




A tangential interpolation framework for the AAA algorithm

Quirin Aumann^{1,2} | Peter Benner^{1,2} | Ion Victor Gosea²  | Jens Saak^{1,2}  | Julia Vettermann¹ 

¹Research Group Mathematics in Industry and Technology, University of Technology Chemnitz, Chemnitz, Germany

²Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg, Germany

Correspondence

Ion Victor Gosea, Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstr. 1, 39106 Magdeburg, Germany. Email: gosea@mpi-magdeburg.mpg.de

Funding information

German Research Foundation, Grant/Award Number: 174223256—TRR 96

Abstract

Interpolation-based data-driven methods, such as the Loewner framework or the Adaptive Antoulas-Anderson (AAA) algorithm, are established and effective approaches to find a realization of a dynamical system from frequency response data (measurements of the system's transfer function). If a system-theoretic representation of the original model is not available or unfeasible to evaluate efficiently, such reduced realizations enable effective analysis and simulation. This is especially relevant for models of interconnected dynamical systems, which typically have a high number of inputs and outputs to model their coupling conditions correctly.

Tangential interpolation is an established strategy to construct accurate reduced-order models while ensuring a reasonably small size even if many inputs and/or outputs have to be considered. In this contribution, we evaluate the applicability and effectiveness of data-driven interpolation methods to compute reduced-order models of dynamical systems with many inputs and outputs. Additionally, we extend AAA to a tangential interpolation setting and thus enable the use of AAA-like methods in the context of transfer function interpolation for systems with many inputs and outputs.

1 | INTRODUCTION

Simulation, control, and optimization of complex dynamical systems are required for many engineering applications. In the following, we consider a multiple-input multiple-output (MIMO) linear time-invariant (LTI) system Σ of order n described by the following system of differential equations:

$$\Sigma : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), & \mathbf{y}(t) = \mathbf{C}\mathbf{x}(t), \end{cases} \quad (1)$$

with states $\mathbf{x}(t) \in \mathbb{R}^n$, control inputs $\mathbf{u}(t) \in \mathbb{R}^m$, observed outputs $\mathbf{y}(t) \in \mathbb{R}^p$, state matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$, and input and output mappings $\mathbf{B} \in \mathbb{C}^{n \times m}$ and $\mathbf{C} \in \mathbb{C}^{p \times n}$. The (matrix-valued) transfer function of the LTI system in frequency (Laplace)

This is an open access article under the terms of the [Creative Commons Attribution](https://creativecommons.org/licenses/by/4.0/) License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2023 The Authors. *Proceedings in Applied Mathematics & Mechanics* published by Wiley-VCH GmbH.

domain is given by $\mathbf{H}(s) \in \mathbb{C}^{p \times n}$, with $s \in \mathbb{C}$, as

$$\mathbf{H}(s) = \mathbf{C}(s\mathbf{I}_n - \mathbf{A})^{-1}\mathbf{B}, \quad (2)$$

where \mathbf{I}_n is an identity matrix of size n . In many applications, n is very large, hindering an efficient evaluation of Equation 1 or Equation 2. Model order reduction methods seek approximations to the original dynamical system, whose output $\hat{\mathbf{y}}(t)$ matches the original output up to a specified tolerance τ in an appropriate norm:

$$\|\mathbf{y}(t) - \hat{\mathbf{y}}(t)\| \leq \tau \cdot \|\mathbf{u}(t)\|.$$

A similar relation can be formulated for the transfer function Equation 2.

If the dynamical system of interest is not given in state-space form Equation 1 or the state-space form cannot be evaluated conveniently, data-driven reduction or identification methods can be employed to obtain a reduced-order (surrogate) model which approximates data obtained from the original system. Some of these methods are based on interpolation [1, 2] or optimization [3, 4].

The Adaptive Antoulas-Anderson (AAA) algorithm [5] has emerged as a versatile tool for data-driven interpolation, combining ideas from interpolation and least-squares optimization. Originally a method for the interpolation of scalar functions, it has been extended to matrix functions in [6] by incorporating a block-wise approach. In the context of model order reduction and system inference, that is, when the matrix transfer function Equation 2 is to be interpolated from data samples, the block approach is limiting the effectiveness of the resulting surrogate model. Especially if systems with many inputs and outputs ($m, p \geq 50$) are considered, the resulting realizations quickly grow too large to be an efficient representation of the original system. In this contribution, we therefore modify AAA with ideas from tangential interpolation to be able to compute compact surrogates of the provided data. Tangential interpolation is widely used in projection-based model order reduction [7].

The remainder of the paper is structured as follows. We shortly review data-driven interpolation in the Loewner framework in Section 2, as it is the foundation of the proposed method. The theoretical foundation for the method is laid out in Section 3 and the resulting algorithm is presented in Section 4. We show the performance of the new method in numerical experiments in Section 5, and conclude the paper in Section 6.

2 | DATA-DRIVEN INTERPOLATION IN THE LOEWNER FRAMEWORK

The Loewner framework [1] is an effective and established methodology that uses only input and output data, that is, transfer function measurements or evaluations, to find a surrogate model, which interpolates the transfer function of the original system Equation 2, and provides a realization of this interpolant as a state-space system. The general procedure is summarized, following [8].

