Online greedy identification of linear dynamical systems

Matthieu Blanke and Marc Lelarge

***Abstract*—This work addresses the problem of exploration in an unknown environment. For linear dynamical systems, we use an experimental design framework and introduce an online greedy policy where the control maximizes the information of the next step. In a setting with a limited number of experimental trials, our algorithm has low complexity and shows experimentally competitive performances compared to more elaborate gradient-based methods.** [1]

I. Introduction

System identification is a problem of great interest in many fields such as econometrics, robotics, aeronautics, mechanical engineering or reinforcement learning [**?**], [**?**], [**?**], [**?**], [**?**]. The task consists in estimating the parameters of an unknown system by sampling trajectories from it as fast as possible. To this end, inputs must be chosen so as to yield maximally informative trajectories. We focus on linear time-invariant (LTI) systems. Let $A \in \mathbb{R}^{d\times d}$ and $B \in \mathbb{R}^{d\times m}$ be two matrices; we consider the following discrete-time dynamics:

$$\begin{aligned} x_0 &= 0, \\ x_{t+1} &= Ax_t + Bu_t + w_t, \quad 0 \leq t \leq T-1 \end{aligned} \tag{1}$$

where $x_t \in \mathbb{R}^d$ is the state, $w_t \sim \mathcal{N}(0, \sigma^2 I)$ is a normally distributed isotropic noise with known variance σ^2 and the control variables $u_t \in \mathbb{R}^m$ are chosen by the controller with the following power constraint:

$$\frac{1}{T}\sum_{t=0}^{T-1} \|u_t\|^2 \leq \gamma^2. \tag{2}$$

The system parameters $(A\,B) := \theta \in \mathbb{R}^{d\times q}$ $(q = d+m)$ are unknown initially and are to be estimated from observed trajectories (x_t). The goal of system identification is to choose the inputs u_t so as to drive the system to the most informative states for the estimation of θ. It may happen that the controller knows B, in which case $\theta = A$ and $q = m$.

System identification is a primary field in control theory. It has been widely studied in the field of optimal design of experiments [**?**], [**?**]. For LTI dynamic systems, classical optimal design approaches provided results for single-input single-output (SISO) systems [**?**], [**?**] or multi-input multi-output (MIMO) systems in the frequency domain or with randomized time-domain inputs [**?**]. More recently, system identification received considerable attention in the machine learning community, with the aim of obtaining finite-time bounds on the estimation error for A [**?**], [**?**], [**?**]. In [**?**] and [**?**], the inputs are optimized in the frequency domain to maximize an optimal design objective, with theoretical estimation rate guarantees. In our approach, we directly optimize deterministic inputs in the time domain for MIMO LTI systems. An important aspect of system identification is the quantity of computational resource and the number of observations needed to reach a certain performance. We study the computational complexity of our algorithms and compare their performance against each other and against an oracle, both on average and on real-life dynamic systems.

A. Notations

In the rest of this work, we note $\theta_\star = (A_\star B_\star)$ the unknown parameter underlying the dynamics. We suppose that the pair $(A_\star, B_\star)$ is controllable: the matrix $C_\star = (B_\star\, A_\star B_\star \,\dots\, A_\star^{d-1}B_\star)$ has rank d. Adopting the notations of [**?**], we define a policy $\pi : (x_{1:t}, u_{0:t-1}) \to u_t$ as a mapping from the past trajectory to future input. The set of policies meeting the power constraint (**??**) is denoted Π_γ. We note $\tau = (x_{1:T}, u_{0:T-1})$ a trajectory, and we extend this notation to $\tau(\pi, T)$ when the trajectory is obtained using a policy π up to time T. We denote by $\mathbb{E}_\theta$ the average for a dynamical system given by (**??**) (where the randomness comes from the noise w_t and possibly from the policy inducing the control u_t).

B. Adaptive identification

Fix an estimator $\hat{\theta} : \tau \mapsto \hat{\theta}(\tau) \in \mathbb{R}^{d\times q}$, yielding an estimate of the parameters from a given trajectory. Our objective is to play the inputs u_t of a policy $\pi \in \Pi_\gamma$ so that the resulting trajectory τ gives a good estimation $\hat{\theta}(\tau)$ for $\theta_\star$. We measure this performance by the mean squared error:

$$\mathrm{MSE}(\pi) = \frac{1}{2}\mathbb{E}_{\theta_\star}\left[\left\|\hat{\theta}\big(\tau(\pi, T)\big) - \theta_\star\right\|_{\mathrm{F}}^2\right]. \tag{3}$$

Of course, this quantity depends on $\theta_\star$ the true parameter of the system which is unknown. A natural way of proceeding is to estimate $\theta_\star$ sequentially, as follows.

Definition 1 (Adaptive system identification): Given an estimate $\hat{\theta}_i$ of $\theta_\star$, the policy for the next sequence of inputs can be chosen so as to minimize a cost function F approximating the MSE (**??**), using $\hat{\theta}_i$ as an approximation of $\theta_\star$. Then, these inputs are played and $\theta_\star$ is re-estimated with the resulting trajectory, and so on. We call planning the phase of minimizing F.

