

A Nonlinear SDP Approach for Matrix Rank Minimization Problem with Applications

Nasser Sadati

Mansoor Isvand Yousefi

Abstract— We consider the problem of minimizing rank of a matrix under linear and nonlinear matrix inequality constraints. This problem arises in diverse applications such as estimation, control and signal processing and it is known to be computationally NP-hard even when constraints are linear. In this paper, we first formulate the RMP as an optimization problem with linear objective and simple nonlinear semialgebraic constraints. We then proceed to solve the problem with augmented Lagrangian method known in nonlinear optimization. Despite of other heuristic and approximate methods in the subject, this method guarantees to find the global optimum in the sense that it does not depend on the choice of initial point for convergence. Several numerical examples demonstrate the effectiveness of the considered algorithm.

I. INTRODUCTION

GIVEN a real matrix $A \in S^n$ which entries are affine function of decision vector x , we are going to minimize the rank of this matrix under linear and nonlinear matrix inequality constraints. We refer to this problem as rank minimization problem and denote it by RMP hereafter. Due to Rank operator, this is a nonlinear nonconvex matrix constrained problem and is known to be computationally NP-hard. Unavoidably, this problem arises in diverse applications in mathematic and engineering so it is of great importance if one can solve this problem exactly or even approximately. In many applications, notions such as order, complexity, or dimension of a model or design can be express as a rank of a certain matrix. One example is reduced order controller design in which one is interested to find a linear time invariant controller of minimum order to stabilize a given LTI system. It is shown that this problem is a RMP in its original form. A more complete list of applications can be found in [1], [2]. In [1] authors have shown that many important output feedback problems can be cast as a RMP including stabilization by static output feedback, stabilization by reduced-order dynamic output feedback, reduced order H_∞ synthesis and reduced-order μ synthesis with constant scaling. Quadratic and bilinear matrix inequalities (BMI) arises in many areas especially in control and combinatorial optimization, can be easily cast as

a LMI feasibility problem under rank constraints which is closely related to RMP. On the other hand, in control, the BMIs have been extensively studied as general framework of many NP-hard problems. Specially, Safanov et al [3] had shown that BMI is a unifying formulation of wide range of system and control problems such as μ/k_m synthesis, decentralize control, robust gain scheduling and simultaneous stabilization. So RMP contain all these problems as a special case.

Recently there has been an increasing attempt to solve this problem efficiently or even costly. In spite of these large amounts of applications, there has not been proposed a suitable algorithm to solve general RMP problem which is mainly due to nonconvex nature of generic RMP and its NP-hardness. Instead, most attempts in this direction pursue two reasonable approaches: (1) find special classes of RMP that can be solved efficiently in polynomial-time (2) propose heuristic and approximate solutions often based on relaxation techniques for nonconvex hard problems. One important special class which efficiently can be solved is addressed in [4]. Although the authors showed that a certain problem in control engineering (fixed-order output feedback problem) can be formulated as this special structure, this special case certainly does not meet all the aforementioned applications. Some other special classes (less important) which can be solved efficiently, were reported in [2].

An example of the second approach was found in [2] where authors provide two heuristic based on convex optimization that approximately solve RMP. Particularly, they show that minimizing the nuclear norm of a matrix yields a global lower bound on the rank, provided that feasible set of the problem is bounded. They also provide another heuristic for general RMP that involves a nonconvex optimization problem which is not generally computationally tractable, unless a very good initial point is provided (for example, using trace heuristic). The accuracy of these approximations heavily depends on a special problem of interest and on the accuracy of initial points. Other heuristic methods include cone complementary method (coordinate descent) [5] and alternative projection [6], which is basically a convex analysis algorithm to find a common point of two convex set. The authors of [6] then utilized this algorithm to find a common point of a convex and a nonconvex set with the hope that algorithm converges for nonconvex case too. However this is not usually the case, except a very good initial point (which is itself another problem) is provided. All other methods almost exhibit the

N.Sadati is with the Electrical Engineering Department, Sharif University of Technology, Tehran, Iran (e-mail: sadati@sina.sharif.edu)

M.I.Yousefi is with the Electrical Engineering Department, Sharif University of Technology, Tehran, Iran (e-mail: mansoor@ee.sharif.edu).

same drawbacks, i.e. performance extremely depends on initial point and the specific problem to be solved so that the method often fails to converge in practice. This is not a priori known for what problem instances which method performs best and what initial point should be supplied and if the algorithm converges, we do not know that the attained point is indeed the solution of the problem.

