2018 European Control Conference (ECC)
June 12-15, 2018. Limassol, Cyprus

Data-driven control design in the Loewner framework: Dealing with stability and noise

Pauline Kergus[1] and Simone Formentin[2] and Charles Poussot-Vassal[1] and Fabrice Demourant[1]

Abstract— **The L-DDC (Loewner Data Driven Control) algorithm is a data-driven controller design method based on frequency-domain input-output data. The identification of the plant is skipped and the controller is designed directly from the measurements using the Loewner approach, known for model approximation and reduction. However, in the L-DDC method, the identified controller is not guaranteed to be stable and the effect of noise on the identified controller is unknown. In this article, we ensure the stability of the controller and propose a solution to deal with noisy data. The method is validated on a numerical example.**

I. INTRODUCTION

In many control engineering applications, no mathematical description of the plant is available or easily accessible. Given some input-output data collected on a system, one can either identify a model of the plant and then, design a controller using any kind of model-based technique (indirect methods), or directly use the experimental data to design a controller (direct methods). The latter option is particularly interesting when a model of the system would be too time-consuming, too complex or too costly to obtain. Furthermore, the identification can result in a complex structure for the plant, and consequently for the controller. A reduction step (for the controller) might be needed, which is also a complex task (see [1]). Of course, an advantage of indirect methods over direct ones is that the model can be used for other purposes (stability and robustness analysis, simulation, etc...). However, direct data-driven methods seem less conservative, and not sensitive to modelling errors, since the selection of the controller is done directly from the experimental data. Moreover, they are less time-consuming since the modelling and/or identification steps are skipped and the resulting control law is tailored to the actual system.

Numerous direct methods have been proposed to try to achieve the best possible performance without using any plant model, beginning with the *unfalsified* concept by Safonov, see [2]. An example is the Iterative Feedback Tuning (IFT, [3]) and its frequency-domain variant in [4]: it finds the controller parameters thanks to an adaptative and iterative control algorithm based on explicit criterion minimization. Another one is Model Free Control (MFC) which has been applied on a complex experimental set-up in [5]. Among all the direct design techniques, the ones proposed in [6], [7] and [8] are iterative methods using the Nyquist criteria so that the obtained controller guarantees the stability of the closed-loop. This is made possible by using an initial stabilizing controller, which is a strong requirement. The methods proposed in [7] and [8] also require a structure for the set of admissible controllers, which can be difficult to choose. Two applications of [6] can be found in [9] and [10].

Another method is the L-DDC algorithm, originally proposed in [11]. This is a one shot method based on frequency-domain data that does not require a initial stabilizing controller. The main advantage of the L-DDC method relies on its simplicity. First, the user does not have to choose a structure for the controller, only the order. As in system identification, the order becomes a tunable parameter allowing to find a compromise between complexity and reliability. Moreover, the specifications are imposed easily through a reference transfer function representing the desired closed-loop behavior. This technique is appealing for engineers for applications when a controller should be synthesized quickly and for which it would be too costly or too complex to identify a model. Moreover, the Loewner framework, which is used to identify the controller, allows to find the minimal representation interpolating a given data-set and it is computationally efficient. However, as the singular value decomposition is sensitive to noise, noisy data have a great impact on the identified controller and no specific solution has been proposed yet to tackle this problem. Furthermore, another limitation of the L-DDC algorithm is that the identified controller can be unstable, which can be a problem if one does not want to introduce unstable dynamics in the open-loop. The objective of this paper is to enhance the L-DDC algorithm by making it more robust to noisy data and by enforcing the stability of the controller, which are, in many applications, strong requirements.

This article is organized in six sections. The problem formulation is detailed in Section II. Section III introduces the Loewner framework which is the frequency-based interpolation technique used in this work and which plays a pivotal role in the proposed approach. Then, the proposed method to design a controller on the basis of frequency-domain data is exposed in Section IV. In comparison with [11], the main contribution is to allow to treat noisy data and to enforce the stability of the controller, these two aspects are detailed in Section IV. Finally, an academic example is considered in Section V to illustrate the method. Conclusions and outlooks are finally given in Section VI.

[1]ONERA, Toulouse, France, pauline.kergus@onera.fr
[2]Dipartimento di Elettronica, Informazione e Biongegneria, Politecnico di Milano, via Ponzio 34/5, 21033 Milano, Italy

II. PROBLEM FORMULATION

The considered problem is to design a controller for a plant P with n_u input-n_y output, respectively denoted u and y, without identifying a dynamical model of the plant. The plant is only characterized by measurement data given as samples of its frequency response $\{\omega_i, \Phi_i\},\ i = 1 \dots N$, where $\Phi_i \in \mathbb{C}^{n_y \times n_u}$ and $\omega_i \in \mathbb{R}_+$. Of course, time-domain data can be used if their Fourier transform gives enough samples of the frequency response in the range of interest. In this paper, we consider noisy data: $\forall i = 1 \dots N, \Phi_i = \mathbf{P}(\imath\omega_i)(1 + N_i)$, where N_i is the noise at ω_i and $\imath$ is the complex variable. Let us denote $\dagger$ the Moore-Penrose inverse. $\overline{s} \in \mathbb{C}$ denotes the complex conjugate of $s \in \mathbb{C}$.

