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

Deterministic Polynomial-Time Actuator Scheduling With Guaranteed Performance

Milad Siami and Ali Jadbabaie

Abstract— **In this paper, the problem of time-varying actuator selection for linear dynamical systems is investigated. By leveraging recent advances in the graph sparsification literature, we develop a framework for designing a sparse actuator schedule for a given large-scale linear system with guaranteed performance bounds using a polynomial-time algorithm. Current approaches based on polynomial time relaxations of the subset selection problem require an extra multiplicative factor of $\log n$ sensors/actuators times the minimal number in order to just maintain controllability/observability. In contrast, we show that there exists a polynomial-time actuator schedule that on average selects only a constant number of actuators at each time, to approximate the controllability/observability metrics of the system when all actuators/sensors are in use.**

I. Introduction

During the past few years, controllability and observability properties of complex dynamical networks have been subjects of intense study in the controls community [1]–[12]. This interest stems from the need to steer or observe the state of large-scale, networked systems such as power networks [13], social networks, biological and genetic regulatory networks [14]–[16], and traffic networks [17]. While the classical notion of controllability, introduced by Kalman in [18] is quite well understood, the network controllability question and the dependence of various measures of controllability or observability on location of sensors and actuators in a networked control system are not fully understood. Often times, one would like to steer or estimate the state of a large-scale, networked control system with as few actuators/sensors as possible, due to issues related to cost and energy depletion. The focus on control/estimation policies using a sparse set of actuators/sensors is motivated by many application domains, ranging from infrastructures networks (e.g., water and power networks) to genomic networks and living cells. For example, energy conservation through efficient utilization of sensors and actuators can help extend the duration of battery life in networks of mobile sensors and multiagent robotic networks; estimating the whole state of the power grid using fewer measurement units will help reduce the cost of monitoring the network for systemic failures, etc.

It is therefore desirable to have a limited number of sensors and actuators without compromising the control or estimation performance. Unfortunately, as the recent work in [1] has shown, the problem of finding a sparse set of input variables such that the resulting system is controllable is NP-hard. Even the presumably easier problem of approximating the minimum number by a constant multiplicative factor of $\log n$ is also NP-hard. Other results in the literature have studied network controllability by exploring approximation algorithms for the closely related subset selection problem [1], [11], [12].

Previous studies have been mainly focused on solving the optimal sensor/actuator placement problem using greedy heuristics, as approximations of the corresponding sparse-subset selection problem. While these papers attempt to find approximation algorithms for finding the best sparse subset, our focus in this paper is to gain new fundamental insights into approximating various controllability metrics compared to when all possible actuators are chosen. Specifically, we are interested in actuator/sensor schedules that reduce the duty cycle of the actuator (sensor) while ensuring a suitable level of controllability (observability) performance for the entire network. Due to need for energy efficiency, we may want to minimize the number of active actuator/sensors at each time. Yet, we would like to have a performance that closely resembles that of the original system, when all available sensor/actuators are active. We investigate sparse sensor and actuator selection as particular instances where discrete geometric structures can be utilized to study network controllability and observability problems (cf. [19]–[21]). Throughout the rest of the paper, we will focus on the actuator selection problem; the dual notion of sensor selection follows similar ideas. A key observation is the close connection between this problem and some classical problems in statistics such as outlier detection, active learning, and optimal experimental design. In recent years, there has been a renewed interest in optimal experiment design which has a long history going back at least 65 years [22], [23].

We propose an alternative to submodularity-based methods and instead use recent advances in theoretical computer science to develop scalable algorithms for sparsifying control inputs. Our approach allows for using the control energy as objective, while submodularity based approaches fail to guarantee performance; since the trace of inverse Gramian is not submodular [24]. Recently, in [25], we showed that there exists an actuator schedule that on average selects only a constant number of actuators at each time, in order to approximate controllability measures by any desired degree

This research was supported in part by the AFOSR complex networks program and a Vannevar Bush Fellowship.

M. Siami and A. Jadbabaie are with the Institute for Data, Systems, and Society, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: `siami, jadbabai@mit.edu`).

of accuracy. We also showed that there are randomized algorithms that choose on average $\mathcal{O}(\log n)$ actuators to approximate controllability measures (and as a result the Gramian) [25]. In this paper, however, we use the recent results in [26] to improve this result and present a deterministic polynomial time algorithm that chooses only on average a constant number of actuators at each time.