Given N measurements $\mathbf{H}_k \in \mathbb{C}^{p \times m}$, $k = 1, \dots, N$, of the transfer function at some locations $s_k \in \mathbb{C}$, the data are partitioned into two disjoint sets:

$$\begin{cases} (\lambda_i, \mathbf{r}_i, \mathbf{w}_i), \text{ where } \lambda_i = s_i, \mathbf{w}_i = \mathbf{H}_i \mathbf{r}_i, i = \{1, \dots, \rho\}, \\ (\mu_j, \boldsymbol{\ell}_j, \mathbf{v}_j), \text{ where } \mu_j = s_{\rho+j}, \mathbf{v}_j^H = \boldsymbol{\ell}_j^H \mathbf{H}_{\rho+j}, j = \{1, \dots, \nu\}, \end{cases}$$

with $N = \rho + \nu$ and right and left tangential directions $\mathbf{r}_i, \boldsymbol{\ell}_j$. For numerical reasons, it is often beneficial to partition the data in an alternating way. The partitioned data are now arranged in the Loewner and shifted Loewner matrices \mathbb{L} and \mathbb{L}_σ given by

$$\mathbb{L}^{(i,j)} = \frac{\mathbf{v}_i \mathbf{r}_j - \boldsymbol{\ell}_i \mathbf{w}_j}{\mu_i - \lambda_j}, \quad \mathbb{L}_\sigma^{(i,j)} = \frac{\mu_i \mathbf{v}_i \mathbf{r}_j - \lambda_j \boldsymbol{\ell}_i \mathbf{w}_j}{\mu_i - \lambda_j}. \quad (3)$$

If the matrix pencil $(\mathbb{L}_\sigma, \mathbb{L})$ is regular, $\mathbf{H}_r(z) = \mathbb{W}(\mathbb{L}_\sigma - z\mathbb{L})^{-1}\mathbb{V}$ tangentially interpolates the given data, such that $\mathbf{H}_r(\lambda_i)\mathbf{r}_i = \mathbf{w}_i$, $\mathbf{1}_j^H \mathbf{H}_r(\mu_j) = \mathbf{v}_j^H$. A state-space realization of the surrogate is then given by

$$\mathbf{E}_\mathbb{L} = -\mathbb{L}, \quad \mathbf{A}_\mathbb{L} = -\mathbb{L}_\sigma, \quad \mathbf{B}_\mathbb{L} = \mathbb{V} = [\mathbf{v}_1^H, \dots, \mathbf{v}_\nu^H]^H, \quad \mathbf{C}_\mathbb{L} = \mathbb{W} = [\mathbf{w}_1, \dots, \mathbf{w}_\rho]. \quad (4)$$

In practical applications (when processing a fairly large number of measurements), the pencil $(\mathbb{L}_\sigma, \mathbb{L})$ is typically singular or simply too large, so a post-processing step is required for the Loewner model Equation 4. Performing a singular value decomposition (SVD) of the pencil extracts the dominant features and removes inherent redundancies in the data. The left (\mathbf{Y}) and right (\mathbf{X}) projection matrices are taken from the truncated SVDs

$$[\mathbb{L} \ \mathbb{L}_s] \approx \mathbf{Y} \mathbf{S}_r^{(1)} \mathbf{X}^H, \quad \begin{bmatrix} \mathbb{L} \\ \mathbb{L}_s \end{bmatrix} \approx \tilde{\mathbf{Y}} \mathbf{S}_r^{(2)} \mathbf{X}^H, \quad (5)$$

where $\mathbf{S}_r^{(1)}, \mathbf{S}_r^{(2)} \in \mathbb{R}^{r \times r}$, $\mathbf{Y} \in \mathbb{C}^{\rho \times r}$, $\mathbf{X} \in \mathbb{C}^{\nu \times r}$, $\tilde{\mathbf{Y}} \in \mathbb{C}^{2\nu \times r}$, $\tilde{\mathbf{X}} \in \mathbb{C}^{r \times 2\rho}$. The truncation index r can, for example, be chosen as the numerical rank (based on a tolerance value $\tau > 0$). The projected Loewner system is then given by

$$\hat{\mathbf{E}}_\mathbb{L} = \mathbf{Y}^H \mathbf{E}_\mathbb{L} \mathbf{X}, \quad \hat{\mathbf{A}}_\mathbb{L} = \mathbf{Y}^H \mathbf{A}_\mathbb{L} \mathbf{X}, \quad \hat{\mathbf{B}}_\mathbb{L} = \mathbf{Y}^H \mathbf{B}_\mathbb{L}, \quad \hat{\mathbf{C}}_\mathbb{L} = \mathbf{C}_\mathbb{L} \mathbf{X}. \quad (6)$$

3 | ONE-SIDED TANGENTIAL INTERPOLATION AND ITS BARYCENTRIC FORMS

Assume, we want to interpolate the transfer function at k interpolation points, that is, $\lambda_1, \dots, \lambda_k$. In what follows, the tangential directional vectors $\mathbf{r}_i \in \mathbb{C}^m$ for $1 \leq i \leq k$ are included explicitly—this will be used to lower the dimension of the computed realizations. As opposed to full-matrix element-wise interpolation, we employ tangential interpolation. Note that the approach is equivalent to the full-matrix case, if the tangential directions are chosen as identity matrices.

