CHAPTER 13

NUMERICAL SOLUTIONS AND CONDITIONING OF ALGEBRAIC RICCATI EQUATIONS

Topics covered

- Results on Existence and Uniqueness of Solutions of the CARE and DARE
- Perturbation Analyses and Condition Numbers
- The Schur Methods, Newton's Methods, and the Matrix Sign Function Methods
- Convergence Results for Newton's Methods
- The Generalized Eigenvector and the Generalized Schur Methods
- Inverse-Free Generalized Schur Methods
- The Schur and Inverse-Free Schur Methods for the Descriptor Riccati Equations
- Comparative Study and Recommendations

13.1 INTRODUCTION

This chapter is devoted to the study of numerical solutions of the continuous-time algebraic Riccati equation (CARE):

$$XA + A^{\mathrm{T}}X + Q - XBR^{-1}B^{\mathrm{T}}X = 0 \tag{13.1.1}$$

and of its discrete counterpart (DARE)

$$A^{\mathrm{T}}XA - X + Q - A^{\mathrm{T}}XB(R + B^{\mathrm{T}}XB)^{-1}B^{\mathrm{T}}XA = 0. \tag{13.1.2}$$

Equation (13.1.1) is very often written in the following compact form:

$$XA + A^{\mathrm{T}}X + Q - XSX = 0, \tag{13.1.3}$$

where

$$S = BR^{-1}B^{\mathrm{T}}. \tag{13.1.4}$$

Equation (13.1.2) can also be written in the compact form:

$$A^{\mathrm{T}}X(I + SX)^{-1}A - X + Q = 0, \tag{13.1.5}$$

where S is again as given by (13.1.4).

These equations have long been subject of research in mathematics, physics, and engineering. They play major roles in many design problems in control and filter theory. As we have seen in Chapter 10, historically, AREs started as an important tool in the solution of *Linear Quadratic Optimization problems.* In recent years, they became a subject of intensive study, both from theoretical and computational viewpoints, because of their important roles in **state-space solutions of H_∞ and robust control problems.** For a brief history of the importance, applications, and historical developments of the AREs, see Bittanti *et al.* (1991).

The following computational methods for the CARE and DARE are widely known in the literature and most of them are discussed in **Section 13.5** of this chapter.

1. **The Eigenvector Methods** (McFarlane 1963; Potter 1966).
2. **The Schur Methods and the Structure-Preserving Schur Methods** (Laub 1979; Byers 1983, 1986a, 1990; Bunse-Gerstner and Mehrmann 1986; Mehrmann 1988; Benner *et al.* 1997c).
3. **The Generalized Eigenvector, the Generalized Schur, and Inverse-Free Generalized Methods** (Pappas *et al.* 1980; Van Dooren 1981; Arnold and Laub 1984; Mehrmann 1991).
4. **The Matrix Sign Function Methods** (Roberts 1980 [1971]; Denman and Beavers 1976; Bierman 1984; Gardiner and Laub 1986; Byers 1987; Kenney and Laub 1995).
5. **Newton's Methods** (Kleinman 1968; Hewer 1971; Benner and Byers 1998; Guo and Lancaster 1998; Guo 1998).

The eigenvector methods are well known to have numerical difficulties in case the Hamiltonian matrix associated with the CARE or the symplectic matrix associated with the DARE has some multiple or near-multiple eigenvalues (the corresponding eigenvectors will be ill-conditioned).

In these cases, the Schur methods, based on the real Schur decompositions of the Hamiltonian matrix for the CARE and of the symplectic matrix for the DARE, should be preferred over the eigenvector methods. The Schur method is widely used in practice for the CARE. Unfortunately, it cannot be applied to the DARE when A is singular. Indeed, even if A is theoretically nonsingular, but is computationally close to a singular matrix, the Schur method for the DARE should be avoided. An alternative for the DARE then is to use the generalized Schur method which is based

on the Schur decomposition of a matrix pencil and does not involve computation of the inverse of A. Having said this, it should be noted that the Schur methods and the generalized Schur methods require explicit computation of the inverse of the matrix R both for the CARE and the DARE. So, when R is close to a singular matrix, the methods of choice are the inverse-free generalized Schur methods.

Newton's methods are iterative in nature and are usually used as iterative refinement techniques for solutions obtained by the Schur methods or the matrix sign function methods. **Table 13.1 presents a comparison of the different methods and recommendation based on this comparison.**

Sections **13.2 and 13.3** deal, respectively, with the results on the **existence** and **uniqueness of the stabilizing solutions** of the CARE and the DARE. **The condition numbers and bounds of the condition numbers** of the CARE and DARE are identified in Section **13.4**.

13.2 THE EXISTENCE AND UNIQUENESS OF THE STABILIZING SOLUTION OF THE CARE

The goal of this section is to derive conditions under which the CARE admits a unique symmetric positive semidefinite stabilizing solution.

For this we first need to develop an important relationship between the CARE and the associated Hamiltonian matrix and some spectral properties of this matrix.

Recall from Chapter 10 that associated with the CARE is the $2n \times 2n$ Hamiltonian matrix:

$$H = \begin{pmatrix} A & -S \\ -Q & -A^{\mathrm{T}} \end{pmatrix}. \tag{13.2.1}$$

The Hamiltonian matrix H has the following interesting spectral property.

Theorem 13.2.1. *For each eigenvalue λ of H, $-\bar{\lambda}$ is also an eigenvalue of H (with the same geometric and algebraic multiplicity as λ).*

Proof. Define the $2n \times 2n$ matrix:

$$J = \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix}, \tag{13.2.2}$$

where I is the $n \times n$ identity matrix. Then it is easy to see that $J^{-1}HJ = -JHJ = -H^{\mathrm{T}}$, which shows that H and $-H^{\mathrm{T}}$ are similar. Hence, λ is also an eigenvalue of $-H^{\mathrm{T}}$. Since the eigenvalues of $-H^{\mathrm{T}}$ are the negatives of the eigenvalues of H, and the complex eigenvalues occur in conjugate pairs, the theorem is proved. ■

The following theorems show that a solution X of the CARE is determined by the associated Hamiltonian matrix.

Theorem 13.2.2. *A matrix X is a solution of the CARE if and only if the columns of $\begin{pmatrix} I \\ X \end{pmatrix}$ span an n-dimensional invariant subspace of the Hamiltonian matrix H defined by* (13.2.1).

Proof. We first prove that if the columns of $\begin{pmatrix} I \\ X \end{pmatrix}$ span an n-dimensional invariant subspace of H, then X is a solution of the CARE.

So, assume there exists an $n \times n$ matrix L such that:

$$H \begin{pmatrix} I \\ X \end{pmatrix} = \begin{pmatrix} I \\ X \end{pmatrix} L. \tag{13.2.3}$$

Multiplying both sides of (13.2.3) by J^{-1}, where J is defined by (13.2.2), we have

$$J^{-1} H \begin{pmatrix} I \\ X \end{pmatrix} = J^{-1} \begin{pmatrix} I \\ X \end{pmatrix} L. \tag{13.2.4}$$

Noting that $J^{-1} = \begin{pmatrix} 0 & -I \\ I & 0 \end{pmatrix}$, we obtain from (13.2.4)

$$\begin{pmatrix} Q & A^{\mathrm{T}} \\ A & -S \end{pmatrix} \begin{pmatrix} I \\ X \end{pmatrix} = \begin{pmatrix} -X \\ I \end{pmatrix} L. \tag{13.2.5}$$

Premultiplying both sides of (13.2.5) by (I, X), we get

$$XA + A^{\mathrm{T}}X + Q - XSX = 0,$$

showing that X satisfies the CARE.

