method	Stabilization	Extensions
TDVF	Pole-residue	L^2 minimization	Pole flipping	Relaxed VF (Gustavsen 2006), Fast VF (Deschrijver et al. 2008)
TDLF	Barycentric form	Interpolation	Post-processing (Gosea et al. 2021b)	Quadratic Bilinear systems (Gosea et al. 2018)

- The input's Fourier coefficients must be sparse, otherwise F becomes ill-conditioned
- Only fully developed outputs are considered

- Different post-processing methods are available to make the ROM stable, (Gosea et al. 2021b)
- The LF has been expanded to bilinear and parametric systems
- It is based on the barycentric form
- The ROM's transfer function H_r interpolates the FOM's one H in $\vec{h} = H(\vec{s})$
- The Fourier coefficients of the input must be sparse, otherwise the LS problem becomes ill-conditioned

2 Outline

1. Time-domain MOR algorithms
2. TDAAA
3. Extension to nonlinear systems

KU LEUVEN

AAA (Nakatsukasa et al. 2018)

Input : $\vec{s}, \vec{h}$, `itmax`, `tol`
Output: $\vec{h}_r$

Initialize $\vec{h}_r \leftarrow \frac{1}{N}\sum \vec{h}, \vec{\sigma} = \emptyset, m = 1$;
while $m <$ `itmax`, $err >$ `tol` **do**
  $\sigma_m \leftarrow \arg\max_{z \in \vec{s}} \|h(z) - h_r(z)\|_\infty, \vec{z} = \vec{s} \setminus \vec{\sigma}$;
  $\vec{w} \leftarrow \mathbb{L}\vec{w} = 0$, where $[\mathbb{L}]_{ij} = \frac{h(z_i) - h(\sigma_j)}{z_i - \sigma_j}$;
  $\vec{h}_r \leftarrow \frac{\sum_{i=1}^{m} w_m f_m / \vec{z} - \sigma_m}{\sum_{i=1}^{m} w_m / \vec{z} - \sigma_m}$;
  Calculate err; $m = m + 1$;
end

KU LEUVEN

Time-Domain AAA

Input : $\vec{t}, \vec{u}, \vec{y}$, `itmax`, `tol`

Output: h_r

$\vec{H} \leftarrow \texttt{fft}\{\vec{y}\}/\texttt{fft}\{\vec{u}\}$;

Initialize $\vec{h}_r \leftarrow \frac{1}{N}\sum \vec{H}, \vec{\sigma} = \emptyset, m = 1$;

while $m <$ `itmax`, $err >$ `tol` **do**

  $\sigma_m \leftarrow \arg\max_{z \in \vec{s}} \|H(z) - h_r(z)\|_\infty, \vec{z} = \vec{s} \setminus \vec{\sigma}$;

  $\vec{w} \leftarrow \mathbb{L}\vec{w} = 0$, where $[\mathbb{L}]_{ij} = \frac{H(z_i) - H(\sigma_j)}{z_i - \sigma_j}$;

  $\vec{h}_r \leftarrow \frac{\sum_{i=1}^{m} w_m H(\sigma_m)/\vec{z} - \sigma_m}{\sum_{i=1}^{m} w_m/\vec{z} - \sigma_m}$;

  Calculate err; $m = m + 1$;

end

$\vec{f} \leftarrow \arg\min_{\vec{x}} \|\vec{x}^T y_r - \vec{y}\|_2$, wth y_r ROM output for u ;

$\vec{h}_r \leftarrow \frac{\sum_{i=1}^{m} w_m f_m/\vec{z} - \sigma_m}{\sum_{i=1}^{m} w_m/\vec{z} - \sigma_m}$;

KU LEUVEN

2 Time-Domain AAA

	Rational form	Method	Stabilization	Variations
TDVF	Pole-residue	L^2 minimization	Pole flipping	Relaxed VF (Gustavsen 2006), Fast VF (Deschrijver et al. 2008)
TDLF	Barycentric form	Interpolation	Post-processing (Gosea et al. 2021b)	Quadratic Bilinear systems (Gosea et al. 2018)
TDAAA	Barycentric form	Interpolation & L^2 minimization	Pole deletion (Costa et al. 2021; Huybrechs et al. 2023), pole flipping	Strictly proper (Gosea et al. 2020), Parametric (Rodriguez et al. 2020)

- Once the poles have been shifted, w_j are found by solving $\mathcal{C}\vec{w} = \vec{b}$, e.g. (Aumann et al. 2023)

KU LEUVEN

- To enforce stability, we use Pole Flipping, even though may be suboptimal, as described in (Gosea et al. 2021b). With a procedure similar to the one in (Aumann et al. 2023), the weights are recalculated to enforce this condition. Another common solution for the stability issue is deleting unstable poles, see (Costa et al. 2021) and (**ONTINUUUMONE**)
- Variations for parametric systems, and multiple extensions for MIMO systems are available, see (Deckers et al. 2022; Gosea et al. 2021a)
- It is based on the barycentric form
- The ROM's transfer function H_r interpolates the FOM's one H in $\vec{h} = H(\vec{s})$, and minimizes the L^2 error on the rest of the domain, combining the two previously seen methods
- For the TDAAA. the output needs to have compact support transform, the opposite of TDVF

2 Pros and cons

- TDVF
  - ✓ the time-domain fitting can be carried out with any input
  - × dependence on initial conditions
- TDLF
  - ✓ non-iterative, faster algorithm
  - × the `fft` is required, constrained choice of input
- TDAAA
  - ✓ authomatic choice of interpolation points and ROM size
  - × the `fft` is required

TDAAA-alt

Input : $\vec{t}, \vec{u}, \vec{y}$, `itmax`, `tol`

Output: h_r

$\vec{Y} \leftarrow \texttt{fft}\{\vec{y}\}$, $\vec{U} \leftarrow \texttt{fft}\{\vec{u}\}$;

Initialize $\vec{h}_r \leftarrow \frac{1}{N}\sum \vec{Y}/\vec{U}, \vec{\sigma} = \emptyset, m = 1$;

while $m <$ `itmax`, $err >$ `tol` **do**

  $\sigma_m \leftarrow \arg\max_{z \in \vec{s}} \|Y(z) - \vec{h}_r(z)U(z)\|_\infty, \vec{z} = \vec{s} \setminus \vec{\sigma}$;

  $\vec{w} \leftarrow \mathbb{L}\vec{w} = -Y(\vec{z})$, $[\mathbb{L}]_{ij} = \frac{Y(z_i)U(\sigma_j) - Y(\sigma_j)U(z_i)}{z_i - \sigma_j}$;

  $\vec{h}_r \leftarrow \frac{\sum_{j=1}^{m} w_j Y(\sigma_j)/\vec{z} - \sigma_j}{1 + \sum_{j=1}^{m} w_j U(\sigma_j)/\vec{z} - \sigma_j}$;

  Calculate err; $m = m + 1$;

end

$\vec{f} \leftarrow \arg\min_{\vec{x}} \|\vec{x}^T y_r - \vec{y}\|_2$, with y_r ROM output for u ;

$\vec{h}_r \leftarrow \frac{\sum_{j=1}^{m} w_j f_j/\vec{z} - \sigma_j}{1 + \sum_{j=1}^{m} w_j U(\sigma_j)/\vec{z} - \sigma_j}$;

