

An Algorithm for Fast Constrained Nuclear Norm Minimization and Applications to Systems Identification

Mustafa Ayazoglu

Mario Sznaier

Abstract—This paper presents a novel algorithm for efficiently minimizing the nuclear norm of a matrix subject to structural and semi-definite constraints. It requires performing only thresholding and eigenvalue decomposition steps and converges Q-superlinearly to the optimum. Thus, this algorithm offers substantial advantages, both in terms of memory requirements and computational time over conventional semi-definite programming solvers. These advantages are illustrated using as an example the problem of finding the lowest order system that interpolates a collection of noisy measurements.

I. INTRODUCTION AND MOTIVATION

The problem of identifying a causal, finite dimensional linear shift invariant system from samples of its output and some a-priori information, has been extensively studied in the past 2 decades. The case where the structure of the model is known up to a set of (unknown) parameters, leads to classical parameter identification methods [24]. When such models are not available a-priori, then non-parametric techniques, roughly divided into subspace identification [31] and operator-theoretic methods [12], [30], must be used.

Subspace-based methods evolved from state-space realization theory [24] leading to a number of algorithms known as the 4SID methods [31]. While these methods are computationally attractive, they cannot incorporate *a priori* information, such as hard bounds on the noise, \mathcal{H}_∞ gain of the system or its stability margin. Indeed, there is no guarantee that the model produced by subspace identification methods is stable, in particular when dealing with noisy data generated by a lightly damped system. Further, there are no a-priori computable bounds on the distance from the identified model to the actual plant.

Alternatively, operator-based methods adopt a worse-case approach, and produce a nominal model of the system and a *worst case* identification error bound. In this context the identification problem can be restated as: given a known input \mathbf{u} and the corresponding output \mathbf{y} corrupted by noise η , find a bounded (for instance in the induced ℓ^2 sense) linear operator \mathcal{L} such that $\mathbf{y} = \mathcal{L}(\mathbf{u}) + \eta$, subject to additional constraints on its structure. For many cases of practical interest, this formulation leads to a convex optimization problem that can be efficiently solved [12]. Moreover, these algorithms converge to the actual plant as the information is completed [30]. However, a difficulty with these methods is that they

lead to high order models (typically the order is roughly the number of data points used in the identification). Thus, in order to avoid high order controllers, a model reduction step is usually necessary. Unfortunately, the resulting model may no longer satisfy some of the a-priori assumptions or interpolate the experimental data within the noise level.

As we show in this paper (see section V for details), the difficulties noted above can be circumvented by developing an approach that combines the advantages of subspace and worst-case methods, leading to low order, interpolatory models with guaranteed error bounds. While in principle this requires solving a challenging constrained rank minimization problem, using a (weighted) nuclear norm as surrogate for rank leads to a convex optimization problem of the form:

$\min_{\mathbf{g}} \|\mathbf{H}(\mathbf{g})\|_{*,\mathbf{w}}$ subject to:

$$\begin{bmatrix} K\mathbf{R}^{-2} & \mathbf{T}(\mathbf{g})^T \\ \mathbf{T}(\mathbf{g}) & K\mathbf{R}^2 \end{bmatrix} \succeq 0 \quad (1)$$

$$\|\mathbf{y} - \mathbf{T}(\mathbf{g})\mathbf{u}\|_\infty \leq \epsilon$$

where (\mathbf{u}, \mathbf{y}) denote the input applied to the system and the corresponding measured output, K and \mathbf{R} are given^a, and where the operators \mathbf{H} and \mathbf{T} have a Hankel and Toeplitz structure, respectively. In principle, this problem can be solved using conventional SDP solvers. However, their poor scaling properties (the computational complexity scales as $\mathcal{O}(\text{number of variables}^3)$) prevents the use of this approach beyond moderately sized problems. Our main result seeks to address this difficulty by developing a computationally efficient algorithm to solve a general class of constrained nuclear norm minimization problems of the form:

$$\begin{aligned} \min_{\mathbf{h}, \mathbf{e}_1, \mathbf{e}_2} & \|\mathcal{S}_H(\mathbf{h})\|_{\mathbf{w},*} + \lambda_1 \|\mathbf{e}_1\|_1 + \frac{\lambda_2}{2} \|\mathbf{e}_2\|_2^2 \\ \text{s.t. } & \mathbf{d} = \mathbf{F}\mathbf{h} + \mathbf{e}_1 + \mathbf{e}_2 \\ & \mathbf{A}\mathbf{h} - \mathbf{b} \geq 0, \quad \mathcal{S}_Q(\mathbf{h}) \succeq 0 \end{aligned} \quad (2)$$

where the matrices $\mathcal{S}(\cdot)$ are affine functions that define the structure of the problem. For instance, in the case of finding low order interpolants, \mathcal{S}_H and \mathcal{S}_Q have a Hankel and Toeplitz structure respectively, and \mathbf{e}_1 and \mathbf{e}_2 are used to handle outliers and measurements noise. Problem (2) can be thought of as a generalization of the *Robust PCA problem* [10], [11]: decomposing a given data matrix \mathbf{D} as $\mathbf{D} = \mathbf{A} + \mathbf{E}$ where \mathbf{A} has low rank and \mathbf{E} is sparse. During

This work was supported in part by NSF grant ECCS-0901433; AFOSR grant FA9559-12-1-0271; and the Alert DHS Center of Excellence under award number 2008-ST-061-ED0001. M. Ayazoglu and M. Sznaier are with the Department of Electrical and Computer Engineering, Northeastern University, Boston, MA, 02115.

^aThese values are related to the ℓ^2 induced gain and time constants of the system, respectively.

the past few years, substantial progress has been made in solving this problem, leading to several computationally efficient algorithms (see for instance [22], [3], [18]). Still, these methods are not capable of handling semi-definite and inequality constraints and thus cannot be applied to problems of the form (1). The main result of this paper is a first order iterative algorithm for solving general problems of the form (2). This algorithm uses only first order information, thus avoiding the computational complexity of interior-points methods and can be shown to converge Q-superlinearly to the optimum. Moreover, it requires performing at each step only thresholding and eigenvalue decomposition steps, both of which are amenable to fast, computationally efficient implementations. In the second part of the paper we illustrate the advantages of the algorithm using as an application the problem of finding low rank interpolants from noisy measurements corrupted with outliers, an issue central to control-oriented identification methods.

II. PRELIMINARIES

For ease of reference, in this section we summarize the notation used in the paper and provide some background material on Augmented Lagrangian Methods.