To prove the converse, we note that if X is a solution of the CARE, then

$$H \begin{pmatrix} I \\ X \end{pmatrix} = \begin{pmatrix} A - SX \\ -Q - A^{\mathrm{T}}X \end{pmatrix} = \begin{pmatrix} A - SX \\ X(A - SX) \end{pmatrix} = \begin{pmatrix} I \\ X \end{pmatrix} (A - SX), \tag{13.2.6}$$

that is, the columns of $\begin{pmatrix} I \\ X \end{pmatrix}$ span an invariant subspace of H. ■

Corollary 13.2.1. *If the columns of $\begin{pmatrix} X_1 \\ X_2 \end{pmatrix}$ span an n-dimensional invariant subspace of the Hamiltonian matrix H associated with the CARE and X_1 is invertible, then $X = X_2 X_1^{-1}$ is a solution of the CARE.*

Proof.

$$\begin{aligned}&\text{The span of the columns of } \begin{pmatrix} X_1 \\ X_2 \end{pmatrix} \\ &\quad = \text{the span of the columns of } \begin{pmatrix} X_1 \\ X_2 \end{pmatrix} X_1^{-1} \\ &\quad = \text{the span of the columns of } \begin{pmatrix} I \\ X_2 X_1^{-1} \end{pmatrix}.\end{aligned}$$

Therefore, by Theorem 13.2.2, we see that $X = X_2 X_1^{-1}$ is a solution of the CARE. ■

The next theorem shows how the eigenvalues of the Hamiltonian matrix H are related to those of the optimal closed-loop matrix.

Theorem 13.2.3. *Let X be a symmetric solution of the CARE. Then the eigenvalues of the Hamiltonian matrix H are the eigenvalues of $A - BK$ together with those of $-(A - BK)^{\mathrm{T}}$, where $K = R^{-1}B^{\mathrm{T}}X$.*

Proof. Define $T = \begin{pmatrix} I & 0 \\ X & I \end{pmatrix}$, where I and X are $n \times n$. Then,

$$\begin{aligned}T^{-1}HT &= \begin{pmatrix} I & 0 \\ -X & I \end{pmatrix} \begin{pmatrix} A & -S \\ -Q & -A^{\mathrm{T}} \end{pmatrix} \begin{pmatrix} I & 0 \\ X & I \end{pmatrix}, \\ &= \begin{pmatrix} A - SX & -S \\ -(A^{\mathrm{T}}X + XA + Q - XSX) & -(A - SX)^{\mathrm{T}} \end{pmatrix}, \\ &= \begin{pmatrix} A - SX & -S \\ 0 & -(A - SX)^{\mathrm{T}} \end{pmatrix}. \qquad (13.2.7)\end{aligned}$$

Thus, the eigenvalues of H are the eigenvalues of $A - SX$ together with those of $-(A - SX)^{\mathrm{T}}$.

The result now follows by noting that:

$$A - SX = A - BR^{-1}B^{\mathrm{T}}X = A - BK.$$

(Recall that $S = BR^{-1}B^{\mathrm{T}}$.) ■

Symmetric Positive Semidefinite Stabilizing Solutions of the CARE

As we have seen in Chapter 10, several applications require a symmetric positive semidefinite stabilizing solution of the associated Riccati equation. We derive in this subsection a necessary and sufficient condition for the existence of such a solution.

Recall that a symmetric solution X of (13.1.1) is a **stabilizing solution** if $A - BK = A - BR^{-1}B^{\mathrm{T}}X = A - SX$ is stable.

Proof of Theorem 13.2.4 below has been taken from Kimura (1997).

Theorem 13.2.4. *Existence and Uniqueness of the Stabilizing Solution. Assume that $R > 0$ and $Q \geq 0$, $Q \neq 0$.*

Then the following conditions are equivalent:

1. *The CARE:*

$$XA + A^{\mathrm{T}}X - XBR^{-1}B^{\mathrm{T}}X + Q = 0 \tag{13.2.8}$$

   has a unique symmetric positive semidefinite stabilizing solution X.
2. *(A, B) is stabilizable and the associated Hamiltonian matrix H has no pure imaginary eigenvalues.*

Proof of necessity. First suppose that X is a stabilizing solution of the CARE. We then show that H does not have an imaginary eigenvalue.

Since X is a stabilizing solution, $A - SX$ is stable, that is $A - BK$ is stable. From Theorem 13.2.3, we then have that n eigenvalues of H are stable and the other n have positive real parts. Thus, H does not have a purely imaginary eigenvalue.

Proof of sufficiency. Next assume that H given in (13.2.1), with $S = BR^{-1}B^{\mathrm{T}}$, has no eigenvalues on the imaginary axis. We shall then show that under the assumption of the stabilizability of (A, B), there exists a unique stabilizing solution of the CARE.

The proof will be divided in several parts.

First of all we note that the stabilizability of (A, B) implies the stabilizability of (A, S).

Since H has no pure imaginary eigenvalues, there are n stable eigenvalues of H (by Theorem 13.2.1).

Then,

$$H\begin{pmatrix} X_1 \\ X_2 \end{pmatrix} = \begin{pmatrix} X_1 \\ X_2 \end{pmatrix} E, \tag{13.2.9}$$

where E is a stable matrix and the columns of $\begin{pmatrix} X_1 \\ X_2 \end{pmatrix}$ form the eigenspace of H corresponding to these stable eigenvalues.

A. **$X_2^T X_1$ is symmetric.**
The relation (13.2.9) can be expressed as

$$AX_1 - SX_2 = X_1 E \tag{13.2.10}$$

and

$$-QX_1 - A^T X_2 = X_2 E. \tag{13.2.11}$$

Multiplying (13.2.10) by X_2^T on the left, we have

$$X_2^T A X_1 - X_2^T S X_2 = X_2^T X_1 E. \tag{13.2.12}$$

Now taking the transpose of (13.2.11), we have

$$X_2^T A = -X_1^T Q - E^T X_2^T.$$

Multiplying the last equation by X_1 to the right, we get

$$E^T X_2^T X_1 = -X_2^T A X_1 - X_1^T Q X_1. \tag{13.2.13}$$

Using (13.2.12) in (13.2.13), we then have

$$E^T X_2^T X_1 + X_2^T X_1 E = -X_2^T S X_2 - X_1^T Q X_1. \tag{13.2.14}$$

Since S and Q are symmetric, the right-hand side matrix is symmetric, and therefore the left-hand side matrix is also symmetric. This means that

$$E^T X_2^T X_1 + X_2^T X_1 E = X_1^T X_2 E + E^T X_1^T X_2$$

or

$$E^T (X_2^T X_1 - X_1^T X_2) + (X_2^T X_1 - X_1^T X_2) E = 0.$$

Since E is stable, this Lyapunov equation has a unique solution which implies that $X_2^T X_1 - X_1^T X_2 = 0$. That is, $X_2^T X_1 = X_1^T X_2$, proving that $X_2^T X_1$ is symmetric.