As detailed in [11], the L-DDC algorithm allows to identify a controller, without any *a priori* structure. The specifications are expressed as a reference transfer function M which represents the desired behavior in closed-loop. The L-DDC algorithm consists in two steps: 1) the closed-loop objective transfer M and the open-loop experimental data of the plant are used to get the frequency response of the "ideal" controller denoted $K(\imath\omega_i)$ for a limited frozen set of frequency values; 2) then, this frequency response will be approximated by a linear time-invariant input-output model $\hat{K}$ of order n. The controller K is "ideal" in the sense that it would give exactly the objective transfer function M if inserted in the closed-loop.

In this paper, two improvements of the second step of the L-DDC algorithm are introduced to both ensure the stability of the identified controller and to be robust to noisy data. In fact, noisy data has a big impact on the controller poles location and no special treatment has been proposed yet. Before detailing these new aspects, the Loewner framework is recalled in Section III.

III. PRELIMINARY RESULTS: LOEWNER-BASED IDENTIFICATION

The Loewner approach, exposed in [12], is usually used for model approximation and reduction. It constructs a descriptor model in state-space form directly from the frequency-domain data so that the model performs a barycentric Lagrange interpolation (see [13] for further details). In this article, it is used to identify an interpolating model $\hat{K}_r$ of the "ideal" controller.

In order to construct such a realization, the following inputs are required: (i) left interpolation point $(\mu_j)_{j=1\dots q} \in \mathbb{C}$ and left tangential directions $(\mathbf{l}_j)_{j=1\dots q} \in \mathbb{C}^{n_y}$, and (ii) right interpolation points $(\lambda_i)_{i=1\dots k}$ and right tangential directions $(\mathbf{r}_i)_{i=1\dots k} \in \mathbb{C}^{n_u}$. The interpolation points correspond to the data of the model to be identified, which are, in our approach, the samples of the frequency response of the "ideal" controller $\{\omega_{i_f}, K(\imath\omega_{i_f})\},\ i_f = 1 \dots N_f$. The computation of the samples $K(\imath\omega_{i_f})$ and the separation of the data between left and right interpolation points, respectively $(\mu_j)_{j=1\dots q}$ and $(\lambda_i)_{i=1\dots k}$, are explained in Section IV.

The following vectors are then defined from the input data:

$$\left\{ \begin{array}{rcll} \mathbf{v}_j^T &=& \mathbf{l}_j^T K(\mu_j) & \forall j = 1 \dots q \\ \mathbf{w}_i &=& K(\lambda_i)\mathbf{r}_i & \forall i = 1 \dots k \end{array} \right. \quad (1)$$

Note that in case of a SISO system, the tangential directions $(\mathbf{l}_j)_{j=1\dots q}$ and $(\mathbf{r}_i)_{i=1\dots k}$ are useless and can be fixed to 1. In the Loewner approach, one seek for a model $\hat{K}$ that interpolates the data as follows:

$$\left\{ \begin{array}{rcl} \mathbf{l}_j^T \hat{K}_r(\mu_j) &=& \mathbf{l}_j^T K(\mu_j) = \mathbf{v}_j^T \ \forall j = 1 \dots q \\ \hat{K}_r(\lambda_i)\mathbf{r}_i &=& K(\lambda_i)\mathbf{r}_i = \mathbf{w}_i \ \forall i = 1 \dots k \end{array} \right. \quad (2)$$

Based on the $(\mu_j, \mathbf{l}_j^T, \mathbf{v}_j^T)$ and $(\lambda_i, \mathbf{r}_i, \mathbf{w}_i)$ data, one can construct the Loewner $\mathbb{L}$ and shifted Loewner $\mathbb{L}_\sigma$ matrices as follows, for all $j = 1 \dots q$ and $i = 1 \dots k$:

$$[\mathbb{L}]_{j,i} = \frac{\mathbf{v}_j^T \mathbf{r}_i - \mathbf{l}_j^T \mathbf{w}_i}{\mu_j - \lambda_i}\ ,\ [\mathbb{L}_\sigma]_{j,i} = \frac{\mu_j \mathbf{v}_j^T \mathbf{r}_i - \lambda_i \mathbf{l}_j^T \mathbf{w}_i}{\mu_i - \lambda_j}\ . \quad (3)$$