A. Notation

\mathbf{M}^T	Transpose of the matrix \mathbf{M}
$\text{Trace}\{\mathbf{M}\}$	Trace of the square matrix \mathbf{M} .
$\sigma_i(\mathbf{M})$	i^{th} largest singular value of \mathbf{M} .
$\langle \mathbf{M}, \mathbf{N} \rangle$	Inner product of square $n \times n$ matrices: $\langle \mathbf{M}, \mathbf{N} \rangle \doteq \text{Trace}\{\mathbf{M}^T \mathbf{N}\}$
$\ \mathbf{M}\ _F$	Frobenious norm: $\ \mathbf{M}\ _F^2 \doteq \langle \mathbf{M}, \mathbf{M} \rangle$
$\mathbf{M} \succeq 0$	\mathbf{M} is positive semidefinite.
$\ \mathbf{M}\ _{\mathbf{W},*}$	(Weighted) nuclear norm: $\ \mathbf{M}\ _* = \sum_i w_i \sigma_i(\mathbf{M})$. In the sequel we will denote $\ \cdot\ _{\mathbf{I},*}$ simply as $\ \cdot\ _*$.
$\ \mathbf{M}\ _1$	ℓ_1 norm: $\ \mathbf{M}\ _1 = \sum_{i,j} M_{ij} $.
$\text{vect}(\mathbf{M})$	Matrix vectorizing operator: $\mathbf{m} = \text{vect}(\mathbf{M})$ is a vector formed by stacking the columns of \mathbf{M} .
$\text{mat}(\mathbf{a})$	Vector to matrix operation: $\text{mat}(\mathbf{a}) \doteq [\mathbf{a}(1 : n - 1) \quad \mathbf{a}(n : 2n - 1) \dots]$

$\mathcal{D}(x, \tau, w)$ Weighted soft thresholding operation:

$$\mathcal{D}(x, \tau, w) = \max \{0, \text{sign}(x)(|x| - \tau w)\}$$

When applied to matrices, $\mathcal{D}(\cdot, \cdot, \cdot)$ acts element-wise, by shrinking each element in the matrix by the product of its corresponding weight $w(i,j)$ times τ . In the sequel, by a slight abuse of notation we will use $\mathcal{D}(x, \tau)$ when $w = 1$.

$\mathcal{H}(x, \tau)$ Hard thresholding operation:

$$\mathcal{H}(x, \tau) = \max \{\tau, x\}$$

ℓ^∞	space of vector valued sequences equipped with the norm: $\ x\ _{\ell^\infty} \doteq \sup_{ij} x_{i,j} $.
$\mathcal{H}_{\infty, \rho}$	space of transfer functions analytic in $ z \leq \rho \leq 1$, equipped with the norm $\ G\ _{\infty, \rho} \doteq \text{ess sup}_{ z < \rho} \bar{\sigma}(G(z))$. The case $\rho = 1$ will be simply denoted \mathcal{H}_∞ .
$\mathcal{B}\mathcal{X}(K)$	closed K -ball in a normed space $\{\mathcal{X}, \ \cdot\ \}$: $\mathcal{B}\mathcal{X}(K) = \{x \in \mathcal{X} : \ x\ _{\mathcal{X}} \leq K\}$.

B. Overview of Augmented Lagrangian Methods

Augmented Lagrangian Methods (ALM) are a class of first order methods that seek to solve constrained optimization problem of the form:

$$\min_{\mathbf{X}} f(\mathbf{X}) \text{ subject to } h(\mathbf{X}) = 0 \quad (3)$$

by iteratively minimizing the augmented Lagrangian:

$$L(\mathbf{X}, \mathbf{Y}, \mu) = f(\mathbf{X}) + \langle \mathbf{Y}, h(\mathbf{X}) \rangle + \frac{\mu}{2} \|h(\mathbf{X})\|_F^2 \quad (4)$$

where \mathbf{Y} and μ are a Lagrange Multiplier matrix and a penalty weight, respectively. As originally shown in [28], unlike classical penalty function methods, the algorithm outlined below will converge to a local minimum (and hence a global one in the case of convex problems), without the need for taking $\mu \rightarrow +\infty$ (see also [29], [5]).

Algorithm II-B: GENERAL ALM

while not converged **do**

1. $\mathbf{X}_{k+1} = \text{argmin}_X L(\mathbf{X}_k, \mathbf{Y}_k, \mu_k)$
2. $\mathbf{Y}_{k+1} = \mathbf{Y}_k + \rho h(\mathbf{X}_{k+1})$
3. $\mu_{k+1} = \rho \mu_k$

end while

The original algorithm has been extended in a number of ways to incorporate the ability to handle inequality constraints. Conn *et. al.* [15], suggested converting these constraints into equality ones through the introduction of slack variables, and handling positivity of these at the subproblem level. This idea has been further generalized to semidefinite constraints in [26], [8], [7], [32], [4].

III. AN EFFICIENT ALGORITHM FOR STRUCTURALLY CONSTRAINED NUCLEAR NORM MINIMIZATION

In this section we develop an efficient, ALM type algorithm for solving general problems of the form:

$$\begin{aligned} \min_{\mathbf{h}, \mathbf{e}_1, \mathbf{e}_2} & \|\mathcal{S}_H(\mathbf{h})\|_{\mathbf{W},*} + \lambda_1 \|\mathbf{e}_1\|_1 + \frac{\lambda_2}{2} \|\mathbf{e}_2\|_2^2 \\ \text{s.t. } & \mathbf{d} = \mathbf{F}\mathbf{h} + \mathbf{e}_1 + \mathbf{e}_2 \\ & \mathbf{A}\mathbf{h} - \mathbf{b} \geq 0 \\ & \mathcal{S}_Q(\mathbf{h}) \succeq 0 \end{aligned} \quad (5)$$

where \mathbf{W} , λ_1 and λ_2 are given weights, \mathbf{d} is a given data vector and where \mathcal{S}_H , \mathcal{S}_Q , are given affine operators that encapsulate the structural and semi-definite constraints of the problem. Further, due to their affine nature, by using suitable

defined matrices \mathbf{S}_h and \mathbf{S}_q , these operators can be rewritten as:

$$\mathcal{S}_H(h) = \mathbf{mat}(\mathbf{S}_h \mathbf{h} + \mathbf{v}_h), \quad \mathcal{S}_Q(h) = \mathbf{mat}(\mathbf{S}_q \mathbf{h} + \mathbf{v}_q) \quad (6)$$

