SIAM J. CONTROL OPTIM.
Vol. 59, No. 6, pp. 4511–4537

ANALYSIS OF A GREEDY RECONSTRUCTION ALGORITHM*

S. BUCHWALD†, G. CIARAMELLA‡, AND J. SALOMON§

Abstract. A novel and detailed convergence analysis is presented for a greedy algorithm that was introduced in [Y. Maday and J. Salomon, *Joint Proceedings of the* 48*th IEEE Conference on Decision and Control and the* 28*th Chinese Control Conference*, 2009, pp. 375–379] for operator reconstruction problems in the field of quantum mechanics. This algorithm is based on an offline/online decomposition of the reconstruction process and on an ansatz for the unknown operator obtained by an a priori chosen set of linearly independent matrices. The presented convergence analysis focuses on linear-quadratic (optimization) problems governed by linear differential systems and reveals the strong dependence of the performance of the greedy algorithm on the observability properties of the system and on the ansatz of the basis elements. Moreover, the analysis allows us to use a precise (and in some sense optimal) choice of basis elements for the linear case and led to the introduction of a new and more robust optimized greedy reconstruction algorithm. This optimized approach also applies to nonlinear Hamiltonian reconstruction problems, and its efficiency is demonstrated by numerical experiments.

Key words. Hamiltonian identification, operator reconstruction, quantum control problems, inverse problems, greedy reconstruction algorithm

AMS subject classifications. 65K10, 81Q93, 34A55, 49N10, 49N45

DOI. 10.1137/20M1373384

1. Introduction.

The identification of Hamiltonian operators plays a fundamental role in the fields of quantum physics and quantum chemistry; see, e.g., [7, 9, 10, 11, 19, 21, 22, 23, 24, 25] and references therein. Even though the overall literature about Hamiltonian identification problems is quite extensive, the mathematical contribution to this area is rather limited. Important mathematical theoretical contributions can be found in [2, 4] and in [8, 13], where uniqueness results for quantum inverse problems are proved by exploiting controllability arguments. Other techniques, based on the so-called Carleman's estimate, are used in [2] to deduce uniqueness results for inverse problems governed by Schrödinger-type equations in the presence of discontinuous coefficients. Excluding these few theoretical results, the literature rather focuses on numerical algorithms.

The term Hamiltonian identification often refers to two distinct problems. On the one hand, it sometimes indicates the inverse problem associated with the identification of a Hamiltonian operator obtained by a numerical fitting of simulated and given experimental data. On the other hand, it occasionally refers to both the problem of designing experimental parameters (allowing an optimized production of experimental data) and the subsequent inverse identification problem. In general, the design of experimental parameters includes the computation of control functions allowing an efficient numerical solving of the inverse problem.

In the latter problem, the algorithms proposed in the literature often combine the computation of control functions with the production of new synthetic (simulated)

*Received by the editors October 13, 2020; accepted for publication (in revised form) June 11, 2021; published electronically November 29, 2021.
https://doi.org/10.1137/20M1373384
Funding: The work of the first author was partially supported by the DFG via the collaborative research center SFB1432, project 425217212.
†Universität Konstanz, Konstanz, Germany (simon.buchwald@uni-konstanz.de).
‡MOX, Politecnico di Milano, Milano, Italy (gabriele.ciaramella@polimi.it).
§INRIA Paris, France (julien.salomon@inria.fr).

data or experimental data. Mathematically, this framework has given rise to two different approaches. The first one [13] consists in a procedure that alternately updates a (shrinking) set of admissible Hamiltonian operators and the trial control field used to generate new data. The second approach [14] is based on a full offline/online decomposition and is inspired by the greedy strategy emerged in the field of approximation theory in the 2000s; see, e.g., [1] and references therein. Even though some mathematical investigations of the first approach can be found in the literature (see [8, 13]), much less is known about the second strategy, for which only preliminary numerical results were presented in [14].

The goal of the present work is to provide a first detailed convergence analysis of the Hamiltonian reconstruction strategy defined in [14]. As a byproduct, this analysis allows us to introduce a new more efficient and robust numerical reconstruction algorithm.

The numerical strategy presented in [14] is based on the ansatz that the unknown operator can be written as a linear combination of a priori given linearly independent matrices. The set of these matrices is denoted by $\mathcal{B}_\mu$. The reconstruction process is then decomposed in an offline phase and an online phase. In the offline phase, a family of control functions is built iteratively in a greedy manner in order to maximize the distinguishability of the system. This phase exploits only the quantum model without any use of laboratory information. The algorithm proposed in [14] for the offline phase, which we call in this paper the greedy reconstruction (GR) algorithm, consists of a sweep over the elements of $\mathcal{B}_\mu$. At every iteration of the GR algorithm, one new element of $\mathcal{B}_\mu$ is considered and a new control function is computed with the goal of splitting the states generated by the new element and the ones already considered in the previous iterations. The computed control functions are experimentally implemented in the online phase to produce laboratory data. These are in turn used to define and solve an identification inverse problem, aiming at fitting the numerical simulations with the corresponding experimental data.

In [14] the heuristic motivation for the offline phase is that this attempts to produce a set of control functions that make the online identification problem uniquely solvable (and easier to be solved) in a neighborhood of the true solution. Starting from this idea we develop a detailed convergence analysis for linear problems (linear-quadratic in the least-squares sense). The analysis of the algorithm for linear problems corresponds to a local analysis performed on linearized equations and provides a first fundamental step toward the study of full nonlinear problems. Our analysis relates very clearly the iterations of the offline phase, and the corresponding computed control functions, to the solvability of the online identification problem. Moreover, the obtained theoretical results will reveal the strong dependence of the performance of the GR algorithm on the observability properties of the system and on the ansatz of the basis elements used to reconstruct the unknown operator. These observations allow us to improve the GR algorithm and introduce a new optimized greedy reconstruction (OGR) algorithm which shows a very robust behavior not only for the linear-quadratic reconstruction problems but also for nonlinear Hamiltonian reconstruction problems.

The paper is organized as follows. In section 2, the notation used throughout this paper is fixed. Section 3 describes the Hamiltonian reconstruction problem and the original GR algorithm introduced in [14]. The GR algorithm is then adapted to linear-quadratic problems in section 4, and the corresponding convergence analysis is presented in section 5. In section 6, we introduce some improvements of the GR algorithm that lead to an OGR algorithm. The OGR algorithm is presented first for linear-quadratic problems and then extended to nonlinear Hamiltonian reconstruction

problems. Within section 6, results of numerical experiments are shown to demonstrate the efficiency and the improved robustness of the new proposed algorithm. Finally, we present our conclusions in section 7.

2. Notation. Consider a positive natural number N. We denote by $\langle \mathbf{v}, \mathbf{w} \rangle := \overline{\mathbf{v}}^\top \mathbf{w}$, for any $\mathbf{v}, \mathbf{w} \in \mathbb{C}^N$ the usual complex scalar product on $\mathbb{C}^N$, and by $\|\cdot\|_2$ the corresponding norm. Further, $|z|$ is the modulus of a complex number z and i is the imaginary unit. The space of Hermitian matrices in $\mathbb{C}^{N\times N}$ is denoted by $\mathrm{Her}(N)$.[1] For any $A \in \mathbb{C}^{N\times N}$, $[A]_{j,k}$ denotes the j,k (with $j,k \le N$) entry of A, and the notation $A_{[1:k,1:j]}$ indicates the upper left submatrix of A of size $k \times j$, namely, $[A_{[1:k,1:j]}]_{\ell,m} := [A]_{\ell,m}$ for $\ell = 1,\dots,k$ and $m = 1,\dots,j$. Similarly, $A_{[1:k,j]}$ denotes the column vector in $\mathbb{C}^k$ corresponding to the first k elements of the column j of A, namely, $[A_{[1:k,j]}]_\ell := [A]_{\ell,j}$ for $\ell = 1,\dots,k$. Finally, the usual inner product of $L^2(0,T;\mathbb{C}^N)$ is denoted by $\langle \cdot,\cdot \rangle_{L^2}$, and $L^2 := L^2(0,T;\mathbb{R})$.

3. Hamiltonian reconstruction and a GR algorithm. Consider the finite-dimensional Schrödinger equation

$$i\dot{\boldsymbol{\psi}}(t) = [H + \epsilon(t)\mu_\star]\boldsymbol{\psi}(t),\ t \in (0,T], \quad \boldsymbol{\psi}(0) = \boldsymbol{\psi}_0, \tag{3.1}$$

governing the time evolution of the state of a quantum system $\boldsymbol{\psi} \in \mathbb{C}^N$, $N \in \mathbb{N}^+$. The internal Hamiltonian H is assumed to be known, and the goal is to identify the unknown dipole moment operator $\mu_\star$ that couples the quantum system to a time-dependent external laser field $\epsilon \in L^2$, which acts as a control function on the system. Both internal Hamiltonian H and dipole operator $\mu_\star$ belong to $\mathrm{Her}(N)$, and $\boldsymbol{\psi}(t)$ lies in $\mathbb{C}^N$. The initial condition is $\boldsymbol{\psi}_0 \in \mathbb{C}^N$ which satisfies $\|\boldsymbol{\psi}_0\|_2 = 1$.

The true dipole operator $\mu_\star$ is unknown and assumed to lie in a space spanned by K linearly independent matrices $\mu_1,\dots,\mu_K$, forming the set $\mathcal{B}_\mu = (\mu_j)_{j=1}^K \subset \mathrm{Her}(N)$, where $K \in \mathbb{N}^+$ satisfies $1 \le K \le \dim \mathrm{Her}(N) = N^2$. Hence, we write $\mu_\star = \mu(\boldsymbol{\alpha}_\star)$ with $\mu(\boldsymbol{\alpha}) := \sum_{j=1}^K \boldsymbol{\alpha}_j \mu_j$ for any $\boldsymbol{\alpha} \in \mathbb{R}^K$.

To identify the true operator $\mu_\star$ one uses a set of control fields $(\epsilon^m)_{m=1}^K \subset L^2$ to perform K laboratory experiments and obtain the experimental data

$$\varphi(\mu_\star, \epsilon^m) := \langle \boldsymbol{\psi}_1, \boldsymbol{\psi}_T(\mu_\star, \epsilon^m) \rangle \text{ for } m = 1,\dots,K. \tag{3.2}$$

Here, $\boldsymbol{\psi}_T(\mu_\star, \epsilon)$ denotes the solution to (3.1) at time $T > 0$, corresponding to the dipole operator $\mu_\star$ and a laser field ϵ. The value $\boldsymbol{\psi}_1 \in \mathbb{C}^N$ is a fixed state with $\|\boldsymbol{\psi}_1\|_2 = 1$ and acts on a state of the quantum system as an observer operator. The measurements are assumed not to be affected by any type of noise.

Using the set of control fields $(\epsilon^m)_{m=1}^K$ and the corresponding experimental data $(\varphi(\mu_\star, \epsilon^m))_{m=1}^K \subset \mathbb{C}$, one solves the nonlinear least-squares problem

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^K} \sum_{m=1}^K |\varphi(\mu_\star, \epsilon^m) - \varphi(\mu(\boldsymbol{\alpha}), \epsilon^m)|^2, \tag{3.3}$$

where $\varphi(\mu(\boldsymbol{\alpha}), \epsilon^m) := \langle \boldsymbol{\psi}_1, \boldsymbol{\psi}_T(\mu(\boldsymbol{\alpha}), \epsilon^m) \rangle$, with $\boldsymbol{\psi}_T(\mu(\boldsymbol{\alpha}), \epsilon^m)$ the solution to (3.1) evaluated at time T corresponding to the dipole operator $\mu(\boldsymbol{\alpha})$ and the laser field ϵ^m. Clearly $\boldsymbol{\alpha}_\star$ is a global solution to (3.3).

[1] Notice that the set of Hermitian matrices forms a (real) vector space if the scalar multiplication is defined with respect to scalars belonging to $\mathbb{R}$. In fact, if $A \in \mathrm{Her}(N)$, then $cA \in \mathrm{Her}(N)$ for any $c \in \mathbb{R}$. However, this is not true for $c \in \mathbb{C}$, since choosing, e.g., $c = i$, the imaginary unit, the transpose conjugate of iA is $-iA$.

In the presented reconstruction problem, several variables are used. Let us clarify their roles in plain words:

- The elements of the basis $\mathcal{B}_\mu$ can be arbitrarily chosen as data.
- Given a basis $\mathcal{B}_\mu$, the true unknown of the problem is $\boldsymbol{\alpha}_\star$ (or equivalently $\mu_\star$).
- The control functions are needed to produce the laboratory data (3.2), which are necessary to assemble the (final) inverse problem (3.3). These control functions are computed (optimized) by the numerical strategy discussed below with the goal of optimizing the conditioning of problem (3.3).

If the control functions $(\epsilon^m)_{m=1}^K$ and the data $(\varphi(\mu_\star, \epsilon^m))_{m=1}^K$ are given, problem (3.3) is a standard parameter-identification inverse problem written in a minimization form. The choice of the laser fields $(\epsilon^m)_{m=1}^K$ can affect significantly the properties of (3.3) and the corresponding solutions. To design an optimized set of control functions, in particular with the goal of improving local convexity properties of (3.3), Maday and Salomon introduced in [14] a numerical strategy which separates the reconstruction process of $\mu_\star$ in offline and online phases. In the offline phase, a GR algorithm computes a set of optimized laser fields $(\epsilon^m)_{m=1}^K$ by exploiting only the quantum model (3.3) and without using any laboratory data. In the online phase, the computed control fields $(\epsilon^m)_{m=1}^K$ are used experimentally to produce the laboratory data $\varphi(\mu_\star, \epsilon^m) := \langle \boldsymbol{\psi}_1, \boldsymbol{\psi}_T(\mu_\star, \epsilon^m) \rangle$ and to define the nonlinear problem (3.3).

