An SVD-free Approach to a Class of Structured Low Rank Matrix Optimization Problems with Application to System Identification

Marco Signoretto[1] and Volkan Cevher[2] and Johan A. K. Suykens[1]

***Abstract*— We present an algorithm to solve a structured low rank matrix optimization problem based on the nuclear norm. We represent the desired structure by a linear map, termed mutation, that we characterize and use in our algorithm. Contrary to alternative techniques for structured low rank matrices, the algorithm is SVD-free, which leads to improved scalability. The idea relies on restating the nuclear norm via an equivalent variational reformulation involving explicit matrix factors. We detail the procedure for a general class of problems and then discuss its application to linear system identification with input and output missing data. A direct comparison between alternative approaches highlights the advantage of SVD-free computations.**

I. Introduction

Various rank minimization problems recently attracted renewed interest in the technical literature. Applications include recommendation systems, multi-task learning, system identification and realization techniques. In all these settings, general notions of model complexity can be conveniently expressed by the rank of an appropriate matrix. In particular, nuclear norm optimization methods for low-rank matrix approximation have been discussed in several recent papers. In the system identification community the idea was first proposed in [7], [6]. The nuclear norm is the largest convex lower bound of the rank function on the spectral unit ball [7]; this fact motivates the use of the nuclear norm in convex relaxation of rank-based problems.

In this paper, we focus on problems where we need to find a matrix that, in addition to being low-rank, is required to have entries partitioned into known disjointed groups. This setting includes various type of structured matrices such as Hankel, Toeplitz and circulant matrices. Generally speaking, it allows to deal with matrices that have reduced degrees of freedom. Our interest arises in particular from concatenated block-Hankel matrices that appear in formulations for input-output linear system identification problems with noisy and/or partially unobserved data [12].

A. Main Contributions

Existing algorithms for nuclear-norm based system identification typically involve at each iteration the singular value decomposition (SVD) of a structured matrix $Y = \mathcal{B}(x) \in \mathbb{R}^{M\times N}$, with $M \leq N$. The cost of this step usually grows as $O(M^2N)$ which hinders the application to large scale problems. The main contribution of this paper is to present an SVD-free solution strategy to a class of structured low rank matrix optimization problems. The desired structure is represented by a class of linear maps $\mathcal{B} : \mathbb{R}^P \to \mathbb{R}^{M\times N}$, termed *mutations*, that we introduce and characterize. We then discuss application to linear system identification problems with noisy and partially unobserved input-output data.

B. Outline and Notation

We conclude this section by presenting our notation. Section II deals with the general problem statement and presents the main idea behind the SVD-free approach. In Section III we characterize mutation operators and show how the definitions can be implemented for general problems. In Section IV we present our solution strategy based on an augmented Lagrangian approach. In Section V we recall an existing approach to linear system identification with missing data and discuss application of our SVD-free approach in contrast to an SVD-based technique. In Section VI we presents experiments. We end the paper by drawing concluding remarks in Section VII.

For a positive integer L we write $\mathbb{N}_L$ to denote the set $\{1,\ldots,L\}$. For a generic matrix $A \in \mathbb{R}^{M\times N}$ we denote by a_{mn} the entry indexed by (m,n); we write a_m to mean the $m-$th column. We write I_L to mean the $L \times L$ identity matrix. For a vector $r \in \mathbb{R}^L$ we denote by $\mathrm{diag}(r)$ the $L \times L$ diagonal matrix satisfying $\mathrm{diag}(r)_{pp} = r_p$, $p \in \mathbb{N}_L$; $\|r\|$ is the Euclidean norm of r. For arbitrary matrices $A, B \in \mathbb{R}^{M\times N}$ the (canonical) inner product is defined as:

$$\langle A, B\rangle = \mathrm{trace}(A^\top B) = \sum_{n\in\mathbb{N}_N}\sum_{m\in\mathbb{N}_M} a_{mn}b_{mn} \ . \quad (1)$$

The corresponding Hilbert-Frobenius norm is $\|A\|_F = \sqrt{\langle A, A\rangle}$. If a matrix A has rank R, then the singular value decomposition (SVD) of A is

$$A = U\Sigma V^\top, \quad \Sigma = \mathrm{diag}(\sigma) \quad (2)$$

where $U \in \mathbb{R}^{M\times R}$ and $V \in \mathbb{R}^{N\times R}$ are matrices with orthonormal columns, and the vector of singular values σ satisfies

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_R > 0 \ .$$

The spectral norm of A is $\|A\|_2 = \sigma_1$. Finally, we denote general linear operators (i.e., operators without explicit matrix representation) by calligraphic letters ($\mathcal{A}, \mathcal{B}, \ldots$). In

[1]M. Signoretto and J. A. K. Suykens are with ESAT-SCD/SISTA Katholieke Universiteit Leuven, Kasteelpark Arenberg 10, B-3001 Leuven (BELGIUM). Email: marco.signoretto@esat.kuleuven.be and johan.suykens@esat.kuleuven.be.