This approach is summarized in Algorithm **??**, which takes as inputs a first guess for the parameters to estimate θ_0 and a policy π_0, the problem parameters σ and γ, a schedule $\{t_0, = 0, t_1, \dots, t_{n-1}, t_n = T\}$, a cost functional F and an estimator $\hat{\theta}$. An adaptive identification algorithm is hence determined by a triplet $(\hat{\theta}, F, \{t_i\})$. A natural estimator is the least squares

This work was not supported by any organization

M. Blanke and M. Lelarge are with INRIA, DI/ENS, PSL Research University, Paris, France. matthieu.blanke@inria.fr, marc.lelarge@inria.fr

[1] Our code is available at https://github.com/MB-29/greedy-identification

Algorithm 1 Sequential system identification

```
inputs initial guess $\theta_0$, $\pi_0$, noise variance $\sigma^2$, power $\gamma^2$,
cost functional $F$, estimator $\hat{\theta}$
output final estimate $\theta_T$
for $0 \leq i \leq n-1$ do
    run the true system $t_{i+1} - t_i$ steps with
    controls $u_t = \pi_i(x_{1:t}, u_{1:t-1})$
    $\theta_i = \hat{\theta}(x_{1:t_i}, u_{1:t_i-1})$            ▷ estimation
    $\pi_i$ solves $\min_{\pi \in \Pi_\gamma} F(\pi; \theta_i, t_{i+1})$      ▷ planning
end for
```

estimator $\hat{\theta} = \hat{\theta}_{\mathrm{LS}}$ which we define in Section **??**. In the rest of this work, we set $\hat{\theta} = \hat{\theta}_{\mathrm{LS}}$.

Example 1 (Random policy): A naive strategy for system identification consists in playing random inputs with maximal energy at each time step. This corresponds to the choice $t_i = i$ and π_i returning $u_t \sim \mathcal{N}(0, \gamma^2/m)$.

Example 2 (Task-optimal pure exploration): In [**?**], the authors propose the following cost function

$$F(\pi; \theta, t) = \mathrm{tr}\left[\big(\Gamma_t\big(\tau(\pi); \theta\big)\big)^{-1}\right], \tag{4}$$

where Γ_t is defined in equation (**??**) below. As we will see in Section **??**, this corresponds to A-optimal experimental design. The authors show that this cost function approximates the MSE in the long time limit at an optimal rate when $T \to +\infty$. In their identification algorithm, they set $t_i = 2^i \times T_0$ for some initial epoch T_0.

Example 3 (Oracle): An oracle is a controller who is assumed to choose their policy with the knowledge of the true parameter $\theta_\star$. It can hence perform one single, offline optimization of $F(\pi; \theta, T) = \mathrm{MSE}(\pi)$ over $\{t_i\} = \{0, T\}$. By definition, the inputs played by the oracle are the optimal inputs for our problem of mean squared error system identification.

C. Contributions

In practice, systems have complex dynamics and can only be approximated locally by linear systems. We still believe that in order to understand complex systems, we need to understand identification of linear systems as on short time scales, we can approximate the complex system with a linear one. In order to be practical, our identification algorithm needs to interact as little as possible with the true system and to take decisions as fast as possible. With previous notations, we are interested in cases where T is small (to ensure that in practice the dynamics remains time-invariant and linear) and where the estimation and planning steps need to be very fast in order to run the algorithm online.

In this work, we explore a setting for linear system identification with hard constraints on the number of interactions with the real system and on the computing resources used for planning and estimation. To the best of our knowledge, finite-time system identification guarantees are only available in the large T limit which makes the hypothesis of linear dynamic quite unlikely. Using a framework based on experimental design, we propose a greedy online algorithm requiring minimal computing resources. The resulting policy gives a control that maximizes the amount of information collected at the next step. We show empirically that for short interactions with the system, this simple approach can actually outperforms more sophisticated gradient-based methods.

We also propose a method to compute an oracle optimal control, against which we can compare the different identification algorithms.

D. Related work

System identification has been studied extensively in the last decades [**?**], [**?**]. The question of choosing the maximally informative input can be tackled in the framework of classical experimental design [**?**], [**?**]. Several methods have been proposed for the particular case of dynamic systems [**?**], [**?**], [**?**] A comprehensive study can be found in [**?**], with a focus on SISO systems.

In the machine learning community, the last few years have seen an increasing interest in finite-time system identification [**?**], [**?**], [**?**], [**?**]. These works typically derive theoretical error rates for linear dynamic system estimation and produce high probability bounds guaranteeing that the estimation is smaller than ε with probability greater than $1 - \delta$ after a certain number of samples. The question of designing optimal inputs is tackled in [**?**], [**?**]. The authors derive an asymptotically optimal algorithm by computing the control in the frequency domain. In [**?**], an approach to control partially nonlinear systems is proposed.

II. Background

It is convenient to describe the structure of the state as a function of the inputs and the noise. By integrating the dynamics (**??**), we obtain the following result.

Proposition 1: The state can be expressed as $x_t = \bar{x}_t + \tilde{x}_t$ with

$$\bar{x}_t = \sum_{s=0}^{t-1} A^{t-1-s} B u_s, \quad \tilde{x}_t = \sum_{s=0}^{t-1} A^{t-1-s} w_s. \tag{5}$$

Note that that $\bar{x}_t = \mathbb{E}_\theta[x_t]$ solves the deterministic dynamics $\bar{x}_{t+1} = A\bar{x}_t + Bu_t$ and $\tilde{x}_t$ has zero mean and is independent of the control. The two terms $\bar{x}_t$ and $\tilde{x}_t$ depend linearly on the Bu_s and the w_s respectively.