KU LEUVEN

2 TDAAA and TDVF

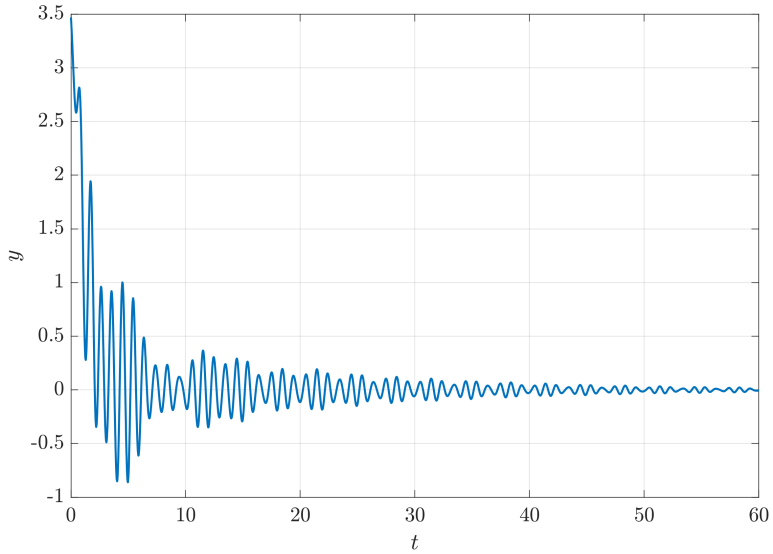
- TDAAA is more flexible than TDVF because it does not need initial guesses for poles or MOR order
- AAA is more stable to noise than LF (Gosea et al. 2021c), while the Relaxed VF (Gustavsen 2006) improves the VF robustness to noise
- The direct use of time data at the end of the algorithm allows to bypass numerical errors in the `fft`
- AAA can be used to find the TDVF starting poles, improving its convergence speed and accuracy

2 System Identification

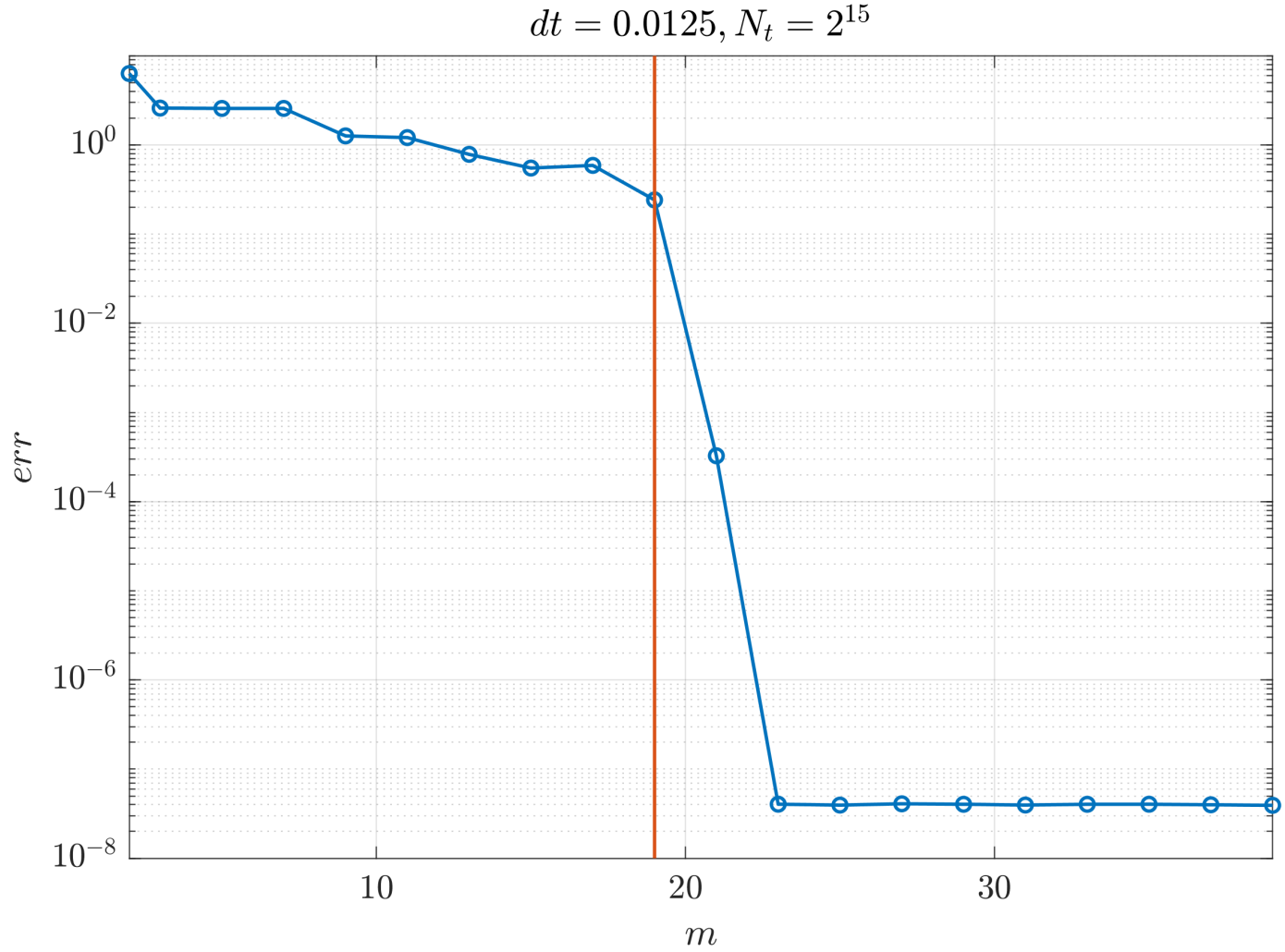
To validate the methods, we consider the LTI system

$$\begin{cases} \dot{\vec{x}}(t) = A\vec{x}(t) + Bu(t) \\ y(t) = C\vec{x}(t) \end{cases},$$

with $A = \text{diag}(\mu_1, \ldots, \mu_n)$, μ_j randomly selected.
TDAAA and TDVF are applied to the system's impulse response.