As explained in [12], one of the main advantages of the Loewner framework is that the minimal Mc Millan order of the interpolating model $\hat{K}_r$ can be obtained by evaluating $r = rank[\mathbb{L}, \mathbb{L}_\sigma]$. By applying the singular value decomposition:

$$[\mathbb{L}, \mathbb{L}_\sigma] = Y\Sigma_l\tilde{X}^*, \qquad \begin{bmatrix} \mathbb{L} \\ \mathbb{L}_\sigma \end{bmatrix} = \tilde{Y}\Sigma_r X^*\ , \quad (4)$$

where $\Sigma_l, \Sigma_r \in \mathbb{R}^{r\times r}$, the descriptor model $\hat{K}_r = (E_r, A_r, B_r, C_r, 0)$ interpolates the data of the "ideal" controller, where the realization matrices of the model $\hat{K}$ are then computed as follows :

$$E_r = -Y^*\mathbb{L}X,\ A_r = -Y^*\mathbb{L}_\sigma X,\ B_r = Y^*V,\ C_r = WX\,. \quad (5)$$

In addition to determining the smallest exact interpolating model, the Loewner framework allows to control the complexity of the identified model: by keeping the n largest singular values of the decomposition of the Loewner pencil only (4), i.e. the first n columns of X and Y, the obtained realization is a n^{th} order one. That is how model reduction is done through the Loewner framework. In [11], the controller model is directly reduced to the objective order n but then the obtained controller is not necessarily stable and its poles can be affected by the noise.

IV. LOEWNER BASED DATA-DRIVEN CONTROL DESIGN

In this section, two additional features are added to the original L-DDC algorithm. The first one is a stability constraint of the identified model: to that purpose, we use the algorithm introduced in [14] which finds the best approximation of a rational model in the RH_∞ spaces of real rational functions in the Hardy space $\mathcal{H}_\infty$. The second one a variant of the Loewner algorithm proposed in [15] that allows to identify an approximation of the original system even for high noise levels. These two modifications of the classical Loewner framework will be detailed before summing up the enhanced L-DDC algorithm.

A. Enforcing stability of the controller

In many applications, for example when input saturation exist, one does not want to introduce unstable dynamics in the open-loop to preserve the internal stability. Therefore it is important to guarantee the stability of the controller.

This was not possible in the original version of the L-DDC algorithm proposed in [11] but, in this paper, we propose to use the algorithm introduced in [14] to ensure the stability of the identified controller. Note that this algorithm has already been combined with the Loewner framework in [16] in the area of model reduction.

Given an unstable continuous LTI descriptor system, the method proposed in [14] allows to find a stable one which is the best approximation of the original system in the space RH_∞. In our case, this algorithm will be applied to the the interpolating descriptor model $\hat{K}_r$, obtained at the end of Section III, which is of order $r = rank[\mathbb{L}, \mathbb{L}_\sigma]$ (McMillan degree). The matrices $E_r, A_r \in \mathbb{R}^{r\times r}$, $B_r \in \mathbb{R}^{r\times n_y}$, $C_r \in \mathbb{R}^{n_u\times r}$ of the system are obtained through Equation (5). The goal is to obtain a r^{th} order stable controller $\hat{K}_r^s$ that is an optimal RH_∞-approximation of $\hat{K}_r$, meaning that $\hat{K}_r^s$ solves:

$$\hat{K}_r^s = \underset{K\in\mathbb{S}^+_{r,n_u,n_y}}{\arg\min} \left|\left|\hat{K}_r - K\right|\right|_\infty . \tag{6}$$

Let us introduce the following notations:

$$\begin{aligned}
\mathbb{S}_{r,n_u,n_y} &= \mathbb{R}^{r\times r}\times\mathbb{R}^{r\times r}\times\mathbb{R}^{r\times n_u}\times\mathbb{R}^{n_y\times r}\times\mathbb{R}^{n_y\times n_u},\\
\mathbb{S}^+_{r,n_u,n_y} &= \left\{(E,A,B,C,D)\in\mathbb{S}_{r,n_u,n_y};\mathbb{C}_{\geq 0}\subset\rho(E,A)\right\},\\
\mathbb{S}^-_{r,n_u,n_y} &= \left\{(E,A,B,C,D)\in\mathbb{S}_{r,n_u,n_y};\mathbb{C}_{\leq 0}\subset\rho(E,A)\right\},
\end{aligned} \tag{7}$$

where $\sigma(E, A)$ is the set of eigenvalues of (E, A) and $\rho(E, A)$ is the resolvent set: $\rho(E, A) = \mathbb{C} \setminus \sigma(E, A)$.

