Model-Free Feedback Control Synthesis From Expert Demonstration

Franco Blanchini, *Senior Member, IEEE*, Fabrizio Dabbene, *Senior Member, IEEE*, Gianfranco Fenu, Felice Andrea Pellegrino, *Member, IEEE*, and Erica Salvato

(Special Section on Data-Driven Analysis and Control)

***Abstract*—We show how it is possible to synthesize a stabilizing feedback control, in the complete absence of a model, starting from the open-loop control generated by an expert operator, capable of driving a system to a specific set-point. We assume that the system is linear and discrete time. We propose two different controls: a linear dynamic and a static, piecewise linear, one. We show the performance of the proposed controllers on a ship steering problem.**

***Index Terms*—Stabilizing feedback control, data-driven control, linear systems.**

I. Introduction

DATA-DRIVEN control techniques enable controller synthesis in those scenarios where first-principle models cannot be formulated, the models are too complex for control engineering, or the identification of model parameters is excessively expensive [1]. In a first stage, data can be used for more accurate inference of dynamic models, i.e., once a sufficient amount of informative data has been collected, they can be used to learn a dynamical model of the plant and to apply the well-known model-based control techniques on it [2]. Besides that, a more interesting challenge is that of completely avoiding the dynamic model design procedure, and directly exploiting experimental measurements to build the controller itself, namely performing *data-driven controller synthesis* [3], [4], [5], [6]. In this context, several solutions have been proposed. A model-free synthesis procedure of minimum-energy open-loop controllers for linear, unconstrained, discrete-time systems is proposed in [7]. It relies only on experimental data collected in a finite number of control experiments. An extension to the case of noisy data is proposed in [8], while some applications are illustrated in [9]. Solutions based on the *fundamental lemma* [10] are instead proposed in [11], [12], facing linear quadratic regulator (LQR) problems, while some extensions to the case of nonlinear systems are presented in [13], [14], where also the case of noisy-corrupted data is considered. Model predictive control (MPC) problems are instead faced by [15], [16], [17], [18].

In the present paper, inspired by the *learning-by-demonstration* paradigm [19], we show how to perform a one-shot synthesis of a controller, relying only on data recorded during a single experiment performed by an expert operator, without involving any learning procedure. We only assume that the operator, who has no knowledge of the model, is able to bring the system toward (or at least close to) the target state. Basically, we split the control problem into two subsequent stages:

1) *expert demonstration:* an operator performs, once, the manual control of the system, carrying it from an initial condition to the target, relying only on his own experience;
2) *controller inference:* a feedback mechanism is synthesized from the recorded operation, which is stabilizing, and able to reproduce the operator command from the same conditions (same initial state).

We stress that the expert demonstration has to be performed one-shot: no repeated experiments or trial-and-error procedures are allowed. Precisely, we assume that the trajectory performed by the operator is *self-absorbing* (as defined in Section I-B), i.e., the final state of the trajectory is absorbed in the convex hull of the previous states and their opposite.

To solve the problem we use the machinery previously introduced in [20], [21], [22]. The difference is that in the above-mentioned works, precise objective functionals were introduced and optimized for the nominal initial condition, based on the model dynamics knowledge. Preliminary results in which data-driven relatively optimal controllers, both in static and dynamic form, are derived directly from experimental data are presented in [23], [25]. Here, instead, we decide not to consider any "mathematical optimality", but rather we assume that the operator's solution, as suggested by his experience, is

Manuscript received 9 December 2022; revised 3 February 2023; accepted 21 February 2023. Date of publication 2 March 2023; date of current version 24 May 2023. This work was supported in part by the Italian Ministry for Research in the Framework of the 2017 Program for Research Projects of National Interest (PRIN) under Grant 2017YKXYXJ. Recommended by Senior Editor M. K. Camlibel. *(Corresponding author: Erica Salvato.)*

Franco Blanchini is with the Dipartimento di Scienze Matematiche, Informatiche e Fisiche, Università degli Studi di Udine, 33100 Udine, Italy (e-mail: blanchini@uniud.it).

Fabrizio Dabbene is with the Institute of Electronics, Computer and Telecommunication Engineering, CNR-IEIIT, 10129 Turin, Italy (e-mail: fabrizio.dabbene@ieiit.cnr.it).

Gianfranco Fenu, Felice Andrea Pellegrino, and Erica Salvato are with the Department of Engineering and Architecture, University of Trieste, 34127 Trieste, Italy (e-mail: fenu@units.it; fapellegrino@units.it; erica.salvato@dia.units.it).

Digital Object Identifier 10.1109/LCSYS.2023.3251575

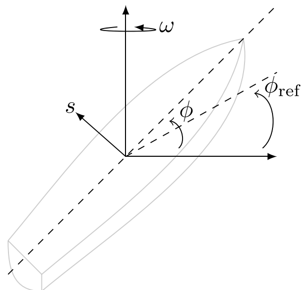

Fig. 1. The ship steering problem.

empirically what we wish the system to replicate during its operation. We suggest two different solutions to the problem, and, as in [20], [21], [22], [23], [25], we assume that the considered model, even if unknown, is linear. The first solution consists of a linear dynamic compensator: we provide explicit formulas to derive the matrices based on the only available demonstration, and we also ensure the desirable property that the compensator, besides stabilizing, is also stable (strong stabilization). In the second approach we propose a nonlinear, but piecewise linear, static feedback controller whose realization is, in general, more complex than the dynamic solution, but, being static, does not lead to compensator stability problems.

