F-2d-06-4

ON THE GENERALIZED EIGENSPACE APPROACH FOR SOLVING RICCATI EQUATIONS *

Li Qiu *

* *Department of Electrical & Electronic Engineering*
Hong Kong University of Science & Technology
Clear Water Bay, Kowloon, Hong Kong
Tel: 852-2358-7067, Fax: 852-2358-1485
Email: eeqiu@ee.ust.hk

Abstract: In this paper, we show that a general discrete time Riccati equation can be solved by finding a deflating subspace of a $2n \times 2n$ matrix pencil. In many applications, the data in the Riccati equation is formed from the squares of the raw physical data. In this case, we show that continuous-time and discrete-time Riccati equations can be solved without squaring the raw data by finding deflating subspaces of larger matrix pencils. Examples show that the solutions without squaring the data have numerical advantages over the existing solutions of Riccati equations using the generalized eigenspace approach. *Copyright © 1999 IFAC*

Keywords: Riccati equations, numerical methods, optimal control, matrix equations

1. INTRODUCTION

For a matrix pair $(A, B) \in \mathbb{C}^{n\times n} \times \mathbb{C}^{n\times m}$ and a Hermitian matrix

$$\begin{bmatrix} Q & S \\ S^* & R \end{bmatrix} \in \mathbb{C}^{(n+m)\times(n+m)},$$

the associated continuous time algebraic Riccati equation (CARE) is of the form (assuming R to be nonsingular)

$$A^*X + XA + Q - (XB + S)R^{-1}(B^*X + S^*) = 0 \quad (1)$$

and the associated discrete time algebraic Riccati equation (DARE) is of the form (assuming $\mathrm{Ker}R \cap \mathrm{Ker}B = \{0\}$)

$$A^*XA - X + Q - (A^*XB + S)(B^*XB + R)^{-1} \times (B^*XA + S^*) = 0. \quad (2)$$

A Hermitian solution X of CARE (1) is said to be stabilizing if all eigenvalues of $A + BF$, where $F = -R^{-1}(B^*X + S^*)$ is the corresponding state feedback gain, have negative real parts. A Hermitian solution X of DARE (2) is said to be stabilizing if all eigenvalues of $A + BF$, where $F = -(B^*XB + R)^{-1}(B^*XA + S^*)$ is the corresponding state feedback gain, are inside the unit circle. The stabilizing solution, if exists, is unique.

The solution of ARE has been extensively studied. Two recent collections of works (Bittanti *et al.*, 1991; Patel *et al.*, 1994) contain good survey papers on this topic. There are several ways to find the stabilizing solution of an ARE. Among them the so-called Schur method has gained popularity in recent years.

The Schur method for CARE (1) is based on the fact that the graph of the unique stabilizing solution of (1), if exists, is equal to the invariant subspace of the $2n \times 2n$ Hamiltonian matrix

$$\begin{bmatrix} A - BR^{-1}S^* & -BR^{-1}B^* \\ -Q + SR^{-1}S^* & -A^* + SR^{-1}B^* \end{bmatrix} \quad (3)$$

corresponding to the eigenvalues with negative real parts. This fact, apart from its theoretic interest, has been the basis of the general purpose

* This research is supported by the Hong Kong Research Grants Council and Sino Software Research Institute.

 ISBN: 0 08 043248 4

Riccati solvers in, such as, MATLAB Control System Toolbox.

The Schur method for DARE (2) leads to generalized eigenproblems. If R is nonsingular, the graph of the unique stabilizing solution of (2), if exists, is equal to the deflating subspace of $2n \times 2n$ symplectic matrix pencil

$$\left(\begin{bmatrix} A - BR^{-1}S^* & 0 \\ -Q + SR^{-1}S^* & I \end{bmatrix}, \begin{bmatrix} I & BR^{-1}B^* \\ 0 & A^* - SR^{-1}B^* \end{bmatrix} \right) \tag{4}$$

corresponding to the eigenvalues inside the unit circle (Pappas *et al.*, 1980; Emami-Naeini and Franklin, 1980). However, applications where R is singular occur frequently. To cope with this, it is proposed in the literature (Emami-Naeini and Franklin, 1980; Van Dooren, 1981) and implemented in MATLAB Control System Toolbox to find the stabilizing solution using the following $(2n + m) \times (2n + m)$ matrix pencil

$$\left(\begin{bmatrix} A & 0 & B \\ Q & -I & S \\ S^* & 0 & R \end{bmatrix}, \begin{bmatrix} I & 0 & 0 \\ 0 & -A^* & 0 \\ 0 & -B^* & 0 \end{bmatrix} \right). \tag{5}$$