The first step is to decompose $\hat{K}_r$ into $\hat{K}_+ \in \mathbb{S}^+_{r,n_u,n_y}$ and $\hat{K}_- \in \mathbb{S}^-_{r,n_u,n_y}$. The existence of such a decomposition is proved in [14] and a method is proposed to compute it (practically, one can use the Matlab function $\mathtt{stabsep}$). Note that $\hat{K}_+$ is the optimal approximation of $\hat{K}_r$ in RH_2.

The unstable part $\hat{K}_- = (E^-, A^-, B^-, C^-, D^-)$ is then used to compute $\hat{K}_r^s$. The controllability and observability gramians of $\hat{K}_-$, denoted $\mathcal{W}_c$ and $\mathcal{W}_o$ respectively, are computed by solving the following generalized Lyapunov equations:

$$\left\{\begin{array}{l} A^-\mathcal{W}_cE^{-T} + E^-\mathcal{W}_cA^{-T} + B^-B^{-T} = 0\\ A^{-T}\mathcal{W}_oE^- + E^{-T}\mathcal{W}_oA^- + C^-C^{-T} = 0\end{array}\right. \tag{8}$$

Let us introduce $\sigma_1 = \sqrt{\max(\sigma(\mathcal{W}_o^T\mathcal{W}_c))}$ and the matrix $R = \mathcal{W}_oE^-\mathcal{W}_cE^{-T} - \sigma_1^2 I$, where I is the identity matrix. The optimal RH_∞-approximation is then given by $\hat{K}_r^s = \hat{K}_+ + (\hat{E}, \hat{A}, \hat{B}, \hat{C}, \hat{D})$, where the matrices $(\hat{E}, \hat{A}, \hat{B}, \hat{C}, \hat{D})$ are computed as follows (the reader can refer to [14] for details):

$$\begin{aligned}
&\hat{E} = E^{-T}R, \qquad \hat{B} = E^{-T}\mathcal{W}_oB^-,\\
&\hat{C} = C^-\mathcal{W}_cE^{-T}, \hat{A} = -A^{-T}R - C^{-T}\hat{C}, \hat{D} = D^-.
\end{aligned} \tag{9}$$

In the next paragraph, a variant of the classical SVD implementation of the Loewner framework is proposed to make the selection of the poles robust in presence of noise.

B. Dealing with noisy data in the Loewner framework

The Loewner framework can identify a system from given noise-free measurements in the frequency-domain (see [1]). An analysis of the effects of noise on the performances of the Loewner approach is provided in [15], it exhibits poor performances for high noise levels. Noise affects the recovered poles: the largest singular values of the Loewner pencil does not necessarily reflect the physical poles of the system and often include noise dynamics. In this case, overmodeling is the only way, in the Loewner context, to capture the physical poles of the original system.

In the classical Loewner framework, the poles of the system are determined through a rank revealing factorization. In order to make the selection of the poles robust to noise, this SVD approach is replaced by ordering the poles of the high-order system according to the norm of the associated residues, as explained in [15]. This approach is recalled hereafter. It will be applied to the stable r^{th}-order stable $\hat{K}_r^s = (E_r^s, A_r^s, B_r^s, C_r^s, D_r^s)$ obtained in the previous paragraph as the optimal approximation of the interpolating controller in RH_∞. The objective is to obtain a stable reduced-order controller $\hat{K}_n$ by selecting the poles in a noise-proof way.

Instead of using the SVD approach, the importance of a pole λ_i of $\hat{K}_r^s$, which is an eigenvalues of (E_r^s, A_r^s), is measured by the norm of the corresponding residue $r_i = (C_r^s\mathbf{x}_i)(\mathbf{y}_i^TE_r^s\mathbf{x}_i)^{-1}(\mathbf{y}_i^TB_r^s)$, where $\mathbf{x}_i \in \mathbb{R}^r$ and $\mathbf{y}_i \in \mathbb{R}^r$ are the right and left eigenvectors of (E_r^s, A_r^s) respectively associated with the eigenvalue λ_i. This strategy is based on the residue expansion of the transfer function: a pole with a larger residue norm contributes more to the response of the system, while the rest do not influence it that much.

As explained in [15], using this technique to select the poles instead of the classical SVD approach, the approximated poles are within appropriate pseudospectra bounds corresponding to the noise level in comparison with the physical poles of the controller.

Then, the n poles of the controller are the ones corresponding to the n^{th} largest residue norms. The poles are then ordered downward, so that λ_1 is the pole with the highest dominance measure and λ_r has the smallest one. The poles of the reduced stable model are $[\lambda_1 \dots \lambda_n]$. After that, it is possible to adjust the residues and the D-term to fit the data by solving the following least squares problem:

$$\underset{r_i,D}{min} \sum_{j=1}^{N} \left|\left|\sum_{i=1}^{n}\frac{r_i}{\imath\omega_j - \lambda_i} + D - K(\imath\omega_j)\right|\right|_F^2 . \tag{10}$$

Finally, the n^{th} order controller $\hat{K}_n(s) = D + \sum_{i=1}^{n}\frac{r_i}{s-\lambda_i}$ is reconstructed as a transfer function by keeping the poles corresponding to the n^{th} largest dominance measures only and the residues obtained by solving (10).