While the online phase consists (mathematically) in solving a classical parameter-identification inverse problem, the offline phase requires the GR algorithm introduced in [14]. The ideal goal of this offline/online framework is to find a good approximation of the unknown operator for which the difference at time T between observed experimental data and numerically computed data is the smallest for any control. In other words, one aims at finding a matrix μ that solves

$$\min_{\mu \in \operatorname{span} \mathcal{B}_\mu} \sup_{\epsilon \in L^2} |\varphi(\mu_\star, \epsilon) - \varphi(\mu, \epsilon)|^2, \tag{3.4}$$

or equivalently an $\boldsymbol{\alpha}$ that solves

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^K} \sup_{\epsilon \in L^2} |\varphi(\mu(\boldsymbol{\alpha}_\star), \epsilon) - \varphi(\mu(\boldsymbol{\alpha}), \epsilon)|^2. \tag{3.5}$$

Therefore, the goal of the GR algorithm is to generate a set of K control functions such that a computed solution to (3.3) is also a solution to (3.4)–(3.5). To do so, the heuristic argument used in [14] is that the GR algorithm must attempt to distinguish numerical data for any two $\mu(\widetilde{\boldsymbol{\alpha}}), \mu(\widehat{\boldsymbol{\alpha}}) \in \operatorname{span} \mathcal{B}_\mu$, $\mu(\widetilde{\boldsymbol{\alpha}}) \neq \mu(\widehat{\boldsymbol{\alpha}})$ without performing any laboratory experiment. Following this idea, Maday and Salomon defined the GR algorithm as an iterative procedure that performs a sweep over the linearly independent matrices $(\mu_k)_{k=1}^K$ and computes a new control field ϵ^{k+1} at each iteration. Suppose that the control fields $\epsilon^1, \dots, \epsilon^k$ are already computed; the new control function ϵ^{k+1} is obtained by two substeps: one first solves the identification problem

$$\min_{\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_k} \sum_{m=1}^{k} \left| \varphi\left(\sum_{j=1}^{k} \boldsymbol{\alpha}_j \mu_j, \epsilon^m \right) - \varphi(\mu_{k+1}, \epsilon^m) \right|^2, \tag{3.6}$$

which gives the coefficients $\boldsymbol{\alpha}_1^k, \dots, \boldsymbol{\alpha}_k^k$, and then computes the new field as

$$\epsilon^{k+1} \in \operatorname{argmax}_{\epsilon \in L^2} \left| \varphi(\mu_{k+1}, \epsilon) - \varphi\left(\sum_{j=1}^{k} \boldsymbol{\alpha}_j^k \mu_j, \epsilon \right) \right|^2. \tag{3.7}$$

Algorithm 3.1. Greedy Reconstruction Algorithm

Require: A set of K linearly independent matrices $\mathcal{B}_\mu = (\mu_1, \dots, \mu_K)$.

1: Solve the initialization problem

$$\max_{\boldsymbol{\epsilon} \in L^2} |\varphi(\mu_1, \epsilon) - \varphi(0,0)|^2, \tag{3.8}$$

which gives the field ϵ^1, and set $k = 1$.

2: **while** $k \leq K - 1$ **do**

3: Fitting step: Find $(\boldsymbol{\alpha}_j^k)_{j=1,\dots,k}$ that solve the problem

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^k} \sum_{m=1}^{k} |\varphi(\mu_{k+1}, \epsilon^m) - \varphi(\mu^{(k)}(\boldsymbol{\alpha}), \epsilon^m)|^2. \tag{3.9}$$

4: Discriminatory step: Find ϵ^{k+1} that solves the problem

$$\max_{\epsilon \in L^2} |\varphi(\mu_{k+1}, \epsilon) - \varphi(\mu^{(k)}(\boldsymbol{\alpha}^k), \epsilon)|^2. \tag{3.10}$$

5: Update $k \leftarrow k + 1$.

6: **end while**

The step of solving Problem (3.6) is called *fitting step*, since one attempts to compute a vector $\boldsymbol{\alpha}^k := [\boldsymbol{\alpha}_1^k, \dots, \boldsymbol{\alpha}_k^k]^\top$ that fits the quantities $\varphi(\sum_{j=1}^k \boldsymbol{\alpha}_j^k \mu_j, \epsilon^m)$ and $\varphi(\mu_{k+1}, \epsilon^m)$. In other words, the new basis element μ_{k+1} is considered, and one identifies an element $\mu^{(k)}(\boldsymbol{\alpha}^k) := \sum_{j=1}^k \boldsymbol{\alpha}_j^k \mu_j$ such that none of the already computed control functions $\epsilon^1, \dots, \epsilon^k$ is capable of distinguishing the observations $\varphi(\mu^{(k)}(\boldsymbol{\alpha}^k), \epsilon)$ and $\varphi(\mu_{k+1}, \epsilon)$ (namely, $\varphi(\mu^{(k)}(\boldsymbol{\alpha}^k), \epsilon^m) \neq \varphi(\mu_{k+1}, \epsilon^m)$ for $m = 1, \dots, k$). The step of solving problem (3.7) is called a *discriminatory step* because one computes a control function ϵ^{k+1} that is capable of distinguishing (discriminating) $\varphi(\mu^{(k)}(\boldsymbol{\alpha}^k), \epsilon^{k+1})$ from $\varphi(\mu_{k+1}, \epsilon^{k+1})$.

The full GR algorithm is stated in Algorithm 3.1.[2] Notice how the algorithm is obtained by a sequence of minimization and maximization problems, mimicking exactly the structure of the min-max problem (3.4)–(3.5).

Notice also that, since the goal of the GR algorithm is to compute control functions that allow one to distinguish between the states of the system corresponding to any possible dipole matrix, the algorithm implicitly attempts to compute control functions that make the online identification problem (3.3) locally strictly convex (hence uniquely solvable). This is an important observation that we will use to begin our convergence analysis.

A general analysis of the GR algorithm in a full nonlinear setting is a very complicated task. As a first step in this direction, we propose in the next section to focus on a linear model. On the one hand, this choice allows us to provide a first detailed analysis of the algorithm. On the other hand, this study corresponds to a local analysis performed on linearized models. Note that linearizing (3.1) around $\boldsymbol{\epsilon} = 0$ gives

$$i\delta\dot{\boldsymbol{\psi}}(t) = H\delta\boldsymbol{\psi}(t) + [\delta\epsilon(t)\mu_\star]\boldsymbol{\psi}(t),\ t \in (0,T], \quad \delta\boldsymbol{\psi}(0) = 0, \tag{3.11}$$

[2]Notice that the initialization problem (3.8) is different from the one considered in [14], which was stated anyway to be arbitrary. The reason for our choice is that (as we will see in the next sections) this slightly modified initialization problem (3.8) will be essential to obtain convergence.

where ψ is a solution of (3.1). Focusing on the case where $\boldsymbol{\psi}(0)$ is an eigenvector of H, i.e., $H\boldsymbol{\psi}(0) = \lambda\boldsymbol{\psi}(0)$, we obtain $\boldsymbol{\psi}(t) = e^{-i\lambda t}\boldsymbol{\psi}(0)$ so that the control term reads as $[\delta\epsilon(t)\mu_\star]\boldsymbol{\psi}(t) = [\lambda\mu_\star\boldsymbol{\psi}(0)]e^{-i\lambda t}\delta\epsilon(t)$. It follows that this framework corresponds to a linear model of the form $\dot{\boldsymbol{y}}(t) = A\boldsymbol{y}(t) + B\boldsymbol{\epsilon}(t)$ (as (4.1) in section 4) with $\boldsymbol{y} = \delta\boldsymbol{\psi}$, $A = H$, $B = \lambda\mu_\star\boldsymbol{\psi}(0)$, $\boldsymbol{y}_0 = 0$, and $\boldsymbol{\epsilon}(t) = e^{-i\lambda t}\delta\epsilon(t)$. Let us also remark that this setting is often used to study theoretically the controllability of Schrödinger-type equations; see, e.g., [3], and the references therein. Furthermore, we wish to remark that it is always possible to rewrite a system of complex differential equations (like (3.11) and (3.1)) into a real (but augmented) system by separating real and imaginary components. For this reason, the analysis presented in section 4 focuses, without loss of generality, on systems of real differential equations.

We conclude this section with a final remark about the laboratory measurements. Throughout this paper, these are assumed to be not affected by any type of noise, even though noise is a significant factor that has to be dealt with; see [13, Remark 1] and references therein. However, the main goal of the present work is the numerical and convergence analysis of the computational framework and the GR algorithm introduced in [14], where noisy effects in taking measurements are also neglected.

4. Linear-quadratic reconstruction problems. Consider a state $\boldsymbol{y}$ whose time evolution is governed by the (real) ordinary differential equation

$$\dot{\boldsymbol{y}}(t) = A\boldsymbol{y}(t) + B_\star\boldsymbol{\epsilon}(t),\ t \in (0,T], \quad \boldsymbol{y}(0) = \boldsymbol{y}_0, \tag{4.1}$$

where $A \in \mathbb{R}^{N\times N}$ is a given matrix for $N \in \mathbb{N}^+$, the initial condition is $\boldsymbol{y}_0 \in \mathbb{R}^N$, and $\boldsymbol{\epsilon} \in E_{ad}$ denotes a control function belonging to E_{ad}, a nonempty and weakly compact subset of $L^2(0,T;\mathbb{R}^M)$ (e.g., a closed, convex, and bounded subset of $L^2(0,T;\mathbb{R}^M)$). The control matrix $B_\star \in \mathbb{R}^{N\times M}$, for $M \in \mathbb{N}^+$, is unknown and assumed to lie in the space spanned by a set of linearly independent matrices $\mathcal{B} = \{B_1,\dots,B_K\} \subset \mathbb{R}^{N\times M}$, $1 \le K \le NM$, and we write $B_\star = \sum_{j=1}^K \boldsymbol{\alpha}_{\star,j}B_j =: B(\boldsymbol{\alpha}_\star)$.

As in the case of the Hamiltonian reconstruction problem, to identify the unknown matrix $B_\star$ one can consider a set of control functions $(\boldsymbol{\epsilon}^m)_{m=1}^K \subset E_{ad}$ and use it experimentally to obtain the data $C\boldsymbol{y}_T(B_\star,\boldsymbol{\epsilon}^m)$, $m = 1,\dots,K$. Here, $\boldsymbol{y}_T(B_\star,\boldsymbol{\epsilon})$ denotes the solution of (4.1) at time T and corresponding to a control function $\boldsymbol{\epsilon}$ and to the control matrix $B_\star$. Further, $C \in \mathbb{R}^{P\times N}$ is a given observer matrix.

As in section 3, the reconstruction process is split into online and offline phases. In the offline phase, the GR algorithm computes the control functions $(\boldsymbol{\epsilon}^m)_{m=1}^K$. These are then used in the online phase, in which the laboratory data

$$C\boldsymbol{y}_T(B_\star,\boldsymbol{\epsilon}^m),\ m = 1,\dots,K, \tag{4.2}$$

are obtained and one solves the identification problem

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^K} \sum_{m=1}^K \|C\boldsymbol{y}_T(B_\star,\boldsymbol{\epsilon}^m) - C\boldsymbol{y}_T(B(\boldsymbol{\alpha}),\boldsymbol{\epsilon}^m)\|_2^2\,. \tag{4.3}$$

As in section 3, several variables are used in the presented reconstruction problem:

- The elements of the basis $\mathcal{B}$ can be arbitrarily chosen as data.
- Given a basis $\mathcal{B}$, the true unknown of the problem is $\boldsymbol{\alpha}_\star$ (or equivalently $B_\star$).
- The control functions are needed to produce the laboratory data (4.2), which are necessary to assemble the (final) inverse problem (4.3).

Algorithm 4.1. Greedy Reconstruction Algorithm (linear-quadratic case)

Require: A set of K linearly independent matrices $\mathcal{B} = (B_1, \dots, B_K)$.

1: Solve the initialization problem

$$\max_{\boldsymbol{\epsilon} \in E_{ad}} \|C\boldsymbol{y}_T(B_1, \boldsymbol{\epsilon}) - C\boldsymbol{y}_T(0,0)\|_2^2, \tag{4.5}$$

which gives the field $\boldsymbol{\epsilon}^1$, and set $k = 1$.

2: **while** $k \leq K - 1$ **do**

3: Fitting step: Find $(\boldsymbol{\alpha}_j^k)_{j=1,\dots,k}$ that solve the problem

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^k} \sum_{m=1}^{k} \left\| C\boldsymbol{y}_T(B_{k+1}, \boldsymbol{\epsilon}^m) - C\boldsymbol{y}_T(B^{(k)}(\boldsymbol{\alpha}), \boldsymbol{\epsilon}^m) \right\|_2^2, \tag{4.6}$$

where $B^{(k)}(\boldsymbol{\alpha}) := \sum_{j=1}^{k} \boldsymbol{\alpha}_j B_j$.

4: Discriminatory step: Find $\boldsymbol{\epsilon}^{k+1}$ that solves the problem

$$\max_{\boldsymbol{\epsilon} \in E_{ad}} \left\| C\boldsymbol{y}_T(B_{k+1}, \boldsymbol{\epsilon}) - C\boldsymbol{y}_T(B^{(k)}(\boldsymbol{\alpha}^k), \boldsymbol{\epsilon}) \right\|_2^2. \tag{4.7}$$

5: Update $k \leftarrow k + 1$.

6: **end while**

As for the Hamiltonian reconstruction problem, the ideal goal of the offline/online framework is to find a good approximation of the unknown operator for which the norm difference at time T between observed experimental data and numerically computed data is the smallest for any control function. In other words, we wish to find a matrix B of the form $B(\boldsymbol{\alpha}) := \sum_{j=1}^{K} \boldsymbol{\alpha}_j B_j$ that solves

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^K} \max_{\boldsymbol{\epsilon} \in E_{ad}} \|C\boldsymbol{y}_T(B_\star, \boldsymbol{\epsilon}) - C\boldsymbol{y}_T(B(\boldsymbol{\alpha}), \boldsymbol{\epsilon})\|_2^2. \tag{4.4}$$

The GR algorithm generates a set of K controls that attempt to distinguish numerical data for any two $B(\widehat{\boldsymbol{\alpha}}) \neq B(\widetilde{\boldsymbol{\alpha}})$, without performing any laboratory experiment. The GR algorithm for linear-quadratic reconstruction problems is given in Algorithm 4.1.

Since the convergence analysis performed in the next sections focuses on Algorithm 4.1, we wish to explain it in more detail. The idea is to generate controls that separate the observations of system (4.1) at time T for the different elements $B_1, \dots, B_K$, making possible the identification of their respective coefficients $\boldsymbol{\alpha}_1^\star, \dots, \boldsymbol{\alpha}_K^\star$ when solving (4.3). The initialization is performed by solving the optimal control problem (4.5), which aims at maximizing the distance (at time T) between the observed state of the uncontrolled system (namely, $\boldsymbol{y}_T(0,0)$ corresponding to $\boldsymbol{\epsilon} = 0$) and the observed state of the system

$$\dot{\boldsymbol{y}}(t) = A\boldsymbol{y}(t) + B_1\boldsymbol{\epsilon}(t), \quad \boldsymbol{y}(0) = \boldsymbol{y}_0.$$

The numerical solution of this maximization problem provides the first control function $\boldsymbol{\epsilon}^1$.

