CAMBRIDGE
UNIVERSITY PRESS

RESEARCH ARTICLE

Learning stable reduced-order models for hybrid twins

Abel Sancarlos[1,2,3], Morgan Cameron[2], Jean-Marc Le Peuvedic[4], Juliette Groulier[2], Jean-Louis Duval[2], Elias Cueto[4] 🔘 and Francisco Chinesta[1,2]*

[1]PIMM Laboratory and ESI Group Chair, ENSAM Institute of Technology, 75013 Paris, France
[2]ESI Group, 3bis rue Saarinen, 94528 Rungis, France
[3]Aragon Institute of Engineering Research, Universidad de Zaragoza, 50018 Zaragoza, Spain
[4]Dassault Aviation, 78 Quai Marcel Dassault, 92210 Saint-Cloud, France
*Corresponding author. E-mail: francisco.chinesta@ensam.eu

Abstract

The concept of "hybrid twin" (HT) has recently received a growing interest thanks to the availability of powerful machine learning techniques. This twin concept combines physics-based models within a model order reduction framework—to obtain real-time feedback rates—and data science. Thus, the main idea of the HT is to develop on-the-fly data-driven models to correct possible deviations between measurements and physics-based model predictions. This paper is focused on the computation of stable, fast, and accurate corrections in the HT framework. Furthermore, regarding the delicate and important problem of stability, a new approach is proposed, introducing several subvariants and guaranteeing a low computational cost as well as the achievement of a stable time-integration.

Impact Statement

Hybrid twins are a special type of digital twins able to learn and correct themselves once significant and persistent biases between data and predictions is found. Here, we develop a method that constructs, under real-time constraints, data-driven corrections to unsatisfactory models within the twin. Of particular importance is the issue of stability of the resulting, corrected model, which is here guaranteed.

1. Introduction

The hybrid twin (HT) paradigm is a powerful tool to make better predictions, increase control performance or improve decision-making (Chinesta et al., 2020; Martín et al., 2020). The main idea, see Figure 1, is to develop on-the-fly data-driven models to correct the gap between data (i.e., measurements) and model predictions. In other words, there are two main ingredients of a HT:

- The first one is to enrich physics description with data.
- The second one is to accelerate physics-based models using model order reduction (MOR) techniques, as in Chinesta et al. (2011, 2015), or Sancarlos et al. (2021).
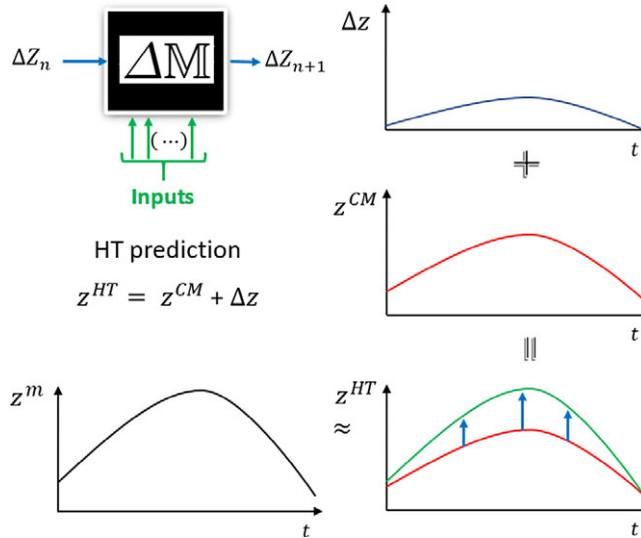
Figure 1. *Diagram illustrating the hybrid twin (HT) concept. The HT is able to correct the discrepancy between the coarse model (CM) and the pseudo-experimental data (PED, denoted by a superscript m). Its prediction z is here denoted by a superscript* HT *whereas the enrichment model is denoted by* Δz. *The superscript* CM *refers to the "coarse model," and* $\Delta \mathbb{M}$ *is the model correction. Both concepts are introduced in Section 2.*

In any case, when addressing dynamical systems in the HT framework, it is important to guarantee the stability of the system when adding corrections to the physical model. It is worth noting that this is an important issue, because sometimes the best model, computed with state-of-the-art algorithms, completely fails to obtain a stable time-integrator. For example, when considering a linear dynamical model by the dynamic mode decomposition (DMD) approach (Schmid, 2010; Kutz et al., 2016), the feasible region constrained by the stability condition is nonconvex (Huang et al., 2016), and no general methodology exists to solve it.

For this reason, this work proposes a new, fast, and efficient methodology, covering several subvariants and guaranteeing a low computational cost as well as the achievement of a stable dynamical system. This technique will therefore be used to add a stable correction term into the HT concept.

The methodology is tested for an industrial case detailed in Section 4.1, where an excellent agreement was observed when employing the HT approach.

The work is organized as follows: in Section 2, the system modeling with the HT concept is presented and compared to the direct (so to speak, from scratch) data-driven approach. An alternative approach that benefits from transfer learning, for instance, can be found in Guastoni et al. (2020). Then, in Section 3, the proposed technique to obtain stable systems is described as well as the subvariants to deal with high-dimensional systems. Finally, in Sections 4 and 5, the results and general conclusions of the present work are discussed, respectively.

2. System Modeling

In what follows, we consider the system as described by a vector $\mathbf{z} \in \mathbb{R}^D$ (with D the number of variables involved in the system evolution). The state (snapshot of the system) at time $t_n = n\Delta t$ is stored at vectors $\mathbf{Z}_n$, with $n \geq 0$, with $\mathbf{Z}_0$ assumed known. In addition, d control parameters are considered, giving rise to a parametric space $\boldsymbol{\mu} \in \mathbb{R}^d$.

We assume the existence of a model $\mathbb{M}^c(\mathbf{z}, \boldsymbol{\mu})$ which we refer to as coarse, since we assume that some form of enrichment is necessary. Often, this model is physics-based, it arises from the corresponding

PDEs governing the problem and can contain nonlinearities. Other times, it is based on Model Based System Engineering modeling. This approximate representation of the reality is sought to be computable under real-time constraints. These constraints depend on the context and can range from some seconds to the order of milliseconds. If the complexity of the model does not allow to obtain such a response under constraints, MOR techniques constitute an appealing alternative. These that can be linear or not, allow us to timely integrate the state of the system.

Because of the simplifying hypotheses involved in the construction of the model $\mathbb{M}^c$, it is expected that model predictions $\mathbf{Z}_n^c$ differ from measurements to some extent, that is, $\|\mathbf{Z}_n - \mathbf{Z}_n^c\| > \epsilon$, for most time steps n, thus needing for a correction.

2.1. Extracting the model of the system from scratch

Several routes exist to construct a model for a given dynamical system. The first one consists in performing a completely data-driven approach from experimental measurements $(\mathbf{Z}_n, \boldsymbol{\mu}_n)$, $n = 0, 1, \ldots$. A valuable option is to consider the so-called DMD to extract a matrix model of a discrete linear system (Schmid, 2010). As suggested by several works, many different systems can be well approximated using this approach (Schmid, 2011; Schmid et al., 2011). If problems arise due to complex system behaviors, the procedure of extending the state vector to a higher dimensional space can often solve the problem (Kutz et al., 2016; Eivazi et al., 2021).

A second alternative could be a variant of the technique presented in Sancarlos et al. (2020), that consists in grouping all the states close to $\mathbf{Z}_n$ and the control parameters close to $\boldsymbol{\mu}_n$ into a set $\mathcal{S}_n$. For the sake of clarity, in what follows $\mathcal{S}$ will refer to one of these generic sets. A linear model for the set $\mathcal{S}$, denoted $\mathbf{M}_{\mathcal{S}}$, which in this case is simply a matrix, is extracted from:

$$\mathbf{M}_{\mathcal{S}} = \arg\min_{\mathbf{N}_{\mathcal{S}}} \left\|\mathbf{X}_1 - \mathbf{N}_{\mathcal{S}}\widetilde{\mathbf{X}}_0\right\|_F^2, \tag{1}$$

with

$$\widetilde{\mathbf{Z}}_n = \begin{bmatrix} \mathbf{Z}_n \\ \boldsymbol{\mu}_n \end{bmatrix},$$

$$\widetilde{\mathbf{X}}_0 = \left[\widetilde{\mathbf{Z}}_0, \ \widetilde{\mathbf{Z}}_1, \ \ldots, \ \widetilde{\mathbf{Z}}_{n_{s-1}}\right],$$

$$\mathbf{X}_1 = [\mathbf{Z}_1, \ \mathbf{Z}_2, \ \ldots, \ \mathbf{Z}_{n_s}],$$

if stability problems do not arise. Note that in this case the model is composed of local matrices defining different linear maps in each set. A reduced version of this approach was extensively discussed in Reille et al. (2019).

Quite often, an issue can appear because of the difficulties to learn stable models when constructing these dynamical systems. For this reason, in Section 3, we propose a new methodology when using these techniques to guarantee that the obtained systems remain stable.