Since the poles of the reduced-order controller come from the stable model $\hat{K}_r^s$, the obtained n^{th}-order controller $\hat{K}_n$ remains stable.

C. Enhanced L-DDC algorithm

This paragraph sums up the L-DDC procedure, indicated in Algorithm 1, to identify a stable controller of a given order n on the basis of potentially noisy frequency-domain measurements from an unknown plant P (see Figure II).

Algorithm 1: L-DDC algorithm

Data:

- Samples of the frequency response of the plant $\{\omega_i, \Phi_i\},\ i = 1 \dots N$. Note that $q = k = N_f$.
- Objective order n for the controller
- Reference transfer function M

Solution:

1) Compute the samples of the frequency response of the "ideal" controller as follows, $\forall i = 1 \dots N$,
$$K(\imath\omega_i) = (\Phi_i - \Phi_i M(\imath\omega_i))^{\dagger} M(\imath\omega_i).$$
2) Using the Loewner framework, compute the interpolating descriptor controller model $\hat{K}_r = (E_r, A_r, B_r, C_r, 0)$ through (5).
3) Decompose $\hat{K}_r$ into the stable part $\hat{K}_+ \in \mathbb{S}^+_{r,n_u,n_y}$ and the unstable part $\hat{K}_- \in \mathbb{S}^+_{r,n_u,n_y}$.
4) Compute the gramians of $\hat{K}_-$ according to (8) and then the matrices $(\hat{E}^s_r, \hat{A}^s_r, \hat{B}^s_r, \hat{C}^s_r, \hat{D}^s_r)$ following (9) to form $\hat{K}^s_r$.
5) Compute the eigenvalues λ_i, $i = 1 \dots r$ of (E^s_r, A^s_r) and the corresponding left and right eigenvectors.
6) Compute the residue for each eigenvalue. The n eigenvalues corresponding to the n^{th} largest residue norms are the poles of the controller.
7) Adjust the residues r_i, $i = 1 \dots n$, and the D-term by solving (10) to obtain he n^{th}-order controller $\hat{K}_n$.

1) From the data to the "ideal" controller:

The idea of the first step of Algorithm 1 is to exploit experimental data $\{\omega_i, \Phi_i\},\ i = 1 \dots N$ to determine the frequency response of the "ideal" controller $K(\imath\omega_{i_f})$, which would allow to get exactly the desired closed-loop transfer materialized by the function M .

Remark 1: If the experimental data is given as a time data-set of input-output measurements of the plant, a Fourier transform allows to estimate samples of the plant frequency response if the input signal sufficiently excite the system.

2) Using the Loewner framework to get an interpolating model:

Then, the data of the "ideal" controller is interpolated through the Loewner framework recalled in Section III (step (2) of Algorithm 1). The interpolatory property of the Loewner framework makes our method easy to use since there no *a priori* structure of the controller is required.

In order to use the Loewner framework, the data are equally separated between left and right interpolation points. In [13], the author recommends to alternate between left and right to avoid rank loss in the Cauchy-like Loewner matrix $\mathbb{L}$. This distribution is used in the previous version of the L-DDC algorithm, and the reader can refer to [11] for further details.

V. SIMULATION RESULTS

The proposed example is drawn from Matlab's Robust Control Toolbox and treats the control design for a SISO 9th-order model of a head-disk assembly in a hard-disk drive. In the Matlab example, **hinfstruct** is used to design a robust controller such that a desired open-loop response is achieved while satisfying a certain performance measure, see [17]. This example was also used in [6].

The desired open-loop transfer function is:
$$L(s) = \frac{s + 10^6}{1000s + 1000},$$
from which the closed-loop reference transfer can be computed. For this application, the method developed in [6] obtains the following controller: $C(z) = 10^{-4}\frac{2.287z^2 - 3.15z + 0.8631}{(z-1)(z-0.8598)}$. The sampling period is $T_e = 2ms$. The presence of an integrator is forced because there is one in the chosen initial stabilizing controller $K_c(z) = \frac{10^{-6}}{z(z-1)}$, giving the controller a certain structure, while the method proposed in this paper does not allow to structure the controller. Then, the only way to compare properly the results of [6] and of the method presented in this paper is to synthesize two controllers of the same order, here $n = 2$.