Assume now that the control functions $\boldsymbol{\epsilon}^1, \dots, \boldsymbol{\epsilon}^k$ are computed. The new element $\boldsymbol{\epsilon}^{k+1}$ is obtained by performing a fitting step (namely, solving problem (4.6)) and a discriminatory step (namely, solving problem (4.7)). In the fitting step, one compares the two systems

$$\begin{cases}\dot{\boldsymbol{y}}(t) = A\boldsymbol{y}(t) + B_{k+1}\boldsymbol{\epsilon}^m(t),\\ \boldsymbol{y}(0) = \boldsymbol{y}_0,\end{cases} \qquad \begin{cases}\dot{\boldsymbol{y}}(t) = A\boldsymbol{y}(t) + \left(\sum_{j=1}^k \boldsymbol{\alpha}_j B_j\right)\boldsymbol{\epsilon}^m(t),\\ \boldsymbol{y}(0) = \boldsymbol{y}_0,\end{cases}$$

with $B^{(k)}(\boldsymbol{\alpha}) := \sum_{j=1}^k \boldsymbol{\alpha}_j B_j$ and for $m \in \{1, \dots, k\}$, and looks for an $\boldsymbol{\alpha} \in \mathbb{R}^k$ for which their observed solutions at time T are as similar as possible (ideally the same, hence indistinguishable). We denote by $\boldsymbol{\alpha}^k = [\boldsymbol{\alpha}_1^k, \dots, \boldsymbol{\alpha}_k^k]^\top$ the vector computed by solving (4.6). This vector is used in the subsequent discriminatory step, which consists in solving the optimal control problem (4.7). Here, we compute a control function $\boldsymbol{\epsilon}^{k+1}$ that maximizes the distance (at time T) between the solutions of the two systems

$$\begin{cases}\dot{\boldsymbol{y}}(t) = A\boldsymbol{y}(t) + B_{k+1}\boldsymbol{\epsilon}(t),\\ \boldsymbol{y}(0) = \boldsymbol{y}_0,\end{cases} \qquad \begin{cases}\dot{\boldsymbol{y}}(t) = A\boldsymbol{y}(t) + \sum_{j=1}^k \boldsymbol{\alpha}_j^k B_j\boldsymbol{\epsilon}(t),\\ \boldsymbol{y}(0) = \boldsymbol{y}_0,\end{cases}$$

where now $\boldsymbol{\alpha}_j^k$ are fixed coefficients and the optimization variable is the control function $\boldsymbol{\epsilon}$. Notice that this maximization problem is well posed, as we will discuss in Lemma 5.2 in section 5.

We wish to remark again that, since the goal of the GR algorithm is to compute control functions that permit us to distinguish between the states of the system corresponding to any possible control matrix, the algorithm implicitly attempts to compute control functions that make the online identification problem locally uniquely solvable.

With these preparations, we are ready to present our convergence analysis.

5. Convergence analysis.

Our analysis is based on a reformulation of the algorithm that highlights the link between convergence and observability. We present the reformulation of the algorithm in a matrix-vector form in section 5.1, where the main idea of our convergence analysis and its relation with the observability properties of the system are first presented. Detailed analyses for fully observable and non-fully observable systems are provided in section 5.2 and section 5.3, respectively.

5.1. Matrix-vector formulation and convergence of the algorithm.

The convergence analysis presented in this section begins by recalling that one of the goals of the GR algorithm is to compute a set of control functions that makes the online identification problem (4.3) strictly convex in a neighborhood of the solution $\boldsymbol{\alpha}_\star$ (and hence locally uniquely solvable). It is then natural to begin with problem (4.3) and prove the following lemma, which gives us an equivalent matrix-vector formulation.

LEMMA 5.1 (online identification problem in matrix form). *Problem* (4.3) *is equivalent to*

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^K} \left\langle \boldsymbol{\alpha}_\star - \boldsymbol{\alpha}, \widehat{W}(\boldsymbol{\alpha}_\star - \boldsymbol{\alpha})\right\rangle, \tag{5.1}$$

where $\widehat{W} \in \mathbb{R}^{K\times K}$ *is defined as*

$$\widehat{W} := \sum_{m=1}^K W(\boldsymbol{\epsilon}^m) \tag{5.2}$$

with $W(\boldsymbol{\epsilon}^m) \in \mathbb{R}^{K\times K}$ *given by*

$$[W(\boldsymbol{\epsilon}^m)]_{\ell,j} := \left\langle \boldsymbol{\gamma}_\ell(\boldsymbol{\epsilon}^m), \boldsymbol{\gamma}_j(\boldsymbol{\epsilon}^m)\right\rangle \quad \textit{for } \ell, j = 1, \dots, K, \tag{5.3}$$

$$\boldsymbol{\gamma}_\ell(\boldsymbol{\epsilon}^m) := \int_0^T Ce^{(T-s)A}B_\ell\boldsymbol{\epsilon}^m(s)ds \quad \textit{for } m, \ell = 1, \dots, K. \tag{5.4}$$

Proof. Define $J(\boldsymbol{\alpha}) := \sum_{m=1}^{K} \|C\boldsymbol{y}_T(B_\star, \boldsymbol{\epsilon}^m) - C\boldsymbol{y}_T(B(\boldsymbol{\alpha}), \boldsymbol{\epsilon}^m)\|_2^2$, and notice that

$$\boldsymbol{y}_T(B_\star, \boldsymbol{\epsilon}^m) = e^{TA}\boldsymbol{y}_0 + \int_0^T e^{(T-s)A} B(\boldsymbol{\alpha}_\star)\boldsymbol{\epsilon}^m(s)ds,$$

$$\boldsymbol{y}_T(B(\boldsymbol{\alpha}), \boldsymbol{\epsilon}^m) = e^{TA}\boldsymbol{y}_0 + \int_0^T e^{(T-s)A} B(\boldsymbol{\alpha})\boldsymbol{\epsilon}^m(s)ds.$$

Recalling that $B(\boldsymbol{\alpha}) = \sum_{j=1}^{K} \boldsymbol{\alpha}_j B_j$, the function $J(\boldsymbol{\alpha})$ can be written as

$$\begin{aligned} J(\boldsymbol{\alpha}) &= \sum_{m=1}^{K} \left\| \int_0^T Ce^{(T-s)A}\Big(\sum_{j=1}^{K}(\boldsymbol{\alpha}_{\star,j} - \boldsymbol{\alpha}_j)B_\ell\Big)\boldsymbol{\epsilon}^m(s)ds \right\|_2^2 \\ &= \sum_{m=1}^{K}\sum_{\ell=1}^{K}\sum_{j=1}^{K}(\boldsymbol{\alpha}_{\star,\ell} - \boldsymbol{\alpha}_\ell)(\boldsymbol{\alpha}_{\star,j} - \boldsymbol{\alpha}_j)\big\langle \boldsymbol{\gamma}_\ell(\boldsymbol{\epsilon}^m), \boldsymbol{\gamma}_j(\boldsymbol{\epsilon}^m)\big\rangle, \end{aligned}$$

where the vectors $\boldsymbol{\gamma}_\ell(\boldsymbol{\epsilon}^m)$ are defined in (5.4). We can now write

$$\begin{aligned} J(\boldsymbol{\alpha}) &= \sum_{\ell=1}^{K}\sum_{j=1}^{K}(\boldsymbol{\alpha}_{\star,\ell} - \boldsymbol{\alpha}_\ell)(\boldsymbol{\alpha}_{\star,j} - \boldsymbol{\alpha}_j)\sum_{m=1}^{K}\big\langle \boldsymbol{\gamma}_\ell(\boldsymbol{\epsilon}^m), \boldsymbol{\gamma}_j(\boldsymbol{\epsilon}^m)\big\rangle \\ &= \left\langle \boldsymbol{\alpha}_\star - \boldsymbol{\alpha}, \sum_{m=1}^{K} W(\boldsymbol{\epsilon}^m)(\boldsymbol{\alpha}_\star - \boldsymbol{\alpha}) \right\rangle = \big\langle \boldsymbol{\alpha}_\star - \boldsymbol{\alpha}, \widehat{W}(\boldsymbol{\alpha}_\star - \boldsymbol{\alpha})\big\rangle, \end{aligned}$$

and the result follows. ∎

Notice that the matrices $W(\boldsymbol{\epsilon}^m)$ defined in (5.3) can be written as $W(\boldsymbol{\epsilon}^m) = \Gamma(\boldsymbol{\epsilon}^m)^\top\Gamma(\boldsymbol{\epsilon}^m)$, where $\Gamma(\boldsymbol{\epsilon}^m) = [\boldsymbol{\gamma}_1(\boldsymbol{\epsilon}^m) \cdots \boldsymbol{\gamma}_K(\boldsymbol{\epsilon}^m)]$. Hence, $W(\boldsymbol{\epsilon}^m)$ are Hermitian and positive semidefinite. This guarantees that $\widehat{W}$ is also Hermitian and positive semidefinite. Therefore, problem (5.1) is uniquely solved by $\boldsymbol{\alpha} = \boldsymbol{\alpha}_\star$ if and only if $\widehat{W}$ is positive definite, meaning that the GR algorithm actually aims at computing a set of control functions $(\boldsymbol{\epsilon}^m)_{m=1}^K$ that makes $\widehat{W}$ positive definite. We then need to study how the positivity of $\widehat{W}$ evolves during the iteration of the algorithm. To do so, the first step is to rewrite the problems (4.5), (4.6), and (4.7) also in a matrix form.

LEMMA 5.2 (the GR Algorithm 4.1 in matrix form). *Consider Algorithm* 4.1. *The following hold:*

- *The initialization problem* (4.5) *is equivalent to*

$$\max_{\boldsymbol{\epsilon}\in E_{ad}} [W(\boldsymbol{\epsilon})]_{1,1}. \tag{5.5}$$

- *The fitting-step problem* (4.6) *is equivalent to*

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^k} \big\langle \boldsymbol{\alpha}, \widehat{W}^k_{[1:k,1:k]}\boldsymbol{\alpha}\big\rangle - 2\big\langle \widehat{W}^k_{[1:k,k+1]}, \boldsymbol{\alpha}\big\rangle, \tag{5.6}$$

  where $\widehat{W}^k = \sum_{m=1}^{k} W(\boldsymbol{\epsilon}^m)$, *and (recalling section* 2*)* $\widehat{W}^k_{[1:k,1:k]} \in \mathbb{R}^{k\times k}$ *denotes the* $k \times k$ *upper-left block of* $\widehat{W}^k$ *and* $\widehat{W}^k_{[1:k,k+1]} \in \mathbb{R}^k$ *is a vector containing the first* k *components of the* $(k+1)$*th column of* $\widehat{W}^k$.
- *The discriminatory-step problem* (4.7) *is equivalent to*

$$\max_{\boldsymbol{\epsilon}\in E_{ad}} \big\langle \boldsymbol{v}, [W(\boldsymbol{\epsilon})]_{[1:k+1,1:k+1]}\boldsymbol{v}\big\rangle, \tag{5.7}$$

  where $W(\boldsymbol{\epsilon})$ *is defined in* (5.3) *and* $\boldsymbol{v} := [(\boldsymbol{\alpha}^k)^\top, -1]^\top$.

Moreover, problems (4.5)–(5.5), (4.6)–(5.6), *and* (4.7)–(5.7) *are well posed.*

Proof. The equivalences between (4.5), (4.6), (4.7) and (5.5), (5.6), and (5.7), respectively, can be proved by similar calculations to the one used in the proof of Lemma 5.1. We omit them for brevity.

Problem (4.6)–(5.6) is a quadratic minimization problem with quadratic function bounded from below by zero. Hence the existence of a minimizer follows.

Problems (4.5)–(5.5) and (4.7)–(5.7) are two classical optimal control problems. Since the admissible set E_{ad} is a weakly compact subset of $L^2(0,T;\mathbb{R}^M)$, the existence of a maximizer follows by standard arguments based on maximizing sequences and weak compactness; see, e.g., [5] and references therein. ∎

Using the matrix representation given in Lemma 5.2, we can now sketch the mathematical meaning of the iterations of the GR algorithm. Assume that at the kth iteration the submatrix $\widehat{W}^k_{[1:k,1:k]}$ is positive definite, but $\widehat{W}^k_{[1:k+1,1:k+1]}$ has a nontrivial (one-dimensional) kernel. The GR algorithm first tries to identify (by solving problem (5.6)) the kernel of $\widehat{W}^k_{[1:k+1,1:k+1]}$, and then attempts to compute (by solving problem (5.7)) a new control function $\boldsymbol{\epsilon}^{k+1}$ such that the matrix $W_{[1:k+1,1:k+1]}(\boldsymbol{\epsilon}^{k+1})$ is positive on the kernel $\widehat{W}^k_{[1:k+1,1:k+1]}$. If these happen, then the new updated matrix $\widehat{W}^{k+1} = \widehat{W}^k + W(\boldsymbol{\epsilon}^{k+1})$ has a positive definite upper-left block $\widehat{W}^{k+1}_{[1:k+1,1:k+1]}$. Moreover, if these two steps hold for any k, then the convergence follows since after the $(K-1)$th iteration the matrix $\widehat{W} = \widehat{W}^K$ is positive definite. Hence, two questions clearly arise:

1. Does the fitting step of the algorithm always compute the nontrivial kernel of $\widehat{W}^k_{[1:k+1,1:k+1]}$ (in case it is truly nontrivial)?
2. Does the discriminatory step of the algorithm always compute a control function $\boldsymbol{\epsilon}^{k+1}$ that makes $\widehat{W}^{k+1}_{[1:k+1,1:k+1]}$ positive definite?

The first question can be answered with the help of the following technical lemma.