To illustrate this fact, consider an example involving a combination of Hankel and constant terms.

$$\mathcal{S}_H(h) = \begin{bmatrix} h(1) & h(2) & 1 & 0 \\ h(2) & h(3) & 0 & 1 \\ 1 & 0 & h(1) & h(2) \\ 0 & 1 & h(2) & h(3) \end{bmatrix}$$

Simple algebra shows that this operator admits a representation of the form (6) with

$$\mathbf{v}_h = [0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0]^T$$

$$\mathbf{S}_h = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \quad (7)$$

$$\mathbf{h} = [h(1) \ h(2) \ h(3)]^T$$

Using the explicit expressions for $\|\cdot\|_{\mathbf{W},*}$, \mathcal{S}_H and \mathcal{S}_Q and introducing slack variables \mathbf{z}_1 and \mathbf{z}_2 , the optimization problem (5) can be rewritten as:

$$\min_{\mathbf{h}, \mathbf{e}_1, \mathbf{e}_2} \sum_i w_i \sigma_i(\mathbf{mat}(\mathbf{S}_h \mathbf{h} + \mathbf{v}_h)) + \lambda_1 \|\mathbf{e}_1\|_1 \dots$$

$$\dots + \frac{\lambda_2}{2} \|\mathbf{e}_2\|_2^2 \quad (8)$$

$$\text{s.t. } \mathbf{d} - \mathbf{Fh} - \mathbf{e}_1 - \mathbf{e}_2 = 0$$

$$\mathbf{Ah} - \mathbf{b} - \mathbf{z}_1 = 0, \quad \mathbf{z}_1 \geq 0$$

$$\mathbf{S}_q \mathbf{h} + \mathbf{v}_q - \mathbf{z}_2 = 0, \quad \mathbf{mat}(\mathbf{z}_2) \succeq 0$$

Since this problem is convex, in principle it can be solved by using Algorithm II-B. However, a difficulty here is that, proceeding in this fashion will still require to numerically solve a constrained optimization problem at each step (to update \mathbf{X}), thus negating the advantages of the method. The main result of this section shows that problem (8) can be solved by solving a related optimization problem where each of the subproblems entailed in using Algorithm II-B admits a closed form solution. Moreover, finding this solution requires using only a combination of thresholding and eigenvalue decomposition steps, both of which are amenable to fast, efficient implementations. Hence the resulting algorithm is capable of handling large sized problems. The key insight is to add a new (vector) variable \mathbf{z}_3 and the associated constraint

$$\mathbf{S}_h \mathbf{h} + \mathbf{v}_h = \mathbf{z}_3 \quad (9)$$

to the problem^b, in order to decouple the norm minimization portions from the structural constraints. In turn, as shown below, this allows for finding computationally efficient closed form solutions at each iteration. Further, in this approach the structural constraints are not enforced in the intermediate steps (but, due to convergence are guaranteed to hold for the

^bthis is known as a ‘‘splitting variable technique’’ in the optimization literature [13], [14], [16], [1].

optimal solution), resulting in substantial speed-up vis-a-vis methods that enforce the constraint at each step.

The augmented lagrangian for the problem (8) with the additional constraint (9) is given by:

$$L(\mathbf{h}, \mathbf{e}_1, \mathbf{e}_2, \mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3, \mathbf{y}_0, \mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3, \mu) = \dots$$

$$\dots \sum_i w_i \sigma_i(\mathbf{mat}(\mathbf{z}_3)) + \lambda_1 \|\mathbf{e}_1\|_1 + \frac{\lambda_2}{2} \|\mathbf{e}_2\|_2^2$$

$$+ \langle \mathbf{y}_0, \mathbf{d} - \mathbf{Fh} - \mathbf{e}_1 - \mathbf{e}_2 \rangle + \frac{\mu}{2} \|\mathbf{Ah} - \mathbf{b} - \mathbf{z}_1\|_2^2$$

$$+ \langle \mathbf{y}_1, \mathbf{Ah} - \mathbf{b} - \mathbf{z}_1 \rangle + \frac{\mu}{2} \|\mathbf{d} - \mathbf{Fh} - \mathbf{e}_1 - \mathbf{e}_2\|_2^2$$

$$+ \langle \mathbf{y}_2, \mathbf{S}_q \mathbf{h} + \mathbf{v}_q - \mathbf{z}_2 \rangle + \frac{\mu}{2} \|\mathbf{S}_q \mathbf{h} + \mathbf{v}_q - \mathbf{z}_2\|_2^2$$

$$+ \langle \mathbf{y}_3, \mathbf{S}_h \mathbf{h} + \mathbf{v}_h - \mathbf{z}_3 \rangle + \frac{\mu}{2} \|\mathbf{S}_h \mathbf{h} + \mathbf{v}_h - \mathbf{z}_3\|_2^2$$

$$\text{s.t. } \mathbf{z}_1 \geq 0 \quad \mathbf{mat}(\mathbf{z}_2) \succeq 0$$

Adding terms of the form of $\langle \mathbf{y}_i, \mathbf{y}_i \rangle$, who do not affect the minimizer, to complete the squares, leads to the more compact form:

$$L(\mathbf{h}, \mathbf{e}_1, \mathbf{e}_2, \mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3, \mathbf{y}_0, \mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3, \mu) = \dots$$

$$\dots \sum_i w_i \sigma_i(\mathbf{mat}(\mathbf{z}_3)) + \lambda_1 \|\mathbf{e}_1\|_1 + \frac{\lambda_2}{2} \|\mathbf{e}_2\|_2^2$$

$$+ \frac{\mu}{2} \left\| \mathbf{d} - \mathbf{Fh} - \mathbf{e}_1 - \mathbf{e}_2 + \frac{\mathbf{y}_0}{\mu} \right\|_2^2$$

$$+ \frac{\mu}{2} \left\| \mathbf{Ah} - \mathbf{b} - \mathbf{z}_1 + \frac{\mathbf{y}_1}{\mu} \right\|_2^2 \quad (10)$$

$$+ \frac{\mu}{2} \left\| \mathbf{S}_q \mathbf{h} + \mathbf{v}_q - \mathbf{z}_2 + \frac{\mathbf{y}_2}{\mu} \right\|_2^2$$

$$+ \frac{\mu}{2} \left\| \mathbf{S}_h \mathbf{h} + \mathbf{v}_h - \mathbf{z}_3 + \frac{\mathbf{y}_3}{\mu} \right\|_2^2$$

$$\text{s.t. } \mathbf{z}_1 \geq 0 \quad \mathbf{mat}(\mathbf{z}_2) \succeq 0$$

As we show next, for each fixed μ, \mathbf{y}_i , the optimal value of $\mathbf{X} \doteq (\mathbf{h}, \mathbf{e}_1, \mathbf{e}_2, \mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3)$ that minimizes this Lagrangian can be found by iteratively performing a sequence of thresholding and eigenvalue decomposition steps.