In this paper we pursue a completely different approach based on nonlinear optimization techniques, motivated by the above drawbacks. Specially using semialgebraic representation of RMP [7], we propose a quasi-augmented Lagrangian optimization technique to find the global optimum of general RMP under LMI constraints. The main idea is that despite of classical augmented Lagrangian, to deal with linear matrix inequalities and avoid unfavorable nonsmoothness, we keep LMI constraints outside the Lagrangian and removing the nonlinear scalar constraints according to the penalization frameworks. For this purpose, a suitable nonlinear locally convex merit function is defined which has the same minimizes as the original problem. We then construct a sequence of quadratic models, each minimized under the LMI constraints. Each of these quadratic subproblems is easily reduced to a SDP using Schur complement lemma and solved via efficient available convex optimization methods. The main advantage of the method is that, despite of other methods in the subject, the considered scheme is globally convergent in the sense that it is not sensitive to the choice of initial point, can be easily modified to incorporate nonlinear matrix inequality constraints, and is completely reliable to be implemented in a software to solve any given general RMP. Moreover, it can be easily implemented via available semidefinite programming packages.

II. PROBLEM FORMULATION

Consider generic RMP

$$\begin{aligned} \min \quad & \text{Rank}(X(x)) \\ \text{subject to} \quad & A(x) < 0 \\ & g_j(x) = 0 \quad j = 1, \dots, m. \end{aligned} \quad (1)$$

Where $x \in R^n$ the vector of decision variables is, $X(x) \in R^{p \times q}$ is a real $p \times q$ matrix whose entries are affine function of x components, and $A(x)$ stands for a linear matrix inequality, i.e.

$$A(x) = A_0 + x_1 A_1 + \dots + x_n A_n \quad (2)$$

In which $A_i \in S^m, i = 0, 1, \dots, n$ are real symmetric $m \times m$ matrices. $g(x)$ corresponds to a nonlinear scalar equality constraints.

The following lemma, known as semidefinite embedding lemma, indicates that one can always correspond to any arbitrary nonsquare matrix, a positive semidefinite matrix whose rank is exactly twice the rank of X .

Lemma 1. Let $X \in R^{p \times q}$ be a given matrix. Then $\text{Rank}X \leq r$ if and only if there exist matrices $Y = Y^T \in R^{p \times p}$ and $Z = Z^T \in R^{q \times q}$ such that

$$\begin{aligned} & \text{Rank}Y + \text{Rank}Z \leq 2r \\ & \begin{bmatrix} Y & X \\ X^T & Z \end{bmatrix} \geq 0 \end{aligned} \quad (3)$$

Proof. The proof of this lemma can be found in [2].

In other words, any general RMP can be reduced to a special RMP with (square) positive semidefinite objective matrix at the cost of increasing problem dimension. Using this lemma, without loss of generality, we assume that in problem (1), X is a (square) positive semidefinite matrix. We now show that RMP can be cast as an optimization problem with linear objective and simple nonlinear semialgebraic equality constraints. Lemma2 is a suitable generalization of similar one recently considered in [7].

Lemma 2. Let $x \in R^n$, $A(x) \in S^m$ and $X(x) \in R^{p \times p} > 0$ are matrices whose entries are affine function of x components. There are nonnegative scalars $v_i, i = 1, 2, \dots, n$ such that the optimum value of

$$\begin{aligned} \min \quad & \text{Rank}(X(x)) \\ \text{subject to} \quad & A(x) < 0 \\ & g(x) = 0 \end{aligned} \quad (4)$$

And

$$\begin{aligned} \min \quad & \sum_{i=1}^n v_i \\ & (v_i - 1)\sigma_i(X) = 0 \\ & A(x) < 0, \quad X > 0 \\ & g(x) = 0 \quad v_i \geq 0, i = 1, 2, \dots, n \end{aligned} \quad (5)$$

Are equal, in which $\sigma_i(X)$ is the i 'th principle minor of X .