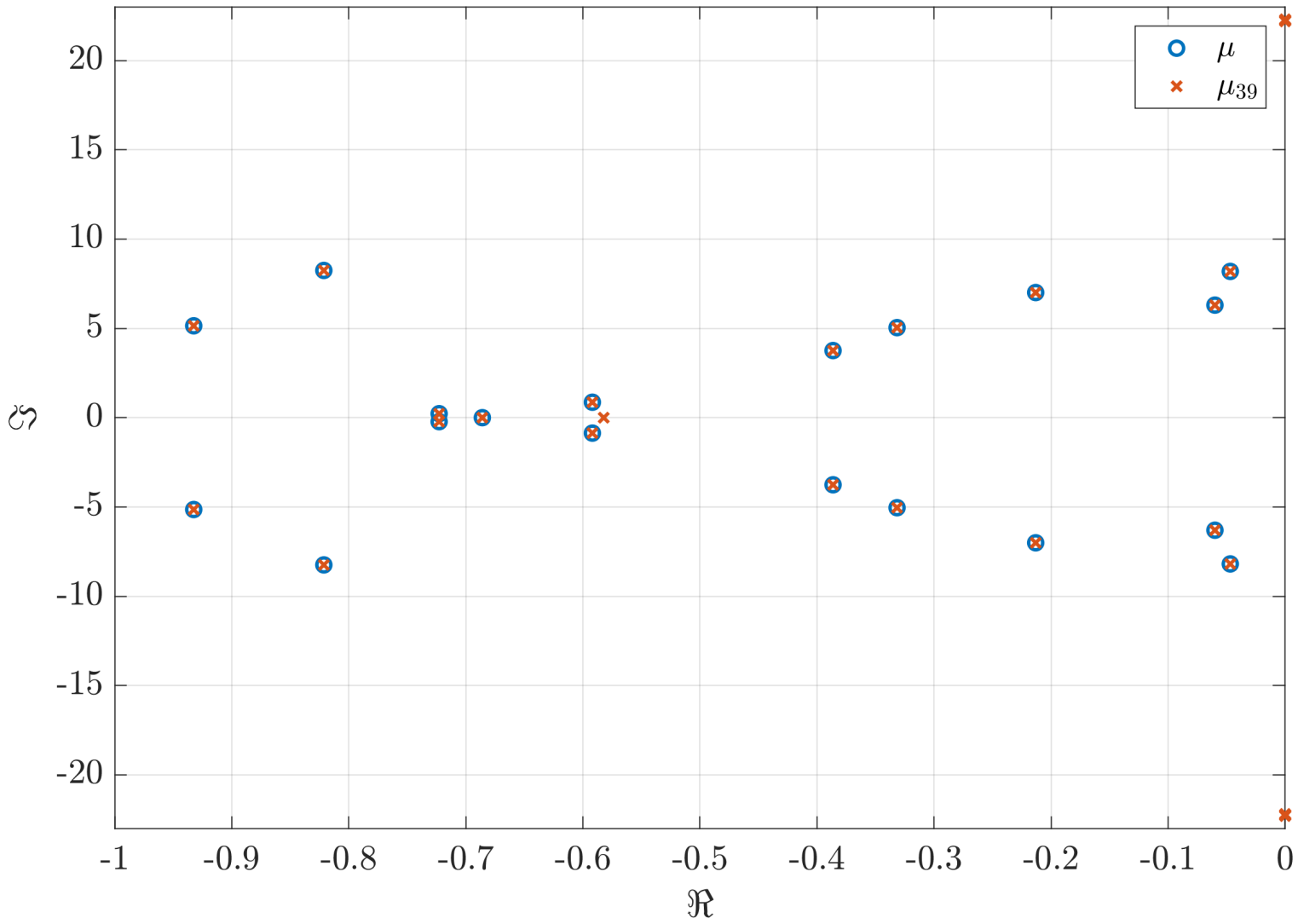


AAA reaches the best accuracy for $m > n$



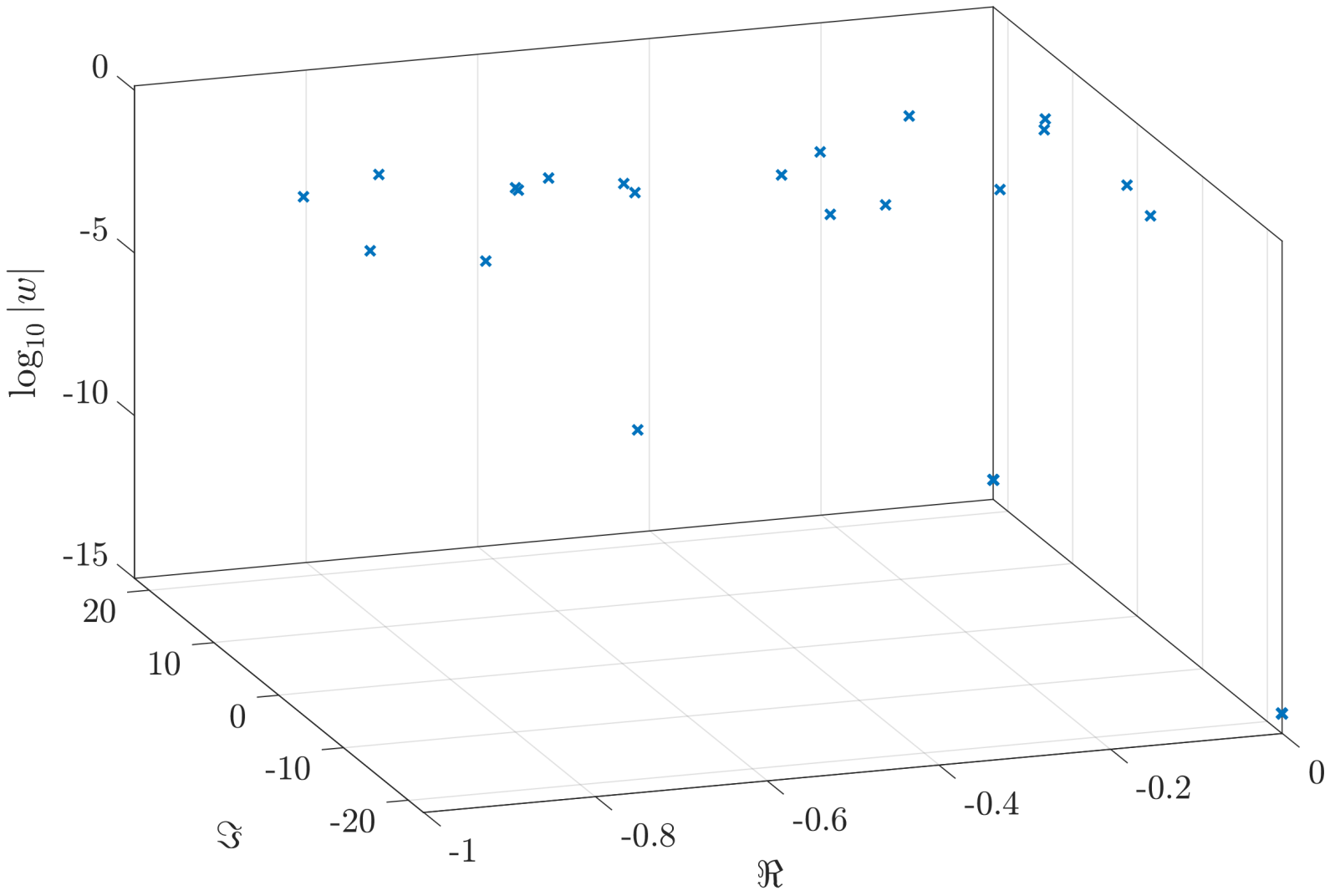
KU LEUVEN

Spurious poles are pruned in the final time-domain LS step



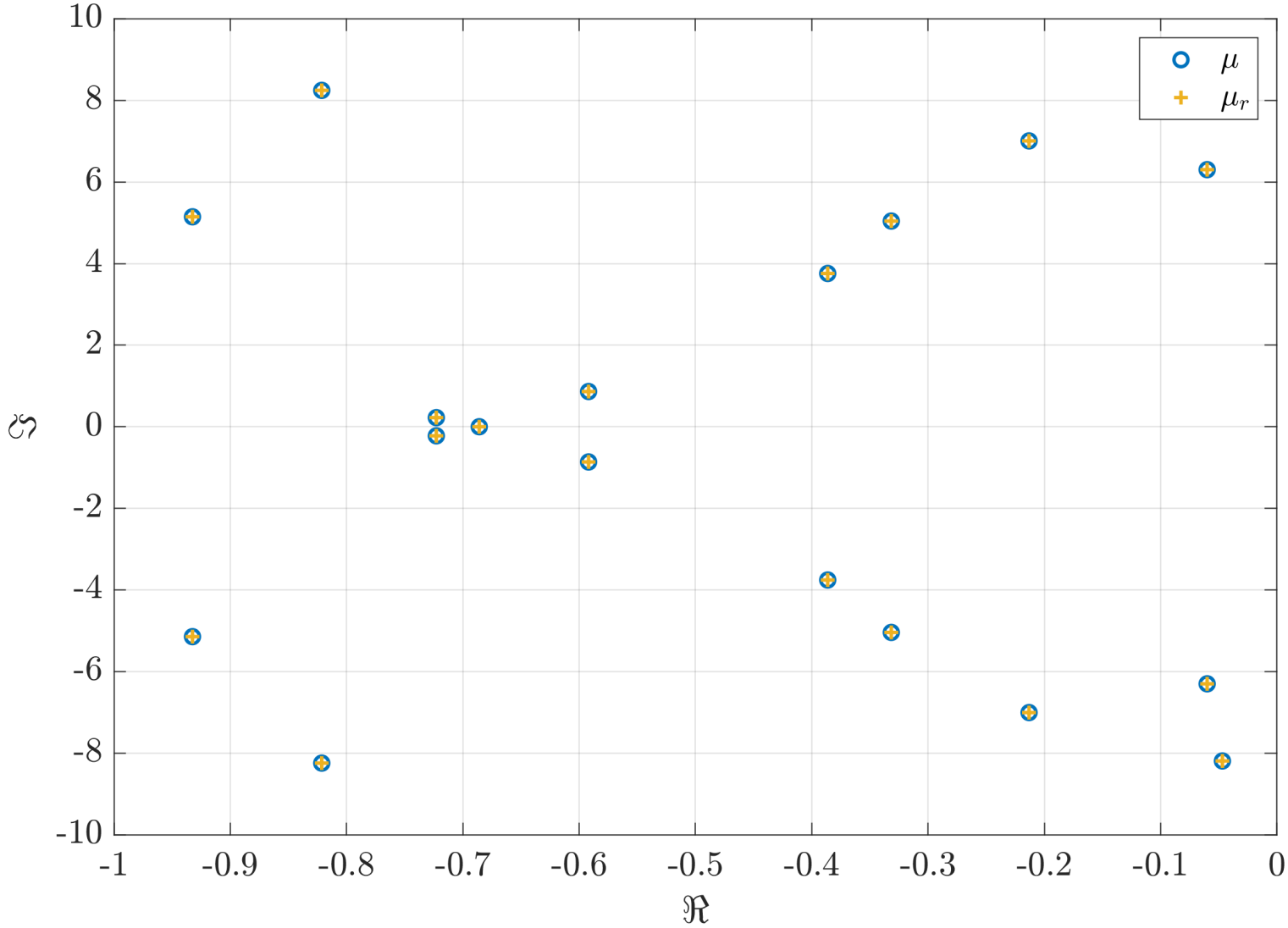
KU LEUVEN

Spurious poles are pruned in the final time-domain LS step



KU LEUVEN

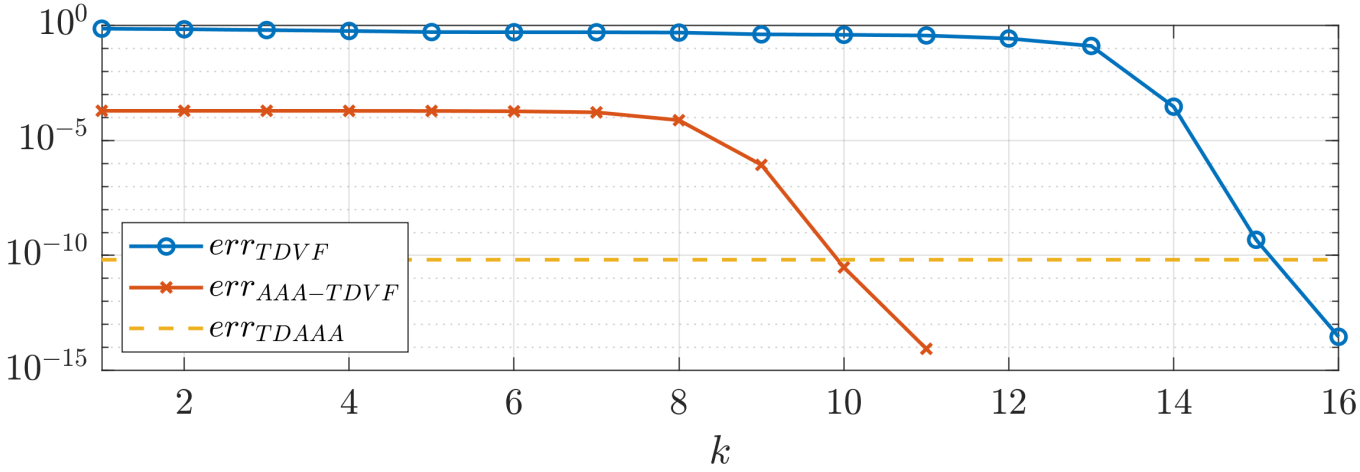
Spurious poles are pruned in the final time-domain LS step

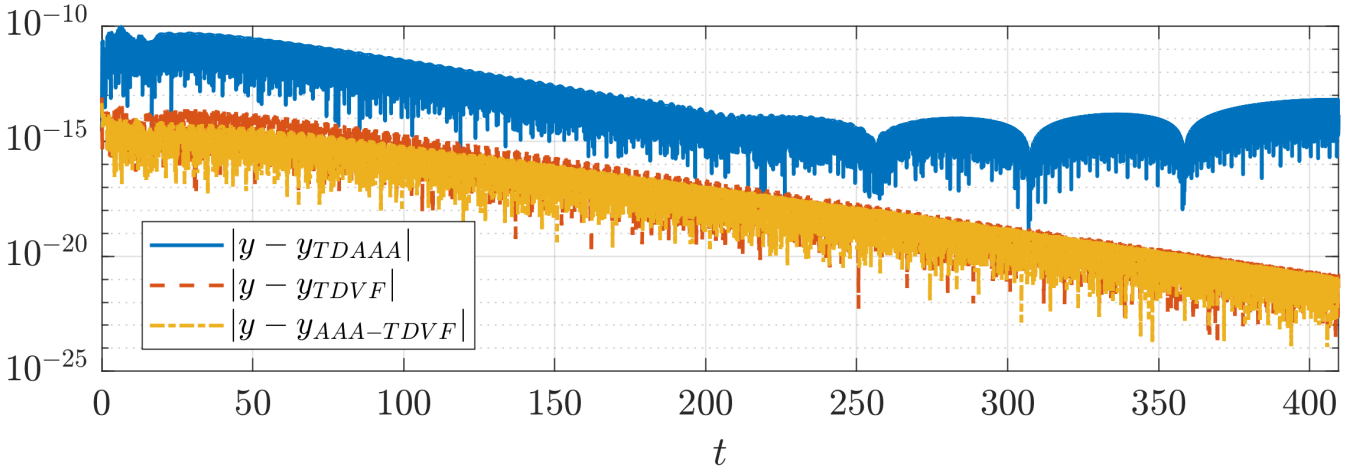


KU LEUVEN