B. **X_1 is invertible.**

Suppose that X_1 is not invertible. Then there exists a vector $d \neq 0$ such that

$$X_1 d = 0. \tag{13.2.15}$$

Now multiplying the transpose of (13.2.10) by d^T to the left and by $X_2 d$ to the right, we have

$$\begin{aligned} d^T X_2^T S X_2 d &= -d^T E^T X_1^T X_2 d + d^T X_1^T A^T X_2 d, \\ &= -d^T E^T X_2^T X_1 d + d^T X_1^T A^T X_2 d = 0 \\ &\quad \text{(because } X_1^T X_2 = X_2^T X_1 \text{ and } X_1 d = 0). \end{aligned}$$

Again, since $S \geq 0$, we must have

$$S X_2 d = 0.$$

The Eq. (13.2.10) therefore yields

$$X_1 E d = 0.$$

As this holds for all $d \in K_{er}(X_1)$, this means that $K_{er}(X_1)$ is E-invariant, that is, there exists an eigenvalue μ of E such that

$$E d' = \mu d', \qquad X_1 d' = 0, \quad d' \neq 0. \tag{13.2.16}$$

Again, multiplying (13.2.11) by d' and using the relation (13.2.16), we obtain

$$(\mu I + A^T) X_2 d' = 0. \tag{13.2.17}$$

Also, from (13.2.10) and (13.2.16), we have

$$S X_2 d' = 0. \tag{13.2.18}$$

Since $\mathrm{Re}(\mu) < 0$ and (A, S) is stabilizable, we conclude from (13.2.18) that

$$X_2 d' = 0. \tag{13.2.19}$$

Finally, $X_2 d' = 0$ and $X_1 d' = 0$ imply that $\begin{pmatrix} X_1 \\ X_2 \end{pmatrix}$ does not have the full rank which contradicts (13.2.9).

Thus, X_1 is nonsingular.

C. **X is symmetric.**
Since X_1 is nonsingular, we have from Corollary 13.2.1 that $X = X_2X_1^{-1}$ is a solution of the CARE and, since $X_2^{\mathrm{T}}X_1$ is symmetric, so is X. This is seen as follows:

$$\begin{aligned} X^{\mathrm{T}} - X &= X_1^{-\mathrm{T}}X_2^{\mathrm{T}} - X_2X_1^{-1} \\ &= X_1^{-\mathrm{T}}(X_2^{\mathrm{T}}X_1)X_1^{-1} - X_1^{-\mathrm{T}}(X_1^{\mathrm{T}}X_2)X_1^{-1} \\ &= X_1^{-\mathrm{T}}(X_2^{\mathrm{T}}X_1 - X_1^{\mathrm{T}}X_2)X_1^{-1} = 0. \end{aligned}$$

D. **X is a stabilizing solution.**
Multiplying (13.2.10) by X_1^{-1} to the right, we obtain

$$A - SX_2X_1^{-1} = X_1EX_1^{-1}.$$

Since E is stable, so is $A - SX_2X_1^{-1} = A - SX$. Thus, X is a stabilizing solution.

E. **X is unique.**
Let X_1 and X_2 be two stabilizing solutions. Then,

$$\begin{aligned} A^{\mathrm{T}}X_1 + X_1A - X_1SX_1 + Q = 0 \\ A^{\mathrm{T}}X_2 + X_2A - X_2SX_2 + Q = 0 \end{aligned}$$

Subtracting these two equations, we have

$$A^{\mathrm{T}}(X_1 - X_2) + (X_1 - X_2)A + X_2SX_2 - X_1SX_1 = 0$$

or

$$(A - SX_1)^{\mathrm{T}}(X_1 - X_2) + (X_1 - X_2)(A - SX_2) = 0.$$

Since the last equation is a homogeneous Sylvester equation and the coefficient matrices $A - SX_1$ and $A - SX_2$ are both stable, it follows that $X_1 - X_2 = 0$, that is, $X_1 = X_2$.

F. **X is positive semidefinite.**
Since X is symmetric and satisfies (13.2.8), Eq. (13.2.8) can be written in the form of the following Lyapunov equation:

$$(A - BK)^{\mathrm{T}}X + X(A - BK) = -Q - XSX,$$

where $K = R^{-1}B^{\mathrm{T}}X$. Furthermore, $A - BK = A - BR^{-1}B^{\mathrm{T}}X = A - SX$ is stable. Thus, X can be expressed in the form **(see Chapter 7)**: $X = \int_0^\infty e^{(A-BK)^{\mathrm{T}}t}(Q + XSX)e^{(A-BK)t}dt$. Since Q and S are positive semidefinite, it follows that X is positive semidefinite. ■

Theorem 13.2.5. *Let (A, B) be stabilizable and (A, Q) be detectable. Assume that $Q \geq 0$, $S \geq 0$. Then the Hamiltonian matrix:*

$$H = \begin{pmatrix} A & -S \\ -Q & -A^{\mathrm{T}} \end{pmatrix}$$

associated with the CARE does not have a purely imaginary eigenvalue.

Proof. The proof is by contradiction.

Suppose that H has a purely imaginary eigenvalue $j\alpha$, where α is a nonnegative real number, and let $\begin{pmatrix} r \\ s \end{pmatrix}$ be the corresponding eigenvector. Then,

$$H\begin{pmatrix} r \\ s \end{pmatrix} = j\alpha\begin{pmatrix} r \\ s \end{pmatrix}, \quad \begin{pmatrix} r \\ s \end{pmatrix} \neq \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \tag{13.2.20}$$

Multiplying both sides of (13.2.20) by (s^*, r^*) to the left, we obtain

$$s^*Ar - r^*Qr - s^*Ss - r^*A^{\mathrm{T}}s = j\alpha(s^*r + r^*s)$$

or

$$(s^*Ar - r^*A^{\mathrm{T}}s) - r^*Qr - s^*Ss = j\alpha(s^*r + r^*s).$$

Considering the real part of this equation, we get

$$-r^*Qr - s^*Ss = 0.$$

Since $S \geq 0$ and $Q \geq 0$, we conclude that

$$Ss = 0 \tag{13.2.21}$$

and

$$Qr = 0. \tag{13.2.22}$$

So, from (13.2.20), we have

$$Ar = j\alpha r \tag{13.2.23}$$

and

$$-A^{\mathrm{T}}s = j\alpha s. \tag{13.2.24}$$

Thus, combining (13.2.23) and (13.2.22), we have $\begin{pmatrix} A - j\alpha I \\ Q \end{pmatrix} r = 0$. Since (A, Q) is detectable, we have $r = 0$. Similarly, using (13.2.24) and (13.2.21), one can show that $s = 0$. This gives us a contradiction that $\begin{pmatrix} r \\ s \end{pmatrix}$ is an eigenvector. Thus, H cannot have a purly imaginary eigenvalue. ■

An Expression for the Stabilizing Solution

Combining Theorem 13.2.4, Corollary 13.2.1, and Theorem 13.2.5, we arrive at the following result:

