algorithms

MDPI

Article

Projection onto the Set of Rank-Constrained Structured Matrices for Reduced-Order Controller Design

Masaaki Nagahara [1,*], Yu Iwai [2] and Noboru Sebe [3]

[1] Institute of Environmental Science and Technology, The University of Kitakyushu, Fukuoka 808-0135, Japan
[2] Faculty of Environmental Engineering, The University of Kitakyushu, Fukuoka 808-0135, Japan
[3] Faculty of Computer Science and Systems Engineering, Kyushu Institute of Technology, Fukuoka 820-8502, Japan
* Correspondence: nagahara@ieee.org

Abstract: In this paper, we propose an efficient numerical computation method of reduced-order controller design for linear time-invariant systems. The design problem is described by linear matrix inequalities (LMIs) with a rank constraint on a structured matrix, due to which the problem is non-convex. Instead of the heuristic method that approximates the matrix rank by the nuclear norm, we propose a numerical projection onto the rank-constrained set based on the alternating direction method of multipliers (ADMM). Then the controller is obtained by alternating projection between the rank-constrained set and the LMI set. We show the effectiveness of the proposed method compared with existing heuristic methods, by using 95 benchmark models from the COMPL$_e$ib library.

Keywords: reduced-order control; rank constraint; linear matrix inequality; alternating projection; convex optimization

check for updates

Citation: Nagahara, M.; Iwai, Y.; Sebe, N. Projection onto the Set of Rank-Constrained Structured Matrices for Reduced-Order Controller Design. *Algorithms* **2022**, *15*, 322. https://doi.org/10.3390/a15090322

Academic Editors: Dunhui Xiao and Shuai Li

Received: 28 July 2022
Accepted: 5 September 2022
Published: 9 September 2022

1. Introduction

It is well known that a stabilizing output-feedback controller and an H^∞ controller of a linear time-invariant system can be obtained by solving linear matrix inequalities (LMIs), assuming that the order of the controller is more than or equal to that of the controlled plant model [1,2]. Since the set of optimization variables described by LMIs is convex, the problem can be efficiently solved by convex optimization solvers such as Sedumi [3], SDPT3 [4], and MOSEK [5]. Also, LMIs are easily coded with YALMIP [6] and CVX [7] on MATLAB, and CVXPY [8] on Python.

Practically, it is preferred for implementation to use a low-order controller, especially a static controller of a high-order plant, which we call a *reduced-order controller*. To obtain a reduced-order controller that has a lower order than the plant is however known to be NP-hard [9] due to a rank constraint [10]. Therefore, we need to employ a heuristic method to efficiently obtain an *approximated* reduced-order controller. Actually, a couple of heuristic methods have been proposed; the XY-centring algorithm [11], the cone complementarity linearization algorithm [12], and alternating projection methods [13,14], to name a few.

More recently, the nuclear norm minimization with LMIs has been proposed to cope with this hard problem [15–19]. This is based on the fact that the nuclear norm of a matrix well approximates the matrix rank [20]. Since the nuclear norm is a convex function and the set described by LMIs is also a convex set, the problem boils down to a convex optimization problem that can be solved very efficiently. The nuclear norm heuristic has been recently applied to, e.g., principal component analysis [21], image denoising [22,23], and system identification [24,25].

Although the nuclear norm heuristic is widely used for rank-constrained problems, we show by numerical examples in this paper that this is not necessarily efficient for reduced-order controller design. Instead, we propose a new method to solve the reduced-order controller design problem by extending the alternating projection method proposed in [13].

The idea is to compute a more precise projection onto the set of rank-constrained structured matrices by the alternating direction method of multipliers (ADMM) [26] . By numerical examples in Section 4, we show that the proposed method significantly improves the precision of the solution compared to the nuclear norm minimization [15] and the original alternating projection method [13].

The organization of this paper is as follows: In Section 2, we show two reduced-order control problems that are described as rank-constrained LMI problems. In Section 3, we propose the alternating projection algorithm to solve the rank-constrained LMI problem. Numerical examples are shown in Section 4 to illustrate the effectiveness of the proposed method with 95 benchmark models from the COMPL$_e$ib library [27]. A summary is given in Section 5.

We note that the MATLAB programs to check the numerical examples and the results of stability tests for 95 benchmark models shown in Section 4 are available at the web page of [28].

Notation

Let A be a matrix. The transpose of A is denoted by $A^\top$, the trace by $\mathrm{tr}(A)$, and the rank by $\mathrm{rank}(A)$. The i-th singular value of A is denoted by $\sigma_i(A)$. In this paper, we use two kinds of matrix norms: one is the Frobenius norm $\|A\|$ of A is defined by

$$\|A\| \triangleq \sqrt{\mathrm{tr}(A^\top A)} = \sqrt{\sum_{i=1}^{n} \sigma_i^2(A)}. \tag{1}$$

Matrix inequalities $A \succ 0$, $A \succeq 0$, $A \prec 0$, and $A \preceq 0$ respectively mean A is positive definite, positive semidefinite, negative definite, and negative semidefinite. For $A \in \mathbb{R}^{n\times m}$ with $r = \mathrm{rank}(A) < n$, $A^\perp$ is a matrix that satisfies

$$A^\perp \in \mathbb{R}^{(n-r)\times n}, \quad A^\perp A = 0, \quad A^\perp A^{\perp\top} \succ 0. \tag{2}$$

By $\mathcal{S}_n$, we denote the set of $n \times n$ real symmetric matrices.