A. Motivating Example

Consider the problem of steering a ship, depicted in Figure 1. The state variables are the sway velocity s, the yaw angular speed ω, and the yaw angle ϕ. The control input u is the rudder angle. The model is presented in [24]. Still, we assume that such a model is not available. The steering operation is performed by an expert operator, or any device capable of producing a satisfactory open-loop control, who drags the state from an initial condition $\bar{x}(0)$, chosen as nominal, to a reference steady state that we assume, for convenience, to be 0, by manually controlling the rudder angle. State and control variables are sensed by suitable sensors and recorded. Assuming that the operator maneuver is successful, we wish to synthesize stabilizing feedback controllers able to perform the same task.

B. Principle of the Self-Absorbing Trajectory (Main Result)

Consider a linear time-invariant discrete-time system and an initial state $\bar{x}(0)$. If there exists a control trajectory $\bar{u}(k)$ $k = 0, \ldots N-1$, such that the corresponding state sequence $\bar{x}(k)$, $k = 0, \ldots N$ has the property that the final state $\bar{x}(N)$ is in the interior of the convex hull of the previous states and their opposite $\pm\bar{x}(k)$ (is "absorbed"), then there exists a feedback controller, with no feedforward action, inferable from $\bar{x}(k)$ and $\bar{u}(k)$. Such a control

1) is stabilizing;
2) it recovers the nominal trajectory for $x(0) = \bar{x}(0)$;
3) if the initial condition is aligned with $\bar{x}(0)$, i.e., $x(0) = \alpha\bar{x}(0)$, then the trajectory is proportional $x(k) = \alpha\bar{x}(k)$;
4) it does not require state initialization: it is either static, or its initial state is supposed to be 0;
5) it is synthesized only once, based exclusively on the single successful experiment, with no knowledge of A and B. No repeated learning procedures are required.

In the sequel, we assume that the operator trajectory is self–absorbing. This property certifies the operator's ability to drive the state $\bar{x}(N)$ "sufficiently close" to 0[1] comparing the size of $\bar{x}(N)$ with the trajectory size.

II. Synthesis Technique

To solve the problem we use the same concepts introduced in [20], [21], [22] for discrete-time and in [26] for continuous-time systems.

Consider the linear time-invariant discrete-time system:

$$x(k+1) = Ax(k) + Bu(k), \tag{1}$$

where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, while $x(k) \in \mathbb{R}^n$ and $u(k) \in \mathbb{R}^m$ denote respectively the state and the input at time $k \in \mathbb{N}$.

Assume that an expert operator provides one successful state trajectory $\mathcal{X}$ and the corresponding input trajectory $\mathcal{U}$:

$$\begin{aligned} \mathcal{X} &= [\bar{x}(0)\ \bar{x}(1)\ \cdots\ \bar{x}(N-1)] \\ \mathcal{U} &= [\bar{u}(0)\ \bar{u}(1)\ \cdots\ \bar{u}(N-1)]. \end{aligned} \tag{2}$$

Assumption 1 (Self–Absorbing Trajectory): The convex hull of vectors $\mathcal{X}$ and their opposite, $-\mathcal{X}$ has a non-empty interior and $\bar{x}(N)$ is in this interior, namely (see [27])

$$\nu = \min\{\|p\|_1 : \bar{x}(N) = \mathcal{X}p,\ p \in \mathbb{R}^N\} < 1.$$

The function $\nu(\bar{x}(N))$ is a polyhedral norm whose unit ball is the convex hull of $\pm\mathcal{X}$. Such a norm "compares" the size of $\bar{x}(N)$ with the trajectory: it is the minimum scaling factor for the convex hull to have $\bar{x}(N)$ still inside. $\nu = 0$ implies $\bar{x}(N) = 0$, the (practically impossible) exact target.

Remark 1: Assumption 1 implies that the nominal trajectory spans the whole state space, namely that $\mathcal{X}$ has rank n, so $N \geq n$. This rank condition is in practice always satisfied. Indeed, the opposite situation in which $\mathcal{X}$ has not this property, namely that all the trajectory vectors are on a proper subspace, is mathematically possible, but in practice unlikely. This resembles the persistent excitation condition in [11], which is also a rank condition, yet for a different matrix, whose entries include both state and control components.

Note that the following equations hold

$$A\mathcal{X} + B\mathcal{U} = \mathcal{X}\mathcal{P} \tag{3}$$

$$\mathcal{P} = \begin{bmatrix} 0 & 0 & \cdots & 0 & p_1 \\ 1 & 0 & \cdots & 0 & p_2 \\ 0 & 1 & \cdots & 0 & \vdots \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & p_N \end{bmatrix} \tag{4}$$

where p_k are the components of vector p in Assumption 1. Based on this setup, we propose a linear dynamic and a nonlinear (but piecewise linear) static controllers.

[1] Including $\bar{x}(N) = 0$ which would be unlikely.