LEMMA 5.3 (on the kernel of Hermitian positive semidefinite matrices). *Consider a symmetric positive semidefinite matrix $\widetilde{G} \in \mathbb{R}^{n\times n}$ of the form*

$$\widetilde{G} = \begin{bmatrix} G & \boldsymbol{b} \\ \boldsymbol{b}^\top & c \end{bmatrix},$$

where $G \in \mathbb{R}^{(n-1)\times(n-1)}$ is symmetric and positive definite, and $\boldsymbol{b} \in \mathbb{R}^{n-1}$ and $c \in \mathbb{R}$ are such that the kernel of $\widetilde{G}$ is nontrivial. Then

$$\ker(\widetilde{G}) = \operatorname{span}\left\{ \begin{bmatrix} G^{-1}\boldsymbol{b} \\ -1 \end{bmatrix} \right\}.$$

Proof. Since the kernel of $\widetilde{G}$ is nontrivial, there exists a nonzero vector $\boldsymbol{u} = \left[\begin{smallmatrix} \boldsymbol{v} \\ d \end{smallmatrix}\right] \in \mathbb{R}^n \setminus \{0\}$ (with $\boldsymbol{v} \in \mathbb{R}^{n-1}$ and $d \in \mathbb{R}$) such that $\widetilde{G}\boldsymbol{u} = 0$. Moreover, since G is positive definite, the kernel of $\widetilde{G}$ must be one-dimensional and equal to the span of $\{\boldsymbol{u}\}$. Using the structure of $\boldsymbol{u}$, we write $\widetilde{G}\boldsymbol{u} = 0$ as

$$\begin{cases} G\boldsymbol{v} + d\,\boldsymbol{b} = 0, \\ \boldsymbol{b}^\top \boldsymbol{v} + dc = 0, \end{cases} \quad \overset{G \text{ invertible}}{\Longleftrightarrow} \quad \begin{cases} \boldsymbol{v} = -dG^{-1}\boldsymbol{b}, \\ -d\,\boldsymbol{b}^\top G^{-1}\boldsymbol{b} + dc = 0. \end{cases} \tag{5.8}$$

Now, suppose that $d = 0$. This implies that $\boldsymbol{v} = -dG^{-1}\boldsymbol{b} = 0$, which in turn implies that $\boldsymbol{u} = 0$. However, this is a contradiction to the fact that $\boldsymbol{u} \neq 0$. Hence $d \neq 0$. The result follows by the right equations in (5.8) (divided by $-d$). ∎

Recalling the equivalent form (5.6) of the fitting-step problem (4.6), one can clearly see that if $\widehat{W}^k_{[1:k,1:k]}$ is positive definite, then the unique solution to (5.6) is given by $\boldsymbol{\alpha}^k = (\widehat{W}^k_{[1:k,1:k]})^{-1}\widehat{W}^k_{[1:k,k+1]}$. On the other hand, if we set

$$\widetilde{G} = \widehat{W}^k_{[1:k+1,1:k+1]},\ G = \widehat{W}^k_{[1:k,1:k]},\ \boldsymbol{b} = \widehat{W}^k_{[1:k,k+1]},\ c = \widehat{W}^k_{[k+1,k+1]},$$

then Lemma 5.3 guarantees that the vector $\boldsymbol{v} := [(\boldsymbol{\alpha}^k)^\top,\ -1]^\top$ spans the kernel of $\widehat{W}^k_{[1:k+1,1:k+1]}$, if this is nontrivial. Therefore, we have

$$\ker(\widehat{W}^k_{[1:k+1,1:k+1]}) = \operatorname{span}\left\{\begin{bmatrix}(\widehat{W}^k_{[1:k,1:k]})^{-1}\widehat{W}^k_{[1:k,k+1]}\\ -1\end{bmatrix}\right\} = \operatorname{span}\left\{\boldsymbol{v} := \begin{bmatrix}\boldsymbol{\alpha}^k\\ -1\end{bmatrix}\right\}.$$

This means that if $\widehat{W}^k_{[1:k+1,1:k+1]}$ has a rank defect, then the GR algorithm finds this defect by the fitting step.

The answer to the second question posed above is more complicated. In order to formulate it properly, we need to recall the definition of observability of an input/output dynamical system of the form

$$\begin{aligned}&\dot{\boldsymbol{y}}(t) = A\boldsymbol{y}(t) + B\boldsymbol{\epsilon}(t), \quad \boldsymbol{y}(0) = \boldsymbol{y}_0,\\ &\boldsymbol{z}(t) = C\boldsymbol{y}(t)\end{aligned} \tag{5.9}$$

with $A \in \mathbb{R}^{N\times N}$, $B \in \mathbb{R}^{N\times M}$, $C \in \mathbb{R}^{P\times N}$; see, e.g., [18].

Definition 5.4 (observable input-output linear systems). *The input-output linear system* (5.9) *is said to be observable if the initial state* $\boldsymbol{y}(0) = \boldsymbol{y}_0$ *can be uniquely determined from input/output measurements. Equivalently,* (5.9) *is observable if and only if the observability matrix*

$$\mathcal{O}_N(C, A) := \begin{bmatrix} C\\ CA\\ \vdots\\ CA^{N-1}\end{bmatrix} \tag{5.10}$$

has full column rank.

Notice that the matrix B does not affect the observability of system (5.9).

We now analyze the convergence of the algorithm in the case of fully observable systems (namely, rank $\mathcal{O}_N(C, A) = N$) in section 5.2 and in case of non-fully observable systems (namely, rank $\mathcal{O}_N(C, A) < N$) in section 5.3.

5.2. The case of fully observable systems. Let us assume that the system is observable, namely, that rank $\mathcal{O}_N(C, A) = N$. We show in this section that this is a sufficient condition for the GR algorithm to make the matrix $\widehat{W}$ positive definite. To do so, we first prove the following lemma regarding the discriminatory step. Notice that the proof of this result is inspired by classical Kalman controllability theory; see, e.g., [6].

Lemma 5.5 (discriminatory-step problem for fully observable systems). *Assume that the matrices* $A \in \mathbb{R}^{N\times N}$ *and* $C \in \mathbb{R}^{P\times N}$ *are such that* rank $\mathcal{O}_N(C, A) = N$. *Let* $\widehat{W}^k_{[1:k,1:k]}$ *be positive definite,* $\boldsymbol{\alpha}^k$ *the solution to the fitting-step problem* (4.6), *and*

$\boldsymbol{v} = [(\boldsymbol{\alpha}^k)^\top, -1]^\top$. *Then any solution* $\boldsymbol{\epsilon}^{k+1}$ *of the discriminatory-step problem* (4.7) *satisfies*

$$\langle \boldsymbol{v}, W_{[1:k+1,1:k+1]}(\boldsymbol{\epsilon}^{k+1})\boldsymbol{v}\rangle = \left\| \int_0^T Ce^{(T-s)A} \left(B_{k+1} - \sum_{j=1}^k \boldsymbol{\alpha}_j^k B_j \right) \boldsymbol{\epsilon}^{k+1}(s) ds \right\|_2^2 > 0$$

for $k = 0, 1, \dots, K-1$.

Proof. Let us define $\widetilde{B} := B_{k+1} - \sum_{j=1}^k \boldsymbol{\alpha}_j^k B_j$. Since the matrices $B_1, \dots, B_{k+1}$ are assumed to be linearly independent, $\widetilde{B}$ is nonzero.

Now, we consider an arbitrary $\delta \in (0, T)$ and define a control function $\widetilde{\boldsymbol{\epsilon}} \in E_{ad}$ as

$$\widetilde{\boldsymbol{\epsilon}}(s) := \begin{cases} 0, & 0 \le s < \delta, \\ \boldsymbol{e}_i, & \delta \le s \le T, \end{cases}$$

where $\boldsymbol{e}_i \in \mathbb{R}^M$ is the ith canonical vector for some index $1 \le i \le M$. Further, we denote by $\widetilde{\boldsymbol{b}}_i$ the ith column of $\widetilde{B}$. Since $\widetilde{B}$ is nonzero, we can choose the index i such that $\widetilde{\boldsymbol{b}}_i \neq 0$. Now, we compute

$$\begin{aligned} \int_0^T Ce^{(T-s)A}\widetilde{B}\widetilde{\boldsymbol{\epsilon}}(s)ds &= \int_\delta^T Ce^{(T-s)A}\widetilde{\boldsymbol{b}}_i ds = \int_\delta^T C\left[\sum_{j=0}^\infty \frac{(T-s)^j A^j}{j!}\right]\widetilde{\boldsymbol{b}}_i ds \\ &\overset{(\star)}{=} \left[\sum_{j=0}^\infty \int_\delta^T \frac{(T-s)^j}{j!} ds\, CA^j\right]\widetilde{\boldsymbol{b}}_i = \left[\sum_{j=0}^\infty \frac{(T-\delta)^{j+1}}{(j+1)!} CA^j\right]\widetilde{\boldsymbol{b}}_i \\ &= \sum_{j=0}^\infty \beta_j(\delta) CA^j \widetilde{\boldsymbol{b}}_i, \end{aligned}$$

where $\beta_j(\delta) := \frac{(T-\delta)^{j+1}}{(j+1)!}$ and we used the dominated convergence theorem (see, e.g., [16, Theorem 1.34]) to interchange integral and infinite sum and obtain the equality $(\star)$. Since the observability matrix $\mathcal{O}_N(C, A)$ has full rank and $\widetilde{\boldsymbol{b}}_i \neq 0$, there exists an index $0 \le j \le N-1$ such that $CA^j\widetilde{\boldsymbol{b}}_i \neq 0$. Hence, $f(\delta) := \sum_{j=0}^\infty \beta_j(\delta) CA^j\widetilde{\boldsymbol{b}}_i$ is an analytic function for $\delta \in (0, T)$ and such that $f \neq 0$.[3] We also know that (nonconstant) analytic functions have isolated roots; see, e.g., [16, Theorem 10.18]. Therefore we can find a $\delta \in (0, T)$ such that $\sum_{j=0}^\infty \beta_j(\delta) CA^j\widetilde{\boldsymbol{b}}_i \neq 0$ and obtain the existence of an $\widetilde{\boldsymbol{\epsilon}} \in E_{ad}$ such that

$$\int_0^T Ce^{(T-s)A}\widetilde{B}\widetilde{\boldsymbol{\epsilon}}(s)ds \neq 0.$$

[3] To see it, recall that $\beta_j(\delta) = \frac{(T-\delta)^{j+1}}{(j+1)!}$, consider a function $g(x) = \sum_{j=0}^\infty \frac{x^{j+1}}{(j+1)!}\gamma_j$, and assume that there exists at least one integer k such that $\gamma_k \neq 0$. Now, if we pick the minimum integer $\widehat{k}$ such that $\gamma_{\widehat{k}} \neq 0$, we have that $g(x) = \frac{x^{\widehat{k}+1}}{(\widehat{k}+1)!}\gamma_{\widehat{k}} + \sum_{j=\widehat{k}+1}^\infty \frac{x^{j+1}}{(j+1)!}\gamma_j$. For $x \to 0$, the first term behaves as $O(x^{\widehat{k}+1})$, while the second term as $O(x^{\widehat{k}+2})$. Hence, there exists a point $y > 0$ such that $g(y) \neq 0$.

This implies that

$$\begin{aligned}\left\langle \boldsymbol{v}, W_{[1:k+1,1:k+1]}(\boldsymbol{\epsilon}^{k+1})\boldsymbol{v}\right\rangle &= \left\| \int_0^T Ce^{(T-s)A}\left(B_{k+1} - \sum_{\ell=1}^{k} \boldsymbol{\alpha}_\ell^k B_\ell\right)\boldsymbol{\epsilon}^{k+1}(s)ds \right\|_2^2 \\ &\geq \left\| \int_0^T Ce^{(T-s)A}\left(B_{k+1} - \sum_{\ell=1}^{k} \boldsymbol{\alpha}_\ell^k B_\ell\right)\widetilde{\boldsymbol{\epsilon}}(s)ds \right\|_2^2 \\ &= \left\| \int_0^T Ce^{(T-s)A}\widetilde{B}\widetilde{\boldsymbol{\epsilon}}(s)ds \right\|_2^2 > 0,\end{aligned}$$

where we have used that $\boldsymbol{\epsilon}^{k+1}$ is a maximizer for problem (4.7). ∎

Now, we can prove our first main convergence result.

THEOREM 5.6 (convergence of the GR algorithm for fully observable systems). *Assume that the matrices $A \in \mathbb{R}^{N\times N}$ and $C \in \mathbb{R}^{P\times N}$ are such that* rank $\mathcal{O}_N(C, A) = N$. *Let $K \in \{1, \dots, MN\}$ be arbitrary, and let $\{\boldsymbol{\epsilon}^1, \dots, \boldsymbol{\epsilon}^K\} \subset E_{ad}$ be a family of controls generated by the GR Algorithm* 4.1. *Then the matrix $\widehat{W}$ defined in* (5.2) *is positive definite, and online identification problem* (4.3) *is uniquely solvable by $\boldsymbol{\alpha} = \boldsymbol{\alpha}_\star$.*

Proof. By Lemma 5.1 it is sufficient to show that the matrix $\widehat{W}$ corresponding to the controls $\boldsymbol{\epsilon}^1, \dots, \boldsymbol{\epsilon}^K$ generated by the algorithm is positive definite. The proof of this claim proceeds by induction.

Lemma 5.5 guarantees that there exists an $\boldsymbol{\epsilon}^1$ such that $[W(\boldsymbol{\epsilon}^1)]_{1,1} > 0$. Now, we assume that $\widehat{W}^k_{[1:k,1:k]}$ is positive definite, and we show that $\widehat{W}^{k+1}_{[1:k+1,1:k+1]}$ is positive definite as well.

If $\widehat{W}^k_{[1:k+1,1:k+1]}$ is positive definite, then

$$\widehat{W}^{k+1}_{[1:k+1,1:k+1]} = \widehat{W}^k_{[1:k+1,1:k+1]} + W(\boldsymbol{\epsilon}^k)_{[1:k+1,1:k+1]}$$

is positive definite as well, since $W(\boldsymbol{\epsilon}^k)_{[1:k+1,1:k+1]}$ is positive semidefinite.

Assume now that the submatrix $\widehat{W}^k_{[1:k+1,1:k+1]}$ has a nontrivial kernel. Since $\widehat{W}^k_{[1:k,1:k]}$ is positive definite (induction hypothesis), problem (5.6) is uniquely solvable with solution $\boldsymbol{\alpha}^k$. Then, by Lemma 5.3 the (one-dimensional) kernel of $\widehat{W}^k_{[1:k+1,1:k+1]}$ is the span of the the vector $\boldsymbol{v} = [(\boldsymbol{\alpha}^k)^\top, -1]^\top$. Finally, using Lemma 5.5 we obtain that the solution $\boldsymbol{\epsilon}^{k+1}$ to the discriminatory-step problem satisfies

$$0 < \left\langle \boldsymbol{v}, [W(\boldsymbol{\epsilon}^{k+1})]_{[1:k+1,1:k+1]}\boldsymbol{v}\right\rangle.$$

Hence, the matrix $[W(\boldsymbol{\epsilon}^{k+1})]_{[1:k+1,1:k+1]}$ is positive definite on the span of $\boldsymbol{v}$. Therefore $\widehat{W}^{k+1}_{[1:k+1,1:k+1]} = \widehat{W}^k_{[1:k+1,1:k+1]} + [W(\boldsymbol{\epsilon}^{k+1})]_{[1:k+1,1:k+1]}$ is positive definite. ∎

Remark 5.7 (uniqueness of solution of the min-max problem (4.4)). Under the assumption that the system is fully observable, the min-max problem (4.4) is also uniquely solvable with $\boldsymbol{\alpha} = \boldsymbol{\alpha}_\star$. To see this, we first note that (4.4) can be written in terms of $W(\boldsymbol{\epsilon})$:

$$\|C\boldsymbol{y}_T(B_\star,\boldsymbol{\epsilon}) - C\boldsymbol{y}_T(B(\boldsymbol{\alpha}),\boldsymbol{\epsilon})\|^2 = \left\| \int_0^T Ce^{(T-s)A} \left(\sum_{j=1}^K (\boldsymbol{\alpha}_j - \boldsymbol{\alpha}_{\star,j}) B_j \right) \boldsymbol{\epsilon}(s) ds \right\|_2^2 = \big\langle (\boldsymbol{\alpha} - \boldsymbol{\alpha}_\star), W(\boldsymbol{\epsilon})(\boldsymbol{\alpha} - \boldsymbol{\alpha}_\star) \big\rangle.$$

Now, similarly as in the proof of Lemma 5.5 and using the full observability of the system, one can show that for any $\widehat{\boldsymbol{\alpha}} \in \mathbb{R}^{NM}$ with $\widehat{\boldsymbol{\alpha}} \neq \boldsymbol{\alpha}_\star$ there exists a control $\boldsymbol{\epsilon}(\widehat{\boldsymbol{\alpha}})$ such that $\big\langle (\widehat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}_\star), W(\boldsymbol{\epsilon}(\widehat{\boldsymbol{\alpha}}))(\widehat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}_\star) \big\rangle > 0$. Therefore the unique solution to (4.4) is $\boldsymbol{\alpha} = \boldsymbol{\alpha}_\star$.

