Three Basic Theorems in Numerical Analysis in Control Engineering Course and Their Application

Hiroaki Mukaidani* and Tadashi Shima

International Journal of Control, Automation and Systems 16(5) (2018) 2321-2333

ISSN:1598-6446 (print version)
eISSN:2005-4092 (electronic version)

To link to this article:
http://dx.doi.org/10.1007/s12555-017-0624-y

Springer

International Journal of Control, Automation and Systems 16(5) (2018) 2321-2333
http://dx.doi.org/10.1007/s12555-017-0624-y

ISSN:1598-6446 eISSN:2005-4092
http://www.springer.com/12555

Three Basic Theorems in Numerical Analysis in Control Engineering Course and Their Application

Hiroaki Mukaidani* and Tadashi Shima

Abstract: Control system design packages like MATLAB, SICLAB, OCTAVE, etc. have become essential components of both undergraduate and graduate courses in the field of systems and controls. In particular, the most important subject related to control system design in the undergraduate course is the analysis of a nonlinear equation that is based on iterative methods. In this paper, applications of three basic theorems, –implicit function theorem, Newton-Kantorovich theorem, and fixed point theorem– are proposed to be taught in the numerical analysis in the control engineering course. In order to demonstrate the usefulness of these theorems, several important features are discussed. Furthermore, a practice exercise based on the practical control problem is discussed for proving the useful subject of the numerical analysis in the control engineering course in the graduate level.

Keywords: Control engineering course, fixed point theorem, implicit function theorem, Newton-Kantorovich theorem.

1. INTRODUCTION

The core curriculum of the electrical and computer engineering course generally includes numerical analysis. After referring to many relevant books [1–3], the need for including Newton's method and related iterative methods in the section on numerical solutions in an introductory course appears to be appropriate. In general, simple and scalar nonlinear equations are treated in this course. The abovementioned need arises due to at least two reasons. First, numerical solutions can be easily depicted by using computer graphics facilities. Second, a problem under consideration can be solved analytically. Although Lyapunov, Sylvester, and Riccati algebraic equations and their associated software packages are well known in the field of systems and controls [4], the methods used in solving these equations are generally not described in detail in related courses. Furthermore, to the best of the author's knowledge, although the numerical computation of three basic theorems – implicit function theorem, Newton-Kantorovich theorem, and fixed point theorem – are well-known, their applications have not been discussed in the engineering course thus far. In fact, these important theorems play a vital role in establishing the controller. For instance, it has been shown that the justification of the existence of solutions to the equations and sensitivity analysis were conducted based on the implicit function theorem [5]. The Newton method for solving the non-symmetric Riccati equations has been considered in [6]. A numerical algorithm, which is based on fixed-point iterations, to obtain the solution of algebraic Lyapunov type equations has been revisited [7]. However, these contributions may not be presented in such course.

An educational framework of control system design is very important for both the theoretical and practical aspects [8]. In particular, it is well-known that recursion is an important problem; there is no doubt that it should be covered in computer science courses. Most textbooks use the Hanoi problem [9] or Fibonacci numbers [10] as the first examples of recursion. These examples often appear as recommended programs in some textbooks. On the other hand, Newton's method was frequently used as an example for solving a nonlinear equations in the electrical and computer engineering course. Although the advantages of using Newton's method are well discussed in textbooks, it should be noted that another important feature –the uniqueness of the convergence solution– has never been entirely discussed in literature. Moreover, it is observed that another famous method of the fixed point iteration for solving a linear equation was not sufficiently documented.

This paper demonstrates that the applications of the numerical computation of the three basic theorems are useful to be taught in the numerical analysis in the control engineering course. The core curriculum could be introduced through mathematical theory and simulation. After establishing the structure of the ARE corresponding to the H_∞ control problem by using the implicit function theorem,

Manuscript received October 10, 2017; revised January 22, 2018; accepted May 3, 2018. Recommended by Associate Editor Do Wan Kim under the direction of Editor Yoshito Ohta.

Hiroaki Mukaidani and Tadashi Shima are with the Graduate School of Engineering, Hiroshima University, Higashi-Hiroshima, 1-4-1, Kagamiyama 739-8527, Japan (e-mails: {mukaida, tadashi}@hiroshima-u.ac.jp).
* Corresponding author.

©ICROS, KIEE and Springer 2018

an algorithm for solving the ARE corresponding this problem related to weakly coupled large-scale systems is established for this course. In order to guarantee the quadratic convergence and uniqueness of the solution, the Newton-Kantorovich theorem is applied. Moreover, the fixed point iterations are combined to attain reduced-order calculations. As a result, it is shown that linear convergence is attained by using the fixed point theorem. Finally, to demonstrate the efficiency of the proposed algorithm, a numerical example is presented. Furthermore, a practice exercise based on the optimization problem related to the practical control problem is provided.

Notation: The notations used in this paper are fairly standard. **block diag** denotes the block diagonal matrix. Superscript T denotes matrix transpose. **Tr** denotes the trace of a matrix. $\lambda_{\max}$ denotes maximum eigenvalue. $\|\cdot\|$ denotes norm of a matrix. det denotes the determinant of a square matrix. I_p denotes the $p \times p$ identity matrix.

2. PRELIMINARY

2.1. Matrix tools for control systems design

This subsection develops some matrix tools that will prove useful to us later. The **vec** operator creates a column vector from a matrix $A = (a_{ij})$ by stacking the column vectors of A one below the other [11].

$$\mathbf{vec}A := \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix}, \; a_k = \begin{bmatrix} a_{1k} \\ a_{2k} \\ \vdots \\ a_{nk} \end{bmatrix}. \tag{1}$$

A permutation matrix is a binary matrix that has only one entry of 1 in each row and each column and 0's elsewhere. Permutation matrices are the matrix representations of permutations. The permutation matrix $\boldsymbol{U}_{nn}$ with n elements is defined as in equation (2).

$$\begin{aligned} \boldsymbol{U}_{nn} := & \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}^T \\ & + \cdots \\ & + \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} \otimes \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}^T. \end{aligned} \tag{2}$$

The permutation matrix has the following useful properties.

$$\begin{aligned} & \boldsymbol{U}_{nn} \in \mathbb{R}^{n^2 \times n^2}, \; \boldsymbol{U}_{nn} = \boldsymbol{U}_{nn}^T, \\ & \boldsymbol{U}_{nn}\boldsymbol{U}_{nn}^T = \boldsymbol{U}_{nn}\boldsymbol{U}_{nn} = I_{n^2}, \\ & (A^T \otimes I_n)\boldsymbol{U}_{nn} = \boldsymbol{U}_{nn}(I_n \otimes A^T), \\ & U_{lm}\mathbf{vec}B = \mathbf{vec}B^T, \; B \in \mathbb{R}^{l \times m}. \end{aligned}$$

$A = (a_{ij})$ is an $m \times n$ matrix and B is a $p \times q$ matrix. The $mp \times nq$ matrix defined by equation (3)

$$A \otimes B = \begin{bmatrix} a_{11}B & a_{12}B & \cdots & a_{1n}B \\ a_{21}B & a_{22}B & \cdots & a_{2n}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1}B & a_{n2}B & \cdots & a_{nn}B \end{bmatrix} \tag{3}$$

is called the Kronecker product.

The Kronecker product can be used to obtain a convenient representation for some matrix equations. Consider for instance the equation $AXB = C$, where A, B, and C are known matrices and matrix X is unknown. It can be rewritten this equation as

$$\mathbf{vec}(AXB) = (B^T \otimes A)\mathbf{vec}X = \mathbf{vec}C.$$

It now follows from the properties of the Kronecker product that the equation $AXB = C$ has a unique solution if and only if A and B are nonsingular.

On the other hand, the following derivative holds.

$$\frac{\partial}{\partial X}\mathbf{Tr}[BXC] = B^T C^T, \tag{4}$$

where $B, C, X \in \mathbb{R}^{n \times n}$.

Let $\mathcal{F}(X)$ be a differentiable $m \times p$ real matrix function of an $n \times q$ matrix of real variables X. The Jacobian matrix of $\mathcal{F}$ at X is the $mp \times nq$ matrix

$$\boldsymbol{D}\mathcal{F}(X) := \frac{\partial \mathbf{vec}\mathcal{F}(X)}{\partial(\mathbf{vec}X)^T}. \tag{5}$$

Many examples are given below and they will be used for deriving the proof of this statement.

$$\frac{\partial \mathbf{vec}AXB}{\partial(\mathbf{vec}X)^T} = B^T \otimes A, \quad \frac{\partial \mathbf{vec}X^T A}{\partial(\mathbf{vec}\,X)^T} = (A^T \otimes I_n)\boldsymbol{U}_{nn}.$$

2.2. Newton's method for solving nonliner equation

In order to demonstrate the efficiency of the matrix tools, a simple example is provided. Let us consider the following nonlinear matrix equation.

$$\mathcal{F}(X) := X^2 - A, \tag{6}$$

where $X, A \in \mathbb{R}^{n \times n}$ and A are known matrices whose elements are scalar.

It can be obtained the following result by using (5).

$$\frac{\partial \mathbf{vec}\mathcal{F}(X)}{\partial(\mathbf{vec}X)^T} = X^T \otimes I_n + I_n \otimes X. \tag{7}$$

Thus, considering Newton's method, the following equation holds.

$$\mathbf{vec}X^{(k+1)} = \mathbf{vec}X^{(k)}$$

$$
\begin{aligned}
&- \left[\frac{\partial \mathbf{vec}\mathcal{F}(X)}{\partial (\mathbf{vec}X)^T} \bigg|_{X=X^{(k)}} \right]^{-1} \mathbf{vec}\mathcal{F}(X^{(k)}) \\
&= \mathbf{vec}X^{(k)} \\
&\quad - [X^{(k)T} \otimes I_n + I_n \otimes X^{(k)}]^{-1} \mathbf{vec}[X^{(k)2} - A].
\end{aligned}
$$

Finally, eliminating $\mathbf{vec}X^{(k)}$, which is the first term on the right hand side of the above equation, the following Newton iteration (8) is obtained.

$$
\mathbf{vec}X^{(k+1)} = [X^{(k)T} \otimes I_n + I_n \otimes X^{(k)}]^{-1} \mathbf{vec}[X^{(k)2} + A]. \tag{8}
$$

2.3. Basic theorems in numerical computation

In order to establish useful and reliable results in the numerical analysis in the control engineering course, the following basic theorems are provided.