A. Linear Dynamic Solution

Take a $(N-n) \times N$ matrix

$$\mathcal{Z} = [0 \; z(1) \; z(2) \; \cdots \; z(N-1)]$$

such that

$$M = \begin{bmatrix} \mathcal{X} \\ \mathcal{Z} \end{bmatrix} = \begin{bmatrix} \bar{x}(0) & \bar{x}(1) & \bar{x}(2) & \cdots & \bar{x}(N-1) \\ 0 & z(1) & z(2) & \cdots & z(N-1) \end{bmatrix} \quad (5)$$

is invertible ($\mathcal{Z}$ exists since $\mathcal{X}$ has rank n, see Remark 1).

Consider the dynamic compensator

$$z(k+1) = Rx(k) + Qz(k) \quad (6)$$
$$u(k) = Tx(k) + Sz(k) \quad (7)$$

where $Q \in \mathbb{R}^{(N-n)\times(N-n)}$, $R \in \mathbb{R}^{(N-n)\times n}$, $S \in \mathbb{R}^{m\times(N-n)}$, and $T \in \mathbb{R}^{m\times n}$ are the solution of the linear equation (which exists for invertible M as in (5))

$$\begin{bmatrix} T & S \\ R & Q \end{bmatrix} \begin{bmatrix} \mathcal{X} \\ \mathcal{Z} \end{bmatrix} = \begin{bmatrix} \mathcal{U} \\ \mathcal{Z}\mathcal{P} \end{bmatrix}. \quad (8)$$

Proposition 1: The control (6)–(7) along with (8) is stabilizing. For $z(0) = 0$ and $x(0) = \alpha\bar{x}(0)$ it produces a trajectory α-proportional to the expert trajectory

$$x(k) = \alpha\bar{x}(k).$$

Proof: We immediately see that the closed-loop matrix is similar to matrix $\mathcal{P}$

$$\begin{bmatrix} A+BT & BS \\ R & Q \end{bmatrix} \begin{bmatrix} \mathcal{X} \\ \mathcal{Z} \end{bmatrix} = \begin{bmatrix} \mathcal{X} \\ \mathcal{Z} \end{bmatrix} \mathcal{P}, \quad (9)$$

hence the closed-loop is stable because $\mathcal{P}$ is a Frobenius matrix that is Schur because the sum of the absolute values of the coefficients in the last column is less than 1.

Moreover, due to the structure of (4), from (9) we get

$$\begin{bmatrix} x(k+1) \\ z(k+1) \end{bmatrix} = \begin{bmatrix} A+BT & BS \\ R & Q \end{bmatrix} \begin{bmatrix} x(k) \\ z(k) \end{bmatrix}$$

for $k = 0, 1, \ldots N-1$. Linearity and the fact that $z(0) = 0$ ensures α-proportionality. ■

Remark 2: In [21] it is shown that the coefficients of $\mathcal{P}$ can be assigned. Here it is not possible, since we do not have the model. Therefore, we must rely on the fact that they are small (operator skill).

In the case of the dynamic state feedback controller, it is advisable that, besides stabilizing, the compensator is itself stable (strong stabilization). A stabilizing compensator can be achieved by suitably fixing Q and R, and computing $\mathcal{Z}$ following the procedure proposed in [20]:

1) take a Schur Q;
2) generate R and $\mathcal{Z} = [0 \; \tilde{\mathcal{Z}}]$ solving the linear equation

$$R\mathcal{X} + Q[0 \; \tilde{\mathcal{Z}}] = [0 \; \tilde{\mathcal{Z}}]\mathcal{P}; \quad (10)$$

3) compute T and S from the equation

$$[T \; S] \begin{bmatrix} \mathcal{X} \\ \mathcal{Z} \end{bmatrix} = \mathcal{U},$$

after checking that the resulting square matrix is invertible.

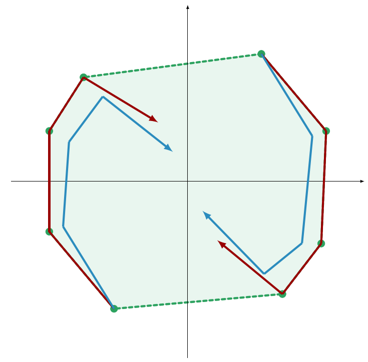

Fig. 2. The nominal trajectory in blue, its $1/\lambda^k$ expansion in red. The vertices of the set $\mathcal{B}_\lambda$ are represented in green.

Equation (10) is of the Sylvester type, with the difference that the first column of $\mathcal{Z} = [0 \; \tilde{\mathcal{Z}}]$ is fixed to be 0. This constraint is compensated by the additional degree of freedom introduced by R. This matrix equation corresponds to $(N-n) \times N$ scalar equations, where the number of unknowns is $(N-n) \times (N-1+n)$, i.e., $(N-n) \times (N-1)$ in $\mathcal{Z}$, and $(N-n) \times n$ in R. The overall number of unknowns exceeds the total number of equations, hence a solution is presumed to exist, with no guarantee, but this can be easily verified. To avoid numerical issues, it is advisable to choose Q such that its eigenvalues are different from those of $\mathcal{P}$.

Remark 3: The suggested procedure ensures closed-loop and, possibly, compensator stability. However, we still have a margin of improvement by repeating the design with different $\mathcal{Z}/Q$ in order to improve performances, in particular when the system starts from non-nominal initial conditions.

