Control Oriented Learning in the Era of Big Data

Mario Sznaier

Abstract—**Recent advances in control, coupled with an exponential growth in data gathering capabilities, have made feasible a wide range of applications that can profoundly impact society. Yet, achieving this vision requires addressing the challenge of extracting control relevant information from large amounts of data, a problem that has proven to be surprisingly difficult. While modern machine learning techniques can handle very large data sets, most control oriented learning algorithms struggle with a few thousand points. The goal of this paper is to point out the reason why dynamic data is challenging and to indicate strategies to overcome this challenge. The main message is twofold (i) computational complexity in control oriented learning is driven both by system order and the presence of uncertainty, rather than the dimension of the data, and (ii) exploiting the underlying sparsity provides a way around the "curse of dimensionality".**

Index Terms—**Identification for Control, Machine Learning, Robust Control, Switched Systems, Uncertain Systems**

I. Introduction

Recent advances in sensing and data collection capabilities provide access to exponentially increasing amounts of data. This data availability opens up a wide range of applications –from safer, self-aware environments and smart cities to autonomous vehicles– that have the potential to profoundly impact society. However, in order to realize this potential, control algorithms will need to extract control relevant information from large amounts of data, often in real time, using computational and power budgets compatible with on-board resources. Unfortunately, in most scenarios, this is beyond the capacity of "traditional" algorithms. For instance, even in well established areas such as Linear Time Invariant (LTI) systems identification, recent results indicate that the sample complexity of finding a state space realization of an n^{th} order system scales as $\mathcal{O}(n(\text{error in realization})^{-2})$, when the states are directly measurable, and as $\mathcal{O}(n^5(\text{error in realization})^{-4})$ for the general case. When taking into account the cost of the identification algorithm, the overall computational complexity grows as $n^3(\text{error in realization})^{-2}$ for state measurements and $n^8(\text{error in realization})^{-4}$ for the general case.

The goal of this paper is to point out the specific features that make the problem of control oriented information extraction harder than other "big data" type problems and point out to alternatives to mitigate the "curse of dimensionality". In particular, data generated by dynamical systems is temporally correlated through the underlying dynamics. On one hand, this correlation renders the learning problem harder, since it limits the use of statistical independence arguments commonly used in machine learning. In fact, the need to account for this correlation is what drives the sample complexity of identification. On the other hand, this same correlation induces an underlying sparse structure that can be exploited to mitigate computational complexity. These observations motivate the main messages of the paper (i) computational complexity in control oriented learning is driven both by system order and the presence of uncertainty, rather than the dimension of the data itself, and (ii) exploiting the underlying sparsity provides a way to mitigate the "curse of dimensionality".

The paper starts by revisiting LTI systems identification, examining the sample and computational complexity of existing methods and potential mitigation strategies that exploit connections to learning in Reproducing Kernel Hilbert Spaces (RKHS). In particular, we show that the problem of identifying parsimonious LTI models can be recast as a regularized atomic norm minimization. In turn, this minimization can be efficiently solved by using a randomized version of the Frank-Wolfe algorithm [1], whose complexity scales linearly with the number of data points. Further, these results can be easily extended to dynamical graphical models, representing dynamic interactions between multiple agents. We conclude the analysis of LTI systems by briefly examining data-driven control methods, where a controller is designed directly from the data, without identifying the plant, by learning a control Lyapunov function [2], [3]. As before, computational complexity is driven both by the order of the system and the presence of uncertainty, with a substantial increase in complexity when only noisy data is available.

We then move to the nonlinear case. Non-linear identification and its connections to machine learning techniques such as manifold embedding and deep learning are currently very active research topics. We argue that, as in the in the LTI case, computational complexity is still driven by the memory of the system and offer some thoughts on how to mitigate this complexity. We start by considering switched linear systems, since as universal approximators [4] they provide tractable approximations to general nonlinear control problems. While identification of switched linear systems is generically NP-hard, we show that it is possible to obtain tractable convex relaxations that scale linearly with the number of data points by exploiting a connection to semi-algebraic optimization. We conclude the paper by briefly examining Koopman operator based methods for identifying generic non-linear dynamics.

The paper is not intended to serve as a comprehensive survey of control oriented learning, a daunting task given the large volume of research carried out in the past decade. In particular, we do not cover reinforcement learning (RL) based methods (see for instance [5], [6] for results related to the performance of model based versus model free RL approaches in LQR/LQG problems). Rather, our goal is to examine a suite of approaches, both for well established areas such as LTI identification and for ones still being developed (e.g. nonlin-

This work was supported in part by NSF grants CNS–1646121, CMMI–1638234, IIS–1814631, ECCS–1808381 and CNS–2038493, and AFOSR grant FA9550-19-1-0005. The author is with the ECE Department, Northeastern University, Boston, MA 02115. email: msznaier@coe.neu.edu

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCSYS.2020.3045664, IEEE Control Systems Letters

ear identification), that highlight the fact that computational complexity is related to system order and uncertainty, rather than merely the number of data points or their dimension.

II. Identification of LTI systems

In this section we revisit some classical results on LTI identification from the perspective of computational complexity, and use recent results on the sample complexity of identification to link this computational complexity to the identification error.

A. Finding realizations via Least Squares

The simplest "control oriented learning" problem is the classical LTI realization problem: given the first $n_M + 1$ Markov parameters of an LTI system, $\mathbf{G}_k \in \mathbb{R}^{p\times m},\ k = 0,\dots,n_M$, find a minimal state space model:

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{w}_k \\ \mathbf{y}_k &= \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k + \mathbf{v}_k \end{aligned} \tag{1}$$

such that $\mathbf{G}_k = \mathbf{C}\mathbf{A}^{k-1}\mathbf{B},\ k = 1,\dots,n_M$. This problem can be solved by using a variant of Ho's algorithm based on a factorization of the Hankel matrix with (block) entries $\mathbf{H}_{i,j} = \mathbf{G}_{i+j-1}$ [7]. Specifically, consider a Hankel matrix $\mathbf{H}$ with n_1 (block) rows and n_2+1 (block) columns, with $n_1+n_2 = n_M$. Assume that $n \doteq \text{rank}(\mathbf{H}) < \min\{n_1, n_2+1\}$ and define the matrices:

$$\begin{aligned} \overleftarrow{\mathbf{H}} &\doteq \text{last } mn_2 \text{ columns of } \mathbf{H} \\ \overrightarrow{\mathbf{H}} &\doteq \text{first } mn_2 \text{ columns of } \mathbf{H} \\ \mathbf{U}, \mathbf{\Sigma_n}, \mathbf{V} &\doteq \text{reduced svd of } \overrightarrow{\mathbf{H}}, \text{ e.g. } \overrightarrow{\mathbf{H}} = \mathbf{U}\mathbf{\Sigma_n}\mathbf{V^T} \\ \mathbf{K}_o &\doteq \mathbf{U}\mathbf{\Sigma}_n^{\frac{1}{2}},\ \mathbf{K}_c \doteq \mathbf{\Sigma}_n^{\frac{1}{2}}\mathbf{V}^T, \end{aligned}$$

Then, a minimal realization of (1) is given by:

$$\begin{aligned} &\mathbf{B} = \text{first } m \text{ columns of } \mathbf{K}_c,\ \mathbf{C} = \text{first } p \text{ rows of } \mathbf{K}_o, \\ &\mathbf{A} = \mathbf{K}_o^\dagger \overleftarrow{\mathbf{H}} \mathbf{K}_c^\dagger,\ \mathbf{D} = \mathbf{G}_0 \end{aligned} \tag{2}$$

where $\mathbf{K}_o^\dagger \doteq (\mathbf{K}_o^T\mathbf{K}_o)^{-1}\mathbf{K}_o^T$ and $\mathbf{K}_c^\dagger \doteq \mathbf{K}_c^T(\mathbf{K}_c\mathbf{K}_c^T)^{-1}$. The computational complexity of the algorithm is dominated by the cost of computing the svd of $\overrightarrow{\mathbf{H}}$ Assuming that the order of the system is n (either given as a prior or estimated from building Hankel matrices with more than n columns) and that $p \geq m$, (more sensors than controls), complexity can be minimized by choosing a "very rectangular" $\mathbf{H}$, with $n_1 \sim n_M - n$ and $n_2 \sim n$, resulting in a computational complexity $\mathcal{O}((n_M - n)^2 p^2 mn + n^3 m^3) \propto n^3$ when $n_M = 2n$.

Note that the bound above is asymptotic, for a fixed number of points, and assuming that the Markov parameters $\mathbf{G}_k$ are known. An interesting question is the *sample complexity* of learning these parameters from experimental data, that is the number of points needed to guarantee, with high probability, a given error bound. As shown in [8], the first n_M Markov parameters can be learned, with a computational cost $\mathcal{O}(Nn_M^3 m^2)$, by performing N experiments, each of length n_M, and solving a least squares problem of the form:

$$\mathbf{G} = \underset{\mathbf{X}\in R^{p\times mn_M}}{argmin} \|\mathbf{Y} - \mathbf{X}\mathbf{U}\|_F^2 = \mathbf{Y}\mathbf{U}^T(\mathbf{U}\mathbf{U}^T)^{-1} \tag{3}$$

where

$$\begin{aligned} &\mathbf{G} \doteq \left[\mathbf{G}_1 \dots \mathbf{G}_{n_M}\right] \\ &\mathbf{Y} \doteq \left[\mathbf{y}^{(1)} \dots \mathbf{y}^{(N)}\right],\ \mathbf{U} \doteq \left[\mathbf{T}_u^{(1)} \dots \mathbf{T}_u^{(N)}\right] \end{aligned}$$

Here $\mathbf{T}_u^{(i)}$ denotes the Toeplitz matrix formed from the input used in the i^{th} experiment and $\mathbf{y}^{(i)} \doteq \left[\mathbf{y}_1^{(i)} \dots \mathbf{y}_{n_M}^{(i)}\right]$ the corresponding output.

In terms of sampling complexity, given $0 < \delta < 1$, assume that $N \geq 8mn_M + \mathcal{O}(log(n_M\delta^{-1}))$. Then, with probability greater than $1 - \delta$:

$$\begin{aligned} \|\mathbf{G} - \mathbf{G}_{true}\| \leq N^{-\frac{1}{2}} n_M (K_1\sigma_v + K_2\sigma_w n_M^{\frac{1}{2}}) log^{\frac{1}{2}}(n_M\delta^{-1}) \\ + \mathcal{O}(N^{-\frac{1}{2}} n_M log^{\frac{1}{2}}(n_M\delta^{-1})) \end{aligned}$$

where σ_v and σ_w denote the covariances of the measurement and process noise. Combining this bound with the results in [9], keeping only the leading terms and choosing $n_1 = n_2 + 1 = n+1$ (the smallest dimensions that allow for estimating a state space realization with n states) shows that the number of experiments N needed to estimate the system matrices within a given error bound satisfies:

$$N \propto \frac{n^5(K_1\sigma_v + K_2\sigma_w n^{\frac{1}{2}})^2 log\frac{n}{\delta}}{(\text{error in matrices})^4} \tag{4}$$

Thus, even in the absence of process noise, the computational complexity of finding a state space model with (probabilistic) error bounds by first learning the Markov parameters and then factoring the corresponding Hankel matrix is dominated by the cost of the learning phase. This complexity roughly scales as $\mathcal{O}(n^8\sigma_v^2(\text{error in matrices})^{-4})$, highlighting the role played by system order and uncertainty.

In the special case where noisy measurement of the states are directly available (e.g. $\mathbf{C} = \mathbf{I}, \mathbf{D} = 0$), the matrices $\mathbf{A}, \mathbf{B}$ can be directly estimated by solving a least squares problem, avoiding the Hankel factorization step. In this scenario, [10] has shown that the estimation error $\propto \sigma_w\sqrt{\frac{(n+m)\log\delta^{-1}}{N}}$, and thus the overall computational cost is $\mathcal{O}(N(n+m)^2 + (n+m)^3) \propto \frac{\sigma_w^2(n+m)^3\log\delta^{-1}}{(\text{error in matrices})^2}$.

An alternative approach that seeks to directly estimate the realization from a single lenght T execution, rather than from N roll-outs, has been proposed in [9]. As shown there, in the case of stable systems, and assuming that the number of Markov parameters estimated is large enough so that $\|\mathbf{C}\mathbf{A}^{n_M-1}\| \sim 0$, the first n_M Markov parameters can be computed in $\mathcal{O}(n_M^3 m^3 + Tn_M^2 m^2)$, with an approximation error $\propto \sqrt{\frac{n_M m}{T}}$. The corresponding error in the estimated realization is, with probability $\geq 1-\delta$, $\propto (n^3 n_M m)^{\frac{1}{4}} T^{-\frac{1}{4}}$. Assuming that n_M is a fixed multiple of n, it follows that, by exploiting stability, the cost of estimating a realization with a given error bound is $\propto \frac{n^6 m^3}{(\text{error in matrices})^4} + n^3 m^3$.