Notice that Theorem 5.6 does not require any particular assumption on the matrices $B_1, \dots, B_K$, which can be arbitrarily chosen with the only constraint to be linearly independent. Moreover, the number $K \in \{1, \dots, MN\}$ can be fixed arbitrarily, and the GR algorithm will compute control functions that permit the exact reconstruction of the coefficients of the linear combination of the first K components of $B_\star$ in a basis $\{B_1, \dots, B_{MN}\}$. To be more precise, if the unknown $B_\star$ belongs to the span of K the linearly independent matrices $B_1, \dots, B_K$ used by the algorithm, then, using the control functions generated by the GR algorithm, the unknown $B_\star$ can be fully reconstructed. If $B_\star$ lies in the span of $\widetilde{K} \in \{K+1, K+2, \dots, MN\}$ linearly independent matrices $B_1, \dots, B_{\widetilde{K}}$, but only the first K of these are used by the algorithm (and in the online identification problem), then one reconstructs exactly the K coefficients corresponding to the first elements $B_1, \dots, B_K$. Furthermore, the ordering of the K considered matrices does not affect the convergence result of Theorem 5.6.

5.3. The case of non-fully observable systems. The observations and results of section 5.2 are no longer true if the system is non-fully observable, that is, $\operatorname{rank} \mathcal{O}_N(C,A) = \mathcal{R} < N$. In this case, the choice of the linearly independent matrices $B_1, \dots, B_K$ and their ordering become crucial for the algorithm. In particular, we are going to show that the method can recover at most $K = \mathcal{R}M$ components of the unknown vector $\boldsymbol{\alpha}_\star$ if appropriate matrices $B_1, \dots, B_K$ are chosen. Moreover, we will see that an inappropriate choice of matrices $B_1, \dots, B_K$ can lead to completely wrong results corresponding to an arbitrarily large error.

For our analysis, we begin by choosing a set of $K = NM$ matrices by exploiting the kernel of the observability matrix. In particular, recalling that $\operatorname{rank} \mathcal{O}_N(C,A) = \mathcal{R} < N$, the rank-nullity theorem allows us to consider a basis $\{\boldsymbol{v}_j\}_{j=1}^N \subset \mathbb{R}^N$ of $\mathbb{R}^N$ such that

$$\boldsymbol{v}_j \notin \ker \mathcal{O}_N(C,A), \qquad j = 1, \dots, \mathcal{R}, \tag{5.11}$$

$$\boldsymbol{v}_j \in \ker \mathcal{O}_N(C,A), \qquad j = \mathcal{R}+1, \dots, N, \tag{5.12}$$

where $\operatorname{span}\{\boldsymbol{v}_j\}_{j=\mathcal{R}+1}^N = \ker \mathcal{O}_N(C,A)$. We now define a basis $\{B_k^{\mathcal{O}}\}_{k=1}^{NM}$ of $\mathbb{R}^{N\times M}$ as

$$\begin{aligned} B_1^{\mathcal{O}} = \boldsymbol{v}_1\boldsymbol{e}_1^\top,\ B_2^{\mathcal{O}} = \boldsymbol{v}_1\boldsymbol{e}_2^\top,\ \dots,\ B_M^{\mathcal{O}} = \boldsymbol{v}_1\boldsymbol{e}_M^\top, \\ B_{M+1}^{\mathcal{O}} = \boldsymbol{v}_2\boldsymbol{e}_1^\top,\ B_{M+2}^{\mathcal{O}} = \boldsymbol{v}_2\boldsymbol{e}_2^\top,\ \dots,\ B_{2M}^{\mathcal{O}} = \boldsymbol{v}_2\boldsymbol{e}_M^\top, \\ \vdots \qquad\qquad \vdots \qquad\qquad \vdots \\ B_{(N-1)M+1}^{\mathcal{O}} = \boldsymbol{v}_N\boldsymbol{e}_1^\top,\ B_{(N-1)M+2}^{\mathcal{O}} = \boldsymbol{v}_N\boldsymbol{e}_2^\top,\ \dots,\ B_{NM}^{\mathcal{O}} = \boldsymbol{v}_N\boldsymbol{e}_M^\top, \end{aligned} \tag{5.13}$$

where $\boldsymbol{e}_\ell \in \mathbb{R}^M$, for $\ell = 1, \dots, M$, are the canonical vectors in $\mathbb{R}^M$. Notice that, since the vectors $(\boldsymbol{v}_j)_{j=1}^N$ are linearly independent, the set $\{B_k^{\mathcal{O}}\}_{k=1}^{NM}$ is a basis of $\mathbb{R}^{N\times M}$.

From a computational point of view, the vectors $\boldsymbol{v}_j$ can be obtained by a singular value decomposition (SVD) of the observability matrix $\mathcal{O}_N(C,A) = U\Sigma V^\top$, where the columns of V form a basis of $\mathbb{R}^N$ and the last $N-\mathcal{R}$ columns of V span the kernel of $\mathcal{O}_N(C,A)$; see, e.g., [20, Theorem 5.2]. Therefore, one can set $\boldsymbol{v}_j = V_{[:,j]}$, $j = 1,\dots,N$.

Our first result for non-fully observable systems says that, if the basis $\{B_k^{\mathcal{O}}\}_{k=1}^{NM}$ is considered, then we can reduce the reconstruction of $B_\star = \sum_{j=1}^{MN} \boldsymbol{\alpha}_{\star,j} B_j^{\mathcal{O}}$ only to the first $\mathcal{R}M$ coefficients $\boldsymbol{\alpha}_1,\dots,\boldsymbol{\alpha}_{\mathcal{R}M}$. This is proved in the next lemma, where we use the notation

$$B_{\mathcal{R}}(\boldsymbol{\alpha}_\star) := \sum_{j=1}^{\mathcal{R}M} \boldsymbol{\alpha}_{\star,j} B_j^{\mathcal{O}}. \tag{5.14}$$

LEMMA 5.8 (online identification problem for non-fully observable systems). *Consider the basis $\{B_k^{\mathcal{O}}\}_{k=1}^{NM}$ constructed as in* (5.13) *(with vectors $\boldsymbol{v}_j$, $j = 1,\dots,N$, as in* (5.11)–(5.12)*). The online least-squares problem* (4.3) *(with $K = MN$) is equivalent to*

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^{\mathcal{R}M}} \sum_{m=1}^{NM} \|C\boldsymbol{y}_T(B_\star,\boldsymbol{\epsilon}^m) - C\boldsymbol{y}_T(B_{\mathcal{R}}(\boldsymbol{\alpha}),\boldsymbol{\epsilon}^m)\|_2^2.$$

Proof. Notice that, for any $\ell \in \{1,2,\dots,NM\}$ and $s \in [0,T]$, there exist N functions $\widetilde{\beta}_j$ such that

$$\begin{aligned} Ce^{(T-s)A}B_\ell^{\mathcal{O}} &= C\sum_{j=0}^{\infty} \frac{(T-s)^j}{j!} A^j B_\ell^{\mathcal{O}} \overset{(\star)}{=} C\left[\sum_{j=0}^{N-1} \widetilde{\beta}_j(s)A^j\right] B_\ell^{\mathcal{O}} \\ &= \left[\widetilde{\beta}_0(s)I_N, \widetilde{\beta}_1(s)I_N, \dots, \widetilde{\beta}_{N-1}(s)I_N\right]\mathcal{O}_N(C,A)B_\ell^{\mathcal{O}}, \end{aligned}$$

where we have used the Cayley–Hamilton theorem (see, e.g., [12, page 109]) to obtain the equality $(\star)$. If $\ell \in \{\mathcal{R}M+1,\dots,NM\}$, then $B_\ell^{\mathcal{O}} = \boldsymbol{v}_j \boldsymbol{e}_i^\top$ with $j \geq \mathcal{R}+1$; hence $\boldsymbol{v}_j \in \ker \mathcal{O}_N(C,A)$ and therefore

$$\mathcal{O}_N(C,A)B_\ell^{\mathcal{O}} = \underbrace{\mathcal{O}_N(C,A)\boldsymbol{v}_j}_{=0} \boldsymbol{e}_i^\top = 0.$$

Hence, $Ce^{(T-s)A}B_\ell^{\mathcal{O}} = 0$ for all $\ell \in \{\mathcal{R}M+1,\dots,NM\}$ and $s \in [0,T]$. Thus

$$\int_0^T Ce^{(T-s)A}B_\ell^{\mathcal{O}}\boldsymbol{\epsilon}(s)ds = 0$$

for any control function $\boldsymbol{\epsilon} \in E_{ad}$. Now, recalling the definition of $J(\boldsymbol{\alpha})$ from the proof of Lemma 5.1, our claim follows by writing the least-squares problem (4.3) as

$$\begin{aligned} J(\boldsymbol{\alpha}) &= \sum_{m=1}^{NM} \left\| \sum_{j=1}^{NM} (\boldsymbol{\alpha}_{\star,j} - \boldsymbol{\alpha}_j) \int_0^T Ce^{(T-s)A}B_j^{\mathcal{O}}\boldsymbol{\epsilon}^m(s)ds \right\|_2^2 \\ &= \sum_{m=1}^{NM} \left\| \sum_{j=1}^{\mathcal{R}M} (\boldsymbol{\alpha}_{\star,j} - \boldsymbol{\alpha}_j) \int_0^T Ce^{(T-s)A}B_j^{\mathcal{O}}\boldsymbol{\epsilon}^m(s)ds \right\|_2^2. \end{aligned}$$

∎

Lemma 5.8 implies that the coefficients $\boldsymbol{\alpha}_{\mathcal{R}M+1}, \dots, \boldsymbol{\alpha}_{MN}$ do not affect the cost function to be minimized. Therefore, as shown in Corollary 5.11, any vector $\boldsymbol{\alpha} \in \mathbb{R}^{MN}$ of the form

$$\boldsymbol{\alpha} = [\boldsymbol{\alpha}_1^\star, \cdots, \boldsymbol{\alpha}_{\mathcal{R}M}^\star, \gamma_{\mathcal{R}M+1}, \cdots, \gamma_{MN}]^\top$$

is a global solution to (4.3) for any $\gamma_j \in \mathbb{R}$, $j = \mathcal{R}M + 1, \dots, MN$. This means that one uses really only the first $\mathcal{R}M$ elements of the basis. In fact, as we are going to show in Lemma 5.9 and Theorem 5.10, only their corresponding coefficients can be reconstructed, while no information can be obtained for the remaining ones. It is therefore natural, for $\operatorname{rank} \mathcal{O}_N(C, A) = \mathcal{R} < N$, to use the GR algorithm with only the first $\mathcal{R}M$ basis elements $B_1^{\mathcal{O}}, \dots, B_{\mathcal{R}M}^{\mathcal{O}}$. In this case, the proof of convergence for the GR algorithm is analogous to what we have done to obtain Theorem 5.6. We first prove a version of Lemma 5.5 adapted to non-fully observable systems.

LEMMA 5.9 (discriminatory-step problem for non-fully observable systems). *Assume that* $\operatorname{rank} \mathcal{O}_N(C, A) = \mathcal{R} < N$ *and that the GR algorithm is run until the* k*th iteration, with* $k < \mathcal{R}M$*, using the linearly independent matrices* $B_1^{\mathcal{O}}, \dots, B_{\mathcal{R}M}^{\mathcal{O}}$ *defined in* (5.13)*. Let* $\widehat{W}^k_{[1:k,1:k]}$ *be positive definite, and let* $\boldsymbol{\alpha}^k$ *be the solution to the fitting-step problem* (4.6)*. Then any solution* $\boldsymbol{\epsilon}^{k+1}$ *of the discriminatory-step problem* (4.7) *satisfies, for* $k = 1, \dots \mathcal{R}M - 1$*,*

$$\left\langle \boldsymbol{v}, W_{[1:k+1,1:k+1]}(\boldsymbol{\epsilon}^{k+1})\boldsymbol{v} \right\rangle = \left\| \int_0^T C e^{(T-s)A} \left(B_{k+1}^{\mathcal{O}} - \sum_{j=1}^{k} \boldsymbol{\alpha}_j^k B_j^{\mathcal{O}} \right) \boldsymbol{\epsilon}^{k+1}(s) ds \right\|_2^2 > 0,$$

where $\boldsymbol{v} := [(\boldsymbol{\alpha}^k)^\top, \ -1]^\top$ *for* $k = 0, 1, \dots, K-1$*.*

Proof. Notice that, since the matrices $B_1^{\mathcal{O}}, \dots, B_{\mathcal{R}M}^{\mathcal{O}}$ are linearly independent and defined as in (5.13), we have that $\mathcal{O}_N(C, A)(B_{k+1}^{\mathcal{O}} - \sum_{j=1}^{k} \boldsymbol{\alpha}_j^k B_j^{\mathcal{O}}) \neq 0$.

With this observation, the result can be proved exactly as Lemma 5.5. ∎

Using Lemma 5.9, we can prove convergence for the GR Algorithm 4.1 in case the matrices $B_1^{\mathcal{O}}, \dots, B_{\mathcal{R}M}^{\mathcal{O}}$ defined in (5.13) are used.

THEOREM 5.10 (convergence of the GR algorithm for non-fully observable systems). *Let* $(\boldsymbol{\epsilon}^m)_{m=1}^{\mathcal{R}M} \subset E_{ad}$ *be a family of controls generated by the GR Algorithm* 4.1 *with* $K = \mathcal{R}M$ *and using the matrices* $B_1^{\mathcal{O}}, \dots, B_{\mathcal{R}M}^{\mathcal{O}}$ *defined in* (5.13)*. Then the least-squares problem*

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^{\mathcal{R}M}} \sum_{m=1}^{\mathcal{R}M} \| C\boldsymbol{y}_T(B_\star, \boldsymbol{\epsilon}^m) - C\boldsymbol{y}_T(B_{\mathcal{R}}(\boldsymbol{\alpha}), \boldsymbol{\epsilon}^m) \|_2^2, \tag{5.15}$$

where $B_{\mathcal{R}}(\boldsymbol{\alpha})$ *is defined in* (5.14)*, is uniquely solvable with* $\boldsymbol{\alpha}_j = \boldsymbol{\alpha}_{\star,j}$*,* $j = 1, \dots, \mathcal{R}M$*.*

Proof. The proof is the same as that of Theorem 5.6, where one should use Lemma 5.9 instead of Lemma 5.5. ∎

Theorem 5.10 allows us to prove the next corollary, which characterizes the result of the GR algorithm when more than $\mathcal{R}M$ basis elements of (5.13) are used.