2.2. Enriching a physics-based model within the HT framework

Constructing a model $\mathbb{M}$ of the physical system from scratch is not the most valuable route as discussed in our former works (Chinesta et al., 2020). Purely data-driven, black-box models are not popular in industry, due to the lack of interpretability and guaranteed error estimators. Thus, a more valuable option consists of constructing corrections to physics-based models—if these provide unsatisfactory results—from an additive correction of this coarse prediction.

In fact, since the coarse model (CM) is expected to perform reasonably well for predicting the state of the system, bias will in general remain reasonably small. If this is true, the correction model will be much less nonlinear, and it will accept a more accurate description from the same amount of data.

Thus, we define the correction contribution $\mathbf{C}_n$ (or, equivalently, the model enrichment) as:

$$\mathbf{Z}_n - \mathbf{Z}_n^c = \mathbf{C}_n, \tag{2}$$

where $\mathbf{Z}_n^c$ refers the model prediction.

Taking as a proof of concept a dynamic linear system with control inputs, the correction term is searched as:

$$\mathbf{C}_{n+1} = \mathbf{W}\mathbf{C}_n + \mathbf{V}\boldsymbol{\mu}_n, \tag{3}$$

where $\mathbf{W}$ and $\mathbf{V}$ are the matrices defining the time evolution of the correction term. Thus, the total response of the system is finally predicted by using:

$$\mathbf{Z}_{n+1} \approx \mathbf{Z}_{n+1}^c + \mathbf{W}\mathbf{C}_n + \mathbf{V}\boldsymbol{\mu}_n. \tag{4}$$

By taking into consideration that $\mathbf{Z}^c$ is stable and integrated independently of the correction term, the stability condition of the system should apply just on the correction term.

We propose an approach to guarantee the construction of stable systems with low computational cost in the next section. In addition, this technique can be used either to build models from scratch or the correction term of the HT approach.

3. Efficiently Learning Stable Linear Dynamical Systems and DMDc Models

Frequently, difficulties to learn stable models arise when learning linear dynamical systems, specially when dealing with high-dimensional data. This is an important issue to deal with because of the growing importance of the data-driven approximations. For instance, it is usual to search the best linear approach of a set of high-dimensional data to make, for example, fast predictions of the system or to develop control strategies. In fact, there is a growing success of techniques such as the DMD to discover dynamical systems from high-dimensional data (Schmid, 2010; Kutz et al., 2016). This success steams from its capability of providing an accurate decomposition of a complex system into spatiotemporal coherent structures while constructing the model dynamics evolving on a low-rank subspace.

However, when operating in the above scenarios, sometimes the best model computed with state-of-the-art algorithms fails to obtain a stable time-integration. In fact, for a given a set of data, the problem to guarantee a stable system is defined by Equation (7) below but, unfortunately, the feasible region constrained by the spectral radius is nonconvex and no general methodology exists to solve it (Huang et al., 2016). Moreover, if a fast procedure is needed to obtain a correction model in a real-time application, this problem is further exacerbated.

For these reasons, this paper proposes a methodology to compute a stable model for a given dataset at low computational cost. The strategy is discussed for a discrete linear system, for a DMD model and for a DMD with control (DMDc) model. In addition, other strategies are discussed.

Let us assume the following dynamical systems defined in Equations (5) and (6), the first one without considering inputs,

$$\mathbf{Z}_{n+1} = \mathbf{M}\mathbf{Z}_n, \tag{5}$$

and the second one considering inputs,

$$\mathbf{Z}_{n+1} = \begin{bmatrix} \mathbf{M} & \mathbf{N} \end{bmatrix} \begin{bmatrix} \mathbf{Z}_n \\ \boldsymbol{\mu}_n \end{bmatrix} = \widetilde{\mathbf{M}}\widetilde{\mathbf{Z}}_n, \tag{6}$$

where $\mathbf{M}$ and $\mathbf{N}$ are the matrices defining the time evolution of the system. To guarantee stability in the above systems, the following condition must be satisfied:

$$\rho(\mathbf{M}) \leq 1, \tag{7}$$

where $\rho(\cdot)$ denotes the spectral radius.

Therefore, in relation to Equation (6), the solution minimizes:

$$\left\|\mathbf{X}_1 - \widetilde{\mathbf{M}}\widetilde{\mathbf{X}}_0\right\|_F^2 \quad \text{s.t.} \quad \rho(\mathbf{M}) \leq 1, \tag{8}$$

with

$$\widetilde{\mathbf{X}}_0 = \left[\widetilde{\mathbf{Z}}_0, \ \widetilde{\mathbf{Z}}_1, \ \dots, \ \widetilde{\mathbf{Z}}_{n_{s-1}}\right],$$
$$\mathbf{X}_1 = \left[\mathbf{Z}_1, \ \mathbf{Z}_2, \ \dots, \ \mathbf{Z}_{n_s}\right],$$

where n_s is the number of different snapshots for the training and the matrices $\widetilde{\mathbf{X}}_0$ and $\mathbf{X}_1$ contain the data to construct the model. In them, each column corresponds to a snapshot of the system at a given time instant.

Unfortunately, as already said, the feasible region constrained by $\rho(\mathbf{M}) \leq 1$ is nonconvex and no general methodology exists to solve it (Huang et al., 2016). This can lead to the problems already discussed where an unstable model is obtained or, in other cases, simply an extremely bad model is extracted due to a failure of the optimization methodology employed.

The proposed approach, which can always guarantee the creation of a stable model, is based on an observation of the following inequality, which is satisfied by any matrix norm:

$$\rho(\mathbf{A}) \leq \|\mathbf{A}\|. \tag{9}$$

To prove it, let ω be an eigenvalue of $\mathbf{A}$, and let $\mathbf{x} \neq 0$ be a corresponding eigenvector. From $\mathbf{A}\mathbf{x} = \omega\mathbf{x}$, we have:

$$\mathbf{A}\mathbf{X} = \omega\mathbf{X},$$

where $\mathbf{X} = [\mathbf{x}, \cdots, \mathbf{x}]$.

It follows that:

$$\|\omega\mathbf{X}\| = |\omega|\|\mathbf{X}\|,$$

and taking into account that,

$$\|\mathbf{A}\mathbf{X}\| \leq \|\mathbf{A}\| \ \|\mathbf{X}\|,$$

it follows that,

$$|\omega| \ \|\mathbf{X}\| \leq \|\mathbf{A}\| \ \|\mathbf{X}\|.$$

Simplifying the above expression by $\|\mathbf{X}\|$ (> 0) gives:

$$|\omega| \ \leq \|\mathbf{A}\|,$$

that taking the maximum over all eigenvalues ω gives the desired proof.

Taking for the reasoning the following induced norm,

$$\|\mathbf{A}\|_1 = \max_{1 \leq j \leq m_1} \sum_{i=1}^{m_2} |a_{ij}|,$$

where m_1 is the number of columns, and m_2 is the number of rows—which is simply the maximum absolute column sum of the matrix. It can be observed that by decreasing the absolute value of the matrix coefficients, a smaller matrix norm is obtained, and if it is decreased sufficiently, a smaller $\rho(\mathbf{A})$ is got, because of Equation (9).

Therefore, the idea to obtain a stable system is to shrink the matrix coefficients of $\mathbf{M}$. In fact, we propose to do so by using the ridge regression (Hastie et al., 2009), also known as a special case of the Tikhonov regularization. Many advantages can be obtained from this choice. For instance, a closed mathematical expression is obtained. This implies that there is no need to use complex optimization procedures (that can fail to converge). In addition, there is a low added computational cost when changing the ordinary least squares problem to the ridge one.

This way, we proved a new feature and use for the ridge regression. Ridge regression was employed in certain way for regression of dynamical systems, see, for example, Erichson et al. (2019), but just to use the classical function of ridge: to deal with ill-posed problems. Now, we have extended the employment of the technique to a broader problem area: the construction of stable dynamical systems.

To reformulate the resolution of the systems (5) and (6), and then extend the procedure for the DMDc (a more general version of the DMD considering control inputs), two options are envisaged. The first one is solving the following problem:

$$
\begin{aligned}
\widehat{\mathbf{M}} &= \arg\min_{\mathbf{M}} \left\{ \|\mathbf{X}_1 - \mathbf{M}\mathbf{X}_0\|_F^2 + \lambda^2 \|\mathbf{M}\|_F^2 \right\} \\
&= \arg\min_{\mathbf{M}} \|[\mathbf{X}_1 \ \ \mathbf{0}] - \mathbf{M}[\mathbf{X}_0 \ \ \lambda\mathbf{I}]\|_F^2 = \arg\min_{\mathbf{M}} \|\overline{\mathbf{X}}_1 - \mathbf{M}\overline{\mathbf{X}}_0\|_F^2,
\end{aligned}
\tag{10}
$$

where:

$$
\mathbf{X}_0 = [\mathbf{Z}_0, \ \mathbf{Z}_1, \ \ldots, \ \mathbf{Z}_{n_s-1}],
$$

$\overline{\mathbf{X}}_1$ and $\overline{\mathbf{X}}_0$ are the augmented matrices, $\mathbf{I}$ is an identity matrix of size $D \times D$, and $\mathbf{0}$ is a zero matrix of size $D \times D$.