Proof. See [7].

Note that when entries of X are affine functions of x components, $\sigma_i(X)$ are polynomials in x , thus the constraint set in (5) defines a semialgebraic set which is well studied in algebraic geometry. Particularly, [7] cited that polynomial optimization problem (5) (without LMI and nonlinear constraints) may be solved utilizing recent results on sum of square polynomial relaxations. However, typically one minor has a large degree and as remarked by [7], the order of moment matrices greatly grows which significantly increases the size of these relaxations. In the next section, we will show how this formulization can be employed in a nonlinear optimization technique to solve general RMP.

III. AUGMENTED LAGRANGIAN

In this section we extend the classical augmented Lagrangian to the case of LMI constraints to solve (5). The augmented Lagrangian method was originally proposed by Powell [8] in 1969 and is considered to be an important option in numerical optimization, to deal with nonlinear equality constraints. Our presentation is inspired by the works of [9],[10] in which authors consider a similar

approach to tackle quadratic nonlinear terms in nonlinear optimization and robust control. In augmented Lagrangian method, both equality and inequality constraints are embedded in objective function with a suitable structure [11],[12]. The main difficulty with AL method is that removing inequality constraints introduces a nondifferentiable term that enforces to use nonsmooth optimization techniques, such as cutting plane or Bundle-type methods, which are not very efficient. Fortunately, inequality constraints in problem (5) are linear and this structure can be further exploited to greatly simplify the solution method. Conn [13] remarked the potential advantages of this strategy. He proposed to keep these linear inequalities constraints explicitly outside the augmented Lagrangian, handling them directly in the level of subproblem minimization. This idea was extended in [10] to matrix value constraints, similar to the one considered here. In their setting, original nonlinear nonconvex hard problem was reduced to a sequence of easily tractable convex subproblems and solved efficiently via semidefinite programming. The major advantage of this approach is that we are able to use extensive well-developed theory of SDPs and many efficient available software packages.

Consider again problem (5). We associated with this problem the following quasi-augmented Lagrangian [11]

$$L(x; \lambda, \mu) = e^T v + \sum_{i=1}^m \lambda_i g_i(x) + (\rho/2) \sum_{j=1}^m g_j^2(x) + \sum \mu_i \sigma_i(X)(1-v_i) + (\delta/2) \sum \sigma_i^2(X)(1-v_i)^2 \quad (6)$$

Where λ and μ are known as Lagrange multiplier estimates, ρ and δ are known as penalty parameters and $e = [1 \ 1 \ \dots \ 1]^T$ (see for instance [11], [12]). Following the idea of Conn [13], for a given ρ and δ we replace problem (5) with the following approximation

$$\min_x L(x; \lambda^*, \mu^*, \rho, \delta) \quad (7)$$

$$A(x) < 0, \quad X(x) > 0$$

In which (λ^*, μ^*) are the solution of the associated dual problem:

$$\max_{\lambda, \mu} L(x^*; \lambda, \mu, \rho, \delta) \quad (8)$$

The theory behind classical augmented Lagrangian is brilliantly clarified in [12] using duality theory. Accordingly, quadratic terms are employed to convexify the Lagrangian, which holds for large enough value of penalty parameters. For this value of penalty parameters keep fixed, Lagrange function is strongly convex in a neighborhood of x^* , and (x^*, λ^*, μ^*) is a saddle point of Lagrange function. As ρ and δ go to infinity, problem approaches penalty/barrier method [11][12], thus would be unavoidably numerically ill-conditioned and extremely difficult to solve by any optimization method sensitive to conditioning. Augmented Lagrangians are appealing for avoidance of this ill-conditioning, because penalty parameter needs to be only larger than a threshold, but not necessary approaches infinity. This threshold is proportional to inverse of constraint violation. Note that to solve (7), one must already