II. Preliminaries and Definitions

A. Mathematical Notations

Throughout the paper, the discrete time index is denoted by k. The sets of real (integer), positive real (integer), and strictly 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 $[n]$. The cardinality of a set σ is denoted by $\mathrm{card}(\sigma)$. Capital letters, such as A or B, stand for real-valued matrices. We use $\mathrm{diag}(x_1, x_2, \ldots, x_n)$ to denote a n-by-n diagonal square matrix with x_1 to x_n on its diagonal. We denote the number of nonzero elements in vector x by $\|x\|_0$. For a square matrix X, $\det(X)$ and $\mathrm{Trace}(X)$ refer to the determinant and the summation of on-diagonal elements of X, respectively. $\mathbb{S}^n_+$ is the positive definite cone of n-by-n matrices. The n-by-n identity matrix is denoted by I_n. Notation $A \preceq B$ is equivalent to matrix $B - A$ being positive semi-definite. The transpose of matrix A is denoted by $A^\top$. The rank, kernel and image of matrix A are referred to by $\mathrm{rank}(A)$, $\ker(A)$ and $\mathrm{Im}(A)$, respectively. Finally, the Moore-Penrose pseudo-inverse of matrix A is denoted by $A^\dagger$.

B. Linear Systems and Controllability

We start with the canonical linear discrete-time, time-invariant dynamics

$$x(k+1) \ = \ A\,x(k) \ + \ B\,u(k),$$

where $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$ and $k \in \mathbb{Z}_+$. The state matrix A describes the underlying structure of the system and the interaction strength between the agents, and matrix B identifies the nodes controlled by an outside controller. One can rewrite the dynamics as follows

$$x(k+1) \ = \ A\,x(k) \ + \ \sum_{i \in [m]} b_i\,u_i(k), \tag{1}$$

where b_i's are columns of matrix $B \in \mathbb{R}^{n \times m}$. Then, the controllability matrix at time t is given by

$$\mathcal{C}(t) \ = \ \left[B \ AB \ A^2B \ \cdots \ A^{t-1}B\right]. \tag{2}$$

It is well-known that from a numerical standpoint it is better to characterize controllability in terms of the Gramian matrix at time t defined as follows:

$$\begin{aligned}\mathcal{W}(t) \ &= \ \sum_{i=0}^{t-1} A^iBB^\top (A^i)^\top \\ &= \ \mathcal{C}(t)\,\mathcal{C}^\top(t).\end{aligned} \tag{3}$$

When looking at time-varying input schedules, we will consider the following linear system with time-varying input matrix $\mathcal{B}(.)$

$$x(k+1) \ = \ A\,x(k) \ + \ \mathcal{B}(k)\,u(k). \tag{4}$$

For the above system, the controllability and Gramian matrices at time step t are defined as

$$\mathcal{C}_*(t) \ = \ \left[\mathcal{B}(t-1) \ A\mathcal{B}(t-2) \ A^2\mathcal{B}(t-3) \ \cdots \ A^{t-1}\mathcal{B}(0)\right],$$

and

$$\mathcal{W}_*(t) \ = \ \mathcal{C}_*(t)\,\mathcal{C}_*^\top(t), \tag{5}$$

respectively.

Assumption 1: Throughout the paper, we assume that the system (1) is controllable (i.e., the controllability matrix has full row rank and the Gramian is positive definite). However, all results presented in this paper can be modified/extended to uncontrollable systems.

C. Controllability Metrics

Typically, a controllability metric is an operator $\rho : \mathbb{S}^n_+ \to \mathbb{R}$ that maps from the set of Gramian matrices of all controllable networks over n to a real number. For many popular choices of ρ, one can see that they satisfy the properties presented below (cf. [27]–[29])

(i) *Homogeneity:* For all $\alpha > 0$,

$$\rho(\alpha A) \ = \ \alpha^{-1}\rho(A);$$

(ii) *Monotonicity:* If $B \preceq A$, then

$$\rho(A) \ \leq \ \rho(B).$$

Some of the important examples of controllability metrics which satisfy Properties 1 and 2 are listed below

- Average control energy: $\mathrm{Trace}\left(\mathcal{W}^{-1}(t)\right)$;
- The volume of the ellipsoid: $(\det \mathcal{W}(t))^{-1/n}$;
- Inverse of the trace: $1/\,\mathrm{Trace}(\mathcal{W}(t))$;
- Inverse of the minimum eigenvalue: $1/\lambda_{\min}(\mathcal{W}(t))$.

Assumption 2: Throughout the paper, we assume that all controllability metrics satisfy Properties 1 and 2.

D. Matrix Reconstruction and Sparsification

In this part, we present a result from the sparsification literature which we use later in our algorithms to find sparse actuator schedules.