Implicit Function Theorem [12]**:** Let $F : \mathbb{R}^{m+n} \to \mathbb{R}^n$ be a continuously differentiable function defined in an open subset of $\mathbb{R}^{m+n}$ that contains the point $(a,\ b)$, where $a \in \mathbb{R}^m$ and $b \in \mathbb{R}^n$. Further, assume that the initial equation is $F(a,\ b) = 0$. Let us take the Jacobian of F evaluated at $(a,\ b)$, $DF(a,\ b) = 0$, and split it into $[X \mid Y]$ where submatrices X and Y are given by

$$
X_{n\times m} = \left(\frac{\partial f_i}{\partial x_j} \bigg|_{(a,\ b)} \right)_{1\le i\le n,\ 1\le j\le m},
$$

$$
Y_{n\times n} = \left(\frac{\partial f_i}{\partial y_j} \bigg|_{(a,\ b)} \right)_{1\le i\le n,\ m+1\le j\le m+n},
$$

so that $DF(a,\ b) = [X|Y]$. If Y is invertible (i.e., the determinant of Y is nonzero), then $F(\mathbf{x},\ \mathbf{y}) = 0$ defines $\mathbf{y}$ as a function of $\mathbf{x}$ near $(a,\ b)$; more precisely, there are open sets A and B such that $a \in A \subseteq \mathbb{R}^m$ and $b \in B \subseteq \mathbb{R}^n$. The sets have the following property: for each $\mathbf{x} \in A$, there is a unique $\mathbf{y} \in B$, such that $F(\mathbf{x},\ \mathbf{y}) = 0$. Since $\mathbf{y}$ is uniquely determined by $\mathbf{x}$ with respect to A and B, it can be called as $G(\mathbf{x})$; so what there exists a function $G : A \to B$ such that for each $\mathbf{x} \in A$, $F(\mathbf{x},\ G(\mathbf{x})) = 0$, and $G(\mathbf{x})$ is the only element in B with that property. Because $F(a,\ b) = 0$, in particular, $G(a) = b$ holds; therefore, this function "locally generalizes" the initial equation $F(a,\ b) = 0$. Furthermore, the theorem guarantees that function G is differentiable; in particular, it turns out that $DG(a) = -Y^{-1}X$.

The implicit function theorem plays an important role in determining the existence of a solution and in the analysis of the asymptotic structure for the weakly coupled large-scale linear quadratic Gaussian (LQG) control problem (see, e.g., [12–14], and the references therein).

Newton-Kantorovich Theorem [15, 16]**:** Assume that $F : \mathbb{R}^n \to \mathbb{R}^n$ is differentiable on a convex set D and that $\|F'(\mathbf{x}) - F'(\mathbf{y})\| \le \gamma\|\mathbf{x} - \mathbf{y}\|$ for all $\mathbf{x},\ \mathbf{y} \in D$. Suppose that there is an $\mathbf{x}^0 \in D$ such that $\|F'(\mathbf{x}^0)^{-1}\| \le \beta$, $\|F'(\mathbf{x}^0)^{-1}F(\mathbf{x}^0)\| \le \eta$ and $\theta := \beta\gamma\eta < 1/2$. Assume that $S := \{\ \mathbf{x} :\ \ \|\mathbf{x} - \mathbf{x}^0\| \le t^*\ \} \subset D$, $t^* = (1 - \sqrt{1 - 2\theta})/(\beta\gamma)$. Then Newton iterations $\mathbf{x}^{k+1} = \mathbf{x}^k - F'(\mathbf{x}^k)^{-1}F(\mathbf{x}^k)$, $k = 0,\ 1,\ \cdots$, are well defined and converge to a solution $\mathbf{x}^*$ of $F(\mathbf{x}) = 0$ in S. Moreover, the solution $\mathbf{x}^*$ is unique in $\tilde{S} \cap D$, where $\tilde{S} := \{\ \mathbf{x} :\ \ \|\mathbf{x} - \mathbf{x}^0\| \le \tilde{t}\ \} \subset D$, $\tilde{t} = (1 + \sqrt{1 - 2\theta})/(\beta\gamma)$ and error estimate is given by

$$
\|\mathbf{x}^* - \mathbf{x}^k\| \le \frac{(2\theta)^{2^k}}{2^k\beta\gamma} = 2^{1-k}(2\theta)^{2^k-1}\eta,\ \ k = 0,\ 1,\ \ldots.
$$

It is noteworthy that the Newton-Kantorovich theorem gives useful results; for example, it provides not only the error estimation and quadratic convergence but also the local uniqueness of the solution for cross-coupled large-scale nonlinear equations [17].

Fixed Point Theorem [18]**:** Let $(X,\ d)$ be a complete metric space. A function $T : X \to X$ is said to be a contraction mapping if there is a constant q with $0 \le q \le 1$ such that $d(T\mathbf{x},\ T\mathbf{y}) \le q \cdot d(\mathbf{x},\ \mathbf{y})$ for all $\mathbf{x},\ \mathbf{y} \in X$. Then, map T admits one and only one fixed point $\mathbf{x}^*$ in X (this means $T\mathbf{x}^* = \mathbf{x}^*$). Furthermore, this fixed point can be found as follows. Start with an arbitrary element $\mathbf{x}^0$ in X and define an iterative sequence by $\mathbf{x}^n = T\mathbf{x}^{n-1}$ for $n = 1,\ 2,\ 3,\ \ldots$. This sequence converges, and its limit is $\mathbf{x}^*$.

It should be noted that the fixed point theorem results in reliable results and therefore a reduction in the dimension of the workspace and linear convergence are both attained.

3. STATE FEEDBACK H_∞ CONTROL PROBLEM FOR WEAKLY COUPLED LARGE-SCALE SYSTEMS

Consider linear time-invariant weakly coupled systems [12–14, 17].

$$
\dot{x}(t) = A_\varepsilon x(t) + B_{1\varepsilon} w(t) + B_{2\varepsilon} u(t), \tag{9a}
$$

$$
z(t) = C_1 x(t) + D_1 u(t), \tag{9b}
$$

where

$$
x(t) := \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix},\ \ u(t) := \begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix},
$$

$$
A_\varepsilon := \begin{bmatrix} A_{11} & \varepsilon A_{12} \\ \varepsilon A_{21} & A_{22} \end{bmatrix},\ \ B_{1\varepsilon} := \begin{bmatrix} 0 & B_{111} & \varepsilon B_{112} \\ 0 & \varepsilon B_{121} & B_{122} \end{bmatrix},
$$

$$
B_{2\varepsilon} := \begin{bmatrix} B_{211} & \varepsilon B_{212} \\ \varepsilon B_{221} & B_{222} \end{bmatrix},
$$

$$
C_1 := \begin{bmatrix} \mathbf{block\ diag}\begin{pmatrix} C_{11} & C_{12} \end{pmatrix} \\ 0 \end{bmatrix},\ \ D_1 := \begin{bmatrix} 0 \\ I_{m_1+m_2} \end{bmatrix}.
$$

$x_i(t) \in \mathbb{R}^{n_i}$ are the state vectors, $w(t) \in \mathbb{R}^l$ is the disturbance, $u_i(t) \in \mathbb{R}^{m_i}$ are the control inputs, $z(t) \in \mathbb{R}^p$ is the controlled output. ε denotes a small positive weak coupling parameter that connects the linear system with other subsystems.

Without loss of generality, let us now consider the H_∞ state feedback control problems under the following basic assumption.

Assumption 1: 1) The pair $(A_\varepsilon,\ B_{1\varepsilon})$ is stabilizable and $(C_1,\ A_\varepsilon)$ is detectable for a given $\varepsilon \in (0,\ \varepsilon^*]$, $\varepsilon^* > 0$.

2) The pair $(A_\varepsilon,\ B_{2\varepsilon})$ is stabilizable for a given $\varepsilon \in (0,\ \varepsilon^*]$, $\varepsilon^* > 0$.

It should be noted that the purpose of this paper is not the study of a numerical computation approach for weakly coupled large-scale systems, but the learning of the principles of various numerical algorithms. Thus, the complication and generalization of the problems should be avoided.

The H_∞ optimal control problem for weakly coupled large-scale systems is given below.

[H_∞ Optimal Control Problem] Given a stabilizable plant (9), find all admissible gain K_ε such that $\|G_\varepsilon(s)\|_\infty < \gamma$, where $G_\varepsilon(s)$ equals the transfer function from w to z.

The following result is well known [19].

Lemma 1: Under Assumption 1, there exists an admissible controller such that $\|G_\varepsilon(s)\|_\infty < \gamma$ if and only if the following condition holds.

The ARE

$$A_\varepsilon^T P_\varepsilon + P_\varepsilon A_\varepsilon - P_\varepsilon(B_{2\varepsilon}B_{2\varepsilon}^T - \gamma^{-2}B_{1\varepsilon}B_{1\varepsilon}^T)P_\varepsilon + C_1^T C_1 = 0 \tag{10}$$

has a unique positive semidefinite stabilizing solution.

Moreover, when this condition holds, the controller is given by (11).

$$u(t) = -\gamma^{-2}B_{2\varepsilon}^T P_\varepsilon x(t). \tag{11}$$

In this section, the asymptotic structure of the solution for the ARE (10) is established. First, for various values of ε, such a structure is predicted by solving the ARE via the OCTAVE code. Now, the matrices of the equation are chosen as follows:

$$A_\varepsilon = \begin{bmatrix} 0 & \varepsilon \\ -2\varepsilon & -2 \end{bmatrix},\ B_{1\varepsilon} = \begin{bmatrix} 1 & \varepsilon \\ \varepsilon & 2 \end{bmatrix},$$
$$B_{2\varepsilon} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},\ C_1^T C_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},\ \gamma = 2.$$

For $\varepsilon = 0.001$, $\varepsilon = 0.0001$ and $\varepsilon = 0.00001$, each solution $P_\varepsilon = P(\varepsilon)$ is given below.

$$P(0.001) = \begin{bmatrix} 1.154700332801003 & 0.000303977305129 \\ 0.000303977305128 & 0.250000167067392 \end{bmatrix},$$
$$P(0.0001) = \begin{bmatrix} 1.154700536323469 & 0.000030397737787 \\ 0.000030397737789 & 0.250000001670674 \end{bmatrix},$$
$$P(0.00001) = \begin{bmatrix} 1.154700538358694 & 0.000003039773787 \\ 0.000003039773786 & 0.250000000016707 \end{bmatrix}.$$

Hence, the following structure should be assumed.

$$P_\varepsilon = \begin{bmatrix} P_1 & \varepsilon P_{12} \\ \varepsilon P_{12}^T & P_2 \end{bmatrix}. \tag{12}$$

It should be noted that the assumption of this structure is also made in [12].