Theorem 1: For fixed μ, \mathbf{y}_i , the minimizer of (10) can be found by iteratively repeating the following update steps.

$$\mathbf{h}^{k+1} = \mathbf{P}(\mathbf{F}^T \mathbf{v}_0 - \mathbf{A}^T \mathbf{v}_1 - \mathbf{S}_q^T \mathbf{v}_2 - \mathbf{S}_h^T \mathbf{v}_3)$$

$$\mathbf{P} = (\mathbf{F}^T \mathbf{F} + \mathbf{A}^T \mathbf{A} + \mathbf{S}_q^T \mathbf{S}_q + \mathbf{S}_h^T \mathbf{S}_h)^{-1}$$

$$\mathbf{v}_0 = \mathbf{d} - \mathbf{e}_1^k - \mathbf{e}_2^k + \frac{\mathbf{y}_0^k}{\mu}$$

$$\mathbf{v}_1 = -\mathbf{b} - \mathbf{z}_1^k + \frac{\mathbf{y}_1^k}{\mu} \quad (11)$$

$$\mathbf{v}_2 = \mathbf{v}_q - \mathbf{z}_2^k + \frac{\mathbf{y}_2^k}{\mu}$$

$$\mathbf{v}_3 = \mathbf{v}_h - \mathbf{z}_3^k + \frac{\mathbf{y}_3^k}{\mu}$$

$$\mathbf{e}_1^{k+1} = \mathcal{D}(\mathbf{d} - \mathbf{Fh}^{k+1} - \mathbf{e}_2^k + \frac{\mathbf{y}_0^k}{\mu}, \frac{\lambda_1}{\mu^k}) \quad (12)$$

$$\mathbf{e}_2^{k+1} = \frac{\mu^k}{(\mu^k + \lambda_2)} \mathbf{v}_5$$

$$\mathbf{v}_5 = \mathbf{d} - \mathbf{Fh}^{k+1} - \mathbf{e}_1^{k+1} + \frac{\mathbf{y}_0^k}{\mu^k} \quad (13)$$

$$\mathbf{z}_1^{k+1} = \mathcal{H}(\mathbf{Ah} - \mathbf{b} + \frac{\mathbf{y}_1^k}{\mu^k}, 0) \quad (14)$$

$$\begin{aligned} \mathbf{z}_2^{k+1} &= \mathbf{vect}(\mathbf{U}\mathcal{H}(\mathbf{D}, 0)\mathbf{U}^T); \text{ where} \\ \mathbf{U}\mathbf{D}\mathbf{U}^T &= \text{svd}(\mathbf{mat}(\mathbf{S}_q\mathbf{h} + \mathbf{v}_q + \frac{\mathbf{y}_2^k}{\mu^k})) \\ \mathbf{z}_3^{k+1} &= \mathbf{vect}(\mathbf{U}\mathcal{D}(\boldsymbol{\Sigma}, \frac{1}{\mu^k}, \mathbf{W})\mathbf{V}^T) \text{ where} \\ \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T &= \text{svd}(\mathbf{mat}(\mathbf{S}_h\mathbf{h}^{k+1} + \mathbf{v}_h + \frac{\mathbf{y}_3^k}{\mu^k})) \\ \mathbf{W} &= \mathbf{diag}\{w_{ii}\} \end{aligned} \quad (15)$$

where \mathcal{D} and \mathcal{H} denote the soft and hard thresholding operators, respectively.

Proof: Given in the Appendix Iteratively repeating the steps above leads to the optimal \mathbf{X}^{k+1} , for given values of the Lagrange multipliers \mathbf{y}_i^k . Once \mathbf{X}^{k+1} is available, the method proceeds as in the standard ALM case, updating the Lagrange multipliers using:

$$\begin{aligned} \mathbf{y}_0^{k+1} &= \mathbf{y}_0^k + \mu^k(\mathbf{d} - \mathbf{F}\mathbf{h} - \mathbf{e}_1 - \mathbf{e}_2) \\ \mathbf{y}_1^{k+1} &= \mathbf{y}_1^k + \mu^k(\mathbf{A}\mathbf{h} - \mathbf{b} - \mathbf{z}_1) \\ \mathbf{y}_2^{k+1} &= \mathbf{y}_2^k + \mu^k(\mathbf{S}_q\mathbf{h} + \mathbf{v}_q - \mathbf{z}_2) \\ \mathbf{y}_3^{k+1} &= \mathbf{y}_3^k + \mu^k(\mathbf{S}_h\mathbf{h} + \mathbf{v}_h - \mathbf{z}_3) \end{aligned} \quad (17)$$

and setting $\mu^{k+1} = \xi\mu^k$, for some $\xi > 1$. The complete algorithm is summarized next.

Algorithm III: SRPCA ALGORITHM EALM

$\xi > 1, \mu_0$
 $\mathbf{h}^0 = \mathbf{d}, \mathbf{e}_1^0 = \mathbf{0}, \mathbf{e}_2^0 = \mathbf{0}, \mathbf{z}_1^0 = \mathbf{0}, \mathbf{z}_2^0 = \mathbf{0}$
 $\mathbf{z}_3^0 = \mathbf{0}, \mathbf{y}_0^0 = \mathbf{0}, \mathbf{y}_1^0 = \mathbf{0}, \mathbf{y}_2^0 = \mathbf{0}, \mathbf{y}_3^0 = \mathbf{0}$
while not converged **do** (outer loop)
 while not converged **do** (inner loop)
 1. Solve \mathbf{h}^{k+1} using eq. (11)
 2. Solve \mathbf{e}_1^{k+1} using eq. (12)
 3. Solve \mathbf{e}_2^{k+1} using eq. (13)
 4. Solve \mathbf{z}_1^{k+1} using eq. (14)
 5. Solve \mathbf{z}_2^{k+1} using eq. (15)
 6. Solve \mathbf{z}_3^{k+1} using eq. (16)
 end while (inner loop)
 7. Update the lagrange multipliers using (17)
 8. $\mu^{k+1} = \xi\mu^k$
end while (outer loop)

Theorem 2: The sequence \mathbf{X}^k generated by Algorithm III converges to a solution of Problem (5). Moreover, $L(\mathbf{X}^k, \mathbf{y}_i^k)$ converges as $\mathcal{O}(\frac{1}{\mu^k})$ to the optimal value of the objective.