Lemma 1 (Dual Set Spectral Sparsification [26]): Let $V = \{v_1, \ldots, v_t\}$ and $U = \{u_1, \ldots, u_t\}$ be two equal cardinality decompositions of identity matrices (i.e., $\sum_{i=1}^t v_iv_i^\top = I_n$ and $\sum_{i=1}^t u_iu_i^\top = I_\ell$) where $v_i \in \mathbb{R}^n$ ($n < t$) and $u_i \in \mathbb{R}^\ell$ ($\ell \leq t$). Given an integer κ with $n < \kappa \leq t$, Algorithm 1 computes a set of weights $c_i \geq 0$ where $i \in [t]$, such that

$$\lambda_{\min}\left(\sum_{i=1}^t c_iv_iv_i^\top\right) \geq \left(1 - \sqrt{\frac{n}{\kappa}}\right)^2,$$

$$\lambda_{\max}\left(\sum_{i=1}^t c_iu_iu_i^\top\right) \leq \left(1 + \sqrt{\frac{\ell}{\kappa}}\right)^2,$$

and

$$\mathrm{card}\,\{i \ : \ c_i > 0, i \in [t]\} \ \leq \ \kappa.$$

Algorithm 1: A Deterministic Dual Set Spectral Sparsification $\mathrm{DualSet}(V, U, \kappa)$.

Input : $V = [v_1, \dots, v_t]$, with $VV^\top = I_n$
$U = [u_1, \dots, u_t]$, with $UU^\top = I_\ell$
$\kappa \in \mathbb{Z}_+$, with $n < \kappa \leq t$

Output: $c = [c_1, c_2, \dots, c_t] \in \mathbb{R}_+^{1\times t}$ with $\|c\|_0 \leq \kappa$

Set $c(0) = 0_{t\times 1}$, $\underline{\mathcal{A}}(0) = 0_{n\times n}$, $\bar{\mathcal{A}}(0) = 0_{\ell\times\ell}$, $\underline{\delta} = 1$, $\bar{\delta} = \frac{1+\sqrt{\frac{\ell}{\kappa}}}{1-\sqrt{\frac{n}{\kappa}}}$

for $\tau = 0 : \kappa - 1$ **do**

- $\underline{\mu}(\tau) = \tau - \sqrt{\kappa n}$
- $\bar{\mu}(\tau) = \bar{\delta}\left(\tau + \sqrt{\kappa\ell}\right)$
- Find an index j such that
$$\mathfrak{U}(u_j, \bar{\delta}, \bar{\mathcal{A}}(\tau), \bar{\mu}(\tau)) \leq \mathfrak{L}(v_j, \underline{\delta}, \underline{\mathcal{A}}(\tau), \underline{\mu}(\tau))$$
- Set
$$\Delta = 2\big(\mathfrak{U}(u_j, \bar{\delta}, \bar{\mathcal{A}}(\tau), \bar{\mu}(\tau)) + \mathfrak{L}(v_j, \underline{\delta}, \underline{\mathcal{A}}(\tau), \underline{\mu}(\tau))\big)^{-1}$$
- Update the j-th component of $c(\tau)$:
$$c(\tau+1) = c(\tau) + \Delta \mathrm{e}_j,$$
- $\underline{\mathcal{A}}(\tau+1) = \underline{\mathcal{A}}(\tau) + \Delta v_j v_j^\top$
- $\bar{\mathcal{A}}(\tau+1) = \bar{\mathcal{A}}(\tau) + \Delta u_j u_j^\top$

end

return $c = \kappa^{-1}\left(1 - \sqrt{\frac{n}{\kappa}}\right) c(\kappa)$

Algorithm 1 greedily selects vectors that satisfy a number of desired properties in each step. These properties will eventually imply the desired bounds on eigenvalues. In Algorithm 1, two parameters $\mathfrak{L}$ and $\mathfrak{U}$ are defined as follows:

$$\mathfrak{L}(v, \underline{\delta}, \underline{\mathcal{A}}, \underline{\mu}) = \frac{v^\top \left(\underline{\mathcal{A}} - (\underline{\mu}+\underline{\delta})I_n\right)^{-2} v}{\underline{\phi}(\underline{\mu}+\underline{\delta}, \underline{\mathcal{A}}) - \underline{\phi}(\underline{\mu}, \underline{\mathcal{A}})} - v^\top \left(\underline{\mathcal{A}} - (\underline{\mu}+\underline{\delta})I_n\right)^{-1} v,$$

and

$$\mathfrak{U}(u, \bar{\delta}, \bar{\mathcal{A}}, \bar{\mu}) = \frac{u^\top((\bar{\mu}+\bar{\delta})I_\ell - \bar{\mathcal{A}})^{-2} u}{\bar{\phi}(\bar{\mu}, \bar{\mathcal{A}}) - \bar{\phi}(\bar{\mu}+\bar{\delta}, \bar{\mathcal{A}})} + u^\top \left((\bar{\mu}+\bar{\delta})I_\ell - \bar{\mathcal{A}}\right)^{-1} u,$$

where

$$\underline{\phi}(\underline{\mu}, \underline{\mathcal{A}}) = \sum_{i=1}^{n} \frac{1}{\lambda_i(\underline{\mathcal{A}}) - \underline{\mu}},$$

and

$$\bar{\phi}(\bar{\mu}, \bar{\mathcal{A}}) = \sum_{i=1}^{\ell} \frac{1}{\bar{\mu} - \lambda_i(\bar{\mathcal{A}})}. \tag{6}$$

This algorithm is a generalization of an algorithm from [30] which is deterministic and at most needs $\mathcal{O}\left(\kappa t(n^2+\ell^2)\right)$. Furthermore, the algorithm needs $\mathcal{O}(\kappa t n^2)$ operations if U contains the standard basis of $\mathbb{R}^t$; we refer the reader to [26] for more details.