On the other hand, there exists a drawback that cannot be ignored for a simple numerical example. In other words, each of the solutions $P(0.001)$, $P(0.0001)$, and $P(0.00001)$ is not a symmetric matrix. It may be interesting to note that these practical and informative results are not shown in both undergraduate and graduate courses in general. As a result, when the ARE under the weakly coupled systems is considered, another numerical algorithm is required. In the following discussion, the focus is on a numerical algorithm that can obtain a symmetric solution for any small value of ε.

Using the implicit function theorem, let us prove the existence of the implicit functions $P_i = P_i(\varepsilon)$ and $P_{12} = P_{12}(\varepsilon)$ of ε such that

$$P_i = P_i(\varepsilon),\ i = 1,\ 2,\ \ P_{12} = P_{12}(\varepsilon). \tag{13}$$

In order to simplify the notation, the following matrix is defined.

$$\begin{aligned} S_\varepsilon &:= B_{2\varepsilon}B_{2\varepsilon}^T - \gamma^{-2}B_{1\varepsilon}B_{1\varepsilon}^T \\ &= \begin{bmatrix} S_1 + \varepsilon^2 S_{11} & \varepsilon(S_{112} + S_{122}) \\ \varepsilon(S_{112} + S_{122})^T & S_2 + \varepsilon^2 S_{22} \end{bmatrix}, \\ S_i &= S_i^T,\ S_{ii} = S_{ii}^T,\ i = 1,\ 2, \\ Q &:= C_1^T C_1 = \textbf{block diag}\begin{pmatrix} Q_1 & Q_2 \end{pmatrix}. \end{aligned}$$

Substituting the solution P_ε of equation (12) into the ARE (10), the set of algebraic matrix equations (14) is obtained.

$$\begin{aligned} \mathcal{G}_1 = {} & A_{11}^T P_1 + P_1 A_{11} + \varepsilon^2(A_{21}^T P_{12}^T + P_{12}A_{21}) \\ & - P_1 S_1 P_1 - \varepsilon^2(P_1 S_{11} P_1 + P_{12} S_{112}^T P_1 \\ & + P_{12} S_{122}^T P_1 + P_1 S_{112} P_{12}^T + P_1 S_{122} P_{12}^T \\ & + P_{12} S_2 P_{12}^T + \varepsilon^2 P_{12} S_{22} P_{12}^T) + Q_1 = 0, \end{aligned} \tag{14a}$$

$$\begin{aligned} \mathcal{G}_{12} = {} & A_{11}^T P_{12} + P_1 A_{12} + A_{21}^T P_2 + P_{12} A_{22} \\ & - P_1 S_1 P_{12} - P_1 S_{112} P_2 - P_1 S_{122} P_2 - P_{12} S_2 P_2 \\ & + \varepsilon^2(P_1 S_{11} P_{12} + P_{12} S_{112}^T P_{12} \\ & + P_{12} S_{122}^T P_{12} + \varepsilon^2 P_{12} S_{22} P_2) = 0, \end{aligned} \tag{14b}$$

$$\begin{aligned} \mathcal{G}_2 = {} & A_{22}^T P_2 + P_2 A_{22} + \varepsilon^2(A_{12}^T P_{12} + P_{12}^T A_{12}) \\ & - P_2 S_2 P_2 - \varepsilon^2(P_2 S_{22} P_2 + P_{12}^T S_{112} P_2 \\ & + P_{12}^T S_{122} P_2 + P_2 S_{112}^T P_{12} + P_2 S_{122}^T P_{12} \\ & + P_{12}^T S_1 P_{12} + \varepsilon^2 P_{12}^T S_{11} P_{12}) + Q_2 = 0, \end{aligned} \tag{14c}$$

where

$$\mathcal{G}(P_\varepsilon) := \begin{bmatrix} \mathcal{G}_1 & \mathcal{G}_{12} \\ \mathcal{G}_{12}^T & \mathcal{G}_2 \end{bmatrix} = A_\varepsilon^T P_\varepsilon + P_\varepsilon A_\varepsilon - P_\varepsilon S_\varepsilon P_\varepsilon + Q = 0.$$

The 0th order solutions $\bar{P}_i$, $i = 1,\ 2$, and $\bar{P}_{12}$ are defined for $\varepsilon \to +0$ for the algebraic matrix equations. Then, the solutions $\bar{P}_i$ satisfy the ARE (15)

$$A_{ii}^T \bar{P}_i + \bar{P}_i A_{ii} - \bar{P}_i S_i \bar{P}_i + Q_i = 0,\ i = 1,\ 2, \tag{15}$$

where $S_i := B_{1ii} B_{1ii}^T - \gamma^{-2} B_{2ii} B_{2ii}^T$.

The ARE (15) produces a unique positive semidefinite stabilizing solution if γ is sufficiently large.

Let
$\gamma_{P_i} = \inf\{\gamma > 0 |$ the ARE (15) has a positive semidefinite stabilizing solution$\}$.

Then, matrix $A_{ii} - S_i \bar{P}_i$ is nonsingular if $\gamma > \gamma_{P_i}$ is chosen. For the solution P_ε of the ARE (10), the result is given for $\varepsilon \to +0$.

Theorem 1: It is assumed that the reduced-order ARE (10), which is independent of the perturbation parameter ε, has a positive semidefinite stabilizing solution. If a parameter $\gamma > \bar{\gamma}_P = \max\{\gamma_{P_1},\ \gamma_{P_2}\}$ is selected, then there exists a small $\bar{\sigma}$ such that for all $\varepsilon \in (0,\ \bar{\sigma})$, the ARE admits a positive semidefinite stabilizing solution P_ε that can be written as equation (16).

$$P_\varepsilon = \bar{P} + O(\varepsilon) = \textbf{block diag}\begin{pmatrix} \bar{P}_1 & \bar{P}_2 \end{pmatrix} + O(\varepsilon). \tag{16}$$

Proof: The proof can be obtained by applying the implicit function theorem [12]. To achieve this, it is sufficient to show that the corresponding Jacobian is nonsingular at $\varepsilon = 0$. It can be shown after some simplification that the Jacobian of the ARE (3) with the limit $\varepsilon \to +0$ is given by (17).

$$\boldsymbol{J} = \begin{bmatrix} \boldsymbol{J}_1 & 0 & 0 \\ * & \boldsymbol{J}_{12} & * \\ 0 & 0 & \boldsymbol{J}_2 \end{bmatrix} \tag{17}$$

with

$$\boldsymbol{J}_i = (A_{ii} - S_i \bar{P}_i)^T \otimes I_{n_i} + I_{n_i} \otimes (A_{ii} - S_i \bar{P}_i)^T,$$
$$\boldsymbol{J}_{12} = (A_{22} - S_2 \bar{P}_2)^T \otimes I_{n_1} + I_{n_2} \otimes (A_{11} - S_1 \bar{P}_1)^T.$$

The Jacobian (17) can be expressed as

$$\det \boldsymbol{J} = \det \boldsymbol{J}_1 \det \boldsymbol{J}_{12} \det \boldsymbol{J}_2. \tag{18}$$

Apparently, $\boldsymbol{J}_i$ and $\boldsymbol{J}_{12}$ are nonsingular because the AREs (15) have positive semidefinite stabilizing solutions. Thus, $\det \boldsymbol{J} \neq 0$, i.e., $\boldsymbol{J}$ is nonsingular at $\varepsilon = 0$. Therefore, the existence of $\bar{\sigma}$ such that the ARE (10) has an asymptotic structure (16) is directly obtained by applying the implicit function theorem.

The remainder of the proof shows that P_ε is a positive semidefinite stabilizing solution. Using the asymptotic structure (16) for ε, equation (19) holds.

$$A_\varepsilon - S_\varepsilon P_\varepsilon = \textbf{block diag}\begin{pmatrix} A_{11} - S_1 \bar{P}_1 & A_{22} - S_2 \bar{P}_2 \end{pmatrix} + O(\varepsilon). \tag{19}$$

The matrices $A_{ii} - S_i \bar{P}_i$ are stable because the ARE (15) has a positive semidefinite stabilizing solution. Therefore, if ε is small, $A_\varepsilon - S_\varepsilon P_\varepsilon$ is also stable. Finally, the positive semidefiniteness can be proved by using the Schur complement [20] for a sufficiently small ε. □

3.1. Newton's method for solving the ARE

In this section, a new algorithm for solving the ARE (10), which can be calculated with a small dimension and attains quadratic convergence, is proposed.

The Schur vector method [21] is being widely used for solving the ARE (3) because the method has good precision and ensures the algorithm's stability. However, it is well known that in this method, the dimensions of the required workspace for the calculations are twice that of the original full system [21]. Furthermore, when epsilon is included in systems such as weakly coupled systems, it is easy to verify that the obtained solutions become asymmetric solutions. In fact, the simulation results for the Schur vector method have shown such a disadvantage. On the other hand, in [22], the following comments have been documented.

"Newton's method is potentially fast and more accurate than the widely used Schur vector method. The break-even point is between six and eight iterations assuming that a Bartels-Stewart-like algorithm is used to solve the algebraic Lyapunov equation (ALE)."

Thus, an algorithm that is based on Newton's method and uses the structure of the solution in equation (16) is considered.

Let us consider Newton's method (20).

$$P_\varepsilon^{(k+1)}(A_\varepsilon - S_\varepsilon P_\varepsilon^{(k)}) + (A_\varepsilon - S_\varepsilon P_\varepsilon^{(k)})^T P_\varepsilon^{(k+1)} + P_\varepsilon^{(k)} S_\varepsilon P_\varepsilon^{(k)} + Q_\varepsilon = 0,\ k = 0,\ 1,\ \dots, \tag{20a}$$

$$P_\varepsilon^{(k)} = \begin{bmatrix} P_1^{(k)} & \varepsilon P_{12}^{(k)} \\ \varepsilon P_{12}^{(k)T} & P_2^{(k)} \end{bmatrix}, \tag{20b}$$

where the initial conditions are chosen as follows.

$$P_\varepsilon^{(0)} = \bar{P} = \textbf{block diag}\begin{pmatrix} \bar{P}_1 & \bar{P}_2 \end{pmatrix}. \tag{21}$$

This method can be established as follows. Using the definition of Newton's method, the following equation holds.

$$\textbf{vec} P_\varepsilon^{(k+1)} = \textbf{vec} P_\varepsilon^{(k)} - \left[\frac{\partial \textbf{vec}\mathcal{G}(P_\varepsilon)}{\partial (\textbf{vec} P_\varepsilon)^T} \bigg|_{P_\varepsilon = P_\varepsilon^{(k)}} \right]^{-1} \textbf{vec}\mathcal{G}(P_\varepsilon^{(k)})$$

$$= \mathbf{vec}P_\varepsilon^{(k)} - \Big[(A_\varepsilon - S_\varepsilon P_\varepsilon^{(k)})^T \otimes I_n + I_n \otimes (A_\varepsilon - S_\varepsilon P_\varepsilon^{(k)})^T\Big]^{-1} \mathbf{vec}\mathcal{G}(P_\varepsilon^{(k)}),$$

where $n := n_1 + n_2$.