B. Static Piecewise Linear Control

In this section, we adopt the Gutman and Cwickel control proposed in [28] as a feedback function. To this aim, we need to modify the solution earlier adopted in [22], where, to enforce stability, some assumptions on the cost function were introduced. Here we do not have cost functions hence we have to revise the theory.

We consider again the setup described above, and we perturb each state and control in the $\mathcal{X}$ and $\mathcal{U}$ trajectories:

$$x_\lambda(k) \doteq \lambda^{-k}\bar{x}(k), \qquad u_\lambda(k) \doteq \lambda^{-k}\bar{u}(k),$$

thus resulting in the following modified trajectories:

$$\tilde{\mathcal{X}} \doteq \mathcal{X}\Lambda \quad \tilde{\mathcal{U}} \doteq \mathcal{U}\Lambda$$

where $\Lambda \doteq \mathrm{diag}\{1, \lambda^{-1}, \lambda^{-2}, \ldots, \lambda^{-N+1}\}$.

Assume that $\lambda < 1$ is chosen close enough to 1 such that

$$x_\lambda(N) \doteq \lambda^{-N}\bar{x}(N) \in \mathrm{conv}\left[\tilde{\mathcal{X}}, -\tilde{\mathcal{X}}\right] \doteq \mathcal{B}_\lambda, \quad (11)$$

where $\mathcal{B}_\lambda = \mathrm{conv}[\tilde{\mathcal{X}}, \; -\tilde{\mathcal{X}}]$ is the convex hull of all the column vectors of $\tilde{\mathcal{X}}$ and their opposite $-\tilde{\mathcal{X}}$. Note that for $\lambda = 1$ inclusion (11) holds by Assumption 1 and since $\bar{x}(N)$ is in the interior, by continuity the inclusion remains satisfied for $\lambda < 1$ sufficiently close to 1. A graphical representation is reported in Figure 2. We can state the following theorem, which states that the set $\mathcal{B}_\lambda$ is λ-contractive for the discrete-time system $x(k+1) = Ax(k) + Bu(k)$.

Theorem 1: Let $\nu_\lambda(x)$ be the norm having $\mathcal{B}_\lambda$ as unit ball. There exists a feedback control $u(k) = \Phi(x(k))$ such that:

$$\nu_\lambda(x(k+1)) = \nu_\lambda(Ax(k) + B\Phi(x(k))) \leq \lambda\nu_\lambda(x(k))$$

Proof: We show the equivalent statement that $\mathcal{B}_\lambda$ is controlled invariant for the modified system

$$x_\lambda(k+1) = \frac{A}{\lambda}x_\lambda(k) + \frac{B}{\lambda}u_\lambda(k). \tag{12}$$

This follows by construction. Take any vertex $\hat{x}$ of $\mathcal{B}_\lambda$. Since $\mathcal{B}_\lambda$ is defined as the convex hull of the points $x_\lambda(k)$, we necessarily have that $\hat{x} = x_\lambda(k)$ for some k. We have

$$\begin{aligned}&\frac{A}{\lambda}x_\lambda(k) + \frac{B}{\lambda}u_\lambda(k) = \frac{A}{\lambda}\lambda^{-k}\bar{x}(k) + \frac{B}{\lambda}\lambda^{-k}\bar{u}(k)\\ &= \lambda^{-(k+1)}[A\bar{x}(k) + B\bar{u}(k)] = \lambda^{-(k+1)}\bar{x}(k+1) \quad \in \mathcal{B}_\lambda.\end{aligned}$$

The inclusion holds by construction, for $k = 0, 1, \dots N-1$, and for $k = N$ because we assumed (11).

Therefore we see that for any $\hat{x}$, vertex of $\mathcal{B}_\lambda$, there is a control that drives the state of (12) inside $\mathcal{B}_\lambda$. This condition is necessary and sufficient for $\mathcal{B}_\lambda$ to be controlled invariant for (12) [27], [28]. In turn, $\mathcal{B}_\lambda$ is invariant for the scaled system (12) if and only if it is λ–contractive for the original system (1). ∎

Now we have the issue of computing the control $u(k) = \Phi(x(k))$. The standard technique for solving

$$\Phi(x) = \arg\min_u \nu_\lambda(Ax + Bu) \tag{13}$$

(yielding $\nu_\lambda(Ax + B\Phi(x)) \leq \lambda\nu_\lambda(x)$) is not viable, because we do not have access to A and B. However, since $\mathcal{B}_\lambda$ has a non–empty interior, we can use the Gutman and Cwikel control [27], [28]. Indeed, [28] provides necessary and sufficient conditions for a control sequence, belonging to a polyhedral constraint set, to stabilize a discrete-time LTI system whose state is also limited by a set of polyhedral constraints. A variable structure linear state feedback controller is presented, given in terms of the controls at the vertices of the polyhedral state constraint set.

The set $\mathcal{B}_\lambda$ is partitioned into simplices $\mathcal{S}_k$ such that:

- the $n+1$ vertices of each simplex $\mathcal{S}_k$ are the origin and n vertices of $\mathcal{B}_\lambda$;
- each k-th simplex $\mathcal{S}_k$ has non-empty interior;
- their union is $\mathcal{B}_\lambda$, i.e., $\bigcup \mathcal{S}_k = \mathcal{B}_\lambda$;
- the pairwise $[\mathcal{S}_h \bigcap \mathcal{S}_k]$, have empty interior $\forall h \neq k$.

