HANKEL MATRIX RANK MINIMIZATION WITH APPLICATIONS TO SYSTEM IDENTIFICATION AND REALIZATION [*]

MARYAM FAZEL[†], TING KEI PONG[‡], DEFENG SUN[§], AND PAUL TSENG[¶]

In honor of Professor Paul Tseng, who went missing while on a kayak trip on the Jinsha river, China, on August 13, 2009, for his contributions to the theory and algorithms for large-scale optimization.

Abstract. In this paper, we introduce a flexible optimization framework for nuclear norm minimization of matrices with linear structure, including Hankel, Toeplitz and moment structures, and catalog applications from diverse fields under this framework. We discuss various first-order methods for solving the resulting optimization problem, including alternating direction methods, proximal point algorithm and gradient projection methods. We perform computational experiments to compare these methods on system identification problem and system realization problem. For the system identification problem, the gradient projection method (accelerated by Nesterov's extrapolation techniques) usually outperforms other first-order methods in terms of CPU time on both real and simulated data; while for the system realization problem, the alternating direction method, as applied to a certain primal reformulation, usually outperforms other first-order methods in terms of CPU time.

Key words. Rank minimization, nuclear norm, Hankel matrix, first-order method, system identification, system realization

1. Introduction.

The matrix rank minimization problem, or minimizing the rank of a matrix subject to convex constraints, has recently attracted much renewed interest. This problem arises in many engineering and statistical modeling applications, where notions of order, dimensionality, or complexity of a model can be expressed by the rank of an appropriate matrix. Thus, choosing the "simplest" model that is consistent with observations or data often translates into finding a matrix with the smallest rank subject to convex constraints. Rank minimization is NP-hard in general, and a popular convex heuristic for it minimizes the nuclear norm of the matrix (the sum of the singular values) instead of its rank [17]. The regularized version of this problem can be written as

$$\min_X \quad \frac{1}{2}\|\mathcal{A}(X) - b\|^2 + \mu\|X\|_*, \tag{1.1}$$

where $X \in \mathbb{R}^{m\times n}$ is the optimization variable and $\mathcal{A} : \mathbb{R}^{m\times n} \to \mathbb{R}^p$ is a linear map, $b \in \mathbb{R}^p$, and μ is the tradeoff parameter between the nuclear norm and the least squares fitting error. Problem (1.1) has been widely studied and recently a variety of efficient algorithms have been developed [5, 9, 26, 29, 30, 37, 50]. A special

[*] This work was initiated by the first, the third and the fourth authors during January to May, 2009 when the third author was spending his sabbatical leave with Professor Paul Tseng at the University of Washington, Seattle. With Professor Tseng's missing from August 2009, it seems that this work would never be finished. Fortunately, the second author, Professor Tseng's last PhD student, has kindly agreed to join our force to get this work done. The third author would also like to thank his PhD student Mr Weimin Miao at the National University of Singapore for discussions on the convergence of alternating direction methods.

[†] Department of Electrical Engineering, University of Washington, Seattle, Washington 98195, U.S.A. (mfazel@ee.washington.edu)

[‡] Department of Mathematics, University of Washington, Seattle, Washington 98195, U.S.A. (tkpong@uw.edu)

[§] Department of Mathematics and Risk Management Institute, National University of Singapore, 10 Lower Kent Ridge Road, Singapore. (matsundf@nus.edu.sg)

[¶] Department of Mathematics, University of Washington, Seattle, Washington 98195, U.S.A.

case of this problem is the matrix completion problem [7, 8] which has applications in collaborative filtering and machine learning. In this problem the measurements are simply a subset of the entries of the matrix. The majority of existing work on algorithms for Problem (1.1) has concentrated on this special case.

In this paper, we focus on problems where we need to find a matrix X that, in addition to being low-rank, is required to have a certain *linear structure*, for example, (block-)Hankel, (block-)Toeplitz, or moment structure. Hankel (and Toeplitz) structures arise in dynamical systems problems discussed in Section 1.1, while moment structure comes up in Lasserre relaxations for minimizing polynomials [27]. We consider Problem (1.1), and represent the desired structure by a linear map $X = \mathcal{H}(y)$, where y is our optimization variable. Note that if $\mathcal{H}(y)$ is a moment matrix we need to add the constraint $\mathcal{H}(y) \succeq 0$.

1.1. Motivating applications.

1.1.1. Applications in linear dynamical systems. Linear time-invariant (LTI) systems have a long and successful history in modeling dynamical phenomena in many fields, from engineering to finance. The goal of fitting an LTI model to observed data gives rise to different classes of optimization problems, depending on whether the model is parametric or black-box, given in time or frequency domain, deterministic or stochastic, as well as on the type of data, e.g., input-output or state measurements (see, e.g., [12, 19, 33]). In all these cases, picking the appropriate model order or complexity, and understanding its tradeoff with the fitting or validation errors is crucial. In the problems described in this section, the system order or complexity can be expressed as the rank of a Hankel-type matrix. We discuss some of these problems in more detail in Sections 4 and 5.

Minimal system realization with time-domain constraints. Consider the problem of designing a discrete-time, linear time-invariant (LTI) dynamical system, directly from convex specifications on the system's response in the time domain (see, e.g., [18], [31]). Such a problem arises in designing filters for signal processing and control applications. The objective is to trade-off the *order* of the linear system with how well the specifications are met. A low-order design is desired since in practice, it translates into a system that is easier and cheaper to build and analyze. Typical specifications are desired rise-time, settling-time, slew-rate, and overshoot of the filter's response to a step input signal. These specifications can be expressed as upper and lower bounds on the step response over a fixed time horizon, say n time samples. Equivalently, they can be written in terms of the impulse response, which translate into linear inequality constraints on the entries of a Hankel matrix whose rank corresponds to the system order or McMillan degree (e.g., [49]). Using the nuclear norm heuristic for rank, we get

$$\begin{array}{ll} \min\limits_{y} & \|\mathcal{H}(y)\|_* \\ \text{s.t.} & l_i \leq \sum_{k=1}^{i} y_k \leq b_i, \;\; i = 1, \ldots, n, \end{array} \tag{1.2}$$

where the optimization variable is $y \in \mathbb{R}^{2n}$ with y_i corresponding to the value of the impulse response at time i, l_i and b_i denoting the bounds on the step response given by the specifications, and $\mathcal{H}(y)$ denoting an $n \times n$ Hankel matrix (see [18] for more details).

Minimal partial realization. A related problem in linear system theory is the minimal partial realization problem for multi-input, multi-output systems: given a sequence of matrices H_k, $k = 1, \ldots, n$, find a minimal state space model, described by a 3-tuple of appropriately sized matrices (A, B, C) such that $H_k = CA^{k-1}B$ [49, Chapter 6]. This problem can be viewed as a more general version of the above realization problem which handled a single input and a single output. In this problem, the order of the system (minimal size of a state-space representation) is equal to the rank of a *block-Hankel* matrix consisting of the H_k. We will discuss a related problem in Section 5.

Input-output system identification (system ID). Identifying a linear dynamical system given noisy and/or partial observations of its inputs and outputs, also related to time-series analysis, is a fundamental problem studied in a variety of fields [53,54], including signal processing, control and robotics; see, e.g., [11,33]. We will discuss this problem and a Hankel rank formulation for it in detail in Section 4.

An interesting variation of this problem is a case where the output information is limited to a few time points, for example the case with a switched output briefly mentioned in [46, Section 1]. In this setup, our observations are the system output sampled at a fixed time T, after an input signal is applied from $t = 0$ to $t = T$. We make output measurements for several different input signals, observing $y_i(T) = \sum_{t=0}^{T} a_i(T-t)h(t)$, where the vector a_i is the ith input signal and $h(t)$ denotes the impulse response. Writing this compactly as $y = Ah$ where $A_{ij} = a_i(T-j)$ and $h = [h(0) \dots h(T)]^T$ is the optimization variable, and replacing rank of the Hankel matrix with its nuclear norm, we get a problem of the form (1.1). However, unlike the main system ID problem, in this setup the linear map $\mathcal{A}$ has a nullspace so the fitting error term is not strongly convex; as discussed later, this affects the choice of algorithm for solving the problem computationally.

Stochastic realization. Another fundamental problem in linear system theory is finding a minimal stochastic ARMA (autoregressive moving average) model for a vector random process, given noisy and/or partial estimates of process covariances [12,34]. The minimal order is the rank of a block-Hankel matrix consisting of the exact covariances. This problem is discussed in detail in Section 5.

1.1.2. Other applications.

Shape from moments estimation. Consider a polygonal region P in the complex plane with ordered vertices $z_1, \dots, z_m$. Complex moments of P are defined as $\tau_k := k(k-1)\int_P z^{k-2}dxdy$, $\tau_0 = \tau_1 = 0$, and can be expressed as $\tau_k = \sum_{i=1}^{m} a_i z_i^k$, where m is the number of vertices and a_i are complex constants. The problem of determining P given its complex moments arises in many applications such as computer tomography, where X-ray is used to estimate moments of mass distribution, and geophysical inversion, where the goal is to estimate the shape of a region from external gravitational measurements [16,38]. Note that the *number* of vertices is equal to the rank of the Hankel matrix consisting of the moments [16,23]. In practice, often only noisy or partial measurements of the complex moments are available, and the challenge is to find a polygon with the minimum number of vertices that is consistent with the measurements. Formulating this problem as a rank minimization problem, we propose using the nuclear norm heuristic for rank. This leads to a problem of the form (1.1), where the optimization variable is the vector of complex moments τ.

Moment matrix rank minimization for polynomial optimization. Suppose $p(x)$, $x \in \mathbb{R}^n$ is a polynomial of degree d. Denote the corresponding moment matrix by $M(y)$, where y is the vectors of moments, i.e., y_i corresponds to the ith monomial (see [27,28]). Moment matrices are important in Lasserre's hierarchy of relaxations for polynomial optimization, where a condition on the rank of the moment matrix in successive relaxations in the hierarchy determines whether the relaxation is exact;

see, e.g., [28, Section 5].

In the dual problem of representing a polynomial as a sum of squares of other polynomials [44], the rank of the coefficients matrix equals the minimum number of squared polynomials in the representation, thus the nuclear norm (or trace) penalty helps find simpler representations. Note that in these problems, we also have an additional positive semidefinite constraint on the desired matrix.

Further applications. Another application arises in video inpainting in computer vision, where features extracted from video frames are interpolated by finding a low-rank completion to a Hankel matrix, and help reconstruct the missing frames or occluded parts of a frame [13].

Finally, our problem formulation also gives a relaxation for the problem known as the Structured Total Least Squares problem [10] studied extensively in the controls community; see [35, 36]. Our algorithms are thus applicable to this set of problems as well. A variety of applications are discussed in [36].

1.2. Our contributions. We introduce a flexible optimization framework for nuclear norm minimization of linearly structured matrices, including Hankel, Toeplitz, and moment matrices. We identify and catalog applications from diverse fields, some of which have not been studied from this perspective before; for example, the shape from moments estimation problem. In view of the wide applicability of the model (1.1), it is important to find efficient algorithms for solving it. Along this direction, recently, Liu and Vandenberghe [31] proposed an interior-point method for solving a reformulation of Problem (1.1), where they used the SDP representation for the nuclear norm and exploited the problem structure to efficiently solve the Newton system. They applied their algorithm to the system identification and system realization problems mentioned above. The cost per iteration of the algorithm grows roughly as $\mathcal{O}(p\,qn^2)$, where $y \in \mathbb{R}^n$ and $X = \mathcal{H}(y) \in \mathbb{R}^{p\times q}$.

In this paper, we derive various primal and dual reformulations of Problem (1.1) (with $X = \mathcal{H}(y)$), and propose several first-order methods for solving the reformulations. In particular, we show that the alternating direction method and the proximal point algorithm can be suitably applied to solve reformulations of (1.1). These methods have been widely used in the literature recently for solving (1.1), when no linear structure is imposed on X; see, e.g., [30, 55]. We discuss implementation detail of these methods in Section 3.1 and Section 3.2. For these methods, typically, each iteration involves a singular value decomposition whose cost grows as $\mathcal{O}(p^2q)$ where $X = \mathcal{H}(y) \in \mathbb{R}^{p\times q}$ and $p < q$; see, e.g., [24, Page 254]. Next, in Section 3.3, assuming that $\mathcal{A}^*\mathcal{A}$ is invertible, we show that the (accelerated) gradient projection algorithms can be efficiently applied to solve a dual reformulation of (1.1). In this approach, each iteration also involves a singular value decomposition and there is an explicit upper bound on the number of iterations required to attain a certain accuracy. This solution approach has been considered recently in [39] for solving the system identification problem.

To demonstrate the effectiveness of our algorithms, we apply them to solve the input-output system identification problem and the stochastic system realization problem. For the system identification problem, we consider both simulated data and real data from the DaISy database [11]. Our computational results show that the accelerated gradient projection algorithm usually outperforms other first-order methods for this application in terms of CPU time. We also observe that that these methods significantly outperform the interior point implementation proposed in [31]. For the system realization problem, we consider only simulated data, and our computational results show that the alternating direction method, as applied to a certain primal reformulation of (1.1), usually outperforms other first-order methods in terms of CPU time.

The rest of this paper is organized as follows. In Section 1.3, we introduce notations used in this paper. We then derive primal and dual reformulations of (1.1) in Section 2 and discuss several first-order methods for solving the reformulations in Section 3. In Section 4 and Section 5, we present computational results of our first-order methods for solving the system identification problem and system realization problem, respectively. We give some concluding remarks in Section 6. Finally, we discuss an alternative formulation for modeling the structured matrix rank minimization problem in Appendix I, and provide a convergence proof that covers all versions of proximal alternating direction methods used in this paper in Appendix II.

1.3. Notation. In this paper, $\mathbb{R}^n$ denotes the n-dimensional Euclidean space. For a vector $x \in \mathbb{R}^n$, $\|x\|$ denotes the Euclidean norm of x. The set of all $m \times n$ matrices with real entries is denoted by $\mathbb{R}^{m\times n}$. For any $A \in \mathbb{R}^{m\times n}$, $\|A\|$ denotes the spectral norm of A, $\|A\|_F$ denotes the Fröbenius norm of A, $\|A\|_*$ denotes the nuclear norm of A, which is the sum of all singular values of A, and $\mathrm{vec}(A)$ denotes the column vector formed by stacking columns of A one by one. For two matrices A and B in $\mathbb{R}^{m\times n}$, $A \circ B$ denotes the Hadamard (entry-wise) product of A and B. If a symmetric matrix A is positive semidefinite, we write $A \succeq 0$. For a linear map $\mathcal{A} : \mathbb{R}^{m\times n} \to \mathbb{R}^p$, $\mathcal{A}^*$ denotes the adjoint of $\mathcal{A}$, $\|\mathcal{A}\|$ denotes the spectral norm of $\mathcal{A}$, while $\sigma_{\max}(\mathcal{A})$ and $\sigma_{\min}(\mathcal{A})$ denote the maximum and minimum singular value of $\mathcal{A}$, respectively. Finally, we denote the identity matrix and identity map by I and $\mathcal{I}$ respectively, whose dimensions should be clear from the context.

