IEEE Control Systems Letters paper presented at
2023 American Control Conference (ACC)
San Diego, CA, USA. May 31 - June 2, 2023

Linear-Sized Sensor Scheduling Using Regret Minimization

Reza Vafaee[1] and Milad Siami[1]

***Abstract*— In this paper, we investigate the problem of time-varying sensor selection for linear time-invariant (LTI) dynamical systems. We develop a framework to design a sparse sensor schedule for a given large-scale LTI system with guaranteed performance bounds using a learning-based algorithm. We show how the observability Gramian matrix of an LTI system can be interpreted as the sum of rank-1 matrices indicating the contribution of the available sensors distributed in time. We then employ a regret minimization framework over density matrices to sparsify this sum of rank-1 matrices to approximate fully sensed LTI dynamics up to a multiplicative factor in some certain observability senses. Our main result provides a linear-sized (in dimension of system) sensor schedule that on the average activates only a constant number of sensors at each time step and significantly improves the previous linearithmic results. Our results naturally apply to the dual problem of actuator selection where a guaranteed approximation to the controllability Gramian will be provided.**

I. Introduction

To capture dynamics, high-dimensionality, and nonlinearity of the large-scale networks proposed by the realistic systems in science and engineering, a set of measurements provided by a set of sensors is required. Different parameters of a system can be measured via different types of sensors, while the sensors normally are distributed in different places to collect specific information of relevant spatial features of the system. Since performing each measurement might be expensive, or real-time processing of the complete set of measurements may impose an unreasonable computational load and so be practically impossible, it is often required to utilize only a limited number of sensors to estimate the overall states of the system. In addition, most high-dimensional dynamical networks present low-dimensional patterns and coherent structures with a few key sensors placed strategically that facilitate making fast control decisions increasingly requested by many engineering applications [1].

Finding the optimal set of these key sensors even for the most straightforward scenarios exhibits a mostly unsolved challenge which can be traced back to the 1970's and 80's [2]–[4]. In a simple problem with a given well-defined cost function, finding the optimal key sensors requires a combinatorial search that is NP-hard and mostly intractable [5]. In the dual problem, it was shown that even obtaining the sparse actuator set such that the resulting system is still controllable can be NP-hard [6], [7]. Therefore, most of the accomplished efforts have been focused on establishing an approximate solution for this problem. As [8] suggests, obtaining the sparse set that makes the resulting system reachable for a particular state is hard and even hard to approximate. In general, the problem of finding the sparse set of sensors or actuators may be optimized for robustness, network control, and consensus problems [9]–[12].

Early attempts such as [2]–[4] have been found to employ *nonlinear integer programming* to obtain an approximate solution. The algorithms proposed in this category, however, do not scale to the popular applications such as smart power grids [13], robotics [14], and epidemiological modeling and suppression [15]. In contrast, resent attempts have proposed to leverage the submodularity of certain performance metrics to design variants of greedy algorithms with performance guarantees [16]. When the metric is submodular, applying the classical rounding algorithms such as *pipage* and randomized rounding to the semi-definite programming (SDP) solution of the relaxed optimization offers computationally fast algorithms with a constant approximation [17]. Although these algorithms provide performance compared to the optimal (best) solution, they require an extra multiplicative factor of $\log n$ [18] and do not work for non-submodular metrics [19], [20]. Very recently, leveraging online learning and regret minimization, Vafaee and Siami in [21] have proposed a rounding method to obtain a so-called $(1+\epsilon)$ approximation solution for a large class of observability metrics (including non-submodular metrics). Their approach provides a time-varying schedule and solves the issue of the submodularity requirement, but the time complexity of their convex optimization part is considerable, which decreases the applicability of the proposed algorithm for large-scale dynamics. Balanced truncation and QR pivoting were exploited in [22] to propose a more computationally efficient algorithm compared to leading greedy and convex optimization approaches. Although their method maximizes the search space of possible selections by providing a faster algorithm, it only works for $\mathcal{H}_2$-norm and returns a static schedule that does not change over time.

While the results discussed so far try to find approximation algorithms to obtain the best sparse set, in some recent work [18], [23], researchers have sought to gain new fundamental insights into approximating various observability/controllability metrics compared to the case when all

This material is based upon work supported by the U.S. Department of Homeland Security under Grant Award Number 22STESE00001-01-00. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. Department of Homeland Security. This research was also supported in part by grants NSF 2208182, NSF 2121121, and ONR N00014-21-1-2431.

[1]Department of Electrical & Computer Engineering, Northeastern University, Boston, MA 02115 USA (e-mails: {vafaee.r, m.siami}@northeastern.edu).

the available sensors/actuators are activated. Siami et al. in [18], inspired by the new advances in randomized linear algebra and graph sparsification, developed deterministic and randomized frameworks to design a time-varying sparse schedule for LTI networks that each time samples on average a constant number of sensors, independent of the system dimension, to approximate a large class of controllability/observability metrics (including some non-submodular metrics). Their result provides a polynomial-time actuator schedule that approximates controllability metrics compared to when all actuators are in use, but it is necessary to sample $O(n \log n)$ active sensors (linearithmic size) to obtain an approximation solution up to a multiplicative factor. This left room to potentially improve the result by proposing a linear-sized algorithm, which is the main goal of this paper.

Contributions: To circumvent the convex relaxation step in the regret-based rounding procedure proposed in [21], our work proposes a novel algorithm for directly applying a regret minimization framework to the set of rank-1 contributions from available sensors. We demonstrate that our approach produces a time-varying schedule that approximates a broad class of observability metrics, including both submodular and non-submodular cases, and achieves comparable performance to a fully-sensed system. Furthermore, our approach yields a linear-sized sensor schedule that activates at most n/ϵ^2 sensors, where $\epsilon \in (0, 1)$ is an approximation factor, and ensures the same performance guarantees as [18]. Overall, our framework provides a simpler and more effective solution to the problem of sensor scheduling for large-scale LTI dynamics.