Using Matlab, samples of the frequency response of the plant $P(\imath\omega_i)$, for $i = 1 \dots N$ with $N = 500$, are computed for 500 logarithmically spaced frequencies in the interval $\left[10,\ 5 \times 10^4\pi\right] rad.s^{-1}$ (the upper limit corresponds to the Nyquist frequency). Then, noise of given Signal Noise Ratio (SNR) is added to consider noisy data: for all $i = 1 \dots N$,$\Phi_i = P(\imath\omega_i)(1 + N_i)$, where the noise is defined as in [15]: $N_i = 10^{-\frac{SNR}{10}}(randn(1) + \imath\ randn(1))$.

Then, the frequency response of the "ideal" controller K, which would give L exactly if inserted, is computed: $\forall i = 1 \dots N,\ K(\imath\omega_i) = \frac{L(\imath\omega_i)}{\Phi_i}$.

The enhanced L-DDC algorithm is then applied. The different steps of Algorithm 1 are illustrated in the noise-free case on Figure 1. First, the frequency response of the ideal controller is represented by the blue dots. Then, these data samples are interpolated through the Loewner framework, giving a 10^{th}-order model $\hat{K}_r$ indicated by the solid yellow line ($r = 10$). Its optimal RH_∞-approximation $\hat{K}^s_r$ is given by the solid magenta line. Finally, after the order reduction, the 2^{nd}-order controller $\hat{K}$ is represented by the cyan dashed line. In this benchmark, in order to compare our method to the one presented in [6], we chose to use $n = 2$.

This example is particularly interesting since the system is a non minimum phase one, i.e. it exhibits a pair of unstable zeros. According to the computation of the "ideal" controller, the interpolating controller of order $r = 10$ will have two unstable poles due to the compensation of the unstable zeros, and this is not a proper solution to control a non minimum phase system. Using the method developed in [14] allows the user to avoid the compensation of unstable zeros and to find the closest option by computing the optimal RH_∞-approximation.

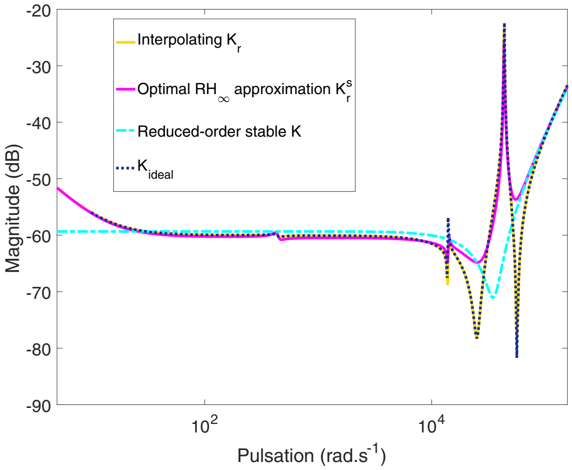

Fig. 1. Frequency response of the "ideal" controller K (blue dots), of the interpolating model $\hat{K}_r$ (solid yellow), its RH_∞-approximation $\hat{K}_r^s$ (solid magenta) and the reduced-order stable controller $\hat{K}_n$ (dash-dotted cyan, $n = 2$).

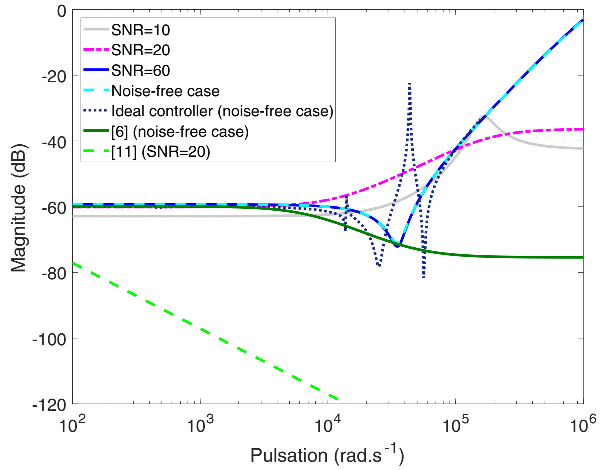

Fig. 2. Frequency response of the "ideal" controller K in the noise-free case (blue dots) and of the identified 2^{nd}-order controllers for different noise levels: SNR=10 (solid grey), SNR=20 (dash-dotted magenta), SNR=60 (solid blue) and in the noise-free case (dashed cyan). The controller obtained in [6] in the noise-free case is indicated by the solid dark green line. The former version of the L-DDC algorithm [11] for SNR=20 gives the controller in dashed light green.

Figure 2 exhibits the results of the identification of the controller for noisy data-sets with different SNR. First, the classic L-DDC algorithm fails to find the "physical" poles that are not due to noise for the reduced-order controller. However, with the enhanced L-DDC algorithm, the selection of the poles is robust to noise. As expected, when the noise level increases, it is more difficult to recover the dynamic of the "ideal" controller, the performances deteriorate with noise, but the identified controllers still show a coherent behaviour.