Linearly spaced and AAA initializations for TDVF are compared: starting from the AAA poles reduces the iterations

▸ Sampling





KU LEUVEN

2 Model Reduction

Consider the Pentzl system, with state matrices

$$A = \text{diag}(A_1, A_2, A_3, -1, \ldots, -1000),$$

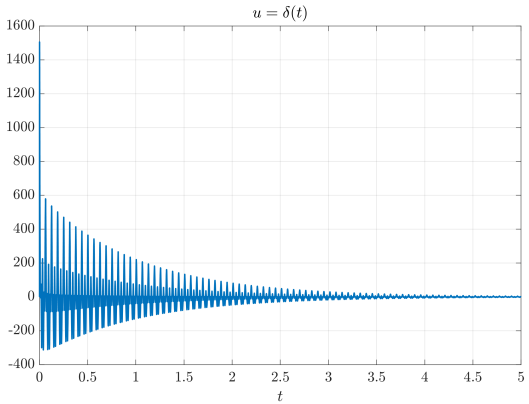
$$\text{with} A_1 = \begin{bmatrix} -1 & 100 \\ -100 & -1 \end{bmatrix},$$

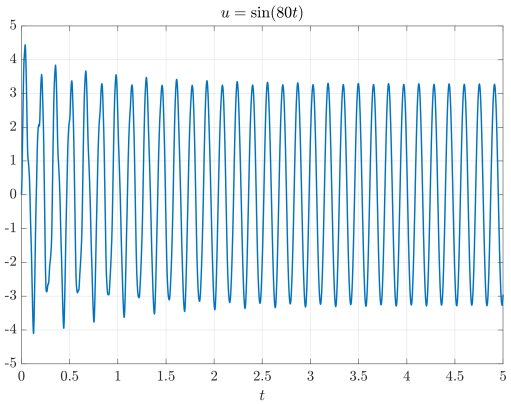
$$A_2 = \begin{bmatrix} -1 & 200 \\ -200 & -1 \end{bmatrix},$$