The solution of the above problem can be computed using the Moore–Penrose pseudoinverse, therefore:

$$
\widehat{\mathbf{M}} = \overline{\mathbf{X}}_1 \left(\overline{\mathbf{X}}_0\right)^{\dagger},
$$

where $\dagger$ is the Moore–Penrose pseudoinverse.

A second procedure to solve Equation (10) is to employ a ridge regression for each variable to be predicted. Concerning the system (6),

$$
\begin{aligned}
\widehat{\mathbf{M}} &= \arg\min_{\widetilde{\mathbf{M}}} \left\{ \|\mathbf{X}_1 - \widetilde{\mathbf{M}}\widetilde{\mathbf{X}}_0\|_F^2 + \lambda^2 \|\mathbf{M}\|_F^2 \right\} \\
&= \arg\min_{\widetilde{\mathbf{M}}} \left\| [\mathbf{X}_1 \ \ \mathbf{0}_1] - \widetilde{\mathbf{M}} \begin{bmatrix} \mathbf{X}_0 & \lambda\mathbf{I} \\ \mathbf{U}_0 & \mathbf{0}_2 \end{bmatrix} \right\|_F^2 = \arg\min_{\widetilde{\mathbf{M}}} \|\overline{\mathbf{X}}_1 - \widetilde{\mathbf{M}}\overline{\mathbf{Y}}_0\|_F^2,
\end{aligned}
\tag{11}
$$

where:

$$
\mathbf{U}_0 = [\boldsymbol{\mu}_1, \ \boldsymbol{\mu}_2, \ \ldots, \ \boldsymbol{\mu}_{n_s-1}],
$$

$\overline{\mathbf{X}}_1$ and $\overline{\mathbf{Y}}_0$ are the augmented matrices, $\mathbf{I}$ is an identity matrix of size $D \times D$, $\mathbf{0}_1$ is a zero matrix of size $D \times D$, and $\mathbf{0}_2$ is a zero matrix of size $d \times D$.

In addition, the same procedures used to solve system (10) can be used to solve (11), either the Moore–Penrose pseudoinverse or the individual ridge regressions.

The above model can be extended in the context of high-dimensional systems when using the DMDc taking the formulation expressed in Equation (11). To do that, we take the singular value decomposition (SVD) of matrix $\overline{\mathbf{Y}}_0 = \widetilde{\boldsymbol{\Xi}}\widetilde{\boldsymbol{\Sigma}}\widetilde{\mathbf{V}}^*$ (where the star symbol * indicates the conjugate transpose). Therefore:

$$
\widetilde{\mathbf{M}} = \overline{\mathbf{X}}_1 \left(\overline{\mathbf{Y}}_0\right)^{\dagger} = \overline{\mathbf{X}}_1 \widetilde{\mathbf{V}}\widetilde{\boldsymbol{\Sigma}}^{-1}\widetilde{\boldsymbol{\Xi}}^*.
$$

Approximations of the operators $\mathbf{M}$ and $\mathbf{N}$ can be found as follows:

$$
\widetilde{\mathbf{M}} = [\mathbf{M}, \ \ \mathbf{N}] = \left[\overline{\mathbf{X}}_1 \widetilde{\mathbf{V}}\widetilde{\boldsymbol{\Sigma}}^{-1}\widetilde{\boldsymbol{\Xi}}_1^*, \ \ \overline{\mathbf{X}}_1 \widetilde{\mathbf{V}}\widetilde{\boldsymbol{\Sigma}}^{-1}\widetilde{\boldsymbol{\Xi}}_2^* \right],
$$

where $\widetilde{r}$ is the truncation value of the SVD applied to decompose matrix $\overline{\mathbf{Y}}_0 = \widetilde{\boldsymbol{\Xi}}\widetilde{\boldsymbol{\Sigma}}\widetilde{\mathbf{V}}^*$, $\widetilde{\boldsymbol{\Xi}}^* = \left[\widetilde{\boldsymbol{\Xi}}_1^*, \ \ \widetilde{\boldsymbol{\Xi}}_2^* \right]$, and the sizes of $\widetilde{\boldsymbol{\Xi}}_1^*$ and $\widetilde{\boldsymbol{\Xi}}_2^*$ are $\widetilde{r} \times D$ and $\widetilde{r} \times d$, respectively.

For high-dimensional systems ($D \gg 1$) a reduced-order approximation can be solved for instead, leading to a more tractable computational model. Thus, we look for a transformation to a lower-dimensional subspace on which the dynamics evolve.

The output space $\overline{\mathbf{X}}_1$ is chosen to find the reduced-order subspace. Consequently, the SVD of $\overline{\mathbf{X}}_1$ is defined as:

$$\overline{\mathbf{X}}_1 = \mathbf{\Xi \Sigma V}^*,$$

where the truncation value for this SVD will be denoted as r. Please note that usually both SVDs will have different truncation values.

Then, by employing the change of coordinates $\mathbf{Z} = \mathbf{\Xi}\widehat{\mathbf{Z}}$ (or equivalently $\widehat{\mathbf{Z}} = \mathbf{\Xi}^*\mathbf{Z}$), the following reduced-order approximations can be obtained,

$$\mathbf{Z}_{n+1} = \mathbf{M}\mathbf{Z}_n + \mathbf{N}\boldsymbol{\mu}_n,$$
$$\mathbf{\Xi}\widehat{\mathbf{Z}}_{n+1} = \mathbf{M}\mathbf{\Xi}\widehat{\mathbf{Z}}_n + \mathbf{N}\boldsymbol{\mu}_n,$$

so that

$$\widehat{\mathbf{Z}}_{n+1} = \widehat{\mathbf{M}}\widehat{\mathbf{Z}}_n + \widehat{\mathbf{N}}\boldsymbol{\mu}_n, \tag{12}$$

where:

$$\widehat{\mathbf{M}} = \mathbf{\Xi}^*\overline{\mathbf{X}}_1\widetilde{\mathbf{V}}\widetilde{\mathbf{\Sigma}}^{-1}\widetilde{\mathbf{\Xi}}_1^*\mathbf{\Xi},$$
$$\widehat{\mathbf{N}} = \mathbf{\Xi}^*\overline{\mathbf{X}}_1\widetilde{\mathbf{V}}\widetilde{\mathbf{\Sigma}}^{-1}\widetilde{\mathbf{\Xi}}_2^*,$$

and the sizes of $\widehat{\mathbf{M}}$ and $\widehat{\mathbf{N}}$ are $r \times r$ and $r \times d$, respectively.

To select the penalty factor, when the standard procedure leads to a matrix that violates the stability condition, several options can be envisaged. Here, we propose to use the bisection method (which guarantees convergence toward the solution) or the regula falsi method (to speed up the process). Of course, faster algorithms can be used such as the Illinois algorithm, but our experience suggests that the former ones are enough in practice.

For example, if the bisection method is selected, the zero of the following function is sought:

$$f(\lambda) = \rho_{\text{desired}} - \rho(\mathbf{M}(\lambda)), \tag{13}$$

where ρ_{desired} is a chosen value very close to one, representing the target when the initial constructed model violates the stability condition.

Taking into consideration that the bracketing interval at step k of the algorithm is $\left[\lambda_k^a, \lambda_k^b\right]$, then the Equation (14) is employed to compute the new solution estimate for the penalty λ_k^c at step k:

$$\lambda_k^c = \frac{\lambda_k^a + \lambda_k^b}{2}. \tag{14}$$

If $f\left(\lambda_k^c\right)$ is satisfactory, the iteration stops. If this is not the case, the sign of $f\left(\lambda_k^c\right)$ is examined and the bracking interval is updated for the following iteration so that there is a zero crossing within the new interval.

4. Application to a Dynamical System

4.1. System to model and types of data

The modeled system corresponds to an air distribution system of an aircraft and is characterized by eight variables defining the state of the system: six temperatures T_n^i, $i = 1, 2, \ldots, 6$, and two pressures p_n^j, $j = 1, 2$. The model should also take into account three control variables μ_n^k, $k = 1, 2, 3$, for each time instant n.

With the knowledge and experience of Dassault Aviation, two models are constructed with the help of the software Dymola (Dassault Systemes, 2021):

- CM: This model deliberately fails to provide accurate predictions due to over-simplification. It is important to note that in industry, this type of model is often physics-based (although this is not mandatory) but still requires an important computing time.
- A high-fidelity model that will therefore be considered as the ground truth (GT), which is consequently still more time consuming. This model is going to emulate in this work the real state of the system.

Due to confidentiality issues, Dassault Aviation simulated different flights with both models and provided three different types of pseudo-experimental data (PED) which are employed in the present work:

- The CM data. These data correspond to the predictions of the CM for the given set of simulated flights.
- The GT data. These data correspond to the predictions of the GT model. It will be considered in the present work for evaluation purposes.
- PED. A white noise is added artificially to the GT data. Consequently, these data will emulate experimental measurements including experimental errors.