know the solution of (8) which itself depend on (7). This can be handled using an iterative procedure: starting from a suitable initial point, the solution of each problem is used in the other problem to update the corresponding variable. When solving primal problem, despite of dual problem, the value of Lagrangian and its gradient and Hessian is almost independent of dual variables near the optimal solution x^* since $g(x) \approx 0$. However, when solving dual problem (8) (the upper level process) we are less flexible in choosing optimization method, since the information on the dual objective is "implicit" (to get its value and gradient, we should solve an auxiliary primal optimization problem that is normally computationally expensive and things like line search become very costly). It can be shown that a simple first order updating rule in outer iterations as

$$\lambda^{j+1} = \lambda^j + \rho g(x_j) \quad (9)$$

Guarantee convergence to the optimal value [12], [9]. We start the procedure with certain ad hoc of ρ and δ and look whether the residuals $|g(x_j)|$ indeed reasonably decreased with j , say by a factor of 0.2. Until this the so, we keep fix the penalty parameter, otherwise we increase the penalty by a certain factor, say 10. We have not yet specified in which way the minimization in problem (7) is done. The standard choice, when the optimal point is non-degenerate, is of course the modified Newton method with line search or with trust region strategy. Newtonian methods yield extremely fast local quadratic Q-convergence to a non-degenerate point, provided that initial point is "closed enough" to the optimal solution. A difficulty with Newton-based methods immediately arises when the Hessian $\nabla^2 L$ is not positive definite, mainly due to initialization. Incorporating line search, one may expect to prove the global convergence of the Newton method with (Armijo) line search only if Newton direction be a descent direction at a non-critical x

$$\nabla f(x) \neq 0 \Rightarrow -(\nabla f)^T [\nabla^2 f]^{-1} (\nabla f) < 0 \quad (10)$$

This holds for convex functions (or more generally elliptic) with non-singular Hessian and bounded initial sublevel set, but not for our nonlinear nonconvex problem (7). To remove this difficulty, Variable Metric methods was suggested, which take a positive definite approximation, albeit rough during the early stages of the descent process, when x_k is far from minimum. This algorithm starts from some positive definite operator W_k and works in parallel with the construction of $\{x_k\}$, accumulates second order information from objective to construct a sequence $\{W_k\}$ of positive definite operators. Only at the end of the process, need these operators approach the desired W^* , whatever it is. W_k can be obtained from the solution of secant equation. Here, we adopted the updating policy

$$W_{k+1} = \begin{cases} \nabla^2 f(x_k) & \text{if } \nabla^2 f(x_k) > 0 \\ \text{a modified version of } \nabla^2 f(x_k) & \text{otherwise} \end{cases} \quad (11)$$

This means that for remote points where Hessian probably is not positive definite we use a modified version of $\nabla^2 f(x_k)$, whereas for near optimal points we

set $w_{k+1} = \nabla^2 f(x_k)$. A number of modifications for $\nabla^2 f(x_k)$ have been introduced, originally in the context of positive definite approximation of matrices [12]. Modified Cholesky factorization seems to be the most appealing method due to numerical stability and low arithmetic cost ($\frac{1}{6}n^3(1+o(1))$ arithmetic operation). In this method a proper diagonal matrix is added to correct the actual Hessian such that (1)if H_k is safely positive definite, $\Delta=0$ (2) if H_k is indefinite, $\|A\|_\infty$ is not much greater than of the most negative eigenvalue of H_k , $\lambda_1(H_k)$ (3) H_k is reasonably well-conditioned (4) the cost of the factorization is only a small multiple of n^2 operations more than $O(n^3)$ cost of standard cholesky factorization. A numerically stable modified cholesky factorization was proposed by [14] and subsequently refined by the same authors. This version has been widely used in numerical optimization since its inception. In this recursive algorithm at the beginning of step j , a $(n-j+1) \times (n-j+1)$ submatrix H_j remains to be factored. Partitioning H_j as

$$H_j = \begin{bmatrix} \alpha_j & h_j^T \\ h_j & \hat{H}_j \end{bmatrix} \quad (12)$$