We denote the application of the algorithm to V and U by

$$[c_1, c_2, \cdots, c_t] = \mathrm{DualSet}(V, U, \kappa).$$

In the next section, we show how various controllability measures can be approximated by selecting a sparse set of actuators.

III. Sparse Actuator Selection Problems

For a given linear system (1) with a general underlying structure, the actuator scheduling problem seeks to construct a schedule of the control inputs that keeps the number of active actuators much less than the original network such that the controllability matrices of the original and the new networks are similar in an appropriately defined sense. Specifically, given a canonical linear, time-invariant system (1) with m actuators and controllability Gramian matrix $\mathcal{W}(t)$ at time t, our goal is to find a *sparse* actuator schedule such that the resulted network with controllability Gramian $\mathcal{W}_s(t)$ is well-approximated, i.e.,

$$\left|\frac{\rho\left(\mathcal{W}(t)\right) - \rho\left(\mathcal{W}_s(t)\right)}{\rho\left(\mathcal{W}(t)\right)}\right| \leq \epsilon, \tag{7}$$

where ρ is any monotone, homogeneous controllability measure that quantifies the difficulty of the control problem for example as a function of the required control energy. The controllability metrics are defined based on the controllability Gramian, therefore "close" Gramian matrices result in approximately same values. Our goal here is to answer the following questions:

- What is the relation between the number of selected actuators and performance/controllability loss?
- Does a proper sparse schedule exist with at most a constant number of active actuators at each time?
- What is the time complexity of choosing the subset of actuators with guaranteed performance bounds?

In the rest of this paper, we show some recent advances in Theoretical Computer Science can be utilized to answer these questions.

A. A Weighted Sparse Actuator Schedule

As a starting point, we allow for scaling of the input signals at chosen inputs while keeping the input scaling bounded. The input scaling allows for an extra degree freedom that could allow for choosing a sparser set of inputs. We obtain the following problem formulation. Given (1), we define a weighted actuator schedule as follows

$$x(k+1) = A\, x(k) + \sum_{i\in[m]} s_i(k)\, b_i\, u_i(k), \;\; k \in \mathbb{Z}_+ \tag{8}$$

where $s_i(k) \geq 0$ shows the strength of the i-th control input at time k. The controllability Gramian (5) at time t for this system can be rewritten as

$$\mathcal{W}_s(t) = \sum_{i=0}^{t-1} \sum_{j\in[m]} s_j^2(i)\left(A^{t-i-1}b_j\right)\left(A^{t-i-1}b_j\right)^\top. \tag{9}$$

Our goal is to reduce number of active actuators *on average*, i.e., we want to choose d actuators where d is defined as

$$d := \frac{\mathrm{card}\left\{(i,k) : i\in[m], k+1 \in [t], s_i(k) > 0\right\}}{t}, \tag{10}$$

so that the controllability Gramian of the fully actuated and the new sparsely actuated system are "close". Of course, this

Algorithm 2: A deterministic greedy-based algorithm to construct a sparse weighted actuator schedule (Theorem 1).

Input : $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, t and d
Output: $s_i(k) \geq 0$ for $(i, k+1) \in [m] \times [t]$

$\mathcal{C}(t) := \begin{bmatrix} B & AB & A^2B & \cdots & A^{t-1}B \end{bmatrix}$
Set $V = \left(\mathcal{C}(t)\mathcal{C}^\top(t)\right)^{-\frac{1}{2}}\mathcal{C}(t)$
Set $U = V$
Run $[c_1, \cdots, c_{mt}] = \mathrm{DualSet}(V, U, dt)$
return $s_i(k) := \sqrt{c_{i+mk}/(1+\frac{n}{dt})}$ for $(i, k+1) \in [m] \times [t]$

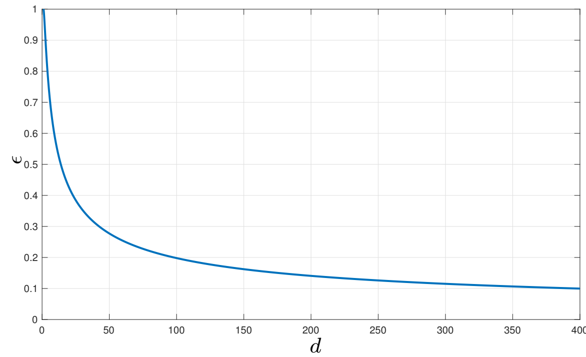

Fig. 1: This plot presents the approximation factor ϵ given by Theorem 1 versus the average number of active actuators $d \in (1, 400]$ when $t = n$.

approximation will require horizon lengths that are potentially longer than the dimension of the state. The definition below formalizes this approximation.