B. Enforcing Stability

A potential problem with the methods discussed above is that the resulting system may not be stable, even if the underlying system generating the data is. In addition, the probabilistic error bounds provided by the sample complexity analysis are not well suited to be used by robust control methods such as $\mathcal{H}_\infty$ that rely on worst case uncertainty bounds. These issues can be addressed using control oriented identification methods (see e.g. [11]), based on interpolation theory. Given $n_p = n_t + n_f$ time and frequency domain inputs $\{u_i^t\}_{i=1}^{n_t}, \{u^f(z_k)\}_{k=1}^{n_f}$, where $z_k = e^{-j\omega_k}$, let y_i^t, y_k^f denote the measurements of the corresponding outputs, corrupted by bounded ℓ_∞ noise with bounds ϵ^t, ϵ^f. It can be shown [12] that existence of a system with all poles in $|z| \leq \rho < 1$ that interpolates the experimental data points within the noise level is equivalent to feasibility of the following semi-definite program (SDP):

$$\begin{aligned}
&\min_{K \geq 0, \mathbf{h}, \boldsymbol{\xi}} K^2 \text{ subject to:} \\
&\mathbf{Z} \doteq \begin{bmatrix} \mathbf{M}_0^{-1} & \mathbf{X} \\ \mathbf{X}^* & K^2 \mathbf{M}_0 \end{bmatrix} \succeq 0 \\
&|y_k^f - \xi_k u_k^f| \leq \epsilon^f, \ k = 1, \ldots, n_f \\
&|y_i^t - (\mathbf{T}(u)\mathbf{h})_i| \leq \epsilon^t, \ i = 1, \ldots, n_t
\end{aligned} \tag{5}$$

where

$$\begin{aligned}
\mathbf{M}_0 &= \begin{bmatrix} \mathbf{P} & \mathbf{S}_0 \mathbf{R}^2 \\ \mathbf{R}^2 \mathbf{S}_0^* & \mathbf{R}^2 \end{bmatrix}, \ \mathbf{X} = \begin{bmatrix} \boldsymbol{\Xi} & 0 \\ 0 & \mathbf{T}^T(\mathbf{h}) \end{bmatrix} \\
\mathbf{S}_0 &= \left[(z_i^{1-j})^* \right]_{ij}, \ i = 1, \ldots, n_f, \quad j = 1, \ldots, n_t \\
\boldsymbol{\Xi} &= \text{diag} \left[\ \xi_1 \quad \ldots \quad \xi_{n_f} \ \right] \\
\mathbf{R} &= \text{diag} \left[\ 1 \quad \rho \quad \rho^2 \quad \ldots \rho^{n_t - 1} \ \right], \\
\mathbf{P} &= \left[\tfrac{z_i^* z_j}{z_i^* z_j - \rho^2} \right]_{ij}, \ i, j = 1, \ldots, n_f \\
\mathbf{h} &= \left[h_1 \ldots h_{n_t} \right]^T \\
\mathbf{T}(h) &= \begin{bmatrix} h_1 & \ldots & 0 \\ \vdots & \ddots & \vdots \\ h_{n_t} & \ldots & h_1 \end{bmatrix}, \ \mathbf{T}(u) = \begin{bmatrix} u_1 & \ldots & 0 \\ \vdots & \ddots & \vdots \\ u_{n_t} & \ldots & u_1 \end{bmatrix}
\end{aligned}$$

where $\mathbf{X}^*$ denotes Hermitian conjugate. As shown in [12], if (5) is feasible, then the set of all systems that have the first n_t elements of the impulse response given by $\mathbf{h}$, frequency response at z_k given by ξ_k and gain $\leq K$ is described by $\mathcal{S}_{\boldsymbol{\xi},\mathbf{h}} = \{G \colon G = \mathcal{F}_\ell[L(z), Q(z)]\}$ where $\mathcal{F}_\ell(.,.)$ denotes lower fractional transformation. Here $L(z)$ depends only on $\boldsymbol{\xi}$ and $\mathbf{h}$, and Q is any transfer function with all poles in $|z| < \rho$ and such that $max_{|z|=\rho}|Q(z)| \leq 1$. In particular, the choice $Q = 0$ (so called central interpolant) leads to the model $G_{central}(z)$ with order no larger than $n_t + n_f$ [12][1].

In this context, the worst case identification error is bounded by the diameter of the set $\mathcal{S}_{\boldsymbol{\xi},\mathbf{h}}$ [11]. For a purely time domain impulse response experiment or a frequency domain experiment with equally spaced data, this error is bounded by

$$\|G(z) - G_{true}(z)\|_{\mathcal{H}_\infty} \leq \begin{cases} 2(n_t \epsilon_t + K \frac{\rho^{n_t}}{1-\rho}) & \text{time domain} \\ 2(\epsilon_f + 2K \frac{\rho\pi}{n_f(1-\rho^2)}) & \begin{array}{l}\text{freq. domain} \\ n_f \gg 4\end{array} \end{cases}$$

While this approach is guaranteed to yield stable systems, if (5) is solved using a standard interior point solver, its complexity grows as $\mathcal{O}(n_p^6)$, and thus can only handle short data records. Even when using a first order Alternating Direction Method of Multipliers (ADMM) algorithm [14], which brings down the complexity of each iteration to $\mathcal{O}(n_p^3)$, a typical laptop is limited to $\approx 10^3$ data points.

C. Regularization based methods

Recently proposed alternatives to subspace and interpolation based approaches are motivated by the success of kernel based methods and Tikhonov regularization in machine learning (see for instance [15] for an excellent tutorial). Consider again the least squares estimator (3) and assume for simplicity that only white Gaussian measurement noise with covariance $\sigma^2 \mathbf{I}$ is present. Then, the estimate $\mathbf{G}$ has covariance $\mathcal{E}\left\{(\mathbf{G} - \mathbf{G}_{true})^T (\mathbf{G} - \mathbf{G}_{true})\right\} = \sigma^2 (\mathbf{U}\mathbf{U}^T)^{-1}$. It can be shown that the estimator (3) is asymptotically efficient, that is, as $N \to \infty$ its covariance approaches the Cramer-Rao limit, so no other unbiased estimator can outperform it [15]. On the other hand, in many scenarios it is advantageous to trade off bias versus variance by incorporating additional degrees of freedom. Consider a Tikhonov type regularized regression

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{argmin} \|\mathbf{y} - \boldsymbol{\theta}\boldsymbol{\Phi}\|_2^2 + \gamma \boldsymbol{\theta} \mathbf{P}^{-1} \boldsymbol{\theta}^T \tag{6}$$

where $\boldsymbol{\Phi}$ is known, $\boldsymbol{\theta}$ is a vector of parameters to be estimated and where $\mathbf{P} \succeq 0$[2] and $\gamma \geq 0$ are the additional degrees of freedom. In the SISO case, problem (3) directly fits this formalism by setting $\boldsymbol{\theta} = \mathbf{G}$ and $\boldsymbol{\Phi} = \mathbf{U}$, while the MIMO case can be accommodated by vectorizing $\mathbf{G}$ and $\mathbf{Y}$ and rearranging the elements of $\mathbf{U}$. It can be shown [15] that in this case, the minimum variance estimator of $\boldsymbol{\theta}_{true}$ is obtained by setting $\gamma = \sigma^2$ and $\mathbf{P} = \boldsymbol{\theta}_{true}^T \boldsymbol{\theta}_{true}$, yielding

$$\begin{aligned}
&\hat{\boldsymbol{\theta}} = \mathbf{y}\boldsymbol{\Phi}^T \mathbf{P} (\boldsymbol{\Phi}\boldsymbol{\Phi}^T \mathbf{P} + \sigma^2 \mathbf{I})^{-1} \\
&\Pi_\theta \doteq \mathcal{E}\{(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_{true})^T (\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_{true})\} = \sigma^2 (\boldsymbol{\Phi}\boldsymbol{\Phi}^T + \sigma^2 \mathbf{P}^{-1})^{-1}
\end{aligned} \tag{7}$$

While this result points out to the advantages of using regularized regression, it is mainly of theoretical importance, since the regularization term depends on the unknown $\boldsymbol{\theta}_{true}$. However, it provides a bridge to the case where the unknown parameters are random variables, with known covariance. Assume that the parameter $\boldsymbol{\theta}$ is a Gaussian random vector with distribution $\mathcal{N}(\boldsymbol{\theta}_o, \mathbf{P})$ and that the noise is white Gaussian, with covariance σ^2. The vector $\mathbf{z} = \begin{bmatrix} \boldsymbol{\theta} - \boldsymbol{\theta}_o & \mathbf{y} - \boldsymbol{\theta}_o \boldsymbol{\Phi} \end{bmatrix}$ is Gaussian with zero mean and covariance $\boldsymbol{\Sigma} \doteq \begin{bmatrix} \mathbf{P} & \mathbf{P}\boldsymbol{\Phi} \\ \boldsymbol{\Phi}^T \mathbf{P} & \boldsymbol{\Phi}^T \mathbf{P} \boldsymbol{\Phi} + \sigma^2 \mathbf{I} \end{bmatrix}$. Next, recall that if two variables $\mathbf{x}_1, \mathbf{x}_2$ have a joint Gaussian distribution with mean $\begin{bmatrix} \boldsymbol{\mu}_1 & \boldsymbol{\mu}_2 \end{bmatrix}$ and covariance $\boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{12}^T & \boldsymbol{\Sigma}_{22} \end{bmatrix}$, then $\mathbf{x}_1 | \mathbf{x}_2 \sim \mathcal{N}(\mu, \boldsymbol{\Pi})$ with:

$$\begin{aligned}
\boldsymbol{\mu} &= \boldsymbol{\mu}_1 + (\mathbf{x}_2 - \boldsymbol{\mu}_2) \boldsymbol{\Sigma}_{22}^{-1} \boldsymbol{\Sigma}_{12}^T \\
\boldsymbol{\Pi} &= \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12} \boldsymbol{\Sigma}_{22}^{-1} \boldsymbol{\Sigma}_{12}^T
\end{aligned} \tag{8}$$

Using this formula to compute the posterior distribution of $(\boldsymbol{\theta} - \boldsymbol{\theta}_o)|\mathbf{y}$ yields: $(\boldsymbol{\theta} - \boldsymbol{\theta}_o)|\mathbf{y} \sim \mathcal{N}(\hat{\boldsymbol{\theta}}, \boldsymbol{\Pi}_{\boldsymbol{\theta}})$, where $\hat{\boldsymbol{\theta}}$ and $\boldsymbol{\Pi}_{\boldsymbol{\theta}}$ are given in (7), with $\mathbf{y}$ replaced by $\mathbf{y} - \boldsymbol{\theta}_o \boldsymbol{\Phi}$. Thus, in a Bayesian framework, $\hat{\boldsymbol{\theta}}$ can be considered as the maximum a-posteriori estimate (MAP) of $\boldsymbol{\theta}$ given its a-priori mean $\boldsymbol{\theta}_o$ and covariance $\mathbf{P}$, and the observations $\mathbf{y}$.

An interesting open question is whether the use of regularization can improve the sample complexity of learning via least

[1] The minimum order interpolant can be obtained by using the degrees of freedom available in Q to minimize the rank of a Loewner matrix [13].

[2] For a singular $\mathbf{P} \doteq \mathbf{V} \begin{bmatrix} \boldsymbol{\Sigma} & 0 \\ \mathbf{0} & 0 \end{bmatrix} \mathbf{V}^T$, $\mathbf{P}^{-1} \doteq \mathbf{V} \begin{bmatrix} \boldsymbol{\Sigma}^{-1} & 0 \\ \mathbf{0} & 0 \end{bmatrix} \mathbf{V}^T$.

squares. An affirmative answer has been given in [16] where the regularization penalty is given in terms of the nuclear norm of the Hankel matrix, leading to a problem of the form

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{argmin}\, \frac{1}{2}\|\mathbf{y} - \boldsymbol{\theta}\boldsymbol{\Phi}\|_2^2 + \lambda\|\mathbf{H}(\boldsymbol{\theta})\|_* \tag{9}$$

where $\mathbf{y} \in R^{N_{\text{reg}}}$ contains data collected from N_{reg} roll-outs. In this context, if $N_{\text{reg}} \geq \min\{n^2, n_M\}$ then $\|\mathbf{H}(\boldsymbol{\theta} - \boldsymbol{\theta})_{\text{true}}\|_* \propto \sqrt{\frac{n_M}{N_{\text{reg}}}}\log n_M$ and thus the number of roll-outs $N_{\text{reg}} \propto \frac{n^3\log^2 n}{(\text{error in the realization})^4}$ compared against $\mathbf{N} \propto \frac{n^5 \log n\delta^{-1}}{(\text{error in the realization})^4}$ for regular least squares. On the other hand, while (3) has an explicit solution, (9) entails solving a semi-definite program. This SDP can be efficiently solved using the ADMM based method proposed in [14], at a cost of $\propto \frac{n^5 log^2 n}{(\text{error in matrices})^4}$ to set up the problem, plus $\mathcal{O}(n^3)$ per iteration compared against an overall cost $\propto n^8$ (n^6 for stable systems) when using least squares. Note that this computational complexity reduction is partially due to the fact that the regularized approach uses just one data point from each roll-out, versus all data points for the approaches discussed in Section II-A.