II. Preliminaries and Definitions

A. Mathematical Notation

Throughout the paper, the discrete time index is denoted by k. The sets of real (integer), non-negative real (integer), and positive real (integer) numbers are represented by $\mathbb{R}$ ($\mathbb{Z}$), $\mathbb{R}_+$ ($\mathbb{Z}_+$) and $\mathbb{R}_{++}$ ($\mathbb{Z}_{++}$), respectively. The set of natural numbers $\{i \in \mathbb{Z}_{++} \ : \ i \leq n\}$ is denoted by $[n]$. Uppercase letters stand for real-valued matrices (e.g., A), uppercase sans script letters illustrate set's names (e.g., A), lowercase bold letters denote vectors (e.g., $\mathbf{b}$), and non-bold lowercase letters are used for scalars and indices (e.g., j) and function names (e.g., $f(\cdot)$), except T that shows the total number of iteration in the regret minimization. For a vector $\boldsymbol{x} = [x_i] \in \mathbb{R}^n$, $\mathrm{diag}(\boldsymbol{x}) \in \mathbb{R}^{n\times n}$ is the diagonal matrix with elements of $\boldsymbol{x}$ sitting orderly on its diagonal. Furthermore, given a square matrix X, $\det(X)$ and $\mathrm{Trace}\, X$ refer to the determinant and the summation of on-diagonal elements of X, respectively. $\mathbb{S}^n_+$ (resp. $\mathbb{S}^n_{++}$) is the positive semi-definite cone (resp. positive definite cone) of n-by-n matrices. Let I denotes the identity matrix whose dimension can be inferred from the context. Notation $A \preceq B$ is equivalent to matrix $B-A$ being positive semi-definite. The transpose and Moore-Penrose pseudoinverse of matrix A are referred to by $A^\top$ and $A^\dagger$, respectively. Operator $\langle A, B\rangle \coloneqq \mathrm{Trace}\, A^\top B$ represents the inner product of two matrices A and B. The L_0-norm that counts the total number of nonzero elements of a vector is referred to by $\|\cdot\|_0$. Moreover, symbol $\|\cdot\|$ denotes the Euclidean norm for vectors and the spectral norm for matrices. Given a matrix $Z \in \mathbb{R}^{n\times m}$,

$$\mathrm{vec}(Z) = [z_{1,1}, \ldots, z_{n,1}, z_{1,2}, \ldots, z_{n,2}, \ldots, z_{1,m}, \ldots, z_{n,m}]^\top$$

denotes the vectorized form of a matrix, whereas vec^{-1} will perform the inverse of this operation.

B. Linear Systems and Observability

We start with a canonical LTI, discrete-time dynamics as follows:

$$\boldsymbol{x}(k+1) = A\boldsymbol{x}(k) + B\boldsymbol{u}(k), \tag{1}$$

$$\boldsymbol{y}(k) = C\boldsymbol{x}(k), \tag{2}$$

where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$, and $k \in \mathbb{Z}_+$. The state matrix A describes the underlying structure of the system and the interaction strength between the agents/states, matrix B identifies how the control input enters the system, and the output matrix C shows how output vector $\boldsymbol{y}$ relates to the state vector. Referring to (1)-(2), the expanded form is given by

$$\begin{bmatrix} \boldsymbol{y}(0) \\ \boldsymbol{y}(1) \\ \cdots \\ \boldsymbol{y}(t-1) \end{bmatrix} = \begin{bmatrix} C \\ CA \\ \cdots \\ CA^{t-1} \end{bmatrix} \boldsymbol{x}(0) + \mathcal{T}(t) \overbrace{\begin{bmatrix} \boldsymbol{u}(0) \\ \boldsymbol{u}(1) \\ \cdots \\ \boldsymbol{u}(t-1) \end{bmatrix}}^{\coloneqq \underline{\boldsymbol{u}}(t)}, \tag{3}$$

where $\mathcal{T}(t)$ is a block matrix with known structure, formed by combining the system matrices A, B, and C, which maps the input to the output over the interval $[0, t-1]$.

In an estimation problem, our goal is typically to recover the initial condition $\boldsymbol{x}(0) = \boldsymbol{x}_0$, given the measurement sequence $\boldsymbol{y}(k)$ and the input sequence $\boldsymbol{u}(k)$ for $0 \leq k \leq t-1$. The response in (3) can be separated into two terms. The second term (the *forced response*) on the right-hand side of (3) is known since the input vector, $\underline{\boldsymbol{u}}(t)$, is given. Therefore, we can subtract the forced response from the vector of measurements on the left to obtain

$$\underline{\boldsymbol{y}}(t) = \mathcal{O}(t)\ \boldsymbol{x}_0, \tag{4}$$

where we have defined $\underline{\boldsymbol{y}}(t)$ and the *t-step observability matrix* $\mathcal{O}(t)$ in an obvious way. Note that we only need to check observability when $\underline{\boldsymbol{u}}(t)$ is zero; a non-zero input changes the value of $\underline{\boldsymbol{y}}(t)$, but in either case, $\underline{\boldsymbol{y}}(t)$ is a known vector. In this paper, we assume $t \in \mathbb{Z}_+$ is the time horizon to estimate (a.k.a the observation horizon).

We define the *t-step observability Gramian matrix* $\mathcal{Q}(t) \coloneqq \mathcal{O}^\top(t)\mathcal{O}(t)$ for the dynamics (1)-(2). The t-step observability Gramian can be also expressed as the sum of rank-one

matrices

$$\mathcal{Q}(t) = \sum_{k=0}^{t-1} \sum_{j\in[p]} (\boldsymbol{c}_j^\top A^k)^\top (\boldsymbol{c}_j^\top A^k) = \sum_{i=1}^{tp} \boldsymbol{o_i} \boldsymbol{o_i}^\top = \sum_{i=1}^{tp} M_i, \tag{5}$$

where $\boldsymbol{c}_j^\top$'s are rows of matrix $C \in \mathbb{R}^{p\times n}$, $\boldsymbol{o_i}^\top$ denotes the i-th row of the observability matrix $\mathcal{O}(t)$, $i = k\cdot p + j$, $j \in [p]$, $k+1 \in [t]$, and finally M_i's are the rank-1 matrices built upon the rows of the observability matrix.

Assumption 1: Throughout the paper, we assume that system (1)-(2) is observable, i.e., the obervability matrix is full column rank, and the Gramian is positive definite (PD).

In time-varying sensor schedules, however, we will deal with linear, discrete-time dynamics with time-varying output matrix $C(k)$, i.e.,

$$\boldsymbol{x}(k+1) = A\boldsymbol{x}(k) + B\boldsymbol{u}(k), \text{ and } \boldsymbol{y}(k) = C(k)\boldsymbol{x}(k). \tag{6}$$

For this network, the t-step observability and Gramian matrices are defined as $\mathcal{O}_\star(t) = \left[C(0)^\top\ ,\ (C(1)A)^\top\ ,\ \cdots\ ,\ (C(t-1)A^{t-1})^\top\right]^\top$, and $\mathcal{Q}_\star(t) = \mathcal{O}_\star^\top(t)\mathcal{O}_\star(t)$, respectively.

C. Definition of Terms

This section is devoted to collect the definitions of some key concepts that are required to understand the rest of the paper.