2. Basic problem formulations.

Consider the following general Hankel matrix rank minimization problem.

$$v := \min_y f(y) := \frac{1}{2}\|\mathcal{A}(y) - b\|^2 + \mu\|\mathcal{H}(y)\|_*, \tag{2.1}$$

where $\mathcal{A} : \mathbb{R}^{m\times n(j+k-1)} \to \mathbb{R}^p$ is a linear map, $b \in \mathbb{R}^p$, $y = \begin{pmatrix} y_0 & \cdots & y_{j+k-2}\end{pmatrix}$ is an $m \times n(j+k-1)$ matrix with each y_i being an $m \times n$ matrix for $i = 1, ..., j+k-2$, and $\mathcal{H}(y) := H_{m,n,j,k}(y)D$ with

$$H_{m,n,j,k}(y) := \begin{pmatrix} y_0 & y_1 & \cdots & y_{k-1} \\ y_1 & y_2 & \cdots & y_k \\ \vdots & \vdots & & \vdots \\ y_{j-1} & y_j & \cdots & y_{j+k-2} \end{pmatrix} \in \mathbb{R}^{mj\times nk},$$

and $D \in \mathbb{R}^{nk\times q}$. We assume without loss of generality that $\sigma_{\max}(D) \le 1$. In this section, we derive primal and dual reformulations of (2.1)

First, using the substitutions $Y = -\mathcal{H}(y)$ and $z = b - \mathcal{A}(y)$, Problem (2.1) can be reformulated as

$$\begin{array}{ll} \min\limits_{Y,z,y} & p(Y,z) := \frac{1}{2}\|z\|^2 + \mu\|Y\|_* \\ \text{s.t.} & Y + \mathcal{H}(y) = 0, \\ & z + \mathcal{A}(y) = b. \end{array} \tag{2.2}$$

From this, we can easily write down the Lagrange dual to (2.2) (and hence, equivalently, to (2.1)) as follows.

$$\begin{aligned}
v &= \min_{Y,z,y}\max_{\gamma,\Lambda}\left\{\frac{1}{2}\|z\|^2+\mu\|Y\|_*-\langle\Lambda,Y+\mathcal{H}(y)\rangle-\langle\gamma,z+\mathcal{A}(y)-b\rangle\right\}\\
&= \max_{\gamma,\Lambda}\min_{Y,z,y}\left\{\frac{1}{2}\|z\|^2+\mu\|Y\|_*-\langle\Lambda,Y+\mathcal{H}(y)\rangle-\langle\gamma,z+\mathcal{A}(y)-b\rangle\right\}\\
&= \max_{\gamma,\Lambda}\min_{Y,z,y}\left\{\frac{1}{2}\|z\|^2-\langle\gamma,z\rangle+\mu\|Y\|_*-\langle\Lambda,Y\rangle-\langle\mathcal{H}^*(\Lambda)+\mathcal{A}^*(\gamma),y\rangle+\langle b,\gamma\rangle\right\}\\
&= \max_{\gamma,\Lambda}\left\{-\frac{1}{2}\|\gamma\|^2+\langle\gamma,b\rangle:\ \mathcal{H}^*(\Lambda)+\mathcal{A}^*(\gamma)=0,\ \Lambda^T\Lambda\preceq\mu^2I\right\},
\end{aligned}$$

where the second equality holds because of strong duality [47, Corollary 28.2.2]. The dual problem can thus be rewritten as the following minimization problem.

$$\begin{array}{ll}
\min\limits_{\gamma,\Lambda} & d(\gamma):=\frac{1}{2}\|\gamma\|^2-b^T\gamma\\
\text{s.t.} & \mathcal{H}^*(\Lambda)+\mathcal{A}^*(\gamma)=0,\\
& \Lambda^T\Lambda\preceq\mu^2I.
\end{array}\tag{2.3}$$

In the special case when $\mathcal{A}^*\mathcal{A}$ is invertible, the objective function of (2.1) is strictly convex and we can derive a reduced dual problem. To this end, notice that

$$\begin{aligned}
v = \min_y\frac{1}{2}\|\mathcal{A}(y)-b\|^2+\mu\|\mathcal{H}(y)\|_* &= \min_y\max_{\Lambda^T\Lambda\preceq\mu^2I}\frac{1}{2}\|\mathcal{A}(y)-b\|^2-\langle\Lambda,\mathcal{H}(y)\rangle\\
= \max_{\Lambda^T\Lambda\preceq\mu^2I}\min_y\frac{1}{2}\|\mathcal{A}(y)-b\|^2-\langle\Lambda,\mathcal{H}(y)\rangle, &
\end{aligned}\tag{2.4}$$

where the third equality holds because of the compactness of the spectral norm ball [47, Corollary 37.3.2]. Writing $\mathcal{R}(\Lambda):=(\mathcal{A}^*\mathcal{A})^{-1}\mathcal{H}^*(\Lambda)$ and $\bar{b}:=(\mathcal{A}^*\mathcal{A})^{-1}\mathcal{A}^*b$, we obtain that

$$v=\max_{\Lambda^T\Lambda\preceq\mu^2I}-\frac{1}{2}(\langle\mathcal{H}^*(\Lambda),\mathcal{R}(\Lambda)\rangle+2\langle\mathcal{H}^*(\Lambda),\bar{b}\rangle+\langle\mathcal{A}^*b,\bar{b}\rangle-\|b\|^2).$$

Hence, the dual problem is equivalent to

$$\begin{array}{ll}
\min\limits_{\Lambda} & d_2(\Lambda):=\frac{1}{2}\langle\mathcal{H}^*(\Lambda),\mathcal{R}(\Lambda)\rangle+\langle\mathcal{H}^*(\Lambda),\bar{b}\rangle+\frac{1}{2}\langle\mathcal{A}^*b,\bar{b}\rangle-\frac{1}{2}\|b\|^2\\
\text{s.t.} & \Lambda^T\Lambda\preceq\mu^2I.
\end{array}\tag{2.5}$$

Before ending this section, we derive an upper bound on the spectral norm of $\mathcal{H}^*$. Notice that $\mathcal{H}^*(\Lambda)=H^*_{m,n,j,k}(\Lambda D^T)$, where for any $W\in\mathbb{R}^{mj\times nk}$

$$\begin{aligned}
H^*_{m,n,j,k}(W) &= H^*_{m,n,j,k}\begin{pmatrix} w_{00} & w_{01} & \cdots & w_{0,k-1}\\ w_{10} & w_{11} & \cdots & w_{1,k-1}\\ \vdots & \vdots & & \vdots\\ w_{j-1,0} & w_{j-1,1} & \cdots & w_{j-1,k-1}\end{pmatrix}\\
&= \begin{pmatrix} w_{00} & w_{01}+w_{10} & w_{02}+w_{11}+w_{20} & \cdots & w_{j-1,k-1}\end{pmatrix}\in\mathbb{R}^{m\times n(j+k-2)}.
\end{aligned}\tag{2.6}$$

It follows from (2.6) that

$$\begin{aligned}\|H^*_{m,n,j,k}(W)\|_F^2 &= \|w_{00}\|_F^2 + \|w_{01}+w_{10}\|_F^2 + \|w_{02}+w_{11}+w_{20}\|_F^2 + \cdots + \|w_{j-1,k-1}\|_F^2\\ &\le \|w_{00}\|_F^2 + 2(\|w_{01}\|_F^2 + \|w_{10}\|_F^2) + \cdots + \|w_{j-1,k-1}\|_F^2 \le \mathbf{r}\|W\|_F^2,\end{aligned}$$

where $\mathbf{r} := \min\{j,k\}$. Combining this estimate with $\sigma_{\max}(D) \le 1$, we obtain that

$$\|\mathcal{H}^*(\Lambda)\|_F^2 \le \mathbf{r}\|\Lambda D^T\|_F^2 \le \mathbf{r}\|\Lambda\|_F^2,$$

and thus the spectral norm of $\mathcal{H}^*$ is less than or equal to $\sqrt{\mathbf{r}}$.

3. Algorithms.

In this section, we discuss several first-order methods for solving (2.2) and (2.3) (and hence (2.1)).

3.1. Alternating direction methods.

In this section, we discuss how the alternating direction method (ADM) can be applied to solve (2.2) and (2.3). To apply the ADM for solving (2.2), we first introduce the augmented Lagrangian function

$$L_\beta(Y,z,y,\gamma,\Lambda) = \frac{1}{2}\|z\|^2 + \mu\|Y\|_* - \langle\Lambda, Y+\mathcal{H}(y)\rangle - \langle\gamma, z+\mathcal{A}(y)-b\rangle + \frac{\beta}{2}\|Y+\mathcal{H}(y)\|_F^2 + \frac{\beta}{2}\|z+\mathcal{A}(y)-b\|^2$$

for each $\beta > 0$. In the classical ADM (see, e.g., [2, Section 3.4.4]), in each iteration, we minimize L_β with respect to (Y,z) and then with respect to y, followed by an update of the multiplier (γ,Λ). While minimizing L_β with respect to (Y,z) admits an easy closed form solution, minimizing L_β with respect to y does not usually have a simple closed form solution due to the complicated quadratic terms. One way to resolve this is to add a proximal term with norm induced by a suitable positive (semi-)definite matrix to "cancel" out the complicated parts. In this approach, we update

$$y^{k+1} = \arg\min_y \left\{ L_\beta(Y^{k+1}, z^{k+1}, y, \gamma^k, \Lambda^k) + \frac{\beta}{2}\|y - y^k\|_{Q_0}^2 \right\},$$

where $\|\cdot\|_{Q_0}$ is the norm induced from the inner product $x^T Q_0 x$,

$$Q_0 := \frac{1}{\sigma}\mathcal{I} - (\mathcal{H}^*\mathcal{H} + \mathcal{A}^*\mathcal{A}) \succ 0 \text{ and } \sigma < \frac{1}{\mathbf{r} + (\sigma_{\max}(\mathcal{A}))^2}$$

so that $\sigma\|\mathcal{H}^*\mathcal{H} + \mathcal{A}^*\mathcal{A}\| \le \sigma(\|\mathcal{H}^*\mathcal{H}\| + \|\mathcal{A}^*\mathcal{A}\|) < 1$. The convergence analysis of this approach has been considered in [25], for example, in the context of variational inequalities [1]. For the sake of completeness, we discuss convergence of this approach in detail in the appendix. We now present our algorithm as follows.

Primal ADM.
Step 0. Input $(y^0, \gamma^0, \Lambda^0)$, $\beta > 0$ and $0 < \sigma < \frac{1}{\mathbf{r} + (\sigma_{\max}(\mathcal{A}))^2}$.
Step 1. Compute the SVD

$$-\mathcal{H}(y^k) + \frac{\Lambda^k}{\beta} = U\Sigma V^T,$$

where U and V have orthogonal columns, Σ is diagonal. Set

$$\begin{aligned}Y^{k+1} &= U\max\left\{\Sigma - \frac{\mu}{\beta}I, 0\right\}V^T, \quad z^{k+1} = \frac{1}{1+\beta}\left(\gamma^k - \beta\mathcal{A}(y^k) + \beta b\right),\\ y^{k+1} &= y^k - \sigma\left(-\frac{1}{\beta}(\mathcal{H}^*\Lambda^k + \mathcal{A}^*\gamma^k) + \mathcal{H}^*(\mathcal{H}(y^k) + Y^{k+1}) + \mathcal{A}^*(\mathcal{A}(y^k) + z^{k+1} - b)\right),\\ \gamma^{k+1} &= \gamma^k - \beta(z^{k+1} + \mathcal{A}(y^{k+1}) - b), \quad \Lambda^{k+1} = \Lambda^k - \beta(Y^{k+1} + \mathcal{H}(y^{k+1})).\end{aligned}$$

[1] The main motivation of introducing the proximal terms in [25] is to weaken the imposed convergent conditions rather than for the sake of cancelation, as was more recently explained in [56].

Step 2. If a termination criterion is not met, go to Step 1.

The ADM can also be applied to solve the dual problem (2.3). In this approach, we make use of the following augmented Lagrangian function

$$l_\beta(\gamma,\Lambda,y) = \frac{1}{2}\|\gamma\|^2 - b^T\gamma + \langle y, \mathcal{H}^*(\Lambda)+\mathcal{A}^*(\gamma)\rangle + \frac{\beta}{2}\|\mathcal{H}^*(\Lambda)+\mathcal{A}^*(\gamma)\|_F^2,$$

for each $\beta > 0$. Notice that the minimizer of l_β with respect to γ is usually not easy to find and the minimizer with respect to Λ does not always admit a simple closed form solution. Thus, as before, we add suitable proximal terms to cancel out complicated terms. More precisely, we update

$$\begin{aligned}
\gamma^{k+1} &:= \operatorname*{arg\,min}_{\gamma}\left\{ l_\beta(\gamma,\Lambda^k,y^k) + \frac{\beta}{2}\|\gamma-\gamma^k\|_{Q_1}^2\right\},\\
\Lambda^{k+1} &:= \operatorname*{arg\,min}_{\Lambda^T\Lambda\preceq\mu^2 I}\left\{ l_\beta(\gamma^{k+1},\Lambda,y^k) + \frac{\beta}{2}\|\Lambda-\Lambda^k\|_{Q_2}^2\right\},
\end{aligned}$$

where

$$Q_1 := \frac{1}{\sigma_1}\mathcal{I} - \mathcal{A}\mathcal{A}^* \succ 0 \text{ and } \sigma_1 < \frac{1}{(\sigma_{\max}(\mathcal{A}))^2}, \quad Q_2 := \frac{1}{\sigma_2}\mathcal{I} - \mathcal{H}\mathcal{H}^* \succ 0 \text{ and } \sigma_2 < \frac{1}{\mathbf{r}},$$

so that $\sigma_1\|\mathcal{A}\mathcal{A}^*\| < 1$ and $\sigma_2\|\mathcal{H}\mathcal{H}^*\| < 1$. The algorithm is described as follows.

Dual ADM.

Step 0. Input (y^0,γ^0,Λ^0), $\beta > 0$, $0 < \sigma_1 < \frac{1}{(\sigma_{\max}(\mathcal{A}))^2}$ and $0 < \sigma_2 < \frac{1}{\mathbf{r}}$.