Definition 1: Given a time horizon $t \geq n$, a weighted actuator schedule (8) is (ϵ, d)-approximation of network (1) if

$$(1-\epsilon)\,\mathcal{W}(t) \;\preceq\; \mathcal{W}_s(t) \;\preceq\; (1+\epsilon)\,\mathcal{W}(t), \qquad (11)$$

where $\mathcal{W}(t)$ and $\mathcal{W}_s(t)$ are the controllability Gramian matrices of (1) and (8), respectively, and parameter d is defined by (10) as the average number of active actuators, and $\epsilon \in (0, 1)$ is the approximation factor.

Remark 1: While it might appear that allowing for the choice of $s_i(k)$ might lead to amplification of input signals, we note that the scaling cannot be too large because the approximation is two-sided. Specifically, by taking the trace from both sides of (11), we can see that the weighted summation of $s_i^2(k)$'s is bounded. Moreover, based on Definition 1, the ranks of matrices $\mathcal{W}(t)$ and $\mathcal{W}_s(t)$ are the same. Thus, the resulting (ϵ, d)-approximation remains controllable (recall that we assume that the original network is controllable).

The next theorem uses results from the sparsification literature to prove the existence of a sparse actuator set for a given linear system. Moreover, this result constructs a solution in polynomial time.

Theorem 1: Given the time horizon $t \geq n$, model (1), and $d > 1$, Algorithm 2 deterministically constructs an actuator schedule (8) which is a (ϵ, d)-approximation of (1) with $\epsilon = \frac{2}{\sqrt{\frac{dt}{n}}+\sqrt{\frac{n}{dt}}}$ in at most $\mathcal{O}\left(dm(tn)^2\right)$ operations.

Fig. 1 depicts the approximation ratio ϵ given by Theorem 1 versus the average number of active actuators d where the time horizon length is the same as the dimension of the state, i.e., $t = n$. As expected the approximation ratio ϵ decreases as the average number of active actuators d increases.

Tradeoffs: Theorem 1 illustrates a tradeoff between the average number of active actuators d and the time horizon t (also known as the time-to-control). This implies that the reduction in the average number of active actuators comes at the expense of increasing time horizon t in order to get the same approximation factor ϵ. Moreover, the approximation becomes more accurate as t and d are increased. Of course, increasing d will require more active actuators and larger t requires a larger control time window.

Remark 2: For a given $d \geq 1$, choosing dn columns of the controllability matrix that form a full row rank matrix (i.e., the system is controllable) is an easy task but finding dn columns of the controllability matrix that approximate the full Gramian matrix is what we are interested in the paper. To do so, we should note that approximating the full Gramian matrix while keeping the number of active actuators less than a constant d at each time is not possible in general. For example, in the case that $A = \mathbf{0}_{n\times n}$ and $B = I_n$, at least all actuators at time $k = 0$ are needed to form a full row rank matrix (or to approximate the full Gramian matrix). However, as we mentioned earlier the number of active actuators on average can be kept constant in order to approximate the full Gramian matrix.

B. Sparse Actuator Schedules with Energy Constraints

In this subsection, based on the energy/budget constraints on the scalings $s_i(.)$'s, three cases are considered as follows

(i) the scaling ratios are bounded, i.e.,

$$\max_{i\in[m],k+1\in[t]} s_i^2(k) \;\leq\; \gamma,$$

(ii) the sum of scaling ratios for each input is bounded, i.e.,

$$\max_{i\in[m]} \sum_{k+1\in[t]} s_i^2(k) \;\leq\; \gamma,$$

(iii) the sum of scaling ratios at each time is bounded, i.e.,

$$\max_{k+1\in[t]} \sum_{i\in[m]} s_i^2(k) \;\leq\; \gamma,$$

where γ is a given positive real number. In the next theorems, we present deterministic sparse actuator schedules with the above budget/energy constraints.

Theorem 2: Given the time horizon $t \geq n$, model (1), and $d > 1$, Algorithm 3 deterministically constructs an actuator schedule (8) in at most $\mathcal{O}\left(dm(tn)^2\right)$ operations such that it has on average at most d active actuators, and the following bound

$$\rho(\mathcal{W}_s(t)) \leq \left(1-\sqrt{\frac{n}{dt}}\right)^{-2} \rho(\mathcal{W}(t))$$

holds for all controllability measures. Moreover, the maxi-

Algorithm 3: A deterministic greedy-based algorithm to construct a sparse weighted actuator schedule (Theorem 2).