1) Systemic Observability Metrics: Similar to [18], [21], [24], we define the notion of *systemic* observability metrics here. These measures are real-valued operators defined on the set of all linear dynamical systems (1)-(2) and quantify various measures of the required energies in the system. All the metrics depend on the observability Gramian matrix, which is a PD matrix (due to Assumption (1)). Therefore, one can define a systemic performance measure as an operator on the set of Gramian matrices of all observable systems over n agents.

Definition 1 (Systemic Observability Metric): A Gramian-based metric $\rho : \mathbb{S}_{++}^n \to \mathbb{R}_+$ is systemic if and only if, for all $\mathcal{A}, \mathcal{B} \in \mathbb{S}_{++}^n$, $\alpha \in [0,1]$, and $t > 0$, it satisfies: (i) *Homogeneity*: $\rho(t\mathcal{A}) = t^{-1}\rho(\mathcal{A})$ [1]; (ii) *Monotonicity*: If $\mathcal{B} \preceq \mathcal{A}$, then $\rho(\mathcal{B}) \geq \rho(\mathcal{A})$; (iii) *Convexity* criterion: $\rho\big(\alpha\mathcal{A} + (1-\alpha)\mathcal{B}\big) \leq \alpha\rho(\mathcal{A}) + (1-\alpha)\rho(\mathcal{B})$.

Several comprehensive studies have been already done on this class of performance metrics [18], [24]. They show that the criteria listed on Definition 1 hold for many popular choices of the observability metric. However, for the sake of brevity, we do not repeat their discussions here and refer the interested readers to [18, Table I] and [24, Table I] for a complete list of systemic measures.

[1]A function ρ is homogeneous if $\rho(t\mathcal{A}) = t^{-\gamma}\cdot\rho(\mathcal{A})$, where γ is the degree of homogeneity. However, throughout this paper, when we say a metric is homogeneous, it means it is homogeneous of degree 1.

2) Whitening Similarity Transformation: Given a non-singular coordinate transformation $\mathbf{T} \in \mathbb{R}^{n\times n}$, the new system realization of the state-space $\left[\begin{array}{c|c} A & B \\ \hline C & 0 \end{array}\right]$ is given by $\left[\begin{array}{c|c} \hat{A} & \hat{B} \\ \hline \hat{C} & 0 \end{array}\right] = \left[\begin{array}{c|c} \mathbf{T}^{-1}A\mathbf{T} & \mathbf{T}^{-1}B \\ \hline C\mathbf{T} & 0 \end{array}\right]$. Furthermore, the t-step observability Gramian matrix of the transformed system is obtained by $\hat{\mathcal{Q}}(t) = \mathbf{T}^\top\mathcal{Q}(t)\mathbf{T}$, where $\mathcal{Q}(t)$ is the t-step observability Gramian matrix of the original state-space.

Definition 2 (Whitening Similarity Transformation): Let $\mathbf{T} \coloneqq \mathcal{Q}(t)^{-1/2}$, then this change of coordinates converts the t-step observability Gramian matrix of the transformed system, $\hat{\mathcal{Q}}(t)$, to identity matrix I. Statisticians often call this process *whitening*, since it converts the covariance matrix of the given samples to the identity matrix.

We remark that since $\mathcal{Q}(t) \succ 0$ (due to Assumption 1), so $\mathcal{Q}(t)^{-1/2} = \mathcal{Q}(t)^{-\top/2} = U\Lambda^{-1/2}U^\top$ exists, where $U\Lambda U^\top$ is the eigen decomposition of the symmetric t-step observability Gramian matrix $\mathcal{Q}(t)$.

3) Leverage Score: The leverage score of the i-th row of matrix $P \in \mathbb{R}^{r\times n}$ is defined as

$$\tau_i = \tau(\boldsymbol{p}_i^\top) = \boldsymbol{p}_i^\top (P^\top P)^\dagger \boldsymbol{p}_i, \tag{7}$$

where $\boldsymbol{p}_i^\top$ is the i-th row of matrix P.

Remark 1: Leverage scores, τ_i's, are the diagonal elements of the projection matrix $P(P^\top P)^\dagger P^\top$, and therefore are always between zero and one (inclusive), i.e., $\tau_i \in [0,1]$ [25].

Remark 2: When P is full column rank, $\sum_{i\in[r]}\tau_i = \operatorname{Trace} P(P^\top P)^\dagger P^\top = \operatorname{Trace} I = n$.

4) Facts and Lemmas: To maintain consistency in our paper, we present a few commonly known results in this section.

Lemma 1: Given a matrix $A \succeq 0$,

$$\lambda_{\min}(A) = \min_{U\in\Omega_n} \langle A, U\rangle, \tag{8}$$

where $\Omega_n \coloneqq \{M \in \mathbb{S}_+^n\ :\ \operatorname{Trace} M = 1\}$.

Fact 1 (Jensen's Inequality): For a *real concave* function φ, numbers $x_1, x_2, \ldots, x_n$ in its domain, and positive weights β_j, *Jensen's inequality* is given by

$$\varphi\Big(\frac{\sum_{j=1}^n \beta_j x_j}{\sum_{j=1}^n \beta_j}\Big) \geq \frac{\sum_{j=1}^n \beta_j \varphi(x_j)}{\sum_{j=1}^n \beta_j}. \tag{9}$$

III. Sparse Sensor Schedule

A. Sparse Sensor Selection Problems

Given dynamics (1)-(2), the goal of the *sparse* sensor selection problem is to design a sparse output/sensor schedule such that the observability performance metric of the original fully sensed and the sparse dynamics are close in an appropriately defined sense. We also try to keep the number of active sensors much less than the fully sensed system in the output schedule. In other words, given a

canonical discrete-time LTI dynamics (1)-(2), which has p sensors, the observability systemic metric $\rho(\cdot)$ that is aligned with the properties addressed in Definition 1, and the t-step observability Gramian matrix of the fully sensed dynamics, $\mathcal{Q}(t)$, the goal is to find a sensor schedule such that the resulting sparse system with the t-step observability Gramian matrix $\mathcal{Q}_s(t)$ is *well-approximated*, i.e.,

$$\left| \log \frac{\rho(\mathcal{Q}_s(t))}{\rho(\mathcal{Q}(t))} \right| \leq \epsilon', \tag{10}$$

where $\epsilon' > 0$ is the approximation factor.