Thus, the operation $(A_\varepsilon - S_\varepsilon P_\varepsilon^{(k)})^T \otimes I_n + I_n \otimes (A_\varepsilon - S_\varepsilon P_\varepsilon^{(k)})^T$ yields

$$\Big[(A_\varepsilon - S_\varepsilon P_\varepsilon^{(k)})^T \otimes I_n + I_n \otimes (A_\varepsilon - S_\varepsilon P_\varepsilon^{(k)})^T\Big] \times (\mathbf{vec}P_\varepsilon^{(k+1)} - \mathbf{vec}P_\varepsilon^{(k)}) + \mathbf{vec}\mathcal{G}(P_\varepsilon^{(k)}) = 0.$$

Moreover, using the formulation $\mathbf{vec}(XA) = [A^T \otimes I_n]\mathbf{vec}X$, $\mathbf{vec}(A^TX) = [I_n \otimes A^T]\mathbf{vec}X$ results in

$$\mathbf{vec}\Big[(P_\varepsilon^{(k+1)} - P_\varepsilon^{(k)})(A_\varepsilon - S_\varepsilon P_\varepsilon^{(k)}) + (A_\varepsilon - S_\varepsilon P_\varepsilon^{(k)})^T (P_\varepsilon^{(k+1)} - P_\varepsilon^{(k)})\Big] + \mathbf{vec}\mathcal{G}(P_\varepsilon^{(k)})$$
$$= \mathbf{vec}\Big[P_\varepsilon^{(k+1)}(A_\varepsilon - S_\varepsilon P_\varepsilon^{(k)}) + (A_\varepsilon - S_\varepsilon P_\varepsilon^{(k)})^T P_\varepsilon^{(k+1)} + P_\varepsilon^{(k)} S_\varepsilon P_\varepsilon^{(k)} + Q_\varepsilon\Big] = 0,$$

which is the desired result.

It should be noted that Newton's method is equivalent to the existing Kleinman algorithm [23].

The algorithm represented by equation (21) has the feature given in the following theorem.

Theorem 2: If the parameter-independent reduced-order ARE (15) has a positive semidefinite stabilizing solution, there exists a small $\tilde{\sigma}$ such that for all $\varepsilon \in (0,\ \tilde{\sigma})$, $0 < \tilde{\sigma} \le \bar{\sigma}$, the iterative algorithm represented by equation (20a) converges to the exact solution of P_ε with a rate equal to that of quadratic convergence; here, $P_\varepsilon^{(k)}$ is positive semidefinite. Moreover, the convergence solutions equal those of P_ε in the ARE (10) in the neighborhood of the initial condition $P_\varepsilon^{(0)} = \bar{P}$. Subsequently, equation (22) holds.

$$\|P_\varepsilon^{(k)} - P_\varepsilon\| = \frac{O(\varepsilon^{2^k})}{\beta\lambda 2^k},\ k = 0,\ 1,\ \dots, \tag{22}$$

where

$$\lambda = 2\|S_\varepsilon\| < \infty,\ \beta = \|[\nabla\mathcal{G}(P_\varepsilon^{(0)})]^{-1}\|,$$
$$\eta = \beta \cdot \|\mathcal{G}(P_\varepsilon^{(0)})\|,\ \theta = \beta\eta\lambda,$$
$$\nabla\mathcal{G}(P_\varepsilon) = \frac{\partial \mathbf{vec}\mathcal{G}(P_\varepsilon)}{\partial(\mathbf{vec}P_\varepsilon)^T},$$
$$\mathcal{G}(P_\varepsilon) = P_\varepsilon A_\varepsilon + A_\varepsilon^T P_\varepsilon - P_\varepsilon S_\varepsilon P_\varepsilon + Q.$$

Proof: The proof follows directly by applying the Newton-Kantorovich theorem [15, 16]. Taking the partial derivative of the ARE (3) with respect to P_ε yields

$$\nabla\mathcal{G}(P_\varepsilon) := \frac{\partial \mathbf{vec}\mathcal{G}(P_\varepsilon)}{\partial(\mathbf{vec}P_\varepsilon)^T} = (A_\varepsilon - S_\varepsilon P_\varepsilon)^T \otimes I_n + I_n \otimes (A_\varepsilon - S_\varepsilon P_\varepsilon)^T. \tag{23}$$

Thus, for any $P_{a\varepsilon}$ and $P_{b\varepsilon} \in \mathbb{R}^{n\times n}$, the following inequality holds.

$$\|\nabla\mathcal{G}(P_{a\varepsilon}) - \nabla\mathcal{G}(P_{b\varepsilon})\| \le \lambda\|P_{a\varepsilon} - P_{b\varepsilon}\|, \tag{24}$$

where $\lambda = 2\|S_\varepsilon\|$.

Moreover, using the stability that is established by equation (19), it is shown that there exists a small $\tilde{\sigma}$ such that for $\varepsilon \in (0,\ \tilde{\sigma})$, $\tilde{\sigma} \le \bar{\sigma}$, $\nabla\mathcal{G}(P_\varepsilon)$ is nonsingular. Therefore, there exists β such that $\|[\nabla\mathcal{G}(P_\varepsilon)]^{-1}\| \equiv \beta$. On the other hand, using Theorem 1, it is easy to show that $\|\mathcal{G}(P_\varepsilon)\| = O(\varepsilon)$. Hence, there exists η such that $\|[\nabla\mathcal{G}(P_\varepsilon)]^{-1}\| \cdot \|\mathcal{G}(P_\varepsilon)\| \equiv \eta = O(\varepsilon)$. Thus, for a sufficiently small ε, there exists θ such that $\theta \equiv \beta\lambda\eta < 2^{-1}$ because $\eta = O(\varepsilon)$. Thus, using the Newton-Kantorovich theorem, there exists a small $\tilde{\sigma}$ such that for all $\varepsilon \in (0,\ \tilde{\sigma})$, $\tilde{\sigma} \le \bar{\sigma}$, the iterative algorithm (20a) has quadratic convergence.

Second, the uniqueness of the solution is discussed. Now, let us define $\bar{t}^* \equiv [1 - \sqrt{1-2\theta}]/(\beta\lambda)$. Clearly, $S \equiv \{\ P_\varepsilon\ :\ \|P_\varepsilon - P_\varepsilon^{(0)}\| \le \bar{t}^*\ \}$ is in the convex set D. In the sequel, since $\|P_\varepsilon - P_\varepsilon^{(0)}\| = O(\varepsilon)$ holds for a small ε, the uniqueness of P_ε is guaranteed for subset S by applying the Newton-Kantorovich theorem. □

3.2. Fixed point algorithm for solving ALE

It is possible to solve the algorithm in equation (20a) by using a linear equation because such an algorithm is based on the ALE. However, this method results in an increase in the workspace dimensions for the numerical computation when the dimensions of the matrices P_1, P_{12}, and P_2 increase. That is, let us consider the following linear equation (25).

$$(20) \Leftrightarrow A^{(k)} \begin{bmatrix} \mathbf{vec}P_1^{(k+1)} \\ \mathbf{vec}P_{12}^{(k+1)} \\ \mathbf{vec}P_2^{(k+1)} \end{bmatrix} = - \begin{bmatrix} \mathbf{vec}\bar{Q}_1 \\ \mathbf{vec}\bar{Q}_{12} \\ \mathbf{vec}\bar{Q}_2 \end{bmatrix}, \tag{25}$$

where $A^{(k)}$ is given in the top of the next page.

In this case, if the dimension n is large, the dimension of $A^{(k)}$ would be quite large because the Kronecker products are used. Hence, in order to reduce the dimension of the workspace, a new algorithm for solving the ALE (20a) that is based on the fixed point algorithm is formulated. Let us consider ALE (26) in the following general form: $P_\varepsilon^{(k+1)} \to X_\varepsilon$, $A_\varepsilon - S_\varepsilon P_\varepsilon^{(k)} \to E_\varepsilon$ and $P_\varepsilon^{(k)} S_\varepsilon P_\varepsilon^{(k)} + Q_\varepsilon \to H_\varepsilon$.

$$E_\varepsilon^T X_\varepsilon + X_\varepsilon E_\varepsilon + H_\varepsilon = 0, \tag{26}$$

where $E_i \in \mathbb{R}^{n_i\times n_i}$ is stable and $X_i = X_i^T \ge 0 \in \mathbb{R}^{n_i\times n_i}$, $H_i = H_i^T \in \mathbb{R}^{n_i\times n_i}$, $i = 1,\ 2$.

$$E_\varepsilon := \begin{bmatrix} E_1 & \varepsilon E_{12} \\ \varepsilon E_{21} & E_2 \end{bmatrix},\ H_\varepsilon := \begin{bmatrix} H_1 & \varepsilon H_{12} \\ \varepsilon H_{12}^T & H_2 \end{bmatrix},$$

$$A^{(k)} := \begin{bmatrix} \bar{A}_1^T \otimes I_{n_1} + I_{n_1} \otimes \bar{A}_1^T & \varepsilon^2(\bar{A}_{21}^T \otimes I_{n_1} + I_{n_1} \otimes \bar{A}_{21}^T) & 0 \\ \bar{A}_{12}^T \otimes I_{n_2} & \bar{A}_2^T \otimes I_{n_1} + I_{n_2} \otimes \bar{A}_1^T & I_{n_1} \otimes \bar{A}_{21}^T \\ 0 & \varepsilon^2(\bar{A}_{21}^T \otimes I_{n_1} + (I_{n_1} \otimes \bar{A}_{21}^T)\boldsymbol{U}_{n_1 n_2}) & (\bar{A}_{12} \otimes I_{n_2})\boldsymbol{U}_{n_1 n_2} + I_{n_2} \otimes \bar{A}_{12} \end{bmatrix},$$

$$A_\varepsilon - S_\varepsilon P_\varepsilon^{(k)} := \begin{bmatrix} \bar{A}_1 & \varepsilon\bar{A}_{12} \\ \varepsilon\bar{A}_{21} & \bar{A}_2 \end{bmatrix}, \; P_\varepsilon^{(k)} S_\varepsilon P_\varepsilon^{(k)} + Q_\varepsilon := \begin{bmatrix} \bar{Q}_1 & \varepsilon\bar{Q}_{12} \\ \varepsilon\bar{Q}_{12}^T & \bar{Q}_2 \end{bmatrix}.$$

$$X_\varepsilon = \begin{bmatrix} X_1 & \varepsilon X_{12} \\ \varepsilon X_{12}^T & X_2 \end{bmatrix}.$$