COROLLARY 5.11 (more on the convergence for non-fully observable systems). *Let* $(\boldsymbol{\epsilon}^m)_{m=1}^{K} \subset E_{ad}$*, with* $K > \mathcal{R}M$*, be a family of controls generated by the GR*

Algorithm 4.1 *using the matrices* $B_1^{\mathcal{O}}, \dots, B_K^{\mathcal{O}}$ *defined in* (5.13). *Then the set of all global minimum points for the least-squares problem,*

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^K} \sum_{m=1}^{K} \left\| C\boldsymbol{y}_T(B_\star, \boldsymbol{\epsilon}^m) - C\boldsymbol{y}_T(B^{(K)}(\boldsymbol{\alpha}), \boldsymbol{\epsilon}^m) \right\|_2^2,$$

is given by $\{\boldsymbol{\alpha} \in \mathbb{R}^K : \boldsymbol{\alpha}_j = \boldsymbol{\alpha}_{\star,j},\ j = 1, \dots, \mathcal{R}M\}$.

Proof. Theorem 5.10 (and Theorem 5.6) and its proof allow us to obtain that, using the first $\mathcal{R}M$ controls generated by the GR algorithm, the matrix $\widehat{W}^{\mathcal{R}M} \in \mathbb{R}^{K\times K}$ has a positive definite upper-left submatrix $\widehat{W}^{\mathcal{R}M}_{[1:\mathcal{R}M,1:\mathcal{R}M]}$ and all the other entries $[\widehat{W}^{\mathcal{R}M}]_{\ell,j}$ are zero. Indeed, recalling the vectors $\boldsymbol{\gamma}_k(\boldsymbol{\epsilon}^m)$ defined in (5.4), for any $B_k^{\mathcal{O}}$ with $k \geq \mathcal{R}M + 1$, we have that $\mathcal{O}_N(C, A)B_k^{\mathcal{O}} = 0$ and thus

$$\boldsymbol{\gamma}_k(\boldsymbol{\epsilon}^m) = \int_0^T Ce^{(T-s)A} B_k^{\mathcal{O}} \boldsymbol{\epsilon}^m(s) ds = 0$$

for any $T > 0$ and any $m = 1, \dots, \mathcal{R}M$. Similarly, the matrices $W(\boldsymbol{\epsilon}^m)$ for $m > \mathcal{R}M$ have the same structure, namely, that their only nonzero components can be the upper-left submatrices $[W(\boldsymbol{\epsilon}^m)]_{[1:\mathcal{R}M,1:\mathcal{R}M]}$. Therefore, the matrix $\widehat{W} = \widehat{W}^K$ has a positive definite upper-left submatrix $\widehat{W}_{[1:\mathcal{R}M,1:\mathcal{R}M]}$, while all its other entries are zero. Therefore, the result follows by Lemma 5.1. ∎

Remark 5.12 (more about the kernel of $\mathcal{O}_N(C, A)$ and identifiability). Corollary 5.11 guarantees that, if the basis $(B_j^{\mathcal{O}})_{j=1}^K$ is used with $K > \mathcal{R}M$, then one can reconstruct exactly $\mathcal{R}M$ coefficients, while nothing can be said about the coefficients α_j for $j > \mathcal{R}M$. This is due to the structure of the matrix $\widehat{W}^{\mathcal{R}M}$, which has a positive definite submatrix $\widehat{W}^{\mathcal{R}M}_{[1:\mathcal{R}M,1:\mathcal{R}M]}$ and is zero elsewhere (as discussed in the proof of Corollary 5.11).

Remark 5.13 (a priori error estimate). Let $\boldsymbol{\alpha}^{approx}$ be the solution to (5.15). Then we get the a priori error estimate

$$B_\star - B_{\mathcal{R}}(\boldsymbol{\alpha}^{approx}) = \sum_{j=\mathcal{R}M+1}^{NM} \boldsymbol{\alpha}_{\star,j} B_j^{\mathcal{O}}.$$

Remark 5.14 (min-max problem). Following the same arguments of the proof of Lemma 5.8, one can show that the min-max problem (4.4) is equivalent to

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^{\mathcal{R}M}} \max_{\boldsymbol{\epsilon}\in E_{ad}} \|C\boldsymbol{y}_T(B_\star, \boldsymbol{\epsilon}) - C\boldsymbol{y}_T(B_{\mathcal{R}}(\boldsymbol{\alpha}), \boldsymbol{\epsilon})\|_2^2. \tag{5.16}$$

Analogously to Remark 5.7, we can conclude that, using the matrices $B_1^{\mathcal{O}}, \dots, B_{\mathcal{R}M}^{\mathcal{O}}$ defined in (5.13), problem (5.16) is uniquely solvable with $\boldsymbol{\alpha}_j = \boldsymbol{\alpha}_{\star,j}$, $j = 1, \dots, \mathcal{R}M$.

The results proved so far for a non-fully observable system are obtained for the special basis $(B_j)_{j=1}^{MN}$ constructed in (5.13). However, it is natural to ask the following questions:

- Is there any basis that permits us to reconstruct more than $\mathcal{R}M$ coefficients?
- Can one reconstruct at least $\mathcal{R}M$ coefficients for any arbitrarily chosen basis?

The answers to both questions are negative. The first one is given by Theorem 5.15.

THEOREM 5.15 (maximal number of identifiable elements). *Let the observability matrix* $\mathcal{O}_N(C,A)$ *be such that* $\operatorname{rank}\mathcal{O}_N(C,A) = \mathcal{R} < N$. *There exists no basis of* $\mathbb{R}^{N\times M}$ *for which one can exactly recover more than* $\mathcal{R}M$ *coefficients.*

Proof. Consider the basis $\mathcal{B} = \{B_k^{\mathcal{O}}\}_{k=1}^{NM} \subset \mathbb{R}^{N\times M}$ constructed as in (5.13) and another arbitrarily chosen basis $\widehat{\mathcal{B}} = \{\widehat{B}_k\}_{k=1}^{NM} \subset \mathbb{R}^{N\times M}$. Any element $\widehat{B} \in \widehat{\mathcal{B}}$ can be written as a linear combination of the elements of $\mathcal{B}$, that is, $\widehat{B} = \sum_{j=1}^{NM} \lambda_j B_j^{\mathcal{O}}$, for appropriate $\lambda_j \in \mathbb{R}$, $j = 1,\dots,MN$. Multiplying $\widehat{B}$ with $\mathcal{O}_N(C,A)$, we get

$$\mathcal{O}_N(C,A)\widehat{B} = \mathcal{O}_N(C,A)\left[\sum_{j=1}^{NM}\lambda_j B_j^{\mathcal{O}}\right] = \sum_{j=1}^{NM}\lambda_j \mathcal{O}_N(C,A)B_j^{\mathcal{O}} = \sum_{j=1}^{\mathcal{R}M}\lambda_j \mathcal{O}_N(C,A)B_j^{\mathcal{O}},$$

where we used that $\mathcal{O}_N(C,A)B_j^{\mathcal{O}} = 0$, for $j \in \{\mathcal{R}+1,\dots,N\}$, to obtain the last equality. Now define the set $\mathcal{D} = \{D_k\}_{k=1}^{NM}$ as $D_k := \mathcal{O}_N(C,A)\widehat{B}_k, \quad k = 1,\dots,NM$. Hence, we can conclude that at most $\mathcal{R}M$ elements of $\mathcal{D}$ are linearly independent. Recalling the proof of Lemma 5.5 and Remark 5.12, this means that for $NM - \mathcal{R}M$ elements of $\widehat{\mathcal{B}}$ there exists a linear combination of the other $\mathcal{R}M$ elements such that the observation at final time T is identical for any control $\boldsymbol{\epsilon}$. Therefore, one can reconstruct at most $\mathcal{R}M$ coefficients for the basis $\widehat{\mathcal{B}}$. ∎

Let us now explain why the answer to the second question is also negative. To do so, we provide the following examples, which show that a wrong choice of a basis leads to inconclusive results.

Example 5.16 (wrong bases lead to inconclusive results). Consider a simple system with

$$A = \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix},\ B_\star = \begin{bmatrix}1 & 1\\ 1 & 1\end{bmatrix},\ C = \begin{bmatrix}1 & 0\\ 0 & 0\end{bmatrix}$$

and the basis of $\mathbb{R}^{2\times 2}$

$$\widehat{B}_1 = \begin{bmatrix}1 & 0\\ 0 & 0\end{bmatrix}, \widehat{B}_2 = \begin{bmatrix}1 & 0\\ 1 & 0\end{bmatrix}, \widehat{B}_3 = \begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix}, \widehat{B}_4 = \begin{bmatrix}0 & 1\\ 0 & 1\end{bmatrix}.$$

Notice that in this case the observability condition does not hold, since one can compute that $\mathcal{R} = \operatorname{rank}\mathcal{O}_N(C,A) = \operatorname{rank}\left[\begin{smallmatrix}1&0&1&0\\0&0&0&0\end{smallmatrix}\right]^\top = 1$. Clearly we have that

$$B_\star = 0\cdot B_1 + 1\cdot B_2 + 0\cdot B_3 + 1\cdot B_4 \ \ (\text{hence } \boldsymbol{\alpha}_\star = [0\,1\,0\,1]^\top).$$

We can now compute for an arbitrarily chosen control $\boldsymbol{\epsilon} \in E_{ad}$ that

$$\begin{aligned}
C\boldsymbol{y}_T(B_\star,\boldsymbol{\epsilon}) - C\boldsymbol{y}_T(B(\boldsymbol{\alpha}),\boldsymbol{\epsilon}) &= C\int_0^T e^{(T-s)A}B_\star\boldsymbol{\epsilon}(s)ds - C\int_0^T e^{(T-s)A}B(\boldsymbol{\alpha})\boldsymbol{\epsilon}(s)ds\\
&= \int_0^T Ce^{(T-s)A}\left(\begin{bmatrix}1 & 1\\ 1 & 1\end{bmatrix} - \begin{bmatrix}\boldsymbol{\alpha}_1+\boldsymbol{\alpha}_2 & \boldsymbol{\alpha}_3+\boldsymbol{\alpha}_4\\ \boldsymbol{\alpha}_2 & \boldsymbol{\alpha}_4\end{bmatrix}\right)\boldsymbol{\epsilon}(s)ds\\
&= \int_0^T \begin{bmatrix}1 & 0\\ 0 & 0\end{bmatrix}\begin{bmatrix}e^{T-s} & 0\\ 0 & e^{T-s}\end{bmatrix}\begin{bmatrix}1-(\boldsymbol{\alpha}_1+\boldsymbol{\alpha}_2) & 1-(\boldsymbol{\alpha}_3+\boldsymbol{\alpha}_4)\\ 1-\boldsymbol{\alpha}_2 & 1-\boldsymbol{\alpha}_4\end{bmatrix}\boldsymbol{\epsilon}(s)ds\\
&= \int_0^T \begin{bmatrix}e^{T-s}(1-(\boldsymbol{\alpha}_1+\boldsymbol{\alpha}_2)) & e^{T-s}(1-(\boldsymbol{\alpha}_3+\boldsymbol{\alpha}_4))\\ 0 & 0\end{bmatrix}\boldsymbol{\epsilon}(s)ds,
\end{aligned}$$

which is zero for any $\boldsymbol{\alpha} = [\boldsymbol{\alpha}_1\ \boldsymbol{\alpha}_2\ \boldsymbol{\alpha}_3\ \boldsymbol{\alpha}_4]^\top \in \mathbb{R}^4$ with $\boldsymbol{\alpha}_1 + \boldsymbol{\alpha}_2 = 1$ and $\boldsymbol{\alpha}_3 + \boldsymbol{\alpha}_4 = 1$ (for any control $\boldsymbol{\epsilon}$). This means that any $\boldsymbol{\alpha} = [\boldsymbol{\alpha}_1\ \boldsymbol{\alpha}_2\ \boldsymbol{\alpha}_3\ \boldsymbol{\alpha}_4]$ with $\boldsymbol{\alpha}_1 + \boldsymbol{\alpha}_2 = 1$ and $\boldsymbol{\alpha}_3 + \boldsymbol{\alpha}_4 = 1$ solves the least-squares problem (4.3), independently on the control functions $\boldsymbol{\epsilon}_1, \dots, \boldsymbol{\epsilon}_4$. Since the online least-squares problem has then infinitely many solutions,[4] one cannot conclude anything about the quality of a computed solution, which has the form $\widehat{B}^{approx} = \left[\begin{smallmatrix} 1 & 1 \\ \boldsymbol{\alpha}_2 & \boldsymbol{\alpha}_4 \end{smallmatrix}\right]$, leading to the error

$$\|B_\star - B_{\mathcal{R}}(\boldsymbol{\alpha}^{approx})\|_F^2 = (1-\boldsymbol{\alpha}_2)^2 + (1-\boldsymbol{\alpha}_4)^2,$$

which can be arbitrarily large (here $\|\cdot\|_F$ denotes the Frobenius norm). Even if one would by chance guess the right coefficients (in this case $\boldsymbol{\alpha}_2 = 1, \boldsymbol{\alpha}_4 = 1$), there would be no way to verify them, since their effect is not observable. Notice also that even if the entries $\widehat{B}_{1,1}^{approx}$ and $\widehat{B}_{1,2}^{approx}$ are correct, it is not possible to certify this or to associate these correct entries to some precise elements of the chosen basis. This example shows that for an arbitrarily chosen basis, one cannot conclude anything about the quality of the computed coefficients or the difference between $B(\boldsymbol{\alpha})$ and $B_\star$.

Example 5.17 (good bases lead to certified results). Consider the same system of Example 5.16, but now let us use the SVD of the observability matrix,

$$\mathcal{O}_2(C,A) = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \frac{\sqrt{2}}{2} & 0 & -\frac{\sqrt{2}}{2} & 0 \\ 0 & 1 & 0 & 0 \\ \frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \sqrt{2} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = U\Sigma V^\top,$$

which gives $\boldsymbol{v}_1 = \left[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right] \notin \ker \mathcal{O}_N(C,A)$, $\boldsymbol{v}_2 = \left[\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}\right] \in \ker \mathcal{O}_N(C,A)$, leading to the basis

$$B_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, B_2 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, B_3 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, B_4 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix},$$

constructed as in (5.13). In this case, we have $\boldsymbol{\alpha}_\star = [1\ 1\ 1\ 1]^\top$. Since the GR algorithm considers only the first two basis elements, one gets the final result $B^{approx} = \left[\begin{smallmatrix} 1 & 1 \\ 0 & 0 \end{smallmatrix}\right]$. Similarly to Example 5.16, the two entries $\widehat{B}_{1,1}^{approx}$ and $\widehat{B}_{1,2}^{approx}$ are correct, but now this is guaranteed by Theorem 5.10. Therefore, in this case, the results obtained are accompanied by precise information on their correctness.