D. Connections with Reproducing Kernel Hilbert Spaces

The optimization (6) can also be viewed in the context of learning functions in a RKHS. Briefly, given a set $\mathcal{X}$, a symmetric function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a Mercer kernel if it is positive definite, that is, for all finite subsets $\{x_{i_1}, \ldots, x_{i_n}\} \subset \mathcal{X}$ the matrix $\mathbf{K}$ with entries $K(x_i, x_j)$ is positive definite [17]. Each Mercer kernel defines a unique Hilbert space $\mathcal{H}_K$ of functions of the form $f(.) = \sum_{i=1}^{s} K(x_i, .)f_i$, for some scalar s and $x_i \in \mathcal{X}$, equipped with the inner product $\langle f, g\rangle \doteq \sum_{i,j} f_i g_j K(x_i, x_j)$ [17].

In this context, given data, one can attempt to learn a function $g \in \mathcal{H}_K$ by solving:

$$\hat{g} = \underset{g\in\mathcal{H}_K}{argmin}\, L(g(x_1), g(x_2), \ldots, g(x_{n_p})) + \gamma\|g\|_{\mathcal{H}_K} \tag{10}$$

Here $L(.)$ is a loss function that depends on $g(.)$ only through $g(x_i)$ and $\|.\|_{\mathcal{H}_K}$ is the norm associated with the inner product induced by the kernel. Problem (6) fits this formalism by defining $\mathcal{X} = \{1, \ldots\ldots, n\}$, $\mathbf{K}(i,j) = \mathbf{P}_{ij}$, $g(x_i) = \theta_i$ and $L = \|\mathbf{y} - \mathbf{g}\boldsymbol{\Phi}\|_2^2$. With this choice of kernel, the corresponding Hilbert space consists of elements of the form $\mathbf{g} = \mathbf{a_g K}$, equipped with the inner product $\langle \mathbf{f}, \mathbf{g}\rangle = \mathbf{a_f K a_{g^T}}$. Thus $\|\mathbf{g}\|_{\mathcal{H}_K} = \mathbf{g}\mathbf{K}^{-1}\mathbf{g}^T$ and we recover (6).

Let $\boldsymbol{\phi}_j(.)$ denote the (normalized) eigenfuctions of $K(.,.)$, with corresponding eigenvalues λ_j. Since $\boldsymbol{\phi}_j(.)$ forms an orthonormal basis of $\mathcal{H}_K$, it follows that its elements admit a description $g(.) = \sum_k c_k\phi_k(.)$. Further, from Mercer's theorem [17], it follows that $\|g\|^2_{\mathcal{H}_K} = \sum_k \frac{c_k^2}{\lambda_k}$. Thus, an alternative, "dictionary based" formulation of (6) is

$$\hat{\mathbf{c}} = \underset{\mathbf{c}}{argmin}\, \|\mathbf{y} - \mathbf{c}\mathbf{D}_{\boldsymbol{\Phi}}\|_2^2 + \gamma\mathbf{c}\boldsymbol{\Lambda}^{-1}\mathbf{c} \tag{11}$$

where the j^{th} row of the dictionary $\mathbf{D}_{\boldsymbol{\Phi}}$ is $\boldsymbol{\phi}_j$, $\boldsymbol{\Lambda}$= diag (λ_r) and $\hat{\mathbf{g}} = \hat{\mathbf{c}}\mathbf{D}_{\boldsymbol{\Phi}}$. An advantage of the formulation (10) (or its atomic counterpart (11)) is that the kernel $\mathbf{K}$ can be used to enforce desirable properties for g such as stability. A review of different kernels and their properties can be found in [18], [19]. In particular, the first order stable spline (FOSS) kernel [18], given by $K(s,t) = \mathcal{E}(g_s g_t) \propto \alpha^{\max\{s,t\}}$ where $0 < \alpha < 1$, is attractive because it enforces exponential stability of the impulse response, while having a single tunable parameter. Further, its eigenfunctions have the explicit form [20]:

$$\phi_j(k) = \sqrt{2}\sin\frac{\alpha^k}{\sqrt{\zeta_j}},\ \zeta_j = \frac{1}{(j\pi - \frac{\pi}{2})^2} \tag{12}$$

where ζ_j is the eigenvalue associated with ϕ_j.

Kernel based approaches are desirable due to their ability to impose properties on g with a computational complexity that scales as $n_g^3 + n_g^2 n_p$, where n_g is the number of elements of g penalized in (10). However, using g to predict future values of the output or to design a controller requires finding a model, a step with comparable computational complexity. Alternatively, (11) leads to an expansion that can be used to predict future values of the output. However, since there is no sparsity prior on $\mathbf{c}$, (11) is infinite dimensional. Thus, obtaining tractable approximations requires truncating the dictionary $\mathbf{D}_{\boldsymbol{\Phi}}$. In addition, in the case of the FOSS Kernel, the resulting expansion cannot be used directly for control design.

E. Atomic Norms and Sparse Optimization

As noted in the last section, the regularized problem (10) leads to expansions of g in terms of the atoms ϕ_i. However, a sparsity prior is needed in order to obtain tractable problems. To further explore this approach, we will consider the problem of finding sparse representations of a given object in terms of the elements of a dictionary $\mathcal{A}$ (the "atoms"). If $\mathcal{A}$ is centrally symmetric ($a \in \mathcal{A} \Rightarrow -a \in \mathcal{A}$), we can assign to each point in space an "atomic norm" $\|\mathbf{g}\|_{\mathcal{A}}$ defined as [21]:

$$\|\mathbf{g}\|_{\mathcal{A}} = \inf\{t > 0\ :\ \mathbf{g} \in t\cdot\text{convex hull}(\mathcal{A})\} \tag{13}$$

Atomic norms play a key role when seeking sparse solutions to optimization problems of the form:

$$\min_{\mathbf{g}} f(\mathbf{g}) \text{ subject to } \|\mathbf{g}\|_{\mathcal{A}}\ \leq\ \tau \tag{14}$$

where τ is used to promote sparsity [21]. Note that (14) can be considered a constrained version of the regularized problem:

$$\min_{\mathbf{g}} f(\mathbf{g}) + \lambda\|\mathbf{g}\|_{\mathcal{A}} \tag{15}$$

which is similar to (11) by taking $f(\mathbf{g}) = \|\mathbf{y} - \mathbf{T_u g}\|^2$. The advantage of the formulation (14) over (15) is that it can be solved using a first order Frank-Wolfe type algorithm [22], which has an optimality gap of $\mathcal{O}(\frac{1}{\text{number of iterations}})$. Recasting system identification into an atomic norm framework requires a suitable set of atoms where the representation is sparse. As shown in [1] one such set is given by: $\mathcal{A} = \mathcal{A}_1 \cup \mathcal{A}_2 \cup \mathcal{A}_3 \cup \mathcal{A}_4$, where:

$$\mathcal{A}_1 = \left\{\Psi_p(z) = \pm\frac{(1-|p|^2)}{2}(\frac{1}{z-p} + \frac{1}{z-p^*})\ :\ p \in \mathbb{D}\right\}$$

$$\mathcal{A}_2 = \left\{\Psi_p(z) = \pm\frac{(1-|p|^2)}{2}(\frac{-j}{z-p} + \frac{j}{z-p^*})\ :\ p \in \mathbb{D}\right\}$$

$$\mathcal{A}_3 = \{\Psi_p(z) = \pm 1\}$$

$$\mathcal{A}_4 = \left\{\Psi_p(z) = \pm\frac{(1-|p|^2)}{z-p}\ :\ p \in [-\rho, \rho]\right\}$$

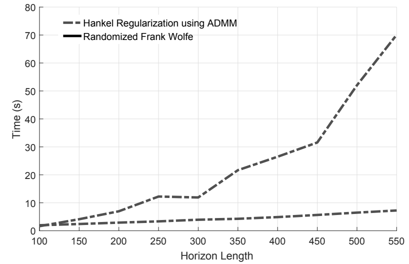

Fig. 1: Cost of Hankel norm regularized LS solved using ADMM against randomized Frank Wolfe. The cost of each ADMM iteration is $\mathcal{O}(n_M^3)$ versus $\mathcal{O}(n_M log(n_M))$ for FW.

and $\mathbb{D}$ is a suitable subset of the unit disk. A potential difficulty here is that this set of atoms is infinite dimensional. In the case of well damped plants, this difficulty can be overcome by simply gridding the unit disk [23]. However, plants with poles close to the stability boundary require using dense grids. This can be avoided using a randomized Frank-Wolfe algorithm (Algorithm 1), proposed in [1], to solve (14). As shown there, this algorithm retains the rate of convergence (albeit now in expected value) of its deterministic counterpart, that is, $\mathcal{E}\{\|\mathbf{T}_u(\mathbf{g}-\mathbf{g}_{\text{opt}})\|_2^2\} \leq \mathcal{O}(\frac{1}{\text{number of iterations}})$. In terms of computational complexity, Step 4 involves inner products of the form $\langle \nabla f(\mathbf{g}_k), \mathbf{a}\rangle$ and Step 5 admits the closed form solution $\alpha_k = \max(0, \min(\alpha_u, 1))$, where

$$\alpha_u = \frac{(\mathbf{T_u g_k} - \mathbf{y})^T(\mathbf{T_u}(\tau\mathbf{a_k} - \mathbf{g_k}))}{(\mathbf{T_u}(\tau\mathbf{a_k} - \mathbf{g_k}))^T(\mathbf{T_u}(\tau\mathbf{a_k} - \mathbf{g_k}))}$$

The computational complexity of these steps is dominated by computing products of the form $\mathbf{T}_u\mathbf{x}$ and $\mathbf{T}_u^T\mathbf{x}$. While in principle this requires $\mathcal{O}(n_M^2)$ multiplications, the Toeplitz structure of $\mathbf{T}_u$ can be exploited to compute these products in $\mathcal{O}(n_M \log n_M)$ ([24], Chapter 4). Thus, the overall computational complexity per iteration is $\mathcal{O}(n_p n_M \log n_M)$ and the algorithm can comfortable handle $\mathcal{O}(10^6)$ data points [24].

Algorithm 1 Randomized FW algorithm for LTI identification

```
1: Initialize $\mathbf{g_0} \leftarrow \tau\{\mathbf{a}_0\}$ for arbitrary $\mathbf{a}_0 \in \mathcal{A}$
2: for  $k$ = 0,1,2,3,..., $k_{max}$ do
3:     Pick $n_p$ poles uniformly distributed over $\mathbb{D}_\rho$, denote the set of these poles $S_k$
4:     $\mathbf{a_k} \leftarrow \{\text{argmin}_{\mathbf{a}\in\mathcal{A}\{S_k\}}\langle\nabla f(\mathbf{g}_k), \mathbf{a}\rangle\}$
5:     $\alpha_k \leftarrow \text{argmin}_{\alpha\in[0,1]} f(\mathbf{g_k} + \alpha[\tau\mathbf{a_k} - \mathbf{g_k}])$
6:     $\mathbf{g_{k+1}} \leftarrow \mathbf{g_k} + \alpha_k[\tau\mathbf{a_k} - \mathbf{g_k}]$
7: end for
```

F. Respecting Structure: Dynamical Graphical Models

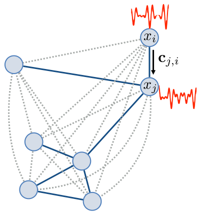

Fig. 2: A dynamical graphical model. Nodes represent time series and edges are dynamical systems operating on these.