The data-generating distribution knowing the parameter θ can be computed using the probability chain rule with the dynamics (**??**):

$$p(\tau|\theta) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2\sigma^2} \sum_{t=0}^{T-1} \|x_{t+1} - Ax_t - Bu_t\|_2^2\right]. \tag{6}$$

We define the log-likelihood (up to a constant):

$$\begin{aligned}\ell(\tau,\theta) &= -\frac{1}{2\sigma^2}\sum_{t=0}^{T-1}\|x_{t+1}-Ax_t-Bu_t\|_2^2 \\ &= -\frac{1}{2\sigma^2}\|Y - Z\theta^\top\|_{\mathrm{F}}^2,\end{aligned} \tag{7}$$

where we have noted $Y = (y_0 \dots y_{T-1})^\top \in \mathbb{R}^{T\times d}$ and $Z = (z_0 \dots z_{T-1}{}^\top) \in \mathbb{R}^{T\times q}$ the observations and the covariates associated to the parameter θ. If $\theta = (A\,B)$, then $y_t = x_{t+1}$, $z_t = \begin{pmatrix}x_t\\ u_t\end{pmatrix}$. If $\theta = A$, then $y_t = x_{t+1} - Bu_t$ and $z_t = x_t$. We also note $U = (u_0 \dots u_{T-1}{}^\top) \in \mathbb{R}^{T\times m}$ the input matrix and $X = (x_0 \dots x_{T-1}{}^\top) \in \mathbb{R}^{T\times d}$ the state matrix. We define the moment matrix $M_t = \sum_{s=0}^{t} z_t z_t{}^\top$ and the Gramians of the system at time t:

$$\Gamma_t(\tau;\theta) = \frac{1}{t}\mathbb{E}_\theta\left[M_{t-1}\right] \tag{8}$$

and $G_t(A) = \sum_{s=0}^{t-1} A^s A^{s\top}$. Note that $Z^\top Z = M_T$

A. Ordinary least squares

Given a trajectory, a natural estimator for the matrix $A_\star$ is the least squares estimator. The theory of least squares provides us with a formula for the mean squared error with respect to the ground truth, which can be used as a measure of the quality of a control.

Proposition 2 (Ordinary least squares estimator):

Given inputs U and noise W, the ordinary least squares (OLS) estimator associated to the resulting trajectory X is

$$\hat{\theta}(\tau) = \big((Z^\top Z)^{-1}Z^\top Y\big)^\top. \tag{9}$$

and its difference to $\theta_\star$ is given by

$$\begin{aligned}\big(\hat{\theta}(\tau) - \theta_\star\big)^\top &= (Z^\top Z)^{-1}Z^\top W \\ &= Z^+ W,\end{aligned} \tag{10}$$

where Z^+ denotes the pseudo-inverse of Z. Noting θ_t the least squares estimator obtained from the trajectory up to time t, we recall the recursive update formula

$$\theta_{t+1}{}^\top = M_{t+1}^{-1}\big(M_t\theta_t + z_t y_t{}^\top\big). \tag{11}$$

The least squares estimator minimizes the quadratic loss $\frac{1}{2}\sum_{t=0}^{T-1}\|x_{t+1} - Ax_t - Bu_t\|_2^2$, which writes

$$\frac{1}{2}\left\|Y - Z\theta^\top\right\|_{\mathrm{F}}^2 = \frac{1}{2}\sum_{j=1}^{d}\|Y_j - Z\theta_j\|_2^2 \tag{12}$$

with Y_j the j-th column of Y and θ_j the j-th row of θ. The d terms of the sum can be minimized independently, with each θ_j minimizing the least squares of the vectorial relation $Y_j = Z\beta$. The solution for θ_j is equal to $\hat{\theta}_j = (Z^\top Z)^{-1}Z^\top Y_j$ (see *e.g.* [**?**]). By concatenating the columns, we obtain that $\hat{\theta}^\top = (Z^\top Z)^{-1}Z^\top Y$, which proves (**??**). Substituting $Y = Z\theta_\star{}^\top + W$ yields (**??**). Note here that the controllability assumption on $(A_\star, B_\star)$ ensures that Z can be made full rank, and hence that the moment matrix $Z^\top Z$ is invertible.

Definition 2 (OLS mean squared error): For a given trajectory τ generated with a matrix $A_\star$ and noise W, the Euclidean mean squared error (MSE) is

$$\begin{aligned}\|\hat{\theta}_{\mathrm{LS}} - \theta_\star\|_{\mathrm{F}}^2 &= \left\|\big((Z^\top Z)^{-1}Z^\top W\big)^\top\right\|_2^2 \\ &= \mathrm{tr}\left[Z(Z^\top Z)^{-2}Z^\top W W^\top\right].\end{aligned} \tag{13}$$

If the noise W and the covariates Z were independent, then the expected error would reduce to the A-optimal design objective $\mathbb{E}[\mathrm{tr}(Z^\top Z)^{-1}]$. It is not the case in our framework since Z is generated with W.