B. Weighted Sensor Schedule

A weighted schedule can be obtained if we scale the output signal by a non-negative factor while keeping the scales bounded. The scalars introduce an extra degree of freedom that allow us to obtain a sparser sensor set. Given (2), we mathematically define a weighted sensor schedule $S = [s_{j,k+1}]$ and scalars $s_{j,k+1} \geq 0$ where $j \in [p]$ and $k+1 \in [t]$. The resulting output dynamics for this schedule are

$$\boldsymbol{y}(k) = \sum_{j \in [p]} s_{j,k+1} \cdot \boldsymbol{e}_j \boldsymbol{c}_j^\top \boldsymbol{x}(k), \ k \in \mathbb{Z}_+, \tag{11}$$

where $\boldsymbol{c}_j^\top$'s are rows of output matrix $C \in \mathbb{R}^{p \times n}$, $\boldsymbol{e}_j$'s are the standard basis for $\mathbb{R}^p$, and $s_{j,k+1} \geq 0$ shows the strength of the j-th output signal at time k. Moreover, the t-step observability Gramian matrix for the sparse dynamics (11) is given by

$$\begin{aligned} \mathcal{Q}_s(t) &= \Big(\operatorname{diag}(\operatorname{vec}(S)) \cdot \mathcal{O}(t) \Big)^\top \underbrace{\Big(\operatorname{diag}(\operatorname{vec}(S)) \cdot \mathcal{O}(t) \Big)}_{\coloneqq \mathcal{O}_s(t)} \\ &= \sum_{i=1}^{tp} (\operatorname{vec}(S)(i))^2 \boldsymbol{o}_i \boldsymbol{o}_i^\top = \sum_{i=1}^{tp} c_i M_i, \end{aligned} \tag{12}$$

where $\mathcal{O}_s(t)$ is the t-step sparse observability matrix, $\boldsymbol{o}_i^\top$ is the i-th row of the observability matrix $\mathcal{O}(t)$, c_i is the square of the i-th entery of $\operatorname{vec}(S)$, and M_i denotes the rank-1 matrix built by $\boldsymbol{o}_i^\top$.

The ultimate goal in sparse sensor selection problems is to decrease the number of active sensors on average d, where

$$d \coloneqq \frac{1}{t} \cdot \|\operatorname{vec}(S)\|_0, \tag{13}$$

such that the t-step observability Gramian of the fully sensed and sparse system are close. Obtaining this approximate sparse system needs horizon length that is potentially longer than the dimension of the state. We borrow the following approximation definition from [18].

Definition 3 ((ϵ, d)-approximation): Given a time horizon $t \geq n$, dynamics (11) with the sparse weighted sensor schedule S is an (ϵ, d)-approximation of dynamics (2), if and only if

$$(1-\epsilon)\mathcal{Q}(t) \preceq \mathcal{Q}_s(t) \preceq (1+\epsilon)\mathcal{Q}(t), \tag{14}$$

where $\mathcal{Q}(t)$ and $\mathcal{Q}_s(t)$ are the t-step observability Gramian matrices for fully sensed and sparse dynamics, respectively. Parameter d, defined in (13), is the average number of active sensors, and finally $\epsilon \in (0,1)$ is the approximation factor. Succinctly, $\mathcal{Q}_s(t) \approx_{\epsilon,d} \mathcal{Q}(t)$ denotes the same condition in this paper.

Remark 3: When ϵ is small enough [2], (ϵ, d)-approximation is a necessary condition for being well-approximated one. Specifically, if $\mathcal{Q}_s(t) \approx_{\epsilon,d} \mathcal{Q}(t)$, then

$$\left| \log \frac{\rho(\mathcal{Q}_s(t))}{\rho(\mathcal{Q}(t))} \right| \leq \epsilon. \tag{15}$$

To obtain (15), we utilize the facts that $e^{-\beta}$ is almost $1-\beta$ when β is appropriately small, and $1+\beta \leq e^\beta$ for all $\beta \in \mathbb{R}$.

Remark 4: Based on (14), we note that the ranks of Gramian matrices $\mathcal{Q}(t)$ and $\mathcal{Q}_s(t)$ are the same. Therefore, the resulting (ϵ, d)-approximation remains observable if the original dynamics are observable.

One can apply the whitening similarity transformation set forth in Definition 2 to reduce (14) to

$$(1-\epsilon)I \preceq \underbrace{\sum_{i=1}^{tp} c_i \overbrace{\hat{\boldsymbol{o}}_i \hat{\boldsymbol{o}}_i^\top}^{\coloneqq \hat{M}_i}}_{\coloneqq \hat{\mathcal{Q}}_s(t)} \preceq (1+\epsilon)I, \tag{16}$$

where $\hat{\boldsymbol{o}}_i^\top$ is the i-th row of the transformed (whitened) observability matrix $\hat{\mathcal{O}}(t) \coloneqq \mathcal{O}(t)\mathcal{Q}(t)^{-1/2}$, $\hat{M}_i$ is the rank-1 matrix of $\hat{\boldsymbol{o}}_i^\top$, $\hat{\mathcal{Q}}_s(t)$ is the whitened sparse observability Gramian, and $c_i = (\operatorname{vec}(S)(i))^2$.

Note: There is a one to one correspondence between the the set of sensors expanding in time and the rows of any types (original, whitened, and normalized whitened) of the t-step observability matrix. Therefore, we use the phrases of "row selection" and "sensor selection" interchangeably in this paper.

After this reduction, the problem of finding a sparse sensor schedule that is (ϵ, d)-approximation of the fully sensed dynamics boils down to choose and potentially rescale td vectors of the set $\hat{\mathsf{M}} \coloneqq \{\hat{\boldsymbol{o}}_i^\top\}_{i=1}^{tp}$ such that (16) holds. We claim that the size of each member of $\hat{\mathsf{M}}$ is less than one. To validate our claim, let us obtain the leverage score of $\boldsymbol{o}_i^\top$ using the formulation discussed in II-C.3

$$\tau_i = \boldsymbol{o}_i^\top \mathcal{Q}(t)^{-1} \boldsymbol{o}_i = \operatorname{Trace} \hat{M}_i = \|\hat{\boldsymbol{o}}_i\|, \tag{17}$$

for all i in $\{1, \ldots, tp\}$. Since the leverage score is bounded by one (see Remark 1), the immediate observation is $\|\hat{\boldsymbol{o}}_i\| \leq 1$, which gives us the claim. For the purpose of this paper, we prefer to normalize the members of the search space, $\hat{\mathsf{M}}$. Let $\overline{M}_i$ be the matrix $\hat{M}_i$ that is scaled by it's corresponding leverage score, i.e.,

$$\overline{M}_i \coloneqq \frac{1}{\tau_i} \cdot \hat{M}_i = \Big(\frac{1}{\sqrt{\tau_i}} \hat{\boldsymbol{o}}_i\Big) \cdot \Big(\frac{1}{\sqrt{\tau_i}} \hat{\boldsymbol{o}}_i\Big)^\top = \overline{\boldsymbol{o}}_i \overline{\boldsymbol{o}}_i^\top, \tag{18}$$