Additionally, in Figures 2 and 3, a comparison is shown between the three types of data (CM, GT, and PED) for a given flight simulation.

At this point, three different approaches were tested:

- Extracting a model from scratch from GT data (Section 4.2).
- Obtaining a model from scratch using the noisy PED (Section 4.3).
- Extracting a correction term to enrich the CM, thus constructing the HT (Section 4.4).
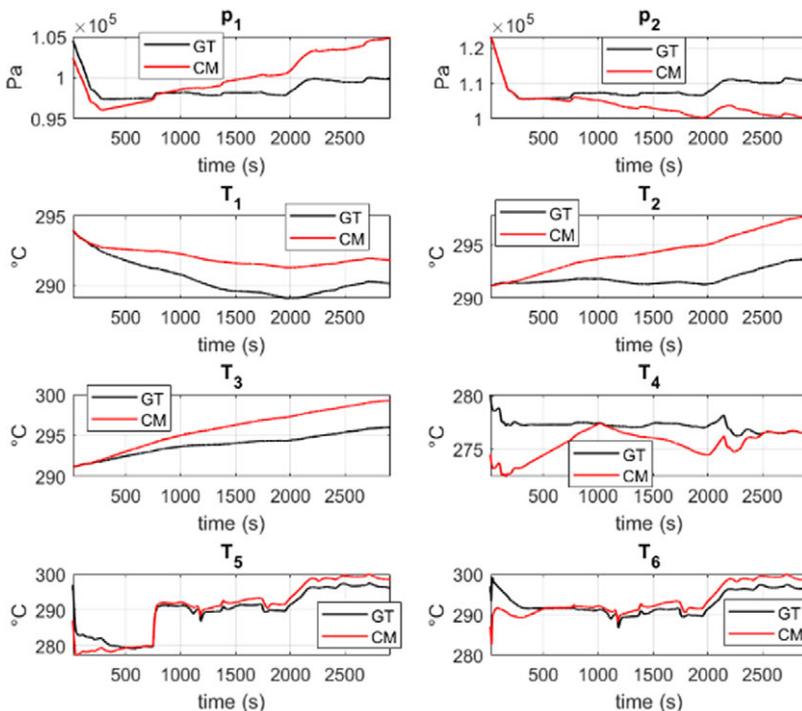
Figure 2. *Comparison of the state evolution* $\mathbf{Z}(t) = [p_1, p_2, T_1, T_2, T_3, T_4, T_5, T_6]$ *for a given flight between the CM and the ground truth (GT).*
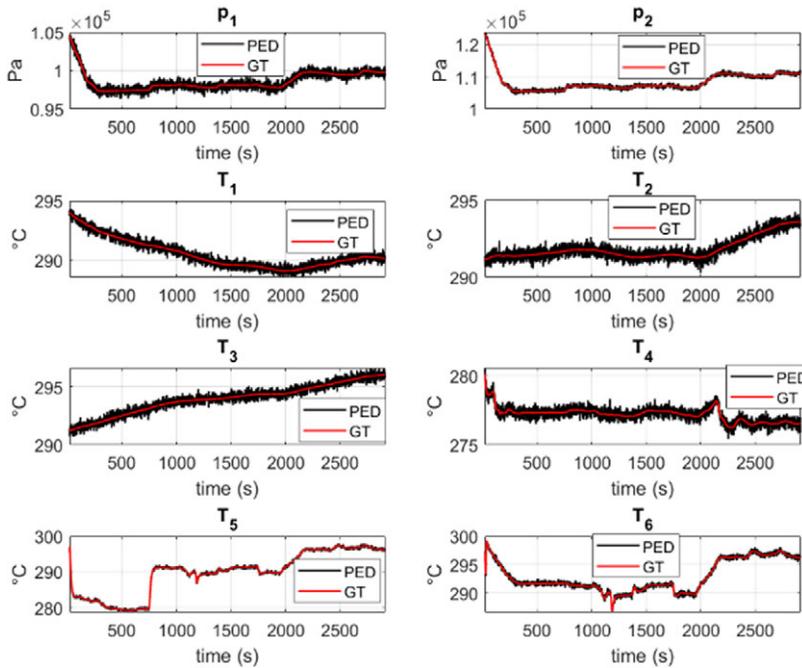
Figure 3. *Comparison of the system evolution, $\mathbf{Z}(t) = [p_1, p_2, T_1, T_2, T_3, T_4, T_5, T_6]$, for a given flight between the ground truth (GT) and the pseudo-experimental (noisy) data, PED.*

Advantages and weaknesses of each approach will be discussed. The mathematical details of the CM and GT models are omitted for confidentiality reasons. However, this is not important for presenting, discussing and employing the proposed methodology and, moreover, a successful outcome will be a sign that the proposed approach can address current industrial needs.

4.2. Extracting a model from scratch using the GT data

4.2.1. Procedure and results

It is interesting to analyse whether the proposed approach presented in Section 3 is able to learn a model from scratch employing the GT data. Therefore, this section is focused on this goal. In next sections, we will analyse if it is able to obtain similar results when learning in the presence of noise (PED data) and finally, we will see the advantages of using the HT rational instead of the complete data-driven approach.

We sketch the technique in the diagram of Figure 4. As it can be noticed, the system is characterized by eight variables defining the state of the system (six temperatures T_n^i, $i = 1, 2, \ldots, 6$, and two pressures p_n^j, $j = 1, 2$) and three control variables μ_n^k, $k = 1, 2, 3$ for each time instant n.

In the present case, we consider the simplest modeling approach in which the model consists in a simple linear application that maps the present state and control parameters $\mathbf{Z}_n$ and $\boldsymbol{\mu}_n$, respectively, onto the next system state $\mathbf{Z}_{n+1}$.

The available data consist in $F = 82$ flights, each one leading to eight time series $\mathbf{Z}_n^f$, with $f = 1, \ldots, F$, and $n = 1, \ldots, n^f$ (the number of collected data depends on the flight, these having different duration). It is important to note that, for any flight $t_{n+1} - t_n = \Delta t$, with constant Δt, for any component state and any flight.

We define the extended state as shown in Section 3:

$$\widetilde{\mathbf{Z}}_n = \begin{pmatrix} \mathbf{Z}_n \\ \boldsymbol{\mu}_n \end{pmatrix}. \tag{15}$$
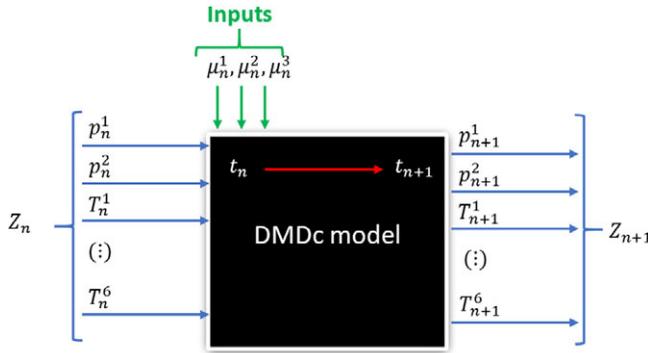
Figure 4. *Diagram illustrating the inputs and the state vector of the proposed DMDc model to reproduce the pruned data of the system.*

To learn the model, we select arbitrarily two flights from the $\mathtt{F}$ available, $\mathtt{f} = \mathtt{r}$ and $\mathtt{f} = \mathtt{s}$, and define the training matrices:

$$\widetilde{\mathbf{X}}_0 = \begin{pmatrix} \mathbf{Z}_0^{\mathtt{r}} & \cdots & \mathbf{Z}_{\mathtt{n}^{\mathtt{r}}}^{\mathtt{r}} & \mathbf{Z}_0^{\mathtt{s}} & \cdots & \mathbf{Z}_{\mathtt{n}^{\mathtt{s}}-1}^{\mathtt{s}} \\ \boldsymbol{\mu}_0^{\mathtt{r}} & \cdots & \boldsymbol{\mu}_{\mathtt{n}^{\mathtt{r}}}^{\mathtt{r}} & \boldsymbol{\mu}_0^{\mathtt{s}} & \cdots & \boldsymbol{\mu}_{\mathtt{n}^{\mathtt{s}}-1}^{\mathtt{s}} \end{pmatrix}, \tag{16}$$

and

$$\mathbf{X}_1 = [\mathbf{Z}_1^{\mathtt{r}} \quad \cdots \quad \mathbf{Z}_{\mathtt{n}^{\mathtt{r}}}^{\mathtt{r}} \quad \mathbf{Z}_1^{\mathtt{s}} \quad \cdots \quad \mathbf{Z}_{\mathtt{n}^{\mathtt{s}}}^{\mathtt{s}}], \tag{17}$$

that allows extracting the model by solving the problem indicated in Equation (11) or its reduced counterpart (12).