For a closed subset Ω of $\mathcal{S}_n$, the projection operator of $X \in \mathcal{S}_n$ onto Ω is denoted by Π_Ω, that is,

$$\Pi_\Omega(X) \in \arg\min_{Z\in\Omega} \|Z - X\|, \tag{3}$$

and the distance from X to Ω is defined by

$$\mathrm{dist}(X, \Omega) \triangleq \min_{Z\in\Omega} \|Z - X\|. \tag{4}$$

2. Reduced-Order Controller Design Problems

In this section, we show two examples of reduced-order controller design.

2.1. Reduced-Order Stabilizing Controllers

Let us consider the following linear time-invariant system:

$$\dot{x}(t) = Ax(t) + Bu(t), \quad y(t) = Cx(t), \quad t \geq 0, \tag{5}$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$, $y(t) \in \mathbb{R}^p$, $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, and $C \in \mathbb{R}^{p\times n}$. For this system, we consider an output-feedback controller $u = Ky$, whose order is assumed to be $n_c < n$. Then, the reduced-order output-feedback controller design is described as the following feasibility problem [10].

Problem 1 (Stabilizing controller)**.** *Find $X_1, X_2 \in \mathcal{S}_n$ such that the rank constraint*

$$\mathrm{rank}\begin{bmatrix} X_1 & I \\ I & X_2 \end{bmatrix} \leq n + n_c, \tag{6}$$

and LMIs

$$-\begin{bmatrix} X_1 & I \\ I & X_2 \end{bmatrix} \preceq 0, \tag{7}$$

$$B^{\perp}(AX_1 + X_1 A^{\top})B^{\perp\top} \preceq -\epsilon I, \tag{8}$$

$$C^{\top\perp}(X_2 A + A^{\top} X_2)C^{\top\perp\top} \preceq -\epsilon I \tag{9}$$

hold for some $\epsilon > 0$

We note that the inequality "$\preceq -\epsilon I$" can be "$\prec 0$," however for the projection-based algorithm described in Section 3, we introduce small $\epsilon > 0$ to make the subsets closed.

2.2. Reduced-Order H^∞ Controllers

Let us consider the following generalized plant:

$$G: \begin{bmatrix} \dot{x}(t) \\ z(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} A & B_1 & B_2 \\ C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ w(t) \\ u(t) \end{bmatrix}, \quad t \geq 0, \tag{10}$$

where we assume that (A, B_2) is stabilizable and (C_2, A) is detectable. Let T_{zw} denote the feedback connection (or the linear fractional transformation) [29] of G and a controller K such that $u = Ky$. Then, the problem is to seek a controller K of order $n_c \leq n$ such that the H^∞ norm of T_{zw} satisfies $\|T_{zw}\|_\infty < \gamma$ with a given $\gamma > 0$. This problem is described as LMIs with a rank constraint [10,30].

Problem 2 (H^∞ controller)**.** *Find* $X_1, X_2 \in \mathcal{S}_n$ *such that the rank constraint* (6)*, the LMI* (7)*, and the following LMIs:*

$$\begin{bmatrix} B_2 \\ D_{12} \\ 0 \end{bmatrix}^{\perp} \begin{bmatrix} AX_1 + X_1 A^{\top} & X_1 C_1^{\top} & B_1 \\ C_1 X_1 & -\gamma I & D_{11} \\ B_1^{\top} & D_{11}^{\top} & -\gamma I \end{bmatrix} \begin{bmatrix} B_2 \\ D_{12} \\ 0 \end{bmatrix}^{\perp\top} \preceq -\epsilon I, \tag{11}$$

$$\begin{bmatrix} C_2^{\top} \\ D_{21}^{\top} \\ 0 \end{bmatrix}^{\perp} \begin{bmatrix} X_2 A + A^{\top} X_2 & X_2 B_1 & C_1^{\top} \\ B_1^{\top} X_2 & -\gamma I & D_{11}^{\top} \\ C_1 & D_{11} & -\gamma I \end{bmatrix} \begin{bmatrix} C_2^{\top} \\ D_{21}^{\top} \\ 0 \end{bmatrix}^{\perp\top} \preceq -\epsilon I \tag{12}$$

hold for some $\epsilon > 0$*.*

3. Algorithms

In this section, we propose a new algorithm based on the projection onto the set of rank-constrained structured matrices.

3.1. Proposed Algorithm

First, we define function $F(X_1, X_2)$ such that $F(X_1, X_2) \preceq 0$ is equivalent to the LMIs to be solved. For Problem 1 for example, $F(X_1, X_2) \preceq 0$ means that the LMIs (7)–(9) hold. Namely, we consider the following problem including Problems 1 and 2.