[2] which is the case in this paper since $\epsilon \in (0,1)$.

for all $i \in \{1,\dots,tp\}$ where $\overline{\boldsymbol{\sigma}}_i \coloneqq (1/\sqrt{\tau_i}) \cdot \hat{\boldsymbol{\sigma}}_i$. It is easy to check that $\overline{M}_i$ is still rank-1 while $\operatorname{Trace}\overline{M}_i = 1$. Note that $\operatorname{Trace}\overline{M}_i = 1$ implies that $\|\overline{\boldsymbol{\sigma}}_i\| = 1$ for all $i \in \{1,\dots,tp\}$. This normalization opens the opportunity to select vectors only based on their directions. We will have enough freedom to appropriately scale them later since this is a weighted schedule problem. One can show $\overline{\boldsymbol{\sigma}}_i$'s are the rows of the *normalized whitened* observability matrix $\overline{\mathcal{O}}(t) \coloneqq \operatorname{diag}(\tau_1,\dots,\tau_{tp})^{-1/2}\hat{\mathcal{O}}(t)$ where $\tau_i = \tau(\boldsymbol{\sigma}_i^\top)$ for all $i \in [tp]$. Furthermore, $\sum_{i=1}^{tp}\hat{M}_i = \sum_{i=1}^{tp}\tau_i \cdot \overline{M}_i = I$.

After these two rounds of reduction, we finally state the problem of weighted sparse sensor scheduling as the following:

Problem 1: Given a time horizon $t \geq n$, the average number of active sensors d, the approximation factor $\epsilon \in (0,1)$, and the set of rank-1 matrices $\overline{\mathsf{M}} \coloneqq \{\overline{M}_i\}_{i=1}^{tp}$, design a weighted sparse sensor schedule

$$S \in \Big\{ Z \in \mathbb{R}_+^{p\times t} \;:\; \|\operatorname{vec}(Z)\|_0 = td \Big\},$$

such that

$$(1-\epsilon)I \preceq \sum_{i=1}^{tp} \overline{c}_i \overline{M}_i \preceq (1+\epsilon)I, \tag{19}$$

where $\overline{c}_i = \tau_i \cdot (\operatorname{vec}(S)(i))^2$.

We notice that constraint (19) can equivalently be rewritten as $(1-\epsilon) \leq \lambda_j(\sum_{i=1}^{tp}\overline{c}_i\overline{M}_i) \leq (1+\epsilon)$ for all $j \in [n]$ where $\lambda_j(\cdot)$ gives the j-th eigenvalue.

The authors of [21] show that the capability of *swapping regret minimization* to solve a single-sided version[3] of Problem 1. A regret sequence was maintained in their algorithm to eventually control the lower bound of the eigenvalues of the t-step sparse observability Gramian. Thus, a natural idea to solve Problem 1 with two-sided constraint is to maintain two regret sequences to simultaneously control the lower and upper bounds of the eigenvalues. This is exactly what we will do in the next section to obtain a solution.

IV. Regret Minimization Solution

Regret minimization is an online optimization framework. The problem of online sequential decision making can be cast as a two-player repeated game, where the environment (adversary) is seen as the opponent. At each stage, the decision maker (player) takes an action from a set of possible actions to allegedly minimizes a cost (or maximizes a reward) that is a function of its action and of the action of the adversary. Adversary's action is independently selected so as to supposedly maximize the cost. We assume that nothing is known a priori about the evolution law for the actions of the adversary, which in particular may depend on the actions of the player and/or on an unobservable state of the adversary, and be nonstationary.

Unfortunately, due to the lack of knowledge about the adversary, certain objectives such as minimizing expected cost raise some controversy as to how one should define the expectation. However, a popular criterion in online learning is to minimize the regret of the player where regret of the player is defined as the cost that could have been accumulated by the player's actions compared to the cost of the best fixed action in hindsight.

In the next section, we will explain the regret minimization over density matrices. The materials are adopted from [26], [27].

A. Regret Minimization over Density Matrices

In this type of regret minimization game, at each stage $\ell \in \{0,\dots,T-1\}$, the player picks an action A_ℓ from the set of density matrices

$$\Omega_n \coloneqq \Big\{ D \in \mathbb{S}_+^n \;:\; \operatorname{Trace} D = 1 \Big\}, \tag{20}$$

also known as *action space*. The player then receives a symmetric feedback matrix F_ℓ and tolerates a loss defined as $\langle A_\ell, F_\ell\rangle$. At this moment, the feedback matrix F_ℓ is available to the player without any limitation. The goal of the player is to minimize the regret with respect to the best fixed action in hindsight, $\mathfrak{U}_{\text{best}} \coloneqq \operatorname{arginf}_{\mathfrak{U}\in\Omega_n} \sum_{\ell=0}^{T-1}\langle \mathfrak{U}, F_\ell\rangle$, defined as

$$\begin{aligned}\operatorname{Regret}(\mathfrak{U}_{\text{best}}) &\coloneqq \sum_{\ell=0}^{T-1}\langle A_\ell, F_\ell\rangle - \sum_{\ell=0}^{T-1}\langle \mathfrak{U}_{\text{best}}, F_\ell\rangle \\ &= \sum_{\ell=0}^{T-1}\langle A_\ell, F_\ell\rangle - \lambda_{\min}\Big(\sum_{\ell=0}^{T-1} F_\ell\Big),\end{aligned} \tag{21}$$

where T is the total number of iteration in the game. The second equality in (21) holds since

$$\inf_{\mathfrak{U}\in\Omega_n}\sum_{\ell=0}^{T-1}\langle \mathfrak{U}, F_\ell\rangle = \inf_{\mathfrak{U}\in\Omega_n}\langle \mathfrak{U}, \sum_{\ell=0}^{T-1} F_\ell\rangle = \lambda_{\min}\Big(\sum_{\ell=0}^{T-1} F_\ell\Big),$$

where the first equality holds because the sum of traces is equal to the trace of sum, and second holds due to Lemma 1. An interesting interpretation of this result is: the total loss for the best action $\mathfrak{U}_{\text{best}}$ can be obtained as the rank-1 projection over the minimum eigenvector of $\sum_{\ell=0}^{T-1} F_\ell$.