Using the original system P, it is possible to compute the resulting open-loops with the controller obtained in [6] and the enhanced L-DDC one. Their frequency response are indicated on Figure 3. Even for a high noise level SNR=10, we still obtain an acceptable open-loop considering the objective L, as seen on Figure 3. Note that the model of the plant P is used here to validate our method on this numerical example only, the design of the controller is completely data-driven.

Note that our method allows to find a stabilizing controller (even if it is not originally guaranteed by the method), and that with a same order for the controller ($n = 2$) as in [6], we achieve similar performances. The main advantage of the enhanced L-DDC algorithm over [6] is that it does not require an initial stabilizing controller, which can sometimes be a strong assumption. However, the method proposed in this paper cannot guarantee that the identified controller stabilizes the plant in closed-loop.

In the noise free case, the $\mathcal{H}_2$ error between the desired open-loop L and the one obtained with the controller designed with the enhanced L-DDC algorithm is $e_{\mathcal{H}_2} = 0.5859$ while the method developed in [6] achieves $e_{\mathcal{H}_2} = 0.7248$. The error is higher than with the enhanced L-DDC algorithm, it can be explained by the fact that the structure of the controller is constrained by the choice of the initial stabilizing controller, while the method proposed in this paper does not require any knowledge *a priori*.

In order to generalize, some statistical informations are given on Figure 4. First, as expected, the median of the error between the desired open-loop and the one obtained with the enhanced L-DDC algorithm increase with the noise level. However, it remains acceptable: for SNR=10 for example, the maximum error does not exceed the one obtained with the method developed in [6].

Figure 4 also exhibits the $\mathcal{H}_2$ error between the desired open-loop and the ones obtained with the controllers identified with the previous version of the L-DDC algorithm presented in [11]. For SNR=20, the results of the identification can vary a lot between two different noisy data-sets with the algorithm of [11]. The extension presented in this paper is more robust to noise: the variance of the error over the 50 tested data-sets is lower for the enhanced L-DDC algorithm than for its previous version from [11] for both SNR values considered here. However, note that if for SNR=20, the method proposed in this paper is robust to noisy data (low variance of the error), the identified controller can vary significantly for higher noise level (the variance for SNR=10 is bigger), but in reasonable bounds.

VI. CONCLUSIONS

In this paper, an extension of the L-DDC (Loewner Data-Driven Control) has been proposed. In addition to the previous version of the L-DDC approach, a post-processing method to obtain an optimal RH_∞-approximation of the controller is used to ensure the stability of the controller. Also, the reduction step is no longer based on the classical SVD implementation of the Loewner framework, which is too sensitive to noise. The selection of the poles is now

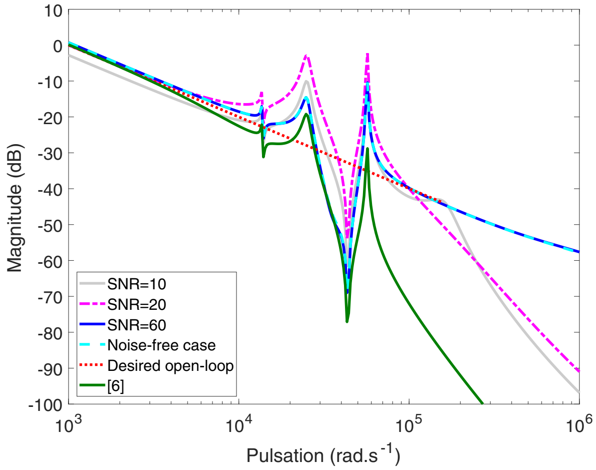

Fig. 3. Frequency response of the desired open-loop L (blue dots) and of the open-loops with the identified 2^{nd}-order controllers for different noise levels: SNR=10 (solid grey), SNR=20 (dash-dotted magenta), SNR=60 (solid blue) and in the noise-free case (dashed cyan). The open-loop achieved in [6] in the noise-free case is indicated by the solid dark green line.