$$A_1 = \begin{bmatrix} -1 & 400 \\ -400 & -1 \end{bmatrix},$$

$$B = C^T = \begin{bmatrix} 10 & \ldots & 10 & 1 & \ldots & 1 \end{bmatrix}$$

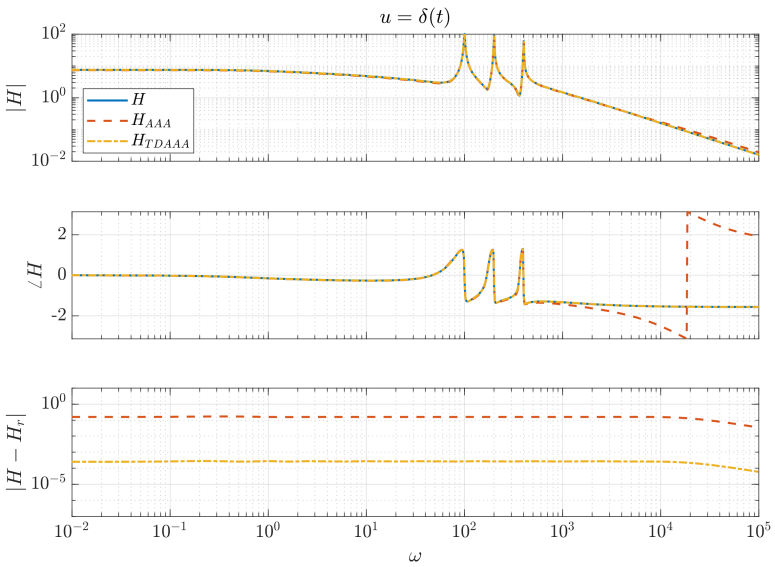
The system response to an impulse and to a sinusoidal input are compared

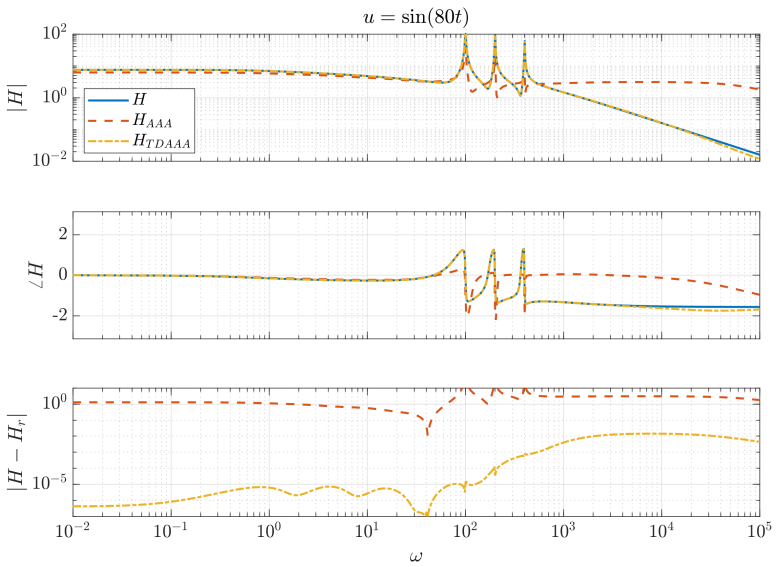




KU LEUVEN

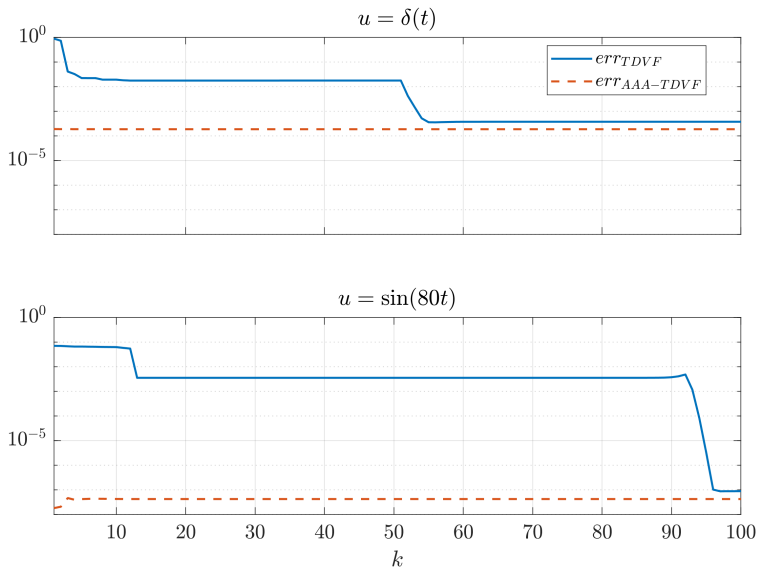
The time data highly improve the MOR accuracy

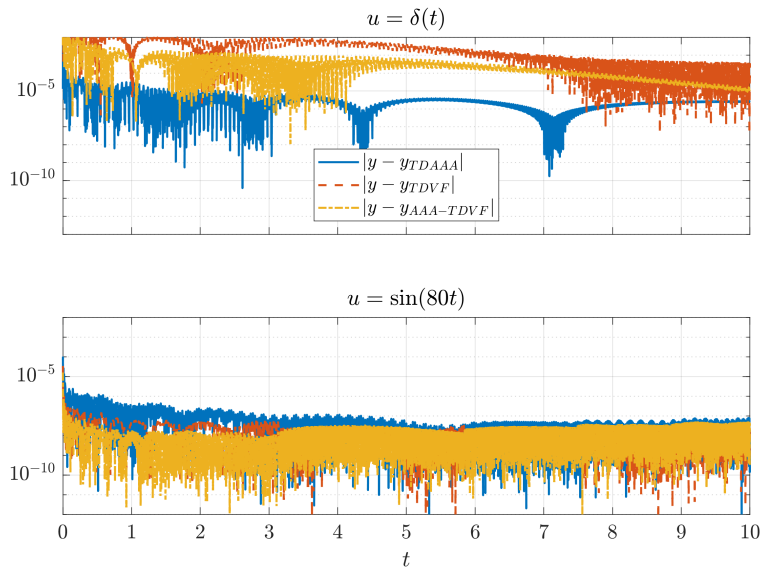




KU LEUVEN

The AAA inizialization improves the TDVF accuracy





KU LEUVEN

3 Outline

1. Time-domain MOR algorithms
2. TDAAA
3. Extension to nonlinear systems

KU LEUVEN

Consider weakly nonlinear SISO systems of the form

$$\begin{cases} \dot{\vec{x}}(t) = A\vec{x}(t) + F(t, \vec{x}) + Bu(t) \\ y(t) = C\vec{x}(t) \end{cases} .$$

3 Frequency response functions

For nonlinear systems the transfer function is not defined. Series of functions are employed to represent the output instead.

- **Volterra series** (or Generalized Frequency Response Functions): series of multivariate functions

$$\begin{cases} Y(j\omega) = \sum_{n=1}^{+\infty} Y_n(j\omega) \\ Y_n(j\omega) = \frac{1/\sqrt{n}}{2\pi^{n-1}} \int_{\omega_1+\cdots+\omega_n=\omega} H_n(\omega_1,\ldots,\omega_n) \prod_{i=1}^{n} U(j\omega_i) d\sigma_\omega, \end{cases}$$

with

$$H_n(\omega_1,\ldots,\omega_n) = \int_{-\infty}^{+\infty} \cdots \int_{-\infty}^{+\infty} h_n(\tau_1,\ldots,\tau_n) \, e^{-(\omega_1\tau_1+\cdots+\omega_n\tau_n)} d\tau_1 \cdots d\tau_n$$

KU LEUVEN

3 Frequency response functions

For nonlinear systems the transfer function is not defined. Series of functions are employed to represent the output instead.