We introduce the generalized reachability matrix with tangential directions, as

$$\mathcal{R} = \begin{bmatrix} (\lambda_1 \mathbf{I} - \mathbf{A})^{-1} \mathbf{B} \mathbf{r}_1 & (\lambda_2 \mathbf{I} - \mathbf{A})^{-1} \mathbf{B} \mathbf{r}_2 & \cdots & (\lambda_k \mathbf{I} - \mathbf{A})^{-1} \mathbf{B} \mathbf{r}_k \end{bmatrix} \in \mathbb{C}^{n \times k}, \quad (7)$$

corresponding to enforcing explicit interpolation, that is, matching of the following measurements:

$$\mathbf{H}(\lambda_1) \mathbf{r}_1, \mathbf{H}(\lambda_2) \mathbf{r}_2, \dots, \mathbf{H}(\lambda_k) \mathbf{r}_k. \quad (8)$$

Here, the tangential vectors $\mathbf{r}_i \in \mathbb{C}^{m \times 1}$ are the so-called “tangential directions.” The matrix $\mathcal{R} \in \mathbb{C}^{n \times k}$ solves a linear Sylvester equation $\mathbf{A} \mathcal{R} + \mathbf{B} \mathbf{R} = \mathcal{R} \mathbf{\Lambda}$, where the matrix pair $(\mathbf{\Lambda}, \mathbf{R})$ explicitly determines the interpolation conditions enforced, that is, they are written as follows: $\mathbf{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_k) \in \mathbb{C}^{k \times k}$, $\mathbf{R} = [\mathbf{r}_1 \ \cdots \ \mathbf{r}_k] \in \mathbb{C}^{m \times k}$.

Additionally, we consider so-called weight vectors $\mathbf{w}_i \in \mathbb{C}^m$ for all $1 \leq i \leq k$. For the moment, their entries will be considered as free parameters. As it will become evident in the subsequent sections, fixing them will be done in accordance with accommodating additional tangential interpolation conditions.

One-sided parametric realization: we show that an LTI parameterized model of dimension $r = k$ can be constructed, having km degrees of freedom, to enforce one-sided interpolation in a realization-free manner. This means that no access to matrices \mathbf{A} , \mathbf{B} , and \mathbf{C} is required. The free parameters are given by the entries of matrix $\hat{\mathbf{B}}$ stored in the weight vectors $\hat{\mathbf{w}}_i$'s.

Lemma 3.1. *The transfer function of the reduced-order model (ROM) described by matrices above:*

$$\hat{\mathbf{A}} = \mathbf{\Lambda} - \hat{\mathbf{B}} \mathbf{R} \in \mathbb{R}^{k \times k}, \quad \hat{\mathbf{B}} = \begin{bmatrix} \hat{\mathbf{w}}_1^T \\ \vdots \\ \hat{\mathbf{w}}_k^T \end{bmatrix} \in \mathbb{R}^{k \times m}, \quad \hat{\mathbf{C}} = \mathbf{C} \mathcal{R} = [\mathbf{H}(\lambda_1) \mathbf{r}_1 \ \cdots \ \mathbf{H}(\lambda_k) \mathbf{r}_k] \in \mathbb{R}^{p \times k}, \quad (9)$$

given by:

$$\hat{\mathbf{H}}(s) = \hat{\mathbf{C}}(s\mathbf{I} - \hat{\mathbf{A}})^{-1} \hat{\mathbf{B}} = \hat{\mathbf{C}}(s\mathbf{I} - \mathbf{\Lambda} + \hat{\mathbf{B}} \mathbf{R})^{-1} \hat{\mathbf{B}}, \quad (10)$$

satisfies the following tangential interpolation conditions, with respect to the data in Equation 8, corresponding to the original system:

$$\hat{H}(\lambda_1)\mathbf{r}_1 = \mathbf{H}(\lambda_1)\mathbf{r}_1, \hat{H}(\lambda_2)\mathbf{r}_2 = \mathbf{H}(\lambda_2)\mathbf{r}_2, \dots, \hat{H}(\lambda_k)\mathbf{r}_k = \mathbf{H}(\lambda_k)\mathbf{r}_k. \quad (11)$$

Proof. The proof is quite straightforward, and it is based on a simple result. Let $1 \leq i \leq k$. Then, by substituting s with λ_i , and by multiplying the resolvent $s\mathbf{I} - \hat{\mathbf{A}}$ with the i th unit vector \mathbf{e}_i from the right, we can write that:

$$(\lambda_i\mathbf{I} - \mathbf{\Lambda} + \hat{\mathbf{B}}\mathbf{R})\mathbf{e}_i = \hat{\mathbf{B}}\mathbf{r}_i \Rightarrow (\lambda_i\mathbf{I} - \mathbf{\Lambda} + \hat{\mathbf{B}}\mathbf{R})^{-1}\hat{\mathbf{B}}\mathbf{r}_i = \mathbf{e}_i. \quad (12)$$

Finally, by multiplying the above equality with $\hat{\mathbf{C}}$ to the left, we obtain that:

$$\hat{\mathbf{C}}(\lambda_i\mathbf{I} - \mathbf{\Lambda} + \hat{\mathbf{B}}\mathbf{R})^{-1}\hat{\mathbf{B}}\mathbf{r}_i = \hat{\mathbf{C}}\mathbf{e}_i = \mathbf{H}(\lambda_i)\mathbf{r}_i \Rightarrow \hat{H}(\lambda_i)\mathbf{r}_i = \mathbf{H}(\lambda_i)\mathbf{r}_i. \quad (13)$$

□

Next, we show how to express the transfer function of the parameterized model that satisfies the k right interpolation conditions above, in a barycentric format. This is a necessary step for extending the classical AAA algorithm, to the tangential directions format, that is, in developing our tangential AAA algorithm.