Then, the piecewise affine control is computed as follows

$$u = F_i x, \quad \text{for} \quad x \in \mathcal{S}_i, \tag{14}$$

where F_i is the gain defined as in [27], [28]:

$$F_i = G_i D_i^{-1}$$

with D_i square invertible matrix of the nonzero vertices of $\mathcal{S}_i$, and G_i the matrix that groups the corresponding control values.

Remark 4: It might happen that the convex hull $\mathcal{B}_\lambda$ is diamond-affine, i.e., formed by n vertices and their opposite. In this case, let us call D the square matrix such that the vertices are $[D \;\; -D]$. The corresponding controls are pairwise equal. In this case, all the gains are equal, i.e., $F_i = F$, for all i. Indeed, any vertex is achieved by taking a column of D or its opposite. The corresponding controls are those associated to D with the same sign pattern. The property follows immediately from the fact that GD^{-1} is invariant if we change the sign to a column of D and the corresponding column of G.

Remark 5: We finally note that norm $\nu_\lambda(x)$ having as unit ball the polytope $\mathcal{B}_\lambda$ is a Lyapunov function: $\nu_\lambda(x(k)) \leq \nu_\lambda(x(0))$. Since for any other norm $\|x\|$ there exist positive C_1 and C_2 such that $C_1\|x\| \leq \nu_\lambda(x) \leq C_2\|x\|$, the control ensures that if $\|x(0)\| \leq a$ then $\|x(k)\| \leq aC_2/C_1\|x(0)\|$.

This type of property holds also for linear control, adopting standard arguments from the linear theory.

C. Dealing With Noise

So far we have considered a noise-free system. Assume now that, due to the presence of a noise $e(k)$, the recorded trajectory is:

$$x(k+1) = Ax(k) + Bu(k) + e(k). \tag{15}$$

Denoting by $\mathcal{E} = [e(0) \; e(1) \; \cdots \; e(N-1)]$ the noise N-step trajectory, we derive the following generalization of (3):

$$A\mathcal{X} + B\mathcal{U} = \mathcal{X}\mathcal{P} + \mathcal{E},$$

where the last column of $\mathcal{P}$ is the vector p as in Assumption 1, that we suppose is still valid.

Again, we modify the trajectory as $\tilde{\mathcal{X}} \doteq \mathcal{X}\Lambda$, $\tilde{\mathcal{U}} \doteq \mathcal{U}\Lambda$, and $\tilde{\mathcal{E}} \doteq \mathcal{E}\Lambda$, to get

$$A\tilde{\mathcal{X}} + B\tilde{\mathcal{U}} = \tilde{\mathcal{X}}\tilde{\mathcal{P}} + \tilde{\mathcal{E}}, \tag{16}$$

where

$$\tilde{\mathcal{P}} \doteq \Lambda^{-1}\mathcal{P}\Lambda = \begin{bmatrix} 0 & 0 & \cdots & 0 & \lambda^{-N+1}p_1 \\ \lambda & 0 & \cdots & 0 & \lambda^{-N+2}p_2 \\ 0 & \lambda & \cdots & 0 & \vdots \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & \cdots & \lambda & p_N \end{bmatrix}.$$

If ν in Assumption 1 is small enough, we have that the induced 1-norm[2] of $\tilde{\mathcal{P}}$ is $\|\tilde{\mathcal{P}}\|_1 \leq \lambda$. Assume that the columns of $\tilde{\mathcal{E}}$ are small in the metric induced by the norm ν_λ

$$\tilde{e}(k-1) = \tilde{\mathcal{X}}q_k, \quad \|q_k\|_1 \leq \mu < 1 - \nu.$$

Let $\mathcal{Q} \doteq [q_1 \; q_2 \; \cdots \; q_N]$ and $\tilde{\mathcal{E}} = \tilde{\mathcal{X}}\mathcal{Q}$ to rewrite (16) as

$$A\tilde{\mathcal{X}} + B\tilde{\mathcal{U}} = \tilde{\mathcal{X}}\Big[\tilde{\mathcal{P}} + \mathcal{Q}\Big]. \tag{17}$$

Then if $\tilde{\lambda} \doteq \lambda + \mu < 1$, the new equation (17), with $\|\tilde{\mathcal{P}} + \mathcal{Q}\|_1 \leq \tilde{\lambda} < 1$ implies contractivity of the unit ball of ν_λ for the linear system, hence closed–loop stability [27].

[2]The induced 1–norm of M is $\max_j \sum_i |M_{ij}|$.