[2] V. Cevher is with the Laboratory for Information and Inference Systems (LIONS), Ecole Polytechnique Federale de Lausanne (EPFL), EPFL STI IEL LIONS ELD 243 Station 11, CH-1015 Lausanne (SWITZERLAND). Email: volkan.cevher@epfl.ch.

particular, for an operator $\mathcal{A} : \mathbb{R}^L \to \mathbb{R}^{M\times N}$ we denote by $\mathcal{A}^*$ its adjoint, i.e., the operator satisfying:

$$\langle \mathcal{A}(y), X\rangle = \langle y, \mathcal{A}^*(X)\rangle \ .$$

II. Problem Statement

A. Main Problem Formulation

For a generic matrix $A \in \mathbb{R}^{M\times N}$ with SVD (2), the *nuclear norm* (a.k.a. trace norm or Schatten-1 norm) is defined as:

$$\|A\|_* = \sum_{r\in\mathbb{N}_R} \sigma_r \tag{3}$$

and is a convex function of A. It can be shown that $\|A\|_*$ is the *convex envelope*[1] on the spectral unit ball of the non-convex function $\mathrm{rank}(A)$ [6]. This makes $\|A\|_*$ a suitable replacement for $\mathrm{rank}(A)$ for a large class of problems that aim at reconstructing a low rank matrix from data.

In this paper, we focus on the situation where we need to find a matrix that, in addition to being low-rank, is required to have entries partitioned into known disjointed groups. For a significant class of problems this task can be accomplished solving the optimization problem:

$$\min_{x\in\mathbb{R}^L} \frac{1}{2}(x-a)^\top H(x-a) + \|\mathcal{B}(x)\|_* \tag{4}$$

for given vector $a \in \mathbb{R}^L$ and positive definite matrix $H \in \mathbb{R}^L \times \mathbb{R}^L$. This formulation regards a structured matrix as the output of a linear map $\mathcal{B} : \mathbb{R}^L \to \mathbb{R}^{M\times N}$, the characterization of which is postponed to Section III. Without loss of generality, we assume in the following that $M \le N$. The quadratic term usually plays the role of a data fitting measure and can subsume a trade-off parameter λ. A conventional choice is, in particular, $H = \lambda\, I_L$. In this setting (4) is a convex problem which implies that converging algorithms are guaranteed to deliver a global solution.

B. Solution Approaches based on SVDs

Consider a matrix $A \in \mathbb{R}^{M\times N}$ with SVD (2). It is well known that the *proximity operator* [1] of the scaled nuclear norm, i.e.,

$$\mathcal{P}_\tau(A) := \arg\min_{Y\in\mathbb{R}^{M\times N}} \frac{1}{2}\|Y-A\|_F^2 + \tau\|Y\|_* \tag{5}$$

corresponds to *soft-thresholding* the singular values [4]:

$$\mathcal{P}_\tau(A) = U\Sigma_+ V^\top, \quad \Sigma_+ = \mathrm{diag}(\{\max(\sigma_r-\tau, 0)\}_{1\le r\le R}) \ . \tag{6}$$

This fact has been used to derive various solution strategies for problem (4), usually starting from an equivalent constrained reformulation with separable objective function:

$$\begin{aligned} \min_{x,Y} \quad & \tfrac{1}{2}(x-a)^\top H(x-a) + \|Y\|_* \\ \text{subject to} \quad & \mathcal{B}(x) = Y \ . \end{aligned} \tag{7}$$

Many existing algorithms, in fact, involves at iteration k the soft-thresholding of the singular values of an $M \times N$ matrix to determine the current estimate $Y^{(k)}$. The main difficulty with this approach relies on the complexity of the SVD: its cost typically grows as $O(M^2N)$ [10, Page 254] which hinders the application to large scale problems.

C. SVD-free Solution Approach

In order to avoid computing SVDs we rely on a known variational characterization of the nuclear norm. It can be shown that the following holds true [17, Lemma 8]:

$$\|Y\|_* = \min_{U,V \ :\ Y=UV^\top} \frac{1}{2}\left(\|U\|_F^2 + \|V\|_F^2\right) \tag{8}$$

where the size of the matrices U and V is not constrained. This suggests the following restatement of problem (4):

$$\begin{aligned} \min_{x,U,V} \quad & \tfrac{1}{2}(x-a)^\top H(x-a) + \tfrac{1}{2}\left(\|U\|_F^2 + \|V\|_F^2\right) \\ \text{subject to} \quad & \mathcal{B}(x) = UV^\top \ . \end{aligned} \tag{9}$$

Note that this problem is non-convex due to the product between U and V. Nonetheless, it is possible to show that if (x_*, U_*, V_*) is a solution to (9), x_* is a solution to (4) and $\mathcal{B}(x_*) = U_*V_*^\top$. A similar result is found in [15, Section 5.3], which discusses an approach based on (8) to solve the unstructured low rank matrix problem:

$$\begin{aligned} \min_{X} \quad & \|X\|_* \\ \text{subject to} \quad & \mathcal{A}(X) = b \ . \end{aligned} \tag{10}$$

Whereas [15] considers factors with reduced size, in this paper we focus on full factors, i.e., we do not constrain the size of the matrices U and V.