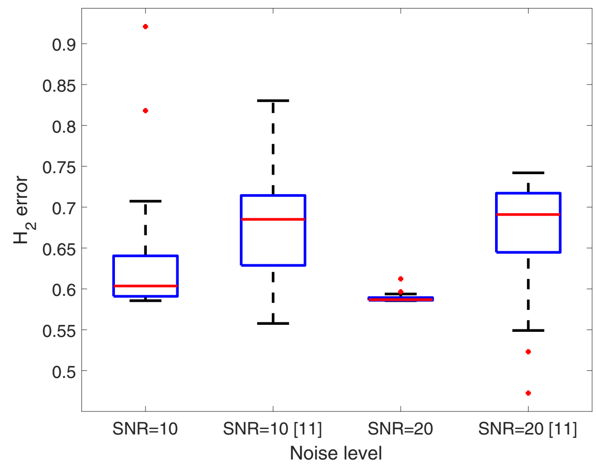

Fig. 4. $\mathcal{H}_2$ error between the desired open-loop L and the ones obtained with the controller identified with the enhanced L-DDC algorithm and the former version of the L-DDC algorithm [11] for different noise levels (SNR=10 and 20, 50 different data-sets were used for both SNR values). The central red mark indicates the median of the error, and the bottom and top edges of the blue box indicate the 25th and 75th percentiles, respectively. The black whiskers extend to the most extreme data points not considered outliers, and the outliers are plotted individually (red crosses).

done according to their residue norms, thus it is based on the pole-residue expansion.

The problem formulation is the same: the specifications are imposed easily through a reference transfer function representing the desired closed-loop behavior. Despite its simplicity, this approach seems to provide good performances. First, the frequency response of the "ideal" controller is computed thanks to frequency-domain data from the plant and the reference transfer. This controller is called "ideal" because it would give exactly the objective if inserted in the closed-loop. Then, the Loewner framework is used to interpolate this "ideal" frequency response. The optimal RH_∞-approximation of the interpolating model is computed and its order is reduced in a noise-proof way to obtain the desired controller.

The main advantage of the L-DDC method relies on its simplicity, the user does not have to choose a structure for the controller, only the order, which becomes a tunable parameter allowing to find a compromise between complexity and reliability. This technique is appealing for engineers for applications when a controller needs to be synthesized quickly and for which it would be too costly or too complex to identify a model. It is a one shot method and the obtained controller is tailored to the actual system.

Furthermore, the improvements of the L-DDC algorithm allow to enforce the stability of the controller and make the approach robust to noise. However, the proposed approach does not allow the use of model-based analysis of stability and robustness, therefore future research will address this point.

References

[1] A.C. Antoulas, S. Lefteriu, and A.C. Ionita. A tutorial introduction to the Loewner framework for model reduction. *Model Reduction and Approximation for Complex Systems*, 2015.

[2] MG Safonov and TC Tsao. The unfalsified control concept and learning. *IEEE Transactions on Automatic Control*, 1997.

[3] H. Hjalmarsson. Iterative feedback tuning: an overview. *International journal of adaptive control and signal processing*, 2002.

[4] L.C. Kammer, R.R. Bitmead, and P.L. Bartlett. Direct iterative tuning via spectral analysis. *Automatica*, 2000.

[5] RE Precup, MB Radac, RC Roman, and EM Petriu. Model-free sliding mode control of nonlinear systems: Algorithms and experiments. *Information Sciences*, 2017.

[6] A. Karimi and C. Kammer. A data-driven approach to robust control of multivariable systems by convex optimization. *Automatica*, 2017.

[7] T Hori, K Yubai, D Yashiro, and S Komada. Data-driven controller tuning for sensitivity minimization. In *International Conference on Advanced Mechatronic Systems*, 2016.

[8] P Apkarian and D Noll. Structured H_∞ control of infinite dimensional systems. *arXiv preprint:1707.02052*, 2017.

[9] A. Nicoletti, M. Martino, and A. Karimi. A data-driven approach to power converter control via convex optimization. In *1st Conference on Control Technology and Applications*, 2017.

[10] C.M. Kammer, A. Pascal Nievergelt, G. Fantner, and A. Karimi. Data-driven controller design for atomic-force microscopy. In *20th IFAC World Congress*, 2017.

[11] P. Kergus, C. Poussot-Vassal, F. Demourant, and S. Formentin. Frequency-domain data-driven control design in the Loewner framework. *20th IFAC World Congress*, 2017.

[12] A.J. Mayo and A.C. Antoulas. A framework for the solution of the generalized realization problem. *Linear algebra and its applications*, 2007.

[13] A.C. Ionita. *Lagrange rational interpolation and its applications to approximation of large-scale dynamical systems*. PhD thesis, Rice University, 2013.

[14] M. Köhler. On the closest stable descriptor system in the respective spaces $\mathcal{RH}_2$ and $\mathcal{RH}_\infty$. *Linear Algebra and its Applications*, 2014.

[15] S. Lefteriu, A.C. Ionita, and A.C. Antoulas. Modeling systems based on noisy frequency and time domain measurements. *Perspectives in Mathematical System Theory, Control, and Signal Processing*, 2010.

[16] I.V. Gosea and A.C. Antoulas. Stability preserving post-processing methods applied in the loewner framework. In *20th Workshop on Signal and Power Integrity*, 2016.

[17] P. Apkarian and D. Noll. Nonsmooth h_∞ synthesis. *Transactions on Automatic Control*, 2006.