In this case, the graph of the stabilizing solution, if exists, is equal to the space spanned by the first $2n$ coordinates of the deflating subspace of (5) corresponding to the eigenvalues inside the unit circle. This although pragmatically overcomes the difficulty brought about by the singular or near singular R, it leaves some disappointment: the use of $(2n + m) \times (2n + m)$ pencil is theoretically unpleasant (lack of symmetry) and numerically undesirable (presence of bigger matrices). In this paper, we show that it is possible to solve (2) using a $2n \times 2n$ matrix pencil.

For a given Hermitian matrix

$$\begin{bmatrix} Q & S \\ S^* & R \end{bmatrix},$$

it is always possible to carry out factorization

$$\begin{bmatrix} Q & S \\ S^* & R \end{bmatrix} = \begin{bmatrix} C^* \\ D^* \end{bmatrix} J \begin{bmatrix} C & D \end{bmatrix} \tag{6}$$

where J is a signature matrix such as

$$J = \begin{bmatrix} I & 0 \\ 0 & -I \end{bmatrix}$$

or some other nonsingular Hermitian matrix. In many applications, such as $\mathcal{H}_2$ and $\mathcal{H}_\infty$ control and various factorizations of rational matrices, the raw data for a Riccati equation is given in terms of (A, B, C, D, J) instead of (A, B, Q, R, S). In this case, CARE (1) becomes

$$A^*X + XA + C^*JC - (XB + C^*JD) \times (D^*JD)^{-1}(B^*X + D^*JC) = 0, \tag{7}$$

and DARE (2) becomes

$$A^*XA - X + C^*JC - (A^*XB + C^*JD) \times (B^*XB + D^*JD)^{-1}(B^*XA + D^*JC) = 0. \tag{8}$$

Although matrices of the form C^*JC and DJD^* appear in the equation, it is desirable to avoid this kind of squares in the numerical solution due to finite wordlength reasons. As soon as the matrix squares are formed, the number of accurate digits is lost by half. No matter what method in solving AREs (1) and (2) is used to solve AREs (7) and (8), it is impossible to recover such loss of accuracy. The readers may wish to examine Examples 1-2 in Section 5 for better appreciation of this point. In this paper, we will show that AREs (7) and (8) can be solved via $(2n + m + p) \times (2n + m + p)$ matrix pencils without forming the matrix squares explicitly. Examples show that the proposed new method does behave better numerically.

An immediate criticism of this proposed method is the involvement of matrices of large dimensions. This is indeed a new problem created in addressing an old problem. On one hand we would like to argue that the computer wordlength is a more precious resource than the computing time. On the other hand, we will show that the $(2n + m + p) \times (2n + m + p)$ matrix pencil is always sparse and structured, so iteration method can be used to find the required eigenspace.

2. SOLUTION OF THE DISCRETE TIME ARE VIA $2N \times 2N$ PENCIL

Since $\mathrm{Ker}B \cap \mathrm{Ker}R = \{0\}$, there exists symmetric matrix X_0 such that $B^*X_0B + R$ is nonsingular. Let

$$\begin{bmatrix} Q_0 & S_0 \\ S_0^* & R_0 \end{bmatrix} = \begin{bmatrix} A^*X_0A - X_0 & A^*X_0B \\ B^*X_0A & B^*X_0B \end{bmatrix} + \begin{bmatrix} Q & S \\ S^* & R \end{bmatrix}$$

and let $\tilde{X}$ be the stabilizing solution of the following Riccati equation

$$A^*XA - X + Q_0 - (A^*XB + S_0)(B^*XB + R_0)^{-1} \times (B^*XA + S_0^*) = 0.$$

Then it is easy to see that $X = \tilde{X} + X_0$ is a stabilizing solution of (2). Hence the following theorem follows immediately.