So far, we have considered only unstructured models. However, in many practical scenarios unstructured models may fail to capture the structure of the interactions between physical agents, allowing for non-realistic interactions. Examples of these scenarios range from models of tightly interacting infrastructures (e.g. the power and communication grids) to biological systems and crowd behavior. Structured interactions can be captured by considering dynamical graphical models, represented by a directed graph structure $G = \{\mathcal{V}, \mathcal{E}\}$, where each node $\mathcal{V}$ corresponds to a given time series (the behavior of a specific agent), and the edges $\mathcal{E}$ are operators relating the values of these series at different time instants, accounting for the dynamics arising from agent interactions (see Fig. 2). The corresponding equations are

$$\begin{aligned} x_j(t) = \quad & \textstyle\sum_{i=1}^n \sum_{k=1}^r c_{ji}(k)x_i(t-k) + \eta_j(t), \\ & t \in [r+1, T_f],\ j = 1, \ldots, n \end{aligned} \tag{16}$$

where $x_j(.)$ denotes the time series at the j^{th} node, $c_{ji}(.)$ are the coefficients of an ARX model relating the present value of the time series at node j to the past values measured at node i, and $\eta_j(t)$ represents measurement noise. The goal of this section is to briefly discuss the problem of identifying these models from experimental data. Note in passing that, unless a regularization criteria is added, the problem is ill posed, since an infinite number of topologies can explain a given set of finite, noisy observations. In the absence of other priors, a suitable regularization is penalizing $|\mathcal{E}|$, the number of edges of the graph, reflecting the fact that usually the simplest solution is the correct one. Let

$$\begin{aligned}
\mathbf{x}_j &\doteq \left[x_j(T_f), \ldots, x_j(r+1)\right]^T,\ \mathbf{X} \doteq \left[\mathbf{x}_1, \ldots, \mathbf{x}_n\right] \\
\mathbf{H}_i &\doteq \begin{bmatrix} x_i(T_f-1) & x_i(T_f-2) & \ldots & x_i(T_f-r) \\ x_i(T_f-2) & x_i(T_f-3) & \ldots & x_i(T_f-r-1) \\ \vdots & \ldots & \ldots & \vdots \\ x_i(r) & \ldots & \ldots & x_i(1) \end{bmatrix} \\
\mathbf{H} &\doteq \left[\mathbf{H}_1 \ \ \ldots \ \ \mathbf{H}_n\right] \\
\boldsymbol{\eta}_j &\doteq \left[\eta_j(T), \ldots, \eta_j(r+1)\right]^T,\ \boldsymbol{\Xi} \doteq \left[\boldsymbol{\eta}_1, \ldots, \boldsymbol{\eta}_n\right] \\
\mathbf{c}_{ji} &\doteq \left[c_{ji}(1), \ldots, c_{ji}(r)\right]^T, \\
\mathbf{c}_j &\doteq \left[\mathbf{c}_{j1}^T \ldots, \mathbf{c}_{jn}^T\right]^T,\ \mathbf{C} \doteq \left[\mathbf{c}_1, \ldots, \mathbf{c}_n\right]
\end{aligned}$$

With this notation, the equations describing the complete model can be written in compact form as:

$$\mathbf{X} = \mathbf{HC} + \boldsymbol{\Xi} \tag{17}$$

and the problem of interest here reduces to:

$$\min \sum_i \|\{\mathbf{c}_i\}\|_0 \text{ s. t. (17) and } \|\boldsymbol{\eta}_i\|_2 \leq \epsilon,\ i = 1, \ldots, n$$

where $\mathbf{c}_i \in \mathbb{R}^r$ and $\|\{\mathbf{c}_i\}\|_0$ denotes the number of non-zero elements of the vector sequence $\mathbf{c}_i$. The value of the objective function is precisely $|\mathcal{E}|$ and, due to its structure, the problem decouples into n subproblems of the form:

$$\min \|\{\mathbf{c}_i\}\|_0 \text{ s. t. } \|\boldsymbol{\eta}_j\|_2 \leq \epsilon \text{ and } \mathbf{x}_j = \sum_i \mathbf{H}_i\mathbf{c}_i + \boldsymbol{\eta}_j \tag{18}$$

A computationally efficient solution to this problem can be obtained by expanding the concept of atomic norm to encompass the case where it is desired to *block*-sparsify a vector sequence. Given a set of atoms $\mathcal{A} = \{a\} \subseteq \mathcal{X}$, assume that it can be partitioned into N centrally symmetric subsets $\mathcal{A}_i$ (the super-atoms), such that $\mathcal{A} = \cup_i \mathcal{A}_i$ and $\mathcal{A}_i \cap \mathcal{A}_j = \emptyset$, $\forall\, i \neq j$ and associate to each super-atom $\mathcal{A}_i = \{a_{i1}, ..a_{in_i}\}$ the matrix $\mathbf{A}_i$ having as its j$^{\text{th}}$ column $\mathbf{a}_{ij}$, the coordinates of the atom a_{ij}

in a suitable basis in $\mathcal{X}$. Given a point $\mathbf{x} \in \mathcal{X}$, its super-atomic norm is defined as:

$$\|\mathbf{x}\|_{s\mathcal{A}} = \quad \min_{\mathbf{c}} \sum_{i=1}^{N} \|\mathbf{c}_i\|_\infty \text{ s.t } \mathbf{x} = \sum_i \mathbf{A}_i \mathbf{c}_i \tag{19}$$

Since the convex envelope of the cardinality of a vector sequence $\{\mathbf{c}\}$, $\|\mathbf{c}_i\|_\infty \leq 1$ is given by $\|\{\mathbf{c}\}\|_{0,env} = \sum_i \|\mathbf{c}_i\|_\infty$ it follows that, minimizing the super-atomic norm indeed promotes block-sparsity. Further, problems involving the minimization of a function subject to super-atomic norm constraints can be efficiently solved by using the following variant of Frank-Wolfe [25]:

Algorithm 2 Minimization of $f(\mathbf{x})$ subject to super-atomic norm constraints

```
Data: set of super-atoms $\mathcal{A} = \{\mathcal{A}_1, \dots, \mathcal{A}_i, \dots\}$
Initialize $\mathbf{x}^{(0)} \leftarrow \tau\mathbf{a}$ for some arbitrary $\mathbf{a} \in \mathcal{A}$
for  $k$ = 0,1,2,3,..., $k_{max}$ do
    $L \leftarrow \arg\min_m \left\{\min_{\|\mathbf{c}\|_\infty \leq 1} \langle \partial f(\mathbf{x}^{(k)}), \sum \mathbf{a}_{im} c_i \rangle\right.$ s.t. $\left.\mathbf{a}_{im} \in \mathcal{A}_m\right\}$
    $\mathbf{c} \leftarrow \arg \min_{\|\mathbf{c}\|_\infty \leq 1} \langle \partial f(\mathbf{x}^{(k)}), \sum \mathbf{a}_{iL} c_i \rangle$ s.t. $\mathbf{a}_{iL} \in \mathcal{A}_L$.
    $\mathbf{a} \leftarrow \sum_i \mathbf{a}_{iL} c_i$
    $\alpha_k \leftarrow \text{argmin}_{\alpha \in [0,1]} f(\mathbf{x}^{(k)} + \alpha[\tau\mathbf{a} - \mathbf{x}^{(k)}])$
    $\mathbf{x}^{(k+1)} \leftarrow \mathbf{x}^{(k)} + \alpha_k[\tau\mathbf{a} - \mathbf{x}^{(k)}]$
end for
```

The ideas discussed above can be used to solve (18) by simply defining each super-atom as $\mathcal{A}_i = \{\mathbf{H}_i(:,t)\},\ t = 1, \dots r$, the collection of columns from the matrices $\mathbf{H}_i$, leading to a super-atomic norm minimization of the form

$$\min \|\mathbf{z}\|_{s\mathcal{A}} \text{ subject to } \|\mathbf{x}_j - \mathbf{z}\|_2 \leq \epsilon \tag{20}$$

where $\mathbf{z} = \sum_i \mathbf{H}_i \mathbf{c}_i$. Finally, imposing soft, rather than hard constraints on the fitting error leads to:

$$\min \|\mathbf{x}_j - \mathbf{z}\|_2 \text{ subject to } \|\mathbf{z}\|_{s\mathcal{A}} \leq \tau \tag{21}$$

which can be efficiently solved using Algorithm 2. As before, this approach only requires computing inner products and thus can handle large data sets. Further, as shown in [25], it can be easily extended to handle unknown inputs, modeling for instance the interaction of the system with its environment. An application of these ideas to find causal interactions between human agents is shown in Fig. 3 [25].

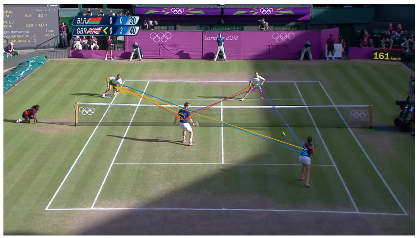

Fig. 3: Identifying causally interacting groups in a video clip [25]

G. Learning a controller directly from data

Many practical scenarios involve designing controllers when a model is not a-priori available. An interesting question is whether the observed data can be used as a proxy for the unknown model, leading to controllers designed directly from the data. Of particular interest to this paper are methods that learn a control Lyapunov function (CLF) directly from the data, since these techniques guarantee closed loop stability.

Consider data generated by an (unknown) LTI system:

$$\mathbf{x}_{k+1} \quad = \quad \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k,\ \mathbf{x} \in \mathbb{R}^n, \mathbf{u} \in \mathbb{R}^m \tag{22}$$

Assume that the system is excited with a suitable input $\mathbf{u}_k, k = 0, \dots, n_p$ and $\mathbf{x}_k$ is measured. The goal is to find a stabilizing feedback law $\mathbf{u} = \mathbf{K}\mathbf{x}$ directly from this measured data. The key observation is that, as noted in Willems' fundamental lemma [26], if the input is persistently exciting, then, in the noiseless case any input/output trajectory of an LTI system can be represented as a linear combination of collected data. Let $\mathbf{U}_{i,n_p} \doteq \begin{bmatrix}\mathbf{u}_i, \dots, \mathbf{u}_{n_p+i-1}\end{bmatrix}$, $\mathbf{X}_{i,n_p} \doteq \begin{bmatrix}\mathbf{x}_i, \dots, \mathbf{x}_{n_p+i-1}\end{bmatrix}$, and assume that u is such that rank$(\mathbf{H}_{u,n_p}) = m(n+1)$, where $\mathbf{H}_{u,n_p}$ denotes the Hankel matrix associated with $\mathbf{u}$, with $n+1$ block rows. As shown in [3], if rank$\left(\begin{bmatrix}\mathbf{U}_{0,n_p} \\ \mathbf{X}_{0,n_p}\end{bmatrix}\right) = n + m$ then, in the noiseless case, given a state feedback control law $\mathbf{u} = \mathbf{K}\mathbf{x}$, the closed loop system satisfies:

$$\mathbf{A} + \mathbf{B}\mathbf{K} = \mathbf{X}_{1,n_p}\mathbf{G}_\mathbf{K} \tag{23}$$

where $\mathbf{G}_\mathbf{K} \in \mathbb{R}^{n_p \times n}$ is any matrix satisfying

$$\begin{bmatrix}\mathbf{K} \\ \mathbf{I}_n\end{bmatrix} = \begin{bmatrix}\mathbf{U}_{0,n_p} \\ \mathbf{X}_{0,n_p}\end{bmatrix}\mathbf{G}_\mathbf{K} \tag{24}$$

Using this description to search for a quadratic CLF via Lyapunov's equation leads to one of the main results in [3]: Let $\mathbf{z}_k \doteq \mathbf{x}_k + \mathbf{w}_k$ denote noisy measurements of the states of (22). If the noise $\mathbf{w}_k$ satisfies

$$\begin{gathered}\begin{bmatrix}\mathbf{0} \\ \mathbf{W}_{0,n_p}\end{bmatrix}\begin{bmatrix}\mathbf{0} \\ \mathbf{W}_{0,n_p}\end{bmatrix}^T \preceq \gamma_1 \begin{bmatrix}\mathbf{U}_{0,n_p} \\ \mathbf{Z}_{0,n_p}\end{bmatrix}\begin{bmatrix}\mathbf{U}_{0,n_p} \\ \mathbf{Z}_{0,n_p}\end{bmatrix}^T \\ \mathbf{W}_{1,n_p}\mathbf{W}_{1,n_p}^T \preceq \gamma_2 \mathbf{Z}_{1,n_p}\mathbf{Z}_{1,n_p}^T\end{gathered} \tag{25}$$

for some $0 < \gamma_1 < 0.5,\ 0 < \gamma_2$ and there exist $\mathbf{Q} \in \mathbb{R}^{n_p \times n}, \alpha \geq 0$ such that

$$\begin{gathered}\begin{bmatrix}\mathbf{Z}_{0,n_p}\mathbf{Q} - \alpha\mathbf{Z}_{1,n_p}\mathbf{Z}_{1,n_p}^T & \mathbf{Z}_{1,n_p}\mathbf{Q} \\ \mathbf{Q}^T\mathbf{Z}_{1,n_p}^T & \mathbf{Z}_{0,n_p}\mathbf{Q}\end{bmatrix} \succ 0 \\ \begin{bmatrix}\mathbf{I}_{n_p} & \mathbf{Q} \\ \mathbf{Q}^T & \mathbf{Z}_{0,n_p}\mathbf{Q}\end{bmatrix} \succ 0 \\ \frac{6\gamma_1 + 3\gamma_2}{1 - 2\gamma_1} < \frac{\alpha^2}{2(2+\alpha)}\end{gathered} \tag{26}$$

Then, the controller $\mathbf{K} = \mathbf{U}_{0,n_p}\mathbf{Q}(\mathbf{Z}_{0,n_p}\mathbf{Q})^{-1}$ stabilizes (22). It can be shown [3] that if $\mathbf{w}_k \equiv 0$, one can take $\alpha = 0$ and ignore the second and third inequalities in (26). If $\mathbf{w}_k \not\equiv 0$, these results hold if the signal to noise ratio is large enough.