TABLE I
THE COMMAND AND STATE TRAJECTORY

$$\mathcal{U} = \begin{bmatrix} -2.0000 & 1.1201 & 0.1384 & -0.0354 & -0.0091 & 0.0002 & 0.0002 & 0 \end{bmatrix}$$

$$\mathcal{X} = \begin{bmatrix} 0 & -1.4836 & 0.0140 & 0.0923 & 0.0177 & 0.0015 & 0.0006 & 0.0003 \\ 0 & -0.8772 & -0.1238 & 0.0045 & -0.0008 & -0.0019 & -0.0006 & -0.0001 \\ 1.0000 & 0.5397 & 0.0525 & -0.0006 & 0.0027 & 0.0017 & 0.0005 & 0.0001 \end{bmatrix}$$

$$\mathcal{Z} = \begin{bmatrix} 0 & -0.9459 & 0.2780 & -0.9496 & 0.4631 & 0.7567 & 0.9052 & -0.3724 \\ 0 & 0.6216 & -0.2406 & -0.3027 & 0.4843 & -0.7683 & 0.3153 & -0.7873 \\ 0 & -0.3342 & -0.6223 & 0.5375 & 0.3367 & 0.8114 & -0.4719 & 0.8904 \\ 0 & -0.1729 & 0.9615 & 0.1930 & 0.6955 & -0.8601 & -0.5894 & 0.3927 \\ 0 & -0.3502 & 0.8323 & 0.7560 & 0.3040 & 0.2020 & -0.0898 & 0.9076 \end{bmatrix}$$

III. NUMERICAL EXAMPLE

A. The Ship Steering Problem: Dynamic Control

We now consider the problem presented in Section I-A, in which we assume a continuous-time model $\dot{x} = A_c x + B_c u$, where:

$$A_c = \begin{bmatrix} -0.8000 & 0.3000 & 0 \\ 0.3000 & -0.8000 & 0 \\ 0 & 1.0000 & 0 \end{bmatrix} \quad B_c = \begin{bmatrix} 1.0000 \\ 0.5000 \\ 0 \end{bmatrix}.$$

The system is thought sampled with sampling time $T = 1$ min. The resulting discrete-time model is $x(k+1) = A_d x(k) + B_d u(k)$, where:

$$A_d = \begin{bmatrix} 0.4697 & 0.1368 & 0 \\ 0.1368 & 0.4697 & 0 \\ 0.0902 & 0.6967 & 1.0000 \end{bmatrix} \quad B_d = \begin{bmatrix} 0.7418 \\ 0.4386 \\ 0.2302 \end{bmatrix}.$$

The numerical values above defined in both models, including the zeros, are assumed *completely unknown*. We consider the problem of steering the ship from an initial yaw angle to a target assumed as 0, i.e., $\bar{x}(0) = [0\ 0\ 1]^\top$ and $\bar{x}(N) = [0\ 0\ 0]^\top$. The operator-generated commands and the corresponding state trajectory, $\mathcal{U}$ and $\mathcal{X}$ respectively, are reported in Table I.

By applying the theory, we first derive the matrix $\mathcal{Z}$, reported in Table I, and then the dynamic controller. Marices Q, T, R and S are set as follows:

$$Q = \begin{bmatrix} 0.0402 & 0.8692 & 1.9093 & -0.1969 & -1.0209 \\ -0.5615 & -0.6964 & 0.4681 & -0.0665 & -1.2683 \\ 0.1918 & 1.3490 & -0.4107 & -0.4539 & 1.8490 \\ -0.7264 & 0.2622 & -1.1572 & -0.9474 & 1.4770 \\ 0.1183 & 0.7653 & -0.6330 & -0.6938 & 1.6350 \end{bmatrix}$$

$$T = \begin{bmatrix} -0.3009 & -1.9987 & -2.0000 \end{bmatrix}$$

$$S = \begin{bmatrix} 0.0493 & 0.0186 & -0.0726 & -0.0633 & 0.2929 \end{bmatrix}$$

$$R = \begin{bmatrix} 6.4922 & -11.5883 & -0.9459 \\ 5.3794 & -7.9888 & 0.6216 \\ -1.9593 & 4.0747 & -0.3342 \\ -2.6665 & 4.3148 & -0.1729 \\ -1.2931 & 1.1627 & -0.3502 \end{bmatrix}.$$

The spectral radius of the closed-loop system is $\rho(\mathcal{P}) = 0.3483$, while the spectral radius of Q is $\rho(Q) = 0.3268$, thus we achieve strong stabilization. The nominal open-loop transient in discrete time is reported in Figure 3. We do not report the discrete-time closed-loop transient from the nominal initial condition, because the figure exactly matches the one shown in Figure 3.

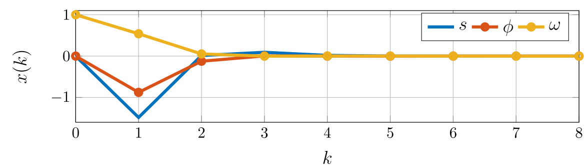

Fig. 3. The nominal open-loop (performed by operator) transient.

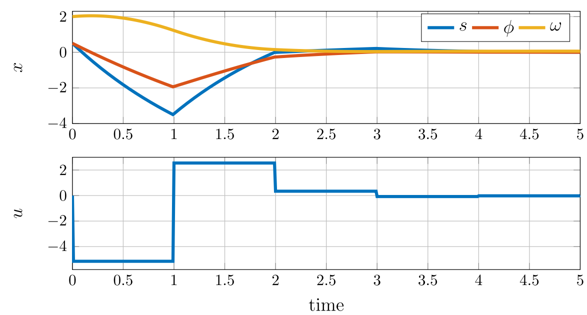

Fig. 4. The feedback transient in continuous time with the piecewise constant input due to the sampled-data implementation starting from a non- nominal initial condition.