B. Classical optimal design

The correlation between Z and W makes the derivation of a tractable expression for the expectation of (**??**) complicated. In this section, we show how a more tractable objective can be computed by applying theory of optimal experimental design [**?**], [**?**]. In the classical theory of optimal design, the informativeness of an experiment is measured by the size of the expected Fisher information.

Definition 3 (Fisher information matrix): Let $\ell(\tau,\theta) = \log p(\tau|\theta)$ denote the log-likelihood of the data-generating distribution knowing the parameter θ. The Fisher information matrix is defined as

$$I(\theta) = -\mathbb{E}_\theta\left[\frac{\partial^2\ell(\tau,\theta)}{\partial\theta^2}\right] \quad \in \mathbb{R}^{qd\times qd}. \tag{14}$$

Proposition 3: For the LTI system (**??**),

$$I(\theta) = \frac{T}{\sigma^2}\mathrm{diag}(\Gamma_T,\dots,\Gamma_T), \tag{15}$$

the number of blocks being d. Furthermore, Γ_T can be computed as

$$\Gamma_T = \frac{1}{T}\sum_{t=0}^{T-1}\bar{z}_t\bar{z}_t^\top + \sigma^2 G_t(A). \tag{16}$$

The log-likelihood (**??**) can be separated into a sum over the θ_j as in (**??**). The quadratic term in θ_j is $\|Z\theta_j\|_2^2 = \theta_j{}^\top Z^\top Z\theta_j$ and the other terms are constant or linear. Differentiating twice and taking the expectation gives $\mathbb{E}_\theta[Z^\top Z]$, which yields the desired result after dividing by $-\sigma^2$.

Following the decomposition (**??**), $z_t z_t{}^\top = \bar{z}_t\bar{z}_t^\top + \tilde{z}_t\tilde{z}_t^\top + \bar{z}_t\tilde{z}_t^\top + \tilde{z}_t\bar{z}_t^\top$. Taking the expectation, we obtain $\mathbb{E}[z_t z_t{}^\top] = \bar{z}_t\bar{z}_t^\top + \sigma^2 G_t(A)$. Summing over t yields the result. Note that the first term is deterministic and depends on the control whereas the second term depends on the noise and not on the control. Therefore, the expected moment matrix is the sum of a noise term and of a deterministic control term.

Definition 4: In classical optimal design, the size of the information matrix is measured by some criterion $\Phi : \mathbb{S}_n^+(\mathbb{R}) \to \mathbb{R}_+$, which is a functional of its eigenvalues $\lambda_1,\dots,\lambda_d \geq 0$. The quantity $\Phi(I)$ represents the amount

TABLE I: Alphabetical design criteria, matrix operator form.

Optimality	$\Phi(\lambda_1, \dots, \lambda_d)$
A-optimality	$-(1/\lambda_1 + \dots + 1/\lambda_d)$
D-optimality	$\log \lambda_1 + \dots \log \lambda_d$
E-optimality	λ_1

of information brought by the experiment and should be maximized.

Example 4: Some of the usual criteria are presented in Table **??**.

The criteria are required to have properties such as homogeneity, monotonicity and concavity in the sense of the Loewner ordering, which can be interpreted in terms of information theory: monotonicity means that a larger information matrix brings a greater amount of information, concavity means that information cannot be increased by interpolation between experiments. We refer to [**?**] for more details.

The theory of classical optimal design leads to the definition of the following optimal design informativeness functional.

Definition 5 (Optimal design functional): Let Φ denote an optimal design criterion. Then the associated cost is defined as

$$\begin{aligned} F_\Phi(\pi; \theta, t) &= -\Phi\left[\Gamma_t\big(\tau(\pi); \theta\big)\right] \\ &= -\Phi\left[\sum_{s=0}^{t-1} \bar{z}_s \bar{z}_s^\top + \sigma^2 G_s(A)\right], \end{aligned} \tag{17}$$

where the $\bar{z}_s$ depend on the inputs u_s through (**??**).

Remark 1: We note from equation (**??**) that Z is affine in U. Hence, $Z^\top Z$ is quadratic in U, and maximizing (**??**) efficiently is challenging even with concavity assumptions on Φ.

C. Small noise regime

The optimal design functional (**??**) can be related to the MSE in the small noise regime $\sigma \ll \gamma$.

Proposition 4: The A-optimal design functional (**??**) is a $\mathcal{O}(\sigma/\gamma)$ approximation of the MSE (**??**):

$$\mathrm{MSE}(\pi) = \frac{1}{2} F_{\mathrm{A}}(\pi; \theta_\star, T) + \mathcal{O}(\sigma/\gamma). \tag{18}$$

We introduce the rescaled variables $\zeta = (1/\gamma)Z$ and $\omega = (1/\sigma)W$ which are of order 1. Extending the notations of equation (**??**), $Z = \bar{Z} + \tilde{Z}$, where the first term is of order γ and the second is of order σ. Therefore, $Z = \bar{Z} + \mathcal{O}(\sigma)$, or equivalently $\zeta = \bar{\zeta} + \mathcal{O}(\sigma/\gamma)$. By Proposition **??**, ζ^+ is differentiable at $\bar{\zeta}$ so $\zeta^+ = \bar{\zeta}^+ + \mathcal{O}(\sigma/\gamma)$. Taking the squared norm and using Cauchy-Schwartz inequality, we obtain

$$\left\|\zeta^+ \omega\right\|^2 = \left\|\bar{\zeta}^+ \omega\right\|^2 + \mathcal{O}(\sigma/\gamma). \tag{19}$$