Where $\alpha_j \in R$ is the j th diagonal element called the pivot, h_j the current vector of elements in the column j below the diagonal and $\hat{H}_j \in R^{(n-j) \times (n-j)}$. The algorithm chooses a nonnegative value Δ_j and then computes

$$L_{jj} = \sqrt{\alpha_j + \Delta_j} \quad L_{ij} = (h_j)_i / L_{jj} \quad i = j+1, \dots, n. \quad (13)$$

$$H_{j+1} = \hat{H}_j - \frac{h_j h_j^T}{\alpha_j + \Delta_j}$$

Challenge in the modified cholesky factorization methods is choosing each Δ_j , as it seems to utilize a kind of look-ahead of $H_k + \Delta$. The algorithm in [14] chooses each Δ_j to be the smallest nonnegative number for which:

$$0 \leq \frac{(\max_i (h_j)_i)^2}{\alpha_j + \Delta_j} \leq \beta^2 \quad (14)$$

Here β is a constant upper bound for which aforementioned conditions (1) and (2) hold. As discussed in [14]:

$$\beta^2 = \max \left\{ \gamma, \zeta / \sqrt{n^2 - 1}, \varepsilon \right\} \quad \text{where } \gamma = \max_i |H_{ii}| \text{ and } \zeta = \max_{j < i} |H_{ij}|$$

Where ε is machine zero. With this setting, the smallest upper bound on $\|A\|$ is

$$n^2 \gamma + 2(n-1)\zeta \quad (15)$$

Which is obtained when $\beta^2 = \zeta / \sqrt{n^2 - 1}$. Since factorization is performed in each inner iteration, this bound clearly confirm that Newton method keep on successful and the approximation error does not affect the overall convergence less than superlinear (which is still very satisfactory). Moreover, when there is a special structure such as sparsity, diagonality or Hankel, the factorization may be done more

Algorithm

(Initializations) Initialize the algorithm by finding the feasible point of LMI constraints. Initialize $\rho_1^1, \delta_1^1, \lambda_1^1, \mu_1^1, \alpha_1, \alpha_2, \beta_1, \beta_2$ and ε . set $j=1$.

(Outer iteration)

for $j=1,2,\dots$ do until convergence

(Stopping test)

set $e = [(1-v_1)\sigma_1(x), \dots, (1-v_n)\sigma_n(x)]^T$
stop the outer iterations if $\|g\|_2^2 + \|e\|_2^2 < \varepsilon^2$

(Inner iteration: Lagrangian minimization)

for $k=1,2,\dots$

given x_k , calculate $g_k^j = \partial L(x_k, \lambda_k^j, \mu_k^j; \rho_k^j, \delta_k^j) / \partial x \big|_{x_k}$

and $H_k^j = \partial^2 L(x_k, \lambda_k^j, \mu_k^j; \rho_k^j, \delta_k^j) / \partial^2 x \big|_{x_k}$

if $\|g_k^j\| < \varepsilon$

stop the inner iteration

end

if H_k^j is not positive definite or near singular

then perform the modified Cholesky factorization

as mentioned before to find diagonal matrix Δ_k^j . set

$$w_k^j = H_k^j + \Delta_k^j$$

end

solve optimization problem with initial condition x_k

$$\min_s s^T g_k^j + \frac{1}{2} s^T w_k^j s$$

$$s^T A(x_k + s) < 0$$

denote the solution of this problem s_k^j

do a line search along s_k^j :

$$\alpha_k^* = \arg \min_{\alpha_k} L(x_k^j + \alpha_k s_k^j, \lambda_k^j, \mu_k^j; \rho_k^j, \delta_k^j)$$

set $x_{k+1}^j = x_k^j + \alpha_k^* s_k^j$

end

set $x_j = x_k^j$

(Multiplier/penalty updating)

$$\lambda^{j+1} = \lambda^j + \rho^j \cdot g(x_j)$$

$$\mu^{j+1} = \mu^j + \delta^j \cdot (1-v^j) \sigma(X_j)$$

$$\rho^{j+1} = \begin{cases} \alpha_1 \rho^j & \text{if } \|(1-v^{j+1})\sigma(x_{j+1})\| > \beta_1 \|(1-v^j)\sigma(x_j)\| \\ \rho^j & \text{otherwise} \end{cases}$$