- **Nonlinear Output Frequency Response Functions**: one-dimensional functions, but dependant on the input

$$Y(j\omega) = \sum_{n=1}^{+\infty} G_n(j\omega) U_n(j\omega)$$

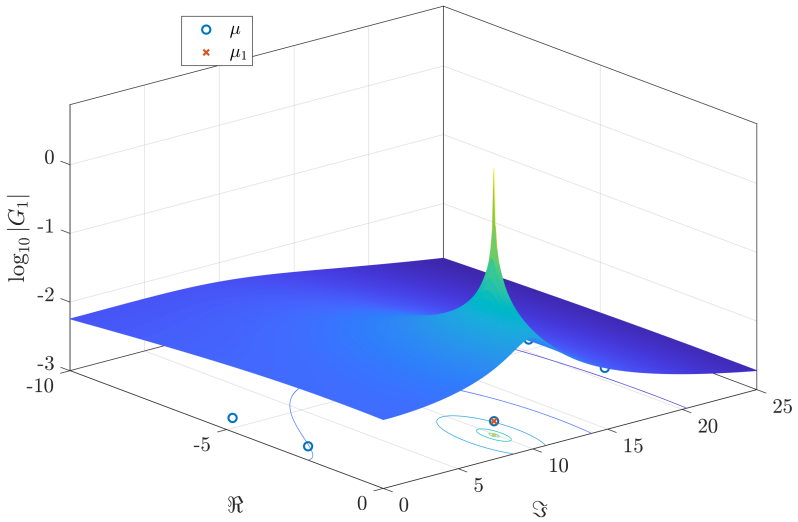
with

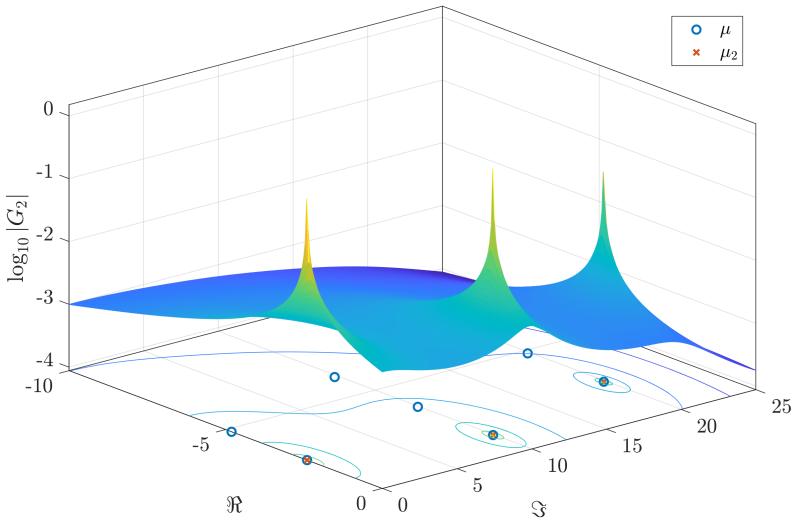
$$G_n(j\omega) = \frac{\int_{\omega_1+\cdots+\omega_n=\omega} H_n(\omega_1, \ldots, \omega_n) \prod_{i=1}^{n} U(j\omega_i) d\sigma_\omega}{\int_{\omega_1+\cdots+\omega_n=\omega} \prod_{i=1}^{n} U(j\omega_i) d\sigma_\omega}.$$

3 Impulse response

AAA can recover the poles of $\sum_{n=1}^{\infty} G_n(j\omega)$ by approximating the system's impulse response. Consider the GFRFs of the system

$$m\ddot{x} + c\dot{x} + k_1 x + k_2 x^2 = u(t)$$





KU LEUVEN

3 Periodic input

For a periodic input $u(t) = \sum_{k=-\infty}^{+\infty} c_k e^{jk\omega t}$, the VS in the time domain becomes

$$y(t) = \sum_{k=-\infty}^{+\infty} d_k e^{jk\omega t},$$

with

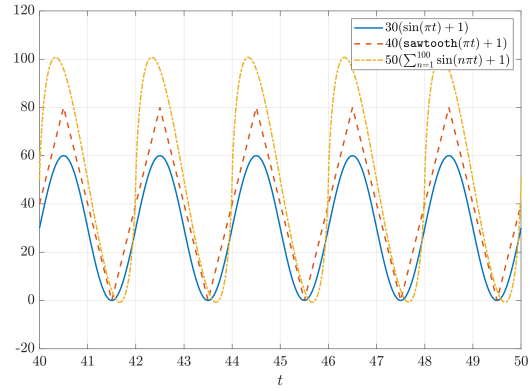
$$d_k = \sum_{n=1}^{+\infty} \sum_{\substack{k_1,\ldots,k_n=-\infty \\ k_1+\cdots+k_n=k}}^{+\infty} c_{k_1} \cdots c_{k_n} H_n(jk_1\omega, \ldots, jk_n\omega).$$

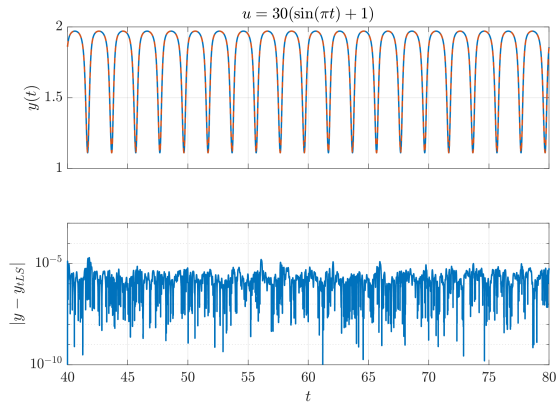
By applying TDAAA it is possible to find a ROM which can be applied to all functions with the same period.

We consider the Chafee-Infante equation

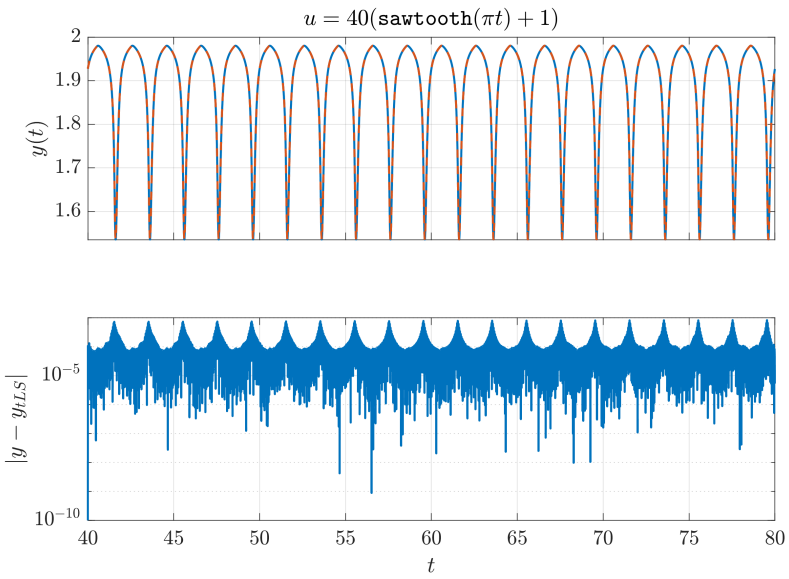
$$\begin{cases} c_t(x,t) = c_{xx}(x,t) + \\ \qquad\qquad c(x,t) - c^3(x,t) \\ c(0,t) = u(t) \\ y(t) = c(1,t) \end{cases},$$