A popular strategy to minimize the regret is *Follow-The-Regularized-Leader* (FTRL). Based on this strategy, the player should pick action

$$A_\ell \coloneqq \operatorname*{argmin}_{\mathcal{Z}\in\Omega_n}\Big\{ r(\mathcal{Z}) + \alpha\sum_{j=0}^{\ell-1}\langle \mathcal{Z}, F_j\rangle \Big\},$$

at stage ℓ where $r(\cdot)$ is a strongly convex function over the simplex known as *regularizer*, and $\alpha > 0$ is the learning rate that balances the loss and the regularization. One can interpret FTRL as the trade-off between minimizing the accumulated loss happened thus far and the value of the

[3]Single-sided is the same problem as Problem 1, but the two-sided constraint (19) is replaced by a single-sided constraint, e.g., $(1-\epsilon)I \preceq \sum_{i=1}^{tp}\overline{c}_i\overline{M}_i$.

regularizer. Different choices for the regularization result in different strategies. Similar to [21], we utilize the $\ell_{1/2}$-*regularizer*, $r(\mathcal{Z}) \coloneqq -2\operatorname{Trace}\mathcal{Z}^{1/2}$ that was proposed for the first time in [26]. Using the $\ell_{1/2}$-regularizer, the player should play the closed-form action

$$A_\ell = \Big(\alpha \sum_{j=0}^{\ell-1} F_j - v_\ell I\Big)^{-2}, \tag{22}$$

where $v_\ell \in \mathbb{R}$ is a unique constant that ensures $\alpha \sum_{j=0}^{\ell-1} F_j - v_\ell I \succ 0$ and $\operatorname{Trace} A_\ell = 1$. We succinctly refer to (22) as $\ell_{1/2}$-strategy in the rest of this paper.

The upper bound for the regret of the $\ell_{1/2}$-strategy (22) is obtained in [26, Theorem 3.2 and 3.3] and repeated in the following Lemma.

Lemma 2 (Regret Upper Bound): Assume in the regret game (21), the player picks actions based on $\ell_{1/2}$-strategy (22), and let $\alpha > 0$ be the learning rate and F_ℓ be a rank-1 feedback matrix satisfying $\langle A_\ell^{1/2}, \alpha F_\ell\rangle \geq -1/2$ for all ℓ, then, for every $\mathfrak{U} \in \Omega_n$

$$\sum_{\ell=0}^{T-1}\langle F_\ell, A_\ell - \mathfrak{U}\rangle \leq 2\alpha \cdot \sum_{\ell=0}^{T-1}\langle A_\ell, F_\ell\rangle \cdot \langle A_\ell^{1/2}, F_\ell\rangle + \frac{\psi_{A_0}(\mathfrak{U})}{\alpha}, \tag{23}$$

where

$$\psi_X(Y) \coloneqq \langle X^{-1/2}, Y\rangle + \operatorname{Trace} X^{1/2} - 2\operatorname{Trace} Y^{1/2}, \tag{24}$$

is the so-called *Bregman divergence* for the $\ell_{1/2}$-regularizer; moreover, $\psi_{A_0}(\mathfrak{U}) \leq 2\sqrt{n}$.

Remark 5: Lemma 2 specifies regret with respect to any fixed matrix $\mathfrak{U} \in \Omega_n$ and not only to the best fixed action $\mathfrak{U}_{\text{best}}$ in hindsight.

In the next, we will finally use the regret game discussed in this section to develop a solution for Problem 1.

B. A Linear-sized Regret Solution

The authors of [21] show that a sequence of actions in a regret game is able to control the minimum eigenvalue of the sparse observability Gramian. Therefore, we aim to maintain two sequences of action matrices to simultaneously control both the minimum and maximum eigenvalues of the sparse matrix as required by (19).

We define the action space Ω_n as the set of all n-by-n positive semi-definite matrices that their trace is equal to one. We define actions A_ℓ and B_ℓ for $\ell \in \{0, \dots, T-1\}$ based on $\ell_{1/2}$-strategy (22) as the two sequences of actions that we maintain during the game to control the range of eigenvalues. Conversely, at each round ℓ, the adversary picks the vector $\overline{\boldsymbol{\theta}}_{k\ell}^\top$ of the set of rows $\{\overline{\boldsymbol{\theta}}_i^\top\}_{i=1}^{tp}$ such that for $\overline{M}_{k\ell} = \overline{\boldsymbol{\theta}}_{k\ell}\overline{\boldsymbol{\theta}}_{k\ell}^\top$ we have

$$\langle \overline{M}_{k\ell}, A_\ell\rangle \leq \langle \overline{M}_{k\ell}, B_\ell\rangle, \tag{25}$$

By using *averaging arguments*, it is guaranteed that for each round ℓ, the adversary can always find a vector $\overline{\boldsymbol{\theta}}_{k\ell}^\top$ such that

$$\begin{aligned}\langle \overline{M}_{k\ell}, A_\ell\rangle &\leq \sum_{i\in[tp]} \frac{\tau_i}{n}\cdot\langle \overline{M}_i, A_\ell\rangle = \frac{1}{n}\langle I, A_\ell\rangle = \frac{1}{n}\operatorname{Trace} A_\ell \\ &= \frac{1}{n}\operatorname{Trace} B_\ell = \sum_{i\in[tp]}\frac{\tau_i}{n}\cdot\langle \overline{M}_i, B_\ell\rangle \leq \langle \overline{M}_{k\ell}, B_\ell\rangle;\end{aligned} \tag{26}$$

therefore, playing the strategy (25) is always possible for the adversary. In (26), we use $\sum_{i=1}^{tp}\tau_i \cdot \overline{M}_i = I$ and the fact that $\operatorname{Trace} D = 1$ for any $D \in \Omega_n$.

The adversary then reflects the respective feedback matrices $F_\ell^A \coloneqq \beta_\ell^A \cdot \overline{M}_{k\ell}$ and $F_\ell^B \coloneqq \beta_\ell^B \cdot \overline{M}_{k\ell}$ to maximize the costs $\langle F_\ell^A, A_\ell\rangle$ and $\langle F_\ell^B, B_\ell\rangle$ at round ℓ. Lemma 3 shows how to define β_ℓ^A and β_ℓ^B for our purpose such that the prerequisite of Lemma 2 is satisfied. We will later use these two coefficients to build our final regret setup that solves Problem 1. The following result is needed to prove Lemma 3.