Problem 3. *Find a pair of matrices* $(X_1, X_2) \in \mathcal{S}_n^2$ *such that*

$$\operatorname{rank}\begin{bmatrix} X_1 & I \\ I & X_2 \end{bmatrix} \leq r, \quad F(X_1, X_2) \preceq 0. \tag{13}$$

Then, we propose alternating projection [31] to solve the rank-constrained LMI problems. For this, we define the two closed subsets of $\mathcal{S}_n^2$:

$$\Omega_r \triangleq \left\{ (X_1, X_2) \in \mathcal{S}_n^2 : \operatorname{rank} \begin{bmatrix} X_1 & I \\ I & X_2 \end{bmatrix} \leq r \right\}, \tag{14}$$

$$\Lambda \triangleq \left\{ (X_1, X_2) \in \mathcal{S}_n^2 : F(X_1, X_2) \preceq 0 \right\}. \tag{15}$$

Then, the problem is that of finding a pair of matrices (X_1, X_2) in $\Omega_r \cap \Lambda$ with $r = n + n_c$. For this, we adapt alternating projection between Ω_r and Λ. The iterative algorithm is given by

$$\begin{aligned} Z[k] &= \Pi_{\Omega_r}(X[k]), \\ X[k+1] &= \Pi_\Lambda(Z[k]), \quad k = 0, 1, 2, \ldots, \end{aligned} \tag{16}$$

where $X[0] = (X_1[0], X_2[0]) \in \mathcal{S}_n^2$ is a given initial guess of (X_1, X_2). The computation of the projection operators Π_{Ω_r} and Π_Λ are shown in the next following subsections.

We show a stability result for this algorithm as follows:

Lemma 1. *The sequences $\{X[k]\}$ and $\{Z[k]\}$ generated by (16) satisfy the following inequalities:*

$$\begin{aligned} \|X[k+1] - Z[k+1]\| &\leq \|X[k] - Z[k]\|, \\ \operatorname{dist}(X[k+1], \Omega_r) &\leq \operatorname{dist}(X[k], \Omega_r), \\ \operatorname{dist}(Z[k+1], \Lambda) &\leq \operatorname{dist}(Z[k], \Lambda), \quad k = 0, 1, 2, \ldots \end{aligned} \tag{17}$$

where dist is the distance function defined in (4).

The proof is given in Appendix A.

Remark 1. *We can also adopt the Dykstra algorithm [32] that gives an element in $\Omega_r \cap \Lambda$ that is a* projection *(i.e., one of the nearest points) on $\Omega_r \cap \Lambda$ from the initial guess $X[0] \in \mathcal{S}_n^2$. The algorithm is described as follows:*

$$\begin{aligned} Z[k] &= \Pi_{\Omega_r}(X[k] + P[k]), \\ P[k+1] &= X[k] + P[k] - Z[k], \\ X[k+1] &= \Pi_\Lambda(Z[k] + Q[k]), \\ Q[k+1] &= Z[k] + Q[k] - X[k+1], \quad k = 0, 1, 2, \ldots, \end{aligned} \tag{18}$$

where we set $P[0] = Q[0] = 0$.

3.2. Projection onto the Set Ω_r of Rank-Constrained Structured Matrices

Here we consider the projection of (X_1, X_2) onto the set Ω_r of rank-constrained structured matrices in (14). This projection can be written by definition as

$$\Pi_{\Omega_r}(X_1, X_2) \in \underset{(Z_1, Z_2) \in \Omega_r}{\arg\min} \ \|Z_1 - X_1\|^2 + \|Z_2 - X_2\|^2. \tag{19}$$

We note that, since the set Ω_r is closed but non-convex, multiple solutions may exist for the minimization in (19) may exist.

For this, we propose a precise projection based on alternating direction method of multipliers (ADMM). For the minimization problem (19), we introduce the indicator function $\mathcal{I}_r$ defined by

$$\mathcal{I}_r(Z) \triangleq \begin{cases} 0, & \text{if } \operatorname{rank}(Z) \leq r, \\ +\infty, & \text{otherwise.} \end{cases} \tag{20}$$

Then the minimization problem in (19) is equivalently described as

$$\begin{aligned} \underset{(Z_1,Z_2)\in\mathcal{S}_n^2,\tilde{Z}\in\mathcal{S}_{2n}}{\text{minimize}} \quad & \|Z_1 - X_1\|^2 + \|Z_2 - X_2\|^2 + \mathcal{I}_r(\tilde{Z}) \\ \text{subject to} \quad & \tilde{Z} = \begin{bmatrix} Z_1 & I \\ I & Z_2 \end{bmatrix} \end{aligned} \tag{21}$$

The convergence of the ADMM algorithm of (21) with nonconvex constraint (20) is discussed in [33].