Theorem 13.2.6. *An Expression for the Unique Stabilizing Solution of the CARE. Suppose that* (A, B) *is stabilizable and* (A, Q) *is detectable. Assume that* $Q \geq 0$ *and* $R > 0$. *Then there exists a unique positive semidefinite stabilizing solution* X *of the CARE:* $XA + A^{\mathrm{T}}X - XBR^{-1}B^{\mathrm{T}}X + Q = 0$. *This solution is given by* $X = X_2X_1^{-1}$, *where the columns of the matrix* $\begin{pmatrix} X_1 \\ X_2 \end{pmatrix}$ *span the invariant subspace of the Hamiltonian matrix* (13.2.1) *associated with its stable eigenvalues.*

Remark

- The following simple example shows that the detectability of (A, Q) is not necessary for the existence of a symmetric positive semidefinite stabilizing solution of the CARE.

$$A = \begin{pmatrix} -1 & 0 \\ 0 & 2 \end{pmatrix}, \qquad B = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \qquad Q = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}, \qquad R = 1.$$

Then (A, B) is stabilizable, but (A, Q) is not detectable. The matrix $X = \begin{pmatrix} 0 & 0 \\ 0 & 4 \end{pmatrix}$ is the stabilizing solution of the CARE and is positive semidefinite.

13.3 THE EXISTENCE AND UNIQUENESS OF THE STABILIZING SOLUTION OF THE DARE

The existence and uniqueness of the stabilizing solution of the DARE can be studied via a symplectic matrix which takes the role of the Hamiltonian matrix of the CARE.

Definition 13.3.1. *A matrix* M *is* **symplectic** *if*

$$J^{-1}M^{\mathrm{T}}J = J^{\mathrm{T}}M^{\mathrm{T}}J = M^{-1},$$

where J *is defined by* (13.2.2).

Assume A is invertible and consider the matrix:

$$M = \begin{pmatrix} A + S(A^{-1})^{\mathrm{T}}Q & -S(A^{-1})^{\mathrm{T}} \\ -(A^{-1})^{\mathrm{T}}Q & (A^{-1})^{\mathrm{T}} \end{pmatrix}, \tag{13.3.1}$$

where $S = BR^{-1}B^{\mathrm{T}}$, $Q = Q^{\mathrm{T}}$, and $S = S^{\mathrm{T}}$.

Then, it can be shown (**Exercise 13.3**) that

1. M is symplectic.
2. If λ is a nonzero eigenvalue of M, so is $1/\bar{\lambda}$.

We now state the discrete counterparts of Theorems 13.2.5 and 13.2.6. The proofs can be found in Lancaster and Rodman (1995).

Theorem 13.3.1. *Let (A, B) be discrete-stabilizable and let (A, Q) be discrete-detectable. Assume that $Q \geq 0$ and $S \geq 0$. Then the symplectic matrix* (13.3.1) *has no eigenvalues on the unit circle.*

Suppose that the symplectic matrix M has no eigenvalues on the unit circle. Then it must have n eigenvalues inside the unit circle and n outside it. As in the continuous-time case, it can then be shown that if the columns of the matrix $\begin{pmatrix} X_1 \\ X_2 \end{pmatrix}$ form a basis for the invariant subspace associated with the eigenvalues inside the unit circle, then X_1 is nonsingular and $X = X_2 X_1^{-1}$ is a unique symmetric positive semidefinite stabilizing solution of the DARE.

Thus, we have the following theorem as the discrete counterpart of Theorem 13.2.6.

Theorem 13.3.2. *An Expression for the Unique Stabilizing Solution of the DARE. Suppose that (A, B) is discrete-stabilizable and (A, Q) is discrete-detectable. Assume that $Q \geq 0$, $R > 0$. Then the DARE:*

$$A^{\mathrm{T}}XA - X + Q - A^{\mathrm{T}}XB(R + B^{\mathrm{T}}XB)^{-1}B^{\mathrm{T}}XA = 0$$

has a unique symmetric positive semidefinite discrete-stabilizing solution X.

Furthermore, X is given by $X = X_2 X_1^{-1}$, where the columns of $\begin{pmatrix} X_1 \\ X_2 \end{pmatrix}$ span the n-dimensional invariant subspace of the symplectic matrix M associated with the eigenvalues inside the unit circle.

13.4 CONDITIONING OF THE RICCATI EQUATIONS

Before we describe the solution methods for Riccati equations, we state some results on the perturbation theory of such equations that will help us identify the *condition numbers* of the equations. These condition numbers, as usual, will help us understand the sensitivity of the solutions of the Riccati equations when the entries of the data matrices are slightly perturbed.

13.4.1 Conditioning of the CARE

Consider first the CARE:

$$A^{\mathrm{T}}X + XA + Q - XSX = 0, \tag{13.4.1}$$

where

$$S = BR^{-1}B^{\mathrm{T}}. \tag{13.4.2}$$

Let ΔA, ΔX, ΔQ, and ΔS be small perturbations in A, X, Q, and S, respectively. Suppose that X is the unique stabilizing solution of the CARE and that $X + \Delta X$ is the unique stabilizing solution of the perturbed Riccati equation:

$$\begin{aligned}(A + \Delta A)^{\mathrm{T}}(X + \Delta X) + (X + \Delta X)(A + \Delta A) \\ + (Q + \Delta Q) - (X + \Delta X)(S + \Delta S)(X + \Delta X) = 0.\end{aligned} \tag{13.4.3}$$

We are interested in finding an upper bound for the relative error $\|\Delta X\|/\|X\|$.

Several results exist in literature. Byers (1985) and Kenney and Hewer (1990) obtained the first-order perturbation bounds and Chen (1988), Konstantinov *et al.* (1990) gave global perturbation bound. Xu (1996) has improved Chen's result and Konstantinov *et al.* (1995) have sharpened the results of Konstantinov *et al.* (1990). The most recent result in this area is due to Sun (1998), who has improved Xu's result. We present below Sun's result and the condition numbers derived on the basis of this result.

Following the notations in Byers (1985), we define three operators:

$$\Omega(Z) = (A - SX)^{\mathrm{T}}Z + Z(A - SX), \tag{13.4.4}$$

$$\Theta(Z) = \Omega^{-1}(Z^{\mathrm{T}}X + XZ) \tag{13.4.5}$$

$$\Pi(Z) = \Omega^{-1}(XZX). \tag{13.4.6}$$

Note: Since the closed-loop matrix $A_C = A - SX$ is stable, Ω^{-1} exists. In fact, if $\Omega(Z) = W$, then

$$Z = \Omega^{-1}(W) = -\int_0^\infty e^{A_C^{\mathrm{T}}t} W e^{A_C t}\, dt,$$

and $\|\Omega^{-1}\|_{\mathrm{F}} = 1/\mathrm{sep}(A_C^{\mathrm{T}}, -A_C)$.

Define $l = \|\Omega^{-1}\|^{-1}$, $p = \|\Theta\|$, and $q = \|\Pi\|$, where $\|\cdot\|$ is any unitarily invariant norm.

Then the following perturbation result due to Sun (1998) holds:

Theorem 13.4.1. *A Perturbation Bound for the CARE. Let X and $X + \Delta X$ be, respectively, the symmetric positive semidefinite stabilizing solutions of the CARE* (13.4.1) *and the perturbed CARE* (13.4.3).