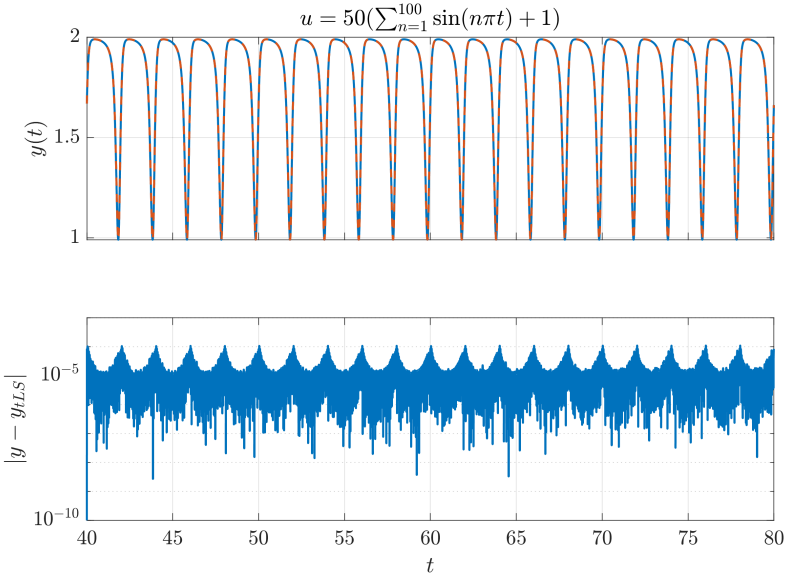
discretized in space





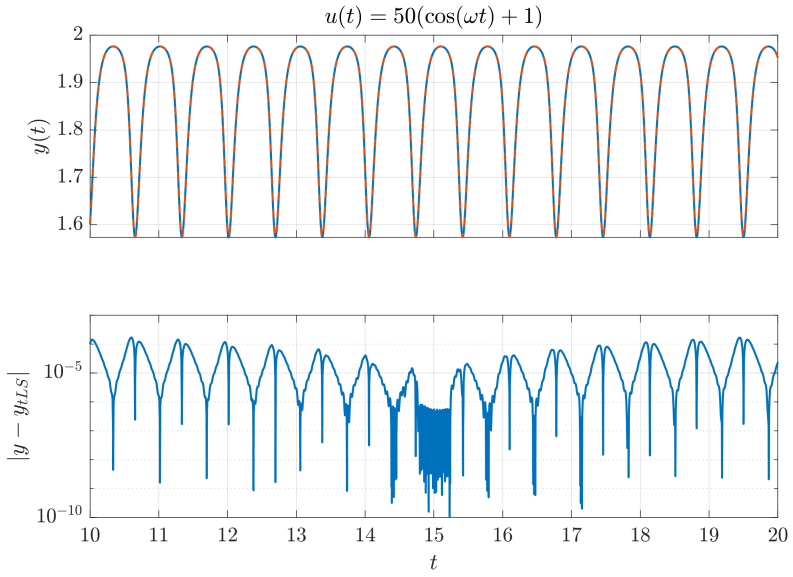
This ROM can be used for any periodic input with the same period
The coefficients f_j have to be recalculated using the first time steps





3 ROM interpolation

Once ROMs for multiple frequencies inputs are defined, they can be interpolated.
We use the `blockAAA` (Gosea et al. 2021a) to generate the new ROM.



3 Conclusion

To sum up,

- Defined TDAAA, a new AAA variant working with time-domain data
- For LTI systems, TDAAA is accurate and can be combined with TDVF to improve its accuracy and convergence speed
- The TDAAA can also be applied to nonlinear systems subject to periodic inputs to generate parametric ROMs

In the future we will work to

- Extend to MIMO systems
- Include transient behaviour in the nonlinear case
- Test other ROM interpolation techniques

Thank you for your attention!

4 Bibliography I

[1] A. C. Antoulas and B. D. Q. Anderson. "On the Scalar Rational Interpolation Problem". In: *IMA Journal of Mathematical Control and Information* 3.2 (1986), pp. 61–88. ISSN: 0265-0754, 1471-6887. DOI: 10.1093/imamci/3.2-3.61. URL: https://academic.oup.com/imamci/article-lookup/doi/10.1093/imamci/3.2-3.61 (visited on 01/04/2022).

[2] Quirin Aumann and Ion Victor Gosea. *Practical challenges in data-driven interpolation: dealing with noise, enforcing stability, and computing realizations*. Jan. 12, 2023. arXiv: 2301.04906[cs,eess,math]. URL: http://arxiv.org/abs/2301.04906 (visited on 03/27/2023).

[3] Stefano Costa and Lloyd N. Trefethen. "AAA-least squares rational approximation and solution of Laplace problems". In: *arXiv:2107.01574 [cs, math]* (July 4, 2021). arXiv: 2107.01574. URL: http://arxiv.org/abs/2107.01574 (visited on 01/04/2022).

[4] Elke Deckers, Stijn Jonckheere, and Karl Meerbergen. *Time integration of finite element models with nonlinear frequency dependencies*. Number: arXiv:2206.09617. June 20, 2022. arXiv: 2206.09617[cs,math]. URL: http://arxiv.org/abs/2206.09617 (visited on 06/22/2022).

[5] D. Deschrijver et al. "Macromodeling of Multiport Systems Using a Fast Implementation of the Vector Fitting Method". In: *IEEE Microwave and Wireless Components Letters* 18.6 (June 2008), pp. 383–385. ISSN: 1531-1309, 1558-1764. DOI: 10.1109/LMWC.2008.922585. URL: http://ieeexplore.ieee.org/document/4530747/ (visited on 06/11/2023).

[6] Ion Victor Gosea and Athanasios C. Antoulas. "Data-driven model order reduction of quadratic-bilinear systems". In: *Numerical Linear Algebra with Applications* 25.6 (Dec. 2018). ISSN: 1070-5325, 1099-1506. DOI: 10.1002/nla.2200. URL: https://onlinelibrary.wiley.com/doi/10.1002/nla.2200 (visited on 03/03/2022).

[7] Ion Victor Gosea and Serkan Gugercin. *The AAA framework for modeling linear dynamical systems with quadratic output*. May 20, 2020. arXiv: 2005.10316[cs,eess,math]. URL: http://arxiv.org/abs/2005.10316 (visited on 06/11/2023).

[8] Ion Victor Gosea and Stefan Güttel. "Algorithms for the Rational Approximation of Matrix-Valued Functions". In: *SIAM Journal on Scientific Computing* 43.5 (Jan. 2021), A3033–A3054. ISSN: 1064-8275, 1095-7197. DOI: 10.1137/20M1324727. URL: https://epubs.siam.org/doi/10.1137/20M1324727 (visited on 02/10/2022).

4 Bibliography II

[9] Ion Victor Gosea, Charles Poussot-Vassal, and Athanasios C. Antoulas. "On enforcing stability for data-driven reduced-order models". In: *2021 29th Mediterranean Conference on Control and Automation (MED)*. 2021 29th Mediterranean Conference on Control and Automation (MED). PUGLIA, Italy: IEEE, June 22, 2021, pp. 487–493. ISBN: 978-1-66542-258-1. DOI: 10.1109/MED51440.2021.9480216. URL: https://ieeexplore.ieee.org/document/9480216/ (visited on 11/22/2022).

[10] Ion Victor Gosea, Qiang Zhang, and Athanasios C. Antoulas. "Data-driven modeling from noisy measurements". In: *PAMM* 20 (S1 Mar. 2021). ISSN: 1617-7061, 1617-7061. DOI: 10.1002/pamm.202000358. URL: https://onlinelibrary.wiley.com/doi/10.1002/pamm.202000358 (visited on 10/26/2022).

[11] B. Gustavsen. "Improving the Pole Relocating Properties of Vector Fitting". In: *IEEE Transactions on Power Delivery* 21.3 (July 2006), pp. 1587–1592. ISSN: 0885-8977. DOI: 10.1109/TPWRD.2005.860281. URL: http://ieeexplore.ieee.org/document/1645204/ (visited on 02/23/2022).

[12] B. Gustavsen and A. Semlyen. "Rational approximation of frequency domain responses by vector fitting". In: *IEEE Transactions on Power Delivery* 14.3 (July 1999), pp. 1052–1061. ISSN: 08858977. DOI: 10.1109/61.772353. URL: http://ieeexplore.ieee.org/document/772353/ (visited on 12/28/2021).

[13] Daan Huybrechs and Lloyd N. Trefethen. "AAA interpolation of equispaced data". In: *BIT Numerical Mathematics* 63.2 (June 2023), p. 21. ISSN: 0006-3835, 1572-9125. DOI: 10.1007/s10543-023-00959-x. URL: https://link.springer.com/10.1007/s10543-023-00959-x (visited on 06/09/2023).