$$\delta^{j+1} = \begin{cases} \alpha_2 \delta^j & \text{if } \|(1-v^{j+1})\sigma(X_{j+1})\| > \beta_2 \|(1-v^j)\sigma(X_j)\| \\ \delta^j & \text{otherwise} \end{cases}$$

end

efficiently with a very low cost [15]. Finally, note that one may use a Gauss-Newton approximation of Hessian matrix instead of the actual one, provided that already calculated a small number of principle minors using for example, polynomial interpolation techniques. This is especially

applicable when the current iterate is near the optimal solution where disregarded term is negligible.

Summarizing, the resulted algorithm for the proposed method is shown in the table of previous page. Note that according to the Schur complement lemma, the nonconvex quadratic optimization problem in inner iteration is indeed a (convex) semidefinite programming problem [15] and is easily solved via available efficient interior point solvers such as SeDuMi1.02 or YALMIP [16].

IV. NUMERICAL RESULTS

In this section the performance of the proposed method is demonstrated via several numerical examples. The numerical results are obtained using a 2GHZ Pentium 4 Pc, with 1G memory. We used SeDuMi semidefinite programming solver [16] to solve the resulting SDPs, build in Matlab 6.5.

Example1. As a first simple example, consider the minimizing the rank of

$$X = \text{diag} \left(\begin{bmatrix} 10-x_2 & x_1 \\ x_1 & 10+x_2 \end{bmatrix}, \begin{bmatrix} x_2 & x_1-2 \\ x_1-5 & 1 \end{bmatrix}, x_1+x_2-10 \right) \quad (16)$$

With $x = [x_1 \ x_2]^T \in \mathbb{R}^2$. We run the proposed algorithm starting from feasible initial point $[3.9312 \ 8.2775]^T$ obtained by solving convex LMI feasibility problem $X > 0$. Set parameters $\delta = 150, \alpha_1 = \alpha_2 = 15, \beta_1 = \beta_2 = 0.5, \varepsilon = 10^{-5}$. We consider numbers less than 10^{-10} to be zero. Note that scalar equality constraints does not exist in this case, and the only LMI constraints are $X > 0$ and $\text{diag}(v_i) > 0$. There are 31 nonlinear equality constraints all eliminated using the AL method, two LMI constraint and 33 variable. The algorithm converges in two iteration to the optimal point $[0 \ 10]^T$ which has rank 2. Resulted equality constraints are satisfied up to the 10^{-5} precision. However, for this problem alternative projection method [6] started from random initial point diverge in this case that means these schemes are very sensitive to the supplied initial points and this is not known a priori for what problems what initial points are appropriate (note that problem (16) is not symmetric).

Example2. In this example we apply the augmented Lagrangian method to the problem of H_∞ model reduction. This problem has been taken from [17]. Consider a discrete time linear time invariant plant P described by following state space equations

$$\begin{aligned} x_{k+1} &= Ax_k + B_1 w_k + B_2 u_k \\ z_k &= C_1 x_k + D_{11} w_k + D_{12} u_k \\ y_k &= C_2 x_k + D_{21} w_k + D_{22} u_k \end{aligned} \quad (17)$$

Where $x \in \mathbb{R}^n, w \in \mathbb{R}^{n_w}, u \in \mathbb{R}^{n_u}$. assume that $(A \ B_2 \ C_2)$ is stabilizable and detectable and plant P is proper, i.e. $D_{22} = 0$. The minimum order H_∞ controller design consists of finding a linear time invariant controller K such that:

- (1) the close loop system is internally stable.
- (2) the H_∞ norm of transfer function from input w to the output z is strictly less than a given γ .