To solve this optimization problem, we first consider the projection $\Pi_{\mathcal{C}_r}(Z)$ onto the set of rank-r matrices

$$\mathcal{C}_r \triangleq \{Z \in \mathbb{R}^{2n\times 2n} : \operatorname{rank}(Z) \le r\}. \tag{22}$$

The projection $\Pi_{\mathcal{C}_r}$ is easily computed via the singular value decomposition $Z = U\Sigma V^\top$. Define Σ_r by setting all but r largest (in magnitude) diagonal entries of Σ to 0. Then, the projection $\Pi_{\mathcal{C}_r}(Z)$ is given by

$$\Pi_{\mathcal{C}_r}(Z) = U\Sigma_r V^\top. \tag{23}$$

Now, the optimization problem in (21) can be efficiently solved by adapting the alternating direction method of multipliers (ADMM) algorithm [26]. The iterative algorithm is given by

$$Z_1[i+1] = \left(1+\frac{\rho}{2}\right)^{-1}\left(X_1 + \frac{\rho}{2}M_{11}[i]\right), \tag{24}$$

$$Z_2[i+1] = \left(1+\frac{\rho}{2}\right)^{-1}\left(X_2 + \frac{\rho}{2}M_{22}[i]\right), \tag{25}$$

$$\tilde{Z}[i+1] = \Pi_{\mathcal{C}_r}\left(\begin{bmatrix} Z_1[i+1] & I \\ I & X_2[i+1] \end{bmatrix} - W[i]\right), \tag{26}$$

$$W[i+1] = W[i] + \tilde{Z}[i+1] - \begin{bmatrix} Z_1[i+1] & I \\ I & Z_2[i+1] \end{bmatrix}, \quad i = 0,1,2,\ldots, \tag{27}$$

where $\rho > 0$ is the step size, and $M_{11}[i], M_{22}[i] \in \mathbb{R}^{n\times n}$ are defined as

$$\begin{bmatrix} M_{11}[i] & M_{12}[i] \\ M_{21}[i] & M_{22}[i] \end{bmatrix} \triangleq \tilde{Z}[i] + W[i]. \tag{28}$$

We show in Appendix B how to obtain this iteration algorithm for solving (21).

3.3. Projection onto the Set Λ Described by LMIs

The projection of (X_1, X_2) onto the set Λ in (15) can be described as convex optimization with LMIs [34,35].

$$\begin{aligned} \underset{Z_1,Z_2,W\in\mathcal{S}_n}{\text{minimize}} \quad & \operatorname{tr}(W) \\ \text{subject to} \quad & F(Z_1, Z_2) \preceq 0, \\ & \begin{bmatrix} W & (Z-X)^\top \\ Z-X & I \end{bmatrix} \succeq \epsilon I, \end{aligned} \tag{29}$$

where $Z = [Z_1\ Z_2]^\top$ and $X = [X_1\ X_2]^\top$.

Remark 2. *Grigoriadis and Skelton [13] have also proposed an alternating projection method. This method is described as*

$$\begin{aligned} Z[k] &= \Pi_{\mathcal{R}}(\Pi_{\mathcal{D}}(X[k])) \\ X[k+1] &= \Pi_{\Lambda}(Z[k]), \quad k = 0,1,2,\ldots, \end{aligned} \tag{30}$$

where

$$\mathcal{D} \triangleq \left\{ Z \in \mathcal{S}_{2n} : Z = \begin{bmatrix} X_1 & 0 \\ 0 & X_2 \end{bmatrix}, X_1, X_2 \in \mathcal{S}_n \right\},$$
$$\mathcal{R} \triangleq \{ Z \in \mathcal{S}_{2n} : \operatorname{rank}(Z + J) \leq r \}, \tag{31}$$
$$J \triangleq \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix}.$$

We note that Ω_r in (14) is the intersection of $\mathcal{D}$ and $\mathcal{R}$. The difference between the proposed algorithm (16) and the above algorithm (30) is the projection onto the set Ω_r of rank-constrained structured matrices. Namely, the algorithm in [13] approximates *the projection Π_{Ω_r} by a composite projection $\Pi_{\mathcal{R}}\Pi_{\mathcal{D}}$.*

Finally, we summarize the proposed algorithm to solve Problem 3 in Algorithm 1.

Algorithm 1 Algorithm to solve Problem 3.

Require: Initial guess $(X_1[0], X_2[0]) \in \mathcal{S}_n$
Ensure: $(X_1, X_2) \in \Omega_r \cap \Omega_\Lambda$
  for $k = 0, 1, 2, \ldots, N-1$ **do**
    $X_1 \leftarrow X_1[k]$ ▷ Projection onto Ω_r
    $X_2 \leftarrow X_2[k]$
    $\tilde{Z}[0] \leftarrow 0$
    $W[0] \leftarrow 0$
    for $i = 0, 1, 2, \ldots, M-1$ **do**
      $Z_1[i+1] \leftarrow \left(1 + \frac{\varrho}{2}\right)^{-1}\left(X_1 + \frac{\varrho}{2} M_{11}[i]\right)$
      $Z_2[i+1] \leftarrow \left(1 + \frac{\varrho}{2}\right)^{-1}\left(X_2 + \frac{\varrho}{2} M_{22}[i]\right)$
      $\tilde{Z}[i+1] \leftarrow \Pi_{\mathcal{C}_r}\left(\begin{bmatrix} Z_1[i+1] & I \\ I & X_2[i+1] \end{bmatrix} - W[i] \right)$
      $[i+1] \leftarrow W[i] + \tilde{Z}[i+1] - \begin{bmatrix} Z_1[i+1] & I \\ I & Z_2[i+1] \end{bmatrix}$
    end for
    $X \leftarrow [Z_1[M], Z_2[M]]^\top$ ▷ Projection onto Ω_Λ
    $(Z_1, Z_2, W) \leftarrow \operatorname*{arg\,min}_{Z_1, Z_2, W \in \mathcal{S}_n} \operatorname{tr}(W)$ subject to $F(Z_1, Z_2) \preceq 0$, $\begin{bmatrix} W & (Z-X)^\top \\ Z - X & I \end{bmatrix} \succeq \epsilon I$,
    $X_1[k+1] \leftarrow Z_1$
    $X_2[k+1] \leftarrow Z_2$
  end for
  $X_1 \leftarrow X_1[N]$
  $X_2 \leftarrow X_2[N]$

4. Numerical Examples

In this section, we show some control examples to illustrate the effectiveness of the proposed algorithm. We use benchmark models listed in the COMPL$_e$ib library [27]. MATLAB programs for the numerical computation in this section can be downloaded from [28]. For numerical optimization in the examples, we use SDPT3 [4] on MATLAB.