Furthermore,

$$\begin{aligned} \mathbb{E}\left[\left\|\bar{\zeta}^+ \omega\right\|^2\right] &= \mathbb{E}\left[\operatorname{tr}\left(\bar{\zeta}(\bar{\zeta}^\top \bar{\zeta})^{-2} \bar{\zeta}^\top \omega \omega^\top\right)\right] \\ &= \operatorname{tr}\left[(\bar{\zeta}^\top \bar{\zeta})^{-1}\right]. \end{aligned} \tag{20}$$

Gathering (**??**) and (**??**), we obtain

$$\frac{1}{2}\mathbb{E}\left[\left\|\zeta^+ \omega\right\|^2\right] = \frac{1}{2}\operatorname{tr}\left[(\bar{\zeta}^\top \bar{\zeta})^{-1}\right] + \mathcal{O}(\sigma/\gamma). \tag{21}$$

Remark 2: In classical least squares regression, the covariates Z are independent of the noise W. As a consequence, the minimziation of the mean squared estimation error leads to the classical A-optimality criterion. This does not hold in general in our framework because the signal and the noise are coupled by the dynamics (**??**). However, Proposition **??** shows that this criterion does hold in the small noise regime at first order in σ/γ. Indeed, when $\sigma \ll \gamma$ the contribution of the noise to the signal is negligible because the deterministic part of the signal is of order γ.

Remark 3: From Proposition **??** and the definition of A-optimality, we see that the MSE approximately scales like $1/T$ when the number of observations increases. This is confirmed by experiments.

III. Online greedy identification

A. One-step-ahead objective

A simple, natural approach for system identification consists in choosing a decision sequentially at each time step. At each time t, the control u_t is chosen with energy γ^2 so as to maximize a one-step-ahead objective. Then, a new observation x_t is collected and the process repeats. Following Section **??**, u_t can be chosen to maximize the value of F_Φ at $t+1$. This corresponds to the choice of functional $F = F_\Phi$ and to the one-step schedule $t_i = i$.

Upon choosing u_t, the policy π_t should select u_t so as to maximize the design criterion Φ applied on the one-step ahead, u_t-dependent information matrix, the past trajectory $x_{0:t}$ being fixed. The one-step-ahead information matrix is $M_{s-1} + \mathbb{E}_{A_s}[z_s {z_s}^\top]$, with $s = t$ when $B_\star$ is estimated (because then then next u_t-dependent covariate is z_t) and $s = t+1$ if $B_\star$ is known, because then the next u_t-dependent covariate is x_{t+1}. Therefore, one-step ahead planning yields the following optimization problem:

$$\begin{aligned} \max_{u \in \mathbb{R}^m} \quad & \Phi\left(\bar{M}_t + z(u) z(u)^\top\right) \\ \text{such that} \quad & \|u\|^2 \leq \gamma^2, \end{aligned} \tag{22}$$

with

$$\bar{M}_t = \begin{cases} M_{t-1} + \sigma^2 G_t(A_t) & \text{if} \quad \theta = (A,\, B) \\ M_t + \sigma^2 G_{t+1}(A_t) & \text{if} \quad \theta = A, \end{cases} \tag{23}$$

and

$$z(u) = \begin{cases} \begin{pmatrix} x_t \\ u \end{pmatrix} & \text{if} \quad \theta = (A,\, B) \\ A_t x_t + B_\star u_t & \text{if} \quad \theta = A. \end{cases} \tag{24}$$

Remark 4: With this greedy policy, the energy constraint imposed for one input ensures that the global power constraint (**??**) is met.

The corresponding identification process is detailed in Algorithm **??**. We will see in Section **??** that problem (**??**) can be solved accurately and at a cheap cost. Moreover, Algorithm **??** offers the advantage of improving the knowledge of $\theta_\star$ at each time step using all the available information on the parameter to plan at each time step. This way, the bias affecting planning due to the uncertainty about $\theta_\star$ is minimized. When planning is performed over larger time sequences, a large bias could impair the identification of the system.

Algorithm 2 Greedy system identification

```
inputs initial guess $\theta_0$, noise variance $\sigma^2$, power $\gamma^2$, time horizon $T$, design criterion $\Phi$
output final estimate $\theta_T$
for $0 \leq t \leq T-1$ do
    $u_t \in \underset{\|u\|^2 \leq \gamma^2}{\operatorname{argmax}} \, \Phi\big(\bar{M}_t + z(u)z(u)^\top\big)$
    play $u_t$, observe $x_{t+1}$
    $M_{t+1} = M_t + x_{t+1}{x_{t+1}}^\top$
    ${\theta_{t+1}}^\top = M_{t+1}^{-1}\big(M_t\theta_t + x_t{y_t}^\top\big)$
end for
```

B. Solving the one-step optimal design problem

We show that the one-step ahead planning for online system identification is equivalent to a convex quadratic program which can be solved efficiently.