Substituting X_ε into the ALE (26), the following set of three linear equations (27) hold.

$$E_1^T X_1 + X_1 E_1 + \varepsilon^2(E_{21}^T X_{12}^T + X_{12} E_{21}) + H_1 = 0, \quad (27a)$$

$$E_1^T X_{12} + X_1 E_{12} + E_{21}^T X_2 + X_{12} E_2 + H_{12} = 0, \quad (27b)$$

$$E_2^T X_2 + X_2 E_2 + \varepsilon^2(E_{12}^T X_{12} + X_{12}^T E_{12}) + H_2 = 0. \quad (27c)$$

By considering the form of (27), the following algorithm in (28) to solve the ALE (26) is proposed.

$$\begin{aligned} &E_1^T X_1^{(k+1)} + X_1^{(k+1)} E_1 \\ &\quad + \varepsilon^2(E_{21}^T X_{12}^{(k)T} + X_{12}^{(k)} E_{21}) + H_1 = 0, \quad (28a) \\ &E_2^T X_2^{(k+1)} + X_2^{(k+1)} E_2 \\ &\quad + \varepsilon^2(E_{12}^T X_{12}^{(k)} + X_{12}^{(k)T} E_{12}) + H_2 = 0, \quad (28b) \\ &E_1^T X_{12}^{(k+1)} + X_{12}^{(k+1)} E_2 \\ &\quad + X_1^{(k+1)} E_{12} + E_{21}^T X_2^{(k+1)} + H_{12} = 0, \quad (28c) \\ &X_i^{(0)} = 0, \; i = 1, \; 2, X_{12} = 0, \; k = 0, \; 1, \; \ldots. \end{aligned}$$

The following theorem indicates the convergence of the algorithm in (28).

Theorem 3: If E_i, $i = 1, 2$ is stable, there exists a small $\hat{\sigma}$ such that for all $\varepsilon \in (0, \; \hat{\sigma})$, $0 < \hat{\sigma}$, the iterative algorithm in (28) converges to the exact solutions of X_i and X_{21} with a rate equal to that of linear convergence. Subsequently, equation (29) can be obtained.

$$\|X_i^{(k)} - X_i\| = O(\varepsilon^{2k}), \; i = 1, \; 2, \quad (29a)$$

$$\|X_{12}^{(k)} - X_{12}\| = O(\varepsilon^{2k}), \; k = 1, \; 2, \; \ldots. \quad (29b)$$

Proof: The proof follows directly by applying the fixed point theorem [18]. First, it is easy to verify that the algorithms in (28) and (30) are identical.

$$\begin{aligned} &X_1^{(k+1)} \\ &:= \mathcal{Z}_1(X_{12}^{(k)}) \\ &= \varepsilon^2 \int_0^\infty \exp(E_1^T s)(E_{21}^T X_{12}^{(k)T} + X_{12}^{(k)} E_{21}) \\ &\quad \times \exp(E_1 s) ds + \int_0^\infty \exp(E_1^T s) H_1 \exp(E_1 s) ds, \end{aligned} \quad (30a)$$

$$\begin{aligned} &X_2^{(k+1)} \\ &:= \mathcal{Z}_2(X_{12}^{(k)}) \\ &= \varepsilon^2 \int_0^\infty \exp(E_2^T s)(E_{12}^T X_{12}^{(k)} + X_{12}^{(k)T} E_{12}) \exp(E_2 s) ds \\ &\quad + \int_0^\infty \exp(E_2^T s) H_2 \exp(E_2 s) ds, \end{aligned} \quad (30b)$$

$$X_{12}^{(k+1)} := \mathcal{Z}_3(X_1^{(k+1)}, \; X_2^{(k+1)}) = \mathcal{Z}_3(X_{12}^{(k)}). \quad (30c)$$

Thus, taking into account the stability of E_1 with regard to (30a), there exist $m_1 > 0$ and $\phi_1 > 0$ such that $\|\exp(E_1^T s)\| \le m_1 \exp(-\phi_1 s)$ [24]. As a result, for any X_{12}^a and X_{12}^b

$$\begin{aligned} &\|\mathcal{Z}_1(X_{12}^a) - \mathcal{Z}_1(X_{12}^b)\| \\ &= \varepsilon^2 \|E_{21}\| \cdot \|X_{12}^a - X_{12}^b\| \int_0^\infty m_1^2 \exp(-2\phi_1 s) ds \\ &= \varepsilon^2 \mathcal{M}_1 \|X_{12}^a - X_{12}^b\|, \end{aligned}$$

there exist $\mathcal{M}_1$ and the parameter $\varepsilon = \varepsilon_1$ such that $\varepsilon^2 \mathcal{M}_1 < 1$. Using a technique similar to that given above, there exist $\mathcal{M}_i$ and the parameter $\varepsilon = \varepsilon_i$, $i = 2, \; 3$ such that $\varepsilon^2 \mathcal{M}_i < 1$.

$$\|\mathcal{Z}_2(X_{12}^a) - \mathcal{Z}_2(X_{12}^b)\| = \varepsilon^2 \mathcal{M}_2 \|X_{12}^a - X_{12}^b\|,$$
$$\|\mathcal{Z}_3(X_{12}^a) - \mathcal{Z}_3(X_{12}^b)\| = \varepsilon^2 \mathcal{M}_3 \|X_{12}^a - X_{12}^b\|.$$

Thus, it is easy to verify that the algorithm in (28) attains linear convergence for the fixed point theorem. □

4. NUMERICAL EXAMPLE

In order to verify the efficiency of the proposed algorithms in (20) and (28), a numerical example is provided. The system matrices of the weakly coupled systems (9) are given as follows [25]:

$$A_{11} = \begin{bmatrix} 0 & 1 & -0.266 & -0.009 \\ -2.75 & -2.78 & -1.36 & -0.037 \\ 0 & 0 & 0 & 1 \\ -4.95 & 0 & -55.5 & -0.039 \end{bmatrix},$$

$$\varepsilon A_{12} = \begin{bmatrix} 0.0024 & 0 & -0.087 & 0.002 \\ -0.185 & 0 & 1.11 & -0.011 \\ 0 & 0 & 0 & 0 \\ 0.222 & 0 & 8.17 & 0.004 \end{bmatrix},$$

$$\varepsilon A_{21} = \begin{bmatrix} 0.021 & 0 & 0.121 & 0.003 \\ -1.1 & 0 & -1.62 & -0.015 \\ 0 & 0 & 0 & 0 \\ -2.43 & 0 & 1.37 & -0.034 \end{bmatrix},$$

$$A_{22} = \begin{bmatrix} -0.21 & 1 & -1.6 & -0.005 \\ -1.9 & -1.8 & 9.3 & -0.12 \\ 0 & 0 & 0 & 1 \\ -3.1 & 0 & -56 & 0.032 \end{bmatrix},$$

$$B_{111} = \begin{bmatrix} 0 \\ 36.1 \\ 0 \\ 0 \end{bmatrix}, B_{122} = \begin{bmatrix} 0 \\ 78.9 \\ 0 \\ 0 \end{bmatrix},$$

$$B_{112} = B_{121} = 0,$$

$$B_{211} = \begin{bmatrix} 0 \\ 3.5 \\ 0 \\ 0 \end{bmatrix}, B_{222} = \begin{bmatrix} 0 \\ 4.2 \\ 0 \\ 0 \end{bmatrix}, B_{212} = B_{221} = 0,$$

$$C_1 = \begin{bmatrix} \sqrt{0.5} I_8 \\ 0_{2\times 8} \end{bmatrix}, D_1 = \begin{bmatrix} 0_{8\times 2} \\ I_2 \end{bmatrix}.$$

The two basic quantities are $\gamma_{P_1} = 9.7396 \times 10^{-2}$ and $\gamma_{P_2} = 5.3678 \times 10^{-2}$. Thus, for every boundary value $\gamma > \bar{\gamma}_P = \max\{\gamma_{P_1}, \gamma_{P_2}\} = 9.7396 \times 10^{-2}$, the ARE (10) has a positive semidefinite stabilizing solution for a sufficiently small ε. On the other hand, using MATLAB, the minimum value of γ^* such that there exists a dynamic feedback controller is $\gamma^* = 9.7396 \times 10^{-2}$ for $\varepsilon = 10^{-3}$.

In order to verify the exactitude of the solution, the error per iteration in Table 1 can be calculated, where $\varepsilon = 0.1$, $\gamma = 10$, and the convergence condition is given by $\|\mathcal{G}(P_\varepsilon^{(k)})\| < 10^{-10}$.

In Table 1, it is easy to verify that the proposed algorithm corresponds to quadratic convergence. Moreover, Table 2 shows the errors $\|\mathcal{G}(P_\varepsilon^{(k)})\|$ per iteration for various values of ε with $\gamma = 10$. Hence, it can be observed from Table 2 that the algorithm in (20a) attains quadratic convergence.

Table 1. Error per iterations.

k	$\|\mathcal{G}(P_\varepsilon^{(k)})\|$
0	6.3881×10^{-1}
1	4.6193×10^{-2}
2	3.1037×10^{-4}
3	8.6281×10^{-10}
4	4.3453×10^{-11}

Table 2. Error per iterations.

k	$\|\mathcal{L}(X_\varepsilon^{(k)})\|$
0	6.3881×10^{-1}
1	7.3149×10^{-1}
2	3.1375×10^{-3}
3	5.5756×10^{-5}
4	5.7489×10^{-7}
5	6.9411×10^{-9}
6	7.4439×10^{-11}

The convergence of the algorithm in (28) is demonstrated. For $\varepsilon = 0.1$, Table 3 shows the errors per iteration for the algorithm for the first iteration of Newton's method, where the convergence condition is given by $\|\mathcal{L}(X_\varepsilon^{(k)})\| = \|E_\varepsilon^T X_\varepsilon^{(k)} + X_\varepsilon^{(k)} E + H_\varepsilon\| < 10^{-10}$. From Table 3, it can be verified that the proposed algorithm satisfies (29). For various values of ε, it can also be found from Table 4 that the proposed algorithm satisfies the linear convergence based on (29). Hence, the combined algorithms in (20) and (28) of this paper are very attractive in the sense that the initial condition (21) of Newton's method guarantees the quadratic convergence for a small epsilon. Furthermore, even if the large-scale systems (1) are composed of two four-dimensional subsystems, the required workspace is four. This feature is very useful from the practical viewpoint. Finally, since OCTAVE programming is easy, these algorithms appear to be useful in the

Table 3. $\|\mathcal{G}(P_\varepsilon^{(k)})\|$.