An interesting point is that, in the noiseless case, the computational complexity of identifying $\begin{bmatrix}\mathbf{A} & \mathbf{B}\end{bmatrix}$, using for instance least squares, is $\mathcal{O}((n+m)^3)$, while the computational complexity of an LMI based controller synthesis is roughly $\mathcal{O}(n^3(0.5n+m)^3)$, when using an ADMM based algorithm to solve the SDP. On the other hand, the LMI (26) has nn_p decision variables and hence its complexity is roughly $\mathcal{O}(n^3 n_p^3)$. Since for the noiseless case one can take $n_p = (m+1)n + m$, it follows that in this scenario, data driven control and the two step process have roughly the same computational complexity ($\approx n^6$). In the case of measurements corrupted by Gaussian noise, as shown in [10] for LQR control, a two step procedure based on a combination of LS identification over N roll-outs and a system-based parameterization of all controllers satisfies

$$\frac{J-J^*}{J^*} \propto \sqrt{\frac{(n+m)\log\delta^{-1}}{N}} \text{ with probability} > 1-\delta$$

where J, J^* denote the actual and optimal $\mathcal{H}_2$ cost, respectively. The cost of identifying $\mathbf{A}, \mathbf{B}$ with identification errors bounded (with probability $1-\delta$) by $\epsilon_{\mathbf{A}}, \epsilon_{\mathbf{B}}$ is $\mathcal{O}(N(n+m)^2 + (n+m)^3)$, while a robust static controller that renders the closed loop robust to these errors, can be found by solving a parametric SDP with $\mathcal{O}(n(n+2m))$ decision variables. On the other hand, approximating the stochastic noise with deterministic noise satisfying an IQC type bound and using the S-Lemma based approach proposed in [27], allows for designing a data driven controller guaranteed to stabilize all plants compatible with the observed data by finding a feasible solution to an LMI of size $(n+3m)\times(n+3m)$. This LMI has $\mathcal{O}(n(n+m))$ decision variables, with an initial cost of $\mathcal{O}((n+n_p)^2(3n+m))$, to set it up. Thus, as in the noiseless case, the complexity of the two step approach and data driven control (dominated in both cases by the number of decision variables) is similar. In the case of ℓ_∞ bounded noise, worst-case sufficient conditions guaranteeing closed loop stability of all plants compatible with the observed data can be obtained by formulating a robust super-stabilization problem and exploiting duality to reduce it to a linear program [28]. However, the ability to handle ℓ_∞ bounded noise comes at the price of increased complexity, since this LP has $\mathcal{O}(n^3 n_p)$ variables and $\mathcal{O}(n^4+n^3 m)$ constraints and thus a computational complexity of at least $\mathcal{O}(n^{10})$ since n_p should be at least n^2.

The data driven approaches covered so far can be considered “model based”, in the sense that they leverage the existence of an underlying model, even though in some cases this model is never found explicitly. This raises the question of whether “model free” approaches that directly learn and optimize a value function can outperform model based ones. A negative answer to this question for the case of LQR control has been given in [5]. As shown there, the sample complexity of estimating the value function using least squares temporal difference learning is $\propto \mathcal{O}(\frac{n}{\sqrt{N}})$, at a computational cost of $\mathcal{O}((n+m)^6)$ per iteration, which is comparable to the cost of a model-based approach. However, as noted in [5], convergence of the model free approach is substantially slower, typically requiring an order of magnitude more iterations.

III. Identification of switched systems

In the previous sections we have addressed identification of LTI systems and argued that computational complexity is driven by the order of the system. In this section we extend these ideas to the case of switched linear systems. These systems are interesting in their own, since they appear in many scenarios, and as tractable approximations to more complex non-linear dynamics. Due to the large volume of research in this area, in the sequel we will cover only a few approaches that highlight the connection between computational complexity, system order and uncertainty. The interested reader is referred for instance to [29] for a comprehensive list of the various switched systems identification techniques that have been proposed in the past two decades. It has been recently shown [30] that finding a switching model that interpolates the data with the minimum number of switches is solvable in polynomial (in n_p) time. On the other hand, many scenarios require fitting the data with a minimum (or known) number of subsystems. Examples of these situations include not only control applications (fault tolerant control and anomaly detection), but also, among others, computer vision and machine learning ones (e.g. activity recognition and subspace clustering of dynamic data). Unfortunately, the minimum number of subsystems scenario leads to a very challenging NP hard problem. Nevertheless, as described below, convex relaxations whose complexity scale linearly with the number of data points can be obtained by exploiting a connection to semi-algebraic geometry, positive measures and positive polynomials.

Consider an error-in-variables switched auto-regressive exogenous (SARX) linear model

$$y_t + \zeta_t = \sum_{k=1}^{n_a} a_k(\gamma_t)(y_{t-k}+\zeta_{t-k}) + \sum_{k=1}^{n_c} b_k(\gamma_t)(u_{t-k}+\eta_{t-k}) \tag{27}$$

where γ_t is the mode variable indicating which subsystem is active at time t. The goal is to identify the parameters $\{a_{k=1}^{n_a}(j), b_{k=1}^{n_c}(j)\}$ that characterize each of the subsystems in (27) from the input/output experimental data $(u_k,\ y_k)$ and the a-priori information $\{n_s, n_a, n_c\}$, where n_s is the number of subsystems and n_a and n_c are their orders.

A. Algebraic Reformulation With a Stochastic Perspective

Consider a trajectory of (27) corresponding to a given known input and an unknown switching sequence. Define

$$\begin{aligned}\mathbf{r}_t &= [-y_t, y_{t-1}, ..., y_{t-n_a}, u_{t-1}, ..., u_{t-n_c}]^T \\ \mathbf{b}_i &= [1, a_1(i), ..., a_{n_a}(i), b_1(i), ..., b_{n_c}(i)]^T\end{aligned} \tag{28}$$

In the noise free case, $\mathbf{b}_i^T\mathbf{r}_t = 0$ holds for all time instants t_i where $\gamma_t = i$. Thus, the corresponding regressors $\mathbf{r}_{t_i}$ live in a subspace normal to $\mathbf{b}_i$ and the vanishing ideal of the arrangement of subspaces defined by the vectors $\mathbf{b}_i, i = 1, \ldots, n_s$ is generated by the polynomial [31]:

$$p_s(\mathbf{r}_t) = \prod_{i=1}^{n_s}(\mathbf{b}_i^T\mathbf{r}_t) = \mathbf{c}_{n_s}^T\mathbf{v}_{n_s}(\mathbf{r}_t) = 0 \tag{29}$$

where $\mathbf{v}_{n_s}(.) \in \mathbb{R}^{m_s}$, with $m_s \doteq \binom{n_a+n_c+n_s}{n_s}$, denotes the Veronese map of degree n_s:

$$\mathbf{v}_{n_s}(\mathbf{r}_t) = \left[y_t^{n_s} \ldots (y_t^{\alpha_1} y_{t-1}^{\alpha_2} \ldots u_{t-n_c}^{\alpha_{n_a+n_c+1}}) \ldots u_{t-n_c}^{n_s}\right]^T$$

and where the entries of the vector $\mathbf{c}_{n_s}$ are only functions of the entries of the vectors $\mathbf{b}_i$. Evaluating this polynomial at each data point and collecting the results in a matrix leads to:

$$\begin{aligned}\mathbf{V}_{n_s}\mathbf{c}_{n_s} \doteq [\mathbf{v}_{n_s}(\mathbf{r}_1)\cdots\mathbf{v}_{n_s}(\mathbf{r}_{n_p})]^T\ \mathbf{c}_{n_s} &= 0 \\ \iff \mathbf{M}_{n_s}\mathbf{c}_{n_s} &= 0\end{aligned}$$

where $\mathbf{M}_{n_s} \doteq \frac{1}{n_p}\mathbf{V}_{n_s}^T\mathbf{V}_{n_s} \in \mathbb{R}^{m_s\times m_s}$ is the empirical moments matrix of $\mathbf{r}_t$. In the noise free case, the identification

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCSYS.2020.3045664, IEEE Control Systems Letters

problem can be solved by using the Generalized Principal Component Algorithm (GPCA) [31], [32]: find a vector $\mathbf{c}_{n_s}$ in the null space[3] of $\mathbf{M}_{n_s}$ and then recover the parameters of each subsystem via polynomial differentiation. The cost of this approach is roughly $\mathcal{O}(m_s^2 n_p + m_s^3)$, where the first term accounts for the cost of computing $\mathbf{M}_{n_s}$ and the second $\mathbf{c}_{n_s}$. Thus, it scales linearly with n_p, but combinatorially with the number of subsystems n_s and their order $n_a + n_c$.

While the approach above works well for noiseless data, a small amount of noise can lead to large identification errors. [34]. In the case where the distribution of the noise depends on a few unknown parameters, [35] proposed to "denoise" $\mathbf{M}_{n_s}$ by noting that its entries are affine functions of the moments of the noise and searching for the values of these parameters that render $\mathbf{M}_{n_s}$ rank defficient. This approach is very effective when the noise distribution depends on just a few parameters (e.g. zero mean with unknown variance), and the number of systems and their order is relatively small. On the other hand, the problem becomes challenging beyond these cases.

An alternative that does not involve an explicit optimization can be obtained by exploiting a connection between the moment matrix $\mathbf{M}_{n_s}$ and the Christoffel polynomial that approximates the support of a probability density supported on the unknown subspaces. Consider an arrangement of subspaces $\mathcal{A}(S) \doteq S_1 \cup S_2 \cup \ldots \cup S_{n_s}$, $S_i \subset \mathbb{R}^d$, where the normal to each subspace is $\mathbf{b}_i$ and let μ denote a probability measure supported in this arrangement. Given a point $\mathbf{x}_o \notin \mathcal{A}(S)$, define the following polynomial optimization problem:

$$P^*_{\mathbf{x}_o,n_s}(\mathbf{x}) = \left\{ \underset{P \in \mathcal{P}^{n_s}_{d,h}}{argmin} \int_\mu P^2(\xi) d\mu \text{ subject to } P(\mathbf{x}_o) = 1 \right\} \tag{30}$$

where $\mathcal{P}^{n_s}_{d,h}$ denotes the set of homogeneous polynomials of degree n_s in d variables. $P^*_{\mathbf{x}_o,n_s}(\mathbf{x})$ can be written as $\mathbf{v}^T_{n_s}(\mathbf{x})\mathbf{c}^*(\mathbf{x}_0)$, where an explicit expression for $\mathbf{c}^*$ in terms of the singular vectors $\mathbf{u}_i$ and singular values σ_i of $\mathbf{M}_s$ is:

$$\mathbf{c}^*(\mathbf{x}_0) = \frac{1}{\sum_{i=1}^{m_s}(\frac{1}{\sqrt{\sigma}_i}\mathbf{u}_i^T\mathbf{v}_{n_s}(\mathbf{x}_0))^2} \sum_{i=1}^{m_s} \frac{1}{\sigma_i} \mathbf{u}_i^T \mathbf{v}_{n_s}(\mathbf{x}_0)\mathbf{u}_i$$

As noted in [36], P^* provides an approximation to the complement of the support set of the measure μ, in the sense that $|P^*_{\mathbf{x}_o}(\mathbf{x})|$ is small in points where $\mu(\mathbf{x})$ is large. Similarly, $Q_{n_s}(\mathbf{x}_o) \doteq \frac{1}{\mathcal{E}\{(P^*_{\mathbf{x}_o,n_s}(\mathbf{x}))^2\}} = \mathbf{v}^T_{n_s}(\mathbf{x}_o)\mathbf{M}^{-1}_{n_s}\mathbf{v}_{n_s}(\mathbf{x}_o)$, the inverse of the Christoffel function corresponding to the Kernel induced in $\mathcal{P}^{n_s}_{d,h}$ by the measure μ, is large where μ is small. Specifically, from Markov's inequality it can be shown that

$$\begin{aligned} &\text{prob}(Q_{n_s}(\mathbf{x}_o) > t_Q m_s) < \frac{1}{t_Q} \text{ and} \\ &\text{prob }((P^*_{\mathbf{x}_o,n_s}(\mathbf{x})^2) > \frac{t_p}{Q_{n_s}(\mathbf{x}_o)}) < \frac{1}{t_p} \end{aligned} \tag{31}$$

These properties can be used to find the subspaces $\mathcal{S}_i$ "one-at-a-time" proceeding as outlined in Algorithm 3 [37]. Let $\hat{\mu}_{\bar{\mathcal{S}}_j}$ denote the distribution of the points supported in $\bar{\mathcal{S}}_j \doteq \cup_{i \neq j} S_i$. The idea is to select a point $\mathbf{x_o} \in \mathcal{S}_j$ and treat this as an outlier to the distribution $\hat{\mu}_{\bar{\mathcal{S}}_j}$ so that $P^*_{\mathbf{x}_o}(\mathbf{x})$ locally approximates the support of $\mathcal{S}_j$. Hence points where $P^*_{\mathbf{x}_o}(\mathbf{x})$ is large can be assigned to $\mathcal{S}_j$. These points are then removed from the population and the algorithm proceeds to the next subsystem. Step 3 selects as the next point to be considered an "outlier" the one having the lowest Q. Intuitively, this choice corresponds to the point that has more mass around it and thus, the highest number of remaining points in the same subspace. Step 4 uses $P^*_{\mathbf{x}_o}(\mathbf{x})$ to find other points in the same subspace. These points are removed from the population and the process continues with one less subspace. Finally, step 10 assigns unassigned points to the subspaces where they have the highest probability of being inliers. The overall complexity of the method is comparable with that of GPCA.