Problem (9) has a differentiable objective function. It can be tackled via a gradient-based solution strategy after embedding the constraint $\mathcal{B}(x) = UV^\top$ by an augmented Lagrangian approach. Details of our approach are given in Section IV. Note that, in contrast, the objective of problem (7) is non-smooth. Correspondingly, computing a solution to (7) requires a sub-gradient approach or an operator splitting technique [1]. This, ultimately, leads to singular values soft-thresholding.

III. Mutations

In this section we introduce a class of operators, termed mutations, giving rise to structured matrices with Hankel, Toeplitz and circulant matrices as special cases. More generally, mutations can encode matrices having entries partitioned into known disjoined groups. We provide formal definitions and then show how these lead to practical "quick and dirty" implementations.

A. Main Definitions

We begin by considering a family of sets $\mathscr{P} = \{\mathscr{P}_1, \mathscr{P}_2, \ldots, \mathscr{P}_L\}$ constituting a partition for the set of entries of an $M \times N$ matrix. That is, we have:

$$\mathscr{P}_l = \left\{(m_1^l, n_1^l), (m_2^l, n_2^l), \ldots, (m_{K_l}^l, n_{K_l}^l)\right\} \ \text{ for } l \in \mathbb{N}_L \tag{11}$$

[1] I.e., the largest convex underestimator.

with $\mathscr{P}_l \neq \emptyset$ for any $l \in \mathbb{N}_L$, $\mathscr{P}_l \cap \mathscr{P}_k = \emptyset$ for $l \neq k$ and $\cup_{l\in\mathbb{N}_L} \mathscr{P}_l = \mathbb{N}_M \times \mathbb{N}_N$. We now introduce the membership function ι:

$$\begin{array}{rccl} \iota: & \mathbb{N}_M \times \mathbb{N}_N & \to & \mathbb{N}_L \\ & (m,n) & \mapsto & \{l \in \mathbb{N}_L \ : \ (m,n) \in \mathscr{P}_l\} \, . \end{array} \tag{12}$$

It is easy to check that ι is a well defined surjective function. Equipped with these definitions we introduce the *mutation operator*, or simply *mutation*, mapping entries of a vector into multiple entries of matrix.

Definition 1 (Mutation):

$$\begin{array}{rccl} \mathcal{B}_{\mathscr{P}}: & \mathbb{R}^L & \to & \mathbb{R}^{M\times N} \\ & x & \mapsto & \left(x_{\iota(m,n)} \ : \ (m,n) \in \mathbb{N}_M \times \mathbb{N}_N\right) \, . \end{array} \tag{13}$$

In the following we drop the subscript $\mathscr{P}$ in $\mathcal{B}_{\mathscr{P}}$ for ease of notation. The operator $\mathcal{B}$ is linear; indeed we have $(\mathcal{B}(x+y))_{mn} = (x+y)_{i(m,n)} = x_{i(m,n)} + y_{i(m,n)}$. Its adjoint is:

$$\begin{array}{rccl} \mathcal{B}^*: & \mathbb{R}^{M\times N} & \to & \mathbb{R}^L \\ & C & \mapsto & \left(\sum_{(m,n)\in\mathcal{P}_l} c_{mn} \ : \ l \in \mathbb{N}_L\right) \, . \end{array} \tag{14}$$

To see this it suffices to observe that we have:

$$\begin{aligned} \langle \mathcal{B}^*(C), x\rangle = \sum_{l\in\mathbb{N}_L} \left\{ x_l \sum_{(m,n)\in\mathcal{P}_l} c_{mn} \right\} = \\ \sum_{l\in\mathbb{N}_L} \left\{ \sum_{(m,n)\in\mathcal{P}_l} c_{mn} x_{\iota(m,n)} \right\} = \\ \sum_{m\in\mathbb{N}_M} \sum_{n\in\mathbb{N}_N} c_{mn} (\mathcal{B}(x))_{mn} = \langle C, \mathcal{B}(x)\rangle \, . \end{aligned} \tag{15}$$

Recall from (11) that K_l is the cardinality of $\mathcal{P}_l$; by equation (13) and (14) we have:

$$\begin{aligned} (\mathcal{B}^*(\mathcal{B}(x)))_l = \sum_{(m,n)\in\mathcal{P}_l} (\mathcal{B}(x))_{mn} = \\ \sum_{(m,n)\in\mathcal{P}_l} x_{\iota(m,n)} = K_l x_l \text{ for any } l \in \mathbb{N}_L \, . \end{aligned} \tag{16}$$

Therefore we see that the operator $\mathcal{B}^*\mathcal{B}$ has matrix representation:

$$G_{\mathcal{B}} = \operatorname{diag}(K_1, K_2, \ldots, K_L) \, . \tag{17}$$

B. A Concrete Example: Hankel Operators

Consider a (block) Hankel operator $\mathcal{H}_Q : \mathbb{R}^{PT} \to \mathbb{R}^{(Q+1)P\times(T-Q)}$ defined, for $x(t) \in \mathbb{R}^P, \ t \in \mathbb{N}_T$, by:

$$\mathcal{H}_Q: \begin{bmatrix} x(1) \\ x(2) \\ \vdots \\ x(T) \end{bmatrix} \mapsto \begin{bmatrix} x(1) & x(2) & \cdots & x(T-Q) \\ x(2) & x(3) & \cdots & x(T-Q+1) \\ \vdots & \vdots & \ddots & \vdots \\ x(Q+1) & x(Q+2) & \cdots & x(T) \end{bmatrix} . \tag{18}$$