Proposition 1: For any D in Ω_n and every $\overline{M}_i$ defined in (18), the following holds

$$\langle \overline{M}_i, D^{1/2}\rangle \leq \langle \overline{M}_i, D\rangle^{1/2}. \tag{27}$$

Proof: Let $U\Lambda U^\top$ be the eigen decomposition for matrix D, $\Lambda = \operatorname{diag}(\lambda_1, \dots, \lambda_n)$, and $\overline{M}_i = \overline{\boldsymbol{\theta}}_i\overline{\boldsymbol{\theta}}_i^\top$. Therefore,

$$\langle \overline{M}_i, D\rangle = \operatorname{Trace}(U^\top\overline{\boldsymbol{\theta}}_i)^\top\Lambda\underbrace{(U^\top\overline{\boldsymbol{\theta}}_i)}_{\coloneqq\check{\boldsymbol{\theta}}_i} = \lambda_1\check{\boldsymbol{\theta}}_{1,i}^2 + \cdots + \lambda_n\check{\boldsymbol{\theta}}_{n,i}^2,$$

where $\check{\boldsymbol{\theta}}_i = (\check{\boldsymbol{\theta}}_{1,i}, \dots, \check{\boldsymbol{\theta}}_{n,i})^\top$. Using the same arguments, one can show that $\langle \overline{M}_i, D^{1/2}\rangle = \sqrt{\lambda_1}\check{\boldsymbol{\theta}}_{1,i}^2 + \cdots + \sqrt{\lambda_n}\check{\boldsymbol{\theta}}_{n,i}^2$. In addition, $\check{\boldsymbol{\theta}}_{1,i}^2 + \cdots + \check{\boldsymbol{\theta}}_{n,i}^2 = \operatorname{Trace}\check{\boldsymbol{\theta}}_i^\top\check{\boldsymbol{\theta}}_i = \operatorname{Trace}\overline{\boldsymbol{\theta}}_i^\top UU^\top\overline{\boldsymbol{\theta}}_i = 1$. Thus, (27) can be refashioned as

$$\sqrt{\frac{\sum_{j=1}^n \check{\boldsymbol{\theta}}_{j,i}^2\lambda_j}{\sum_{j=1}^n\check{\boldsymbol{\theta}}_{j,i}^2}} \geq \frac{\sum_{j=1}^n\check{\boldsymbol{\theta}}_{j,i}^2\sqrt{\lambda_j}}{\sum_{j=1}^n\check{\boldsymbol{\theta}}_{j,i}^2}, \tag{28}$$

for all i in $[tp]$, and (28) holds since it is a Jensen's inequality (see Fact 1) for the choice of $\sqrt{\cdot}$ as the real concave function $\varphi(\cdot)$ over the positive numbers $\check{\boldsymbol{\theta}}_{1,i}^2, \dots, \check{\boldsymbol{\theta}}_{n,i}^2$. ∎

Lemma 3: Taking $\beta_\ell^A = -\langle A_\ell, \overline{M}_{k\ell}\rangle^{-1/2}$, $\beta_\ell^B = \langle B_\ell, \overline{M}_{k\ell}\rangle^{-1/2}$, and $\alpha \in (0, 1/2)$ ensure that the prerequisite of Lemma 2 is satisfied for both sequences.

Proof: To show for the sequence A_ℓ, i.e., to prove $\alpha\cdot\langle F_\ell^A, A_\ell^{1/2}\rangle > -1/2$ holds, we can write

$$\alpha\cdot\langle F_\ell^A, A_\ell^{1/2}\rangle = \alpha\cdot\Big\langle -\frac{\overline{M}_{k\ell}}{\langle A_\ell, \overline{M}_{k\ell}\rangle^{1/2}}, A_\ell^{1/2}\Big\rangle \overset{\text{(a)}}{\leq} -\alpha \leq -\frac{1}{2},$$

where inequality (a) holds due to Proposition 1. To obtain the result for B_ℓ, we know B_ℓ and so $B_\ell^{1/2}$ are in S_+^n; therefore, $\langle B_\ell, \overline{M}_{k\ell}\rangle = \overline{\boldsymbol{\theta}}_{k\ell}^\top B_\ell\overline{\boldsymbol{\theta}}_{k\ell}$ and similarly $\langle B_\ell^{1/2}, \overline{M}_{k\ell}\rangle$ are non negative which means $\langle F_\ell^B, B_\ell^{1/2}\rangle \geq 0$. Finally, since $\alpha \in (0, 1/2)$, then $\alpha\cdot\langle F_\ell^B, B_\ell^{1/2}\rangle \geq 0$. ∎

Theorem 1 (Linear-sized Regret Solution): Given the time horizon $t \geq n$, dynamics (6), and the approximation factor $\epsilon \in (0, 1/4)$, $T = n/\epsilon^2$ rounds of the regret game (21)

as described below deterministically constructs a weighted sensor schedule such that the resulting dynamics (12) is an $(\epsilon, T/t)$-approximation of dynamics (1)-(2).

Regret game setup: At each round $\ell \in \{0, \dots, T-1\}$, the player picks two different actions A_ℓ and B_ℓ based on $\ell_{1/2}$-strategy (22) for some $\alpha \in (0, 1/4)$ to separately minimize the costs $\langle F_\ell^A, A_\ell \rangle$ and $\langle F_\ell^B, B_\ell \rangle$, respectively. At every round ℓ, the adversary then chooses vector $\overline{\mathbf{o}}_{k\ell}^\top$ from the row vector set $\{\overline{\mathbf{o}}_i^\top\}_{i=1}^{tp}$ such that (25) holds and then reflects the respective feedback matrices $F_\ell^A = -\langle A_\ell, \overline{M}_{k\ell} \rangle^{-1/2} \cdot \overline{M}_{k\ell}$ and $F_\ell^B = \langle B_\ell, \overline{M}_{k\ell} \rangle^{-1/2} \cdot \overline{M}_{k\ell}$ to maximize the costs.

Due to space limitations, we omit the proof of Theorem 1 in this paper. The proof is technical and lengthy, requiring more than a page of space. We plan to provide the full proof in the extended version of the paper, which will be submitted to a journal.

The sensor schedule resulting from Theorem 1 has, on average, $n/t\epsilon^2$ active sensors. However, note that this value represents the most pessimistic estimate for the average number of active sensors, since the selection at each round of the game in Theorem 1 is made with replacement. As a result, it's possible for the same sensor to be chosen more than once during the game, potentially reducing the number of sampled individual sensors as well as the average number of active sensors d

Theorem 1 provides a linear-sized sparse schedule with n/ϵ^2 active sensors, which yields an (ϵ, d)-approximation of the fully sensed network. We note that these results improve upon the randomized results of [18, Theorem 2] for (ϵ, d)-approximation, as their algorithm requires $O(n \log n/\epsilon^2)$ active sensors, which contains an additional factor of $\log n$, as well as a potentially large universal constant.