4.1. Stabilizing Static Controllers

We consider all benchmark linear time-invariant models whose order is less than 1000 from COMPL$_e$ib. There are 95 benchmark models to be checked. For these models, we solve Problem 1 by five methods:

1. Nonsmooth H^∞ synthesis [36] with `hinfstruct` funciton in MATLAB (NSH^∞)
2. Cone complementarity linearization algorithm [12] (CCL)
3. Nuclear norm minimization [15] (NNM)
4. Alternating projection with approximate projection onto Ω_r [13] (GS96)
5. Alternating projection with the proposed precise projection in Section 3 (Proposed)

Table 1 summarizes the number of successful results (i.e., a static stabilizing controller is obtained), and the average computational time.

Table 1. Stabilizing static controller design results.

	NSH^∞ [36]	**CCL [12]**	**NNM [15]**	**GS96 [13]**	**Proposed**
# success	75	61	53	56	59
CPU time [s]	0.365	56.5	4.58	663	666

From this table, we can say that the Nonsmooth H^∞ method [36] is the best among the five methods. We note that the alternating methods by [13] and our method show a very long average CPU time since there are a few models (e.g., `EB5` and `JE1`) for which it takes a very long CPU time by the se methods. For example, GS96 takes 53,450 [s] and the proposed method takes 52,315 [s] for `EB5`. However, there are models for which the proposed method successfully gives a stabilizing static controller while some of the other methods fail. We summarize the results in Table 2.

Table 2. Stabilizing static controller results (the full list available at [28]).

	NSH^∞ [36]	**CCL [12]**	**NNM [15]**	**GS96 [13]**	**Proposed**
`HE6`	unstable	**stable**	unstable	**stable**	**stable**
`HE7`	**stable**	**stable**	unstable	**stable**	**stable**
`REA3`	unstable	**stable**	unstable	unstable	**stable**
`DIS1`	**stable**	**stable**	unstable	unstable	**stable**
`PAS`	unstable	**stable**	**stable**	**stable**	**stable**
`TF1`	unstable	unstable	unstable	unstable	**stable**
`TF2`	unstable	**stable**	**stable**	**stable**	**stable**
`NN1`	**stable**	unstable	unstable	unstable	**stable**
`NN11`	**stable**	**stable**	unstable	**stable**	**stable**
`NN12`	unstable	unstable	unstable	**stable**	**stable**
`FS`	unstable	**stable**	**stable**	**stable**	**stable**

An advantage of the proposed method is found in this table. Since stabilizing static controller design is in general a non-convex problem, and hence there may be no unified approach that gives a solution for any plant models. For example, no methods but the proposed method can compute a stabilizing static controller for `TF1`. We emphasize our method is an effective method that may provide a solution to a reduced-order controller design problem that cannot be solved by standard methods as nonsmooth H^∞, cone complementarity linearization, and nuclear norm minimization.

4.2. Stabilizing Low-Order Controllers

In this section, we focus on the model TF1. In this model, the state-space matrices are given as follows:

$$A = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ -0.088 & 0.0345 & 0 & 0 & 1 & -0.0032 & 0 \\ 0 & 0 & 0.05 & 0 & 0 & 0 & -0.00001 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0.09 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix},$$
$$C = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \quad D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}. \tag{32}$$

As explained in the previous subsection, a static controller is obtained only with the proposed method. Here, we consider low-order controllers with order $n_c = 1$ and $n_c = 2$. It is observed that all methods but the nonsmooth H^∞ method return stabilizing controllers with order 1 and 2. The results are summarised in Table 3.

Table 3. Results of stabilizing low-order controllers with $n_c = 1$ and 2 for TF1.

	NSH^∞ [36]	**CCL [12]**	**NNM [15]**	**GS96 [13]**	**Proposed**
TF1 ($n_c = 1$)	unstable	**stable**	**stable**	**stable**	**stable**
CPU time [s]	0.016910	0.214947	0.177307	0.194252	1.836873
TF1 ($n_c = 2$)	unstable	**stable**	**stable**	**stable**	**stable**
CPU time [s]	0.016012	0.263448	0.205307	0.206729	4.250059