Then, for sufficiently small $[\Delta Q, \Delta A, \Delta S]$,

$$\frac{\|\Delta X\|}{\|X\|} \lesssim \frac{\|Q\|}{l\|X\|} \cdot \frac{\|\Delta Q\|}{\|Q\|} + p\frac{\|A\|}{\|X\|} \cdot \frac{\|\Delta A\|}{\|A\|} + \frac{q\|S\|}{\|X\|} \cdot \frac{\|\Delta S\|}{\|S\|}. \tag{13.4.7}$$

Using the results of Theorem 13.4.1, Sun has defined a set of condition numbers of the CARE.

The numbers:

$$\kappa^{\text{AB}}_{\text{CARE}}(Q) = \frac{1}{l}, \qquad \kappa^{\text{AB}}_{\text{CARE}}(A) = p, \qquad \text{and} \qquad \kappa^{\text{AB}}_{\text{CARE}}(S) = q$$

are the **absolute condition numbers** of X with respect to Q, A, S, respectively.

The numbers:

$$\kappa^{\text{REL}}_{\text{CARE}}(Q) = \frac{\|Q\|}{l\|X\|}, \qquad \kappa^{\text{REL}}_{\text{CARE}}(A) = \frac{p\|A\|}{\|X\|}, \qquad \text{and} \qquad \kappa^{\text{REL}}_{\text{CARE}}(S) = \frac{q\|S\|}{\|X\|}$$

are then the **relative condition numbers**.

Moreover, the scalar:

$$\kappa^{\text{REL}}_{\text{CARE}}(X) = \frac{1}{\|X\|}\sqrt{\left(\frac{\|Q\|}{l}\right)^2 + (p\|A\|)^2 + (q\|S\|)^2} \tag{13.4.8}$$

can be regarded as the **relative condition number of X**.

Using a local linear estimate, Byers (1985) has obtained an approximate condition number given by

$$\kappa^{\text{B}}_{\text{CARE}} = \frac{1}{\|X\|_{\text{F}}}\left(\frac{\|Q\|_{\text{F}}}{l} + p\|A\|_{\text{F}} + q\|S\|_{\text{F}}\right),$$

in which the operator norm $\|\cdot\|$ for defining l, p, and q is induced by the Frobenius norm $\|\cdot\|_{\text{F}}$.

The above is known as **Byers' approximate condition number.** Indeed, taking the Frobenius norm in (13.4.8), and comparing (13.4.8) with Byers' condition number, one obtains:

Theorem 13.4.2.

$$\frac{1}{\sqrt{3}}\kappa_{\text{CARE}}^{\text{B}} \leq \kappa_{\text{CARE}}^{\text{REL}}(X) \leq \kappa_{\text{CARE}}^{\text{B}}$$

Expressions for l, p, and q: If the operator norm $\|\cdot\|$ for defining l, p, and q is induced by the Frobenius norm $\|\cdot\|_{\text{F}}$, then it can be shown (Sun 1998) that

$$l = \|T^{-1}\|_2^{-1}, \qquad p = \|\text{T}^{-1}(I_n \otimes X + (X^{\text{T}} \otimes I_n)E)\|_2$$

and

$$q = \|T^{-1}(X^{\text{T}} \otimes X)\|_2,$$

where

$$T = I_n \otimes (A - SX)^{\text{T}} + (A - SX)^{\text{T}} \otimes I_n, \qquad \text{and } A - SX \text{ is stable.}$$

E is the vec-permutation matrix:

$$E = \sum_{i,j=1}^{n} (e_i e_j^{\text{T}}) \otimes (e_j e_i^{\text{T}}).$$

Remark

- A recent paper of Petkov *et al.* (1998) contains results on estimating the quantities l, p, and q.

Estimating Conditioning of the CARE using Lyapunov Equations

Computing the quantities l, p, and q using the Kronecker products is computationally intensive. On the other hand (*using the 2-norm in the definition of Byers' condition number*) Kenney and Hewer (1990) have obtained an upper and a lower bound of $\kappa_{\text{CARE}}^{\text{B}}$ by means of solutions of certain Lyapunov equations, which are certainly computationally much less demanding than computing Kronecker products. Using these results, ill-conditioning of κ_{CARE} can be more easily detected.

Assume that $A - SX$ is stable and let H_k be the solution to the Lyapunov equation:

$$(A - SX)^{\text{T}} H_k + H_k(A - SX) = -X^k, \quad k = 0, 1, 2. \tag{13.4.9}$$

Furthermore, let's define $H_1^{(1)}$ as follows:

Set $\tilde{Q} = 2X$ and solve the successive Lyapunov equations for $\tilde{H}$ and H, respectively:

$$(A - SX)^{\mathrm{T}}\tilde{H} + \tilde{H}(A - SX) = \tilde{Q} \tag{13.4.10}$$

and

$$(A - SX)H + H(A - SX)^{\mathrm{T}} = \tilde{H}. \tag{13.4.11}$$

Let

$$W = 2XH \quad \text{and} \quad H_1^{(1)} = \Theta\left(\frac{W}{\|W\|}\right).$$

Define

$$U = \frac{\|H_0\|\|Q\| + 2\|H_0\|^{1/2}\|H_2\|^{1/2}\|A\| + \|H_2\|\|S\|}{\|X\|} \tag{13.4.12}$$

and

$$L = \frac{\|H_0\|\|Q\| + \|H_1^{(1)}\|\|A\| + \|H_2\|\|S\|}{\|X\|}, \tag{13.4.13}$$

Then it has been shown that:

$$L \leq \kappa_{\mathrm{CARE}}^{\mathrm{B}} \leq U \tag{13.4.14}$$

From the relations (13.4.12)–(13.4.14), we see that $\kappa_{\mathrm{CARE}}^{\mathrm{B}}$ will be large (and consequently the CARE will be ill-conditioned) if H_0, $H_1^{(1)}$, and H_2 have large norms (relative to that of X). Conversely, if the norms of H_0, $H_1^{(1)}$, and H_2 are not large, then the CARE will be well-conditioned.

If the norms vary widely in the sense that there is a mixture of large and small norms, then there will be **selective sensitivity.** More specifically, the ratios:

$$r_1 = \frac{\|H_0\|\|Q\|}{\|X\|}, \qquad r_2 = \frac{\|H_1^{(1)}\|\|A\|}{\|X\|}, \qquad \text{and } r_3 = \frac{\|H_2\|\|S\|}{\|X\|}$$

measure, respectively, the sensitivity of X with respect to perturbations in the matrix Q, the matrix A, and the matrix S.

Example 13.4.1. An Ill-Conditioned CARE.

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 0.0010 & 4 & 5 \\ 0 & 7 & 8 \end{pmatrix}, \qquad B = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \qquad R = 1.$$

$$Q = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 5 & 3 \\ 1 & 3 & 5 \end{pmatrix}.$$

$$X = 10^9 \begin{pmatrix} 0 & 0.0003 & 0.0004 \\ 0.0003 & 4.5689 & 5.3815 \\ 0.0004 & 5.3815 & 6.3387 \end{pmatrix}.$$

The residual norm of the solution X: $\|XA + A^{\mathrm{T}}X - XSX + Q\| = O(10^{-5})$.

$$\|H_0\| = 5.6491 \times 10^8, \quad \|H_1\| = 1.8085 \times 10^9, \quad \text{and} \quad \|H_2\| = 4.8581 \times 10^{18}.$$

U and L **are both of order 10^8.**

Thus, the Riccati equation is expected to be ill-conditioned with the given data.