[14] Yuji Nakatsukasa, Olivier Sète, and Lloyd N. Trefethen. "The AAA algorithm for rational approximation". In: *SIAM Journal on Scientific Computing* 40.3 (Jan. 2018), A1494–A1522. ISSN: 1064-8275, 1095-7197. DOI: 10.1137/16M1106122. arXiv: 1612.00337. URL: http://arxiv.org/abs/1612.00337 (visited on 12/28/2021).

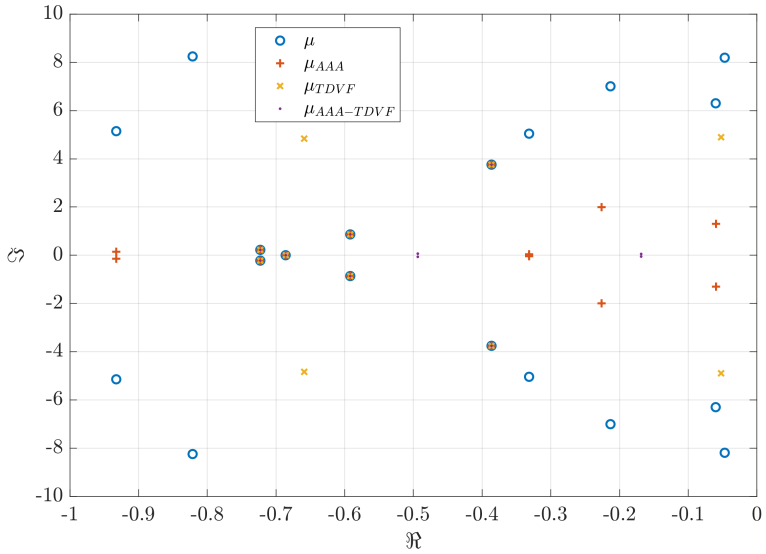
[15] Benjamin Peherstorfer, Serkan Gugercin, and Karen Willcox. "Data-Driven Reduced Model Construction with Time-Domain Loewner Models". In: *SIAM Journal on Scientific Computing* 39.5 (Jan. 2017), A2152–A2178. ISSN: 1064-8275, 1095-7197. DOI: 10.1137/16M1094750. URL: https://epubs.siam.org/doi/10.1137/16M1094750 (visited on 12/28/2021).

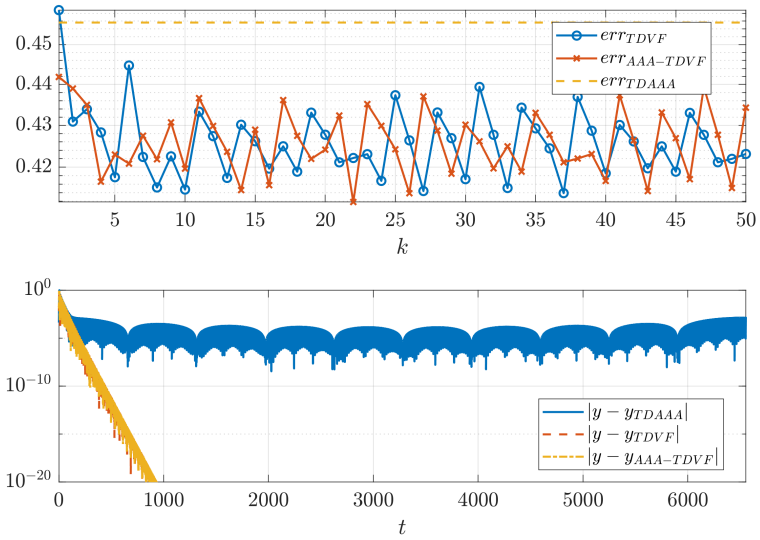
4 Bibliography III

[16] Andrea Carracedo Rodriguez and Serkan Gugercin. "The p-AAA algorithm for data driven modeling of parametric dynamical systems". In: *arXiv:2003.06536 [cs, eess, math]* (Mar. 18, 2020). arXiv: 2003.06536. URL: http://arxiv.org/abs/2003.06536 (visited on 12/28/2021).

4 Sampling

The sampling should be considered carefully, both for TDAAA and TDVF

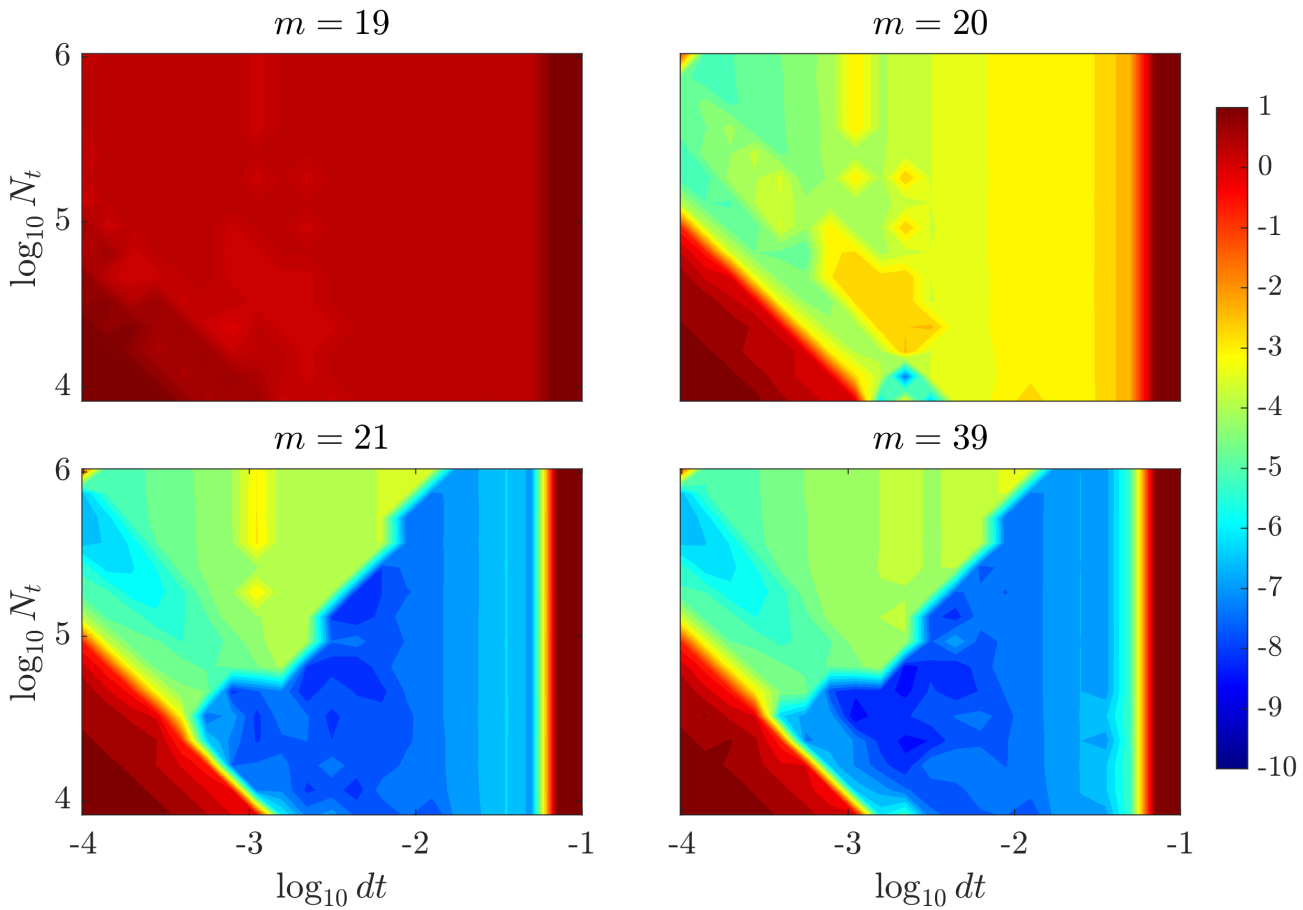




KU LEUVEN

4 Sampling

The sampling should be considered carefully, both for TDAAA and TDVF



KU LEUVEN