We have also performed a simulation, from the non-nominal initial condition $x(0) = \begin{bmatrix} 0.5 & 0.5 & 2.0 \end{bmatrix}^\top$ in continuous-time. Results are reported in Figure 4. The code of the performed experiment can be found in https://github.com/EricaSalvato/Model-free-feedback-control-synthesis-from-expert-demonstration.git

B. The Ship Steering Problem: Static Control

We now report experimental results obtained by computing the static version of the controller. We actually fell in the lucky case considered in Remark 4, if λ is not too small. For instance, if we take $\lambda = 0.9$, it turns out that the convex hull of the modified trajectory is diamond-affine, namely, its vertices are the first three columns of $\mathcal{X}$

$$D = \begin{bmatrix} 0 & -1.6485 & 0.0173 \\ 0 & -0.9746 & -0.1528 \\ 1.0000 & 0.5997 & 0.0648 \end{bmatrix}$$

and their opposite.

We take the corresponding scaled control actions

$$G = \begin{bmatrix} -2.0000 & 1.2446 & 0.1709 \end{bmatrix}.$$

In these cases the overall piecewise linear control is linear or, in other words, all the gains F_i are equal

$$F_i = GD^{-1} = \begin{bmatrix} -0.2997 & -2.0005 & -2.0000 \end{bmatrix}$$

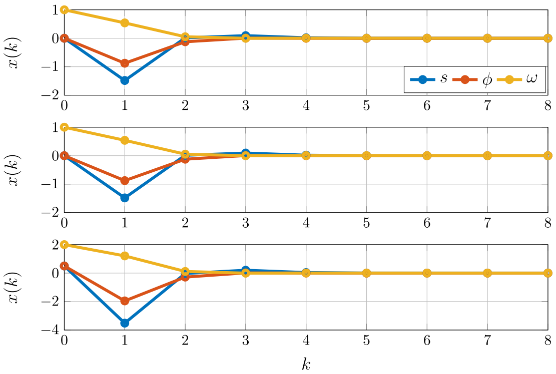

Fig. 5. The feedback transient with the static gain (discrete-time). Top: the operator-induced transient. Middle: the closed-loop transient for the nominal initial condition. Bottom: the closed-loop transient for the non-nominal initial condition.

TABLE II
AVERAGE OF THE MAXIMUM EIGENVALUE AS A FUNCTION OF δ

$\delta = \|e(k)\|_\infty / \|x(0)\|_\infty$	0	10 %	20%	40%
dynamic linear max $\|\lambda_i\|$	0.3483	0.8808	0.9015	0.9089
static linear max $\|\lambda_i\|$	0.3064	0.4719	0.5649	0.7030

In Figure 5 we report the nominal λ-scaled open transient, the closed-loop transient for the nominal initial condition (which is almost identical), and the closed loop transient for a non-nominal initial condition $x(0) = [0.5000\ 0.5000\ 2.0000]^\top$.

To test the procedure under noise, we performed the following experiment. We generated $N_{tr} = 10,000$ trajectories of (15) with e randomly generated with uniformly distributed components $|e_i| \leq \delta \|x(0)\|_\infty$. Then: (i) for each self–absorbing trajectory, a controller was designed (the other trajectories are not suitable, hence rejected); (ii) the average value of the maximum modulus eigenvalue was computed. The results are in Table II for different δ.

IV. Conclusion

We showed that, for discrete-time linear systems, whenever a desired open-loop trajectory brings the final state close enough to the origin from a nominal initial state, a stabilizing feedback able to repeat the same trajectory for the same initial condition can be synthesized without a model.

Two solutions have been proposed: a dynamic linear control (possibly itself stable, besides stabilizing), and a piece-wise linear one. The former has the problem that the compensator order grows with the trajectory length. The latter stabilizing control, has no stability issue, being static, however, its implementation can be difficult due to the large number of symplicial sectors involved.

As an open problem we stress that, although we investigated the noise effect in the trajectory recording, the provided invariant set is not robustly invariant. Indeed it is based on a single realization of the noise, while a robustly invariant set requires a characterization of the worst-case noise. We leave the problem for future investigation.