Indeed, this is an example of mixed sensitivity. Note that the ratios r_2 and r_3 are large, but r_1 is quite small. Thus, X **should be sensitive with respect to perturbation in A and S.** This is verified as follows.

Let M_{new} stand for a perturbed version of the matrix M and X_{new} stands for the new solution of the ARE with the perturbed data.

Case 1. *Perturbation in A.* Let A be perturbed to $A + \Delta A$, where

$$\Delta A = 10^{-8} \begin{pmatrix} 3.169 & 2.668 & 3.044 \\ -1.259 & -0.5211 & -2.364 \\ 2.798 & 3.791 & -3.179 \end{pmatrix}.$$

The matrices B, Q, and R remain unperturbed.

Then,

$$\text{Relative error in } X\text{: } \frac{\|X_{\text{new}} - X\|}{\|X\|} = 4.1063 \times 10^{-5},$$

$$\text{Relative perturbation in } A\text{: } \frac{\|A_{\text{new}} - A\|}{\|A\|} = 4.9198 \times 10^{-9}.$$

Case 2. *Perturbation in B.* Let B be perturbed to $B + \Delta B$, where

$$\Delta B = 10^{-8} \begin{pmatrix} -4.939 \\ 0.7715 \\ -0.9411 \end{pmatrix}.$$

A, Q, R remain unperturbed.

$$\text{Relative error in} X\text{:} \ \frac{\|X_{\text{new}} - X\|}{\|X\|} = 7.5943 \times 10^{-5},$$

$$\text{Relative perturbation in} B\text{:} \ \frac{\|B_{\text{new}} - B\|}{\|B\|} = 5.086 \times 10^{-8}.$$

Case 3. *Perturbation in Q.* The matrix Q is perturbed such that the relative perturbation in Q

$$\frac{\|Q_{\text{new}} - Q\|}{\|Q\|} = 4.048 \times 10^{-9}.$$

The matrices A, B, and R remain unperturbed.

$$\text{Then the relative error in } X\text{:} \ \frac{\|X_{\text{new}} - X\|}{\|X\|} = 4.048 \times 10^{-9}.$$

Note: All the solutions to the CARE in this example were computed using the Schur method (**Algorithm 13.5.1**) followed by Newton's iterative refinement procedure (**Algorithm 13.5.8**). The residual norms of the solutions obtained by the Schur method alone were of order 10^5. On the other hand, the residual norm of the solution with the Schur method followed by Newton's iterative procedure was, in each case, of order 10^{-5}.

Example 13.4.2. A Well-Conditioned CARE.

$$A = \begin{pmatrix} -1 & 1 & 1 \\ 0 & -2 & 0 \\ 0 & 0 & -3 \end{pmatrix}, \qquad B = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix},$$

$$Q = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad \text{and} \quad R = 1.$$

In this case $\|H_0\| = 0.3247$, $\|H_1\| = 0.1251$, $\|H_2\| = 0.0510$, and $U = 3.1095$.

The CARE is, therefore, expected to be well-conditioned.

Indeed, if (1, 1) entry of A is perturbed to -0.9999999, and the other data remain unchanged, then we find

Relative error in X: $\|X_{\text{new}} - X\|/\|X\| = 5.6482 \times 10^{-8}$.

Relative perturbation in A: $\|A_{\text{new}} - A\|/\|A\| = 3.1097 \times 10^{-8}$, where A_{new} and X_{new}, respectively, denote the perturbed A and the solution of the CARE with the perturbed data.

Conditioning and Accuracy

Suppose that $\hat{X}$ is an approximate stabilizing solution of the CARE:

$$XA + A^{\mathrm{T}}X - XSX + Q = 0,$$

where $S = BR^{-1}B^{\mathrm{T}}$ and let $\mathrm{Res}(\hat{X}) = \hat{X}A + A^{\mathrm{T}}\hat{X} - \hat{X}S\hat{X} + Q$.

Then the question arises: If $\mathrm{Res}(\hat{X})$ **is small, does it guarantee that the error in the solution is also small (Exercise 13.8).** In the case of linear system problem, it is well-known that the smallness of the residual does not guarantee that the error in the solution is small, if the linear system problem is ill-conditioned. Similar result can be proved in the case of the Riccati equations (see Kenney *et al.* 1990). **The result basically says that even if the residual is small, the computed solution may be inaccurate, if the CARE is ill-conditioned.** On the other hand, if $\mathrm{Res}(\hat{X})$ is small and the CARE is well-conditioned, then the solution is guaranteed to be accurate. Below, we quote a recent result of Sun (1997a) which is an improvement of the result of Kenney *et al.* (1990).

Theorem 13.4.3. *Residual Bound of an Approximate Stabilizing Solution. Let* $\hat{X} \geq 0$ *approximate the positive semidefinite stabilizing solution* X *to the CARE. Define the linear operator* $T : \mathbb{R}^{n\times n} \to \mathbb{R}^{n\times n}$ *by*

$$T(Z) = (A - S\hat{X})^{\mathrm{T}}Z + Z(A - S\hat{X}), \qquad Z = Z^{\mathrm{T}} \in \mathbb{R}^{n\times n}.$$

Assuming that $4\|T^{-1}\|\|T^{-1}(\mathrm{Res}(\hat{X}))\|\|S\| < 1$ *for any unitarily invariant norm* $\|\cdot\|$, *then*

$$\frac{\|\hat{X} - X\|}{\|\hat{X}\|} \leq \frac{2}{1 + \sqrt{1 - 4\|T^{-1}\|\|T^{-1}\mathrm{Res}(\hat{X})\|\|S\|}} \frac{\|T^{-1}\mathrm{Res}(\hat{X})\|}{\|\hat{X}\|}.$$

13.4.2 Conditioning of the DARE

Consider now the DARE:

$$A^{\mathrm{T}}XA - X + Q - A^{\mathrm{T}}XB(R + B^{\mathrm{T}}XB)^{-1}B^{\mathrm{T}}XA = 0.$$

The condition number of the DARE, denoted by κ_{DARE}, may be obtained by means of the **Frechet derivative** of the DARE (Gudmundsson *et al.* 1992).

$$\text{Define } A_{\mathrm{d}} = A - B(R + B^{\mathrm{T}}XB)^{-1}B^{\mathrm{T}}XA, \qquad S = BR^{-1}B^{\mathrm{T}}. \tag{13.4.15}$$

Assume that X is the stabilizing solution of the DARE. Then the condition number of the DARE is given by:

$$\kappa_{\mathrm{DARE}} = \frac{\|[Z_1, Z_2, Z_3]\|_2}{\|X\|_{\mathrm{F}}}, \tag{13.4.16}$$

where

$$Z_1 = \|A\|_F P^{-1}(I \otimes A_d^T X + (A_d^T X \otimes I)E), \tag{13.4.17}$$

$$Z_2 = -\|S\|_F P^{-1}(A^T X(I + SX)^{-1} \otimes A^T X(I + SX)^{-1}), \tag{13.4.18}$$

and

$$Z_3 = \|Q\|_F P^{-1}. \tag{13.4.19}$$

In the above, E is the vec-permutation matrix:

$$E = \sum_{i,j=1}^{n} e_i e_j^T \otimes e_j e_i^T, \tag{13.4.20}$$

and P is a matrix representation of the Stein operator:

$$\Omega(Z) = Z - A_d^T Z A_d. \tag{13.4.21}$$

Note that, since A_d is discrete-stable, P^{-1} exists.