The state-space matrices A_K, B_K, C_K, D_K of the second order controller ($n_c = 2$) with the proposed method is given by

$$A_K = \begin{bmatrix} -1.804 \times 10^4 & 1.356 \times 10^4 \\ 9512 & -7151 \end{bmatrix}, \quad B_K = \begin{bmatrix} -9193 & -2019 & -240.4 & -1.77 \times 10^4 \\ 4848 & 1066 & 126.8 & 9333 \end{bmatrix},$$
$$C_K = \begin{bmatrix} -4412 & 3317 \\ -1.075 \times 10^5 & 8.08 \times 10^4 \end{bmatrix}, \tag{33}$$
$$D_K = \begin{bmatrix} -2249 & -496 & -58.91 & -4329 \\ -5.478 \times 10^4 & -1.203 \times 10^4 & -1433 & -1.055 \times 10^5 \end{bmatrix}.$$

4.3. H^∞ Static Controllers

Then, we consider the reduced-order H^∞ control problem formulated in Problem 2. Here we choose AC4, NN1, NN12, and HE6 from COMPL$_e$ib. AC4 is from a autopilot control problem for an air-to-air missile discussed in [37]. NN1 and NN12 are academic test problems proposed in [38] and [39], respectively. HE6 is a helicopter model that has four inputs, 20 states, and six outputs [40]. For these plant models, we seek the H^∞ static controller by using the bisection method on γ. Namely, we first give a sufficiently large upper bound $\overline{\gamma}$, for example $\overline{\gamma} = 100$, and a sufficiently small lower bound $\underline{\gamma}$ (e.g., $\underline{\gamma} = 0$). Then we set $\gamma = (\underline{\gamma} + \overline{\gamma})/2 = 50$ and solve Problem 2. If there is a feasible solution, then we update the upper bound to $\overline{\gamma} = \gamma = 50$, otherwise we set the lower bound to $\underline{\gamma} = \gamma = 50$. We note that the problem is assumed to be infeasible if a solution of Problem 2 is not found after 15 iterations of (16). We repeat this process until sufficient accuracy is achieved.

Table 4 shows the obtained upper bounds of γ for the chosen models.

Table 4. Upper bounds γ of the H^∞ norm by static controller.

Model	AC4	NN1	NN12	HE6
γ	1.000	74.72	28.09	520.0

The obtained static controllers are given as follows:

$$\begin{aligned}
K_{\mathrm{AC4}} &= \begin{bmatrix} -0.3228 & -0.07534 \end{bmatrix}, \\
K_{\mathrm{NN1}} &= \begin{bmatrix} 3.5 & 60.22 \end{bmatrix}, \\
K_{\mathrm{NN12}} &= \begin{bmatrix} 22.81 & -35.61 \\ -5.179 & 8.137 \end{bmatrix}, \\
K_{\mathrm{HE6}} &= \begin{bmatrix} 83.25 & -0.5581 & -0.5931 & 0.1238 & 0.1546 & -0.02533 \\ -24.27 & 7.876 & 0.7263 & 0.0309 & 0.3272 & -0.6042 \\ -15.85 & 0.1609 & -6.578 & -2.062 & 1.677 & 0.1478 \\ 65.77 & -1.413 & 9.07 & -16.61 & -0.9596 & -0.04191 \end{bmatrix}.
\end{aligned} \tag{34}$$

It is easy to check that the obtained static controllers really achieve the H^∞ norm listed in Table 2. These numerical examples demonstrate the effectiveness of the proposed method.

5. Conclusions

In this paper, we have proposed a novel design method of reduced-order controllers based on projection onto the set of rank-constrained structured matrices. We compared the proposed method with existing methods by numerical examples. We have shown that the proposed algorithm successfully solved benchmark problems that other methods could not. This is thanks to the precise computation of the projection onto the set of rank-constrained structured matrices. Future work includes reduced-order controller design with sparsity constraints on the controller realization, which is a challenging problem that should take two non-convex rank and sparsity constraints.

Author Contributions: Data curation, Y.I.; Investigation, M.N., Y.I. and N.S.; Writing—original draft, M.N.; Writing—review & editing, N.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by JSPS KAKENHI Grant Numbers JP20H02172, JP20K21008, and JP19H02301.

Data Availability Statement: The MATLAB programs and experimental results can be downloaded from https://github.com/nagahara-masaaki/MR accessed on 20 July 2022.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Proof of Lemma 1

From the definitions of (3) and (4), and the iterative algorithm (16), we have

$$\begin{aligned}
\mathrm{dist}(X[k], \Omega_r) &= \|Z[k] - X[k]\| \le \|Z[k-1] - X[k]\|, \\
\mathrm{dist}(Z[k], \Lambda) &= \|X[k+1] - Z[k]\| \le \|X[k] - Z[k]\|, \\
\mathrm{dist}(X[k+1], \Omega_r) &= \|Z[k+1] - X[k+1]\| \le \|Z[k] - X[k+1]\|, \\
\mathrm{dist}(Z[k+1], \Lambda) &= \|X[k+2] - Z[k+1]\| \le \|X[k+1] - Z[k+1]\|,
\end{aligned} \tag{A1}$$

for $k = 0, 1, 2, \ldots$. These inequalities easily imply (17).