Lemma 3.2. With $\mathbf{\Lambda}_s = s\mathbf{I}_k - \mathbf{\Lambda}$, the transfer function Equation 10 of the one-sided interpolatory ROM in Equation 9, can be written in barycentric form as:

$$\hat{H}(s) = \hat{\mathbf{C}}\mathbf{\Lambda}_s^{-1}\hat{\mathbf{B}}(\mathbf{I}_m + \mathbf{R}\mathbf{\Lambda}_s^{-1}\hat{\mathbf{B}})^{-1} = \left(\sum_{i=1}^k \frac{\mathbf{H}(\lambda_i)\mathbf{r}_i\hat{\mathbf{w}}_i^T}{s - \lambda_i} \right) \left(\mathbf{I}_m + \sum_{i=1}^k \frac{\mathbf{r}_i\hat{\mathbf{w}}_i^T}{s - \lambda_i} \right)^{-1}. \quad (14)$$

Proof. To formally prove this result, we will make use of the Sherman–Morrison–Woodbury matrix inversion formula (e.g., [9])

$$(\hat{\mathbf{M}} + \hat{\mathbf{U}}\hat{\mathbf{V}})^{-1} = \hat{\mathbf{M}}^{-1} - \hat{\mathbf{M}}^{-1}\hat{\mathbf{U}}(\mathbf{I}_k + \hat{\mathbf{V}}\hat{\mathbf{M}}^{-1}\hat{\mathbf{U}})^{-1}\hat{\mathbf{V}}\hat{\mathbf{M}}^{-1}. \quad (15)$$

The particular structure of the ROM allows the resolvent $s\mathbf{I}_k - \hat{\mathbf{A}}$ to be rewritten as follows:

$$\hat{\mathbf{A}} = \mathbf{\Lambda} - \hat{\mathbf{B}}\mathbf{R} \Rightarrow s\mathbf{I}_k - \hat{\mathbf{A}} = s\mathbf{I}_k - \mathbf{\Lambda} + \hat{\mathbf{B}}\mathbf{R} = s\mathbf{I}_k - \mathbf{\Lambda} + \hat{\mathbf{B}}\mathbf{R}. \quad (16)$$

Using Equation 15, the transfer function $\hat{H}(s)$ is rewritten as:

$$\begin{aligned} \hat{H}(s) &= \hat{\mathbf{C}}(s\mathbf{I}_k - \hat{\mathbf{A}})^{-1}\hat{\mathbf{B}} = \hat{\mathbf{C}}(\mathbf{\Lambda}_s + \hat{\mathbf{B}}\mathbf{R})^{-1}\hat{\mathbf{B}} \\ &= \hat{\mathbf{C}}\mathbf{\Lambda}_s^{-1}\hat{\mathbf{B}} - \hat{\mathbf{C}}\mathbf{\Lambda}_s^{-1}\hat{\mathbf{B}}(\mathbf{I}_m + \mathbf{R}\mathbf{\Lambda}_s^{-1}\hat{\mathbf{B}})^{-1}\mathbf{R}\mathbf{\Lambda}_s^{-1}\hat{\mathbf{B}} \\ &= \hat{\mathbf{C}}\mathbf{\Lambda}_s^{-1}\hat{\mathbf{B}} - \hat{\mathbf{C}}\mathbf{\Lambda}_s^{-1}\hat{\mathbf{B}}(\mathbf{I}_m + \hat{\mathbf{X}})^{-1}\hat{\mathbf{X}} \\ &= \hat{\mathbf{C}}\mathbf{\Lambda}_s^{-1}\hat{\mathbf{B}}\left[\mathbf{I}_m - (\mathbf{I}_m + \hat{\mathbf{X}})^{-1}\hat{\mathbf{X}}\right] = \hat{\mathbf{C}}\mathbf{\Lambda}_s^{-1}\hat{\mathbf{B}}(\mathbf{I}_m + \hat{\mathbf{X}})^{-1}, \end{aligned} \quad (17)$$

where $\hat{\mathbf{X}} = \mathbf{R}\mathbf{\Lambda}_s^{-1}\hat{\mathbf{B}}$. Hence, the transfer function $\hat{H}(s)$ is explicitly written in the required format, that is:

$$\hat{H}(s) = \hat{\mathbf{C}}\mathbf{\Lambda}_s^{-1}\hat{\mathbf{B}}(\mathbf{I}_m + \mathbf{R}\mathbf{\Lambda}_s^{-1}\hat{\mathbf{B}})^{-1}. \quad (18)$$