$k \setminus \varepsilon$	$\varepsilon = 10^{-2}$	$\varepsilon = 10^{-3}$	$\varepsilon = 10^{-4}$	$\varepsilon = 10^{-5}$	$\varepsilon = 10^{-6}$
0	6.3881×10^{-2}	6.3881×10^{-3}	6.3881×10^{-4}	6.3881×10^{-5}	6.3883×10^{-6}
1	3.5601×10^{-4}	3.5458×10^{-6}	3.5444×10^{-8}	3.4186×10^{-10}	2.7829×10^{-11}
2	3.5601×10^{-4}	2.2232×10^{-11}	2.0435×10^{-11}	2.1110×10^{-11}	
3	2.5761×10^{-11}				

Table 4. $\|\mathcal{L}(X_\varepsilon^{(k)})\|$.

$k \setminus \varepsilon$	$\varepsilon = 10^{-2}$	$\varepsilon = 10^{-3}$	$\varepsilon = 10^{-4}$	$\varepsilon = 10^{-5}$	$\varepsilon = 10^{-6}$
0	6.3881×10^{-2}	6.3881×10^{-3}	6.3881×10^{-4}	6.3881×10^{-5}	6.3881×10^{-6}
1	7.3149×10^{-3}	7.3149×10^{-5}	7.3149×10^{-7}	7.3150×10^{-9}	7.3258×10^{-11}
2	3.1375×10^{-7}	3.1215×10^{-11}	1.4713×10^{-13}	1.0316×10^{-13}	
3	5.5843×10^{-11}				

numerical analysis in the control engineering course.

5. PRACTICE EXERCISE

Finally, the following optimization problem is discussed for the practice exercise in numerical analysis in the control engineering course. Let us consider the following example exercise.

Exercise: Solve the following optimization problem with inequality constraint.

$$\min_{P,K,\varepsilon} \mathbf{Tr}[P] \tag{31a}$$

subject to

$$\begin{aligned}\boldsymbol{F}_1(P,K,\varepsilon) &:= P(A+BKC)+(A+BKC)^TP+\varepsilon PDD^TP \\ &\quad +\varepsilon^{-1}(E_1+E_2KC)^T(E_1+E_2KC) \\ &\quad +C^TK^TRKC+Q=0,\end{aligned} \tag{31b}$$

where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{\ell\times n}$, $D \in \mathbb{R}^{n\times p}$, $E_1 \in \mathbb{R}^{p\times n}$, $E_2 \in \mathbb{R}^{p\times m}$, $Q = Q^T > 0$, $Q \in \mathbb{R}^{n\times n}$ and $R = R^T > 0$, $R \in \mathbb{R}^{m\times m}$ are given. On the other hand, $P \in \mathbb{R}^{n\times n}$, $K \in \mathbb{R}^{m\times \ell}$ and $\varepsilon \in \mathbb{R}$ are the optimization variables.

It should be noted that this problem originated in [26]. As an extension of the Lagrange theory for a classical optimization problem, it can be extended to the so-called Karush-Kuhn-Tucker (KKT) conditions by the following theorem.

Karush-Kuhn-Tucker (KKT) conditions [20]**:** Assume that $f(\boldsymbol{x})$, $h_i(\boldsymbol{x}) \in \mathbb{R}$, $i = 1, \ldots, m$ are all differentiable. Let us consider the following basic mathematical optimization problem subject to m inequality constraints.

$$\min_{\boldsymbol{x}} f(\boldsymbol{x}) \tag{32a}$$

$$\text{subject to } h_i(\boldsymbol{x}) \le 0,\ i = 1, \ldots, m. \tag{32b}$$

If the function $f(\boldsymbol{x})$ attains at point $\boldsymbol{x}^*$ a local minimum subject to the set $K = \{\boldsymbol{x} \mid f_i(\boldsymbol{x}) \le 0 (i = 1, \ldots, m)\}$, then there exists a vector of Lagrange multipliers $\boldsymbol{S}_0$, $\boldsymbol{S}_0 = \begin{bmatrix} S_1 & \cdots & S_m \end{bmatrix}$ such that the following conditions are satisfied:

$$\frac{\partial f(\boldsymbol{x}^*)}{\partial \boldsymbol{x}} + \sum_{i=1}^{m} S_i \frac{\partial h_i(\boldsymbol{x}^*)}{\partial \boldsymbol{x}} = 0, \tag{33a}$$

$$h_i(\boldsymbol{x}^*) \le 0, \tag{33b}$$

$$S_i h_i(\boldsymbol{x}^*) = 0, \tag{33c}$$

$$S_i \ge 0,\ i = 1, \ldots, m. \tag{33d}$$

In other words, the conditions (33) are necessary conditions for a local minimum.

In [26], it should be noted that the theory of Lagrange multipliers was used to derive the necessary conditions except for ε. Specifically, for the variable ε, linear search was considered. In contrast with [26], a complete optimization problem is solved using the KKT conditions in this paper.

First, let us define the following Hamiltonian (34)

$$L(P,G,K,\varepsilon) := \mathbf{Tr}[P] + \mathbf{Tr}[G\boldsymbol{F}_1(P,K,\varepsilon)], \tag{34}$$

where $G \in \mathbb{R}^{n\times n}$ is a symmetric matrix of Lagrange multipliers.

Then, the higher order nonlinear matrix equations can be derived using the KKT conditions.

$$\frac{\partial L}{\partial G} = \boldsymbol{F}_1(P,K,\varepsilon) = 0, \tag{35a}$$

$$\frac{\partial L}{\partial P} = \boldsymbol{F}_2(P,G,K,\varepsilon) = GA_K^T + A_KG + I_n = 0, \tag{35b}$$

$$\begin{aligned}\frac{\partial L}{\partial K} &= [B^TP + \varepsilon^{-1}E_2^TE_1 \\ &\quad + (\varepsilon^{-1}E_2^TE_2 + R)KC]G = 0,\end{aligned} \tag{35c}$$

$$\begin{aligned}\frac{\partial L}{\partial \varepsilon} &= \boldsymbol{F}_4(P,G,K,\varepsilon) \\ &= \mathbf{Tr}[GPDD^TP] - \varepsilon^{-2}\mathbf{Tr}[GE_K^TE_K] = 0,\end{aligned} \tag{35d}$$

where $A_K = A + BKC + \varepsilon DD^TP$ and $E_K = E_1 + E_2KC$.

5.1. Computational algorithms

In order to obtain the solutions, P, G, K and ε, Newton's method can be used.

$$\begin{aligned}&\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - [\boldsymbol{J}^{(k)}]^{-1}\boldsymbol{f}(\boldsymbol{x}^{(k)}), \\ &\boldsymbol{x}^{(0)} = \boldsymbol{x}^0,\ \ k = 0, 1, 2, 3, \ldots,\end{aligned} \tag{36}$$

where

$$\boldsymbol{x}^{(k)} := \begin{bmatrix} (\mathbf{vec}P^{(k)})^T & (\mathbf{vec}G^{(k)})^T & (\mathbf{vec}K^{(k)})^T & \varepsilon^{(k)} \end{bmatrix}^T,$$

$$\boldsymbol{f}(\boldsymbol{x}^{(k)}) := \begin{bmatrix} (\mathbf{vec}\boldsymbol{F}_1^{(k)})^T & \cdots & (\mathbf{vec}\boldsymbol{F}_4^{(k)})^T \end{bmatrix}^T,$$

$$\boldsymbol{F}_1^{(k)} := \boldsymbol{F}_1(P^{(k)}, K^{(k)}, \varepsilon^{(k)}),$$

$$\boldsymbol{F}_i^{(k)} := \boldsymbol{F}_i(P^{(k)}, G^{(k)}, K^{(k)}, \varepsilon^{(k)}),\ i = 2, 3, 4,$$

$$\begin{aligned}\boldsymbol{F}_3 = \boldsymbol{F}_3(P,G,K,\varepsilon) &= [B^TP + \varepsilon^{-1}E_2^TE_1 \\ &\quad + (\varepsilon^{-1}E_2^TE_2 + R)KC]GC^T = 0,\end{aligned}$$

$$\boldsymbol{J}^{(k)} := \boldsymbol{J}(P^{(k)}, G^{(k)}, K^{(k)}, \varepsilon^{(k)})$$

$$\begin{aligned}\nabla\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{J}(P, G, K, \varepsilon) &= \frac{\partial \boldsymbol{f}(\boldsymbol{x})}{\partial \boldsymbol{x}} \\ &= \begin{bmatrix} J_{11} & 0 & J_{13} & J_{14} \\ J_{21} & J_{22} & J_{23} & J_{24} \\ J_{31} & J_{32} & J_{33} & J_{34} \\ J_{41} & J_{42} & J_{43} & J_{44} \end{bmatrix},\end{aligned}$$

$$\boldsymbol{x} := \begin{bmatrix} (\mathbf{vec}P)^T & (\mathbf{vec}G)^T & (\mathbf{vec}K)^T & \varepsilon \end{bmatrix}^T,$$

$$J_{i1} = \frac{\partial \mathbf{vec}\boldsymbol{F}_i}{\partial (\mathbf{vec}P)^T},\ J_{i2} = \frac{\partial \mathbf{vec}\boldsymbol{F}_i}{\partial (\mathbf{vec}G)^T},$$

$$J_{i3} = \frac{\partial \mathbf{vec}\boldsymbol{F}_i}{\partial (\mathbf{vec}K)^T},\ J_{i4} = \frac{\partial \mathbf{vec}\boldsymbol{F}_i}{\partial \varepsilon},$$

$$
\begin{aligned}
J_{11} &:= I_n \otimes A_K^T + A_K^T \otimes I_n, \\
J_{13} &:= C^T \\
&\quad \otimes (PB + \varepsilon^{-1}(E_1^T E_2 + C^T K^T E_2^T E_2) + C^T K^T R) \\
&\quad + (PB + \varepsilon^{-1}(E_1^T E_2 + C^T K^T E_2^T E_2) + C^T K^T R) \\
&\quad \otimes C^T, \\
J_{14} &:= \mathbf{vec}(PDD^T P - \varepsilon^{-2} E_K^T E_K), \\
J_{21} &:= G \otimes (\varepsilon DD^T) + (\varepsilon DD^T) \otimes G, \\
J_{22} &:= I_n \otimes A_K + A_K \otimes I_n, \\
J_{23} &:= (GC^T) \otimes B + B \otimes (GC^T), \\
J_{24} &:= \mathbf{vec}(GPDD^T + DD^T PG),\ J_{31} := (CG) \otimes B^T, \\
J_{32} &:= C \otimes [B^T P + \varepsilon^{-1} E_2^T E_1 + (\varepsilon^{-1} E_2^T E_2 + R) KC], \\
J_{33} &:= (CGC^T) \otimes (\varepsilon^{-1} E_2^T E_2 + R), \\
J_{34} &:= -\varepsilon^{-2} \mathbf{vec}[(E_2^T E_1 + E_2^T E_2 KC) GC^T)] \\
J_{41} &:= [\mathbf{vec}(DD^T PG + GPDD^T)]^T, \\
J_{42} &:= [\mathbf{vec}(PDD^T P - \varepsilon^{-2} E_K^T E_K)]^T, \\
J_{43} &:= -\varepsilon^{-2}(E_2^T E_1 GC^T + CGE_1^T E_2 U_{m\ell} \\
&\quad + E_2^T E_2 KCGC^T + CGC^T K^T E_2 E_2 U_{m\ell}), \\
J_{44} &:= 2\varepsilon^{-3} \mathbf{Tr}[GE_K^T E_K].
\end{aligned}
$$

The initial conditions $\boldsymbol{x}^0$ can be chosen such that the closed loop system is asymptotically stable.