Since entries of the matrix $\mathcal{H}_Q(x)$ are partitioned into disjoint groups, $\mathcal{H}_Q(x)$ can be regarded as the output of a mutation operator. If we consider the scalar case $P = 1$, in particular, one can see that the corresponding mutation is defined upon the partition of entries corresponding to skew-diagonals. If we let $Q = T/2$, then:

$$\mathscr{P}_l := \begin{cases} \{(l,1),(l-1,2),\ldots,(1,l)\} & \text{if } l \leq Q \\ \{(l-Q+1,Q),(l-Q+2,Q-1),\ldots,(Q+1,l-Q)\} & \text{if } l > Q \, . \end{cases} \tag{19}$$

In this case, the membership function ι is given by:

$$\begin{array}{rccl} \iota: & \mathbb{N}_{Q+1} \times \mathbb{N}_{T-Q} & \to & \mathbb{N}_T \\ & (m,n) & \mapsto & m+n-1 \, . \end{array} \tag{20}$$

Correspondingly, one has:

$$\mathcal{B}: \quad x \quad \mapsto \quad (x_{m+n-1} \ : \ (m,n) \in \mathbb{N}_{Q+1} \times \mathbb{N}_{T-Q}) \tag{21}$$

$$\mathcal{B}^*: \quad C \quad \mapsto \quad \left(\sum_{m+n-1=t} c_{mn} \ : \ t \in \mathbb{N}_T \right) . \tag{22}$$

C. Implementation via Linear Indexing

In practice, rather than indexing entries by pairs of indices, as in equation (11), it is usually more convenient to use a single (linear) index. In MATLAB, for example, the mutation (forward) operator can be implemented as follows:

```
function Y=forwardOp(x,linSets,sizeY)
    Y=zeros(sizeY);
    for i=1:numel(linSets)
        Y(linSets{i})=x(i);
    end
```

where `linSets` is a cell array containing the linear indices for the $l-$th group $\mathscr{P}_l$ and `sizeY` is $2-$dimensional vector containing the dimensions of the output matrix. In turn, its adjoint (backward) operator can be implemented as follows:

```
function y=backwardOp(X,linSets)
    y=zeros(numel(linSets),1);
    for i=1:numel(linSets)
        y(i)=sum(X(linSets{i}));
    end
```

This applies to a general partition $\mathscr{P}$. Efficient implementations can be given for matrices with more specialized structure (e.g., Hankel structure).

IV. SVD-free Algorithm via Augmented Lagrangian Approach

Armed with the definition of mutation and its adjoint we are now ready to discuss the proposed augmented Lagrangian approach (a.k.a. *alternating direction method of multipliers*, ADMM) to (9). A full derivation of the general method can be found in classical books, e.g., [2]. A recent review with focus on optimization problems arising in machine learning and statistics is found in [3]. The effectiveness of the approach on nuclear norm problems arising in system identification and realization has been demonstrated in [8], along with several other first-order methods.

A. Main Iteration

The augmented Lagrangian for problem (9) reads:

$$\begin{aligned} L_\mu(x,U,V;Z) = \tfrac{1}{2}(x-a)^\top H(x-a) + \tfrac{1}{2}\left(\|U\|_F^2 + \|V\|_F^2\right) \\ + \left\langle Z, \mathcal{B}(x) - UV^\top \right\rangle + \tfrac{\mu}{2} \left\| \mathcal{B}(x) - UV^\top \right\|_F^2 \end{aligned} \tag{23}$$

Each iteration of the proposed approach consists now of successive minimization of L_μ over x, U and V followed by a simple update of the dual variable Z:

$$\begin{cases} x^{(k+1)} := \arg\min_x L_\mu(x, U^{(k)}, V^{(k)}; Z^{(k)}) & \text{(24a)} \\ U^{(k+1)} := \arg\min_U L_\mu(x^{(k+1)}, U^{(k)}, V^{(k)}; Z^{(k)}) & \text{(24b)} \\ V^{(k+1)} := \arg\min_V L_\mu(x^{(k+1)}, U^{(k+1)}, V^{(k)}; Z^{(k)}) & \text{(24c)} \\ Z^{(k+1)} := Z^{(k)} + \mu\left(\mathcal{B}(x^{(k+1)}) - U^{(k+1)} V^{(k+1)\top}\right) . & \text{(24d)} \end{cases}$$

Note that computing a new estimate for the primal variables $x,\ U$ and V in step (24a), (24b) and (24c) involves simply the solution of a set of linear equations since L_μ is quadratic in $x,\ U$ and V, respectively. This is to be contrasted with the ADMM approach proposed in [12] to solve problem (7). In fact, the augmented Lagrangian $M_\mu(x, Y)$ corresponding to problem (7) is quadratic only in x; solving $\min_Y M_\mu(x^{(k+1)}, Y)$, on the other hand, amounts at evaluating the soft thresholding operator (6) and therefore requires computing the SVD of an $M \times N$ matrix.