Theorem 1. Let X_0 be any Hermitian matrix such that $B^*X_0B + R$ is nonsingular and

$$\begin{bmatrix} Q_0 & S_0 \\ S_0^* & R_0 \end{bmatrix} = \begin{bmatrix} A^*X_0A - X_0 & A^*X_0B \\ B^*X_0A & B^*X_0B \end{bmatrix} + \begin{bmatrix} Q & S \\ S^* & R \end{bmatrix}.$$

Let $\begin{bmatrix} V_1 \\ V_2 \end{bmatrix}$ span the deflating subspace of

$$\left(\begin{bmatrix} A - BR_0^{-1}S_0^* & 0 \\ -Q + S_0R_0^{-1}S_0^* & I \end{bmatrix}, \begin{bmatrix} I & BR_0^{-1}B^* \\ 0 & A^* - S_0R_0^{-1}B^* \end{bmatrix} \right) \tag{9}$$

corresponding to the stable eigenvalues. Then (2) has a stabilizing solution if and only if V_1 is invertible. In this case, the stabilizing solution is given by $X = X_0 + V_2V_1^{-1}$.

 ISBN: 0 08 043248 4

One may prefer a matrix pencil which directively leads to X without the extra addition $\tilde{X} + X_0$. This is possible.

Theorem 2. Let X_0 be any Hermitian matrix such that $R_0 := B'X_0B + R$ is nonsingular and $\begin{bmatrix} U_1 \\ U_2 \end{bmatrix}$ span the deflating subspace of

$$\left(\begin{bmatrix} A - BR_0^{-1}S_0^* & 0 \\ -Q + SR_0^{-1}S_0^* & I \end{bmatrix}, \begin{bmatrix} I - BR_0^{-1}B^* & BR_0^{-1}B^* \\ SR_0^{-1}B^*X_0 & A^* - SR_0^{-1}B^* \end{bmatrix} \right) \quad (10)$$

corresponding to the stable eigenvalues. Then (2) has a stabilizing solution if and only if U_1 is invertible. In this case, the stabilizing solution is given by $X = U_2U_1^{-1}$.

3. SOLUTION WITHOUT SQUARE-UP

When (6) holds, CARE (1) becomes CARE (7). It is desirable to solve CARE (7) without forming Q, S, R.

Consider matrix pencil

$$\left(\begin{bmatrix} A & B & 0 & 0 \\ C & D & 0 & -J^{-1} \\ 0 & 0 & -A^* & -C^* \\ 0 & 0 & -B^* & -D^* \end{bmatrix}, \begin{bmatrix} I & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \right). \quad (11)$$

Proposition 1. The matrix pencil (11) has $p + m$ eigenvalues at infinity and the rest of the $2n$ eigenvalues are finite and symmetric to the imaginary axis.

Let $\mathcal{V}$ be the deflating subspace of (11) corresponding to the finite eigenvalues with negative real parts and let a basis of $\mathcal{V}$ be given by the columns of $\begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix}$, partitioned consistently with that of (11).

Theorem 3. CARE (7) has a stabilizing solution if and only if V_1 is invertible. In this case, the unique stabilizing solution of (7) is given by $X = V_3V_1^{-1}$ and the corresponding feedback gain is given by $F = V_2V_1^{-1}$.

The DARE (8) can also be solved without squares. Consider matrix pencil

$$\left(\begin{bmatrix} A & B & 0 & 0 \\ C & D & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} I & 0 & 0 & 0 \\ 0 & 0 & 0 & J^{-1} \\ 0 & 0 & A^* & C^* \\ 0 & 0 & B^* & D^* \end{bmatrix} \right). \quad (12)$$

Proposition 2. The matrix pencil (12) has at least p eigenvalues at 0, at least m at ∞, and the rest $2n$ eigenvalues are symmetric to the unit circle.

Let $\mathcal{U}$ be the deflating subspace of (12) corresponding to the eigenvalues inside the unit circle. Then it is easy to see that span $\begin{bmatrix} 0 \\ 0 \\ 0 \\ I \end{bmatrix} \subset \mathcal{U}$. Let $\mathcal{V}$ be any complement of span $\begin{bmatrix} 0 \\ 0 \\ 0 \\ I \end{bmatrix}$ in $\mathcal{U}$ and let a basis of $\mathcal{V}$ be given by the columns of $\begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix}$.

Theorem 4. The Riccati equation (8) has a stabilizing solution if and only if V_1 is invertible. In this case, the unique stabilizing solution of (8) is given by $X = V_3V_1^{-1}$ and the corresponding state feedback gain is given by $F = V_2V_1^{-1}$.