It can be shown such a controller exist with order $n_k \leq n$ if and only if there exist symmetric positive definite matrices $R, S \in \mathbb{R}^{n \times n}$ such that [18]

$$\begin{aligned} \begin{bmatrix} R & I \\ I & S \end{bmatrix} &\geq 0 \\ \begin{bmatrix} N_{12} & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} ARA^T - R & ARC & B \\ CRA & -\gamma I + CRC & D \\ B & D & -\gamma \end{bmatrix} \begin{bmatrix} N_{12} & 0 \\ 0 & I \end{bmatrix} &< 0 \\ \begin{bmatrix} N_{12} & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} ASA - S & ASB & C \\ BSA & -\gamma I + BSB & D \\ C & D & -\gamma \end{bmatrix} \begin{bmatrix} N_{12} & 0 \\ 0 & I \end{bmatrix} &< 0 \\ \text{Rank} \begin{bmatrix} R & I \\ I & S \end{bmatrix} &\leq n + n_k \end{aligned} \quad (18)$$

Reduced order stable H_∞ controller is thus a RMP:

$$\begin{aligned} \min \text{Rank} \begin{bmatrix} R & I \\ I & S \end{bmatrix} \\ \text{under LMI constraints (18)} \end{aligned} \quad (19)$$

Based on the solutions of this problem, one can immediately reconstruct H_∞ controller by solving another linear matrix inequality as in [18] (or even analytical formula). We apply the augmented Lagrangian Method to problem (19) and provide some numerical results. Using the prediction-error identification method in MATLAB identification toolbox, a 4-order closed loop linear system is generated for each subsystem. System dynamic and numerical data $A \in \mathbb{R}^{4 \times 4}, B_1 \in \mathbb{R}^{4 \times 1}, B_2 \in \mathbb{R}^{4 \times 1}, C_1 \in \mathbb{R}^{3 \times 4}, C_2 \in \mathbb{R}^{2 \times 4}$

$D_{11} \in \mathbb{R}^{3 \times 1}, D_{12} \in \mathbb{R}^{3 \times 1}, D_{21} \in \mathbb{R}^{2 \times 1}$ all calculated using this linear model. The overall order of the MIMO system is 16. The objective matrix is a 8×8 positive semidefinite matrix $(R, S \in \mathbb{R}^{4 \times 4})$. There are 128 nonlinear constraints, all eliminated with augmentation. For those components which directly depend on principle minors, derivatives are computed with the aim of interpolation techniques with uniform complex sampling. Constant matrices in (18) all are calculated according to the associated transfer function. We run the algorithm with $\delta = 100, \alpha_1 = \alpha_2 = 10, \beta_1 = \beta_2 = 0.3, \varepsilon = 10^{-5}$. Table I depicts the performance of the augmented Lagrangian method for reduced order controller design. We first disregard rank constraint and minimize γ under LMI constraints. The result is $\gamma_{opt} = 1.8102$ with a 16 order controller. We then increase γ and solve the optimization problem (19), finding the order of the resulted controller. Table I illustrates that beyond $\gamma = 7$ the order of the controller is not reduced significantly. Specially this means that in the limit case where $\gamma \rightarrow \infty$, the aforementioned objectives (1) and (2) reduce to design merely a stabilizing controller which according to formulization of [18] has degree at least 9. In other words, with proper design, one can reject disturbances smaller than $\gamma = 1/7$ using a controller with a degree as much as the degree of the corresponding stabilizer controller (note that the merely stabilizing controller is not unique, however, typically the stabilizing controllers find with LMI-based formulizations

does not achieve the performance level of asymptotic problem (19). Particularly, our simulation revealed that for the given system, the stabilizing controller only reject disturbances up to the $\gamma = 2.3070$. The price paid for this is the “amount” of stability that can be exactly take into account with decay rate notions). Moreover, the trade off between performance level γ and controller order n_k can be made precise using multiobjective optimization. Finally, we remark that the solution of inner iterations does not need not to be very exact in the early steps of the algorithm, because in outer iterations multipliers are far away of the optimal multipliers. As the outer multipliers approach the optimal ones, the accuracy of inner iterations needs to be improved. This is with analogy of two-level barrier paths following interior point algorithm in which the trade off between number of inner and outer iterations are discussed. This may lead to short-step and long-step variants, with proper choice of penalty parameters.