B. Detailed Steps

By letting $\partial L_\mu / \partial x = 0$, one sees that step (24a) requires the solution in x of:

$$(H + \mu G_{\mathcal{B}})x = Ha + \mathcal{B}^* \left(\mu U^{(k)} V^{(k)\top} - Z^{(k)} \right) \quad (25)$$

where Ha can be precomputed. Recall the definition of $G_{\mathcal{B}}$ in (17). When $H = \lambda\, I_L$, (25) becomes simply:

$$x = q \odot (Ha) + q \odot \left(\mathcal{B}^* \left(\mu U^{(k)} V^{(k)\top} - Z^{(k)} \right) \right) \quad (26)$$

where $\odot$ denotes Hadamard product and $q \in \mathbb{R}^L$ is defined entry-wise by: $q_l = 1/(\lambda + \mu K_l),\ l \in \mathbb{N}_L$. Likewise, by letting $\partial L_\mu / \partial U = 0$ one obtains for step (24b):

$$U \left(I + \mu\, V^{(k)\top} V^{(k)} \right) = \left(\mu\, \mathcal{B} \left(x^{(k+1)} \right) + Z^{(k)} \right) V^{(k)} , \quad (27)$$

which can be solved in U without explicitly computing the inverse of $I + \mu\, V^{(k)\top} V^{(k)}$. In a similar fashion one obtains that step (24c) amounts at solving in V:

$$V \left(I + \mu\, U^{(k+1)\top} U^{(k+1)} \right) = \left(\mu\, \mathcal{B} \left(x^{(k+1)} \right) + Z^{(k)} \right)^\top U^{(k+1)} . \quad (28)$$

C. Optimality, Stopping Criterion and Convergence

In order to test for optimality, we rely on the equivalence between problem (7) and (9). That is, we let $Y^{(k)} = U^{(k)} V^{(k)\top}$ and check whether $(x^{(k)}, Y^{(k)})$ is optimal with respect to problem (7). We follow [12] and define the residuals and tolerances in the stopping criterion as:

$$R_p = \mathcal{B} \left(x^{(k)} \right) - Y^{(k)} \quad (29)$$

$$r_d = \mu\, \mathcal{B}^* \left(Y^{(k-1)} - Y^{(k)} \right) \quad (30)$$

$$\epsilon_p = \sqrt{MN}\epsilon_{\text{abs}} + \epsilon_{\text{rel}} \max \left\{ \left\| \mathcal{B} \left(x^{(k)} \right) \right\|_F , \left\| Y^{(k)} \right\|_F \right\} \quad (31)$$

$$\epsilon_d = \sqrt{L}\epsilon_{\text{abs}} + \epsilon_{\text{rel}} \left\| \mathcal{B}^*(Z) \right\|_2 \quad (32)$$

where we let $\epsilon_{\text{rel}} = 10^{-3},\ \epsilon_{\text{abs}} = 10^{-6}$. We then terminate if $\|R_p\|_F \leq \epsilon_p$ and $\|r_d\| \leq \epsilon_d$. Rather than using a fixed penalty parameter μ, we go along with the common prescription of adapting μ to improve convergence, see, e.g., [3] and reference therein. Specifically, we follow [11], [12] and adapt μ at the end of each iteration by letting:

$$\mu = \begin{cases} \tau\mu & \text{if } \|R_p\|_F > \nu \|r_d\| \\ \mu/\tau & \text{if } \|r_d\| > \nu \|R_p\|_F \\ \mu & \text{otherwise .} \end{cases} \quad (33)$$

In this scheme one has $\nu, \tau > 1$; we took $\nu = 10$ and $\tau = 2$, as in [12].

There are many convergence results discussed in the literature for algorithms based on augmented Lagrangians. In particular, standard arguments apply to the ADMM approach in [12] for the convex problem (7). The same arguments do not immediately apply to our algorithm for the non-convex problem (9). Nevertheless our experiments show that the approach described above always lead (up to numerical precision) to an optimal solution (X^*, U^*, V^*) to problem (9) and, therefore, to an optimal solution $(x^*, U^* V^{*\top})$ to problem (7).

V. Linear System Identification by Nuclear Norm Optimization

Applications of the nuclear norm heuristic in system identification and realization have been extensively studied in the literature, among others in [13], [14], [8], [9]. Here we summarize the main ideas, including difficulties with missing data, and then turn to the formulation of interest.

Consider a state-space linear time invariant system driven by an external input $w(t) \in \mathbb{R}^P$:

$$\begin{cases} x(t+1) &= Ax(t) + Bw(t) \\ y(t) &= Cx(t) + Dw(t) . \end{cases} \quad (34)$$

In the latter, A, B, C, D are the system matrices with the appropriate dimensions, $y(t) \in \mathbb{R}^M$ and $x(t) \in \mathbb{R}^S$ is the latent state vector, with S being the order of the system.