4. THE POWER METHOD

An immediate criticism of the proposed use of $(2n+m+p) \times (2n+m+p)$ pencils in solving ARE is the involvement of matrices of large dimensions. This is indeed a new problem created in addressing an old problem. On one hand we would like to argue that the computer wordlength is a more precious resource than the computing time. On the other hand, we see that the $(2n + m + p) \times (2n + m + p)$ matrix pencil is always sparse and structured. It is hopeful that the sparseness and the structure can be utilized to reduce the computational complexity. In this section, we present a crude method based on power iteration to find the required eigenspaces of the matrix pencils involved.

Let us first consider the problem of finding the eigenspace corresponding to the m dominant eigenvalues of a general regular $n \times n$ pair (M, N). First let us assume that N is nonsingular. In this case, the eigenvalues and corresponding eigenspaces are identical to those of $N^{-1}M$. The following algorithm, called block or subspace power iteration, is well known for the eigenspace of the dominant eigenvalues (Saad, 1992).

Algorithm 1.

Start. Choose an initial m dimensional subspace $\mathcal{V}_0$.

Iteration. Until convergence, compute $\mathcal{V}_{i+1} = N^{-1}M\mathcal{V}_i$.

For generic initial subspace, this algorithm will converge to the required eigenspace if $|\lambda_m(M, N)| > |\lambda_{m+1}(M, N)|$.

Now let us consider the matrix pencil (11). The eigenspace that we are interested in is the one

 ISBN: 0 08 043248 4

corresponding to the eigenvalues on the left hand side of the complex plane. We need to transform the problem so that these eigenvalues become dominant ones. Observe that the map $(M_0, N_0) \to (M_0+\alpha N_0, M_0-\alpha N_0)$ where $\alpha > 0$ maps all eigenvalues in the left hand side of the complex plane to the outside of the unit circle. This technique is called shifting. After shifting, matrix pencil (11) becomes

$$(M,N) = \left(\begin{bmatrix} A-\alpha I & B & 0 & 0 \\ C & D & 0 & -J^{-1} \\ 0 & 0 & -A^*-\alpha I & -C^* \\ 0 & 0 & -B^* & -D^* \end{bmatrix}, \begin{bmatrix} A+\alpha I & B & 0 & 0 \\ C & D & 0 & -J^{-1} \\ 0 & 0 & -A^*+\alpha I & -C^* \\ 0 & 0 & -B^* & -D^* \end{bmatrix} \right).$$

Since the pair (M, N) is still sparse and structured, both $M\mathcal{V}$ and $N^{-1}\mathcal{V}$ can be computed efficiently without forming (M, N) explicitly.

For the pencil (12), we are interested in the eigenspace corresponding to the eigenvalues inside the unit circle. Apparently, shifting can also be used to transform these eigenvalues to the dominant ones. The map $(M_0, N_0) \to (N_0-\alpha M_0, M_0-\alpha N_0)$ for $-1 < \alpha < 1$ does the job. In this case, it amounts to consider the following matrix pair.

$$(M,N) = \left(\begin{bmatrix} I-\alpha A & -\alpha B & 0 & 0 \\ -\alpha C & -\alpha D & 0 & J^{-1} \\ 0 & 0 & A^*-\alpha I & C^* \\ 0 & 0 & B^* & D^* \end{bmatrix}, \begin{bmatrix} A-\alpha I & B & 0 & 0 \\ C & D & 0 & -\alpha J^{-1} \\ 0 & 0 & I-\alpha A^* & -\alpha C^* \\ 0 & 0 & -\alpha B^* & -\alpha D^* \end{bmatrix} \right). \quad (13)$$

The sparseness of the matrix pair (13) is maximized if $\alpha = 0$ is chosen, but it results in a singular N. Experiment shows that Algorithm 1 still works if $N^{-1}\mathcal{V}$ is interpreted as the preimage of $\mathcal{V}$ under N. This leads the following conjecture.

Conjecture 1. Let Algorithm 1 be applied to a regular matrix pencil (M, N). If $|\lambda_m(M,N)| > |\lambda_{m+1}(M,N)|$, then for generic $\mathcal{V}_0$, $\mathcal{V}_i$ converges to $\mathcal{V}$.

The effort in proving this conjecture is underway.