It follows that

$$\hat{\mathbf{C}}\Lambda_s^{-1}\hat{\mathbf{B}} = \begin{bmatrix} \mathbf{H}(\lambda_1)\mathbf{r}_1(s-\lambda_1)^{-1} & \cdots & \mathbf{H}(\lambda_k)\mathbf{r}_k(s-\lambda_k)^{-1} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{w}}_1^T \\ \vdots \\ \hat{\mathbf{w}}_k^T \end{bmatrix} = \sum_{i=1}^k \frac{\mathbf{H}(\lambda_i)\mathbf{r}_i\hat{\mathbf{w}}_i^T}{s-\lambda_i}. \quad (19)$$

Next, we have that

$$(\mathbf{I}_m + \mathbf{R}\Lambda_s^{-1}\hat{\mathbf{B}})^{-1} = \left(\mathbf{I}_m + \begin{bmatrix} \mathbf{r}_1 & \cdots & \mathbf{r}_k \end{bmatrix} \begin{bmatrix} s-\lambda_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & s-\lambda_k \end{bmatrix}^{-1} \begin{bmatrix} \hat{\mathbf{w}}_1^T \\ \vdots \\ \hat{\mathbf{w}}_k^T \end{bmatrix} \right)^{-1} = \left(\mathbf{I}_m + \sum_{i=1}^k \frac{\mathbf{r}_i\hat{\mathbf{w}}_i^T}{s-\lambda_i} \right)^{-1}. \quad (20)$$

Putting together the last three identities proves Equation 14. \square

4 | THE TANGENTIAL AAA ALGORITHM: τ AAA

The steps depicted in what follows go along the lines of the classical AAA algorithm and its block variant. Therefore, the proposed algorithm inherits the greedy structure of AAA, but we modify the least-squares problem, such that the new barycentric form and the vector-valued weights \mathbf{w}_i are used. In this direction, assume we want to (approximately) enforce h extra left tangential interpolation conditions given by:

$$\mathbf{e}_1^T \hat{\mathbf{H}}(\mu_1) = \mathbf{e}_1^T \mathbf{H}(\mu_1), \quad \mathbf{e}_2^T \hat{\mathbf{H}}(\mu_2) = \mathbf{e}_2^T \mathbf{H}(\mu_2), \quad \dots, \quad \mathbf{e}_h^T \hat{\mathbf{H}}(\mu_h) = \mathbf{e}_h^T \mathbf{H}(\mu_h). \quad (21)$$

For any value of j , so that $1 \leq j \leq h$, we will use the barycentric formula of $\hat{\mathbf{H}}(s)$ in Equation 14, in order to formulate equality constraints that will lead to solving a least-squares problem. Hence, we can write:

$$\begin{aligned} \mathbf{e}_j^T \hat{\mathbf{H}}(\mu_j) = \mathbf{e}_j^T \mathbf{H}(\mu_j) &\Rightarrow \mathbf{e}_j^T \left(\sum_{i=1}^k \frac{\mathbf{H}(\lambda_i)\mathbf{r}_i\hat{\mathbf{w}}_i^T}{\mu_j - \lambda_i} \right) \left(\mathbf{I}_m + \sum_{i=1}^k \frac{\mathbf{r}_i\hat{\mathbf{w}}_i^T}{\mu_j - \lambda_i} \right)^{-1} = \mathbf{e}_j^T \mathbf{H}(\mu_j) \\ &\Rightarrow \mathbf{e}_j^T \left(\sum_{i=1}^k \frac{\mathbf{H}(\lambda_i)\mathbf{r}_i\hat{\mathbf{w}}_i^T}{\mu_j - \lambda_i} \right) = \mathbf{e}_j^T \mathbf{H}(\mu_j) \left(\mathbf{I}_m + \sum_{i=1}^k \frac{\mathbf{r}_i\hat{\mathbf{w}}_i^T}{\mu_j - \lambda_i} \right) \\ &\Rightarrow \left(\sum_{i=1}^k \frac{\mathbf{e}_j^T (\mathbf{H}(\mu_j) - \mathbf{H}(\lambda_i))\mathbf{r}_i}{\mu_j - \lambda_i} \right) \hat{\mathbf{w}}_i^T = -\mathbf{e}_j^T \mathbf{H}(\mu_j). \end{aligned} \quad (22)$$

Next, introduce the following (quasi) Loewner matrix $\mathbb{L} \in \mathbb{C}^{h \times k}$, and also the data matrix $\hat{\mathbf{Z}} \in \mathbb{C}^{h \times m}$ containing the left tangentially interpolated data, and the matrix of variables $\hat{\mathbf{B}} \in \mathbb{C}^{k \times m}$ (previously defined above):

$$(\mathbb{L})_{j,i} = \frac{\mathbf{e}_j^T (\mathbf{H}(\mu_j) - \mathbf{H}(\lambda_i))\mathbf{r}_i}{\mu_j - \lambda_i}, \quad \hat{\mathbf{F}} = \begin{bmatrix} \mathbf{e}_1^T \mathbf{H}(\mu_1) \\ \vdots \\ \mathbf{e}_h^T \mathbf{H}(\mu_h) \end{bmatrix} \in \mathbb{C}^{h \times m}, \quad \hat{\mathbf{B}} = \begin{bmatrix} \hat{\mathbf{w}}_1^T \\ \vdots \\ \hat{\mathbf{w}}_k^T \end{bmatrix} \in \mathbb{C}^{k \times m}. \quad (23)$$