It is well known that Newton's method attains local quadratic convergence. However, the computation of the Jacobi matrix $\boldsymbol{J}$ is complicated. Moreover, for all k, the Jacobi matrix $\boldsymbol{J}$ should be nonsingular. In order to avoid such disadvantages, the steepest descent method can be used.

Consider the following difference equation based on the steepest descent method.

$$
\begin{aligned}
\boldsymbol{x}^{(k+1)} &= \boldsymbol{x}^{(k)} - \mu\{\boldsymbol{x}^{(k)} - \boldsymbol{g}(\boldsymbol{x}^{(k)})\}, \\
\boldsymbol{x}^{(0)} &= \boldsymbol{x}^0,\ \ k = 0,\ 1,\ 2,\ 3,\ \ldots,
\end{aligned} \tag{37}
$$

where

$$
\begin{aligned}
&\boldsymbol{g}(\boldsymbol{x}^{(k)}) \\
&:= \left[\ (\mathbf{vec}\tilde{P}^{(k)})^T\ \ (\mathbf{vec}\tilde{G}^{(k)})^T\ \ (\mathbf{vec}\tilde{K}^{(k)})^T\ \ \tilde{\varepsilon}^{(k)}\ \right]^T, \\
&\mathbf{vec}\tilde{P}^{(k)} := \boldsymbol{G}_1(K^{(k)},\ \varepsilon^{(k)}), \\
&\mathbf{vec}\tilde{G}^{(k)} := \boldsymbol{G}_2(P^{(k)},\ K^{(k)},\ \varepsilon^{(k)}), \\
&\tilde{K}^{(k)} := \boldsymbol{G}_3(P^{(k)},\ G^{(k)},\ \varepsilon^{(k)}), \\
&\tilde{\varepsilon}^{(k)} := \boldsymbol{G}_4(P^{(k)},\ G^{(k)},\ K^{(k)}), \\
&\boldsymbol{G}_1(K,\ \varepsilon) = [A_K^T \otimes I_n + I_n \otimes A_K^T]^{-1} \mathbf{vec}(\varepsilon PDD^T P \\
&\qquad - \varepsilon^{-1} E_K^T E_K - C^T K^T RKC - Q), \\
&\boldsymbol{G}_2(P,\ K,\ \varepsilon) = -[A_K \otimes I_n + I_n \otimes A_K]^{-1} \mathbf{vec} I_n, \\
&\boldsymbol{G}_3(P,\ G,\ \varepsilon) = -(\varepsilon^{-1} E_2^T E_2 + R)^{-1} (B^T P + \varepsilon^{-1} E_2^T E_1) \\
&\qquad \times GC^T (CGC^T)^{-1}, \\
&\boldsymbol{G}_4(P,\ G,\ K) = \sqrt{\frac{\mathbf{Tr}[GE_K^T E_K]}{\mathbf{Tr}[GPDD^T P]}}.
\end{aligned}
$$

Furthermore, μ is a small positive parameter.

Next, the convergence proof is considered. The Taylor expansion expresses $\boldsymbol{g}(\boldsymbol{x})$ as follows:

$$
\boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{g}(\boldsymbol{\alpha}) + \nabla \boldsymbol{g}(\boldsymbol{\alpha})(\boldsymbol{x} - \boldsymbol{\alpha}) + O\left((\boldsymbol{x} - \boldsymbol{\alpha})^2\right), \tag{38}
$$

where $\boldsymbol{x} = \boldsymbol{\alpha}$ is a solution for equations $\boldsymbol{f}(\boldsymbol{x}) = 0$ or the equivalently to $\boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{x}$. Hence, we have

$$
\begin{aligned}
&\|\boldsymbol{x}^{(k+1)} - \boldsymbol{\alpha}\| \\
&= \|\boldsymbol{x}^{(k)} - \boldsymbol{\alpha} - \mu\{\boldsymbol{x}^{(k)} - \boldsymbol{\alpha} - \boldsymbol{g}(\boldsymbol{x}^{(k)}) + \boldsymbol{g}(\boldsymbol{\alpha})\} \\
&\le \left\|(1-\mu) I_{\bar{n}} + \mu \nabla \boldsymbol{g}(\boldsymbol{\alpha})\right\| \cdot \|\boldsymbol{x}^{(k)} - \boldsymbol{\alpha}\| \\
&\quad + \mu O\left(\|\boldsymbol{x}^{(k)} - \boldsymbol{\alpha}\|^2\right),
\end{aligned} \tag{39}
$$

where $\bar{n} := 2n^2 + m\ell + 1$.

In this case, by letting $\boldsymbol{\beta} = \boldsymbol{\alpha} + O(\mu)$, we have

$$
\begin{aligned}
\|\boldsymbol{x}^{(k+1)} - \boldsymbol{\alpha}\| &\le \left\|(1-\mu) I_{\bar{n}} + \mu \nabla \boldsymbol{g}(\boldsymbol{\beta}) + O(\mu^2)\right\| \\
&\quad \times \|\boldsymbol{x}^{(k)} - \boldsymbol{\alpha}\| + \mu O\left(\|\boldsymbol{x}^{(k)} - \boldsymbol{\alpha}\|^2\right).
\end{aligned} \tag{40}
$$

Finally, the considered difference equation is locally and asymptotically stable, if

$$
\left\|(1-\mu) I_{\bar{n}} + \mu \nabla \boldsymbol{g}(\boldsymbol{\beta}) + O(\mu^2)\right\| < 1. \tag{41}
$$

This means that if condition (41) holds, the steepest descent method converges because of the fixed point theorem. In other words, if $\boldsymbol{\beta}$ is very close to a solution $\boldsymbol{\alpha}$, the steepest descent method converges to the solution under the condition $\boldsymbol{x}_0 = \boldsymbol{\beta}$.

It should be noted that, in comparison to [27], the proposed algorithm that is based on the steepest descent method has been improved. When we consider equation of $\boldsymbol{f}(\boldsymbol{x}) = 0$, the previous algorithm in [27] is $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \mu \boldsymbol{f}(\boldsymbol{x}^{(k)})$. The proposed algorithm is (37), where $\boldsymbol{\alpha} - \boldsymbol{g}(\boldsymbol{\alpha}) = 0$ is satisfied when $\boldsymbol{x} = \boldsymbol{\alpha}$ is a solution of $\boldsymbol{f}(\boldsymbol{x}) = 0$. Here, the nonlinear equation part $\boldsymbol{f}(\boldsymbol{x})$ has been changed into $\boldsymbol{x} - \boldsymbol{g}(\boldsymbol{x})$. As a result, the region of convergence improves. In fact, although the direct application of the existing algorithm resulted in a failure, it was observed that the proposed technique has converged to the solution set in the numerical example.

Remark 1: The considered problem is usually came to optimization problem with bilinear matrix inequalities (BMI) constraints or rank-constrained linear matrix inequality (LMI) conditions. It is well known that these problems are NP-hard as non-convex optimization problem and it is hard to solve.

5.2. Numerical result

In order to demonstrate the efficiency of the algorithms, the following practical example based on the lateral axis dynamics of the L-1011 aircraft [26] is discussed. The system matrices are given as follows.

$$A = \begin{bmatrix} -2.98 & 0.93 & 0 & -0.034 \\ -0.99 & -0.21 & 0.035 & -0.0011 \\ 0 & 0 & 0 & 1 \\ 0.39 & -5.555 & 0 & -1.89 \end{bmatrix},$$

$$B = \begin{bmatrix} -0.032 \\ 0 \\ 0 \\ -1.6 \end{bmatrix}, \quad C = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad D = \begin{bmatrix} 1.5 \\ 0 \\ 0 \\ 0 \end{bmatrix},$$

$$E_1 = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix}, \quad E_2 = 0, \quad Q = I_4, \quad R = 1.$$

The initial conditions are given below.

$$\begin{aligned} &\varepsilon^{(0)} = 0.01, \; K^{(0)} = \begin{bmatrix} 1 & 1 \end{bmatrix}, \\ &P^{(0)}(A + BK^{(0)}C) + (A + BK^{(0)}C)^T P^{(0)} \\ &\quad + \varepsilon^{(0)} P^{(0)} DD^T P^{(0)} + \{\varepsilon^{(0)}\}^{-1} E_K^{(0)T} E_K^{(0)} \\ &\quad + Q + C^T K^{(0)T} R K^{(0)} C = 0, \\ &G^{(0)}(A + BK^{(0)}C + \varepsilon^{(0)} DD^T P^{(0)})^T \\ &\quad + (A + BK^{(0)}C + \varepsilon^{(0)} DD^T P^{(0)}) G^{(0)} + I_4 = 0, \\ &E_K^{(0)} = E_1 + E_2 K^{(0)} C. \end{aligned}$$

By using the Newton's method (36) or the steepest descent method (37), the static output feedback gain K and other variables P, G, ε are solved as

$$K = \begin{bmatrix} 1.5004 & 1.6336 \end{bmatrix},$$

$$P = \begin{bmatrix} 7.9952 & -2.1392 \times 10 & -4.4583 \times 10^{-1} & -1.0360 \times 10^{-1} \\ -2.1392 \times 10 & 6.5126 \times 10 & 4.7683 \times 10^{-1} & -1.7530 \times 10^{-1} \\ -4.4583 \times 10^{-1} & 4.7683 \times 10^{-1} & 1.9967 & 6.9586 \times 10^{-1} \\ -1.0360 \times 10^{-1} & -1.7530 \times 10^{-1} & 6.9586 \times 10^{-1} & 5.6390 \times 10^{-1} \end{bmatrix},$$

$$G = \begin{bmatrix} 3.1262 \times 10^{-1} & -7.0744 \times 10^{-1} & 1.2169 & 2.3559 \times 10^{-1} \\ -7.0744 \times 10^{-1} & 4.3857 & -8.0194 & -1.1964 \\ 1.2169 & -8.0194 & 2.0472 \times 10 & -5.0000 \times 10^{-1} \\ 2.3559 \times 10^{-1} & -1.1964 & -5.0000 \times 10^{-1} & 1.8737 \end{bmatrix},$$

$$\varepsilon = 3.0425 \times 10^{-2},$$

$$\min_{P,K,\varepsilon} \mathbf{Tr}[P] = 7.5682 \times 10.$$

It should be noted that Newton's method (36) converges to the exact solution with an accuracy of 10^{-11} after nine iterations. The error per iterations are given in Table 5.