Step 1. Set

$$\gamma^{k+1} = \frac{\sigma_1}{\sigma_1+\beta}\left(b + \beta\frac{\gamma^k}{\sigma_1} - \beta\left(\frac{\mathcal{A}(y^k)}{\beta} + \mathcal{A}(\mathcal{H}^*(\Lambda^k)+\mathcal{A}^*(\gamma^k))\right)\right).$$

Compute the SVD

$$\Lambda^k - \sigma_2\left(\frac{1}{\beta}\mathcal{H}(y^k) + \mathcal{H}(\mathcal{H}^*(\Lambda^k)+\mathcal{A}^*(\gamma^{k+1}))\right) = U\Sigma V^T,$$

where U and V have orthogonal columns, Σ is diagonal. Set

$$\begin{aligned}
\Lambda^{k+1} &= U\min\{\Sigma,\mu I\}V^T,\\
y^{k+1} &= y^k + \beta(\mathcal{H}^*(\Lambda^{k+1})+\mathcal{A}^*(\gamma^{k+1})).
\end{aligned}$$

Step 2. If a termination criterion is not met, go to Step 1.

From Theorem 8.1 in the appendix, for the sequence $\{(Y^k,z^k,y^k,\gamma^k,\Lambda^k)\}$ generated from Primal ADM, $\{(Y^k,z^k,y^k)\}$ converges to a solution of the problem (2.2) while $\{(\gamma^k,\Lambda^k)\}$ converges to a solution of the problem (2.3). Similarly, for the sequence $\{(y^k,\gamma^k,\Lambda^k)\}$ generated from Dual ADM, $\{y^k\}$ converges to a solution of (2.1) and $\{(\gamma^k,\Lambda^k)\}$ converges to a solution of (2.3).

3.2. Dual proximal point algorithm. In this section, we discuss how the proximal point algorithm (PPA) can be applied to solve (2.3). We first introduce the following Lagrangian function l for (2.3):

$$l(\gamma,\Lambda,y) = \frac{1}{2}\|\gamma\|^2 - b^T\gamma + \langle y, \mathcal{H}^*(\Lambda)+\mathcal{A}^*(\gamma)\rangle.$$

The essential objective function in (2.3) can then be represented as

$$g(\gamma,\Lambda) := \sup_y \, l(\gamma,\Lambda,y) = \begin{cases} \frac{1}{2}\|\gamma\|^2 - b^T\gamma & \text{if } (\gamma,\Lambda) \in \mathcal{F}_D, \\ +\infty & \text{if } (\gamma,\Lambda) \notin \mathcal{F}_D, \end{cases} \tag{3.1}$$

where $\mathcal{F}_D := \{(\gamma,\Lambda) : \mathcal{H}^*(\Lambda) + \mathcal{A}^*(\gamma) = 0, \Lambda^T\Lambda \preceq \mu^2 I\}$ is the feasible set of (2.3).

Fix $\lambda > 0$. For any (γ,Λ), define the Moreau-Yosida regularization of g at (γ,Λ) associated with λ by

$$G_\lambda(\gamma,\Lambda) = \min_{\substack{\Gamma^T\Gamma\preceq\mu^2 I \\ \alpha}} g(\alpha,\Gamma) + \frac{1}{2\lambda}\|(\alpha,\Gamma) - (\gamma,\Lambda)\|_F^2 .$$

Using this definition and (3.1), we obtain that

$$\begin{aligned} G_\lambda(\gamma,\Lambda) = & \min_{\substack{\Gamma^T\Gamma\preceq\mu^2 I \\ \alpha}} \sup_y l(\alpha,\Gamma,y) + \frac{1}{2\lambda}\|(\alpha,\Gamma) - (\gamma,\Lambda)\|_F^2 \\ = & \sup_y \min_{\substack{\Gamma^T\Gamma\preceq\mu^2 I \\ \alpha}} \frac{1}{2}\|\alpha\|^2 - b^T\alpha + \langle y, \mathcal{H}^*(\Gamma) + \mathcal{A}^*(\alpha)\rangle + \frac{1}{2\lambda}\|(\alpha,\Gamma) - (\gamma,\Lambda)\|_F^2 \\ = & \sup_y \left\{ \min_{\Gamma^T\Gamma\preceq\mu^2 I} \left\{ \langle \mathcal{H}(y), \Gamma\rangle + \frac{1}{2\lambda}\|\Lambda - \Gamma\|_F^2 \right\} \right. \\ & \left. + \min_\alpha \left\{ \frac{1}{2}\|\alpha\|^2 - b^T\alpha + \langle \mathcal{A}(y), \alpha\rangle + \frac{1}{2\lambda}\|\alpha - \gamma\|^2 \right\} \right\}, \end{aligned}$$

where the second equality holds due to the Slater condition [47, Corollary 28.2.2]. Furthermore, it is not hard to show that

$$\min_{\Gamma^T\Gamma\preceq\mu^2 I} \left\{ \langle \mathcal{H}(y), \Gamma\rangle + \frac{1}{2\lambda}\|\Lambda - \Gamma\|_F^2 \right\} = \langle \mathcal{H}(y), \Lambda\rangle - \frac{\lambda}{2}\|\mathcal{H}(y)\|_F^2 + \frac{1}{2\lambda}\|\mathcal{P}_\mu(\Lambda - \lambda\mathcal{H}(y))\|^2,$$

where $\mathcal{P}_\mu(W)$ is the unique optimal solution to the following convex optimization problem:

$$\min_Z \|Z\|_* + \frac{1}{2\mu}\|Z - W\|_F^2 .$$

Moreover, we have

$$\min_\alpha \left\{ \frac{1}{2}\|\alpha\|^2 - b^T\alpha + \langle \mathcal{A}(y), \alpha\rangle + \frac{1}{2\lambda}\|\alpha - \gamma\|^2 \right\} = \frac{1}{2\lambda}\|\gamma\|^2 - \frac{\lambda}{2(\lambda+1)} \left\| \mathcal{A}(y) - b - \frac{\gamma}{\lambda} \right\|^2 .$$

Thus, $G_\lambda(\gamma,\Lambda) = \sup_y \Theta_\lambda(y;(\gamma,\Lambda))$, where

$$\Theta_\lambda(y;(\gamma,\Lambda)) := \langle \mathcal{H}(y), \Lambda\rangle - \frac{\lambda}{2}\|\mathcal{H}(y)\|_F^2 + \frac{1}{2\lambda}\|\mathcal{P}_\mu(\Lambda - \lambda\mathcal{H}(y))\|_F^2 + \frac{1}{2\lambda}\|\gamma\|^2 - \frac{\lambda}{2(\lambda+1)} \left\| \mathcal{A}(y) - b - \frac{\gamma}{\lambda} \right\|^2 .$$

Recall from the Moreau-Yosida regularization theory that $\mathcal{P}_\mu(\cdot)$ is globally Lipschitz continuous with modulus 1 and that $\|\mathcal{P}_\mu(\cdot)\|_F^2$ is continuously differentiable with $\nabla(\|\mathcal{P}_\mu(Y)\|_F^2) = 2\mathcal{P}_\mu(Y)$. Hence, $\Theta_\lambda(\cdot;(\gamma,\Lambda))$ is a continuously differentiable concave function in y with

$$\nabla_y \Theta_\lambda(y;(\gamma,\Lambda)) = \mathcal{H}^*(\Lambda - \lambda\mathcal{H}(y)) - \mathcal{H}^*\mathcal{P}_\mu(\Lambda - \lambda\mathcal{H}(y)) - \frac{\lambda}{\lambda+1}\mathcal{A}^*(\mathcal{A}(y) - b - \frac{\gamma}{\lambda}).$$

In addition, we have that

$$\begin{aligned}
&\|\nabla_y\Theta_\lambda(y';(\gamma,\Lambda))-\nabla_y\Theta_\lambda(y;(\gamma,\Lambda))\|_F\\
&\le\left(\lambda\|\mathcal{H}^*\mathcal{H}\|+\frac{\lambda}{\lambda+1}\|\mathcal{A}^*\mathcal{A}\|\right)\|y'-y\|_F+\|\mathcal{H}^*\mathcal{P}_\mu(\Lambda-\lambda\mathcal{H}(y'))-\mathcal{H}^*\mathcal{P}_\mu(\Lambda-\lambda\mathcal{H}(y))\|_F\\
&\le\left(\lambda\mathbf{r}+\frac{\lambda}{\lambda+1}(\sigma_{\max}(\mathcal{A}))^2\right)\|y'-y\|_F+\lambda\mathbf{r}\|y'-y\|_F,
\end{aligned}$$

which implies that $\nabla_y\Theta_\lambda(\cdot;(\gamma,\Lambda))$ is Lipschitz continuous with Lipschitz modulus

$$2\lambda\mathbf{r}+\frac{\lambda}{\lambda+1}(\sigma_{\max}(\mathcal{A}))^2. \tag{3.2}$$

We are now ready to describe the PPA for solving the dual problem (2.3).

```
Dual PPA.
Step 0. Input $(y^0,\gamma^0,\Lambda^0)$ and $\lambda_0>0$.
Step 1. (Find an approximate maximizer $y^{k+1}\approx\arg\max\Theta_{\lambda_k}(y;(\gamma^k,\Lambda^k))$.)
        Input $u^0:=y^k$. Let $s_k, intol_k>0$, $1>t,\sigma>0$.
        While $\|\nabla_y\Theta_{\lambda_k}(u^l;(\gamma^k,\Lambda^k))\|>intol_k$ do
          (a) Let $\bar{s}_l$ be the largest element of $\{s_k,ts_k,t^2s_k,\cdots\}$ satisfying
                $\Theta_{\lambda_k}(u[s];(\gamma^k,\Lambda^k))>\Theta_{\lambda_k}(u^l;(\gamma^k,\Lambda^k))+\sigma s\|\nabla_y\Theta_{\lambda_k}(u^l;(\gamma^k,\Lambda^k))\|_F^2$,
              where $u[s]=u^l+s\nabla_y\Theta_{\lambda_k}(u^l;(\gamma^k,\Lambda^k))$.
          (b) Set $u^{l+1}\leftarrow u[\bar{s}_l]$, $l\leftarrow l+1$.
        End (while)
        Set $y^{k+1}\leftarrow u^{l+1}$.
Step 2. Compute the SVD
                $\Lambda^k-\lambda_k\mathcal{H}(y^{k+1})=U\Sigma V^T$.
        Set
                $\gamma^{k+1}=-\frac{\lambda_k}{\lambda_k+1}\left(\mathcal{A}(y^{k+1})-b-\frac{\gamma^k}{\lambda_k}\right)$ and $\Lambda^{k+1}=U\min(\Sigma,\mu I)V^T$.
Step 3. If a termination criterion is not met, update $\lambda_k$. Go to Step 1.
```

3.3. Dual gradient projection methods. In this section we assume that $\mathcal{A}^*\mathcal{A}$ is invertible. Recall that in this case, the dual of (2.1) is given by (2.5). Moreover, we have

$$\begin{aligned}
\|\nabla d_2(\Lambda_1)-\nabla d_2(\Lambda_2)\|_F^2&=\|\mathcal{H}((\mathcal{A}^*\mathcal{A})^{-1}\mathcal{H}^*(\Lambda_1-\Lambda_2))\|_F^2\le\mathbf{r}\|(\mathcal{A}^*\mathcal{A})^{-1}\mathcal{H}^*(\Lambda_1-\Lambda_2)\|_F^2\\
&\le\frac{\mathbf{r}}{(\sigma_{\min}(\mathcal{A}^*\mathcal{A}))^2}\|\mathcal{H}^*(\Lambda_1-\Lambda_2)\|_F^2\le\left(\frac{\mathbf{r}}{\sigma_{\min}(\mathcal{A}^*\mathcal{A})}\right)^2\|\Lambda_1-\Lambda_2\|_F^2.
\end{aligned}$$

This shows that the gradient of d_2 is Lipschitz continuous with Lipschitz constant

$$L_{\mathrm{D}}:=\frac{\mathbf{r}}{\sigma_{\min}(\mathcal{A}^*\mathcal{A})}.$$

Since the projection onto the feasible set of (2.5) is simple, we can apply the gradient projection (GP) methods to solve (2.5). One simple version is described as follows.

Dual GP.

Step 0. Input Λ^0 such that ${\Lambda^0}^T\Lambda^0 \preceq \mu^2 I$ and $L > \frac{L_{\rm D}}{2}$.

Step 1. Compute the SVD

$$\Lambda^k - \frac{1}{L}\nabla d_2(\Lambda^k) = U\Sigma V^T,$$

where U and V have orthogonal columns, Σ is diagonal. Set

$$\Lambda^{k+1} = U\min\{\Sigma, \mu I\}V^T.$$

Step 2. If a termination criterion is not met, go to Step 1.

The iterate generated by the Dual GP satisfies

$$d_2(\Lambda^k) - v = \mathcal{O}(\frac{L}{k});$$

see, e.g., [52, Theorem 1]. Hence, for faster convergence, a smaller L is favored. Also, note that if y^* and Λ^* are solutions to (2.1) and (2.5) respectively, then we can see from (2.4) that

$$y^* = \mathcal{R}(\Lambda^*) + \bar{b},$$

where $\mathcal{R}(\Lambda) := (\mathcal{A}^*\mathcal{A})^{-1}\mathcal{H}^*(\Lambda)$ and $\bar{b} := (\mathcal{A}^*\mathcal{A})^{-1}\mathcal{A}^*b$. Hence, the sequence

$$y^k := \mathcal{R}(\Lambda^k) + \bar{b}$$

converges to a solution to (2.1), which can be used to check for termination; see Section 4.1.

The Dual GP can be accelerated using Nesterov's extrapolation techniques (see, e.g., [40–43, 52]). This method has also been used in [45, 50] for nuclear norm related problems. The method is described below.

Dual AGP.

Step 0. Input Λ^0 such that ${\Lambda^0}^T\Lambda^0 \preceq \mu^2 I$ and $L \geq L_{\rm D}$. Initialize $\Lambda^{-1} = \Lambda^0$ and $\theta_{-1} = \theta_0 = 1$. Go to Step 1.

Step 1. Set

$$\Psi^k = \Lambda^k + \left(\frac{\theta_k}{\theta_{k-1}} - \theta_k\right)(\Lambda^k - \Lambda^{k-1}).$$

Compute the SVD

$$\Psi^k - \frac{1}{L}\nabla d_2(\Psi^k) = U\Sigma V^T,$$

where U and V have orthogonal columns, Σ is diagonal. Update

$$\Lambda^{k+1} = U\min\{\Sigma, \mu I\}V^T, \quad \theta_{k+1} = \frac{\sqrt{{\theta_k}^4 + 4{\theta_k}^2} - {\theta_k}^2}{2}.$$

Step 2. If a termination criterion is not met, go to Step 1.