A. Subspace-based Identification

It is not difficult to see that the recursive equations (34) lead to the matrix equation:

$$\mathcal{H}_Q(y) = \Gamma X + H \mathcal{H}_Q(w), \quad (35)$$

where

$$\Gamma = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^Q \end{bmatrix},\ H = \begin{bmatrix} D & 0 & 0 & \dots & 0 \\ CB & D & 0 & \dots & 0 \\ CAB & CB & D & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{Q-1}B & CA^{Q-2}B & CA^{Q-3}B & \dots & D \end{bmatrix},$$

and $X = [x_1,\ x_2,\ \dots,\ x_{T-R}]$. By right-multiplying both sides of (35) by the orthogonal complement of $\mathcal{H}_Q(w)$, denoted by $W^\perp$, one obtains:

$$\mathcal{H}_Q(y) W^\perp = \Gamma X W^\perp . \quad (36)$$

It can be shown, under mild conditions on w and X, that this matrix has rank S equal to the order of the system, see, e.g., [19, Section 3.3]. Since $\mathcal{H}_R(y) W^\perp$ does not depends either upon the system matrices or the states, it follows that one can estimate the order of the system solely based on input-output measurements:

$$w_{\text{m}} = \begin{bmatrix} w_m(1) \\ w_m(2) \\ \vdots \\ w_m(T) \end{bmatrix} \in \mathbb{R}^{PT},\ y_{\text{m}} = \begin{bmatrix} y_m(1) \\ y_m(2) \\ \vdots \\ y_m(T) \end{bmatrix} \in \mathbb{R}^{MT} . \quad (37)$$

This fact has found applications in *subspace-based identification methods*, see, e.g., [5], [18]. In the simplest case one works directly with $\mathcal{H}_Q(y_{\rm m})W_{\rm m}^{\perp}$ where $W_{\rm m}^{\perp}$ denotes the orthogonal complement constructed based upon observed inputs. Due to noise and model's errors, however, this matrix is usually not low rank. This fact makes it difficult to estimate the model order and to find the system matrices.

B. Identification by Nuclear Norm Optimization

To overcome this issue, [13] proposed the following convex optimization problem based on the nuclear-norm:

$$\hat{y} = \arg \min_{y \in \mathbb{R}^{MT}} \lambda \left\| y - y_{\rm m} \right\|^2 + \| \mathcal{H}_Q(y) W_{\rm m}^{\perp} \|_* \ . \qquad (38)$$

In general, $\mathcal{H}_Q(\hat{y})U_{\rm m}^{\perp}$ is (close to) rank-deficient. This makes it easier to determine the model order and to find the system matrices, see [13]. Notably, similar ideas can be used to derive penalties for complex dynamical behaviours beyond system identification; [16], for instance, considers a similar approach for multi-period portfolio optimization.

C. Nuclear Norm Optimization: the Case of Missing Data

Problem (38) can be easily modified to account for missing data in the output measurements $y_{\rm m}$; it is not easily extended, however, to handle missing data in the measured inputs $w_{\rm m}$. This is because $W_{\rm m}^{\perp}$ in the objective function depends nonlinearly on $w_{\rm m}$. To cope with missing input and output observations, [12] proposed the following formulation:

$$\min_{w,y} \frac{\lambda_1}{2} \sum_{i \in \mathscr{M}_w} (w_i - w_{m\,i})^2 + \frac{\lambda_2}{2} \sum_{i \in \mathscr{M}_y} (y_i - y_{m\,i})^2 + \| F(w,y) \|_* \qquad (39)$$

where $w \in \mathbb{R}^{PT}$, $y \in \mathbb{R}^{MT}$, and $\mathscr{M}_u \subset \mathbb{N}_{PT}$ (resp. $\mathscr{M}_y \subset \mathbb{N}_{MT}$) is a set of indices of observed inputs (resp. outputs); $F(w,y)$ is obtained stacking block-Hankel matrices:

$$F(w,y) = \left[\begin{array}{c} \mathcal{H}_Q(w) \\ \mathcal{H}_Q(y) \end{array} \right] \in \mathbb{R}^{(P+M)(Q+1) \times (T-Q)} \ . \qquad (40)$$

The algorithm is motivated by the fact that, under mild conditions, the generating model (34) implies that:

$$\operatorname{rank}(F(w,y)) = S + \operatorname{rank}(\mathcal{H}_Q(w)) \qquad (41)$$

see, e.g., [19, Section 3.3].