Then, as soon as the model is extracted, we proceed to predict the state evolution for each one of the $\mathtt{F}$ flights, from their initial states, by simply writing at each time t_n, $n = 1, \ldots, \mathtt{n}^{\mathtt{f}}$,

$$\widetilde{\mathbf{Z}}_n^{\mathtt{f}} = \begin{pmatrix} \mathbf{Z}_n^{\mathtt{f}} \\ \boldsymbol{\mu}_n^{\mathtt{f}} \end{pmatrix}, \tag{18}$$

and applying the updating of Equations (6) or (12). Training data are composed of two flights while the other eighty are used to test the performance of the present approach.

Figure 5 compares the predicted states at each time instant by integrating the just unveiled model from the initial condition. It employs a GT data series corresponding to one particular, previously unseen flight. It can be observed that the proposed approach achieves an excellent agreement for variables p_1, p_2, T_1, T_2, T_3, and T_4.

On the other hand, although the error in variables T_5 and T_6 is larger it achieves to follow the general trend, despite the fast time evolutions that these variables exhibit. The same tendency is observed in all the flights as in Figure 6) proves. To better capture the fast evolutions of these two variables, the procedure described in Section 2.1 for addressing nonlinear behaviors (Sancarlos et al., 2020) could be employed. However, in this work it is preferred to improve the accuracy in the prediction of these special variables by employing a Hybrid approach, for the reasons exposed in Section 4.4.

In Figure 7, it is observed that a similar accuracy is obtained for more than 85 % of the flights in the testing set (concerning variables p_1, p_2, T_1, and T_2). A similar conclusion follows from Figure 8 for variables T_3 and T_4. Therefore, it is concluded that the model has a good ability for generalization taking into consideration that just two flights are considered in the training.

However, the error in variables T_5 and T_6 can reach high values in a considerable part of the testing set. To address that, the HT rationale will be proposed and discussed.

Before that, we will see what happens if stability is not enforced when extracting the GT model from scratch in Section 4.2.2 as well as the improvements of the proposed procedure.
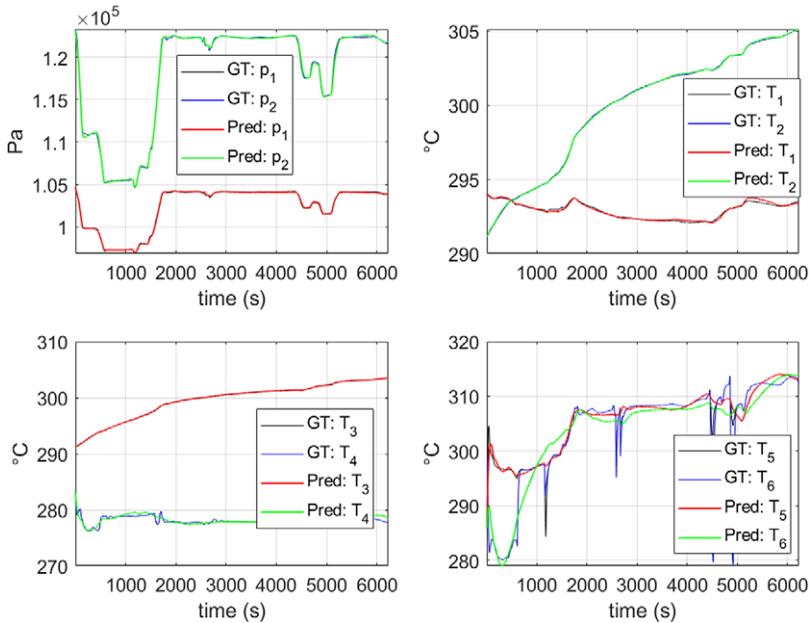
Figure 5. *Prediction of the GT data using the proposed technique for a flight which is not used in the training set. "GT" refers to GT data series described in Section 4.1 and "Pred" refers to the stabilized DMDc model obtained with the proposed approach discussed in Section 3.*
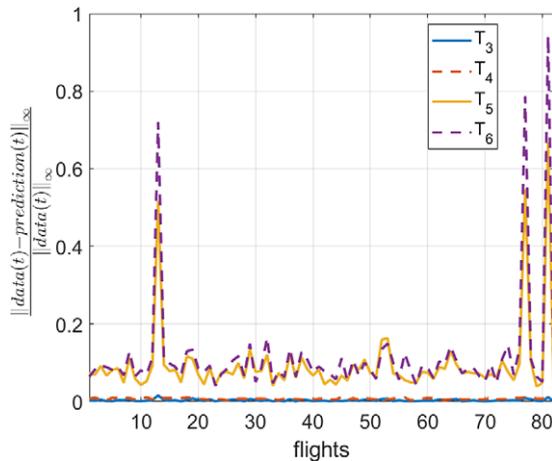
Figure 6. *Error in the prediction of T_3, T_4, T_5, and T_6 for different flights which are not in the training set. The prediction error in variables T_5 and T_6 is higher than the other ones due to their fast time evolution.*

4.2.2. Checking explicitly the improvements of the proposed approach to learn stable systems: Example when proceeding from the GT data

In the above section, the results when modeling the GT data were shown when using the stabilization procedure for the DMDc proposed in this work. However, it is interesting to analyze what happens when using other algorithms or by simply using standard procedures to observe the benefits of the proposed stabilization.

To this end, the GT data for a particular flight is considered. The time evolution of the system can be observed in Figure 9.
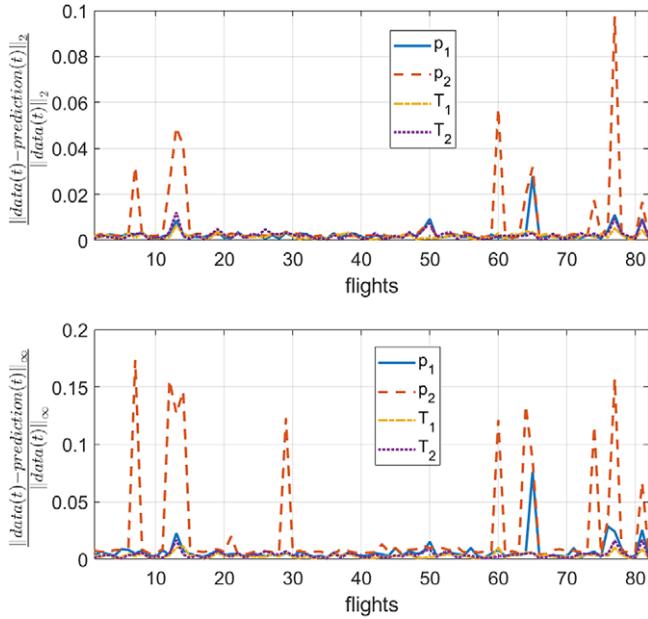
Figure 7. *Error in the prediction of p_1, p_2, T_1, and T_2 of the proposed technique for lights which are not in the training set.*
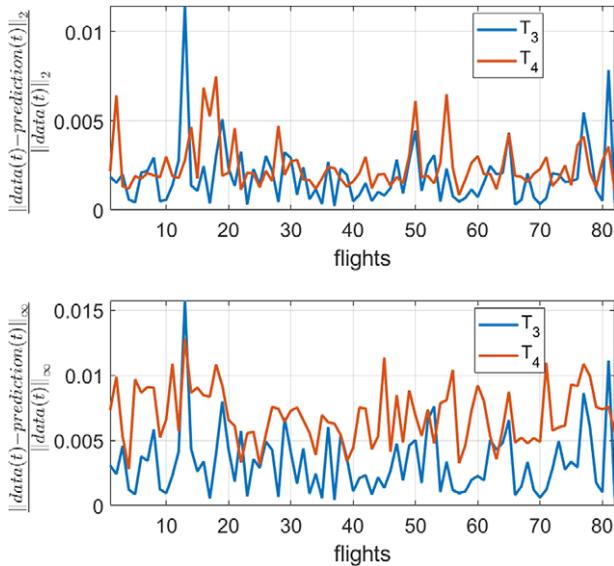
Figure 8. *Error in the prediction of T_3 and T_4 of the proposed technique for flights which are not in the training set.*

The first objective is to see if we can approximate the dynamics of Figure 9 by constructing a standard DMDc model without stabilizing it. Moreover, we are interested in observing if stability issues arise.

Therefore, a model is constructed following the DMDc procedure with the state vector and inputs exposed in Section 4.2.1.

As shown in Figure 10 below, the DMDc model reports stability issues giving useless predictions.

Nevertheless, by applying the proposed stabilization approach to the DMDc algorithm (as discussed in Section 3), a stable model is easily obtained quick and fast. The flight predicted by the proposed approach is shown in Figure 11. It is worth noting the great improvement observed by comparing the standard nonstabilized DMDc model (Figure 10) with the stabilized one (Figure 11) able to capture complex dynamics while completely overcoming the stability issues.

Now, in the next section, we will examine the possibility of unveiling an accurate model from noisy data. In this way, two important aspects will be analyzed: the interest of employing a HT approach and the ability of the proposed technique to filter noise.

4.3. *Extracting a model from scratch using noisy data (PED)*

In this section, we attempt to unveil a model from scratch using noisy data. This will allow us to study the robustness of the approach in the filtering process.