Proposition 5: For D-optimality and A-optimality, there exists a symmetric matrix $Q \in \mathbb{R}^{m\times m}$ and $b \in \mathbb{R}^m$ the problem (**??**) is equivalent to

$$\begin{aligned} \min_{u\in\mathbb{R}^d} \quad & u^\top Q u - 2b^\top u \\ \text{such that} \quad & \|u\|^2 \leq \gamma^2. \end{aligned} \tag{25}$$

From Proposition **??**, we find that

$$\begin{aligned} \log\det\big(\bar{M_t} + z(u)z(u)^\top\big) &= \log\det\bar{M_t} \\ &+ \log\big(1 + z(u)^\top \bar{M_t}^{-1} z(u)\big). \end{aligned} \tag{26}$$

Similarly, from Corollary **??** ,

$$\begin{aligned} -\mathrm{tr}\left[\left(\bar{M_t} + z(u)z(u)^\top\right)^{-1}\right] &= 1 - \mathrm{tr}\left[\bar{M_t}^{-1}\right] \\ &- \frac{1}{1 + z(u)^\top \bar{M_t}^{-1} z(u)}. \end{aligned} \tag{27}$$

Maximizing these quantities with respect to u amounts to maximizing $z(u)^\top \bar{M_t}^{-1} z(u)$. The matrix $\bar{M_t}^{-1}$ is symmetric because the M_t and the G_t are symmetric, and so are its diagonal submatrices. Given the affine dependence of z in u and the (possible) block structure of z and M_t, $z(u)^\top \bar{M_t}^{-1} z(u)$ is of the form $u^\top Q u - 2b^\top u$, up to a constant. We provide an explicit formula for Q and b in the case where $\theta = A$ in Remark **??** We now characterize the minimizers of Problem (**??**). If a minimizer can be found in the interior of the constraining sphere, then Q is positive semidefinite and the problem can be tackled using unconstrainted optimization. We thus consider the equality constrained problem

$$\begin{aligned} \min_{u\in\mathbb{R}^d} \quad & u^\top Q u - 2b^\top u \\ \text{such that} \quad & \|u\|^2 = \gamma^2. \end{aligned} \tag{28}$$

Proposition 6: Note $\{\alpha_i\}$ the eigenvalues of Q, and u_i and b_i the coordinates of u_* and b in a corresponding orthonormal basis. Then a minimizer u_* satisfies the following equations for some nonzero scalar μ:

$$u_i = b_i/(\alpha_i + \mu) \quad \text{and} \quad \sum_i \frac{{b_i}^2}{(\alpha_i+\mu)^2} = \gamma^2. \tag{29}$$

By the Lagrange multiplier theorem there exists a nonzero scalar μ such that $Qu_* - b = -\mu u_*$, where μ can be scaled such that $Q + \mu I_d$ is nonsingular. Inverting the optimal condition and expanding the equality constraint gives the two conditions. Problem (**??**) can hence be solved at the cost of a scalar root-finding and an eigenvalue decomposition. In [**?**], bounds are provided so as to initialize the root-finding method efficiently.

Remark 5: In the case where $B_\star$ (*i.e.* $\theta = A$), Q and b have the following expressions:

$$Q = -B^\top \bar{M_t}^{-1} B, \quad b = B^\top \bar{M_t}^{-1} A_t x_t. \tag{30}$$

IV. Gradient-based identification

In this section, we propose a gradient-based approach to planning. In a sequential identification scheme of Algorithm **??**, the cost functions (**??**) and (**??**) can be optimized by projected gradient descent. This builds on the following remark.

Remark 6 (Differentiability of the functionals): The functionals (**??**) and (**??**) are differentiable functions of the output. Indeed, X is an affine function of the inputs as shown in Proposition **??**, and the controllability of (A, B) guarantees that $Z^\top Z$ is positive definite. Furthermore, the operations of pseudo-inverse (see Proposition **??**) and the optimal design criteria of Table **??** are differentiable over the set of positive definite matrices.

The gradients with respect to U can either be derived analytically (see [**?**], section 6 for the derivation of an adjoint equation) or automatically in an automatic differentiation framework. We rescale U at each step to ensure the power constraint is met. The t_i are chosen arbitrarily. The computational complexity of the algorithm is linear in T: each gradient step backpropagates through the planning time interval.

A. Gradient-based optimal design

We propose a gradient-based method to optimize U by performing gradient descent directly on U in functional (**??**). Note that we optimize the inputs directly in the time domain, whereas other approaches such as [**?**] perform optimization in the frequency domain by restricting the control to periodic inputs.

B. Gradient through the oracle MSE

Given the true parameters $\theta_\star = (A_\star \, B_\star)$, the optimal control for the MSE minimizes the MSE cost (**??**), as explained Example **??**. However, the dependency between Z and W makes this functional complicated to evaluate and to minimize with respect to the inputs, even when the true parameters

$\theta_\star$ are known. We propose a numerical method to minimize (**??**) using automatic differentiation an Monte-Carlo sampling. Given one realization of the noise and inputs U, the gradient of the squared error (**??**) can be computed automatically in an automatic differentiation framework. Then, one can sample a batch of b noise matrices $W_1, \dots, W_b \sim \mathcal{N}(0, \sigma^2 I)$ and approximate the gradient of (**??**) by

$$\nabla \mathrm{MSE}(U) \simeq \frac{1}{b} \sum_{i=1}^{b} \nabla_U \mathrm{tr}\left[Z(Z^\top Z)^{-2} Z^\top W_i {W_i}^\top \right]. \quad (31)$$

Although we do not have convergence guarantees due to the lack of structure of the objective function, the gradient descent does converge in practice, to a control that outperforms the adaptive controls.