Algorithm 3 One at a time algebraic SARX identification

```
 1: Inputs:
        $\mathbf{X_y} \leftarrow \mathbf{H}_{\mathbf{y}_t} \in \mathbb{R}^{(n_a+1)\times(N-n_a-1)}$
        $\mathbf{X_u} \leftarrow \mathbf{H}_{\mathbf{u}_t} \in \mathbb{R}^{n_c\times(N-n_a-1)}$
        $\mathcal{X}_a \leftarrow \begin{bmatrix}\mathbf{X_y}\\ \mathbf{X_u}\end{bmatrix}$,$n_s \leftarrow$ number of subsystems, $k \leftarrow 1$
 2: for $k := n_s$ to 2 do
 3:     Set $\mathbf{x}_o = argmin_{\mathbf{x}}\, \mathbf{v}^T_{k-1}(\mathbf{x})\mathbf{M}_{k-1}\mathbf{v}_{k-1}(\mathbf{x})$.
 4:     Compute $P_{\mathbf{x}^*_o,k}(\mathbf{x})$.
 5:     Select $t_p$ and assign points where $P^2_{\mathbf{x}_{o,k}}(\mathbf{x}) \geq \frac{t_p}{Q_{k-1}(\mathbf{x}_o)})$ to the subspace $\mathcal{S}_k$.
 6: end for
 7: for $j := 1$ to $n_s$ do
 8:     Compute $Q_{1,\mathcal{S}_j}(\mathbf{x})$ for each subspace
 9: end for
10: Assign each point $\mathbf{x}$ to the cluster $j$ corresponding to the smallest $Q_{1,\mathcal{S}_j}(\mathbf{x})/\|Q_{1,\mathcal{S}_j}(.)\|$
```

B. A spectral clustering based approach

An alternative to the algebraic approach outlined above is to approach the switched systems identification problem from a clustering viewpoint and use tools developed in the machine learning community to solve these problems. In particular, we will focus on spectral clustering techniques that recast the problem into a (generalized) eigenvalue problem. In this context the data is represented using a similarity graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$ where each node $V_i \in \mathcal{V}$ corresponds to a data point $\mathbf{x}_i$, $\mathcal{E}$ is the set of edges connecting these nodes, and each element W_{ij} of the weighting matrix $\mathbf{W} \in \mathbb{R}^{n_p \times n_p}$ measures the similarity between V_i and V_j. The corresponding Degree and Laplacian matrix are given by:

$$\mathbf{D} = \text{diag}\{d_1, \ldots, d_n\}; \; \mathbf{L} = \mathbf{D} - \mathbf{W} \text{ where } d_i = \sum_j W_{ij}$$

It can be shown [38] that $\mathbf{L}$ is positive semi-definite and always has an eigenvalue at zero. In the ideal case where $w_{i,j} = 0$ if $\mathbf{x}_i, \mathbf{x}_j$ belong to different connected components in the graph, the multiplicity of this zero eigenvalue equals

[3] Under mild conditions this vector is unique up to a scaling constant, since the vanishing ideal of the arrangement is a principal ideal [33].

the number of connected components in the graph and the corresponding eigenspace is spanned by the indicator vectors of those components. In the non-ideal case, if w_{ij} is small when $\mathbf{x}_i, \mathbf{x}_j$ do not belong to the same component, then the number of close-to-zero eigenvalues indicates the number of components [29], [38], with the corresponding eigenvectors characterizing each of the clusters.

In principle these ideas can be used to segment the experimental data into clusters where a single system is active and then using any LTI identification method to recover a model from these clusters. However, implementing this approach requires defining a suitable similarity measure between two data segments $d_s \doteq \{y_k, u_k\}_{k=s}^{s+h-1}$ and $d_t \doteq \{y_k, u_k\}_{k=t}^{t+h-1}$ such that $w_{t,s}$ is large only if both segments where generated by the same subsystem. A suitable similarity measure that respects the underlying dynamics can be obtained form the Hankel matrices corresponding to d_t, d_s as follows. Given positive semidefinite matrices $\mathbf{X}, \mathbf{Y}$ the regularized Jensen-Bregman log det divergence is defined as:

$$J_{ld,\sigma}(\mathbf{X}, \mathbf{Y}) = \log \frac{|0.5(\mathbf{X} + \mathbf{Y}) + \sigma \mathbf{I}|}{|(\mathbf{X} + \sigma \mathbf{I})(\mathbf{Y} + \sigma \mathbf{I})|^{\frac{1}{2}}} \tag{32}$$

As shown in [39], $\lim_{\sigma \to 0} J_{ld,\sigma}(\mathbf{X}, \mathbf{Y}) < \infty$ if and only if $\mathbf{X}, \mathbf{Y}$ share the same null space. Further, if $\|\mathbf{X}\|_* = \|\mathbf{Y}\|_* = 1$, then there exists $\underline{\sigma} > 0$ that depends only on the angle between the null spaces of $\mathbf{X}, \mathbf{Y}$ such that for $\sigma \leq \underline{\sigma}$, there exists τ such that $J_{ld,\sigma}(\mathbf{X}, \mathbf{Y}) < \tau$ if $\mathbf{X}, \mathbf{Y}$ have the same null space and $J_{ld,\sigma}(\mathbf{X}, \mathbf{Y}) > \tau$ otherwise.

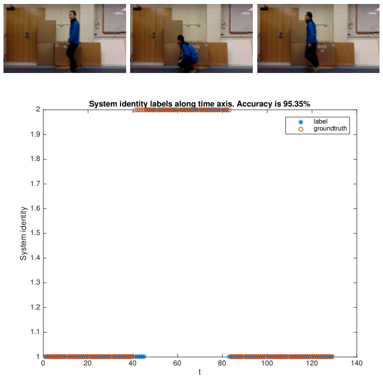

Fig. 4: Top: Sample frames of a subject walking and squatting. Bottom: Assigned labels (blue) versus ground truth (red). The J_{ld} based segmentation achieved 95.35% accuracy with a modest computational burden (0.29s to process 130 frames).

Consider now two data segments d_s, d_t such that the corresponding Hankel matrices, $\mathbf{H}_s, \mathbf{H}_t$, each having $n = n_a + n_c + 1$ rows and $n_{\text{col}} \geq n$ colums, are rank deficient (and hence the data has been generated by a system of order at most n). Under appropriate minimality and persistence of excitation hypothesis, d_s, d_t are generated by the same system if and only if the corresponding Hankel matrices, and hence the Gramians $\mathbf{G} \doteq \frac{\mathbf{H}\mathbf{H}^T}{\|\mathbf{H}\mathbf{H}^T\|_*}$, share the same left null space. It follows that a suitable similarity score is given by $w_{t,s} \doteq e^{-J_{ld,\sigma}(\mathbf{G}_s, \mathbf{G}_t)}$. In the ideal, noiseless case, when $\sigma \to 0$, $J_{ld} \to \infty$ and hence $w_{t,s} \to 0$ unless d_s, d_t have been generated by the same subsystem. Thus $w_{t,s}$ provides perfect separation between clusters This observation leads to a two step procedure for segmenting the data and identifying clusters generated by the same subsystem. In the first step, a sliding window is used to segment the data by searching for points where there is a sharp increase in the J_{ld}, indicating a switch. Next, segments are grouped into clusters generated by a single subsystem by performing a spectral clustering step using the similarity score $w_{t,s}$. Under suitable dwell time constraints, this approach is guaranteed to correctly cluster all segments corresponding to the same underlying dynamics even in the presence of noise, provided that the noise level is below a given threshold related to the subspace angle between the subspaces spanned by each subsystem [29]. An application of these ideas to the problem of segmenting a video containing multiple activities into sub-activities, each characterized by an affine model is shown in Fig. 4.

In terms of computational complexity, the J_{ld} can be computed in $\mathcal{O}(n^3)$, and this computation has to be performed $\mathcal{O}(n_p)$ times, to segment the data. Once the data is segmented, the cost of the spectral clustering step is $\mathcal{O}(n_s^3)$, where n_s is the number of segments, or, equivalently, the number of switches. Thus, the overall cost of the algorithm is roughly $\mathcal{O}(n_s^3 + n_p n^3)$. Hence computational complexity scales linearly with the number of data points but, contrary to algebraic approaches, only polynomially, rather than combinatorial, in the number of subsystems and their order. On the other hand, this approach requires a dwell time $T > 3n_a + n_c$, while the algebraic approach can be applied to arbitrarily fast switching.

C. A moments based approach

An alternative approach to SARX identification is to recast the problem into a quadratically constrained feasibility problem. This approach does require neither dwell time nor small noise assumptions and is interpolatory, at the price of increased computational burden.

Let $\mathbf{b}_i$ and $\mathbf{r}_k$ be defined as in (28) and introduce a variable $s_{i,j} \in \{0, 1\}$ that indicates whether the submodel $\mathcal{S}_i$ is active at time j. Then, there exist $\mathbf{b}_i$, $i = 1, \ldots, n_s$ such that for all k, $|\mathbf{b}_i^T \mathbf{r}_k| \leq \epsilon$ for at least one i if and only if the following set of inequalities in the indicator variables s_{ij} is feasible:

$$|s_{i,j} \mathbf{b}_i^T \mathbf{r}_j| \leq \epsilon s_{i,j}, \forall_{i=1}^{n_s} \forall_{j=1}^{n_p} \tag{33a}$$

$$s_{i,j}^2 = s_{i,j}, \; \forall_{i=1}^{n_s} \forall_{j=1}^{n_p} \tag{33b}$$

$$\Sigma_{i=1}^{n_s} s_{i,j} = 1, \forall_{j=1}^{n_p} \tag{33c}$$

$$\mathbf{b}_i^T \mathbf{b}_i = 1, \forall_{i=1}^{n_s} \tag{33d}$$

Here (33a) is equivalent to $|\mathbf{b}_i^T \mathbf{r}_j| \leq \epsilon$ if $s_{i,j} \neq 0$ (hence $\mathbf{x}_j \in \mathcal{S}_i$) and trivially satisfied otherwise; (33b) imposes that $s_{i,j} \in \{0, 1\}$ and (33c) forces each sample $\mathbf{r}_i$ to be assigned to exactly one subspace; Thus, if (33) is feasible, then the identified sub-systems are characterized by the models $\{\mathbf{b}_i\}$. On the other hand, infeasibility of (33) indicates that the observed data cannot be explained (within the noise level) using n_s submodels. Collecting all variables in a vector $\mathbf{v}$

$$\mathbf{v} \doteq \quad [\mathbf{b}_1^T, \cdots, \mathbf{b}_{n_s}^T, s_{1,1}, \cdots, s_{n_s,1}, \cdots, s_{1,n_p}, \cdots, s_{n_s,n_p}]^T$$

and defining the rank 1 matrix $\mathbf{M} = \begin{bmatrix} 1 & \mathbf{v}^T \\ \mathbf{v} & \mathbf{v}\mathbf{v}^T \end{bmatrix}$, the inequalities in (33) can be written in compact form as $\text{Trace}(\mathbf{Q}_k \mathbf{M}) \leq 0, \forall_{k=1}^{K}$ where $K = n_s + n_p(n_s + 1)$ (the constraints (33b) are enforced by simply using the same variable for s_{ij} and

s_{ij}^2 in $\mathbf{M}$). Thus, the SARX problem can be reduced to the following constrained rank minimization:

$$\min \text{rank}(\mathbf{M}) \text{subject to: } \begin{cases} \text{Tr}(\mathbf{Q}_k\mathbf{M}) \leq 0, \forall_{k=1}^{K} \\ \mathbf{M} \succeq \mathbf{0},\ \mathbf{M}(1,1) = 1 \end{cases} \tag{34}$$