The convergence of the power iteration depends on $\frac{|\lambda_{m+1}(M,N)|}{|\lambda_m(M,N)|}$. Examples show that the convergence can be very slow for simple problems. It is an interesting problem how to take advantage of the sparseness and structure of the matrix pencils to get a uniformly better algorithm than the unstructured QZ algorithm which is the most commonly used one to find the required eigenspaces.

5. NUMERICAL EXAMPLES

Two MATLAB .m files, caresys.m and daresys.m, are written which implement the $(2n + m + p) \times (2n + m + p)$ pencil methods in Section 3 for CARE (7) and DARE (8) respectively. Both caresys.m and daresys.m take as input variables (A, B, C, D, J) and *option* which determines if the eigenspace is computed by the QZ method or the power method. These programs are compared with care.m and dare.m in the MATLAB Control Systems Toolbox. The computation is done in a SPARCstation 10 using MATLAB version 5.0.

Due to time limitation, we are only able to finish the numerical experiment for the discrete-time case. The numerical experiment for continuous-time case is underway and will be added to the paper in the next stage.

Traditional measure of the quality of an approximate solution $\hat{X}$ is $\|R(\hat{X})\|$ where $R(X)$ is the left hand side of the ARE. When the exact solution is unknown, this is probably is the only choice. However, how to compute $R(\hat{X})$ becomes a new nontrivial numerical problem we choose not to address in this paper. The computation of $R(\hat{X})$ is directly related to the computation of

$$F = -(D^*JD)^{-1}(B^*X + D^*JC)$$

in the continuous time case and

$$F = -(B^*XB + D^*JD)^{-1}(B^*XA + D^*JC)$$

in the discrete time case since

$$R(\hat{X}) = A^*X + XA + C^*JC + (XB + C^*JD)F$$

and

$$R(\hat{X}) = A^*XA - X + C^*JC + (A^*XB + C^*JD)F$$

respectively. In applications, it is F, not X, that is needed. In the following examples, the exact solutions are all known. Hence we use $\|\hat{X} - X\|$ as the measure of the quality of solutions.

Example 2. Suppose

$$\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] = \left[\begin{array}{cc|cc} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1+\epsilon \\ \hline 1 & 1 & 0 & 0 \\ 1 & 1+\epsilon & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right]$$

and $J = I$. It is easy to verify that for nonzero $\epsilon > 0$ the unique stabilizing solution to CARE (7) is $X = I$. However, since

$$Q = C^*C = \begin{bmatrix} 2 & 2+\epsilon \\ 2+\epsilon & 2+2\epsilon+\epsilon^2 \end{bmatrix}$$

is deemed rank one if ϵ^2 is less than the machine epsilon, the matrix

$$\begin{bmatrix} A - BR^{-1}S^* & -BR^{-1}B^* \\ -Q + SR^{-1}S^* & -A^* + SR^{-1}B^* \end{bmatrix}$$

 ISBN: 0 08 043248 4

is deemed to have eigenvalues on the imaginary axis. In fact, care.m in the MATLAB Control System Toolbox fails when $\epsilon \leq 10^{-8}$.

Example 3. Suppose

$$\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] = \left[\begin{array}{cc|cc} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ \hline 1 & 1 & 0 & 0 \\ 1 & 1+\epsilon & 0 & 0 \end{array}\right]$$

and $J = I$. It is easy to verify that for nonzero $\epsilon > 0$ the unique stabilizing solution to DARE (8) is

$$X = C^*C = \begin{bmatrix} 2 & 2+\epsilon \\ 2+\epsilon & 2+2\epsilon+\epsilon^2 \end{bmatrix}.$$

However, since

$$Q = C^*C = \begin{bmatrix} 2 & 2+\epsilon \\ 2+\epsilon & 2+2\epsilon+\epsilon^2 \end{bmatrix}$$

is deemed rank one if ϵ^2 is less than the machine epsilon, the matrix pencil

$$\left(\begin{bmatrix} A & 0 & B \\ Q & -I & S \\ S^* & 0 & R \end{bmatrix}, \begin{bmatrix} I & 0 & 0 \\ 0 & -A^* & 0 \\ 0 & -B^* & 0 \end{bmatrix} \right)$$

is deemed to have generalized eigenvalues on the imaginary axis. Indeed, dare.m in the MATLAB Control System Toolbox fails for this example when $\epsilon \leq 10^{-8}$.