Input : $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, t and d
Output: $s_i(k) \geq 0$ for $(i, k+1) \in [m] \times [t]$

$\mathcal{C}(t) := \begin{bmatrix} B & AB & A^2B & \cdots & A^{t-1}B \end{bmatrix}$

Set $V = \left(\mathcal{C}(t)\mathcal{C}^\top(t)\right)^{-\frac{1}{2}}\mathcal{C}(t)$

Set $U = \begin{bmatrix} \underbrace{\mathrm{e}_1, \ldots, \mathrm{e}_{mt}}_{=I_{mt}} \end{bmatrix}$ // where $\mathrm{e}_i \in \mathbb{R}^{mt}$ for $i \in [mt]$ are the standard basis vectors for $\mathbb{R}^{mt}$

Run $[c_1, \cdots, c_{mt}] = \mathrm{DualSet}(V, U, dt)$

return $s_i(k) := \sqrt{c_{i+mk}}$ for $(i, k+1) \in [m] \times [t]$

Algorithm 4: A deterministic greedy-based algorithm to construct a sparse weighted actuator schedule (Theorem 3).

Input : $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, t and d
Output: $s_i(k) \geq 0$ for $(i, k+1) \in [m] \times [t]$

$\mathcal{C}(t) := \begin{bmatrix} B & AB & A^2B & \cdots & A^{t-1}B \end{bmatrix}$

Set $V = \left(\mathcal{C}(t)\mathcal{C}^\top(t)\right)^{-\frac{1}{2}}\mathcal{C}(t)$

Set $U = \frac{1}{m}\begin{bmatrix} \underbrace{\mathrm{e}_1, \ldots, \mathrm{e}_t}_{=I_t}, \cdots, \underbrace{\mathrm{e}_1, \ldots, \mathrm{e}_t}_{=I_t} \end{bmatrix}$
// where $\mathrm{e}_i \in \mathbb{R}^t$ for $i \in [t]$ are the standard basis vectors for $\mathbb{R}^t$ and $UU^\top = I_t$

Run $[c_1, \cdots, c_{mt}] = \mathrm{DualSet}(V, U, dt)$

return $s_i(k) := \sqrt{c_{i+mk}}$ for $(i, k+1) \in [m] \times [t]$

mum scaling ratio over all time and inputs is bounded by

$$\max_{i\in[m], k+1\in[t]} s_i^2(k) \;\leq\; \gamma,$$

where $\gamma = \left(1 + \sqrt{\frac{m}{d}}\right)^2$.

According to this result, the scaling becomes smaller as the ratio m/d decreases as expected.

Theorem 3: Given the time horizon $t \geq n$, model (1), and $d > 1$, Algorithm 4 deterministically constructs an actuator schedule (8) in $\mathcal{O}\left(dm(tn)^2\right)$ operations such that it has on average at most d active actuators, and the following

$$\rho(\mathcal{W}_s(t)) \leq \left(1 - \sqrt{\frac{n}{dt}}\right)^{-2} \rho(\mathcal{W}(t))$$

holds for all controllability measures. Moreover, the sum of scaling ratios for all inputs is bounded by

$$\max_{i\in[m]} \sum_{k=0}^{t-1} s_i^2(k) \;\leq\; \gamma,$$

where $\gamma = m\left(1 + \sqrt{\frac{1}{d}}\right)^2$.

Theorem 4: Given the time horizon $t \geq n$, model (1), and $d > 1$, Algorithm 5 deterministically constructs an actuator schedule (8) in $\mathcal{O}\left(dm(tn)^2\right)$ operations such that it has on

Algorithm 5: A deterministic greedy-based algorithm to construct a sparse weighted actuator schedule (Theorem 4).

Input : $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, t and d
Output: $s_i(k) \geq 0$ for $(i, k+1) \in [m] \times [t]$

$\mathcal{C}(t) := \begin{bmatrix} B & AB & A^2B & \cdots & A^{t-1}B \end{bmatrix}$

Set $V = \left(\mathcal{C}(t)\mathcal{C}^\top(t)\right)^{-\frac{1}{2}}\mathcal{C}(t)$

Set $U = \frac{1}{t}\begin{bmatrix} \underbrace{\mathrm{e}_1, \ldots, \mathrm{e}_m}_{=I_m}, \cdots, \underbrace{\mathrm{e}_1, \ldots, \mathrm{e}_m}_{=I_m} \end{bmatrix}$
// where $\mathrm{e}_i \in \mathbb{R}^m$ for $i \in [m]$ are the standard basis vectors for $\mathbb{R}^m$ and $UU^\top = I_m$

Run $[c_1, \cdots, c_{mt}] = \mathrm{DualSet}(V, U, dt)$

return $s_i(k) := \sqrt{c_{i+mk}}$ for $(i, k+1) \in [m] \times [t]$

average at most d active actuators, and the following

$$\rho(\mathcal{W}_s(t)) \leq \left(1 - \sqrt{\frac{n}{dt}}\right)^{-2} \rho(\mathcal{W}(t))$$

holds for all controllability measures. Moreover, the sum of scaling ratios at each time is bounded by

$$\max_{k+1\in[t]} \sum_{i=1}^{m} s_i^2(k) \;\leq\; \gamma,$$

where $\gamma = t\left(1 + \sqrt{\frac{m}{dt}}\right)^2$.