Appendix B. ADMM Algorithm

The minimization problem in (19) is described as a standard form for ADMM [26], and the iteration algorithm is directly obtained by

$$\begin{aligned}\begin{bmatrix} Z_1[i+1] \\ Z_2[i+1] \end{bmatrix} &= \underset{Z_1,Z_2}{\arg\min}\, f_1(Z_1, Z_2; \tilde{Z}[i], W[i]), \\ \tilde{Z}[i+1] &= \underset{\tilde{Z}}{\arg\min}\, f_2(\tilde{Z}; Z_1[i+1], Z_2[i+1], W[i]), \\ W[i+1] &= W[i] + \tilde{Z}[i+1] - \begin{bmatrix} X[i+1] & I \\ I & Y[i+1] \end{bmatrix}, \\ i &= 0, 1, 2, \dots, \end{aligned} \tag{A2}$$

where

$$f_1(Z_1, Z_2; \tilde{Z}, W) \triangleq \|Z_1 - X_1\|^2 + \|Z_2 - X_2\|^2 + \frac{\rho}{2}\left\| \tilde{Z} - \begin{bmatrix} Z_1 & I \\ I & Z_2 \end{bmatrix} + W \right\|^2, \tag{A3}$$

$$f_2(\tilde{Z}; Z_1, Z_2, W) \triangleq \mathcal{I}_r(\tilde{Z}) + \frac{\rho}{2}\left\| \tilde{Z} - \begin{bmatrix} Z_1 & I \\ I & Z_2 \end{bmatrix} + W \right\|^2. \tag{A4}$$

First, for function f_1, we have

$$\begin{aligned} f_1(Z_1, Z_2; \tilde{Z}[i], W[i]) = c\left\| Z_1 - c^{-1}\left(X_1 + \frac{\rho}{2} M_{11}[i] \right) \right\|^2 \\ + c\left\| Z_2 - c^{-1}\left(X_2 + \frac{\rho}{2} M_{22}[i] \right) \right\|^2 + \text{const.} \end{aligned} \tag{A5}$$

where $c \triangleq 1 + \rho/2$, and $M_{11}[i]$, $M_{22}[i]$ are defined in (28). From (A5), we have the first two steps (24) and (25).

Then, by the definition of projection, a minimizer of $f_2(\tilde{Z}; Z_1[i+1], Z_2[i+1], W[i])$ in (A4) is obtained by the right-hand side of (26).

References

1. Iwasaki, T.; Skelton, R.E. Parametrization of all stabilizing controllers via quadratic Lyapunov functions. *J. Optim. Theory Appl.* **1995**, *85*, 291–307.
2. Scherer, C.; Gahinet, P.; Chilali, M. Multiobjective output-feedback control via LMI optimization. *IEEE Trans. Autom. Control* **1997**, *42*, 896–911.
3. Sturm, J.F. Using SeDuMi 1.02, A Matlab toolbox for optimization over symmetric cones. *Optim. Methods Softw.* **1999**, *11*, 625–653.
4. Toh, K.C.; Todd, M.J.; Tütüncü, R.H. SDPT3 — A Matlab software package for semidefinite programming, Version 1.3. *Optim. Methods Softw.* **1999**, *11*, 545–581.
5. Available online: https://www.mosek.com/ (accessed on 20 July 2022).
6. Lofberg, J. YALMIP: A toolbox for modeling and optimization in MATLAB. In Proceedings of the 2004 IEEE International Conference on Robotics and Automation, New Orleans, LA, USA, 26 April–1 May 2004; pp. 284–289.
7. Grant, M.; Boyd, S. Graph implementations for nonsmooth convex programs. In *Recent Advances in Learning and Control*; Blondel, V., Boyd, S., Kimura, H., Eds.; Lecture Notes in Control and Information Sciences; Splinger: Berlin/Heidelberg, Germany, 2008; Volume 371, pp. 95–110.
8. Diamond, S.; Boyd, S. CVXPY: A Python-embedded modeling language for convex optimization. *J. Mach. Learn. Res.* **2016**, *17*, 1–5.
9. Toker, O.; Özbay, H. On the NP-hardness of solving bilinear matrix inequalities and simultaneous stabilization with static output feedback. In Proceedings of the 1995 American Control Conference (ACC'95), Seattle, WA, USA, 21–23 June 1995; Volume 4, pp. 2525–2526.
10. Ghaoui, L.E.; Gahinet, P. Rank Minimization under LMI constraints: A Framework for Output Feedback Problems. In Proceedings of the European Control Conference 1993, Groningen, The Netherlands, 28 June–1 July 1993; Volume 3.
11. Iwasaki, T.; Skelton, R.E. The XY-centring algorithm for the dual LMI problem: A new approach to fixed-order control design. *Int. J. Control* **1995**, *62*, 1257–1272.