Proof: Omitted due to space constraints, follows along the lines of the proof of Theorem 1 in [3] ■

IV. AN ALTERNATING DIRECTIONS METHOD

While the algorithm above is guaranteed to converge Q-linearly to the optimal solution, the exact minimization of L in steps 1–6 could entail many iterations. Motivated by [6], we propose to avoid these iterations by considering an alternating directions type algorithm, where $(\mathbf{h}, \mathbf{e}_1, \mathbf{e}_2, \mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3)$ are updated only once in each iteration. The entire algorithm can be summarized as

Algorithm IV: WEIGHTED SRPCA ADMM ALGORITHM

Initializations are the same as (III)

while not converged **do**

1. Run the inner loop of Algorithm (III) once.
2. Update the lagrange multipliers using (17)

end while

As shown in [2] the algorithm above also generates a sequence \mathbf{X}^k that converges to the solution of Problem (5). Further, consistent numerical experience shows that updating the weight μ at each step using a law of the form $\mu^{k+1} = \xi\mu^k$ leads to improved convergence properties, although no formal proof of this fact is currently available.

V. APPLICATION: FINDING MINIMUM RANK INTERPOLANTS

In this section we illustrate the advantages of the proposed method by applying it to the problem of robustly identifying the lowest order system, consistent with given a-priori information, that interpolates a set of given experimental measurements (or establishing that none exists). Specifically, given N samples, corrupted by additive noise, of the time response of an unknown system \mathcal{G} , $y(k) = [G * u]_k + \eta_t(k)$, $k = 0, \dots, N$, where $*$ denotes convolution, and where the input $u(k)$ is known, and the following *a priori* assumptions

- 1) \mathcal{G} belongs to the class $\mathcal{BH}_{\infty, \rho}(K)$ i.e. the set of exponentially stable systems with a stability margin of $(1 - \rho)$, and a peak response to complex exponential inputs of K .
 - 2) A bound on the measurement noise: $\|\eta_t\|_{\ell_\infty} \leq \epsilon$
- our goal is twofold:

- 1) Firstly, to establish whether or not the consistency set^c:

$$\mathcal{T}(y) \doteq \left\{ \mathcal{G} \in \mathcal{BH}_{\infty, \rho}(K) : y_k = [G * u]_k + \eta_k, \|\eta\|_\infty \leq \epsilon \right\}$$

is empty.

- 2) If $\mathcal{T}(y) \neq \emptyset$, find the lowest order system $\mathcal{G} \in \mathcal{T}(y)$.

Next, recall that (see for instance [27]), given a finite vector sequence $\{\mathbf{g}_k\}_{k=1}^N$, there exists a system $\mathcal{G} \in \mathcal{H}_{\infty, \rho}(K)$ that has $\{\mathbf{g}_k\}$ as the first N elements of its impulse response sequence if and only if the following LMI holds:

$$\mathbf{L}(\mathbf{g}) \doteq \begin{bmatrix} K\mathbf{R}^{-2} & (\mathbf{T}_g^N)^T \\ \mathbf{T}_g^N & K\mathbf{R}^2 \end{bmatrix} \succeq 0 \quad (18)$$

where \mathbf{T}_g^N denotes the (finite) Toeplitz matrix associated with the sequence $\{\mathbf{g}\}$ and where $\mathbf{R} = \text{diag}[1 \ \rho \ \rho^2 \ \dots \ \rho^{N_t-1}]$

In principle, model order constraints can be handled by simply placing a constraint on the rank of the Hankel matrix associated with the sequence $\{\mathbf{g}\}$. Note however that this does not guarantee that when the complete interpolant is formed, its rank will equal that of the truncated Hankel

^cThe diameter of this set $e^*(y) \doteq d\{\mathcal{T}(y)\}$ defines an *intrinsic local* worst-case identification error, in the sense that this is the best that can be achieved by any interpolatory algorithm.

matrix. As shown next, this difficulty can be circumvented by considering an augmented optimization problem. To this effect, we recall next the following from [34], providing a bound on the Hankel singular values of the interpolant:

Theorem 3: Given $\epsilon > 0$, select n so that

$$2K \frac{\rho^{n+2}}{1-\rho} < \epsilon$$

Define $\mathbf{g}_{aug}^T \doteq [g_0 \ \dots \ g_N \ g_{N+1} \ \dots \ g_n]^T$ and consider the following feasibility problem:

$$\mathbf{L}(\mathbf{g}_{aug}) \doteq \begin{bmatrix} K\mathbf{R}^{-2} & (\mathbf{T}_{g_{aug}}^n)^T \\ \mathbf{T}_{g_{aug}}^n & K\mathbf{R}^2 \end{bmatrix} \geq 0 \quad (19)$$

$$\|\mathbf{y} - \mathbf{T}_{g_{aug}}^N \mathbf{u}\|_\infty \leq \epsilon \quad (20)$$

$$\text{rank}(H_{g_{aug}}^{n,n}) \leq n_{red} \quad (21)$$

If this problem admits a feasible solution \mathbf{g}_{aug} , then the Hankel singular values of the associated interpolant $G_{aug}(z)$ satisfy $\sigma_i^H(G_{aug}) \leq \epsilon$ for all $i \geq n$.