D. Identification with Missing Data as a Structured Low Rank Matrix Optimization Problem

By letting $x = [w^{\top}, y^{\top}]^{\top}$ it is immediate to recognize that $F(w,y) = \mathcal{B}(x)$, for a suitably defined mutation $\mathcal{B}$. Now let $S_w \in \mathbb{R}^{PT \times PT}$, $S_y \in \mathbb{R}^{MT \times MT}$ be diagonal matrices with:

$$(S_w)_{ii} = \left\{ \begin{array}{ll} 1, & \text{if } i \in \mathscr{M}_w \\ 0, & \text{otherwise} \end{array} \right. , \ (S_y)_{ii} = \left\{ \begin{array}{ll} 1, & \text{if } i \in \mathscr{M}_y \\ 0, & \text{otherwise} \end{array} \right. .$$

By simply taking:

$$H_\lambda = \left[\begin{array}{cc} \lambda_1 S_w & O \\ O^{\top} & \lambda_2 S_y \end{array} \right] \ \text{ and } \ a = \left[\begin{array}{c} w_{\rm m} \\ y_{\rm m} \end{array} \right]$$

where $O \in \mathbb{R}^{PT \times MT}$ denotes a matrix of zeros, we can restate problem (39) as the structured low rank matrix optimization problem introduced in Section II:

$$\min_{x \in \mathbb{R}^{(P+M)T}} \frac{1}{2}(x-a)^{\top} H_\lambda (x-a) + \| \mathcal{B}(x) \|_* \ . \qquad (42)$$

E. SVD-based and SVD-free Approaches

The ADMM approach proposed in [12] tackles problem (42) by finding a solution to the equivalent constrained problem formulation (7). In the present setting, this requires at every step the SVD of a $(P+M)(Q+1) \times (T-Q)$ matrix. The computational burden then depends upon the embedding dimension Q as well as the actual dimensionality P and M of the input and output vectors $u(t)$ and $y(t)$, respectively. In contrast, the SVD-free solution approach considers a reformulation of the type (9). As detailed in Section IV, this leads to replace the SVD step by the update equations (27) and (28) necessary to find new estimates for the matrix factors U and V. A solution (x^*, U^*, V^*) to (7) is then readily linked to a solution $(\tilde{x}, \tilde{Y})$ of (9) by letting $\tilde{x} := x^*, \ \tilde{Y} := U^* V^{*\top}$.

VI. Experiments

The purpose of this Section is to compare a MATLAB implementation of the SVD-free approach detailed in Section IV to a MATLAB implementation of the algorithm proposed in [12] for linear system identification with missing data [2]. Our focus is on the following metrics:

obj val :	attained value in problem (42)
feasibility :	primal feasibility $\|Y - \mathcal{B}(x)\|_F / \|Y\|_F$
model fit :	averaged identification performance
CPU time :	time in seconds used by the process .

When evaluating the model fit we considered the same measure as in [12]: we took $\text{fit} = 100(1 - \|\hat{z} - z\| / (z - \text{mean}(z)))$ where z represents a sequence of a scalar input or output and $\hat{z}$ is the corresponding estimate. We then reported the average fit across the whole set of P input and M output sequences. A fair comparison is ensured by the use of the same stopping criterion across the two methods, see Section IV-C.

A. Experimental Setting

We considered the same experimental setting as in [8, Section 4.1.1]. That is, we generated random inputs $u(t) \in \mathbb{R}^P, \ t \in \mathbb{N}_T$ with standard Gaussian entries and let S be the true order of the system. We then generated matrices $A \in \mathbb{R}^{S \times S}$, $B \in \mathbb{R}^{S \times P}$, $C \in \mathbb{R}^{M \times S}$ and $D \in \mathbb{R}^{M \times P}$ with i.i.d. standard Gaussian entries, and normalize them to have spectral norm equal to one; the initial state vector $x(1) \in \mathbb{R}^S$ was also generated with standard Gaussian entries. The output $y(t) \in \mathbb{R}^M, \ t \in \mathbb{N}_T$ was then obtained according to the state-space model (34). We then added zero-mean Gaussian noise with variance $\sigma^2 = 0.01$ to produce

[2] The MATLAB code for the latter was downloaded from `http://www.zhang-liu.com/software/simio.html`.

TABLE I

RESULTS FOR $V\% = 20,\ P = 2,\ O = 3$.

	obj val	feasibility (10^{-3})	model fit	CPU time
	$M = 5,\ T = 1500$			
SVD-free	1016.79	0.30	79	0.67
SVD-based	1017.36	0.87	79	5.61
	$M = 15,\ T = 1500$			
SVD-free	1166.05	0.51	75	1.96
SVD-based	1165.71	0.30	75	8.65
	$M = 20,\ T = 4000$			
SVD-free	2079.09	0.22	76	6.78
SVD-based	2081.47	0.86	76	47.01
	$M = 40,\ T = 4000$			
SVD-free	2501.61	0.39	70	22.32
SVD-based	2501.76	0.37	70	120.16
	$M = 50,\ T = 10000$			
SVD-free	4803.46	0.30	67	111.28
SVD-based	4803.18	0.92	67	825.77

the final input and output measurements vectors y_m and u_m, defined as in (37). Finally we picked uniformly at random a set of observed entries $\mathscr{M}_u$ and $\mathscr{M}_y$; the cardinality of the set of missing entries was a fraction $V\%$ of the total number $(P + M)T$ of generated input and output measurements.