The sequence generated from the Dual AGP satisfies

$$d_2(\Lambda^k) - v = \mathcal{O}(\frac{L}{k^2});$$

see, e.g., [52, Theorem 1]. Hence, for faster convergence, a smaller L is favored. To generate a suitable primal variable at each iteration, instead of setting $y^k = \mathcal{R}(\Lambda^k) + \bar{b}$ as above, we can also initialize y^0 to be the zero matrix and update the sequence in Step 1 according to

$$y^{k+1} = (1 - \theta_k) y^k + \theta_k (\mathcal{R}(\Psi^k) + \bar{b}).$$

For this choice of y^k, it can be shown, following the proof of [51, Corollary 2], that

$$0 \leq f(y^{k+1}) + d_2(\Lambda^{k+1}) = \mathcal{O}(L_{\mathrm{D}} \theta_k^2).$$

Since $\theta_k \leq \frac{2}{k+2}$, the duality gap falls below a given tolerance $tol > 0$ after at most $\mathcal{O}(\sqrt{\frac{L_{\mathrm{D}}}{tol}}\,)$ iterations. In our implementation of Dual AGP, we use both choices of $\{y^k\}$.

4. System identification. In this section, we consider the problem of identifying a linear dynamical system from observations of its inputs and outputs. Given a sequence of inputs u_t and measured (noisy) outputs $\tilde{y}_t$, $t = 1, \ldots, N$, the goal is to find a discrete-time, linear time-invariant state space model,

$$\begin{aligned} x_{t+1} &= A x_t + B u_t, \\ y_t &= C x_t + D u_t, \end{aligned} \tag{4.1}$$

that satisfies $y_t \approx \tilde{y}_t$ and is low-order (i.e., corresponds to a low-dimensional state vector x_t). To determine the model, we need to find the A, B, C, D matrices, the initial state x_0, and the model order r. As described in [31, Eq. (23)] (see also [32]), under reasonable assumptions, the minimal model order is equal to the rank of the matrix $H_{m,1,r+1,N+1-r}(y) U^\perp$, which we relax to the nuclear norm. Thus, the tradeoff between the fitting error and the model order is captured by the optimization problem,

$$\min_y \frac{1}{2} \|y - \tilde{y}\|_F^2 + \mu \|H_{m,1,r+1,N+1-r}(y) U^\perp\|_*, \tag{4.2}$$

where $\tilde{y} \in \mathbb{R}^{m \times (N+1)}$, $N \geq 1$, $r \geq 0$, $\mu > 0$,

$$H_{m,1,r+1,N+1-r}(y) = \begin{pmatrix} y_0 & y_1 & \cdots & y_{N-r} \\ y_1 & y_2 & \cdots & y_{N-r+1} \\ \vdots & \vdots & & \vdots \\ y_r & y_{r+1} & \cdots & y_N \end{pmatrix} \in \mathbb{R}^{m(r+1) \times (N+1-r)},$$

and $U^\perp \in \mathbb{R}^{(N+1-r) \times q}$ is a matrix whose columns form an orthogonal basis of the null space of $H_{m,1,r+1,N+1-r}(u)$ (so $U^\perp$ has $N - r + 1$ rows), where u is the input to the system (see [31] for further details). Hence in particular ${U^\perp}^T U^\perp = I$. This problem corresponds to (2.1) with $D = U^\perp$[2], $\mathcal{A}(y) = \mathrm{vec}(y)$ and $b = \mathrm{vec}(\tilde{y})$. Thus

$$\sigma_{\max}(\mathcal{A}) = \sigma_{\max}(D) = 1, \qquad \mathcal{H}^*(\Lambda) = H^*_{m,1,r+1,N+1-r}(\Lambda (U^\perp)^T).$$

[2] notice that this D is different from that in (4.1).

4.1. Computational results. In this section, we compare different algorithms for solving (4.2) on random and real data. Specifically, we consider Primal and Dual ADM, Dual PPA, Dual GP, and Dual AGP. Moreover, taking into account the fact that $\mathcal{A} = \mathcal{I}$, we also consider a variant of Primal ADM (referred to as Primal ADM2) by considering the following augmented Lagrangian function

$$\ell_\beta(Y, y, \Lambda) = \frac{1}{2}\|y - \tilde{y}\|_F^2 + \mu\|Y\|_* - \langle \Lambda, Y + \mathcal{H}(y)\rangle + \frac{\beta}{2}\|Y + \mathcal{H}(y)\|_F^2,$$

and replacing the minimization with respect to y by

$$y^{k+1} := \arg\min_y \left\{ \ell_\beta(Y^{k+1}, y, \Lambda^k) + \frac{\beta}{2}\|y - y^k\|_{Q_3}^2 \right\},$$

where

$$Q_3 := \frac{1}{\sigma}\mathcal{I} - \mathcal{H}^*\mathcal{H} \succ 0 \text{ and } \sigma < \frac{1}{\mathbf{r}},$$

so that $\sigma\|\mathcal{H}^*\mathcal{H}\| < 1$; as well as a variant of Dual ADM with $Q_1 = 0$ (referred to as Dual ADM2). The convergence of these two variants follows from Theorem 8.1.

We initialize all algorithms except Dual PPA at the origin, where the latter algorithm is initialized at an approximate solution obtained from Dual ADM [3]. We terminate the algorithms by checking relative duality gap with $tol = 1e - 4$, i.e.,

$$\frac{f(y^k) + d(-\mathcal{H}^*(\mathcal{P}(\Lambda^k)))}{\max\{1, |d(-\mathcal{H}^*(\mathcal{P}(\Lambda^k)))|\}} < 1e - 4 \quad \text{or} \quad \frac{f(y^k) + d_2(\Lambda^k)}{\max\{1, |d_2(\Lambda^k)|\}} < 1e - 4,$$

with the $\{(y^k, \Lambda^k)\}$ defined in each of Sections 3.1, 3.2 and 3.3, and $\mathcal{P}(\Lambda^k)$ is the projection of Λ^k onto the spectral norm ball with radius μ. The termination criterion is checked every 10 iterations except for Dual PPA, where we do this every (outer) iteration, as the relative duality gap is required for updating λ_k and $intol_k$. [4] We also early terminate the algorithms if the change in Fröbenius norm of successive iterates is small ($< 1e - 8$ for each variable) or the maximum number of iteration hits 2000. We set $\beta = \frac{\mu\mathbf{r}}{2\sigma_{\max}(\tilde{y})}$ and $\sigma = \frac{0.95}{\mathbf{r}+1}$ for Primal ADM. We use the same β for Primal ADM2 and set $\sigma = \frac{0.95}{\mathbf{r}}$. Furthermore, we set $\beta = \frac{\sigma_{\max}(\tilde{y})}{16\mu\mathbf{r}}$, $\sigma_1 = 0.95$ and $\sigma_2 = \frac{0.95}{\mathbf{r}}$ for Dual ADM and use the same β and σ_2 for Dual ADM2. We take $L = \frac{L_D}{1.95}$ for Dual GP and $L = L_D$ for Dual AGP. All codes are written in Matlab and run on a Dell POWEREDGE 1950 with Matlab 7.8 and Debian 5.0.6.

4.1.1. Random data. We generate randomly a matrix $u = \begin{pmatrix} u_0 & \cdots & u_N \end{pmatrix} \in \mathbb{R}^{p\times(N+1)}$ with Gaussian entries, and let $\bar{r}$ be the true order of the system. We then generate matrices $A \in \mathbb{R}^{\bar{r}\times\bar{r}}$, $B \in \mathbb{R}^{\bar{r}\times p}$, $C \in \mathbb{R}^{m\times\bar{r}}$ and $D \in \mathbb{R}^{m\times p}$ with i.i.d. Gaussian entries, and normalize them to have spectral norm 1. We also generate

[3] We terminate Dual ADM by checking relative duality gap with $tol = 5e - 3$, checked every 5 iterations. We also early terminate the algorithm if the change in Fröbenius norm of successive iterates is small ($< 1e - 8$ for each variable) or the maximum number of iteration hits 2000.

[4] For Dual PPA, we set $t = 0.3$, $\sigma = 1e - 4$, $s_0 = \frac{1.95}{L}$, with L defined as in (3.2). For each fixed k and any $l \geq 1$, we set $s_l = 1.11s_{l-1}$ if $\bar{s}_{l-1} = s_k$, and $s_l = s_{l-1}$ otherwise. The inner iterations are terminated when l hits 1000. The parameters λ_k are initialized at $\lambda_0 = 1$ and the update follows the rule that $\lambda_{k+1} = 2\lambda_k$ if $\text{gap}_k > 0.95\text{gap}_{k-1}$, where gap_k denotes the relative duality gap in the kth outer iteration. The $intol_k$ decreases from 0.04 based on the value and the decrease of gap_k, and is bounded below by $1e - 3$.

a vector $x_0 \in \mathbb{R}^{\bar{r}}$, again with Gaussian entries. The output $\bar{y} = \begin{pmatrix} \bar{y}_0 & \cdots & \bar{y}_N \end{pmatrix} \in \mathbb{R}^{m\times(N+1)}$ is then generated using a state-space model: for each $t = 0, ..., N$,

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t \\ \bar{y}_t &= Cx_t + Du_t. \end{aligned}$$

To model the measurement noise, we add noise to the output $\bar{y}$ to get $\tilde{y} = \bar{y} + \sigma\epsilon$, where ϵ has Gaussian entries with variance 1 and $\sigma > 0$. Finally, $U^\perp$ is a matrix whose columns form an orthonormal basis of the nullspace of $H_{p,1,2\bar{r}+2,N-2\bar{r}}(u)$. Note that in theory, we require the r used in determining the size of the Hankel matrix to be larger than the true order of the system. However in practice, we often don't know the true system order, and only have a guess or estimate for it. Therefore, when we set the size of the Hankel matrix in our problem, as a rule of thumb, we use twice the estimated order.

In the tests below, we consider $p = 5$, $m = 5, 10$, $\bar{r} = 10, 20$ and $N + 1 = 2000, 4000$. We pick $\sigma = 5e - 2$, which corresponds roughly to a 5% noise. The statistics of the test problems used are reported in Table 4.1. We run our algorithms for $\mu = 1e - 2, 1e - 1, 1, 10$. We observed that Primal ADM2 usually outperforms Primal ADM, while the two Dual ADMs perform almost identically. Thus, we do not report the performances of Primal ADM and Dual ADM2.

Our computational results are reported in Table 4.2, where **iter** stands for the number of iterations, **cpu** is the CPU time taken and **obj** is the primal objective value at termination corresponding to each algorithm. The words "Primal" and "Dual" are abbreviated as "P." and "D." respectively. For Dual PPA, the CPU time and number of iterations for Dual ADM used for initialization are in parenthesis, and the CPU time not in parenthesis refers to the total runtime. The word "max" denotes the maximum number of iterations. The fastest algorithm(s) in each instance is highlighted in bold. We see that the gradient projection algorithms usually work best, with Dual AGP usually faster and more robust (i.e., solving all instances within 2000 iterations) than Dual GP.

Table 4.1
Parameters for the randomly generated test problems.

P	$N+1$	$\bar{r}$	m	q
1	2000	10	5	1869
2	2000	10	10	1869
3	2000	20	5	1749
4	2000	20	10	1749
5	4000	10	5	3869
6	4000	10	10	3869
7	4000	20	5	3749
8	4000	20	10	3749

4.1.2. Real data from DaISy database. In this section, we consider eight benchmark problems from DaISy (Database for the Identification of Systems) [11]. A brief description of the data is given in the Table 4.3. Each data set is given in form of a y (output) and a u (input). We take the first 25% of the inputs and outputs for testing purposes. We set $\bar{r} = 20$ and generate the matrix $U^\perp$ so that the column space forms an orthonormal basis of the nullspace of $H_{p,1,2\bar{r}+2,N-2\bar{r}}(u)$,

Table 4.2
Computational results for problems from Table 4.1.