12. El Ghaoui, L.; Oustry, F.; AitRami, M. A cone complementarity linearization algorithm for static output-feedback and related problems. *IEEE Trans. Autom. Control* **1997**, *42*, 1171–1176. https://doi.org/10.1109/9.618250.
13. Grigoriadis, K.M.; Skelton, R.E. Low-order control design for LMI problems using alternating projection methods. *Automatica* **1996**, *32*, 1117–1125.
14. Yang, K.; Orsi, R.; Moore, J.B. A projective algorithm for static output feedback stabilization. In Proceedings of the 2nd IFAC Symposium on System, Structure and Control, Oaxaca, Mexico, 8–10 December 2004; pp. 263–268.
15. Fazel, M.; Hindi, H.; Boyd, S. Rank minimization and applications in system theory. In Proceedings of the 2004 American Control Conference, Boston, MA, USA, 30 June–2 July 2004; Volume 4, pp. 3273–3278.
16. Amirifar, R.; Sadati, N. Low-order H_∞ controller design for an active suspension system via LMIs. *IEEE Trans. Ind. Electron.* **2006**, *53*.
17. Recht, B.; Xu, W.; Hassibi, B. Necessary and sufficient conditions for success of the nuclear norm heuristic for rank minimization. In Proceedings of the 2008 47th IEEE Conference on Decision and Control, Cancun, Mexico, 9–11 December 2008; pp. 3065–3070.
18. Recht, B.; Fazel, M.; Parrilo, P.A. Guaranteed minimum-rank solutions of linear matrix equations via nulcear norm minimization. *SIAM Rev.* **2010**, *52*, 451–501.
19. Doelman, R.; Verhaegen, M. Sequential convex relaxation for robust static output feedback structured control. *IFAC-PapersOnLine* **2017**, *50*, 15518–15523.
20. Vidyasagar, M. *An Introduction to Compressed Sensing*; SIAM: Philadelphia, PA, USA, 2019.
21. Mi, J.X.; Zhang, Y.N.; Lai, Z.; Li, W.; Zhou, L.; Zhong, F. Principal Component Analysis based on Nuclear norm Minimization. *Neural Netw.* **2019**, *118*, 1–16.
22. Liu, S.; Hu, Q.; Li, P.; Zhao, J.; Liu, M.; Zhu, Z. Speckle Suppression Based on Weighted Nuclear Norm Minimization and Grey Theory. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 2700–2708. https://doi.org/10.1109/TGRS.2018.2876339.
23. Tian, R.; Sun, G.; Liu, X.; Zheng, B. Sobel Edge Detection Based on Weighted Nuclear Norm Minimization Image Denoising. *Electronics* **2021**, *10*, 655.
24. Gunes, B.; van Wingerden, J.W.; Verhaegen, M. Tensor Nuclear Norm LPV Subspace Identification. *IEEE Trans. Autom. Control* **2018**, *63*, 3897–3903. https://doi.org/10.1109/TAC.2018.2800772.
25. Varanasi, S.K.; Jampana, P. Nuclear norm subspace identification of continuous time state–space models with missing outputs. *Control Eng. Pract.* **2020**, *95*, 104239.
26. Boyd, S.; Parikh, N.; Chu, E.; Peleato, B.; Eckstein, J. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.* **2011**, *3*, 1–122.
27. Leibfritz, F. *COMPL$_e$ib: Constraint Matrix Optimization Problem Library*; Tech. Report; University of Trier: Trier, Germany, 2005.
28. Available online: https://github.com/nagahara-masaaki/MR/ (accessed on 20 July 2022).
29. Doyle, J.C.; Glover, K.; Khargonekar, P.P.; Francis, B.A. State-space solutions to standard H_2 and H_∞ control problems. *IEEE Trans. Autom. Control* **1989**, *34*, 831–847.
30. Iwasaki, T. *LMI and Control*; Shokodo: Tokyo, Japan, 1997.
31. Escalante, R.; Raydan, M. *Alternating Projection Methods*; SIAM: Philadelphia, PA, USA, 2011.
32. Boyle, J.P.; Dykstra, R.L. A Method for Finding Projections onto the Intersection of Convex Sets in Hilbert Spaces. In *Advances in Order Restricted Statistical Inference, Lecture Notes in Statistics*; Dykstra, R., Robertson, T., Wright, F.T., Eds.; Springer: New York, NY, USA, 1986; Volume 37.
33. Bai, J.; Li, J.; Dai, P. Novel alternating update method for low rank approximation of structured matrices. *Appl. Numer. Math.* **2017**, *121*, 223–233.
34. Boyd, S.; Ghaoui, L.E.; e. Feron.; Balakrishnan, V. *Linear Matrix Inequalities in System and Control Theory*; SIAM: Philadelphia, PA, USA, 1994.
35. Nagahara, M.; Ogura, M.; Yamamoto, Y. Iterative Greedy LMI for Sparse Control. *IEEE Contr. Syst. Lett.* **2022**, *6*, 986–991.
36. Apkarian, P.; Noll, D. Nonsmooth H_∞ Synthesis. *IEEE Trans. Autom. Control* **2006**, *51*, 71–86.
37. Fares, B.; Apkarian, P.; Noll, D. An augmented Lagrangian method for a lass of LMI-constrained problems in robust control theory. *Int. J. Control* **2001**, *74*, 348–360.
38. Miller, L.F.; Cohran, R.G.; Howze, J.W. Output feedback stabilization of a spectral radius functional. *Int. J. Control* **1978**, *27*, 455–462.
39. Rosenthal, J.; Wang, X.A. Output feedback pole placement with dynamic compensators. *IEEE Trans. Autom. Control* **1996**, *41*, 830–843.
40. Skogestad, S.; Postlethwaite, I. *Multivariable Feedback Control*; John Wiley & Sons: Chichester, UK; New York, NY, USA; Brisbane, Australia; Toronto, ON, Canada; Singapore, 1996.