Lemma 4.1. *By imposing the extra h interpolation conditions in Equation (22), we need to solve the following linear system of h equations in km variables written in matrix format, that is, $\mathbb{L}\hat{\mathbf{B}} = -\hat{\mathbf{F}}$.*

Remark 4.2. After solving the system of equations above, for matrix $\hat{\mathbf{B}}$, we put together the realization of the rational interpolant as in Equation 9. As in the single-input single-output (SISO) case, we need to impose that $h \gg km$, so that the Loewner matrix in Equation 23 has more rows than columns. Hence, the system of equations will be solved only approximately.

ALGORITHM 1 The tangential AAA algorithm, τ AAA.

Require: A (discrete) set of sample points $\Gamma \subset \mathbb{C}$ with N points, function $\mathbf{H}(s)$ (or the evaluations of $\mathbf{H}(s)$ on the set Γ , i.e., the sample values), left and right tangential directions $\{\mathbf{e}_i\}_{i=1}^N \in \mathbb{C}^p$ and $\{\mathbf{r}_i\}_{i=1}^N \in \mathbb{C}^m$, and an error tolerance $\epsilon > 0$.

Ensure: A rational approximant $\mathbf{R}(s)$ of order $(n-1, n)$ displayed in a barycentric form, and an equivalent dynamical system Σ in state space form.

- 1: Initialize $j = 1$, $\Gamma^{(1)} \leftarrow \Gamma$, and $\mathbf{R}_0(s) \leftarrow N^{-1} \sum_{i=1}^N \mathbf{H}(\gamma_i)$.
- 2: **while** $\|\mathbf{H}(s) - \mathbf{R}_{j-1}(s)\| > \epsilon$ **do**
- 3: Select a point $z_j \in \Gamma^{(j)}$ for which $\|\mathbf{h}(s) - \mathbf{R}_{j-1}(s)\|$ attains a maximal value, given:

$$\mathbf{R}_{j-1}(s) := \left(\sum_{k=0}^{j-1} \frac{\mathbf{H}_k \mathbf{r}_k \mathbf{w}_k^{(j-1)\top}}{s - z_k} \right) \left(\mathbf{I} + \sum_{k=0}^{j-1} \frac{\mathbf{r}_k \mathbf{w}_k^{(j-1)\top}}{s - z_k} \right)^{-1}.$$

- 4: **if** $\|\mathbf{H}(z_j) - \mathbf{R}_{j-1}(z_j)\| \leq \epsilon$ **then**
- 5: Return \mathbf{R}_{j-1} .
- 6: **else**
- 7: $\mathbf{H}_j \leftarrow \mathbf{H}(z_j)$ and $\Gamma^{(j+1)} \leftarrow \Gamma^{(j)} \setminus \{z_j\}$.
- 8: **end if**
- 9: Find the weights $\mathbf{W}^{(j)} = [\mathbf{w}_1^{(j)}, \dots, \mathbf{w}_j^{(j)}]$ by solving a least squares problem over $z \in \Gamma^{(j+1)}$

$$\sum_{k=0}^j \mathbf{e}_k^\top \mathbf{H}(s) \frac{\mathbf{r}_k \mathbf{w}_k^{(j)\top}}{s - z_k} \approx \sum_{k=0}^j \mathbf{e}_k^\top \mathbf{H}_k \frac{\mathbf{r}_k \mathbf{w}_k^{(j)\top}}{s - z_k} \Leftrightarrow \left(\sum_{k=0}^j \frac{\mathbf{e}_k^\top \mathbf{H}(s) \mathbf{r}_k - \mathbf{e}_k^\top \mathbf{H}_k \mathbf{r}_k}{s - z_k} \right) \mathbf{w}_k^{(j)\top} \approx 0 \Leftrightarrow \mathbb{L}^{(j)} \mathbf{W}^{(j)} = \mathbb{H},$$

where $\mathbb{H} = [\mathbf{H}_1, \dots, \mathbf{H}_j]^\top$. The weights are obtained by solving $\mathbf{W}^{(j)} = \mathbb{L}^\dagger \mathbb{H}$, where \cdot^\dagger denotes the pseudo-inverse.

- 10: $j \leftarrow j + 1$.

11: **end while**

The resulting method in algorithm form is summarized in 1.

5 | NUMERICAL EXPERIMENTS

In the following, we demonstrate the methods discussed above by applying them on two benchmark examples available from the MOR-Wiki.¹ We first consider a MIMO system with a moderate amount of inputs and outputs to be able to compare τ AAA to b AAA, the block variant of AAA from [6]. We also compute surrogate models using the Loewner framework in combination with an SVD (SVD-LF), as outlined in Section 2. The full potential of the tangential approach is afterward shown by applying it to compute a surrogate model for a dynamical system with $m = p = 89$ inputs and outputs. The numerical experiments have been conducted on a laptop equipped with an AMD Ryzen 7 PRO 5850U and 12 GB RAM running Linux Mint 21 as operating system. All algorithms have been implemented and run with MATLAB R2021b Update 2 (9.11.0.1837725).