After applying the technique in different flights and studying the reconstruction error by considering different extended states $\widetilde{\mathbf{Z}}$, the proposed model is composed of:

$$\widetilde{\mathbf{Z}}_n = \begin{pmatrix} \mathbf{Z}_n \\ \boldsymbol{\mu}_n \\ \boldsymbol{\mu}_{n-1} \\ \boldsymbol{\omega}_n \\ \mathbf{W}_n \end{pmatrix}, \tag{19}$$

where:

$$\boldsymbol{\omega}_n = \sum_{i=0}^{n} \mu_i (t_{i+1} - t_i),$$

and
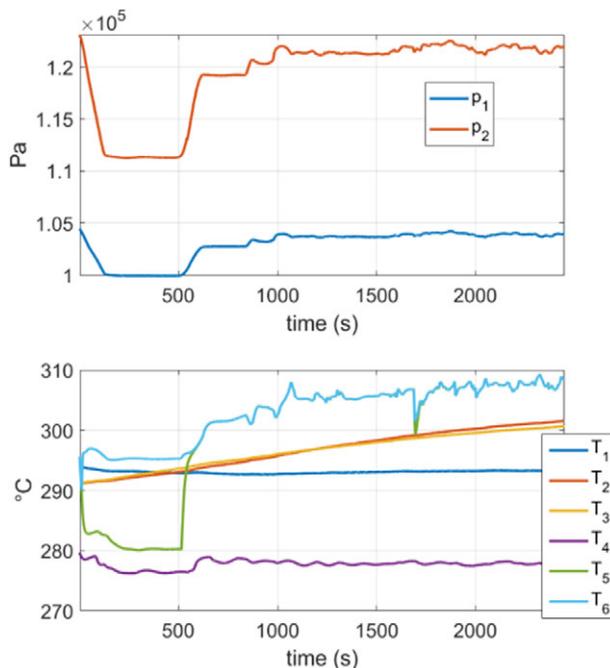
Figure 9. *GT data for the example in* Section 4.2.2.
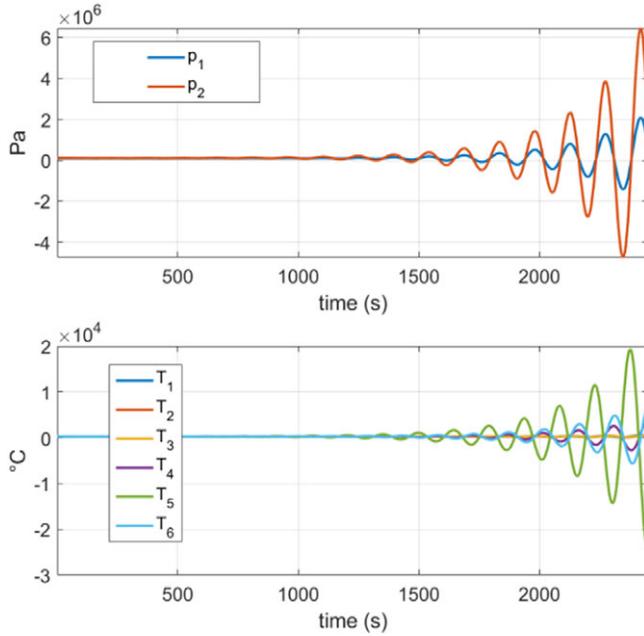
Figure 10. *Prediction* $\mathbf{Z}(t) = [p_1, p_2, T_1, T_2, T_3, T_4, T_5, T_6]$ *of the GT obtained through DMDc. This prediction tries to reproduce the flight of Figure 9 but fails to provide with stable results.*
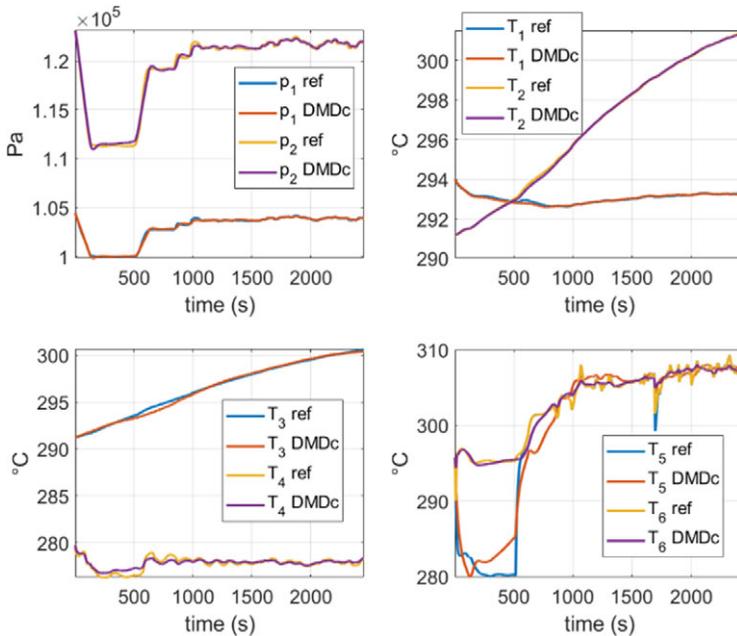
Figure 11. *Comparison between the reference dynamics of Figure 9 and the prediction of the modified, stable DMDc model. Huge improvements are observed when comparing with Figure 10. The state vector of the system is* $\mathbf{Z}(t) = [p_1, p_2, T_1, T_2, T_3, T_4, T_5, T_6]$.

$$\mathbf{W}_n = \sum_{i=0}^{n} \boldsymbol{\omega}_i (t_{i+1} - t_i).$$

The same methodology of the previous section is applied, by using the new extended states $\widetilde{\mathbf{Z}}$. This way, we can address the more complex behavior of the noisy data. In this case, nine flights of different duration are used for the training set.

In Figure 12, a comparison is shown between the dynamics predicted by the DMDc model obtained from scratch using the PED and the PED itself (see Section 4.1), for a flight contained in the training set. Here, it is observed that the model can capture the dynamics of the system with an excellent accuracy, by just employing the initial state of the system and the corresponding control inputs while filtering the noise contained in the training data.

In Figure 13, a comparison is shown between the dynamics predicted by the DMDc model obtained from scratch using the PED and the PED itself (see Section 4.1), for a flight contained in the testing set (never considered in the model construction). In these plots, it is shown that a good agreement is obtained for all the variables with the exception of T_5 and T_6. Similar results are reported in the other flights of the testing set.

Variables T_5 and T_6 are more challenging to predict because of their fast time evolution. Nevertheless, we are going to deal with them in the next section. Therefore, the HT concept, which follows, is expected leading to more accurate results.

4.4. Extracting the correction model: HT paradigm

As discussed in Section 2.2, constructing a model of the real system from scratch is not the most valuable route when addressing complex systems. For the system analyzed in the present work, outputs $p_1, p_2, T_1,$
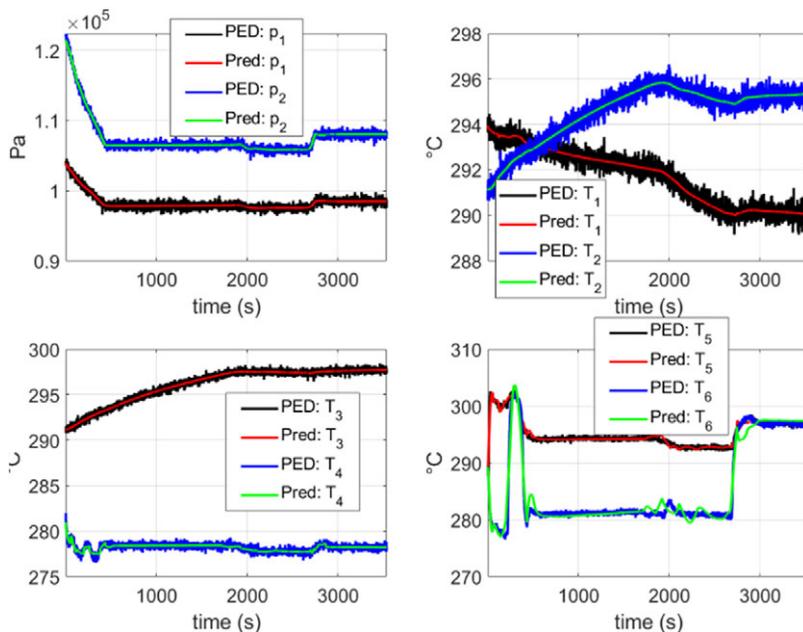
Figure 12. *Comparison between the model obtained from scratch using PED data and the PED data itself. In this figure, the reconstruction of a flight contained in the training set is shown. "PED" refers to the pseudo-experimental data with noise described in Section 4.1 and "Pred" refers to the stabilized DMDc model obtained with the proposed approach discussed in Section 3. It can be observed that an excellent agreement is obtained for every variable while filtering the noise.*

T_2, T_3, and T_4 are predicted to a great accuracy for the approaches shown in Sections 4.2 and 4.3. On the other hand, outputs T_5 and T_6 present difficulties. In these cases, an interesting option consists in expressing the state of the system from an additive correction of the CM. Therefore, in this case, the proposed model is going to capture just the ignorance that the CM contains.