Clearly, the original problem is feasible if and only if this problem admits a rank-1 solution. Interestingly, as shown in [40], forcing $\mathbf{M}_o$, the top left submatrix of $\mathbf{M}$ involving only the variables $\mathbf{b}_i$, to be rank 1 is sufficient to enforce rank$(\mathbf{M}) = 1$. Replacing rank by its convex surrogate, trace [41], leads to the following relaxation:

$$\min \text{Tr}(\mathbf{M}_o) \text{ subject to: } \begin{cases} \text{Tr}(\mathbf{Q}_k\mathbf{M}) \leq 0, \forall_{k=1}^{K} \\ \mathbf{M} \succeq \mathbf{0},\ \mathbf{M}(1,1) = 1 \end{cases} \tag{35}$$

The approach outlined above works well for moderately sized problems. However, the computational complexity of solving (34) is $\mathcal{O}(n_s^4 n_p^4)$, if using an interior point method, or $\mathcal{O}(n_s^3 n_p^3)$ per iteration, in the case of an ADMM based algorithm, limiting the approach to relatively few points. As shown next, these difficulties can be circumvented by exploiting the sparse structure of the problem. To this effect, partition the constraints in (33) into the $n_p + 1$ sets $\mathcal{P}_j$, $j = 0, 1, \ldots, n_p$:

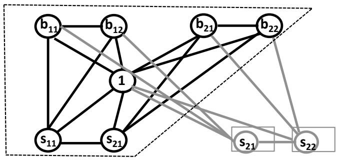

Fig. 5: Correlative sparsity graph corresponding to (33) for points in two lines in $\mathbb{R}^2$, showing a sample clique (enclosed by the doted line). For simplicity only the portion corresponding to two points is shown.

$$\mathcal{P}_0 : \left\{ \mathbf{b}_i^T \mathbf{b}_i = 1, \forall_{i=1}^{n_s} \right.$$

$$\forall_{j=1}^{n_p}, \mathcal{P}_j : \begin{cases} |s_{i,j}\mathbf{b}_i^T\mathbf{r}_j| \leq \epsilon s_{i,j}, \forall_{i=1}^{n_s} \\ s_{i,j}^2 = s_{i,j}, s_{i,j} \geq 0,\ \forall_{i=1}^{n_s} \\ \sum_{i=1}^{n_s} s_{i,j} = 1. \end{cases}$$

It is easily seen that $\mathcal{P}_0$ is only associated with variables $\mathbf{v}_0 = [\mathbf{b}_1^T, \ldots, \mathbf{b}_{n_s}^T]^T \in \mathbb{R}^{nn_s}$ and $\mathcal{P}_j$ is only associated with variables $\mathbf{v}_j = [\mathbf{v}_0^T, s_{1,j}, \ldots, s_{n_s,j}]^T \in \mathbb{R}^{(n+1)n_s}$. It can be shown that the sets $\mathbf{v}_j$ provide a clique decomposition of the correlative sparsity graph[4] associated with (35). Hence, from Grone's Theorem (see the Appendix), the constraint $\mathbf{M} \succeq 0$ can be replaced by $\mathbf{M}_j \succeq 0$, where $\mathbf{M}_j = \begin{bmatrix} 1 & m_j(\mathbf{v}_j^T) \\ m_j(\mathbf{v}_j) & m_j(\mathbf{v}_j\mathbf{v}_j^T) \end{bmatrix}$ for $j = 0, 1, \ldots, n_p$, and where $m_j(\bullet)$ is a variable located in the same position as $\bullet$ in $\begin{bmatrix} 1 & \mathbf{v}_j^T \\ \mathbf{v}_j & \mathbf{v}_j\mathbf{v}_j^T \end{bmatrix}$ The advantage of this reformulation is that, contrary to (34), it involves $n_p + 1$ matrices of dimension at most $[(n+1)n_s + 1] \times [(n+1)n_s + 1]$. Hence, it can be solved using an ADMM algorithm with cost $\mathcal{O}(n_p n^3 n_s^3)$, per iteration, which scales linearly with the number of data points.

IV. Prospects for nonlinear systems

Learning the parameters of non-linear systems with a given structure is considerably more involved, even if a portion of the model is known. For example, [42] established that worst case identification of the linear portion of a Wiener model from noisy data is generically NP hard in the number of data points *even if the nonlinearity is known*. On the other hand, depending on the scenario, it may suffice to learn a predictive black box model, without any attempt to impose a specific structure. This approach leads to nonlinear autoregressive models (NARX), where the next output is expanded in terms of given functions of its past values, and the goal is to learn the parameters of this expansion. A difficulty here is determining the correct basis functions while avoiding overfitting. Recently, substantial interest has been devoted to the use of Koopman operator based methods [43]–[45] as an alternative to NARX approaches. Given a non-linear discrete time system of the form:

$$\boldsymbol{\xi}_{k+1} = f(\boldsymbol{\xi}_k) \text{ where } \boldsymbol{\xi}_k = \begin{bmatrix} \mathbf{x}_{k-n+1}^T & \ldots & \mathbf{x}_k^T \end{bmatrix}^T,\ \mathbf{x}_j \in \mathbb{R}^d \tag{36}$$

let $\mathbb{H}$ denote a Hilbert space of functions $\boldsymbol{\psi}(\boldsymbol{\xi})\colon \mathbb{R}^{dn} \to \mathbb{R}^{mn}$(the observables). The Koopman $\mathcal{K}$ operator acts on the elements of $\mathbb{H}$, by propagating their values one step into the future:

$$(\mathcal{K} \circ \boldsymbol{\psi})(\boldsymbol{\xi}_k) = (\boldsymbol{\psi} \circ f)(\boldsymbol{\xi}_k) = \boldsymbol{\psi}(\boldsymbol{\xi}_{k+1}) \tag{37}$$

$\mathcal{K}$ is a linear operator, albeit typically infinite dimensional. When it has a countable set of eigenfunctions $\boldsymbol{\phi}_i(.)$ with eigenvalues μ_i, the observables $\boldsymbol{\psi}(.)$ can be propagated as follows. Let $\mathbf{a} = \begin{bmatrix} a_1 \ldots \end{bmatrix}^T$ denote the coordinates of $\boldsymbol{\psi}(.)$ in the basis spanned by $\boldsymbol{\phi}(.)$, that is

$$\boldsymbol{\psi}(.) = \sum a_i \boldsymbol{\phi}_i(.) \doteq \boldsymbol{\Phi}(.)\mathbf{a}, \text{ where: } \boldsymbol{\Phi}(.) = \begin{bmatrix} \boldsymbol{\phi}_1(.) \ldots \end{bmatrix}$$

Then

$$(\mathcal{K} \circ \boldsymbol{\psi})(.) = \sum a_i \mu_i \boldsymbol{\phi}_i(.) = \boldsymbol{\Phi}(.)\mathbf{M}\mathbf{a}, \text{where } \mathbf{M} = \text{diag}(\mu_i)$$

In particular, if the state $\boldsymbol{\xi} \in \text{span}\{\boldsymbol{\phi}_i\}$, then $\boldsymbol{\xi}_{k+1} = \boldsymbol{\Phi}(\boldsymbol{\xi}_k)\mathbf{M}\mathbf{a}$. While this approach allows for finding linear representations of (36), identifying the Koopman eigenfunctions from data is not trivial. Recent Deep Learning motivated approaches [44], [45] proposed encoder/decoder type architectures that map states $\boldsymbol{\xi}$ to latent variables $\mathbf{y}$ and impose approximately linear dynamics for the evolution of the latter. A salient feature of these approaches is that the states $\boldsymbol{\xi}$ are no longer required to be in the span of the Koopman eigenfuctions. As shown in [45], the use of a nonlinear decoder to map $\mathbf{y}$ back to $\boldsymbol{\xi}$ avoids overfitting. Still, at the moment there is no systematic way of selecting some of the parameters (e.g dimension of the latent variables, order of the dynamics). As an alternative, the recent work in [46] proposes a convex optimization approach to data-driven identification of Koopman operators. This approach uses delay coordinates and kernel based methods to identify a manifold of latent variables where the dynamics are linear. As shown in [46], the problems of finding the embedding manifold, the associated Koopman operators and the mapping back to state-space can be recast as rank-constrained SDPs. In turn, these can be relaxed to convex optimizations using the standard weighted nuclear

[4] This graph has one vertex per variable with edges connecting vertices corresponding to variables that appear in the same constraint.

norm surrogate for rank. Interestingly, these SDPs exhibit chordal sparsity where the size of the cliques is now related to both the "memory" of the system and the local geometry of the nonlinearity, allowing once again for algorithms that scale linearly with the number of data points, but polynomially with the order of the dynamics

In terms of data driven control of nonlinear systems, [47] has shown that if the dynamics admit an expansion

$$f(x,u) = \sum_i a_i \phi_i(x) + b_i \gamma_i(x) u$$

in terms of a know basis, and noisy measurements of the state x are available for training, then a controller guaranteed to stabilize all plants compatible with the a-priori information and experimental data can be synthesized by learning a scalar density function via a sum-of-squares optimization. While successful, at this point this technique is limited to relatively simple systems, due to the computational complexity of this optimization. The issue of whether this computational complexity can be mitigated by searching for descriptions (and associated controllers) with probabilistic, rather than worst-case, correctedness guarantees is wide open.

V. Conclusions

A wide range of applications with potential for profound societal impact have become feasible thanks to the ease of collecting data. However, achieving their full potential requires addressing the challenge of extracting control relevant information from large amounts of data. This problem has proven to be surprisingly difficult, compared to other "big data" scenarios that can be routinely handled by modern machine learning. As pointed out in this paper, this difficulty arises from the "interconnectivity" of data generated by dynamical systems. Indeed, one of the main messages of the paper is that the limiting factor in control oriented learning is the "memory" of the system. This validates the common wisdom of using low order models, if need be with higher fitting error to be handled by a robust controller. Low order models are desirable not only because the order of the model is reflected in the order of the controller, but also because the computational complexity of control oriented learning increases at least polynomially with the order of the model. On the other hand, the interconnectivity of the data offers a way to mitigate this complexity by exploiting the underlying structure (reflected for instance in the chordal sparsity of an associated graph) to develop algorithms that scale linearly with the number of data points.

VI. Acknowledgements

The author is grateful to the anonymous reviewers, the Associate Editor and the Senior Editor for the many suggestions for improving the original manuscript.

References

[1] B. Yilmaz, K. Bekiroglu, C. Lagoa, and M. Sznaier. A randomized algorithm for parsimonious model identification. *IEEE Transactions on Automatic Control*, 63(2):532–539, Feb 2018.

[2] T. Dai and M. Sznaier. A moments based approach to designing mimo data driven controllers for switched systems. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 5652–5657, 2018.

[3] C. De Persis and P. Tesi. Formulas for data-driven control: Stabilization, optimality, and robustness. *IEEE Transactions on Automatic Control*, 65(3):909–924, 2020.

[4] L. Breiman. Hinging hyperplanes for regression, classification and function approximation. *IEEE Trans. Inf. Theory*, pages 999–1013, 1993.

[5] Stephen Tu and Benjamin Recht. Least-squares temporal difference learning for the linear quadratic regulator. In *Proceedings of the 35th ICML*, pages 5005–5014, Jul 2018.

[6] Sahin Lale, Kamyar Azizzadenesheli, Babak Hassibi, and Anima Anandkumar. Logarithmic regret bound in partially observable linear dynamical systems. *https://arxiv.org/pdf/2003.11227.pdf*, 2020.

[7] H. Zeiger and A. McEwen. Approximate linear realizations of given dimension via ho's algorithm. *IEEE Transactions on Automatic Control*, 19(2):153–153, 1974.

[8] Yang Zheng and Na Li. Non-asymptotic identification of linear dynamical systems using multiple trajectories, 2020.

[9] Samet Oymak and Necmiye Ozay. Non-asymptotic identification of lti systems from a single trajectory. *https://arxiv.org/pdf/1806.05722.pdf*, 2019.

[10] Sarah Dean, Horia Mania, Nikolai Matni, Benjamin Recht, and Stephen Tu. On the sample complexity of the linear quadratic regulator. *Foundations of Computational Mathematics*, 20(4):633–679, 2020.

[11] R. Sánchez Peña and M. Sznaier. *Robust Systems Theory and Applications*. Wiley & Sons, Inc., 1998.