C. An Unweighted Sparse Actuator Schedule

In the previous subsections, we allowed for re-scaling of the input to come up with a sparse approximation of the Gramian. Here, we assume that the actuator/signal strength cannot be arbitrarily set for individual active actuators and only can be 0 or 1. Given a time horizon $t \geq n$, our problem is to compute an actuator schedule (8) where $s_i(k) \in \{0, 1\}$ for all $(i, k+1) \in [m] \times [t]$. As before, the controllability Gramian $\mathcal{W}_s(t)$ at time t for this schedule is given by (9). Optimal actuator selection can now be formulated as a combinatorial optimization problem, and the optimal dynamic strategy is given as:

$$\begin{aligned} &\underset{s_i(k)}{\text{Minimize}} \;\; \rho\left(\mathcal{W}_s(t)\right) \\ &\text{subject to:} \\ &\quad s_i(k) \in \{0,1\} \text{ for all } (i, k+1) \in [m] \times [t], \\ &\quad \sum_{i=1}^{m}\sum_{k=0}^{t-1} s_i(k) \;\leq\; dt, \end{aligned} \tag{12}$$

where d is the average number of active actuators at each time, t is a time horizon, and m is the total number of actuators.

The exact combinatorial optimization problem (12) is intractable and NP-hard; however, it is straightforward to solve a continuous relaxation of this optimization problem for the case of convex controllability metrics. To find a near-optimal solution of optimization problem (12), one can use a variety of standard methods for optimal experimental design (greedy methods, sampling methods, the classical pipage rounding method combined with SDP). In [25], we use a regret minimization of the least eigenvalues of positive

Algorithm 6: A deterministic greedy-based algorithm to construct a sparse unweighted actuator schedule (Theorem 5).

Input : $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, t and d

Output: $s_i(k)$ for $(i, k+1) \in [m] \times [t]$

$\mathcal{C}(t) := \begin{bmatrix} B \ AB \ A^2B \ \cdots \ A^{t-1}B \end{bmatrix}$

Set $V = \left(\mathcal{C}(t)\mathcal{C}^\top(t)\right)^{-\frac{1}{2}} \mathcal{C}(t)$

Set $U = \begin{bmatrix} \underbrace{\mathrm{e}_1, \ldots, \mathrm{e}_{mt}}_{=I_{mt}} \end{bmatrix}$

// where $\mathrm{e}_i \in \mathbb{R}^{mt}$ for $i \in [mt]$ are the standard basis vectors for $\mathbb{R}^{mt}$

Run $[c_1, \cdots, c_{mt}] = \mathrm{DualSet}(V, U, dt)$

return $s_i(k) := \lceil \sqrt{c_{i+mk}} \rceil$ for $(i, k+1) \in [m] \times [t]$

semi-definite matrices based on [23] to obtain a constant approximation ratio of (12) for all *systemic* controllability metrics.

In the following result, we use our results from Subsection III-A to obtain an unweighted sparse actuator schedule with guaranteed performance bound.

Theorem 5: Assume that time horizon $t \geq n$, dynamics (1), and $d > 1$ are given. Then polynomial-time Algorithm 6 deterministically constructs an actuator schedule (8) with $s_i(k) \in \{0, 1\}$ such that it has on average at most d active actuators, and the following

$$\rho(\mathcal{W}_s(t)) \ \leq \ \left(\frac{1 + \sqrt{\frac{m}{d}}}{1 - \sqrt{\frac{n}{dt}}}\right)^2 \rho(W(t)),$$

holds for all controllability measures.

In view of this result, one can choose any constant number greater than one as the number of active actuators on average to construct a sparse unweighted actuator schedule in order to approximate controllability measures. This, however comes at the cost of an extra $\left(1 + \sqrt{\frac{m}{d}}\right)^2$ factor in terms of the energy cost compared to the weighted sparse actuator schedule (cf. Theorem 2).

IV. Conclusions

In this paper, we showed how recent advances in matrix reconstruction and sparsification literature can be utilized to develop subset selection tools for choosing a relatively small subset of actuators to approximate certain controllability measures. Current approaches based on polynomial time relaxations of the subset selection problem require an extra multiplicative factor of $\log n$ sensors/actuators times the minimal number in order to just maintain controllability/observability. Furthermore, when the control energy is chosen as the cost, submodularity-based approaches fail to guarantee the performance using greedy methods. In contrast, we show that there exists a polynomial-time actuator schedule that on average selects only a constant number of actuators at each time, to approximate controllability measures. A potential future direction is to see whether this approach can be used to develop an efficient scheme for minimal reachability problems.

References

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

[2] F. Pasqualetti, S. Zampieri, and F. Bullo, "Controllability metrics, limitations and algorithms for complex networks," *IEEE Transactions on Control of Network Systems*, vol. 1, no. 1, pp. 40–52, 2014.