		D. GP	D. AGP	D. PPA	P. ADM2	D. ADM
P	μ	iter/cpu/obj	iter/cpu/obj	iter/cpu/obj	iter/cpu/obj	iter/cpu/obj
1	0.01	**6/ 1.4/3.76e+0**	9/ 2.2/3.76e+0	12(5)/ 7.7(1.4)/3.76e+0	10/ 2.7/3.76e+0	10/ 2.7/3.76e+0
	0.10	**10/ 2.2/2.87e+1**	10/ 2.4/2.87e+1	1(10)/ 5.3(2.7)/2.87e+1	30/ 8.2/2.87e+1	20/ 5.2/2.87e+1
	1.00	150/ 31.8/1.70e+2	70/ 16.0/1.70e+2	5(30)/ 32.5(7.9)/1.70e+2	50/ 13.6/1.70e+2	**50/ 12.9/1.70e+2**
	10.00	1310/272.4/1.01e+3	**240/ 54.1/1.01e+3**	2(205)/ 77.8(52.8)/1.01e+3	250/ 67.0/1.01e+3	340/ 86.2/1.01e+3
2	0.01	**7/ 3.8/5.19e+0**	10/ 5.8/5.19e+0	7(5)/ 12.2(3.2)/5.19e+0	10/ 6.5/5.19e+0	10/ 6.3/5.19e+0
	0.10	**20/ 10.4/3.29e+1**	20/ 11.2/3.29e+1	2(10)/ 16.4(6.3)/3.29e+1	50/ 32.1/3.29e+1	40/ 24.1/3.29e+1
	1.00	210/108.0/1.05e+2	90/ 49.9/1.05e+2	6(40)/ 71.9(24.2)/1.05e+2	70/ 44.3/1.05e+2	**70/ 41.4/1.05e+2**
	10.00	900/456.1/4.26e+2	420/227.9/4.26e+2	6(270)/226.2(159.3)/4.26e+2	**310/193.2/4.26e+2**	430/249.2/4.26e+2
3	0.01	9/ 4.2/4.57e+0	10/ 5.0/4.57e+0	**1(5)/ 3.9(2.8)/4.57e+0**	10/ 5.6/4.57e+0	10/ 5.4/4.57e+0
	0.10	380/169.2/2.06e+1	**70/ 33.5/2.06e+1**	23(20)/ 51.1(10.6)/2.06e+1	390/215.9/2.06e+1	540/277.0/2.06e+1
	1.00	730/319.8/8.60e+1	180/ 84.6/8.60e+1	10(70)/121.3(36.1)/8.60e+1	140/ 76.3/8.60e+1	**120/ 61.1/8.60e+1**
	10.00	max/858.3/3.70e+2	**420/193.7/3.70e+2**	7(525)/370.9(265.6)/3.70e+2	1130/607.0/3.70e+2	820/409.6/3.70e+2
4	0.01	9/ 16.4/8.69e+0	10/ 19.2/8.69e+0	**1(5)/ 14.4(10.3)/8.69e+0**	10/ 20.2/8.69e+0	10/ 19.9/8.69e+0
	0.10	460/803.4/3.65e+1	**80/145.3/3.65e+1**	24(20)/211.5(39.1)/3.65e+1	470/956.6/3.65e+1	650/1197.9/3.65e+1
	1.00	400/678.5/1.33e+2	170/299.7/1.33e+2	3(65)/448.5(120.2)/1.33e+2	140/275.0/1.33e+2	**110/198.6/1.33e+2**
	10.00	max/3291.2/6.19e+2	**470/818.8/6.19e+2**	2(485)/1237.8(875.7)/6.19e+2	1070/2069.0/6.19e+2	770/1359.6/6.19e+2
5	0.01	**6/ 3.9/4.67e+0**	8/ 5.5/4.67e+0	8(5)/ 17.5(4.2)/4.67e+0	10/ 8.2/4.67e+0	10/ 7.9/4.67e+0
	0.10	**10/ 6.3/3.73e+1**	10/ 6.7/3.73e+1	3(5)/ 23.6(4.1)/3.73e+1	10/ 8.2/3.73e+1	10/ 7.9/3.73e+1
	1.00	150/ 87.9/1.75e+2	**60/ 38.3/1.75e+2**	7(25)/ 89.6(19.4)/1.75e+2	60/ 48.3/1.75e+2	**50/ 38.2/1.75e+2**
	10.00	1070/633.7/9.39e+2	270/172.7/9.39e+2	2(175)/204.7(133.0)/9.39e+2	**210/168.2/9.39e+2**	280/211.4/9.39e+2
6	0.01	**6/ 8.9/7.57e+0**	9/ 14.0/7.57e+0	3(5)/ 28.7(9.1)/7.57e+0	10/ 18.2/7.57e+0	10/ 17.4/7.57e+0
	0.10	**10/ 14.5/5.56e+1**	10/ 15.6/5.56e+1	3(5)/ 55.5(9.1)/5.56e+1	20/ 36.3/5.56e+1	10/ 17.4/5.56e+1
	1.00	200/279.3/1.63e+2	80/120.6/1.63e+2	10(30)/193.9(51.4)/1.63e+2	80/144.5/1.63e+2	**60/101.0/1.63e+2**
	10.00	470/649.5/7.75e+2	**250/376.0/7.75e+2**	2(185)/477.7(314.1)/7.75e+2	**210/378.1/7.75e+2**	290/486.3/7.75e+2
7	0.01	**7/ 9.1/6.91e+0**	10/ 13.9/6.91e+0	6(5)/ 30.8(8.2)/6.91e+0	10/ 16.3/6.91e+0	10/ 15.6/6.91e+0
	0.10	170/206.9/3.66e+1	**50/ 65.8/3.66e+1**	11(20)/115.8(30.7)/3.66e+1	250/392.8/3.66e+1	350/512.5/3.66e+1
	1.00	360/440.3/1.32e+2	130/171.7/1.32e+2	14(50)/375.0(75.4)/1.32e+2	100/159.4/1.32e+2	**90/133.2/1.32e+2**
	10.00	980/1192.2/6.09e+2	**410/538.5/6.09e+2**	2(375)/820.8(559.0)/6.09e+2	790/1247.9/6.09e+2	580/856.8/6.09e+2
8	0.01	**7/ 23.3/1.29e+1**	10/ 35.3/1.29e+1	3(5)/ 73.9(20.0)/1.29e+1	10/ 40.1/1.29e+1	10/ 38.6/1.29e+1
	0.10	230/695.8/6.34e+1	**60/191.8/6.34e+1**	13(20)/340.2(76.1)/6.34e+1	340/1290.5/6.34e+1	470/1614.9/6.34e+1
	1.00	320/1004.8/1.77e+2	140/468.2/1.77e+2	8(50)/860.1(184.6)/1.77e+2	100/396.1/1.77e+2	**90/328.2/1.77e+2**
	10.00	850/2633.1/8.53e+2	**430/1403.1/8.53e+2**	2(370)/2024.6(1335.6)/8.53e+2	790/3045.9/8.53e+2	580/2073.0/8.53e+2

where $u = \begin{pmatrix} u_0 & \cdots & u_N \end{pmatrix} \in \mathbb{R}^{p\times(N+1)}$. We apply the five algorithms from Table 4.2 to solve these problems for different values of μ, using the same parameters as in the previous section. The results are reported in Table 4.4. We see that Dual AGP is usually either the fastest or the second fastest algorithm, and is the most robust.

Table 4.3
Description of test problems, taken from DaISy [11].

P	**Description**	$N+1$	m	q
1	CD Player arm	513	2	388
2	Continuous Stirring Tank Reactor	1876	2	1793
3	Hair Dryer	251	1	168
4	Steam Heat Exchanger	1001	1	918
5	Heat Flow density	421	1	296
6	Industrial Winding Process	626	2	375
7	Glass Furnace	312	6	145
8	Industrial Dryer	217	3	50

Table 4.4

Computational results for problems from Table 4.3.

		D. GP	D. AGP	D. PPA	P. ADM2	D. ADM
P	μ	iter/cpu/obj	iter/cpu/obj	iter/cpu/obj	iter/cpu/obj	iter/cpu/obj
1	0.01	60/ 1.9/4.60e-1	**30/ 1.0/4.60e-1**	15(10)/ 1.6(0.4)/4.60e-1	160/ 6.1/4.60e-1	220/ 7.8/4.60e-1
	0.10	1470/ 46.6/2.69e+0	180/ 6.1/2.69e+0	**34(30)/ 5.6(1.1)/2.69e+0**	470/ 17.6/2.69e+0	640/ 22.3/2.69e+0
	1.00	max/ 61.5/4.49e+0	1200/ 39.3/4.48e+0	39(215)/ 25.0(7.6)/4.48e+0	**450/ 16.4/4.48e+0**	530/ 18.1/4.48e+0
	10.00	max/ 61.0/6.33e+0	max/ 64.5/5.59e+0	**47(max)/124.4(69.2)/5.57e+0**	max/ 72.0/5.88e+0	max/ 67.5/5.61e+0
2	0.01	**10/ 1.4/5.71e+1**	**10/ 1.5/5.71e+1**	2(290)/ 61.1(49.1)/5.71e+1	max/358.2/5.71e+1	max/337.1/5.71e+1
	0.10	30/ 4.1/5.69e+2	**10/ 1.5/5.69e+2**	2(65)/ 27.0(11.0)/5.69e+2	1060/189.1/5.69e+2	1370/230.6/5.69e+2
	1.00	50/ 6.8/5.58e+3	**20/ 3.0/5.58e+3**	2(35)/ 25.5(6.0)/5.58e+3	770/136.8/5.58e+3	1060/178.2/5.58e+3
	10.00	130/ 17.6/5.29e+4	**50/ 7.4/5.29e+4**	8(30)/ 55.9(5.1)/5.29e+4	550/ 97.5/5.29e+4	750/125.7/5.29e+4
3	0.01	**20/ 0.1/5.38e-1**	**20/ 0.2/5.38e-1**	10(25)/ 0.4(0.2)/5.38e-1	180/ 1.6/5.38e-1	250/ 2.1/5.38e-1
	0.10	240/ 1.7/4.15e+0	**80/ 0.6/4.15e+0**	25(30)/ 0.9(0.3)/4.15e+0	410/ 3.5/4.15e+0	560/ 4.6/4.15e+0
	1.00	550/ 3.7/3.37e+1	160/ 1.2/3.37e+1	14(50)/ 1.7(0.4)/3.37e+1	**110/ 0.9/3.37e+1**	**130/ 1.0/3.37e+1**
	10.00	**390/ 2.6/2.05e+2**	390/ 2.9/2.05e+2	17(350)/ 4.7(2.9)/2.05e+2	720/ 6.1/2.05e+2	530/ 4.2/2.05e+2
4	0.01	**7/ 0.2/4.50e+1**	**10/ 0.3/4.50e+1**	1(95)/ 4.3(3.5)/4.50e+1	80/ 3.0/4.50e+1	140/ 5.1/4.50e+1
	0.10	30/ 0.8/4.43e+2	**10/ 0.3/4.43e+2**	3(30)/ 4.5(1.1)/4.43e+2	280/ 10.5/4.43e+2	360/ 13.2/4.43e+2
	1.00	100/ 2.8/4.28e+3	**20/ 0.6/4.28e+3**	3(10)/ 6.0(0.4)/4.28e+3	190/ 7.0/4.28e+3	260/ 9.4/4.28e+3
	10.00	160/ 4.4/4.07e+4	**80/ 2.4/4.07e+4**	8(15)/ 14.5(0.6)/4.07e+4	110/ 4.0/4.07e+4	150/ 5.4/4.07e+4
5	0.01	**9/ 0.1/7.79e-1**	**10/ 0.1/7.79e-1**	**1(5)/ 0.1(0.1)/7.79e-1**	**10/ 0.1/7.79e-1**	**10/0.1/7.79e-1**
	0.10	410/ 4.1/3.62e+0	**80/ 0.9/3.62e+0**	**25(20)/ 1.0(0.3)/3.62e+0**	260/ 3.2/3.62e+0	360/ 4.4/3.62e+0
	1.00	850/ 8.6/1.48e+1	360/ 4.0/1.48e+1	7(105)/ 2.7(1.3)/1.48e+1	210/ 2.6/1.48e+1	**160/ 2.0/1.48e+1**
	10.00	**700/ 7.0/8.00e+1**	700/ 7.5/8.00e+1	3(795)/ 11.7(9.8)/8.00e+1	1740/ 21.2/8.00e+1	1260/ 15.3/8.00e+1
6	0.01	**10/ 0.3/2.04e+0**	**10/ 0.4/2.04e+0**	8(10)/ 1.1(0.4)/2.04e+0	20/ 0.8/2.04e+0	20/ 0.8/2.04e+0
	0.10	190/ 6.6/1.31e+1	**60/ 2.2/1.31e+1**	18(15)/ 4.9(0.6)/1.31e+1	120/ 4.8/1.31e+1	160/ 6.2/1.31e+1
	1.00	max/ 64.9/4.59e+1	320/ 11.2/4.59e+1	47(105)/ 36.1(4.0)/4.59e+1	230/ 8.9/4.59e+1	**200/ 7.4/4.59e+1**
	10.00	990/ 32.0/2.18e+2	**830/ 28.8/2.18e+2**	16(760)/ 37.7(28.7)/2.18e+2	1740/ 67.3/2.18e+2	1270/ 46.8/2.18e+2
7	0.01	**10/ 0.7/3.35e+0**	**10/ 0.7/3.35e+0**	8(10)/ 2.0(0.8)/3.35e+0	20/ 1.6/3.35e+0	20/ 1.5/3.35e+0
	0.10	340/ 22.5/2.44e+1	**70/ 5.0/2.45e+1**	17(20)/ 5.7(1.4)/2.45e+1	230/ 17.6/2.44e+1	310/ 21.4/2.44e+1
	1.00	1640/101.4/1.42e+2	250/ 16.5/1.42e+2	25(75)/ 24.1(5.0)/1.42e+2	190/ 13.4/1.42e+2	**170/ 11.0/1.42e+2**
	10.00	max/122.3/4.50e+2	**530/ 34.5/4.50e+2**	13(420)/ 44.4(27.8)/4.50e+2	1190/ 83.8/4.50e+2	860/ 55.3/4.50e+2
8	0.01	**10/ 0.1/1.43e+0**	**10/ 0.1/1.43e+0**	17(20)/ 0.5(0.2)/1.43e+0	110/ 1.1/1.43e+0	150/ 1.4/1.43e+0
	0.10	250/ 2.0/1.22e+1	**70/ 0.6/1.22e+1**	**14(30)/ 0.7(0.3)/1.22e+1**	490/ 5.0/1.22e+1	680/ 6.3/1.22e+1
	1.00	max/ 15.8/6.69e+1	350/ 3.1/6.69e+1	**62(50)/ 2.5(0.4)/6.69e+1**	770/ 7.6/6.69e+1	1060/ 9.5/6.69e+1
	10.00	max/ 15.1/2.49e+2	1460/ 11.9/2.49e+2	45(325)/ 15.6(2.9)/2.49e+2	**990/ 9.3/2.49e+2**	1250/ 10.7/2.49e+2

To illustrate the performance of Dual AGP on real data, we consider the first problem in Table 4.3 and attempt to identify the system order using solutions obtained from Dual AGP. Following [31], we use 200 data points for the identification experiment, and 600 data points for model validation. We draw parameters μ between 0.1 and 10, solve the corresponding Problem (4.2) by Dual AGP, then estimate the state-space matrices A, B, C and D in (4.1) as done in [31, Section 5.2] and compute our identification and validation errors as in [31, Eq. (5.7)].

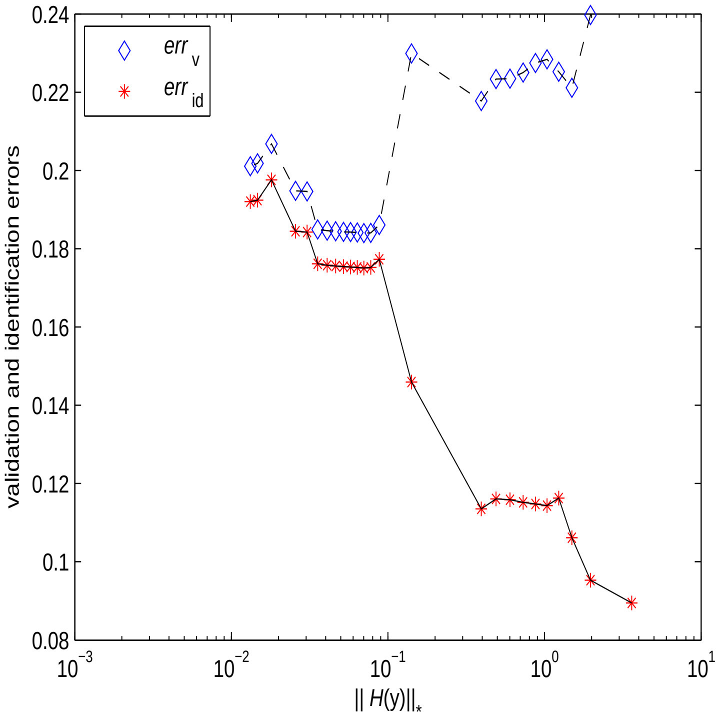
We identify the corresponding parameter μ^* at which a "branching" occurs; see Figure 4.1, which plots identification and validation errors against the nuclear norm of $\mathcal{H}(y)$ for the whole range of μ used. The estimated system order is then set to be the rank of $\mathcal{H}(y^*)$, where y^* is the approximate solution of (4.2) with $\mu = \mu^*$. Table 4.5 shows some identification errors ($err_{\rm id}$) and validation errors ($err_{\rm v}$), as well as the corresponding rank and nuclear norm of $\mathcal{H}(y^*)$. From this and Figure 4.1, we conclude that the estimated system order in this case is 3, which agrees with the result obtained in [31, Table 5.2].