References

[1] F. Tao, H. Zhang, A. Liu, and A. Y. C. Nee, "Digital twin in industry: State-of-the-art," *IEEE Trans. Ind. Informat.*, vol. 15, no. 4, pp. 2405–2415, Apr. 2019.
[2] A. U. Levin and K. S. Narendra, "Control of nonlinear dynamical systems using neural networks. II. Observability, identification, and control," *IEEE Trans. Neural Netw.*, vol. 7, no. 1, pp. 30–42, Jan. 1996.
[3] M. C. Campi and S. M. Savaresi, "Direct nonlinear control design: The virtual reference feedback tuning (VRFT) approach," *IEEE Trans. Autom. Control*, vol. 51, no. 1, pp. 14–27, Jan. 2006.
[4] C. Novara, L. Fagiano, and M. Milanese, "Direct feedback control design for nonlinear systems," *Automatica*, vol. 49, no. 4, pp. 849–860, 2013.
[5] S. Formentin, C. Novara, S. M. Savaresi, and M. Milanese, "Data-driven inversion-based control of nonlinear systems," *IFAC-PapersOnLine*, vol. 48, no. 28, pp. 1343–1348, 2015.
[6] C. Novara and S. Formentin, "Data-driven inversion-based control of nonlinear systems with guaranteed closed-loop stability," *IEEE Trans. Autom. Control*, vol. 63, no. 4, pp. 1147–1154, Apr. 2018.
[7] G. Baggio, V. Katewa, and F. Pasqualetti, "Data-driven minimum-energy controls for linear systems," *IEEE Contr. Syst. Lett.*, vol. 3, pp. 589–594, 2019.
[8] G. Baggio and F. Pasqualetti, "Learning minimum-energy controls from heterogeneous data," in *Proc. Amer. Control Conf.*, 2020, pp. 3991–3996.
[9] G. Baggio, D. S. Bassett, and F. Pasqualetti, "Data-driven control of complex networks," *Nat. Commun.*, vol. 12, no. 1, p. 1429, 2021.
[10] J. C. Willems, P. Rapisarda, I. Markovsky, and B. L. M. De Moor, "A note on persistency of excitation," *Syst. Control Lett.*, vol. 54, no. 4, pp. 325–329, 2005.
[11] C. De Persis and P. Tesi, "Formulas for data-driven control: Stabilization, optimality, and robustness," *IEEE Trans. Autom. Control*, vol. 65, no. 3, pp. 909–924, Mar. 2020.
[12] M. Rotulo, C. De Persis, and P. Tesi, "Data-driven linear quadratic regulation via semidefinite programming," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 3995–4000, 2020.
[13] C. De Persis and P. Tesi, "Designing experiments for data-driven control of nonlinear systems," 2021, *arXiv:2103.16509*.
[14] C. De Persis and P. Tesi, "Low-complexity learning of linear quadratic regulators from noisy data," *Automatica*, vol. 128, Jun. 2021, Art. no. 109548.
[15] H. Yang and S. Li, "A data-driven predictive controller design based on reduced Hankel matrix," in *Proc. 10th Asian Control Conf.*, 2015, pp. 1–7.
[16] J. Coulson, J. Lygeros, and F. Dörfler, "Data-enabled predictive control: In the shallows of the DeePC," in *Proc. 18th Eur. Control Conf.*, 2019, pp. 307–312.
[17] J. Coulson, J. Lygeros, and F. Dörfler, "Regularized and distributionally robust data-enabled predictive control," in *Proc. IEEE 58th Conf. Decis. Control*, 2019, pp. 2696–2701.
[18] J. Berberich, J. Köhler, M. A. Müller, and F. Allgöwer, "Data-driven model predictive control with stability and robustness guarantees," *IEEE Trans. Autom. Control*, vol. 66, no. 4, pp. 1702–1717, Apr. 2021.
[19] H. Ravichandar, A. S. Polydoros, S. Chernova, and A. Billard, "Recent advances in robot learning from demonstration," *Annu. Rev. Control Robot. Auton. Syst.*, vol. 3, pp. 297–330, May 2020.
[20] F. Blanchini and F. A. Pellegrino, "Relatively optimal control and its linear implementation," *IEEE Trans. Autom. Control*, vol. 48, no. 12, pp. 2151–2162, Dec. 2003.
[21] F. Blanchini and F. A. Pellegrino, "Relatively optimal control with characteristic polynomial assignment and output feedback," *IEEE Trans. Autom. Control*, vol. 51, no. 2, pp. 183–191, Feb. 2006.
[22] F. Blanchini and F. A. Pellegrino, "Relatively optimal control: A static piecewise-affine solution," *SIAM J. Control Optim.*, vol. 46, no. 2, pp. 585–603, 2007.
[23] F. A. Pellegrino, F. Blanchini, G. Fenu, and E. Salvato, "Closed-loop control from data-driven open-loop optimal control trajectories," in *Proc. Eur. Control Conf.*, 2022, pp. 1379–1384.
[24] F. A. Pellegrino, F. Blanchini, G. Fenu, and E. Salvato, "Data-driven dynamic relatively optimal control," in *Proc. Eur. Control Conf.*, 2023.
[25] A. Alessandri, S. Donnarumma, S. Vignolo, M. Figari, and M. Martelli, "System control design of autopilot and speed pilot for a patrol vessel by using LMIs," in *Proc. 16th Int. Congr. Int. Maritime Assoc. Mediterr.*, 2015, pp. 1–7.
[26] F. Blanchini, P. Colaneri, Y. Fujisaki, S. Miani, and F. A. Pellegrino, "A Youla–Kučera parameterization approach to output feedback relatively optimal control," *Syst. Control Lett.*, vol. 81, pp. 14–23, Jul. 2015.
[27] F. Blanchini and S. Miani, *Set-Theoretic Methods in Control* (Systems & Control: Foundations & Applications). Boston, MA, USA: Birkhäuser, 2015.
[28] P.-O. Gutman and M. Cwikel, "Admissible sets and feedback control for discrete-time linear dynamical systems with bounded controls and states," *IEEE Trans. Autom. Control*, vol. 31, no. 4, pp. 373–376, Apr. 1986.

Open Access funding provided by 'Università degli Studi di Trieste' within the CRUI CARE Agreement