Algorithm 3 Planning by projected gradient descent

```
inputs $A_t$, $\sigma$, $\gamma$, $T$, $\eta$, $H_t$
output control $U \in \mathbb{R}^{(T-t)\times m}$
for $0 \leq j \leq n_{\mathrm{gradient}}$ do
    $G(U) = F[X(U)|H_t]$
    $U = U - \eta \nabla G(U)$
    $U = (\gamma\sqrt{T}/\|U\|_{\mathrm{F}}) \times U$
end for
```

V. Performance study

A. Complexity analysis

Definition 6 (Performance): Let θ_T denote the estimation produced by the learning algorithm at the end of identification. The performance of the policy π is measured by the average error over the experiments on the true system: $\varepsilon = \mathrm{MSE}(\pi)$. We study the performance of our algorithms as a function of the number of observations T and C the computational cost. We also introduce the computational rate $c = C/T$.

Algorithm **??** and the gradient identification algorithm have linear time complexity. Hence, we define c_{greedy} and c_{gradient} for a given number of gradient iterations. In practice, we find that $c_{\mathrm{greedy}} \ll c_{\mathrm{gradient}}$, where c_{gradient} is the computational rate needed for the gradient descent to converge. As pointed out in Remark **??**, the squared error essentially scales like $1/T$. This is verified experimentally. Given the previous observations, we postulate that the performance of our algorithms takes the form

$$\varepsilon(C, T) = \eta(c)/T. \quad (32)$$

We build an experimental diagram where we plot the average estimation error for $\theta_\star = A_\star$ as a function of the two types of resource T and C for the gradient algorithm. Increasing C allows for more gradient steps. We run trials with random matrices $A_\star$ of size $d = 4$, with $B = I_d$. We set $\gamma = 1$, $\sigma = 10^{-2}$, $T \in [60, 220]$. The gradient algorithm optimizes the A-optimality functional (**??**) with a batch size of $b = 100$ and $\{t_i\} = \{0, 10, T/2, T\}$. The obtained performances are to be compared with those of the greedy algorithm (with the A-optimality cost function), which has a fixed, small computational rate c. Our diagrams are plotted on Fig. **??**.

Fig. 1: Experimental (T, C) diagram. **Left** Performance of the gradient algorithm, with varying T and C (varying number of gradient steps). **Right** Relative performance of the gradient algorithm with respect to the greedy algorithm: negative means that the gradient performs better.

Our diagrams show that the greedy algorithm is preferable in a phase of low computational rate: $C < c \times T$, as suggested by (**??**). The phase separation corresponds to a relatively high number of gradient steps. Indeed, the iso-performance along this line are almost vertical, meaning that the gradient descent has almost converged. Furthermore, the maximum performance gain of the gradient algorithm relatively to the greedy algorithm is of 10%.

B. Average estimation error

We now test the performances of our algorithms on random matrices, with the same settings as in the previous experiment. For the gradient algorithm, the minimal number of gradient iterations to reach maximum performance for was found to be $n_{\mathrm{gradient}} = 120$. For each matrix $A_\star$, we also compute an oracle optimal control using Algorithm **??** with a batch size of $b = 100$, and run a random input baseline (see Example **??**), and the TOPLE algorithm of [**?**].

Both the gradient algorithm and the greedy algorithm closely approach the oracle. The former performs slightly better than the latter in average. However, the computational cost of the gradient algorithm is far larger, as Table **??** shows. Indeed, the number of gradient steps to reach convergence in this setting is found to be of order $n_{\mathrm{gradient}} \simeq 100$. Note that the number of sub-gradient steps for the TOPLE algorithm is found to be $n_{\mathrm{TOPLE}} \simeq 1000$, and so $n_{\mathrm{TOPLE}} \simeq 20 \times n_{\mathrm{gradient}}$.

Fig. 2: Identification error for random $A_\star$, averaged over 1000 samples.

TABLE II: Average computational rate for the different algorithms.

	Random	**TOPLE [?]**	**Gradient**	**Greedy**
c	1	$n_{\text{TOPLE}} \times 0.02$	$n_{\text{gradient}} \times 0.5$	2.36

TABLE III: Frobenius error for $A_\star$ in the lateral system of the aircraft, $T = 150$. Our oracle algorithm reaches an error of $8.0{\times}10^{-2}$. The computational time is expressed in an arbitrary unit.