B. Results

Our goal is to compare alternative algorithms across different regimes rather than achieve the best fit. Therefore we did not tune the values of the regularization parameters λ_1 and λ_2 in (39); we simply took $\lambda_1 = \lambda_2 = 1$ and perform multiple experiments for different values of $S,\ P,\ T,\ M,\ O$ and $V\%$. In the experiments both the algorithms gave as output low rank solutions that were comparable in terms of objective value, feasibility and model fit. In Table 1 we reported the result of a set of representative experiments in which we kept fixed $O,\ P,\ S,\ V\%$ and varied M, T. In all the cases we took $Q = 7$. All results were obtained averaging over 20 Monte Carlo runs; in each run we kept fixed the system matrices and generated at random a new initial state and input sequence.

VII. CONCLUSIONS

We have presented an SVD-free algorithm for a class of structured low rank matrix optimization problems. Specifically, we have considered matrices having entries partitioned into known disjoint groups. This type of structured matrices were regarded as the output of a class of operators, termed mutations, that we defined and characterized. This setting includes, as a special case, concatenated block-Hankel matrices that appear in formulations for input-output linear system identification with missing data. We then focused on this task and compared an existing SVD-based algorithm with the proposed SVD-free approach; the experiments show that the proposed algorithm delivers low rank solutions with comparable quality (in terms of objective value, feasibility and model fit) at a substantially lower computational price than the existing SVD-based algorithm.

ACKNOWLEDGMENT

Research supported by Research Council KUL: GOA/10/09 MaNet , PFV/10/002 (OPTEC), several PhD/postdoc & fellow grants; Flemish Government: IOF: IOF/KP/SCORES4CHEM; FWO: PhD/postdoc grants, projects: G.0320.08 (convex MPC), G.0557.08 (Glycemia2), G.0588.09 (Brain-machine), G.0377.09 (Mechatronics MPC); G.0377.12 (Structured systems) research community (WOG: MLDM); IWT: PhD Grants, projects: Eureka-Flite+, SBO LeCoPro, SBO Climaqs, SBO POM, EUROSTARS SMART, iMinds 2012. Belgian Federal Science Policy Office: IUAP P7/ (DYSCO, Dynamical systems, control and optimization, 2012-2017). EU: ERNSI, FP7-EMBOCON (ICT-248940), FP7-SADCO (MC ITN-264735), ERC ST HIGHWIND (259 166), ERC AdG A-DATADRIVE-B. COST: Action ICO806: IntelliCIS.

V. Cevher was supported by the European Commision under Grant MIRG-268398, ERC Future Proof and SNF 200021-132548 .

REFERENCES

[1] H.H. Bauschke and P.L. Combettes. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer Verlag, 2011.

[2] D. P. Bertsekas and J. N. Tsitsiklis. *Parallel and distributed computation*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[3] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011.

[4] J.F. Cai, E.J. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.

[5] B. De Moor, M. Moonen, L. Vandenberghe, and J. Vandewalle. A geometrical approach for the identification of state space models with singular value decomposition. In *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 2244–2247. IEEE, 1988.

[6] M. Fazel. *Matrix rank minimization with applications*. PhD thesis, Elec. Eng. Dept., Stanford University, 2002.

[7] M. Fazel, H. Hindi, and S.P. Boyd. A rank minimization heuristic with application to minimum ordersystem approximation. In *Proceedings of the American Control Conference, 2001*, volume 6, pages 4734–4739, 2001.

[8] M. Fazel, T. K. Pong, D. Sun, and P. Tseng. Hankel matrix rank minimization with applications in system identification and realization. *Submitted*, 2012.

[9] P.M.O. Gebraad, J.W. van Wingerden, G.J. van der Veen, and M. Verhaegen. LPV subspace identification using a novel nuclear norm regularization method. In *American Control Conference (ACC), 2011*, pages 165–170. IEEE, 2011.

[10] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, third edition, 1996.

[11] BS He, H Yang, and SL Wang. Alternating direction method with self-adaptive penalty parameters for monotone variational inequalities. *Journal of Optimization Theory and applications*, 106(2):337–356, 2000.

[12] Z. Liu, A. Hansson, and L. Vandenberghe. Nuclear norm system identification with missing inputs and outputs. *Submitted to System and Control Letters*, 2012.

[13] Z. Liu and L. Vandenberghe. Interior-point method for nuclear norm approximation with application to system identification. *SIAM Journal on Matrix Analysis and Applications*, 31(3):1235–1256, 2009.

[14] K. Mohan and M. Fazel. Reweighted nuclear norm minimization with application to system identification. In *American Control Conference (ACC), 2010*, pages 2953–2959. IEEE, 2010.

[15] B. Recht, M. Fazel, and P.A. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Rev.*, 52:471–501, 2010.

[16] M. Signoretto and J.A.K. Suykens. Dynopt: Incorporating dynamics into mean-variance portfolio optimization. In *IEEE Symposium on Computational Intelligence for Financial Engineering & Economics (CIFEr)*, 2013, to appear.

[17] N. Srebro. *Learning with matrix factorizations*. PhD thesis, Massachusetts Institute of Technology, 2004.

[18] P. Van Overschee and B. De Moor. *Subspace Identification for the Linear Systems: Theory–Implementation–Applications*. Kluwer Academic Publishers, 1996.

[19] M. Verhaegen and P. Dewilde. Subspace model identification part 1. the output-error state-space model identification class of algorithms. *International journal of control*, 56(5):1187–1210, 1992.