Finally, we apply the interior point method in [31] to solve the first problem in Table 4.3. We use the code (written in Python and calling cvxopt 1.1.3) available on

TABLE 4.5
Identification errors and validation errors for CD player arm data.

μ	$\mathrm{rank}(\mathcal{H}(y^*))$	err_{id}	err_{v}	$\|\mathcal{H}(y^*)\|_*$
0.9	8	0.1161	0.2234	0.4901
1	8	0.1135	0.2177	0.3947
1.5	5	0.1459	0.2299	0.1414
2	3	0.1773	0.1861	0.0881
2.5	3	0.1752	0.1841	0.0778
3	3	0.1751	0.1840	0.0702
3.5	3	0.1752	0.1841	0.0637

FIG. 4.1. *Plot of* err_{id} *and* err_{v} *against* $\|\mathcal{H}(y)\|_*$.



the authors' webpage, compiled with Python 2.5.2. While the interior point method provides a solution of higher accuracy (the relative duality gap is usually at most $1e-6$, rather than the $1e-4$ as required in our first order methods), it is much slower. We observe that the number of iterations and CPU time for the interior point method are not sensitive to the change of μ, and are usually around 9 iterations and 450 seconds respectively. In particular, when $\mu \leq 1$, this method is significantly slower than Dual AGP, which takes between 1 and 40 seconds for this range of μ. Thus, the first order methods appear more suitable for larger scale identification problems.

5. Stochastic system realization. Another fundamental problem in linear system theory is finding a minimal stochastic ARMA (autoregressive moving average) model for a vector random process, given noisy and/or partial estimates of process covariances. Covariance estimates are often obtained from sample averages of a sequence of noisy observations of the process, hence including both measurement noise and error due to the finite sample size. Here we focus on a form of this problem, described in [32, section II.B]; see also [34]. Consider a state-space model of an ARMA process $y_t \in \mathbb{R}^n$,

$$\begin{aligned} x_{t+1} &= Ax_t + Be_t, \\ y_t &= Cx_t + e_t, \end{aligned}$$

where $x_t \in \mathbb{R}^r$, and e_t is white noise with covariance matrix Q. The process covariances $h_i = \mathbf{E}(y_t {y_{t+i}}^T)$ satisfy

$$h_0 = CPC^T + Q, \quad h_t = CA^{t-1}D, \ t \geq 1,$$

where $D = APC^T + BQ$, and $P = \mathbf{E}(x_t x_t^T)$ satisfies the Lyapunov equation $P = APA^T + BQB^T$. In a stochastic realization problem, we are given noisy estimates of h_i, denoted by $\tilde{h}_i$, $i = 1, \ldots, T-1$, and the goal is to find the minimal model order r as well as the model parameters A, B, C, Q. It is known that the minimal order is equal to the rank of the block-Hankel matrix $H_{n,n,j,k}$ consisting of the exact process covariances [34] (as in the system ID problem, we need the Hankel matrix to be large enough, that is, j and k should be larger than the rank).

A general form of this problem, allowing for both noise and missing covariance information, can be stated as follows,

$$\min_y \frac{1}{2}\|w \circ (y - \tilde{h})\|_F^2 + \mu \|H_{m,n,j,k}(y)\|_*, \tag{5.1}$$

where $\circ$ denotes the Hadamard (or entry-wise) product, and $w = \begin{pmatrix} w_0 & \cdots & w_{j+k-2} \end{pmatrix}$ is an $m \times n(j+k-1)$ matrix, with each w_i being an $m \times n$ zero matrix or a matrix of all ones (denoting the case where some covariance blocks are unknown or missing), or a matrix with $0, 1$ entries (denoting the case where some covariance entries are missing). The problem corresponds to (2.1) with $D = I$, $\mathcal{A}(y) = \mathrm{vec}(w \circ y)$ and $b = \mathrm{vec}(w \circ \tilde{h})$. Thus,

$$\sigma_{\max}(\mathcal{A}) = \sigma_{\max}(D) = 1, \quad \mathcal{H}^*(\Lambda) = H^*_{m,n,j,k}(\Lambda).$$

5.1. Computational results. In this section, we compare different algorithms for solving (5.1). Specifically, we consider Primal ADM, Dual ADM and Dual PPA. Moreover, since the action of $(\mathcal{H}^*\mathcal{H} + \mathcal{A}^*\mathcal{A})^{-1}$ and $(\mathcal{I} + \mathcal{A}^*\mathcal{A})^{-1}$ on vectors can be easily computed, we also consider the variant of Primal ADM with $Q_0 = 0$ (referred to as Primal ADM3) and the variant of Dual ADM with $Q_1 = 0$ (referred to as Dual ADM3). The convergence of these two variants follows from [2, Proposition 4.2] and Theorem 8.1, respectively. Furthermore, notice that the quadratic term in (5.1) is *not* strictly convex in y, the dual problem will then have additional linear constraints that makes gradient projection expensive computationally. Thus, we do not consider gradient projection algorithms for solving (5.1).

We initialize the algorithms similarly as in the previous section, and terminate the algorithms by checking

$$\max\left\{\frac{f(y^k) + d(-w \circ \mathcal{H}^*(\mathcal{P}(\Lambda^k)))}{\max\{1, |d(-w \circ \mathcal{H}^*(\mathcal{P}(\Lambda^k)))|\}}, \frac{\|\mathcal{H}^*(\mathcal{P}(\Lambda^k)) - w \circ \mathcal{H}^*(\mathcal{P}(\Lambda^k))\|_F}{\max\{1, \|\mathcal{H}^*(\mathcal{P}(\Lambda^k))\|_F\}}\right\} < 1e-4,$$

with $\{(y^k, \Lambda^k)\}$ defined as in Sections 3.1 and 3.2, and $\mathcal{P}$ is the projection onto the spectral norm ball with radius μ [5]. For the ADMs, we set $\beta = \frac{\mu \mathbf{r}}{2\sigma_{\max}(h)}$ and $\sigma = \frac{0.95}{\mathbf{r}+1}$ for Primal ADM and use the same β for Primal ADM3. We set $\beta = \frac{\sigma_{\max}(h)}{8\mu\mathbf{r}}$, $\sigma_1 = 0.95$ and $\sigma_2 = \frac{0.95}{\mathbf{r}}$ for Dual ADM and use the same β and σ_2 for Dual ADM3.

Following [32, Section II(B)], we generate matrices $A \in \mathrm{I\!R}^{r\times r}$, $B \in \mathrm{I\!R}^{r\times n}$ and $C \in \mathrm{I\!R}^{n\times r}$ with i.i.d. Gaussian entries. These matrices are then normalized to have spectral norm 1. We also randomly generate an initial state $x_0 \sim N(0, I)$ and noise vectors $e_t \sim N(0, I)$ for $t = 0, ..., T-1$. We then generate an output $\bar{y}_t$, $t = 0, ..., T-1$, according to the state space model:

$$\begin{aligned} x_{t+1} &= Ax_t + Be_t, \\ \bar{y}_t &= Cx_t + e_t. \end{aligned}$$

To model measurement noise, we can further add noise to $\bar{y}$ and get $\tilde{y} = \bar{y} + \sigma\epsilon$, where ϵ has i.i.d. Gaussian entries with variance 1. We then set, for each $i = 0, ..., k-1$,

$$\tilde{h}_i = \frac{1}{T}\sum_{t=0}^{T-1-i} \tilde{y}_{t+i}\tilde{y}_t^T,$$

and $\tilde{h}_i$ is zero for $i \geq k$. Finally, set $w = \begin{pmatrix} w_0 & \cdots & w_{j+k-2} \end{pmatrix}$ such that $w_0 = \cdots = w_{k-1}$ equals the matrix of all ones, and zero otherwise. In the tests below, we consider $T = 1000$, $n = 20, 30$ and $k = 100, 200$. We use $r = 10$ and hence set $j = 21$. The statistics of the test problems used are reported in Table 5.1; recall that $q = nk$ in this case. We run our algorithms for a range of values of μ, namely $\mu = 1e-2, 1e-1, 1, 10$, in our simulations below to study the performance of the algorithms for different values of μ. The computational results are reported in Table 5.2. We see that Primal ADM3 works best, followed by Dual ADM [6]. We also see that Dual ADM works better when μ is small, while Primal ADM3 works better when μ is large (≥ 0.1).

TABLE 5.1
Parameters for the randomly generated test problems.

P	k	n
1	100	20
2	100	30
4	200	20
5	200	30

We also adapted Cadzow's method [4], an approach based on alternating projections, that has been used in engineering applications for similar problems. We start by specifying an estimate $\tilde{r}$ of the order and $\tilde{\epsilon} > 0$ of the noise level. In each iteration, we first project onto the set of matrices with a Hankel structure to obtain H^k, then project onto the ball centered at $\tilde{h}$ with radius $\tilde{\epsilon}$ to obtain Y^k, and finally project onto

[5] For Dual PPA, we initialize the parameters λ_k are initialized at $\lambda_0 = 1$ if $\mu < 0.5$ and $\lambda_0 = 10$ otherwise, and update according to that $\lambda_{k+1} = 2\lambda_k$ if $\text{gap}_k/\text{gap}_{k-1} > 0.95$, where gap_k denotes the relative duality gap in the kth outer iteration. The $intol_k$ is decreased from 0.0016 according to the value of gap_k, and is bounded below by $1e-4$.

[6] The two versions of Dual ADM perform almost identically. Hence, we do not report test results for the variant with $Q_1 = 0$

Table 5.2
Computational results for problems from Table 5.1.

		P. ADM	P. ADM3	D. ADM	D. PPA
P	μ	iter/cpu/obj	iter/cpu/obj	iter/cpu/obj	iter/cpu/obj
1	0.01	170/292.1/5.38e+0	160/274.1/5.38e+0	**110/170.7/5.38e+0**	3(15)/243.6(23.8)/5.38e+0
	0.10	**120/194.2/2.73e+1**	**120/193.5/2.73e+1**	150/216.1/2.73e+1	29(15)/884.0(22.7)/2.73e+1
	1.00	220/354.9/4.18e+1	**30/ 47.2/4.18e+1**	160/230.1/4.18e+1	2(55)/1137.7(80.6)/4.18e+1
	10.00	210/345.0/4.38e+1	**20/ 31.7/4.38e+1**	220/322.6/4.38e+1	3(95)/205.1(140.3)/4.38e+1
2	0.01	220/1373.1/9.12e+0	160/991.8/9.12e+0	**160/904.3/9.12e+0**	3(25)/931.0(143.9)/9.12e+0
	0.10	80/465.7/4.99e+1	**70/403.0/4.99e+1**	90/475.6/4.99e+1	16(20)/2462.4(109.1)/4.99e+1
	1.00	200/1116.2/7.04e+1	**30/164.9/7.04e+1**	140/697.7/7.04e+1	4(35)/5878.5(178.6)/7.04e+1
	10.00	210/1195.6/7.16e+1	**20/109.5/7.16e+1**	190/969.5/7.16e+1	4(90)/1074.5(461.1)/7.16e+1
3	0.01	190/625.2/7.16e+0	160/516.8/7.16e+0	140/405.0/7.16e+0	**3(15)/374.4(42.6)/7.16e+0**
	0.10	70/221.5/4.02e+1	60/180.8/4.02e+1	**50/137.8/4.02e+1**	9(15)/704.9(33.2)/4.02e+1
	1.00	180/428.0/5.29e+1	**30/ 69.8/5.29e+1**	130/271.4/5.29e+1	5(25)/2662.0(53.4)/5.29e+1
	10.00	180/432.6/5.33e+1	**20/ 46.4/5.33e+1**	190/400.1/5.33e+1	4(90)/411.5(192.5)/5.33e+1
4	0.01	220/1927.8/1.35e+1	190/1657.2/1.35e+1	**150/1171.2/1.35e+1**	3(20)/1411.0(161.6)/1.35e+1
	0.10	100/824.4/8.74e+1	**50/414.0/8.74e+1**	70/517.7/8.74e+1	4(15)/2288.2(118.7)/8.74e+1
	1.00	140/1141.3/1.23e+2	**30/241.4/1.23e+2**	110/789.9/1.23e+2	4(25)/8607.5(187.9)/1.23e+2
	10.00	180/1482.8/1.25e+2	**20/159.9/1.25e+2**	190/1380.5/1.25e+2	4(85)/1453.2(627.7)/1.25e+2

the (nonconvex) set of matrices with rank less than $\tilde{r}$ to obtain R^k. The algorithm is terminated when

$$\frac{\max\{\|H^k - Y^k\|_F, \|R^k - Y^k\|_F\}}{\max\{1, \|H^k\|_F\}} < 1e-3.$$

In the examples we tested, the performance and convergence of this algorithm is very sensitive to the values of the two parameters, the noise level and the order estimate. On the other hand, the convex optimization approach presented here only depends on one parameter (μ), and the performance does not change significantly as μ varies.

Finally, we mention that another common approach to solving the system realization problem is via subspace methods [34]. For a comparison between the nuclear norm approach and the subspace method applied to this problem, we refer the readers to [32, Section 2B], where Problem (5.1) was solved via an interior-point solver.

6. Concluding remarks. As a systematic approach to capturing the tradeoff between a model's order and complexity with fitting (and validation) errors, a tradeoff that commonly arises in diverse modeling applications, we studied the optimization problem of minimizing the nuclear norm of matrices with linear structure, including Hankel, Toeplitz, and moment structures. We then focused on first-order methods for solving the resulting optimization problem. In our computational experiments, the gradient projection method (accelerated by Nesterov extrapolation techniques) usually outperforms other first-order methods in terms of CPU time on both real and simulated data for the system identification problem, while for the system realization problem, the alternating direction method, as applied to a certain primal reformulation, usually outperforms other first-order methods in terms of CPU time. In our tests, we also observe that these methods outperform the interior point implementation proposed in [31] for system identification problems.

An interesting direction for future work on this problem is whether there are conditions under which the nuclear norm heuristic can be theoretically guaranteed to find the minimum rank solution. In order to make analogies with the existing low-rank matrix recovery framework, we can consider the following problem: how many generic, random linear measurements of a rank-r, $n \times n$ Hankel matrix suffice for correct recovery of the Hankel matrix? If we ignore the Hankel structure, existing results on recovery from random Gaussian measurements require $\mathcal{O}(nr)$ measurements [6]; however, it is expected that the true number would be much lower due to the Hankel structure.