Proof: Follows from Corollary 1 in [34] ■

A. Illustrative Examples

Combining Theorem 3 with the usual (weighted) nuclear norm relaxation of rank [17], it follows that interpolants with order arbitrarily closed to the minimum can be (approximately) found by solving a sequence of problems of the form:

$$\begin{aligned} \min_{\mathbf{g}} \|\mathbf{H}(\mathbf{g})\|_{*,\mathbf{W}} + \lambda_1 \|\mathbf{e}_1\|_1 + \frac{\lambda_2}{2} \|\mathbf{e}_2\|_2^2 \text{ subj. to:} \\ \begin{bmatrix} K\mathbf{R}^{-2} & \mathbf{T}(\mathbf{g})^T \\ \mathbf{T}(\mathbf{g}) & K\mathbf{R}^2 \end{bmatrix} \succeq 0 \\ \mathbf{y} - \mathbf{T}(\mathbf{g})\mathbf{u} - \mathbf{e}_1 - \mathbf{e}_2 = 0 \end{aligned} \quad (22)$$

where we have replaced the hard bounds on the interpolation error given by (20) with soft ones, and allowed for the existence of outliers (the sparse signal \mathbf{e}_1). In the remainder of this section we use the proposed algorithm to solve Problem (22) in several scenarios and compare its performance against the results obtained using both conventional subspace identification methods [23] and Hankel rank minimization. In all cases we used as an estimate of the order of the resulting interpolant the number of Hankel singular values above 1% of the maximum. In addition, we used $\mu_0 = 10^{-3}$ and $\xi = 1.05$ as the internal parameters of the augmented Lagrangian method, and we initialized all variables to zero.

Experiment 1 –Noisy Data

In this example we compare the results obtained when applying our algorithm (SDPNC), subspace identification (SSID), and Hankel rank minimization (HRM) to the problem of identifying a system from noisy data. For this experiment we used a second order system with poles at $z = 0.8 \pm 0.51j$ and a single zero at $z = -1$. The corresponding transfer function is given by

$$S(z) = (0.1z + 0.1)/(z^2 - 1.6z + 0.9) \quad (23)$$

The experimental data consisted of $N = 20$ samples of the step response of this system, corrupted by 23% uniformly distributed measurement noise. The *a-priori* information was

n	Our Method(secs)	SDP solver 1st iter. (secs)
86	13.8	469.4
96	17.3	771.5
106	21.3	1495.3
116	25.7	2331.4

TABLE I
RUNNING TIME COMPARISON

assumed to be $\rho = (1.05)^{-1}$, $K = 50$ and $\epsilon = 0.3$. Figure 1 shows the step responses of the system identified by the proposed algorithm, using weights $\lambda = 10000$ and $\lambda_2 = 500$. For comparison, we also show the results of subspace identification and of Hankel rank minimization, obtained by solving:

$$\begin{aligned} \min_{\mathbf{h}, \mathbf{e}_2} \|\mathbf{H}(\mathbf{g})\|_{*,\mathbf{W}} + \frac{\lambda_2}{2} \|\mathbf{e}_2\|_2^2 \\ \text{s.t. } \mathbf{y} - \mathbf{T}(\mathbf{g})\mathbf{u} - \mathbf{e}_2 = 0 \end{aligned} \quad (24)$$

with $\lambda_2 = 500$. As shown in the figure, the system obtained using SSID fits poorly the experimental data and is not consistent with the *a priori* information, since it has poles at $|z| = 0.98 > \rho$. Similarly, HRM leads to an open-loop unstable system. The failure of these methods is due to a combination of relatively large noise and poor damping, coupled with the fact that neither one of them can guarantee that the resulting interpolant will be analytic in a region of the form $|z| > \rho$, with ρ given. For comparison, the proposed method leads to a low order stable system that interpolates the experimental data within the noise bounds and is consistent with the *a priori* information.

Next, we compare the computational costs of solving problem (22) using the proposed algorithm versus standard interior point methods (in this case we used SDPT3). The critical parameter determining computational complexity is the number of variables, which in this case is related to the horizon n . Running time comparisons between these two methods as a function of n are given in Table I. As shown there, the proposed algorithm outperforms SDPT3 by at least one order of magnitude for low values of n and this gap substantially increases with n . These figures were obtained using MATLAB 7.13 on a Windows 7 64-bit machine with a 2.7GHz processor and 24GB RAM.

Experiment 2 –Missing Data, Noise and Outliers

In this experiment we illustrate the ability of our method to handle very challenging scenarios, characterized by a combination of noise, missing data and outliers. In this case we used $N = 50$ elements of the step response of system (23), corrupted with 17% uniformly distributed measurement noise, outliers (probability 25%) and missing data (probability 5%). This is intended to simulate a situation where data is transmitted over a wireless channel, with a certain probability of being substantially corrupted by interference (outliers) or being dropped altogether. Fig. 2 compares the results obtained using our algorithm (with $\lambda = 100$) against

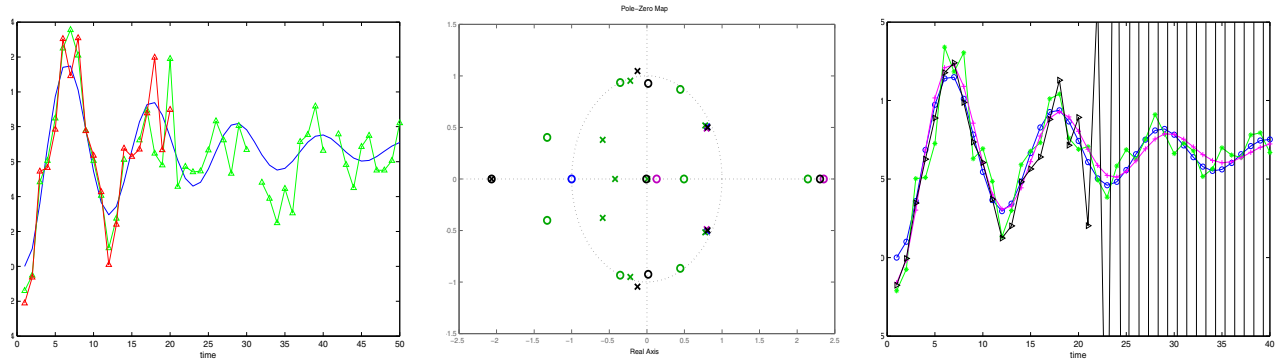


Fig. 1. Left: Step response data used in the experiments (blue, red and green denote the clean signal and the noisy data used in the experiments 1 and 2, respectively). Center: pole zero map of the identified systems for experiment 1 ((original (blue), SDPNC (magenta), SSID (green), and NNC (black)). Right: step responses of the original (blue circles) and identified systems: SDNPC (magenta crosses), SSID (green stars) and HRM (black triangles)

those obtained using HRM and SSID^d. As shown there, the proposed algorithm successfully identified a low order interpolant. On the other hand, the systems identified using SSID and HRM fit poorly the data.