	Random	**TOPLE [?]**	**MSE gradient**	**Greedy**
Error	1.1×10^{-1}	8.6×10^{-2}	8.3×10^{-2}	8.2×10^{-2}
Time	1	55.7	25	1.13

C. Identification of an aircraft system

We now study a more realistic setting from the field of aeronautics: we apply system identification to an aircraft system. We use the numerical values issued in a report from the NASA [**?**]. The lateral motion of a Lockheed Jet star is described by the slideslip and roll angles and the roll and yaw rates $(\beta, \phi, p, r)^\top := x$. The control variables are the aileron and rudder angles $(\delta_\text{a}, \delta_\text{r}) := u$. The linear dynamics for an aircraft flying at 573.7 meters/sec at 6.096 meters are given by the following matrix, obtained after discretization and normalization of the continuous-time system [**?**]:

$$A_\star = \begin{pmatrix} .955 & -.0113 & 0 & -.0284 \\ 0 & 1 & .0568 & 0 \\ -.25 & 0 & -.963 & .00496 \\ .168 & 0 & -.00476 & -.993 \end{pmatrix}, \tag{33}$$

$$B_\star = 0.1 \times \begin{pmatrix} 0 & 0.0116 \\ 0 & 0 \\ 1.62 & .789 \\ 0 & -.87 \end{pmatrix}, \tag{34}$$

and $\sigma = 1$, $\gamma \simeq 4$ deg. We apply our algorithms on this LTI system. Our results are summarized in Table **??**.

As we can see, the greedy algorithm outperforms the gradient-based algorithms, both in performance and in computational cost. This could be explained by the fact that the signal-to-noise ratio in this system is of order 1, hence the estimation bias in planning is large and it is more effective to plan one-step-ahead than to do planning over large epochs. We obtain similar results for the longitudinal system of a C-8 Buffalo aircraft [**?**].

VI. Conclusion

In this work, we explore a setting for linear system identification with hard constraints on the number of interactions with the real system and on the computing resources used for planning and estimation. We introduce a greedy online algorithm requiring minimal computing resources and show empirically that for small values of interactions with the system, it can actually outperform more sophisticated gradient-based methods. Extension of this approach to optimal control for the LQR is an interesting direction of future research.

VII. Matrix calculus

Proposition 7: On a domain where X has linearly independent columns, X^+ is a differentiable function of X and

$$\mathrm{d}X^+ = -X^+\mathrm{d}XX^+ + X^+{X^+}^\top \mathrm{d}X^\top (I - XX^+). \tag{35}$$

See [**?**].

Lemma 1: Let $A \in \mathbb{R}^{k\times\ell}$ and $B \in \mathbb{R}^{n\times m}$. Then

$$\det(I_{k,m} + AB) = \det(I_{n,\ell} + BA). \tag{36}$$

Proposition 8: Let $M \in \mathbb{R}^{d\times d}$ be a nonsingular matrix and $x, y \in \mathbb{R}^d$. Then

$$\det(M + xy^\top) = \det M \times (1 + y^\top M^{-1}x). \tag{37}$$

$$M + xy^\top = M(I + M^{-1}xy^\top) \tag{38}$$

Apply Lemma **??**:

$$\begin{aligned} \det(M + xy^\top) &= \det M \times \det(I_d + M^{-1}xy^\top) \\ &= \det M \times \det(I_1 + y^\top M^{-1}x) \\ &= \det M \times (1 + y^\top M^{-1}x). \end{aligned} \tag{39}$$

See [**?**].

Proposition 9: Let $0 < A \le B$ be positive definite matrices of $\mathbb{R}^{d\times d}$, and $x \in \mathbb{R}^d$. Then

$$\log\det(A + xx^\top) - \log\det A \ge \log\det(B + xx^\top) - \log\det B. \tag{40}$$

By Proposition **??**,

$$\log\det(A + xx^\top) - \log\det A = \log(1 + x^\top A^{-1}x) \tag{41}$$

Since $0 < A \le B$, both matrices are nonsingular and $0 < B^{-1} \le A^{-1}$. Hence,

$$\begin{aligned} \log(1 + x^\top A^{-1}x) &\ge \log(1 + x^\top B^{-1}x) \\ &= \log\det(B + xx^\top) - \log\det B \end{aligned} \tag{42}$$

Proposition **??** admits the following generalization.

Proposition 10: Let $M \in \mathbb{R}^{d\times d}$ be a nonsingular matrix and let $x_1, \dots, x_n, y_1, \dots, y \in \mathbb{R}^d$. Then

$$\det\left(M+\sum_{i=1}^{n} x_i {y_i}^\top\right) = \det M + \sum_{i=1}^{n} {x_i}^\top \mathrm{adj}\left(M+\sum_{j=1}^{i-1} x_j {y_j}^\top\right) y_i \tag{43}$$

See [**?**].

Proposition 11: Let $M \in \mathbb{R}^{d\times d}$ be a nonsingular matrix and $x, y \in \mathbb{R}^d$. Then $(M + xy^\top)$ is nonsingular and

$$(M + xy^\top)^{-1} = (I_d - \frac{1}{1 + x^\top M^{-1} y} xy^\top)M^{-1} \tag{44}$$

Corollary 1: Let $M \in \mathbb{R}^{d\times d}$ be a nonsingular matrix and $x, y \in \mathbb{R}^d$. Then

$$\mathrm{tr}\left[(M + xy^\top)^{-1}\right] = \mathrm{tr}[M^{-1}] - \frac{y^\top M^{-1} x}{1 + x^\top M^{-1} y} \tag{45}$$