5.1 | International Space Station

This system models the structural response of the Russian Service Module of the International Space Station (ISS) [10]. The model has $n = 270$ states, $m = 3$ inputs, and $p = 3$ outputs. The dataset used for the computations contains transfer function measurements at 400 logarithmically distributed points in the range $\omega = [10^{-1}, 10^2] \cdot \mathbf{i}$. We employ τ AAA, b AAA, and SVD-LF to compute real-valued surrogate models with order $r = 60$ each. The transfer functions of the original model and its surrogates are given in Figure 1a, the corresponding point-wise relative errors are plotted in Figure 1b.

¹ <http://modelreduction.org>

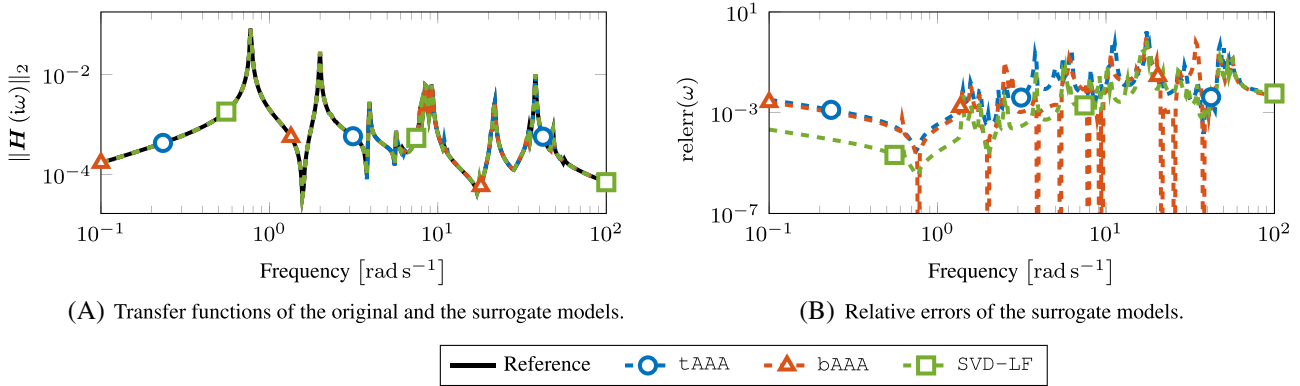


FIGURE 1 Sigma plots of the transfer functions and relative errors of reduced-order models of the ISS model. ISS, International Space Station.

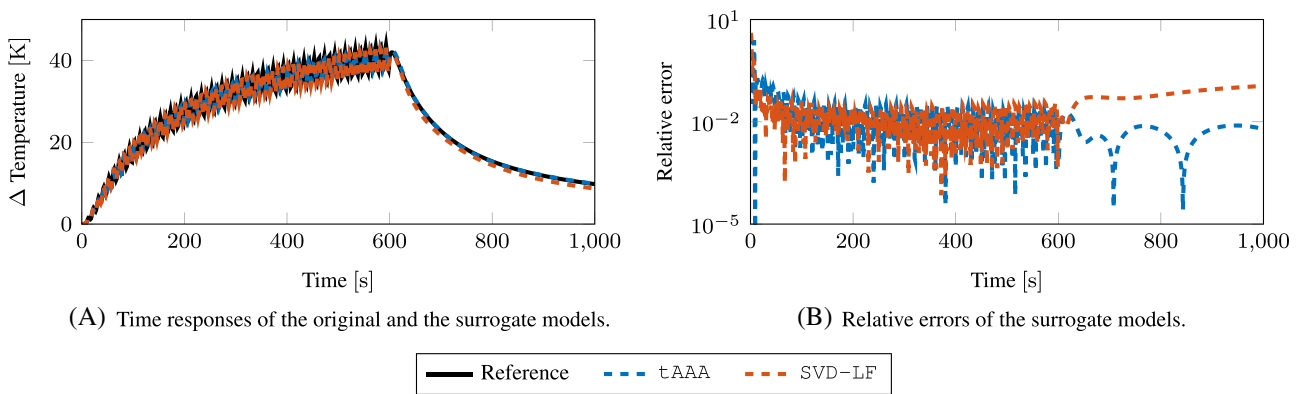


FIGURE 2 Time responses and corresponding relative errors of surrogate models approximating the machine tool model.

All methods yield models with comparable accuracy, while the model obtained from SVD-LF has the lowest error and the model obtained from bAAA is least accurate. The slightly better accuracy of SVD-LF could be expected, as the SVD step considers all available data at once, while for the AAA variants, the iterative nature adds individual data points one by one.