V. CONCLUSION

A method based on nonlinear nonconvex optimization for matrix rank minimization was proposed. The proposed method is globally convergent in the sense that it is not sensitive to the choice of the initial point. Further research is needed to demonstrate the trade off between performance and computational complexity, both numerically and analytically. Particularly, this paper reveals that recent advances in nonlinear optimization open another promising approach to solve broad class of problems in system and control theory.

REFERENCES

- [1] L.EL Ghaoui and P.Gahinet. “ Rank minimization under LMI constraints: framework for output feedback problems”. In *Proc.European Control Conf., The Netherlands, 1993*.
- [2] M.Fazel, *Matrix Rank Minimization with applications, Phd Thesis, Stanford, 2002*.
- [3] M.G.Safanov, K.C.Goh, and J.H.Ly. “Control system synthesis via bilinear matrix inequalities”, In *Proceedings of American Control Conference, Volume 1, pages 45-9, Baltimore, MD, 1994*.
- [4] M.Mesbahi, G.P.Papavasilopoulos. “ On the rank minimization problem over a positive semidefinite linear matrix inequality”,*IEEE Transaction on Automatic Control, AC-42(2):239-43, February 1997*.
- [5] L. EL.Gaoui, F.Oustry, and M.Ait Rami, “ A cone complementary linearization algorithm for static output-feedback and related problems”, *IEEE Transaction on Automatic Control,AC-42(8):1171-1176,1997*.
- [6] K.M.Grigoriadis and E.B.Beran, “Alternative projection algorithms for linear matrix inequalities problems with rank constraints”, *SIAM, pp.251-267, 1999*.
- [7] A. d’Aspremont “A Semidefinite Representation for some Minimum Cardinality Problems”, *Proceedings. 42nd IEEE Conference on Decision and Control, Vol. 5, pp. 4985-4990, December 2003*.
- [8] M.J.D.Powell, *A method for nonlinear constraints in minimization problems, in Optimization, R.Fletcher(ed), Academic press, London,1969*.
- [9] B.Fares, P.Apkarian, D.Noll, “An Augmented Lagrangian method for a class of LMI constraint problems in robust control theory”,*Int.J of Robust Control,vol.74,no.4,pp.2974-2984,2001*.
- [10] D.Noll and M.Torki and P.Apkarian. “Partially augmented lagrangian method for matrix inequalities constraints”, *SIAM Journal on Optimization, vol.15 (1):161:184, 2004*.

- [11] D.P.Bertsekas, *Nonlinear programming, 2nd Edition, Athena Scientific, 2000*.
- [12] A.Nemirovski, *Lecture notes on optimization II: Standard numerical methods for nonlinear continuous optimization, Technion-Israel Institute of Technology, spring 1999*.
- [13] A.R.Conn, “Convergence properties of an augmented Lagrangian algorithm for optimization with a combination of general equality and linear constraints”, *SIAM J.Optim.,vol.6,pp.674-703, 1996*.
- [14] P.E.Gill and W.Murray, “Newton type methods for unconstraint and linearly constraint optimization”,*Math.Program.,1974(27):311-350*.
- [15] S.P.Boyd, L.Vandenberghe, *Convex Optimization, Cambridge University Press, 2004*.
- [16] J.Sturm, “Using SeDuMi 1.02,a MATLAB toolbox for optimization over symmetric cones”,*Optimization methods and software,vol.11-12, pp.625-653,1999,special issue on Interior Point Methods (CD supplement with software)*.
- [17] B.T.Bupp, D.S.Bernstein and V.T.Coppola, “A benchmark problem for nonlinear control design: problem statement, experimental testbed and passive nonlinear compensation”,in *Proc. American Control Conference,pp.4363-7,1995*.
- [18] P.Gahinet and P.Apkarian. “ A linear matrix inequality approach to H_∞ contro”. *Int.J.Robust and Nonlinear Control, 4:421-488,1994*.

γ	1.8021	2.5	3.5	5	7
n_k	16	14	14	12	11
#inner iterations	*	8	6	8	11
#outer iterations	*	22	17	19	12

Table I. Trade off between performance level, controller order and amount of computation.