However, daresys.m works well for $\epsilon \geq 10^{-15}$, especially when the power method option is used. The results of daresys.m for different ϵ is summerized in Table 1. It is seen that the power method works much better than the QZ method in this example. The reason is probably because the error of MATLAB's internal QZ program, which can not be controlled by an user, is too big. The power method also converges very fast. This is due to the fact that the generalized eigenvalues of the matrix pencil involved has eigenvalues at infinity and zero only. A final observation is that F is much more sensitive than X to computation errors. The reason for this is not clear.

The above examples are not pathological cases. In the so-called low-gain and high-gain design of feedback control, the extreme values of the system data occur in quite a natural way.

Example 4. A program daresym.m, which implements the $2n \times 2n$ pencil method for DARE, is written. Programs daresym.m, daresys.m, and dare.m are tested using the benchmark examples in (Benner *et al.*, 1996). For most of the examples (examples 1-11), they all work almost equally well. Example 12 has the following data:

$$A = \begin{bmatrix} 0 & \alpha \\ 0 & 0 \end{bmatrix}; B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}; C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}; D = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

The exact solution is given by

$$X = \begin{bmatrix} 1 & 0 \\ 0 & 1+\alpha^2 \end{bmatrix}.$$

Table 2 gives the computation results for several different values of α. The relative residue means

$$\|R(\hat{X})\|_F / \|\hat{X}\|_F.$$

The data in Example 13 is given in the following way:

$$A_0 = \mathrm{diag}(0,1,3), V = I - \frac{2}{3}vv', v' = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}.$$

$$A = VA_0V, B = I, C = \begin{bmatrix} \sqrt{\alpha}I \\ 0 \end{bmatrix}, D = \begin{bmatrix} 0 \\ \sqrt{\alpha}I \end{bmatrix}.$$

The exact solution is

$$X = V\mathrm{diag}\left(\alpha, \alpha\frac{1+\sqrt{5}}{2}, \alpha\frac{9+\sqrt{85}}{2}\right)V.$$

Table 3 shows the computation results. When $\alpha = 10^{14}$, both dare and daresym give the warning message "May not find symmetric solution; *spectrum too near unit circle.*", but daresys works perfectly. The relative error means

$$\|\hat{X} - X\|_F / \|X\|_F.$$

The relative residue means

$$\|R(\hat{X})\|_F / \|\hat{X}\|_F.$$

6. CONCLUSION

In this paper, we first proved that a general regular DARE can be solved by using a $2n \times 2n$ matrix pencil. This result is presented mainly to give a parallel development to the CARE theory. It also indicates that there is not much reason to use the $(2n+m) \times (2n+m)$ matrix pencil (5) in solving the DARE (2).

The main point of this paper is that square of data should be avoided in numerical computation. We have shown that in the case when the raw data for the Riccati equation is given in terms of (A, B, C, D, J) instead of (A, B, Q, R, S), it is possible to compute the solution of Riccati equations without squaring the data even though the squares appear in the equation. The study carried out here is preliminary. It is of interest to see if there are other methods without square-up with better numerical properties and how we can improve the numerical properties of the methods developed in this paper. Further study in the following directions is underway:

(1) Carry out further numerical experiment to compare the ARE solvers based on the matrix pencils (11) and (12) and those based on the traditional matrix pencils with data square-up.

 ISBN: 0 08 043248 4

		daresys (QZ)	daresys (power)
$\epsilon = 10^{-8}$	$\|\hat{X} - X\|_F$	1.95×10^{-15}	1.54×10^{-15}
	$\|\hat{F} - F\|_F$	5.44×10^{-16}	9.61×10^{-16}
$\epsilon = 10^{-12}$	$\|\hat{X} - X\|_F$	2.58×10^{-9}	8.82×10^{-16}
	$\|\hat{F} - F\|_F$	1.32×10^{3}	1.05×10^{-15}
$\epsilon = 10^{-14}$	$\|\hat{X} - X\|_F$	2.78×10^{-3}	1.78×10^{-15}
	$\|\hat{F} - F\|_F$	1.20×10^{13}	1.48×10^{-15}
$\epsilon = 10^{-15}$	$\|\hat{X} - X\|_F$	6.86×10^{-2}	9.93×10^{-16}
	$\|\hat{F} - F\|_F$	4.32×10^{14}	1.04×10^{-15}

Table 1. For Example 2.