One of the advantages of this concept is that the main response is provided by the physics-based model, thus guaranteeing that the model is going to exhibit a behavior coherent with the physical phenomenon under scrutiny as well as being explained by practitioners. In addition, the part of the response which has difficulties in being modeled—for instance, the appearance of degradation of the system—can be approximated by the data-driven model.

The HT concept is illustrated in Figure 1. Note that only the first measurement is mandatory to run what we coined as the $\Delta\mathbb{M}$. Therefore, knowing the initial state of the system, the real response can be reproduced adding the correction model to the CM without further measurements.

Again, the extended state $\Delta\widetilde{\mathbf{Z}}$ for the discrepancy model is:

$$\Delta\widetilde{\mathbf{Z}}_n = \begin{pmatrix} \Delta\mathbf{Z}_n \\ \boldsymbol{\mu}_n \\ \boldsymbol{\mu}_{n-1} \\ \boldsymbol{\omega}_n \\ \mathbf{W}_n \end{pmatrix}, . \tag{20}$$

Nine flights are considered in the training set. As expected for a real-life application, the measured data (i.e., the PED) is employed to obtain the discrepancy to be modeled within the HT concept.
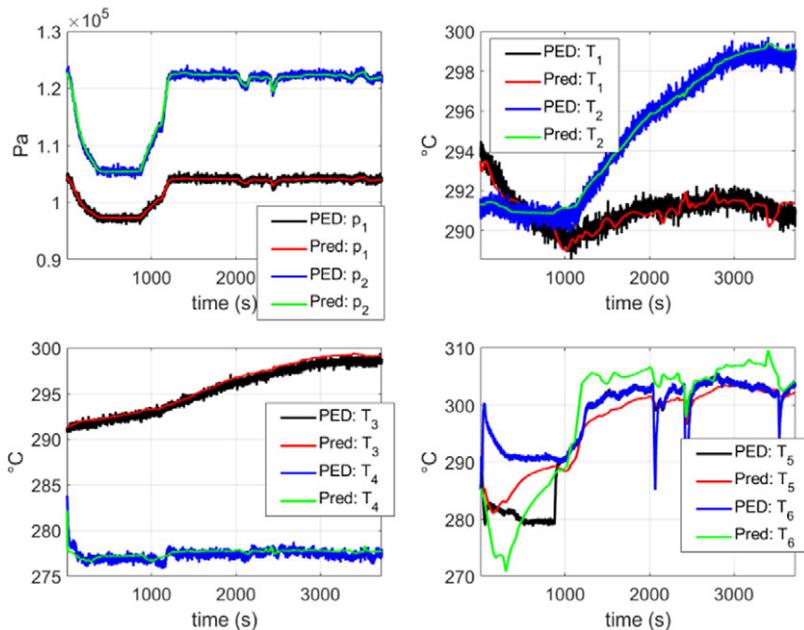
Figure 13. *Comparison between the model obtained from scratch using PED data and the PED data itself. In this figure, the reconstruction of a flight which is not contained in the training set is shown. "PED" refers to the pseudo-experimental data with noise described in Section 4.1 and "Pred" refers to the stabilized DMDc model obtained with the proposed approach discussed in Section 3. It can be observed that a good agreement is obtained for all the variables with the exception of T_5 and T_6.*

Figures 14–16 are obtained by just integrating from the initial state of the system and by employing and enriching the CM prediction without any further measurement. This proves the excellent agreement that can be achieved within the HT rationale.

In Figure 14, a comparison is shown between the HT prediction and the PED for a previously unseen flight. In these plots, an excellent agreement is noticed for all the variables. Moreover, predictions filter the noisy measurements.

An error criterion is defined to compare the prediction of the HT approach with the accuracy of the measuring instruments:

$$\mathrm{err}_i(t) = \frac{z_i^{\mathrm{GT}}(t) - z_i^{HT}(t)}{\Delta z_i^{\mathrm{max,GT}}}, \tag{21}$$

$$\mathrm{err}_{\mathrm{meas}}^{\mathrm{max}} = \frac{\max\left(z_i^{\mathrm{GT}} - z_i^m\right)}{\Delta z_i^{\mathrm{max,GT}}}, \tag{22}$$

where:

- i refers to the ith variable z_i of the state vector.
- z_i^{HT} is the predicted value of the HT for z_i.
- z_i^m is the measured value of z_i. These data include the corresponding noise. In other words, these data are the PED.
- z_i^{GT} is the true value of z_i which is theoretically unknown and cannot be accessed by an observer in a real application. We use this value for evaluation purposes.
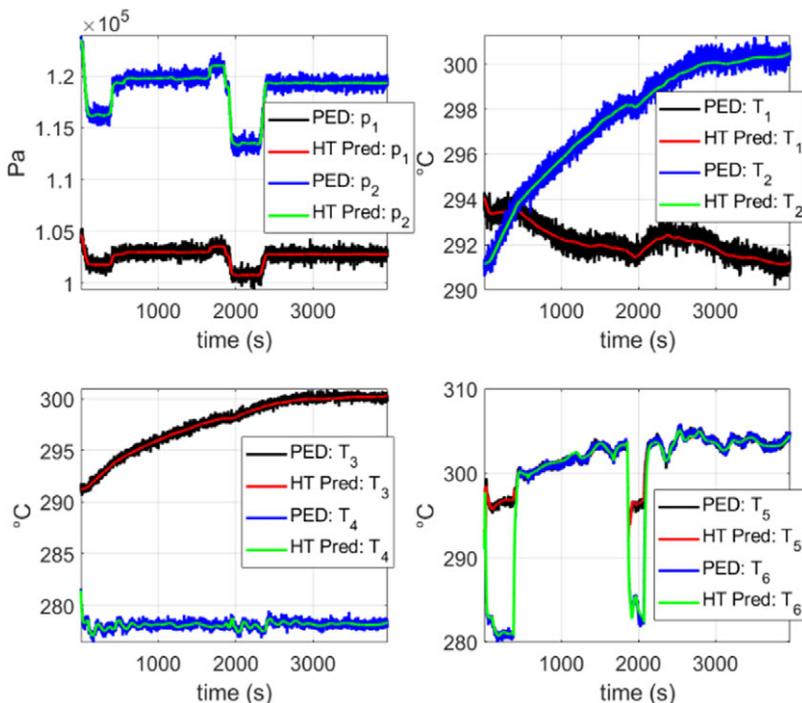
Figure 14. *Prediction of the hybrid twin (HT) approach considering a flight in the testing set. "PED" refers to the pseudo-experimental data with noise described in Section 4.1 and "HT Pred" refers to the HT approach whose correction term corresponds to a stabilized DMDc model obtained with the methodology discussed in Section 3. The correction term was constructed using the PED. It can be observed that an excellent agreement is obtained for all the variables.*
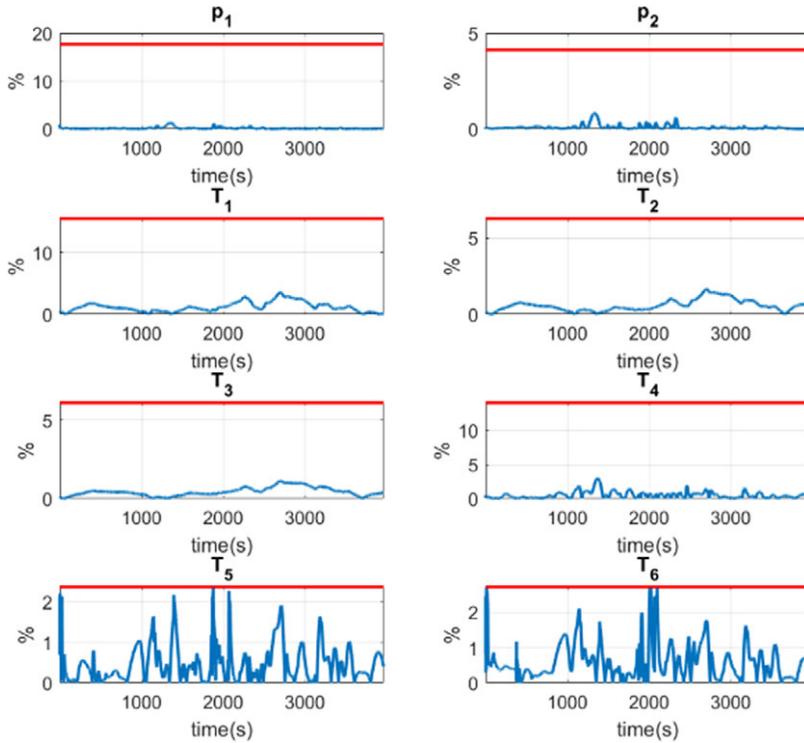
Figure 15. *Error of the hybrid twin (HT) approach (blue line) considering a flight which is not used for the training. The red line refers to the maximum error in the pseudo measurements (PED). The error criterion is defined in Equations (21) (blue line) and (22) (red line).*

- $\Delta z_i^{\max,\mathrm{GT}}$ is the difference between the maximum and the minimum value of z_i^{GT} considering all the flights.