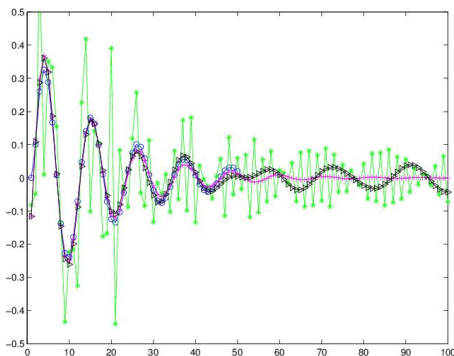


Fig. 2. Impulse responses of the systems identified using different approaches in Experiment 2. Blue line with circle markers: original signal. Magenta lines with crosses: output of the system identified using SDPNC. Green line with star markers: SSID. Black lines with triangular markers: HRM

VI. CONCLUSIONS

Many problems of practical interest involve minimizing the nuclear norm of a matrix subject to structural and semi-definite constraints. Examples include robust identification from noisy measurements, potentially corrupted with outliers (the time domain case leads to mixed Hankel/Toeplitz structures, while the frequency domain involves Loebner matrices) or moments based relaxations of constrained polynomial optimization problems [21]. While in principle these problems are convex and hence can be solved using conventional SDP solvers, in practice, this approach is limited to

^dsince SSID cannot handle missing data, for this algorithm we considered a more benign scenario, with only noise and outliers but no missing data.

moderately sized problems, due to the poor scaling properties of interior-point methods. Motivated by this difficulty, in the past few years considerably attention has been devoted to developing fast, scalable first order methods. However, while capable of handling large size problems, up to now applicability of these methods was limited by their inability to handle structural and semi-definite constraints. The main result of this paper removes this limitation, providing a fast algorithm for solving a large class of constrained nuclear norm minimization problems. Notably, this algorithm requires performing, at each step only thresholding and eigenvalue (or polar) decomposition steps, allowing for fast, computationally efficient implementations. These results were illustrated with both academic examples and a non-trivial identification problem arising in the context of vision-based tracking. As shown there, in all cases the proposed algorithm resulted in substantial improvement, both in computational time and memory requirements, over SDP-solver based approaches.

REFERENCES

- [1] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo. Fast image recovery using variable splitting and constrained optimization. Technical Report arXiv:0910.4887, Oct 2009.
- [2] M. Ayazoglu. *Fast Sparse Subspace Identification Tools with Applications to Dynamic Vision*. PhD thesis, ECE Dept., Northeastern University, Boston, MA, USA, 2012.
- [3] M. Ayazoglu, M. Sznaier, and O. Camps. Fast algorithms for structured robust principal component analysis. In *2012 IEEE CVPR*.
- [4] N. S. Aybat and G. Iyengar. Unified approach for minimizing composite norms. Under review in Math. Prog. Ser. A, 2010.
- [5] D. Bertsekas. *Constrained Optimization and Lagrange Multiplier Method*. Academic Press, 1982.
- [6] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*, 3(1):1–122, January 2011.
- [7] S. Burer. Optimizing a polyhedral-semidefinite relaxation of completely positive programs. *Mathematical Programming Computation*, 2(1):1–19, March 2010.
- [8] S. Burer and D. Vandembussche. Solving lift-and-project relaxations of binary integer programs. *SIAM Journal on Optimization*, 16:726–750.
- [9] J. Cai and S. Osher. Fast singular value thresholding without singular value decomposition. *UCLA CAM Report*, pages 1–14, 2010.

- [10] E. J. Candes, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *J. ACM*, (3), 2011.
- [11] V. Chandrasekaran, S. Sanghavi, P. Parrilo, and A. Willsky. Rank-sparsity incoherence for matrix decomposition. *Siam J. Optim.*, (2):572–596, 2011.
- [12] J. Chen and G. Gu. *Control Oriented System Identification, An \mathcal{H}_∞ Approach*. John Wiley, New York, 2000.
- [13] P.L. Combettes and J.-C. Pesquet. A douglas–rachford splitting approach to nonsmooth convex variational signal recovery. *Selected Topics in Signal Processing, IEEE Journal of*, 1(4):564–574, dec. 2007.
- [14] P.L. Combettes and V. Wajs. Signal recovery by proximal forward backward splitting. *SIAM Journal on Multiscale Modeling & Simulation*, 4:1168–1200, dec. 2005.
- [15] A. R. Conn, N. Gould, A. Sartenaer, and Ph. L. Toint. Convergence properties of an augmented lagrangian algorithm for optimization with a combination of general equality and linear constraints. *SIAM J. on Optimization*, 6:674–703, March 1996.
- [16] J. Eckstein and D. P. Bertsekas. On the douglas-rachford splitting method and the proximal point algorithm for maximal monotone operators. *Mathematical Programming*, 55:293–318, 1992.
- [17] M. Fazel. *Matrix Rank Minimization with Applications*. PhD thesis, Stanford, California, USA, 2002.
- [18] M. Fazel, T. K. Pong, S. Defeng, and P. Tseng. Hankel matrix rank minimization with applications to system identification. 2012.
- [19] N. J. Higham. Computing a nearest symmetric positive semidefinite matrix. *Linear Algebra and its Applications*, 103:103–115, 1988.
- [20] N. J. Higham. Fast polar decomposition of an arbitrary matrix. *SIAM Journal on Scientific and Statistical Computing*, 11:648–655, 1990.
- [21] J.B. Lasserre. Global optimization with polynomials and the problem of moments. *SIAM J. Optimization*, 11(3):796–817, 2001.
- [22] Z. Lin, M. Chen, and L. Wu. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. *Analysis, math.OC(Technical Report UILU-ENG-09-2215):09–2215, 2009*.
- [23] Z. Liu and L. Vandenberghe. Interior-point method for nuclear norm approximation with application to system identification. *SIAM J. Matrix Anal. Appl.*, 31:1235–1256, November 2009.
- [24] L. Ljung. *System Identification: Theory for the User*. Prentice-Hall, Englewood Cliffs, N.J., 1987.
- [25] Y. Ni, J. Sun, X. Yuan, S. Yan, and L. Cheong. Robust low-rank subspace segmentation with semidefinite guarantees. *Data Mining Workshops, International Conference on*, 0:1179–1188, 2010.
- [26] D. Noll, M. Torke, and P. Apkarian. Partially augmented lagrangian method for matrix inequality constraints. *SIAM J. on Optimization*, 15:161–184, January 2005.
- [27] P. A. Parrilo, M. Sznaier, and R. S. Sanchez Pena. Mixed time/frequency domain based robust identification. *Automatica*, 34(11):1375–1389, 1998.
- [28] M. J. D. Powell. A method for nonlinear constraints in minimization problems. In *Optimization (Sympos., Univ. Keele, Keele, 1968) R.Fletcher(ed)*, 1969.
- [29] R. T. Rockafellar. The multiplier method of hestenes and powell applied to convex programming. pages 555–562, 1973.
- [30] R. Sánchez Peña and M. Sznaier. *Robust Systems Theory and Applications*. Wiley & Sons, Inc., 1998.
- [31] P. Van Overschee and B. De Moor. Subspace algorithms for the stochastic identification problem. *Automatica*, 29(3):649–660, May 1993.
- [32] Z. Wen, Goldfarb D., S. Ma, and Scheinberg K. Row by row methods for semidefinite programming, tech. Technical report, Columbia University, 2009.
- [33] J. Xu. Reweighted nuclear norm minimization for matrix completion. preprint, 2011.
- [34] B. Yilmaz, M. Ayazoglu, M. Sznaier, and C. Lagoa. Convex relaxations for robust identification of wiener systems and applications. In *Proc. 2011 IEEE Conf. Dec. Control*, pages 2812–2818, 2011.