Another future research direction involves applying our algorithms to a broader range of applications identified in the introduction. Some of these problems have further structure that the algorithms can exploit.

7. Appendix I: an alternative formulation. In this appendix, we describe an alternative formulation for modeling the structured matrix rank minimization. Instead of minimizing a least square fitting error, we constrain the difference $\mathcal{A}(y) - b$ in a set modeling uncertainty. The problem is then formulated as follows:

$$\begin{array}{rl} \min & \|\mathcal{H}(y)\|_* \\ \text{s.t.} & \mathcal{A}(y) - b \in B, \end{array} \tag{7.1}$$

where B is the closed convex set modeling uncertainty. For instance, Problem (1.2) can be modeled as (7.1) with $B = [l_1, b_1] \times [l_2, b_2] \times \cdots \times [l_n, b_n]$.

As in Section 2, we can rewrite (7.1) as

$$\begin{array}{rl} v_1 := \min & \|Y\|_* \\ \text{s.t.} & Y + \mathcal{H}(y) = 0, \\ & z + \mathcal{A}(y) = b, \\ & z \in B. \end{array} \tag{7.2}$$

It is then not hard to show that the dual to (7.2) is equivalent to solving

$$\begin{array}{rl} -v_1 = \min & s_B(\gamma) - b^T\gamma \\ \text{s.t.} & \mathcal{H}^*(\Lambda) + \mathcal{A}^*(\gamma) = 0, \\ & \Lambda^T\Lambda \preceq I, \end{array} \tag{7.3}$$

where $s_B(\gamma) := \max_{y \in B} y^T\gamma$ is the support function of the set B.

Unlike (2.3), the objective function of (7.3) is in general not differentiable, hence gradient projection methods are not easily applicable. However, the ADMs and the dual PPA can be suitably applied to solve (7.2) and (7.3). The efficiency in solving the subproblems depends on the specific form of B.

8. Appendix II: convergence of the proximal alternating direction method. In this appendix, we provide a convergence proof of a proximal alternating direction method, which covers all versions of proximal ADMs used in this paper. Consider the convex optimization problem with the following separable structure

$$\begin{array}{rl} \min & f(x) + g(y) \\ \text{s.t.} & Ax + By = c, \end{array} \tag{8.1}$$

where $f : \mathcal{X} \to (-\infty, +\infty]$ and $g : \mathcal{Y} \to (-\infty, +\infty]$ are closed proper convex functions, $A : \mathcal{X} \to \mathcal{Z}$ and $B : \mathcal{Y} \to \mathcal{Z}$ are linear operators, $\mathcal{X}, \mathcal{Y}$ and $\mathcal{Z}$ are real finite dimensional Euclidean spaces with inner product $\langle \cdot, \cdot \rangle$ and its induced norm $\| \cdot \|$. The proximal alternating direction method (proximal ADM) for solving (8.1) takes the following form:

Proximal ADM

Step 0. Input $(x^0, y^0, z^0) \in \mathcal{X} \times \mathcal{Y} \times \mathcal{Z}$.

Step 1. Set

$$\begin{cases} x^{k+1} = \arg\min_{x \in \mathcal{X}} f(x) - \langle z^k, Ax \rangle + \frac{\lambda}{2}\|Ax + By^k - c\|^2 + \frac{1}{2}\|x - x^k\|_S^2, \\ y^{k+1} = \arg\min_{y \in \mathcal{Y}} g(y) - \langle z^k, By \rangle + \frac{\lambda}{2}\|Ax^{k+1} + By - c\|^2 + \frac{1}{2}\|y - y^k\|_T^2, \\ z^{k+1} = z^k - \lambda(Ax^{k+1} + By^{k+1} - c), \end{cases} \tag{8.2}$$

where $\lambda > 0$, S and T are self-adjoint positive semidefinite operators on $\mathcal{X}$ and $\mathcal{Y}$ respectively.

Step 2. If a termination criterion is not met, go to Step 1.

When $S = 0$ and $T = 0$, the proximal ADM (8.2) reduces to the classical ADM introduced by Glowinski and Marroco [22] and Gabay and Mericier [21]. It was shown by Eckstein and Bertsekas [15] that the ADM, as a special case of Douglas-Rachford splitting [20], is actually an application of the proximal point algorithm on the dual problem by means of a specially-constructed splitting operator. Based on the same argument by further applying a change of variable to the operators, Eckstein [14] presented the first proximal ADM as in (8.2) with $S = \mu_1 I$ and $T = \mu_2 I$ for positive constants $\mu_1 > 0$ and $\mu_2 > 0$. Later, He et al. [25] further extended the idea of Eckstein [14] to monotone variational inequalities to allow λ, S, and T to be replaced by different parameters λ_k, S_k, and T_k in each iteration. The convergence results provided in [14] and [25] for the proximal ADM both need S and T to be positive definite, which limits the applications of the method. However, we notice that by slightly revising the proof provided by He et al. in [25], one may readily prove the following theorem. For clarity and completeness, we include a proof here.

For technical reasons, we assume the following constraint qualification:

CQ There exists $(x_0, y_0) \in \mathrm{ri}(\mathrm{dom}\, f \times \mathrm{dom}\, g) \cap P$, where P is the constraint set in (8.1).

Under **CQ**, it follows from [47, Corollary 28.2.2] and [47, Corollary 28.3.1] that $(\bar{x}, \bar{y}) \in \mathcal{X} \times \mathcal{Y}$ is an optimal solution to Problem (8.1) if and only if there exists a Lagrange multiplier $\bar{z} \in \mathcal{Z}$ such that

$$A^*\bar{z} \in \partial f(\bar{x}), \quad B^*\bar{z} \in \partial g(\bar{y}), \quad A\bar{x} + B\bar{y} - c = 0, \tag{8.3}$$

where ∂f and ∂g are the subdifferential mappings of f and g respectively. Moreover, any $\bar{z} \in \mathcal{Z}$ satisfying (8.3) is an optimal solution to the dual problem of (8.1).

THEOREM 8.1. *Assume that the solution set of* (8.1) *is nonempty and* **CQ** *holds. Let* $\{(x^k, y^k, z^k)\}$ *be generated from the proximal ADM. Then* $\{(x^k, y^k)\}$ *converges to an optimal solution to* (8.1) *and* $\{z^k\}$ *converges to an optimal solution to the dual problem of* (8.1) *if one of the following conditions holds:*

(a) *f and g are strongly convex;*

(b) *f is strongly convex and $B^*B + T$ is positive definite;*

(c) *g is strongly convex and $A^*A + S$ is positive definite;*

(d) *S is positive definite and B is injective;*

(e) *T is positive definite and A is injective;*

(f) *S and T are positive definite.*

Proof. We first show that the sequence generated by proximal ADM is bounded. Notice that the iteration scheme (8.2) of the proximal ADM can be rewritten as: for

$k = 0, 1, 2, \cdots,$

$$\begin{cases} 0 \in \partial f(x^{k+1}) - A^*[z^k - \lambda(Ax^{k+1} + By^k - c)] + S(x^{k+1} - x^k), \\ 0 \in \partial g(y^{k+1}) - B^*[z^k - \lambda(Ax^{k+1} + By^{k+1} - c)] + T(y^{k+1} - y^k), \\ z^{k+1} = z^k - \lambda(Ax^{k+1} + By^{k+1} - c). \end{cases} \tag{8.4}$$

On the other hand, since the subdifferential mappings of the closed proper convex functions are maximal monotone [48, Theorem 12.17], there exist some constants $\sigma_1, \sigma_2 \geq 0$ such that

$$\begin{aligned} \langle u - \bar{u}, x^{k+1} - \bar{x}\rangle \geq \sigma_1 \|x^{k+1} - \bar{x}\|^2, \quad \forall u \in \partial f(x^{k+1}), \bar{u} \in \partial f(\bar{x}), \\ \langle v - \bar{v}, y^{k+1} - \bar{y}\rangle \geq \sigma_2 \|y^{k+1} - \bar{y}\|^2, \quad \forall v \in \partial g(y^{k+1}), \bar{v} \in \partial g(\bar{y}), \end{aligned} \tag{8.5}$$

where $\sigma_1 > 0$ if f is strongly convex, $\sigma_2 > 0$ if g is strongly convex, and $(\bar{x}, \bar{y}, \bar{z})$ satisfies (8.3). Combining (8.5) with (8.3) and (8.4), we have

$$\begin{aligned} \langle A^*[z^k - \lambda(Ax^{k+1} + By^k - c)] - S(x^{k+1} - x^k) - A^*\bar{z}, x^{k+1} - \bar{x}\rangle \geq \sigma_1 \|x^{k+1} - \bar{x}\|^2, \\ \langle B^*[z^k - \lambda(Ax^{k+1} + By^{k+1} - c)] - T(y^{k+1} - y^k) - B^*\bar{z}, y^{k+1} - \bar{y}\rangle \geq \sigma_2 \|y^{k+1} - \bar{y}\|^2. \end{aligned}$$

Let $w_e^k := w^k - \bar{w}$ for notational simplicity, where w represents x, y, z respectively. Then the two inequalities above can be rewritten as follows:

$$\begin{aligned} \langle z_e^k - \lambda(Ax_e^{k+1} + By_e^k), Ax_e^{k+1}\rangle - \langle S(x^{k+1} - x^k), x_e^{k+1}\rangle \geq \sigma_1 \|x_e^{k+1}\|^2, \\ \langle z_e^k - \lambda(Ax_e^{k+1} + By_e^{k+1}), By_e^{k+1}\rangle - \langle T(y^{k+1} - y^k), y_e^{k+1}\rangle \geq \sigma_2 \|y_e^{k+1}\|^2. \end{aligned}$$

Adding up these two inequalities we obtain that

$$\begin{aligned} &\langle z_e^k, Ax_e^{k+1} + By_e^{k+1}\rangle - \lambda\langle Ax_e^{k+1} + By_e^k, Ax_e^{k+1}\rangle - \lambda\langle Ax_e^{k+1} + By_e^{k+1}, By_e^{k+1}\rangle \\ &- \langle S(x^{k+1} - x^k), x_e^{k+1}\rangle - \langle T(y^{k+1} - y^k), y_e^{k+1}\rangle \geq \sigma_1 \|x_e^{k+1}\|^2 + \sigma_2 \|y_e^{k+1}\|^2. \end{aligned}$$

Using the relation $z^{k+1} = z^k - \lambda(Ax_e^{k+1} + By_e^{k+1})$ and the elementary relations $\langle u, v\rangle = \frac{1}{2}(\|u\|^2 + \|v\|^2 - \|u - v\|^2) = \frac{1}{2}(\|u + v\|^2 - \|u\|^2 - \|v\|^2)$, we obtain further that

$$\begin{aligned} &\|x_e^{k+1}\|_S^2 + \|y_e^{k+1}\|_T^2 + \lambda\|By_e^{k+1}\|^2 + \frac{1}{\lambda}\|z_e^{k+1}\|^2 \leq \|x_e^k\|_S^2 + \|y_e^k\|_T^2 + \lambda\|By_e^k\|^2 + \frac{1}{\lambda}\|z_e^k\|^2 \\ &\quad - (2\sigma_1\|x_e^{k+1}\|^2 + 2\sigma_2\|y_e^{k+1}\|^2 + \|x^{k+1} - x^k\|_S^2 + \|y^{k+1} - y^k\|_T^2 + \lambda\|Ax_e^{k+1} + By_e^k\|^2). \end{aligned} \tag{8.6}$$

From (8.6), we see immediately that the sequence $\{\|x_e^k\|_S^2 + \|y_e^k\|_T^2 + \lambda\|By_e^k\|^2 + \frac{1}{\lambda}\|z_e^k\|^2\}$ is monotonically nonincreasing (and thus bounded), and

$$\lim_{k\to\infty} 2\sigma_1\|x_e^{k+1}\|^2 + 2\sigma_2\|y_e^{k+1}\|^2 + \|x^{k+1} - x^k\|_S^2 + \|y^{k+1} - y^k\|_T^2 + \lambda\|Ax_e^{k+1} + By_e^k\|^2 = 0. \tag{8.7}$$

As an immediate consequence of these two facts, we see that the sequences

$$\begin{aligned} \{\|x_e^{k+1}\|_S^2\}, \{\|y_e^{k+1}\|_T^2\}, \{\|By_e^{k+1}\|^2\}, \{\|z_e^{k+1}\|^2\}, \\ \{\sigma_1\|x_e^{k+1}\|^2\}, \{\sigma_2\|y_e^{k+1}\|^2\}, \{\|Ax_e^{k+1} + By_e^k\|^2\} \end{aligned}$$

are bounded. Thus, the sequence $\{\|By_e^{k+1}\|^2 + \|y_e^{k+1}\|_T^2\}$ is bounded. Moreover, since

$$\|Ax_e^{k+1}\| \le \|Ax_e^{k+1} + By_e^k\| + \|By_e^k\|, \tag{8.8}$$

it follows that the sequence $\{\|Ax_e^{k+1}\|^2 + \|x_e^{k+1}\|_S^2\}$ is also bounded. From the boundedness of the above sequences, it is now routine to deduce that the sequence $\{(x^k, y^k, z^k)\}$ is bounded under the assumption of the theorem.

Since the sequence $\{(x^k, y^k, z^k)\}$ is bounded, there exists a subsequence $\{(x^{k_i}, y^{k_i}, z^{k_i})\}$ that converges to a limit point, say $(x^\infty, y^\infty, z^\infty)$. We next show that (x^∞, y^∞) is an optimal solution to Problem (8.1) and z^∞ is a corresponding Lagrange multiplier.