The condition number (13.4.16) measures the sensitivity of the stabilizing solution X of the DARE with respect to first-order perturbations.

Assume that the bounds for ΔA, ΔS, and ΔQ are **sufficiently small**. Then, using first-order perturbation only, it can be shown **(Exercise 13.7)** that the following quantity is an **approximate condition number** of the DARE:

$$\frac{2\|A\|_F^2\|Q\|_F/\|X\|_F + \|A\|_F^2\|S\|_F\|X\|_F}{\text{sep}_d(A_d^T, A_d)}, \tag{13.4.22}$$

where

$$\text{sep}_d(A_d^T, A_d) = \min_{X \neq 0} \frac{\|A_d^T X A_d - X\|_F}{\|X\|_F}. \tag{13.4.23}$$

Note: The quantity $\text{sep}(A_d^T, A_d)$ can be computed as the minimum singular value of the matrix:

$$A_d^T \otimes A_d^T - I_{n^2}.$$

Remark

- A perturbation theorem for the DARE, analogous to Theorem 13.4.1 (for the CARE), and the absolute and relative condition numbers using the results of that theorem can be obtained. For details see Sun (1998).

Also, a recent paper of Sima *et al.* (2000) contains efficient and reliable condition number estimators both for the CARE and DARE.

Example 13.4.3. (An Ill-Conditioned DARE.) Let's take A, B, Q, and R the same as in Example 13.4.1.

Let $A_{\text{new}} = \begin{pmatrix} 0.9980 & 2 & 3 \\ 0.0010 & 4 & 5 \\ 10^{-8} & 7 & 8 \end{pmatrix}$. Let B, Q, and R remain unchanged.

The solution X of the DARE (computed by MATLAB function **dare**) is

$$X = 10^{10} \begin{pmatrix} 0.0000 & 0.0005 & 0.0005 \\ 0.0005 & 5.4866 & 6.4624 \\ 0.0005 & 6.4624 & 7.6118 \end{pmatrix}.$$

The solution X_{new} of the perturbed version of the DARE is

$$X_{\text{new}} = 10^{10} \begin{pmatrix} 0.0000 & 0.0005 & 0.0005 \\ 0.0005 & 5.4806 & 6.4554 \\ 0.0005 & 6.4554 & 7.6036 \end{pmatrix}.$$

Relative error in X: $\|X - X_{\text{new}}\|/\|X\| = 0.0010$, while the perturbations in A were of order $O(10^{-4})$.

Example 13.4.4. (A Well-Conditioned DARE.) Let

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \\ 3.999 & 6 & 7 \end{pmatrix}.$$

Take B, Q, and R the same as in Example 13.4.1.

Let

$$A_{\text{new}} = \begin{pmatrix} 0.9990 & 2 & 3 \\ 2 & 3 & 4 \\ 4 & 6 & 7 \end{pmatrix}, \quad B_{\text{new}} = B, \quad Q_{\text{new}} = Q, \quad \text{and} \quad R_{\text{new}} = R.$$

Then both the relative error in X and the relative perturbation in A are of $O(10^{-4})$. In this case, $\text{sep}(A_{\text{d}}^{\text{T}}, A_{\text{d}}) = 0.0011$.

13.5 COMPUTATIONAL METHODS FOR RICCATI EQUATIONS

The computational methods (**listed in the Introduction**) for the AREs can be broadly classified into three classes:

- The Invariant Subspace Methods
- The Deflating Subspace Methods
- Newton's Methods.

The eigenvector, Schur vector, and matrix sign function methods are examples of the invariant subspace methods. The generalized eigenvector and generalized Schur vector methods are examples of the deflating subspace methods.

The following methods have been included in our discussions here. For the CARE:

- **The eigenvector method** (Section 13.5.1)
- The **Schur method** (Algorithm 13.5.1)
- The **Hamiltonian Schur method** (Section 13.5.1)
- **The inverse-free generalized Schur method** (Algorithm 13.5.3)
- **The matrix sign function method** (Algorithm 13.5.6)
- **Newton's method** (Algorithm 13.5.8)
- **Newton's method with line search** (Algorithm 13.5.9).

For the DARE:

- **The Schur method** (Section 13.5.1)
- **The generalized Schur method** (Algorithm 13.5.2)
- The **inverse-free generalized Schur method** (Algorithm 13.5.4).
- The **matrix sign function method** (Algorithm 13.5.7)
- **Newton's method** (Algorithm 13.5.10)
- **Newton's method with line search** (Algorithm 13.5.11).

13.5.1 The Eigenvector and Schur Vector Methods

An invariant subspace methods for solving the CARE (DARE) is based on computing a stable invariant subspace of the associated Hamiltonian (symplectic) matrix; that is the subspace corresponding to the eigenvalues with the negative real parts (inside the unit circle). If this subspace is spanned by $\binom{X_1}{X_2}$ and X_1 is invertible, then $X = X_2X_1^{-1}$ is a stabilizing solution.

To guarantee the existence of such a solution, it will be assumed throughout this section that (A, B) *is stabilizable (discrete-stabilizable) and the Hamiltonian matrix H (symplectic matrix M) does not have an imaginary eigenvalue (an eigenvalue on the unit circle).* Note that a sufficient condition for the existence of a unique positive *semidefinite* stabilizing solution of the CARE(DARE) was given Theorem 13.2.6 (Theorem 13.3.2).

The Eigenvector Method for the CARE

Let H be diagonalizble and have the eigendecomposition:

$$V^{-1}HV = \begin{pmatrix} -\bar{\Lambda} & 0 \\ 0 & \Lambda \end{pmatrix},$$

where $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_n)$ and $\lambda_1, \ldots, \lambda_n$ are the n eigenvalues of H with positive real parts.

Let V be partitioned conformably:

$$V = \begin{pmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{pmatrix}$$

such that $\begin{pmatrix} V_{11} \\ V_{21} \end{pmatrix}$ is the matrix of eigenvectors corresponding to the stable eigenvalues. Then it is easy to see that

$$H \begin{pmatrix} V_{11} \\ V_{21} \end{pmatrix} = \begin{pmatrix} V_{11} \\ V_{21} \end{pmatrix} (-\bar{\Lambda}).$$

Thus, $X = V_{21} V_{11}^{-1}$ is the unique stabilizing solution.

Remark

- **The eigenvector method, in general, cannot be recommended for practical use.** The method becomes highly unstable if the Hamiltonian matrix H is defective or nearly defective, that is, if there are some multiple or near multiple eigenvalues of H. In these cases, the matrix V_{11} will be poorly conditioned, making $X = V_{21} V_{11}^{-1}$ inaccurate; and this might happen even if the CARE itself is not ill-conditioned.