[3] Y.-Y. Liu and A.-L. Barabási, "Control principles of complex systems," *Reviews of Modern Physics*, vol. 88, no. 3, p. 035006, 2016.

[4] P. V. Chanekar, N. Chopra, and S. Azarm, "Optimal actuator placement for linear systems with limited number of actuators," in *2017 American Control Conference (ACC)*, May 2017, pp. 334–339.

[5] 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.

[6] 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, 2016.

[7] A. Olshevsky, "Minimum input selection for structural controllability," in *2015 American Control Conference (ACC)*, July 2015, pp. 2218–2223.

[8] S. Pequito, G. Ramos, S. Kar, A. P. Aguiar, and J. Ramos, "The robust minimal controllability problem," *Automatica*, vol. 82, pp. 261 – 268, 2017.

[9] E. Nozari, F. Pasqualetti, and J. Cortés, "Time-invariant versus time-varying actuator scheduling in complex networks," in *2017 American Control Conference (ACC)*, May 2017, pp. 4995–5000.

[10] A. Yazıcıoğlu, W. Abbas, and M. Egerstedt, "Graph distances and controllability of networks," *IEEE Transactions on Automatic Control*, vol. 61, no. 12, pp. 4125–4130, 2016.

[11] 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, 2016.

[12] S. Pequito, S. Kar, and A. P. Aguiar, "On the complexity of the constrained input selection problem for structural linear systems," *Automatica*, vol. 62, pp. 193–199, 2015.

[13] A. Chakrabortty and M. D. Ilić, *Control and optimization methods for electric smart grids*. Springer, 2011, vol. 3.

[14] F. A. Chandra, G. Buzi, and J. Doyle, "Glycolytic oscillations and limits on robust efficiency," *Science*, vol. 333, no. 6039, pp. 187–192, 2011.

[15] L. Marucci, D. A. Barton, I. Cantone, M. A. Ricci, M. P. Cosma, S. Santini, D. di Bernardo, and M. di Bernardo, "How to turn a genetic circuit into a synthetic tunable oscillator, or a bistable switch," *PLoS One*, vol. 4, no. 12, p. e8083, 2009.

[16] I. Rajapakse, M. Groudine, and M. Mesbahi, "What can systems theory of networks offer to biology?" *PLoS computational biology*, vol. 8, no. 6, p. e1002543, 2012.

[17] M. Siami and J. Skaf, "Structural analysis and optimal design of distributed system throttlers," *IEEE Transactions on Automatic Control*, vol. 63, no. 12, pp. 540–547, Feb. 2018.

[18] R. E. Kalman, "Mathematical description of linear dynamical systems," *Journal of the Society for Industrial and Applied Mathematics, Series A: Control*, vol. 1, no. 2, pp. 152–192, 1963.

[19] V. Tzoumas, M. A. Rahimian, G. J. Pappas, and A. Jadbabaie, "Minimal actuator placement with bounds on control effort," *IEEE Transactions Control of Network Systems*, vol. 3, no. 1, pp. 67–78, 2016.

[20] V. Tzoumas, A. Jadbabaie, and G. J. Pappas, "Minimal reachability problems," in *Proceedings of the 54th IEEE Conference on Decision and Control*, 2015, pp. 4220–4225.

[21] V. Tzoumas, K. Gatsis, A. Jadbabaie, and G. J. Pappas, "Resilient monotone submodular function maximization," *arXiv preprint arXiv:1703.07280*, 2017.

[22] O. Kempthorne, *The design and analysis of experiments.* Wiley, 1952.

[23] Z. Allen-Zhu, Y. Li, A. Singh, and Y. Wang, "Near-optimal design of experiments via regret minimization," in *Proceedings of the 34th International Conference on Machine Learning*, vol. 70. PMLR, 06–11 Aug 2017, pp. 126–135.

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

[25] A. Jadbabaie, A. Olshevsky, and M. Siami, "Limitations and tradeoffs in minimum input selection problems," in 2018 American Control Conference, accepted.

[26] C. Boutsidis, P. Drineas, and M. Magdon-Ismail, "Near-optimal column-based matrix reconstruction," *SIAM Journal on Computing*, vol. 43, no. 2, pp. 687–717, 2014.

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

[28] ——, "Growing linear dynamical networks endowed by spectral systemic performance measures," *IEEE Transactions on Automatic Control*, vol. 63, no. 8, 2018.

[29] ——, "Systemic measures for performance and robustness of large–scale interconnected dynamical networks," in *Proceedings of the 53rd IEEE Conference on Decision and Control*, Dec. 2014.

[30] J. Batson, D. A. Spielman, and N. Srivastava, "Twice-ramanujan sparsifiers," *SIAM Journal on Computing*, vol. 41, no. 6, pp. 1704–1721, 2012.