Table 5. Error per iterations.

k	Errors
0	1.1705×10^4
1	6.0932×10^3
2	2.6824×10^3
3	1.0178×10^3
4	4.7000×10^2
5	4.8751×10^2
6	1.7830×10
7	8.7176×10^{-1}
8	1.8507×10^{-5}
9	1.8945×10^{-12}

On the other hand, under the same convergence accuracy, the steepest descent method (37) converge after 98 iterations for $\mu = 0.9$. It should be noted that although the required number of iterations is different, two numerical algorithms yield the same solutions.

It can be observed that Newton's method involves the least number of iterations. However, taking into account the difficulty in the derivation of Newton's method, the steepest descent method is more suitable for implementation, even though the convergence is slow.

On the other hand, the existing algorithm in [26] yields the following results:

$$K = \bar{K} = \begin{bmatrix} 1.5004 & 1.63431 \end{bmatrix}, \; \varepsilon = \bar{\varepsilon} = 0.0305,$$
$$\min_{P,K,\varepsilon} \mathbf{Tr}[\bar{P}] = 75.68.$$

Therefore, it can be easily observed that the exact cost bound obtained by means of Newton's or steepest descent method is very close to the cost bound in [26]. However, since these results are very close and the technique does not require complex calculations unlike Newton's method, it is suitable to use the steepest descent method owing to easy derivation.

6. CONCLUSION

In this paper, the applications of basic theorems have been proposed to be taught in the numerical analysis in the control engineering course. After establishing the structure of the ARE that is related to the H_∞ control problem via the implicit function theorem, an algorithm for solving the ARE for weakly coupled large-scale systems has been proposed. Since the proposed iterative algorithm is derived by Newton's method, it is noteworthy that the quadratic convergence and uniqueness of the obtained solution are guaranteed. Moreover, in order to simplify the algebraic manipulation, the fixed point iterations have been combined. As a result, it has been shown that linear convergence is attained. A numerical example has shown excellent results. Furthermore, the practice exercise based

on the optimization problem is discussed. After establishing the optimality conditions via the KKT conditions, several numerical algorithms are applied. Finally, the potential and usefulness of these informative proposals for the numerical analysis in the control engineering course are demonstrated.

The three theorems constitute an important long-standing educational area. However, these methods have not yet been discussed with regard to the coupled large systems that have been studied in [28, 29]. It is expected that these applications can be addressed in future studies.

REFERENCES

[1] C. T. Kelley, *Iterative Methods for Linear and Nonlinear Equations*, SIAM, Philadelphia, 1995

[2] Z. Gajic and X. Shen, *Parallel Algorithms for Optimal Control of Large Scale Linear Systems*, Springer Verlag, London, 1993.

[3] Z. Gajic, M. Lim, D. Skataric, W. Su, and V. Kecman, *Optimal Control:Weakly Coupled Systems and Applications*, CRC Press Taylor & Francis Group, Florida, 2009,

[4] W. F. Arnold and A. J. Laub, "Generalized eigenproblem algorithms and software for algebraic Riccati equations," *Proc. IEEE*, vol. 72, no. 12, pp. 1746-1754, 1984.

[5] E. Kostina and O. Kostyukova, "Generalized implicit function theorem and its application to parametric optimal control problems," *J. Mathematical Analysis and Applications*, vol. 320, no. 2, pp. 736-755, 2006.

[6] M. -T. Lim and Z. Gajic, "Subsystem-level optimal control of weakly coupled linear stochastic systems composed of N subsystems," *Optimal Control Applications and Method*, vol. 20, no. 2, pp. 93-112, 1999.

[7] K. Kodra and Z. Gajic, "Optimal control for a new class of singularly perturbed linear systems," *Automatica*, vol. 81, pp. 203-208, 2017.

[8] M. Canale and S. Casale-Brunet, "A multidisciplinary approach for model predictive control education: a Lego Mindstorms NXT-based framework," *Int. J. Control, Automation and Systems*, vol. 12, no. 5, pp. 1030-1039, 2014.

[9] F. B. Chedid and T. Mogi, "A simple iterative algorithm for the towers of Hanoi problem," *IEEE Trans. Education*, vol. 39, no. 2, pp. 274-275, May 1996.

[10] I. Stojmenovic, "Recursive algorithms in computer science courses: Fibonacci numbers and binomial coefficients," *IEEE Trans. Education*, vol. 43, no. 3, pp. 273-276, Aug. 2000.

[11] J. R. Magnus and H. Neudecker, *Matrix Differential Calculus With Applications in Statistics and Econometrics*, John Wiley and Sons, New York, 1999.

[12] Z. Gajić, D. Petkovski, and X. Shen, *Singularly Perturbed and Weakly Coupled Linear System-a Recursive Approach*, Lecture Notes in Control and Information Sciences, vol.140. Springer-Verlag, Berlin, 1990.

[13] H. Mukaidani, "Optimal numerical strategy for Nash games of weakly coupled large-scale systems," *Dyn. Continuous, Discrete and Impulsive Systems, Series B: Applications and Algorithms*, vol. 13, no. 2, pp. 249-268, 2006.

[14] H. Mukaidani, "Numerical computation of sign indefinite linear quadratic differential games for weakly coupled large-scale systems," *Int. J. Control*, vol. 80, no. 1, pp. 75-86, 2007.

[15] T. Yamamoto, "A method for finding sharp error bounds for Newton's method under the Kantorovich assumptions," *Numerische Mathematik*, vol. 49, no. 2-3, pp. 203-220, 1986.

[16] J. M. Ortega, *Numerical Analysis, A Second Course*, SIAM, Philadelphia, 1990.

[17] H. Mukaidani, "A numerical analysis of the Nash strategy for weakly coupled large-scale systems," *IEEE Trans. Automatic Control*, vol. 51, no. 8, pp. 1371-1377, Aug. 2006.

[18] A. Granas and J. Dugundji, *Fixed Point Theory*, Springer-Verlag, New York 2003.

[19] J. C. Doyle, K. Glover, P. P. Khargonekar, and B. A. Francis, "State space solution to standard H_2, and H_∞ control problems," *IEEE Trans. Automatic Control*, vol. 34, no. 8, pp. 831-847, Aug. 1989.

[20] S. Boyd and L. L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, 2004.

[21] A. J. Laub, "A Schur method for solving algebraic Riccati equations," *IEEE Trans. Automatic Control*, vol. 24, no. 6, pp. 913-921, Dec. 1979.

[22] P. Benner and R. Byers, "An exact line search method for solving generalized continuous-time algebraic Riccati equations," *IEEE Trans. Automatic Control*, vol. 43, no. 1, pp. 101-107, Jan. 1998.

[23] D. L. Kleinman, "On an iterative technique for Riccati equation computations," *IEEE Trans. Automatic Control*, vol. 13, no. 1, pp. 114-115, Jan. 1968.

[24] S. Kodama and N. Suda, *Matrix Theory for System Control*, Corona-Publishing, Tokyo, 1978 (in Japanese).

[25] J. D. Delacour, M. Darwish, and J. Fantin, "Control strategies for large-scale power systems," *Int. J. Control*, vol. 27, no. 5, pp. 753-767, 1978.

[26] S. O. R. Moheimani and I. R. Petersen, "Optimal guaranteed cost control of uncertain systems via static and dynamic output feedback," *Automatica*, vol. 32, no. 4, pp. 575-579, 1996.

[27] H. Mukaidani and H. Xu, "Infinite horizon linear-quadratic Stackelberg games for discrete-time stochastic systems," *Automatica*, vol. 76, pp. 301-308, 2017.

[28] Y. Xu, R. Lu, P. Shi, J. Tao, and S. Xie, "Robust estimation for neural networks with randomly occurring distributed delays and Markovian jump coupling," *IEEE Tran. Neural Networks and Learning Systems*, vol. 29, no. 4, pp. 845-855, 2017.

[29] Y. Xu, R. Lu, H. Peng, K. Xie, and A. Xue, "Asynchronous dissipative state estimation for stochastic complex networks with quantized jumping coupling and uncertain measurements," *IEEE Trans. Neural Networks and Learning Systems*, vol. 28, no. 2, pp. 268-277, Feb. 2017.

Hiroaki Mukaidani received his B.S. in Integrated Arts and Sciences from Hiroshima University in 1992 and his M.Eng. and Dr.Eng. degrees in Information Engineering form Hiroshima University in 1994 and 1997, respectively. In April 1998, he joined the Faculty of Information Science at Hiroshima City University as a Research Associate. Since April 2002, he has been with the Graduate School of Education at Hiroshima University as an Assistant Professor and Associate Professor. He became a Full Professor at the Institute of Engineering at Hiroshima University in April 2012. He spent 10 months (from November 2007 to September 2008) as a research fellow at the Department of Electrical and Computer Engineering (ECE), University of Waterloo. His research interests include robust control, dynamic games, and the application of stochastic systems. Dr. Mukaidani is a member of the Institute of Electrical and Electronic Engineers (IEEE).

Tadashi Shima received his M.Sci. and Dr.Sci. degrees from Osaka University in 1989 and 1992, respectively. In April 1990, he joined the Graduate School of Engineering Science of Osaka University as an Assistant Professor. He moved to the Hiroshima University in October 1994. From 1994 to 2006, he served as an Associate Professor in the Faculty of Integrated Arts and Sciences and he has been with the Department of Information Engineering as an Associate Professor since 2006. His research interests include stochastic processes, spectral analysis. He is a member of the Mathematical Society of Japan.