V. Concluding Remarks

Inspired by a growing body of work on online learning and graph sparsification, this paper presents a simple deterministic framework for designing a linear-sized time-varying sensor schedule for large-scale LTI networks. Our proposed method uses regret minimization as an online optimization framework to maintain two sequences of actions over density matrices. We provide a regret bound that yields a sparse schedule, which on average samples a constant number of active sensors to approximate a wide range of observability measures of the fully sensed networks up to a multiplicative factor. Specifically, our proposed framework samples at most n/ϵ^2 sensors, which removes the $\log n$ factor from the sample complexity of the best available competitive result. We also note that similar results can be obtained for the actuator scheduling problem.

References

[1] K. Manohar, B. W. Brunton, J. N. Kutz, and S. L. Brunton, "Data-driven sparse sensor placement for reconstruction: Demonstrating the benefits of exploiting known patterns," *IEEE Control Systems Magazine*, vol. 38, no. 3, pp. 63–86, 2018.

[2] M. Athans, "On the determination of optimal costly measurement strategies for linear stochastic systems," *IFAC Proceedings Volumes*, vol. 5, no. 1, pp. 303–313, 1972.

[3] P. Müller and H. Weber, "Analysis and optimization of certain qualities of controllability and observability for linear dynamical systems," *Automatica*, vol. 8, no. 3, pp. 237–246, 1972.

[4] M. Morari and G. Stephanopoulos, "Minimizing unobservability in inferential control schemes," *International Journal of Control*, vol. 31, no. 2, pp. 367–377, 1980.

[5] D. Ge, X. Jiang, and Y. Ye, "A note on the complexity of l p minimization," *Mathematical programming*, vol. 129, no. 2, pp. 285–299, 2011.

[6] A. Olshevsky, "Minimal controllability problems," *IEEE Transactions on Control of Network Systems*, vol. 1, no. 3, pp. 249–258, 2014.

[7] V. Tzoumas, M. A. Rahimian, G. J. Pappas, and A. Jadbabaie, "Minimal actuator placement with bounds on control effort," *IEEE Transactions on Control of Network Systems*, vol. 3, no. 1, pp. 67–78, 2015.

[8] A. Jadbabaie, A. Olshevsky, G. J. Pappas, and V. Tzoumas, "Minimal reachability is hard to approximate," *IEEE Transactions on Automatic Control*, vol. 64, no. 2, pp. 783–789, 2018.

[9] N. E. Leonard and E. Fiorelli, "Virtual leaders, artificial potentials and coordinated control of groups," in *Proceedings of the 40th IEEE Conference on Decision and Control (Cat. No. 01CH37228)*, vol. 3. IEEE, 2001, pp. 2968–2973.

[10] A. Rahmani, M. Ji, M. Mesbahi, and M. Egerstedt, "Controllability of multi-agent systems from a graph-theoretic perspective," *SIAM Journal on Control and Optimization*, vol. 48, no. 1, pp. 162–186, 2009.

[11] J. C. Doyle, "Guaranteed margins for lqg regulators," *IEEE Transactions on automatic Control*, vol. 23, no. 4, pp. 756–757, 1978.

[12] J. Doyle and G. Stein, "Multivariable feedback design: Concepts for a classical/modern synthesis," *IEEE transactions on Automatic Control*, vol. 26, no. 1, pp. 4–16, 1981.

[13] Y. Susuki and I. Mezic, "Nonlinear koopman modes and a precursor to power system swing instabilities," *IEEE Transactions on Power Systems*, vol. 27, no. 3, pp. 1182–1191, 2012.

[14] K. Khosoussi, G. S. Sukhatme, S. Huang, and G. Dissanayake, "Designing sparse reliable pose-graph slam: A graph-theoretic approach," in *Algorithmic Foundations of Robotics XII*. Springer, 2020, pp. 17–32.

[15] J. L. Proctor and P. A. Eckhoff, "Discovering dynamic patterns from infectious disease data using dynamic mode decomposition," *International health*, vol. 7, no. 2, pp. 139–145, 2015.

[16] T. H. Summers, F. L. Cortesi, and J. Lygeros, "On submodularity and controllability in complex dynamical networks," *IEEE Transactions on Control of Network Systems*, vol. 3, no. 1, pp. 91–101, 2015.

[17] S. N. Ravi, V. Ithapu, S. Johnson, and V. Singh, "Experimental design on a budget for sparse linear models and applications," in *International Conference on Machine Learning*, 2016, pp. 583–592.

[18] M. Siami, A. Olshevsky, and A. Jadbabaie, "Deterministic and randomized actuator scheduling with guaranteed performance bounds," *IEEE Transactions on Automatic Control*, vol. 66, no. 4, pp. 1686–1701, 2020.

[19] Z. Allen-Zhu, Y. Li, A. Singh, and Y. Wang, "Near-optimal design of experiments via regret minimization," in *International Conference on Machine Learning*. PMLR, 2017, pp. 126–135.

[20] A. Olshevsky, "On (non)supermodularity of average control energy," *IEEE Transactions on Control of Network Systems*, 2018, to appear.

[21] R. Vafaee and M. Siami, "Learning-based sensor selection with guaranteed performance bounds," in *2022 American Control Conference (ACC)*. IEEE, 2022, pp. 1459–1465.

[22] K. Manohar, J. N. Kutz, and S. L. Brunton, "Optimal sensor and actuator selection using balanced model reduction," *IEEE Transactions on Automatic Control*, vol. 67, no. 4, pp. 2108–2115, 2021.

[23] M. Siami and A. Jadbabaie, "A separation theorem for joint sensor and actuator scheduling with guaranteed performance bounds," *Automatica*, vol. 119, p. 109054, 2020.

[24] M. Siami and N. Motee, "Network abstraction with guaranteed performance bounds," *IEEE Transactions on Automatic Control*, vol. 63, no. 11, 2018.

[25] M. B. Cohen, Y. T. Lee, C. Musco, C. Musco, R. Peng, and A. Sidford, "Uniform sampling for matrix approximation," in *Proceedings of the 2015 Conference on Innovations in Theoretical Computer Science*, 2015, pp. 181–190.

[26] Z. Allen-Zhu, Z. Liao, and L. Orecchia, "Spectral sparsification and regret minimization beyond matrix multiplicative updates," in *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, 2015, pp. 237–245.

[27] Z. Allen-Zhu, Y. Li, A. Singh, and Y. Wang, "Near-optimal discrete optimization for experimental design: A regret minimization approach," *Mathematical Programming*, vol. 186, no. 1, pp. 439–478, 2021.