Using Equations (21) and (22), it is possible to compare the accuracy of the HT with the one obtained by measuring the data (i.e., the PED). Moreover, the relative errors are computed taking as a base the maximum variation of each variable when regarding all the available flights. Figure 17 shows the maximum variation of a signal as well as the deviation caused by the noise to illustrate the concept.

Observing Figures 15 and 16, we can confirm that the HT concept allow us to improve, not only the accuracy of the CM but the accuracy of the measuring instruments regardless of whether or not flights come from training by filtering the noise. In this way, two goals are achieved at the same time: enriching the CM by learning the difference with the measured data while filtering the noise.

Therefore, the HT concept is a valuable route to enrich physics-based models with data-driven corrections. It is important to note that for HT to be applied, the CM must not be extremely bad, since in this case a direct data-driven or reduced-order modeling approach would be preferred (because there is no point in correcting such a model).

5. Conclusions

This work presents a fast and efficient methodology, covering several variants to learn dynamical models while guaranteeing a low computational cost as well as the achievement of stable dynamical time integrations. This technique was used with success to predict a practical scenario under the HT rationale, being able to impose stability in the correction term. In addition, the proposed technique filters noise improving the knowledge of the system state.
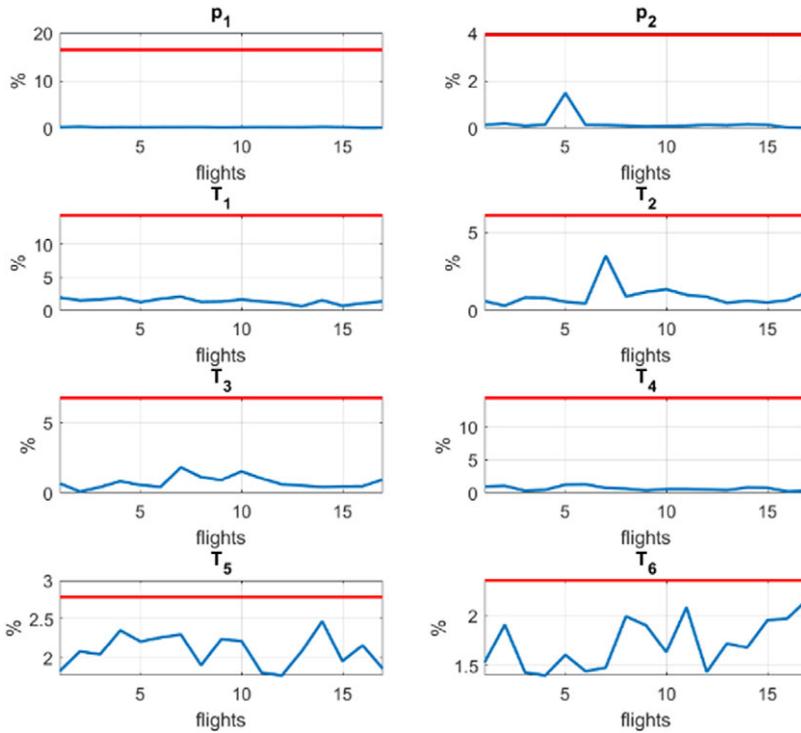
Figure 16. *Error of the hybrid twin (HT) approach (blue line) considering different flights which are not used for the training. The red line refers to the maximum error in the pseudo measurements. The error assigned to a flight is the mean value of the error defined in Equation (21).*
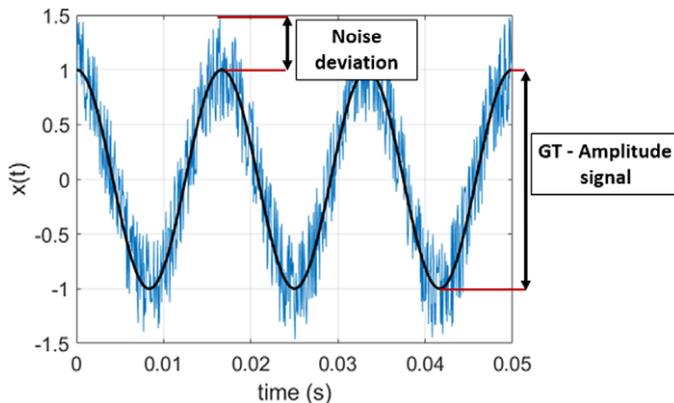
Figure 17. *Sine wave with noise. In the plot, the maximum variation of the signal is indicated as well as the deviation caused by the noise to illustrate the concept used to define the error criterion.*

We also compared the proposed technique in two scenarios: *when it is employed to obtain models from scratch* versus *when it is employed for an enrichment in the HT rationale*.

We concluded that for more complex systems the HT paradigm seems advantageous for two reasons. The first one is that more complex behaviors can be captured (as variables T_5 and T_6). The second one is that, in the HT, the main response is relied on the physics-based model thus guaranteeing that the model is

going to exhibit a behavior coherent with the physical phenomenon at hand. Consequently, just the part of the response which has difficulties in being modeled is carried out by the data-driven model, for instance, degradation or aging.

Competing Interests. The authors declare no competing interests exist.

Data Availability Statement. Data not available due to commercial restrictions. Due to the nature of this research, participants of this study did not agree for their data to be shared publicly, so supporting data is not available.

Author Contributions. Conceptualization, A.S., F.C.; Methodology, A.S., F.C., E.C.; Data curation, A.S., M.C., J.-M.L.P., J.G.; Data visualization, A.S., E.C.; Writing—original draft, A.S., E.C., F.C. All authors approved the final submitted draft.

References

Chinesta F, **Cueto E**, **Abisset E**, **Duval J and Khaldi FE** (2020) Digital and hybrid twins. A new paradigm in data-based engineering and engineered data. *Archives of Computational Methods in Engineering 27*, 105–134.

Chinesta F, **Huerta A**, **Rozza G and Willcox K** (2015) Model order reduction. In *Encyclopedia of Computational Mechanics*. Hoboken, NJ: John Wiley and Sons.

Chinesta F, **Ladevèze P and Cueto E** (2011) A short review in model order reduction based on proper generalized decomposition. *Archives of Computational Methods in Engineering 18*(4), 395–404.

Dassault Systemes (2021) Dymola 2021 x.

Eivazi H, **Guastoni L**, **Schlatter P**, **Azizpour H and Vinuesa R** (2021) Recurrent neural networks and Koopman-based frameworks for temporal predictions in a low-order model of turbulence. *International Journal of Heat and Fluid Flows 90*, 108816.

Erichson NB, **Mathelin L**, **Kutz JN and Brunton SL** (2019) Randomized dynamic mode decomposition. *SIAM Journal on Applied Dynamical Systems 18*(4), 1867–1891.

Guastoni L, **Güemes A**, **Ianiro A**, **Discetti S**, **Schlatter P**, **Azizpour H and Vinuesa R** (2020) Convolutional-Network Models to Predict Wall-Bounded Turbulence from Wall Quantities. arXiv:2006.12483v1.

Hastie T, **Tibshirani R and Friedman JH** (2009) *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York: Springer.

Huang W, **Cao L**, **Sun F**, **Zhao D**, **Liu H and Yu S** (2016) Learning stable linear dynamical systems with the weighted least square method. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI'16*. New York: AAAI Press, pp. 1599–1605.

Kutz JN, **Brunton SL**, **Brunton BW and Proctor JL** (2016) *Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems*. Philadelphia, PA: SIAM-Society for Industrial and Applied Mathematics.

Martín CA, **Méndez AC**, **Sainges O**, **Petiot E**, **Barasinski A**, **Piana M**, **Ratier L and Chinesta F** (2020) Empowering design based on hybrid twintm: Application to acoustic resonators. *Designs 4*(4), 1–14.

Reille A, **Hascoet N**, **Ghnatios C**, **Ammar A**, **Cueto E**, **Duval J**, **Chinesta F and Keunings R** (2019) Incremental dynamic mode decomposition: A reduced-model learner operating at the low-data limit. *Comptes Rendus Mecanique 347*, 780–792.

Sancarlos A, **Cameron M**, **Abel A**, **Cueto E**, **Duval J-L and Chinesta F** (2020) From rom of electrochemistry to AI-based battery digital and hybrid twin. *Archives of Computational Methods in Engineering 28*, 1–37.

Sancarlos A, **Cueto E**, **Chinesta F and Duval J** (2021) A novel sparse reduced order formulation for modeling electromagnetic forces in electric motors. *SN Applied Sciences 3*, 355.

Schmid P (2010) Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics 656*, 5–28.

Schmid P (2011) Application of the dynamic mode decomposition to experimental data. *Experiments in Fluids 50*, 1123–1130.

Schmid P, **Li L**, **Juniper M and Pust O** (2011) Applications of the dynamic mode decomposition. *Theoretical and Computational Fluid Dynamics 25*(41000), 249–259.