		dare	daresym	daresys
$\alpha = 10^{6}$	$\hat{X} - X$	$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 \\ 0 & 2.2122 \times 10^{7} \end{bmatrix}$	$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$
	Relative residue	4.8828×10^{-16}	2.2122×10^{-5}	1.2207×10^{-16}
$\alpha = 10^{10}$	$\hat{X} - X$	$\begin{bmatrix} 0 & 0 \\ 0 & 8.2412 \times 10^{6} \end{bmatrix}$	Completely wrong	$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$
	Relative residue	8.2416×10^{-14}	Completely wrong	1.0000×10^{-20}

Table 2. Example 12 in [3]

		dare	daresym	daresys
$\alpha = 10^{5}$	Relative error	2.5238×10^{-11}	5.2532×10^{-15}	1.3222×10^{-15}
	relative residue	4.8828×10^{-16}	4.7712×10^{-15}	1.1008×10^{-15}
$\alpha = 10^{10}$	Relative error	1.4289×10^{-7}	4.2733×10^{-7}	1.0192×10^{-14}
	Relative residue	1.3024×10^{-7}	3.8965×10^{-7}	9.8596×10^{-16}
$\alpha = 10^{14}$	Relative error	4.1923×10^{-4}	7.2512×10^{-4}	2.3031×10^{-15}
	Relative residue	3.7490×10^{-4}	6.5812×10^{-4}	2.6729×10^{-15}

Table 3. Example 13 in [3]

(2) What can be said about the structure of the matrix pencils (11) and (12)? How can we use the structure and sparseness to simplify computation and to increase accuracy? Is there a structure preserving QZ decomposition which can be used to find the eigenspace needed? For structure preserving Schur decomposition with application in solving AREs, see (Paige and Van Loan, 1981; Byers, 1986; Bunse-Gerstner and Faßbender, 1997).

(3) *Extend our results to possibly accommodate* descriptor systems (with an E matrix in the appropriate places in the AREs) or singular AREs (singular R in continuous-time case and $Ker B \cap Ker R \neq \{0\}$ *in discrete-time case*). See (Arnold, III and Laub, 1984; Ionescu *et al.*, 1997) for motivation and existing literature.

7. REFERENCES

Arnold, III, M. F. and A. J. Laub (1984). Generalized eigenproblem algorithms and software for algebraic Riccati equations. *Proceedings of IEEE* **72**, 1746–1754.

Benner, P., A. J. Laub and V. Mehrmann (1996). A collection of benchmark examples for the numerical solution of algbraic Riccati equation II: discrete-time case. preprint.

Bittanti, S., Laub, A. J. and Willems, J. C., Eds.) (1991). *The Riccati Equation*. Springer-Verlag, Berlin.

Bunse-Gerstner, A. and H. Faßbender (1997). A Jacobi-like method for solving algebraic Riccati equations on parallel computers. *IEEE Trans. Automat. Contr.* **42**, 1071–1084.

Byers, R. (1986). A Hamiltonian QR algorithm. *SIAM J. Sci. Stat. Comput.* **7**, 212–229.

Emami-Naeini, A. and G. F. Franklin (1980). Comments on 'On the numerical solution of the discrete-time algebraic Riccati equation'. *IEEE Trans. Automat. Contr.* **25**, 1015–1016.

Ionescu, V., C. Oara and M. Weiss (1997). General matrix pencil techniques for the solution of algebraic Riccati equations: a unified approach. *IEEE Trans. Automat. Contr.* **42**, 1085–1097.

Paige, C. and C. Van Loan (1981). A Schur decomposition for Hamiltonian matrices. *Linear Algebra and its Applications* **41**, 11–32.

Pappas, T., A. J. Laub and N. R. Sandell Jr. (1980). On the numerical solution of discrete-time algebraic Riccati equation. *IEEE Trans. Automat. Contr.* **25**, 631–641.

Patel, R. V., Laub, A. J. and Van Dooren, P. M., Eds.) (1994). *Numerical Linear Algebra Techniques for Systems and Control*. IEEE Press.

Saad, Y. (1992). *Numerical Methods for Large Eigenvalue Problems*. Manchester University Press, Manchester, UK.

Van Dooren, P. (1981). A generalized eigenvalue approach for solving Riccati equations. *SIAM J. Sci. Stat. Comput.* **2**, 121–135.

 ISBN: 0 08 043248 4