These examples show clearly that without an a priori knowledge about the observability of the system (and hence about the "quality" of the basis), the GR algorithm leads to inconclusive results. Even though we have presented in this section a way to construct a basis which permits a precise analysis of the obtained results, this is generally not possible for nonlinear problems, like the Hamiltonian reconstruction problem described in section 3. Is it then possible to modify the GR algorithm in order to distinguish automatically between "good" and "bad" elements of a given set of matrices? The answer is given in section 6, where we first introduce an improved GR algorithm for linear-quadratic problems and then extend it to nonlinear problems.

6. Improvements of the algorithm. The previous section ended with two examples showing clearly that a wrong choice of the basis elements and their ordering can lead to inconclusive results. Even though this issue can be avoided for linear problems by using the observability matrix (and constructing a basis as in (5.13)), this strategy does generally not apply to nonlinear problems. For this reason, we introduce

[4] Notice that these solutions are also solutions to the min-max problem (4.4).

an OGR algorithm, in which the basis elements are selected during the iterations (in a greedy fashion) as the ones that maximize the discrimination functions. In particular, we introduce in section 6.1 the OGR algorithm for linear-quadratic problems and show by numerical experiments that this leads to an automatic appropriate selection of the basis elements, even though the observability matrix is not considered at all. Once the new algorithm is introduced for linear systems, it is then natural to extend it to nonlinear problems. We consider this extension in section 6.2 for Hamiltonian reconstruction problems and show the efficiency of our new OGR algorithm by direct numerical experiments.

6.1. OGR for linear-quadratic problems. Consider an arbitrary set of linearly independent matrices $(B_j)_{j=1}^K \subset \mathbb{R}^{N\times M}$. We wish to modify the GR Algorithm 4.1 in order to choose at every iteration one element B_j which leads to a control function capable of improving the rank of the matrix $\widehat{W}^k_{[1:k+1,1:k+1]}$. The idea is to replace the sweeping process of the GR Algorithm 4.1 with a more robust and parallel testing of all the matrices. At each iteration, the element associated with the maximal discriminating value is chosen and removed from the set $(B_j)_{j=1}^K$, while the corresponding control function is added to the set of already computed control functions. Therefore, the dimension of the set $(B_j)_{j=1}^K$ reduces by one at each iteration, and the algorithm is stopped either if all the K matrices are chosen or as soon none of the remaining ones can be discriminated by the others. This idea leads to the OGR Algorithm 6.1.

In this algorithm, we clearly extended the greedy character of the original GR algorithm to the choice of the next basis element. At each iteration, we consider all remaining basis elements as the potential next one. We select the one which yields the largest function value in the respective discrimination (maximization) step. In other words, one computes the basis element for which one can split the observation the most from all previous basis elements. It is important to remark that at each iteration one solves several fitting-step problems and several discriminatory-step problems. However, their solving can be performed in parallel, since the single problems are independent one from another.

Notice that a selected element B_{k+1} will not be linearly dependent on previously chosen elements (after multiplication with the observability matrix). This is proven in the next theorem, which also motivates the stopping criterion used in steps 2–4 and 11–13 of the algorithm.

THEOREM 6.1 (linearly independence of selected basis elements). *Assume that the OGR Algorithm* 6.1 *selected already k linearly independent matrices B_j, $j = 1,\dots,k$. At iteration $k+1$, the new selected matrix B_{k+1} is such that $\mathcal{O}_N(C,A)B_{k+1}$ is linearly independent from the matrices $\mathcal{O}_N(C,A)B_j$, $j = 1,\dots,k$, if and only if*

$$\left\| C\boldsymbol{y}_T(B_{\ell_{k+1}},\boldsymbol{\epsilon}^{k+1}) - C\boldsymbol{y}_T(B^{(k)}(\boldsymbol{\alpha}^{\ell_k}),\boldsymbol{\epsilon}^{k+1}) \right\|_2^2 > 0.$$

Proof. If the matrix $\mathcal{O}_N(C,A)B_{k+1}$ is linearly independent from the other matrices $\mathcal{O}_N(C,A)B_j$, $j = 1,\dots,k$, then one can show as in the proof of Lemma 5.9 that

$$\left\| C\boldsymbol{y}_T(B_{\ell_{k+1}},\boldsymbol{\epsilon}^{k+1}) - C\boldsymbol{y}_T(B^{(k)}(\boldsymbol{\alpha}^{\ell_k}),\boldsymbol{\epsilon}^{k+1}) \right\|_2^2 > 0.$$

Now, we prove the other implication by contraposition. Assume that there exists a vector $\boldsymbol{\alpha} \in \mathbb{R}^k$ such that $\mathcal{O}_N(C,A)(B_{k+1} - \sum_{j=1}^k \boldsymbol{\alpha}_j B_j) = 0$ holds. This vector $\boldsymbol{\alpha}$ is a solution of the fitting-step problem with cost-function value equal to zero. However,

the corresponding cost function of the discriminatory-step problem (6.2) is zero for any control function $\boldsymbol{\epsilon}$. The result follows by contraposition. ∎

Notice that if Algorithm 6.1 stops at step 3, then the chosen basis does not allow one to distinguish the states corresponding to controlled and uncontrolled systems. In this case, entering in the while loop would be useless since the first discriminatory step will certainly fail in producing a large enough discrimination value and the algorithm will terminate at steps 11 and 12.

Theorem 6.1 shows exactly that the OGR algorithm manages to identify among the elements of the given set $(B_j)_{j=1}^K$ the ones that do not lie in the kernel of $\mathcal{O}_N(C, A)$. For instance, let us consider again the system of Example 5.16, for which we have shown that the GR algorithm leads to inconclusive results. If we use instead the OGR Algorithm 6.1, this performs two iterations and selects only two basis elements, one among $\widehat{B}_1$ and $\widehat{B}_2$ and the other among $\widehat{B}_3$ and $\widehat{B}_4$. This can be shown by performing calculations similar to the ones of Example 5.16. In particular, in the initialization step the four matrices produce the same cost-function value. Hence, any of them can be selected by the algorithm. Assume that the element $\widehat{B}_1$ is picked (hence $\ell_1 = 1$), and consider the first iteration of the algorithm ($k = 1$). At the fitting step the algorithm computes a coefficient $\boldsymbol{\alpha}_1^2 = 1$ for $\widehat{B}_2$ and some coefficients $\boldsymbol{\alpha}_1^3$ and $\boldsymbol{\alpha}_1^4$ corresponding to $\widehat{B}_3$ and $\widehat{B}_4$. Now, $\boldsymbol{\alpha}_1^2 = 1$ leads to a cost function of the discriminatory step which is zero for any control functions, while for $\boldsymbol{\alpha}_1^3$ and $\boldsymbol{\alpha}_1^4$

Algorithm 6.1. Optimized Greedy Reconstruction Algorithm (linear-quadratic case)

Require: A set of K linearly independent matrices $\mathcal{B} = (B_1, \dots, B_K)$ and a tolerance $\mathrm{tol} > 0$.

Solve the initialization problem

$$\max_{\ell \in \{1,\dots,K\}} \max_{\boldsymbol{\epsilon} \in E_{ad}} \|C\boldsymbol{y}_T(B_\ell, \boldsymbol{\epsilon}) - C\boldsymbol{y}_T(0,0)\|_2^2 ,$$

which gives the field $\boldsymbol{\epsilon}^1$ and the index ℓ_1.
if $\left\|C\boldsymbol{y}_T(B_{\ell_1}, \boldsymbol{\epsilon}^1) - C\boldsymbol{y}_T(0,0)\right\|_2^2 < \mathrm{tol}$ **then**
  stop and display "Error: all basis elements have no observable effect."
end if
Swap B_1 and B_{ℓ_1} in $\mathcal{B}$ and set $k = 1$.
while $k \le K - 1$ **do**
  for $\ell = k+1, \dots, K$ **do**
    Fitting step: Find $(\boldsymbol{\alpha}_j^\ell)_{j=1,\dots,k}$ that solve the problem

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^k} \sum_{m=1}^{k} \left\| C\boldsymbol{y}_T(B_\ell, \boldsymbol{\epsilon}^m) - C\boldsymbol{y}_T(B^{(k)}(\boldsymbol{\alpha}), \boldsymbol{\epsilon}^m) \right\|_2^2 . \tag{6.1}$$

  end for
  Extended discriminatory step: Find $\boldsymbol{\epsilon}^{k+1}$ and ℓ_{k+1} that solve the problem

$$\max_{\ell \in \{k+1,\dots,K\}} \max_{\boldsymbol{\epsilon} \in E_{ad}} \left\| C\boldsymbol{y}_T(B_\ell, \boldsymbol{\epsilon}) - C\boldsymbol{y}_T(B^{(k)}(\boldsymbol{\alpha}^\ell), \boldsymbol{\epsilon}) \right\|_2^2 . \tag{6.2}$$

  if $\left\| C\boldsymbol{y}_T(B_{\ell_{k+1}}, \boldsymbol{\epsilon}^{k+1}) - C\boldsymbol{y}_T(B^{(k)}(\boldsymbol{\alpha}^{\ell_k}), \boldsymbol{\epsilon}^{k+1}) \right\|_2^2 < \mathrm{tol}$ **then**
    stop and return the selected $(B_j)_{j=1}^k$ and the computed $(\boldsymbol{\epsilon}^m)_{m=1}^k$.
  end if
  Swap B_{k+1} and $B_{\ell_{k+1}}$ in $\mathcal{B}$ and update $k \leftarrow k+1$.
end while

there exists a control function leading to a nonzero value of the discriminatory cost. Therefore, the algorithm selects either $\widehat{B}_3$ or $\widehat{B}_4$. Let us assume that $\widehat{B}_4$ is picked ($\ell_2 = 4$) and hence the two elements $\widehat{B}_2$ and $\widehat{B}_4$ are swapped. In the fitting step of the second iteration ($k = 2$), the algorithm computes $\boldsymbol{\alpha}^3 = [\,0\,,\,1\,]^\top$ and $\boldsymbol{\alpha}^4 = [\,1\,,\,0\,]^\top$. Both of these two vectors lead to a discriminatory cost that is zero for any control. Hence, since the discriminatory step does not find any positive function value, the algorithm stops and returns $\widehat{B}_{\ell_1} = \widehat{B}_1$ and $\widehat{B}_{\ell_2} = \widehat{B}_4$ and the corresponding controls. If one uses the two selected basis elements and the corresponding control functions in the online phase, then one obtains the result $\boldsymbol{\alpha} = [\,1\,,\,1\,]^\top$, which is not the exact solution shown in Example 5.16. This is due to the non-full observability of the system, which implies that $\mathcal{O}_N(C,A)\widehat{B}_1 = \mathcal{O}_N(C,A)\widehat{B}_2$ and $\mathcal{O}_N(C,A)\widehat{B}_3 = \mathcal{O}_N(C,A)\widehat{B}_4$. This means that the observations generated by the elements $\widehat{B}_1$ and $\widehat{B}_3$ cannot be distinguished by the ones created by $\widehat{B}_2$ and $\widehat{B}_4$. The non-full observability of the system cannot be overcome by any numerical strategy. The OGR algorithm can nevertheless identify automatically all the observable degrees of freedom of the considered system.

Let us now demonstrate the efficiency of our new OGR algorithm by direct numerical experiments. We consider an experiment with two randomly chosen $N \times N$ full-rank real matrices A and C with $N = 10$. The unknown $B_\star$ is a randomly chosen real $N \times N$ matrix. In this case the system is fully observable; nevertheless, we construct the basis elements to be used in the GR and OGR algorithm as in (5.13) (by an SVD of the observability matrix), but we order the elements randomly. We then run the GR Algorithm 4.1 and compute the rank of the matrix $\widehat{W}^k$ at every iteration k. This leads to the results shown in Figure 6.1 by the blue curve. The rank increases monotonically during the iterations and becomes full after about 30 iterations. However, the curve is not strictly monotonically increasing since the rank does not increase at each iteration. If we repeat the same experiment (with the same matrices) using the OGR Algorithm 6.1, we obtain the red curve in Figure 6.1. This curve is strictly monotonically increasing in the first part and becomes constant only once the rank has become full. In particular, at each iteration the rank increases by 10, and the OGR algorithm could be in principle stopped much earlier than the original GR algorithm, and many fewer control functions (hence laboratory experiments) are needed to fully reconstruct the unknown operator $B_\star$. This experiment clearly shows the high potential of the OGR algorithm, which is capable of choosing among the elements $B_1, \ldots, B_K$ in an optimized fashion.

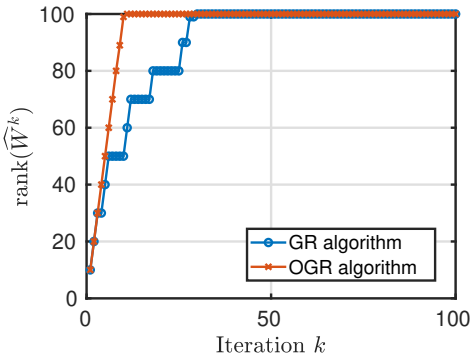

FIG. 6.1. *Rank of the matrix $\widehat{W}^k$ corresponding to the GR algorithm (blue curve) and OGR algorithm (red curve) for a fully observable system. Both algorithms make use of a basis constructed as in* (5.13).

Let us conclude this section with two important observations. First, the improvement proposed in Algorithm 6.1 allows one to even enrich the set $(B_j)_{j=1}^K$ used as input in Algorithm 6.1 with other new elements that can be linearly dependent on $B_1, \dots, B_K$. In this case, if we denote by $(B_j)_{j=1}^{\widetilde{K}}$ for $\widetilde{K} > K$ the enriched set, then Theorem 6.1 guarantees that the OGR algorithm will automatically pick some elements of the enriched set $(B_j)_{j=1}^{\widetilde{K}}$ such that $\mathcal{O}_N(C, A)B_j$ are linearly independent for all selected B_j. Hence, the corresponding discriminatory cost-function values will be strictly positive. Second, the OGR algorithm can be extended to more general nonlinear reconstruction problems, and we propose in section 6.2 an efficient extension for the Hamiltonian reconstruction problem described in section 3.

6.2. OGR for nonlinear problems. The extension of the OGR Algorithm 6.1 to the nonlinear Hamiltonian reconstruction problem of section 3 is formally rather straightforward and given by Algorithm 6.2. However, there is one key addition represented by steps 7, 8, and 9. In these steps, each of the matrices B_ℓ, $\ell = k+1, \dots, K$ (that have not been selected in the first k iterations of the algorithm) is orthogonalized with respect to the already selected matrices B_ℓ, $\ell = 1, \dots, k$. This can be achieved by a single Gram–Schmidt step for each B_ℓ, $\ell = k+1, \dots, K$. The

Algorithm 6.2. Optimized Greedy Reconstruction Algorithm (Hamiltonian case)

Require: A set of K matrices $\mathcal{B}_\mu = (\mu_\ell)_{\ell=1,\dots,K}$ and a tolerance tol > 0.