APPENDIX

A. Explicit Closed Form Expressions for the Solutions to the Intermediate Optimization Problems.

At the beginning of the $(k + 1)^{th}$ iteration, the values of $\mathbf{e}_1^k, \mathbf{e}_2^k, \mathbf{z}_1^k, \mathbf{z}_2^k, \mathbf{z}_3^k, \mathbf{y}_1^k, \mathbf{y}_2^k, \mathbf{y}_3^k, \mu^k$ are known and the goal is to minimize L with respect to \mathbf{h}^{k+1} . Solving

$\frac{\partial L}{\partial \mathbf{h}} = 0$ leads to (11). It is worth emphasizing that since \mathbf{P} is a constant matrix that depends only on the structure of the problem, its inversion needs to be performed once, and not at every step. Next consider the problem of minimizing L w.r.t. \mathbf{e}_1 . It is easily seen that the minimizer is given by

$$\mathbf{e}_1^{k+1} = \underset{\mathbf{e}_1}{\operatorname{argmin}} \frac{\lambda_1}{\mu^k} \|\mathbf{e}_1\|_1 + \dots \dots \frac{1}{2} \left\| \mathbf{d} - \mathbf{F}\mathbf{h}^{k+1} - \mathbf{e}_2^k + \frac{\mathbf{y}_0^k}{\mu} - \mathbf{e}_1 \right\|_2^2$$

a problem whose explicit solution is given in [22], leading to (12). Note in passing that by using the results from [33], the operator $\mathcal{D}(\cdot)$ can be applied with a “mask” to handle cases with missing data or outliers. The update (13) follows from solving $\frac{\partial L}{\partial \mathbf{e}_2} = 0$.

Next consider the problem of minimizing L w.r.t. \mathbf{z}_1 . Taking into account only the terms that depend explicitly on \mathbf{z}_1 leads to:

$$\mathbf{z}_1^{k+1} = \underset{\mathbf{z}_1}{\operatorname{argmin}} \left\| \mathbf{A}\mathbf{h} - \mathbf{b} - \mathbf{z}_1 + \frac{\mathbf{y}_1^k}{\mu^k} \right\|_2^2 \quad \text{s.t. } \mathbf{z}_1 \geq 0$$

a problem whose optimal solution is given by $(\mathbf{z}_1)_i = \max \left\{ (\mathbf{A}\mathbf{h} - \mathbf{b} - \mathbf{z}_1 + \frac{\mathbf{y}_1^k}{\mu^k})_i, 0 \right\}$, where $(\mathbf{v})_i$ denotes the i^{th} component of \mathbf{v} , leading to (14). Similarly, minimizing L w.r.t. \mathbf{z}_2 , leads to the following optimization problem:

$$\mathbf{z}_2^{k+1} = \mathbf{vect}(\underset{\mathbf{z}_2}{\operatorname{argmin}} \left\| \mathbf{mat}(\mathbf{S}_q \mathbf{h} + \mathbf{v}_q + \frac{\mathbf{y}_2^k}{\mu^k} - \mathbf{z}_2) \right\|_F^2) \quad \text{s.t. } \mathbf{mat}(\mathbf{z}_2) \geq 0$$

To obtain a closed form solution for this problem, note that it is equivalent to

$$\mathbf{Z}_2^{k+1} = \underset{\mathbf{Z}_2}{\operatorname{argmin}} \|\mathbf{D} - \mathbf{Z}_2\|_F^2 \quad \text{s.t. } \mathbf{Z}_2 \succeq 0$$

where \mathbf{D} is diagonal matrix containing the singular values of $\mathbf{mat}(\mathbf{S}_q \mathbf{h} + \mathbf{v}_q + \frac{\mathbf{y}_2^k}{\mu^k})$, and where we used the fact that $\|\cdot\|_F$ is unitarily invariant. Clearly, the solution to this problem is a diagonal matrix, with $\mathbf{Z}_2^{k+1}_{i,i} = \max\{\mathbf{D}_{i,i}, 0\}$, leading to (15). From a computational standpoint, performing this step requires performing a svd, which could limit the size of the problems that can be handled. This can be avoided by using the method in [19] to replace it with a polar decomposition step that requires performing only matrix multiplication operations[20].

Finally, minimizing L w.r.t. \mathbf{z}_3 leads to

$$\mathbf{z}_3^{k+1} = \mathbf{vect}(\underset{\mathbf{z}_3}{\operatorname{argmin}} \sum_i w_i \sigma_i(\mathbf{mat}(\mathbf{z}_3)) + \dots \dots \frac{\mu^k}{2} \left\| \mathbf{mat}(\mathbf{S}_h \mathbf{h}^{k+1} + \mathbf{v}_h + \frac{\mathbf{y}_3^k}{\mu} - \mathbf{z}_3) \right\|_F^2) \quad (25)$$

whose explicit solution is given by (16) [33]. Computationally efficient implementations can be obtained by using partial svd tools or avoiding the svd step altogether, proceeding as suggested in [9]. Finally, it is worth emphasizing that when $\mathcal{S}_H(\cdot)$ and $\mathcal{S}_Q(\cdot)$ coincide, then the \mathbf{z}_2 update step can be removed and semidefiniteness and nuclear norm minimization problem simultaneously handled proceeding as in Theorem 16 in [25].