[12] P.A. Parrilo, M. Sznaier, R.S. Sanchez Pena, and T. Inanc. Mixed time/frequency-domain based robust identification. *Automatica*, 34(11):1375 – 1389, 1998.

[13] R. Singh and M. Sznaier. A loewner matrix based convex optimization approach to finding low rank mixed time/frequency domain interpolants. In *2020 American Control Conference (ACC)*, 2020.

[14] M. Sznaier, M. Ayazoglu, and T. Inanc. Fast structured nuclear norm minimization with applications to set membership systems identification. *IEEE Transactions on Automatic Control*, 59(10):2837–2842, 2014.

[15] G. Pillonetto, F. Dinuzzo, T. Chen, G. De Nicolao, and L. Ljung. Kernel methods in system identification, machine learning and function estimation: A survey. *Automatica*, 50(3):657 – 682, 2014.

[16] Yue Sun, Samet Oymak, and Maryam Fazel. Finite sample system identification: Optimal rates and the role of regularization. In *Proceedings of the 2nd Conference on Learning for Dynamics and Control*, pages 16–25, 2020.

[17] F. Cucker and S. Smale. On the mathematical foundations of learning. *Bulletin of the American mathematical society*, 39(1):1–49, 2002.

[18] G. Pillonetto and G. De Nicolao. A new kernel-based approach for linear system identification. *Automatica*, 46(1):81 – 93, 2010.

[19] T. Chen, H. Ohlsson, and L. Ljung. On the estimation of transfer functions, regularizations and gaussian processes revisited. *Automatica*, 48(8):1525 – 1535, 2012.

[20] G. Pillonetto, T. Chen, A. Chiuso, G. De Nicolao, and L. Ljung. Regularized linear system identification using atomic, nuclear and kernel-based norms: The role of the stability constraint. *Automatica*, 69:137 – 149, 2016.

[21] V. Chandrasekaran, B. Recht, P. A. Parrilo, and A. S. Willsky. The Convex Geometry of Linear Inverse Problems. *Foundations of Computational Mathematics*, 12(6):805–849, October 2012.

[22] A. Tewari, P. Ravikumar, and I. S. Dhillon. Greedy algorithms for structurally constrained high dimensional problems. *Advances in Neural Information Processing Systems 24*, pages 882–890, 2011.

[23] P. Shah, B. N. Bhaskar, G. Tang, and B. Recht. Linear system identification via atomic norm regularization. In *51st IEEE Conf. Dec. Control*, pages 6265–6270, Dec 2012.

[24] Burak Yilmaz. *Sparsity Based Methods in Systems Identification*. PhD thesis, Northeastern University, 2015.

[25] Y. Wang, O. Camps, and M. Sznaier. A super-atomic norm minimization approach to identifying sparse dynamical graphical models. *American Control Conference*, 2016.

[26] J. C. Willems, P. Rapisarda, I. Markovsky, and B. de Moor. A note on persistency of excitation. *Systems & Control Letters*, 54:325–329, 2005.

[27] H. J. van Waarde, M. K. Camlibel, and M. Mesbahi. From noisy data to feedback controllers: non-conservative design via a matrix s-lemma. *https://arxiv.org/abs/2006.00870*, 2020.

[28] T. Dai and M. Sznaier. Data driven robust superstable control of switched systems. *IFAC-PapersOnLine*, 51(25):402 – 408, 2018.

[29] X. Zhang, M. Sznaier, and O. Camps. Set membership efficient identification of error-in variables switched systems using a riemannian embedding, 2020.

[30] N. Ozay. An exact and efficient algorithm for segmentation of arx models. In *2016 American Control Conference*, pages 38–41, 2016.

[31] R. Vidal, S. Soatto, Y. Ma, and S. Sastry. An algebraic geometric approach to the identification of a class of linear hybrid systems. In *42nd Conf. on Dec. and Control*, volume 1, pages 167–172, 2003.

[32] R. Vidal, Y. Ma, and S. Sastry. Generalized principal component analysis (gpca). *IEEE Trans. PAMI*, 27(12):1945–1959, 2005.

[33] Y. Ma, A. Y Yang, H. Derksen, and R. Fossum. Estimation of subspace arrangements with applications in modeling and segmenting mixed data. *SIAM review*, 50(3):413–458, 2008.

[34] N. Ozay, C. Lagoa, and M. Sznaier. Set membership identification of switched linear systems with known number of subsystems,. *Automatica.*, pages 180–191, January 2015.

[35] S. Hojjatinia, C.. Lagoa, and F. Dabbene. Identification of switched autoregressive exogenous systems from large noisy datasets. *International Journal of Robust and Nonlinear Control*, April 2020.

[36] E. Pauwels and J. B Lasserre. Sorting out typicality with the inverse moment matrix sos polynomial. In *Advances in Neural Information Processing Systems*, pages 190–198, 2016.

[37] B. Ozbay, O. Camps, and M. Sznaier. Efficient identification of error-in-variables switched systems via a sum-of-squares polynomial based subspace clustering method. In *58th IEEE Conf. Dec. and Control*, pages 3429–3434, 2019.

[38] Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.

[39] X. Zhang, Y. Wang, M. Gou, M. Sznaier, and O. Camps. Efficient temporal sequence comparison and classification using gram matrix embeddings on a riemannian manifold. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[40] Y. Cheng, Y. Wang, O. Camps, and M. Sznaier. Subspace clustering with priors via sparse quadratically constrained quadratic programming. *CVPR*, 2016.

[41] K. Mohan and M. Fazel. Iterative reweighted algorithms for matrix rank minimization. *J. of Machine Learning Research*, 13(110):3441–3473, 2012.

[42] M. Sznaier. Computational complexity analysis of set membership identification of hammerstein and wiener systems. *Automatica*, 45(3):701 – 705, 2009.

[43] I. Mezić. Analysis of Fluid Flows via Spectral Properties of the Koopman Operator. *Annual Review of Fluid Mechanics*, 45:357–378, January 2013.

[44] B. Lusch, J. N. Kutz, and S. L. Brunton. Deep learning for universal linear embeddings of nonlinear dynamics. *Nature Communications*, 9(1), 2018.

[45] S.E. Otto and C. W. Rowley. Linearly recurrent autoencoder networks for learning dynamics. *SIAM Journal on Applied Dynamical Systems*, 18(1):558–593, 2019.

[46] M. Sznaier. A convex optimization approach to learning koopman operators, 2020.

[47] T. Dai and M. Sznaier. A semi-algebraic optimization approach to data-driven control of continuous-time nonlinear systems. *IEEE Control Systems Letters*, 5(2):487–492, 2021.

[48] F. Alizadeh, J.-P. Haeberly, and M. L. Overton. Primal-dual interior-point methods for semidefinite programming: Convergence rates, stability and numerical results. *SIAM Journal on Optimization*, 8(3):746–768, 1998.

[49] Z. Wen, D. Goldfarb, and W. Yin. Alternating direction augmented lagrangian methods for semidefinite programming. *Math. Prog. Comp.*, 2:203–230, 2010.

[50] R. Grone, C. R. Johnson, E. M. Sá, and H. Wolkowicz. Positive definite completions of partial hermitian matrices. *Linear Algebra Appl.*, 58:109–124, 1984.

[51] Y. Zheng, G. Fantuzzi, A. Papachristodoulou, P. Goulart, and A. Wynn. Chordal decomposition in operator-splitting methods for sparse semidefinite programs. *Mathematical Programming*, 180(1):489–532, 2020.

[52] J. Dancis. Positive semidefinite completions of partial hermitian matrices. *Linear Algebra and its Applications*, 175:97 – 114, 1992.

Appendix: Solving Large SDPs

Many learning problems reduce to an SDP of the form:

$$\begin{aligned}&\min_{\mathbf{X}\succeq 0} \text{Trace } (\mathbf{C}\mathbf{X}) \text{ subject to}\\ &\text{Trace } (\mathbf{A}_i\mathbf{X}) = b_i, i=1,\ldots,m,\ \mathbf{X},\mathbf{A}_i \in \mathbb{R}^{n\times n}\end{aligned} \tag{38}$$

In the case of unstructured $\mathbf{X}$, this problem can be efficiently solved using interior point methods. These methods converge in few iterations, each having a computational complexity of $m^2n^2 + mn^3$ [48]. Alternatively, ADMM based methods [49] avoid computing the Hessian, thus reducing computational complexity to $mn^2 + n^3$ per iteration, at the cost of more iterations. An advantage of ADMM methods is that they can be easily adapted to handle structured variables [14], and, as discussed below, to exploit sparsity.

Typically, in most problems arising in this paper, only a small number of entries of $\mathbf{X}$ appear in the objective and trace constraints, while the role of the other entries is just to enforce that $\mathbf{X} \succeq 0$. Thus, these variables do not have to be explicitly found, allowing for a substantial computational complexity reduction. Specifically, to the optimization (38) one can associate a graph $\mathcal{G}(\mathcal{V},\mathcal{E})$ with n vertices in $\mathcal{V}$ and edge set $\mathcal{E}$, where there is an edge between vertices j and ℓ if the element (j,ℓ) of $\mathbf{C}$ or any of the matrices $\mathbf{A}_i$ is nonzero. Given a graph $\mathcal{G}$, define the cone

$$\mathbb{S}^n_+(\mathcal{E},?) \doteq \{\mathbf{X} \in \mathbb{S}^n_+ \colon \mathbf{X}_{i,j} \text{ given if } (i,j)\in\mathcal{E}\}$$

that is, the cone of matrices with entries fixed over the edges $\mathcal{E}$ than can be completed to be PSD. When the graph $\mathcal{G}$ is chordal, the cone $\mathbb{S}^n_+(\mathcal{E},?)$ can be characterized using the following result (Grone's Theorem):

Theorem 1 (Grone [50]). *Let $\mathcal{G}(\mathcal{V},\mathcal{E})$ be a chordal graph with a set of maximal cliques $\{\mathcal{C}_1,\ldots,\mathcal{C}_{n_c}\}$. Then, $\mathbf{X}\in\mathbb{S}^n_+(\mathcal{E},?)$ if and only if $\mathbf{X}_k = \mathbf{E}_{\mathcal{C}_k}\mathbf{X}\mathbf{E}^T_{\mathcal{C}_k} \in \mathbb{S}^{|\mathcal{C}_k|}_+$, $k=1,\ldots,n_c$, where the $0/1$ matrix $\mathbf{E}_{\mathcal{C}_k}$ selects the variables of $\mathbf{X}$ corresponding to edges in the clique $\mathcal{C}_k$.*

This results allows for decomposing the large PSD constraint into a collection of smaller ones, reducing the computational complexity of an ADMM based method to $\mathcal{O}((m+n_c)n^2 + \sum_{i=1}^{n_c}|\mathcal{C}_k|^3)$ per iteration [51]. A similar complexity reduction applies to rank minimization problems of the form:

$$\begin{aligned}&\min_{\mathbf{X}\succeq 0} \text{rank } (\mathbf{X}) \text{ subject to}\\ &\text{Trace } (\mathbf{A}_i\mathbf{X}) = b_i, i=1,\ldots,m,\ \mathbf{X},\mathbf{A}_i \in \mathbb{R}^{n\times n}\end{aligned} \tag{39}$$

In this case the minimum rank over all possible matrix completions over the cone $\mathbb{S}^n_+(\mathcal{E},?)$ has an explicit expression, given by Dancis' Theorem:

Theorem 2 (Dancis [52]). *Let $\mathcal{G}(\mathcal{V},\mathcal{E})$ be a chordal graph with a set of maximal cliques $\{\mathcal{C}_1,\mathcal{C}_2,\ldots,\mathcal{C}_{n_c}\}$. Then, for any $\mathbf{X}\in\mathbb{S}^n_+(\mathcal{E},?)$ there exist at least one minimum rank PSD completion where rank$(\mathbf{X}) = \max_{1\leq k\leq n_c}$ rank$(\mathbf{E}_{\mathcal{C}_k}\mathbf{X}\mathbf{E}^T_{\mathcal{C}_k})$*

Combining the theorems above leads to the following result:

Corollary 1. *The optimization* (39) *is equivalent to:*

$$\begin{aligned}&\min \sum_k \textit{rank } (\mathbf{E}_{\mathcal{C}_k}\mathbf{X}\mathbf{E}^T_{\mathcal{C}_k}) \textit{ subject to}\\ &\mathbf{E}_{\mathcal{C}_k}\mathbf{X}\mathbf{E}^T_{\mathcal{C}_k} \succeq 0\\ &\textit{Trace } (\mathbf{A}_i\mathbf{X}) = b_i, i=1,\ldots,n_c\end{aligned} \tag{40}$$

Since rank minimization problems are generically NP-hard, a standard convex relaxation is to replace rank by its convex envelope, trace [41]. This substitution leads to a structured SDP that can be solved using the chordal decomposition described above.