To this end, we first note from (8.7) that

$$\begin{gathered}\lim_{k\to\infty} \sigma_1\|x_e^{k+1}\| = 0,\ \lim_{k\to\infty} \sigma_2\|y_e^{k+1}\| = 0,\ \lim_{k\to\infty} \|Ax_e^{k+1} + By_e^k\| = 0,\\ \lim_{k\to\infty} \|S(x^{k+1} - x^k)\| = 0,\ \lim_{k\to\infty} \|T(y^{k+1} - y^k)\| = 0.\end{gathered} \tag{8.9}$$

Thus, if either S or T is positive definite, we have

$$\lim_{k\to\infty} \|x^{k+1} - x^k\| = 0 \ \text{ or } \ \lim_{k\to\infty} \|y^{k+1} - y^k\| = 0. \tag{8.10}$$

Similarly, if either f or g is strongly convex, then (8.10) also holds. Thus, (8.10) holds under the assumption of the theorem. Since

$$\begin{aligned}\|Ax_e^{k+1} + By_e^{k+1}\| &\le \|Ax_e^{k+1} + By_e^k\| + \|B(y^{k+1} - y^k)\|,\\ \|Ax_e^{k+1} + By_e^{k+1}\| &\le \|Ax_e^{k+2} + By_e^{k+1}\| + \|A(x^{k+2} - x^{k+1})\|,\end{aligned}$$

we obtain from (8.9), (8.10) and the definition of z^{k+1} in (8.4) that

$$\lim_{k\to\infty} \|z^{k+1} - z^k\| = \lim_{k\to\infty} \lambda\|Ax_e^{k+1} + By_e^{k+1}\| = 0. \tag{8.11}$$

Taking limits on both sides of (8.4) along the subsequence $\{(x^{k_i}, y^{k_i}, z^{k_i})\}$, using (8.9), (8.11) and invoking the closedness of the graphs of ∂f and ∂g [3, Page 80], we obtain that

$$A^* z^\infty \in \partial f(x^\infty), \quad B^* z^\infty \in \partial g(y^\infty), \quad Ax^\infty + By^\infty - c = 0,$$

i.e., $(x^\infty, y^\infty, z^\infty)$ satisfies (8.3). From the discussion prior to the theorem, we conclude that (x^∞, y^∞) is an optimal solution to Problem (8.1) and z^∞ is a corresponding Lagrange multiplier.

Finally, we show that $(x^\infty, y^\infty, z^\infty)$ is actually the unique limit of $\{(x^k, y^k, z^k)\}$ and hence complete the proof. To this end, recall that $(x^\infty, y^\infty, z^\infty)$ satisfies (8.3). Hence, we could replace $(\bar{x}, \bar{y}, \bar{z})$ with $(x^\infty, y^\infty, z^\infty)$ in the above arguments, starting from (8.5). As a consequence, the subsequence $\{\|x_e^{k_i}\|_S^2 + \|y_e^{k_i}\|_T^2 + \lambda\|By_e^{k_i}\|^2 + \frac{1}{\lambda}\|z_e^{k_i}\|^2\}$ now converges to 0 as $i \to \infty$. Since this sequence is also non-increasing, we conclude that the whole sequence converges to zero, i.e.,

$$\lim_{k\to\infty} \|x_e^k\|_S^2 + \|y_e^k\|_T^2 + \lambda\|By_e^k\|^2 + \frac{1}{\lambda}\|z_e^k\|^2 = 0. \tag{8.12}$$

From this, we see immediately that $\lim_{k\to\infty} z^k = z^\infty$. Moreover, if g is strongly convex or $B^*B + T$ is positive definite, then we see from (8.12) and (8.9) that

$$\lim_{k\to\infty} \sigma_2\|y_e^{k+1}\| = 0 \ \text{ and } \ \lim_{k\to\infty} \|y_e^k\|_T^2 + \|By_e^k\|^2 = 0,$$

and hence $\lim_{k\to\infty} y^k = y^\infty$. On the other hand, if f is strongly convex or $A^*A + S$ is positive definite, then we see from (8.12), (8.8) and (8.9) that

$$\lim_{k\to\infty} \sigma_1\|x_e^{k+1}\| = 0 \quad \text{and} \quad \lim_{k\to\infty} \|x_e^k\|_S^2 + \|Ax_e^k\|^2 = 0,$$

and hence $\lim_{k\to\infty} x^k = x^\infty$. Therefore, we have shown that the whole sequence $\{(x^k, y^k, z^k)\}$ converges to $(x^\infty, y^\infty, z^\infty)$ under the assumption of the theorem. □

Remark 8.1. *Without the two terms* $\sigma_1\|x_e^{k+1}\|^2$ *and* $\sigma_2\|y_e^{k+1}\|^2$*, the inequality* (8.6) *is actually a special case of the one used in Theorem 1 in [25].*

Remark 8.2. *The conditions* (a)–(f) *for the convergence of* $\{(x^k, y^k, z^k)\}$ *in Theorem 8.1 can be replaced by assuming that the following two slightly stronger conditions hold:*

(i) f *is strongly convex, or* A *is injective, or* S *is positive definite;*

(ii) g *is strongly convex, or* B *is injective, or* T *is positive definite;*

except for the case that both A *and* B *are injective.*

Remark 8.3. *Theorem 8.1 provides general conditions for the convergence of the proximal ADM. It includes some recent results in the literature as special cases. For example, it has been considered by Attouch and Soueycatt [1] for the case* $S = \frac{1}{\lambda}I$ *and* $T = \frac{1}{\lambda}I$*. Moreover, Zhang et al. considered the case when* $B = I$ *and* S *is chosen to be positive definite in [56] , and established a slight weaker convergence result, that is, all the cluster points of* $\{(x^k, y^k)\}$ *and* $\{z^k\}$ *are optimal solutions to the primal and dual problems, respectively. Furthermore, Yang and Zhang [55] applied the proximal ADM to solve* l_1*-norm minimization problems in compressive sensing, allowing the step length to be chosen under some conditions rather than only* 1*. It is notable that when the step length is chosen to be 1, the iteration scheme in [55] simply reduces to* (8.2) *with* $A = I$ *and a symmetric and positive definite matrix* T*. The proof of the convergence in [55] is very similar to the one provided in this appendix but particularly for the* l_1*-norm minimization problems.*

REFERENCES

[1] H. Attouch and M. Soueycatt. Augmented Lagrangian and proximal alternating direction methods of multipliers in Hilbert spaces. Applications to games, PDE's and control. *Pac. J. Optim.* 5, pp. 17–37 (2009).

[2] D. Bertsekas and J. N. Tsitsiklis. *Parallel and Distributed Computation: Numerical Methods.* Prentice Hall (1989).

[3] J. M. Borwein and A. S. Lewis. *Convex Analysis and Nonlinear Optimization.* Springer, 2nd edition (2006).

[4] J. A. Cadzow. Signal enhancement – a composite property mapping algorithm. *IEEE T. Acoust. Speech* 36, pp 49–62 (1988).

[5] J.-F. Cai, E. J. Candés, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM J. Optim.* 20, pp. 1956–1982 (2010).

[6] E. J. Candés and Y. Plan. Tight oracle bounds for low-rank matrix recovery from a minimal number of random measurements. *IEEE Trans. Inf. Theory* 57, pp. 2342–2359 (2011).

[7] E. J. Candés and Y. Plan. Matrix completion with noise. *P. IEEE* 98, pp. 925–936 (2010)

[8] E. J. Candés and B. Recht. Exact matrix completion via convex optimization. *Found. Comput. Math.* 9, pp. 717–772 (2009).

[9] S. Ma, D. Goldfarb and L. Chen. Fixed point and Bregman iterative methods for matrix rank minimization. *Math. Program.* 128, pp. 321–353 (2011).

[10] B. De Moor. Total least squares for affinely structured matrices and the noisy realization problem. *IEEE Trans. Signal Process.* 42, pp. 3104–3113 (1994).

[11] B. De Moor, P. De Gersem, B. De Schutter, and W. Favoreel. DAISY: A database for the identification of systems. *Journal A* 38, pp. 4–5 (1997).

[12] B. De Schutter. Minimal state-space realization in linear system theory: an overview. *J Comput. Appl. Math.* 121, pp. 331–354 (2000).

[13] T. Ding, M. Sznaier, and O. I. Camps. A rank minimization approach to video inpainting. *Proc. IEEE Conf. Comput. Vision*, Rio de Janeiro, Brazil, pp. 1–8, Oct (2007).

[14] J. Eckstein. Some saddle-function splitting methods for convex programming. *Optim. Method Softw.* 4, pp. 75–83 (1994).

[15] J. Eckstein and D. P. Bertsekas. On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. *Math. Prog.* 55, pp. 293–318 (1992).

[16] M. Elad and P. Milanfar. Shape from moments – an estimation theory perspective. *IEEE Trans. Signal Process.* 52, pp. 1814–1829 (2004).

[17] M. Fazel. *Matrix Rank Minimization with Applications.* Ph. D. Thesis, Stanford University (2002).

[18] M. Fazel, H. Hindi and S. Boyd. Log-det heuristic for matrix rank minimization with applications to Hankel and Euclidean distance matrices. *Proc. American Control Conf.*, Denver, Colorado, June (2003).

[19] M. Fazel, H. Hindi and S. Boyd. A Rank Minimization Heuristic with Application to Minimum Order System Approximation. *Proc. American Control Conf.*, Arlington, Virginia, pp. 4734–4739, June (2001).

[20] D. Gabay. Applications of the method of multipliers to variational inequalities. In M. Fortin and R. Glowinski, eds., *Augmented Lagrangion Methods: Applications to the Solution of Boundary Problems.* North-Holland, Amsterdam, 1983.

[21] D. Gabay and B. Mercier. A dual algorithm for the solution of nonlinear variational problems via finite element approximations. *Computers and Mathematics with Applications.* 2, pp. 17-40 (1976).

[22] R. Glowinski and A. Marroco. Sur l'approximation, par elements finis d'ordre un, et la resolution, par penalisation-dualit'e, d'une classe de problemes de Dirichlet non lineares. *Revue Frrancaise d'Automatique, Informatique et Recherche Op'erationelle.* 9 (R-2), pp. 41-76 (1975).

[23] G. H. Golub, P. Milanfar and J. Varah. A stable numerical method for inverting shape from moments. *SIAM J. Sci. Comput.* 21, pp. 1222–1243 (1999).

[24] G. H. Golub and C. F. Van Loan. *Matrix Computation.* Johns Hopkins University Press, Baltimore, 3rd edition (1996).

[25] B. He, L. Liao, D. Han and H. Yang. A new inexact alternating directions method for monotone variational inequalities. *Math. Program.* 92, pp. 103–118 (2002).

[26] R. H. Keshavan, A. Montanari and S. Oh. Matrix completion from a few entries. *IEEE Trans. Inf. Theory* 56, pp. 2980–2998 (2010).

[27] J. B. Lasserre. Global optimization with polynomials and the problem of moments. *SIAM J. Optim.* 11, pp. 796–817 (2001).

[28] M. Laurent. Sums of squares, moment matrices and optimization over polynomials. In *Emerging Applications of Algebraic Geometry*, Vol. 149 of IMA Volumes in Mathematics and its Applications, M. Putinar and S. Sullivant (eds.), Springer, pp. 157–270 (2009).

[29] K. Lee and Y. Bresler. ADMiRA: atomic decomposition for minimum rank approximation. *IEEE Trans. Inf. Theory* 56, pp. 4402–4416 (2010).

[30] Y. Liu, D.F. Sun, and K.-C. Toh. An implementable proximal point algorithmic framework for nuclear norm minimization. *Mathematical Programming*, DOI: 10.1007/s10107-010-0437-8.

[31] Z. Liu and L. Vandenberghe. Interior-point method for nuclear norm approximation with application to system identification. *SIAM. J. Matrix Anal. A.* 31, pp. 1235–1256 (2009).

[32] Z. Liu and L. Vandenberghe. Semidefinite programming methods for system realization and identification. *Proc. 48th IEEE Conference on Decision and Control*, pp. 4676 – 4681 (2009).

[33] L. Ljung. *System Identification: Theory for the User.* Prentice Hall (1999).

[34] J. Mari, P. Stoica and T. McKelvey. Vector ARMA estimation: a reliable subspace approach. *IEEE Trans. Signal Process.* 48, pp. 2092–2104 (2000).

[35] I. Markovsky, J. C. Willems, S. Van Huffel and B. De Morr. *Exact and Approximate Modeling of Linear Systems: A Behavioral Approach.* SIAM (2006).

[36] I. Markovsky, J. C. Willems, S. Van Huffel, B. De Morr and R. Pintelon. Application of structured total least squares for system identification and model reduction. *IEEE T. Automat. Contr.* 50, pp. 1490–1500 (2005).

[37] R. Meka, P. Jain and I. S. Dhillon. Guaranteed rank minimization via singular value projection. *Proc. Neural Information Process. Systems Conf.*, pp. 937-945, December (2010).

[38] P. Milanfar, G. Verghese, W. Karl and A. Willsky, Reconstructing polygons from moments with connections to array processing. *IEEE Trans. Signal Process.* 43, pp. 432–443 (1995).

[39] K. Mohan and M. Fazel. Reweighted nuclear norm minimization with application to system identification. *Proc. American Control Conference* (2010).

[40] Y. Nesterov. A method for solving a convex programming problem with convergence rate $O(1/k^2)$. *Soviet Math. Dokl.* 27(2), pp. 372–376 (1983).

[41] Y. Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course.* Kluwer Academic Publishers (2003).

[42] Y. Nesterov. Excessive gap technique in nonsmooth convex minimization. *SIAM Journal on Optimization* 16, pp. 235–249 (2005).

[43] Y. Nesterov. Smooth minimization of non-smooth functions. *Math. Program.* 103, pp. 127–152 (2005).

[44] P. A. Parrilo. Semidefinite programming relaxations for semialgebraic problems. *Math. Program.* 96, pp. 293–320 (2003).

[45] T. K. Pong, P. Tseng, S. Ji and J. Ye. Trace norm regularization: reformulations, algorithms, and multi-task learning. *SIAM J. Optim.* 20, pp. 3465-3489 (2010).

[46] B. Recht, M. Fazel and P. Parrilo. Guaranteed minimum rank solutions of matrix equation via nuclear norm minimization. *SIAM Review* 52, pp. 471–501 (2010).

[47] R. T. Rockafellar. *Convex Analysis.* Princeton University Press, Princeton (1970).

[48] R. T. Rockafellar. and R. J-B. Wets. *Variational Analysis.* Springer (1998).

[49] E. D. Sontag. *Mathematical Control Theory: Deterministic Finite Dimensional Systems.* Second Edition, Springer (1998).

[50] K.-C. Toh and S. Yun. An accelerated proximal gradient algorithm for nuclear norm regularized least squares problems. *Pac. J. Optim.* 6, pp. 615–640 (2010).

[51] P. Tseng. On accelerated proximal gradient methods for convex-concave optimization. Submitted to *SIAM J. Optim.* (2008).

[52] P. Tseng. Approximation accuracy, gradient methods, and error bound for structured convex optimization. *Math. Program.* 125, pp. 263–295 (2010).

[53] J. C. Willems. From time series to linear system – part I. finite dimensional linear time invariant systems. *Automatica* 22, pp. 561–580 (1986).

[54] J. C. Willems. From time series to linear system – part II. exact modelling. *Automatica* 22, pp. 675–694 (1986).

[55] J. Yang and Y. Zhang. Alternating direction algorithms for ℓ_1-problems in compressive sensing. *SIAM J. Sci. Comput.* 33, pp. 250–278 (2011).

[56] X. Zhang, M. Burger and S. Osher. A unified primal-dual algorithm framework based on Bregman iteration. *J. Sci. Comput.* 46, pp. 20–46 (2011).