5.2 | Machine tool model

We now consider the model of a generic 3-axis machine tool with time-varying coupling conditions. Using a technique from [11], the coupling can be reformulated to a superposition of precomputed input and output mappings and therefore the model can be formulated as an LTI dynamical system. The model consists of two components ($n_1 = 91181$, $m_1 = 89$, $p_1 = 89$; $n_2 = 74392$, $m_2 = 8$, $p_2 = 8$), which move relative to each other. More details about the system and the modeling process can be found in [12]. Due to the high number of inputs and outputs in the first subsystem, bAAA cannot be applied here. The dataset used as input for τ AAA and SVD-LF consists of the sampled transfer function at 200 logarithmically distributed frequencies in $\omega = [1 \times 10^{-6}, 1 \times 10^{-2}] \cdot i$. We set the tolerance for τ AAA to $\epsilon = 1 \times 10^{-6}$ and truncate all states of the system obtained from the Loewner framework that correspond to singular values $\sigma < \sigma_{\max} \cdot 10^{-14}$. The surrogate model is investigated in the time domain, so all unstable poles have to be truncated in a post-processing step [12, 13]. The resulting models are of orders $r_1^{\tau\text{AAA}} = 157$, $r_2^{\tau\text{AAA}} = 117$ for τ AAA, and $r_1^{\text{SVD-LF}} = 112$, $r_2^{\text{SVD-LF}} = 69$ for SVD-LF. The time responses of the original system and the two reduced-order models are given in Figure 2a, the corresponding relative errors are plotted in Figure 2b.

Both surrogate models approximate the original system with acceptable accuracy. The model computed with τ AAA approximates the cooling phase of the system (starting after 600 s) slightly better than SVD-LF.

6 | CONCLUSIONS

In this paper, we proposed τ AAA, a modification of the AAA algorithm that incorporates ideas from tangential interpolation. This extends the capabilities of AAA-like methods to the interpolation of matrix functions of relatively large size. In the context of this work, we used this property to compute surrogate models of multi-input, multi-output (MIMO) dynamical systems with many inputs and outputs. We demonstrated the versatility and effectiveness of the new method by comparing it to the classic Loewner framework and to a block variant of AAA. τ AAA shows similar and in some cases even superior accuracy and can be applied in cases where a block-based strategy is not applicable.

ACKNOWLEDGMENTS

Funded by the German Research Foundation—Project-ID 174223256—TRR 96.

Open access funding enabled and organized by Projekt DEAL.

ORCID

Ion Victor Gosea  <https://orcid.org/0000-0003-3580-4116>

Jens Saak  <https://orcid.org/0000-0001-5567-9637>

Julia Vettermann  <https://orcid.org/0000-0003-4094-7840>

REFERENCES

1. Mayo, A. J., & Antoulas, A. C. (2007). A framework for the solution of the generalized realization problem. *Linear Algebra and its Applications*, 425(2–3), 634–662.
2. Astolfi, A. (2010). Model reduction by moment matching for linear and nonlinear systems. *IEEE Transactions on Automatic Control*, 55(10), 2321–2336.
3. Gustavsen, B., & Semlyen, A. (1999). Rational approximation of frequency domain responses by vector fitting. *IEEE TransPD*, 14(3), 1052–1061.
4. Goyal, P., Duff, I. P., & Benner, P. (2023). Inference of continuous linear systems from data with guaranteed stability. *Subject classifications: Machine Learning (cs.LG) and Dynamical Systems (math.DS)*, available online at: <http://arxiv.org/abs/2301.10060>.
5. Nakatsukasa, Y., Sète, O., & Trefethen, L. N. (2018). The AAA algorithm for rational approximation. *SIAM Journal on Scientific Computing*, 40(3), A1494–A1522.
6. Gosea, I. V., & Güttel, S. (2021). Algorithms for the rational approximation of matrix-valued functions. *SIAM Journal on Scientific Computing*, 43(5), A3033–A3054.
7. Antoulas, A. C., Beattie, C. A., & Gugercin, S. (2020). *Interpolatory methods for model reduction*. Computational Science & Engineering, Society for Industrial and Applied Mathematics, Philadelphia, PA.
8. Antoulas, A. C., Lefteriu, S., & Ionita, A. C. (2017). A tutorial introduction to the Loewner framework for model reduction. In P. Benner, M. Ohlberger, A. Cohen, & K. Willcox (Eds.), *Model reduction and approximation* (pp. 335–376). SIAM.
9. Golub, G. H., & Van Loan, C. F. (2013). *Matrix computations* (4th ed.). Johns Hopkins Studies in the Mathematical Sciences, Johns Hopkins University Press, Baltimore, MD.
10. Gugercin, S., Antoulas, A. C., & Bedrossian, M. (2001). Approximation of the international space station 1r and 12a flex models. *Proceedings of the IEEE Conference on Decision and Control* (pp. 1515–1516).
11. Hernández-Becerro, P., Spescha, D., & Wegener, K. (2020). Model order reduction of thermo-mechanical models with parametric convective boundary conditions: Focus on machine tools. *Computational Mechanics*, 67(1), 167–184.
12. Aumann, Q., Benner, P., Saak, J., & Vettermann, J. (2023). *Model order reduction strategies for the computation of compact machine tool models*. Lecture notes in production engineering (pp. 132–145). Springer International Publishing.
13. Benner, P., & Werner, S. W. R. (2021). MORLAB—the Model Order Reduction LABORatory. *Model reduction of complex dynamical systems* (pp. 393–415). Springer International Publishing.

How to cite this article: Aumann, Q., Benner, P., Gosea, I. V., Saak, J., & Vettermann, J. (2023). A tangential interpolation framework for the AAA algorithm. *Proceedings in Applied Mathematics and Mechanics*, 23, e202300183. <https://doi.org/10.1002/pamm.202300183>