1: Solve the initialization problem

$$\max_{n\in\{1,\dots,K\}} \max_{\epsilon\in L^2} |\varphi(\mu_n, \epsilon) - \varphi(0,0)|^2, \tag{6.3}$$

which gives the field ϵ^1 and the index ℓ_1.

2: **if** $|\varphi(\mu_{\ell_1}, \epsilon^1) - \varphi(0,0)|^2 <$ tol **then**

3: **stop** and display "Error: all basis elements have no observable effect."

4: **end if**

5: Swap μ_1 and μ_{ℓ_1} in $\mathcal{B}_\mu$ and set $k = 1$ and $\widetilde{K} = K$.

6: **while** $k \le K - 1$ **do**

7: Orthogonalize each matrix μ_ℓ, $\ell = k+1, \dots, \widetilde{K}$, with respect to the set $(\mu_1, \dots, \mu_k)$.

8: Remove the zero elements from $\mathcal{B}_\mu$ and shift the indices of the remaining elements.

9: Update $\widetilde{K} \leftarrow \operatorname{card} \mathcal{B}_\mu$.

10: **for** $\ell = k+1, \dots, \widetilde{K}$ **do**

11: Fitting step: Find $(\boldsymbol{\alpha}_j^\ell)_{j=1,\dots,k}$ that solve the problem

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^k} \sum_{m=1}^{k} |\varphi(\mu_\ell, \epsilon^m) - \varphi(\mu^{(k)}(\boldsymbol{\alpha}), \epsilon^m)|^2. \tag{6.4}$$

12: **end for**

13: Extended discriminatory step: Find ϵ^{k+1} and ℓ_{k+1} that solve the problem

$$\max_{\ell\in\{k+1,\dots,\widetilde{K}\}} \max_{\epsilon\in L^2} |\varphi(\mu_\ell, \epsilon) - \varphi(\mu^{(k)}(\boldsymbol{\alpha}^\ell), \epsilon)|^2. \tag{6.5}$$

14: **if** $|\varphi(\mu_{\ell_{k+1}}, \epsilon^{k+1}) - \varphi(\mu^{(k)}(\boldsymbol{\alpha}^{\ell_k}), \epsilon^{k+1})|^2 <$ tol **then**

15: **stop** and return the selected $(\mu_j)_{j=1}^k$ and the computed $(\epsilon^m)_{m=1}^k$.

16: **end if**

17: Swap μ_{k+1} and $\mu_{\ell_{k+1}}$ in $\mathcal{B}_\mu$ and update $k \leftarrow k+1$.

18: **end while**

orthogonalization is required to avoid that the algorithms picks a new matrix B_{k+1} such that either the angle between B_{k+1} and $(B_1, \dots, B_k)$ is very small or (in the worst case) B_{k+1} is linearly dependent from $(B_1, \dots, B_k)$. These two situations could lead to numerical problems in the final online identification phase. Moreover, by eliminating linearly dependent elements, one avoids the solves of several unnecessary fitting and discriminatory problems (even though solvable in parallel).

A few more computational aspects must be discussed. First, the maximization problems characterizing the initialization step and the discriminatory steps are nonlinear optimal control problems that we solve numerically by the monotonic scheme discussed in [15], in the setting described in [14]; see also [5, 14, 15, 17] and references therein. Second, the fitting-step problems are highly nonlinear minimization problems having generally several local minima. Since not all local minima correspond to an effective defect (rank deficiency in the linear-quadratic case) to be compensated, every fitting-step problem is solved multiple times using different randomly chosen initializations. The solution corresponding to the smallest functional value is then chosen. Each fitting-step problem is solved by a BFGS descent-direction method. Third, all optimization problems that are solved in the fitting steps and in the discriminatory steps are independent one from another. Therefore, they can be solved in parallel as in the linear case.

Let us now show the efficiency of the OGR Algorithm 6.2 by direct numerical experiments. We consider the same test case as in [14], where the unknown Hamiltonian and the controlled Hamiltonian μ are assumed to be real-symmetric. More precisely, the matrix H and the randomly generated $\mu_\star$ are

$$H = 10^{-2}\begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 4 \end{bmatrix}, \ \mu_\star = \begin{bmatrix} 3.3617 & 3.4347 & 0.8416 \\ 3.4347 & 3.7763 & 4.7552 \\ 0.8416 & 4.7552 & 4.4226 \end{bmatrix}.$$

The final time is $T = 4000\pi$. The states ψ_0 and ψ_1 are

$$\psi_0 = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^\top, \quad \psi_1 = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^\top.$$

Now, we perform the following experiment. Since the unknown $\mu_\star$ is a 3×3 symmetric matrix, we choose for the set $\mathcal{B}_\mu$ the following $K = 6$ linearly independent canonical matrices:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \tag{6.6}$$

which form a basis for the space of 3×3 symmetric matrices with real entries, and compute 6 control functions by the OGR Algorithm 6.2. Once these functions are obtained, one must reconstruct the unknown true dipole matrix by solving the online nonlinear least-squares problem (3.3). To do so, we use the standard MATLAB function `fminunc` (a BFGS descent-direction minimization algorithm) initialized by a randomly chosen vector. To test the robustness of the control functions computed by the OGR Algorithm 6.2, we consider a six-dimensional hypercube centered in the global minimum point $\mu_\star$ and given relative radius r and repeat the minimization for 1000 initialization vectors randomly chosen in this hypercube. We then count the number of times that the optimization algorithm converges to the global solution $\mu_\star = \mu(\boldsymbol{\alpha}_\star)$ up to a tolerance of $Tol = 0.005$ (half of the smallest considered radius), meaning that $\frac{\|\mu_\star - \mu(\boldsymbol{\alpha}^0)\|_F}{\|\mu_\star\|_F} \leq Tol$, where $\|\cdot\|_F$ denotes the Frobenius norm. Repeating this experiment for different values of the radius r of the hypercube, we obtain the results reported in the first row of Table 6.1.

TABLE 6.1
Numbers of runs (over 1000*) that converged to the true solution* $\mu_\star$.

Hypercube radius r	0.01	0.10	0.50	1.00
GR (canonical basis)	814	0	0	0
GR (random basis)	655	3	0	0
OGR (extended random basis)	1000	134	15	5

These results show clearly the lack of robustness of the controls generated by the GR algorithm: for the very small radius $r = 0.01$ of the hypercube, the minimization converged to the true solution for only 80% of the cases over the 1000 runs, and for $r > 0.01$ in none of the cases the minimization converged to the solution.

Next, to test the effect of the chosen basis $\mathcal{B}_\mu$, we repeat the same experiment using 6 randomly chosen linearly independent symmetric matrices μ_ℓ, $\ell = 1, \dots, 6$. The obtained results of this second test are shown in the second row of Table 6.1. These are clearly worse and very unsatisfactory.

Finally, we repeat the experiment using the OGR Algorithm 6.2 with a set of 12 matrices, namely, the 6 unit basis elements shown above and the 6 linearly independent random matrices chosen for the second experiment. We obtain the results shown in the third row of Table 6.1. These are much better results. For $r = 0.01$ all the 1000 runs converged to the solution $\mu_\star$. Even though the number of times that the optimization algorithm converged to the true solution decays as the radius r increases, in the case $r = 0.10$ more than 100 of runs converged to $\mu_\star$. These results show the improved efficiency of the new proposed OGR algorithm. This improvement is even more evident if we consider a more general example where the unknown matrix lies in $\mathrm{Her}(N)$. In this case, the canonical basis for $\mathrm{Her}(3)$ is composed of the matrices given (6.6) together with the three matrices

$$\begin{bmatrix} 0 & -i & 0 \\ i & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & -i \\ 0 & 0 & 0 \\ i & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -i \\ 0 & i & 0 \end{bmatrix}. \tag{6.7}$$

Let us now consider two examples. First, we choose an observer vector $\psi_1 = [0\,0\,1]^\top$ and a (randomly generated) matrix $\mu_\star$ given by

$$\mu_\star = \begin{bmatrix} -0.3243 & -3.4790 + 0.7359i & -0.5338 + 1.9254i \\ -3.4790 - 0.7359i & -3.8342 & -1.1697 + 2.0256i \\ -0.5338 - 1.9254i & -1.1697 - 2.0256i & 1.0551 \end{bmatrix}.$$

All the other data (namely, T, H, and ψ_0)[5] are the same as the ones considered in the real-symmetric example. If we repeat the experiments of the real-symmetric case, we obtain the results of Table 6.2.

TABLE 6.2
Numbers of runs (over 1000*) that converged to the true solution* $\mu_\star$.

Hypercube radius r	0.01	0.10	0.50	1.00
GR (canonical basis)	908	13	1	0
GR (random basis)	596	4	0	0
OGR (extended random basis)	1000	277	32	7

[5]Notice that one can always consider a (real) diagonal form of H. In fact, for any $H \in \mathrm{Her}(N)$ one can always diagonalize it by a classical Schur decomposition and change the variable of the system by using the orthogonal matrix of the eigenvectors of H.

TABLE 6.3
Numbers of runs (over 1000*) that converged to the true solution* $\mu_\star$.

Hypercube radius r	0.01	0.10	0.50	1.00
GR (canonical basis)	1000	757	15	2
GR (random basis)	648	212	49	3
OGR (extended random basis)	1000	992	214	36

If one repeats the experiments for a different observer vector $\psi_1 = \frac{1}{\sqrt{3}}[1\,1\,1]^\top$, the results shown in Table 6.3 are obtained.
Table 6.2 and Table 6.3 show very clearly the improved efficiency and robustness of control functions generated by the OGR algorithm. These allow one to identify the solution $\mu_\star$ in a much larger number of statistical runs.

7. Conclusions. In this work, we provided a novel and detailed convergence analysis for the GR algorithm introduced in [14] for Hamiltonian reconstruction problems in the field of quantum mechanics. The presented convergence analysis has considered linear-quadratic (optimization, least-squares) problems and revealed the strong dependence of the performance of the GR algorithm on the observability properties of the system and on the ansatz of the basis elements used to reconstruct the unknown operator. This allowed us to introduce a precise (and in some sense optimal) choice of the basis elements for the linear case and led to the introduction of an OGR algorithm applicable also to the nonlinear Hamiltonian reconstruction problem. Numerical experiments demonstrated the efficiency of the new proposed numerical algorithm.

REFERENCES

[1] M. BARRAULT, Y. MADAY, N. C. NGUYEN, AND A. T. PATERA, *An "empirical interpolation" method: Application to efficient reduced-basis discretization of partial differential equations*, C. R. Math. Acad. Sci. Paris, 339 (2004), pp. 667–672.
[2] L. BAUDOUIN AND A. MERCADO, *An inverse problem for Schrödinger equations with discontinuous main coefficient*, Appl. Anal., 87 (2008), pp. 1145–1165.
[3] K. BEAUCHARD AND C. LAURENT, *Local controllability of* 1*D linear and nonlinear Schrödinger equations with bilinear control*, J. Math. Pures Appl., 94 (2010), pp. 520–554.
[4] S. BONNABEL, M. MIRRAHIMI, AND P. ROUCHON, *Observer-based Hamiltonian identification for quantum systems*, Automatica, 45 (2009), pp. 1144–1155.
[5] A. BORZÌ, G. CIARAMELLA, AND M. SPRENGEL, *Formulation and Numerical Solution of Quantum Control Problems*, SIAM, Philadelphia, PA, 2017.
[6] J. CORON, *Control and Nonlinearity*, Math. Surveys Monogr., American Mathematical Society, Providence, RI, 2007.
[7] A. DONOVAN AND H. RABITZ, *Exploring the Hamiltonian inversion landscape*, Phys. Chem., 16 (2014), pp. 15615–15622.
[8] Y. FU AND G. TURINICI, *Quantum Hamiltonian and dipole moment identification in presence of large control perturbations*, ESAIM Control Optim. Calc. Var., 23 (2017), pp. 1129–1143.
[9] J. M. GEREMIA AND H. RABITZ, *Global, nonlinear algorithm for inverting quantum-mechanical observations*, Phys. Rev. A, 64 (2001), 022710.
[10] J. M. GEREMIA AND H. RABITZ, *Optimal Hamiltonian identification: The synthesis of quantum optimal control and quantum inversion*, J. Chem. Phys., 118 (2003), pp. 5369–5382.
[11] J. M. GEREMIA, W. ZHU, AND H. RABITZ, *Incorporating physical implementation concerns into closed loop quantum control experiments*, J. Chem. Phys., 113 (2000), pp. 10841–10848.
[12] R. A. HORN AND C. R. JOHNSON, *Matrix Analysis*, 2nd ed., Cambridge University Press, Cambridge, UK, 2012.
[13] C. LE BRIS, M. MIRRAHIMI, H. RABITZ, AND G. TURINICI, *Hamiltonian identification for quantum systems: Well posedness and numerical approaches*, ESAIM Control Optim. Calc. Var., 13 (2007), pp. 378–395.

[14] Y. Maday and J. Salomon, *A greedy algorithm for the identification of quantum systems*, in Proceedings of the 48th IEEE Conference on Decision and Control, held jointly with the 28th Chinese Control Conference, 2009, pp. 375–379.
[15] Y. Maday, J. Salomon, and G. Turinici, *Monotonic time-discretized schemes in quantum control*, Numer. Math., 103 (2006), pp. 323–338.
[16] W. Rudin, *Real and Complex Analysis*, 3rd ed., McGraw-Hill, New York, 1987.
[17] J. Salomon, *Convergence of the time-discretized monotonic schemes*, ESAIM Math. Model. Numer. Anal., 41 (2007), pp. 77–93.
[18] E. D. Sontag, *Mathematical Control Theory: Deterministic Finite Dimensional Systems*, 2nd ed., Springer-Verlag, Berlin, 1998.
[19] M. Tadi and H. Rabitz, *Explicit method for parameter identification*, J. Guidance Control Dyn., 20 (1997), pp. 486–491.
[20] L. N. Trefethen and D. Bau, *Numerical Linear Algebra*, SIAM, Philadelphia, PA, 1997.
[21] Y. Wang, D. Dong, B. Qi, J. Zhang, I. R. Petersen, and H. Yonezawa, *A quantum Hamiltonian identification algorithm: Computational complexity and error analysis*, IEEE Trans. Automat. Control, 63 (2018), pp. 1388–1403.
[22] S. Xue, R. Wu, D. Li, and M. Jiang, *A gradient algorithm for Hamiltonian identification of open quantum systems*, Phys. Rev. A, 103 (2021), 022604.
[23] J. Zhang and M. Sarovar, *Quantum Hamiltonian identification from measurement time traces*, Phys. Rev. Lett., 113 (2014), 080401.
[24] W. Zhou, S. Schirmer, E. Gong, H. Xie, and M. Zhang, *Identification of Markovian open system dynamics for qubit systems*, Chinese Sci. Bull., 57 (2012), pp. 2242–2246.
[25] W. Zhu and H. Rabitz, *Potential surfaces from the inversion of time dependent probability density data*, J. Chem. Phys., 111 (1999), pp. 472–480.