The eigenvector method, in principle, is applicable even when H is not diagonalizable by computing the principal vectors, **but again is not recommended in practice.**

MATCONTROL note: The eigenvector method for the CARE has been implemented in MATCONTROL function **riceigc**.

The Eigenvector Method for the DARE

An analogous method for the DARE can be developed by taking the eigendecomposition of the associated symplectic matrix M. However, since forming the matrix M requires computation of A^{-1}, **the eigenvector method for the DARE works only when A is nonsingular. But even in this case, the results will be inaccurate if A is ill-conditioned.** Moreover, the method will have the same sort of difficulties as those mentioned above for the CARE. We, thus, skip the description of the eigenvector method for the DARE.

The Schur Vector Method for the CARE

The numerical difficulties of the eigenvector method for the CARE may somehow be reduced or eliminated if the Hamiltonian matrix H is transformed to **an ordered Real Schur form** (RSF) by using the QR iteration algorithm, rather than using its eigendecomposition.

Let $U^{\mathrm{T}}HU$ be an **ordered Real Schur matrix:**

$$U^{\mathrm{T}}HU = \begin{pmatrix} T_{11} & T_{12} \\ 0 & T_{22} \end{pmatrix},$$

where the eigenvalues of H with negative real parts have been stacked in T_{11} and those with positive real parts are stacked in T_{22}.

Let

$$U = \begin{pmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{pmatrix}$$

be a conformable partitioning of U. Then,

$$H \begin{pmatrix} U_{11} \\ U_{21} \end{pmatrix} = \begin{pmatrix} U_{11} \\ U_{21} \end{pmatrix} T_{11}.$$

Thus, the matrix $X = U_{21}U_{11}^{-1}$ is then the unique stabilizing solution of the CARE.

The above discussion leads to the following algorithm, called the *Schur algorithm*, due to Laub (1979).

Algorithm 13.5.1. *The Schur Algorithm for the CARE*

Inputs.

A—An $n \times n$ matrix

B—An $n \times m$ $(m \leq n)$ matrix

Q—An $n \times n$ symmetric matrix

R—An $m \times m$ symmetric matrix.

Output.

X—The unique stabilizing solution of the CARE.

Step 1. *Form the Hamiltonian matrix*

$$H = \begin{pmatrix} A & -BR^{-1}B^{\mathrm{T}} \\ -Q & -A^{\mathrm{T}} \end{pmatrix}.$$

Step 2. *Transform H to the* ***ordered RSF:***

$$U^{\mathrm{T}}HU = \begin{pmatrix} T_{11} & T_{12} \\ 0 & T_{22} \end{pmatrix},$$

where the n eigenvalues of H with negative real parts are contained in T_{11}.

Step 3. *Partition U conformably:*

$$U = \begin{pmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{pmatrix}.$$

Step 4. *Compute the solution X by solving the linear systems:*

$$XU_{11} = U_{21}.$$

Software for the ordered RSF

The ordered RSF of H can be obtained by transforming H first to the RSF by orthogonal similarity, followed by another orthogonal similarity applied to the RSF to achieve the desired ordering of the eigenvalues (**See Chapter 4**).

There exists an efficient algorithm and an associated software developed by Stewart (1976) for this purpose: Algorithm 506 of the Association for Computing Machinery Trans. Math Software (1976), pp. 275–280. See also the LAPACK routine STRSEN.

The MATLAB program **ordersch** from MATCONTROL can also be used for this purpose.

Flop-count: The Schur method is based on reduction to RSF, which is done by QR iterations algorithm; so, an exact flop-count cannot be given. However, assuming that the average number of iterations per eigenvalue is 2, about $200n^3$ flops will be necessary to execute the algorithm. (This count also takes into account of the ordering of RSF).

Example 13.5.1. Consider solving the CARE with:

$$A = \begin{pmatrix} -1 & 1 & 1 \\ 0 & -2 & 0 \\ 0 & 0 & -3 \end{pmatrix}, \qquad Q = I_{3\times 3}, \qquad S = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix},$$

$$B = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \qquad R = 1.$$

Step 1. Form the Hamiltonian matrix

$$H = \begin{pmatrix} A & -S \\ -Q & -A^{\mathrm{T}} \end{pmatrix} = \left(\begin{array}{ccc|ccc} -1 & 1 & 1 & -1 & -1 & -1 \\ 0 & -2 & 0 & -1 & -1 & -1 \\ 0 & 0 & -3 & -1 & -1 & -1 \\ \hline -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & -1 & 2 & 0 \\ 0 & 0 & -1 & -1 & 0 & 3 \end{array}\right).$$

Step 2. Transform the Hamiltonian matrix to the ordered RSF:

$$U^{\mathrm{T}}HU = \begin{pmatrix} T_{11} & T_{12} \\ 0 & T_{22} \end{pmatrix},$$

$$= \left(\begin{array}{ccc|ccc} -2.9940 & -0.0216 & 1.3275 & & & \\ 0 & -2.1867 & 0.7312 & & * & \\ 0 & -0.2573 & -1.9055 & & & \\ \hline & & & 2.9940 & 1.3285 & 0.2134 \\ & 0 & & 0 & 1.9623 & 0.2434 \\ & & & 0 & -0.7207 & 2.1298 \end{array}\right).$$

The eigenvalues of T_{11} are: -2.9940, $-2.0461+0.4104j$, $-2.0461-0.4104j$. Thus, all the stable eigenvalues are contained in T_{11}.

Step 3. Extract U_{11} and U_{21} from U:

$$U_{11} = \begin{pmatrix} 0.4417 & 0.3716 & 0.7350 \\ 0.0053 & -0.8829 & 0.3951 \\ -0.8807 & 0.1802 & 0.3986 \end{pmatrix}, \qquad U_{21} = \begin{pmatrix} 0.1106 & 0.0895 & 0.3260 \\ 0.0232 & -0.1992 & 0.1552 \\ -0.1285 & 0.0466 & 0.1199 \end{pmatrix}.$$

Step 4. Compute the stabilizing solution:

$$X = U_{21}U_{11}^{-1} = \begin{pmatrix} 0.3732 & 0.0683 & 0.0620 \\ 0.0683 & 0.2563 & 0.0095 \\ 0.0620 & 0.0095 & 0.1770 \end{pmatrix}.$$

The eigenvalues of $A - SX$ are: $-2.0461 + 0.4104j$, $-2.0461 - 0.4104j$, -2.9940. Thus, $A - SX$ is stable, that is, X is a unique stabilizing solution.

MATCONTROL note: The Schur method for the CARE (using ordered RSF) has been implemented in MATCONTROL function **ricschc**.

Stability Analysis of the Schur Method and Scaling

The round-off properties of the Schur method are quite involved. It can be shown (Petkov *et al.* 1991) that the relative error in the computed solution is proportional to $\|U_{11}^{-1}\|/\text{sep}(T_{11}, T_{22})$.

This means that the Schur method can be numerically unstable even if the CARE is not ill-conditioned. For example, the Schur method can be unstable if the Hamiltonian matrix H is nearly defective.

However, the difficulty can be overcome by proper scaling (Kenney *et al.* 1989). **Thus, for all practical purposes, the Schur method, when combined with an appropriate scaling, is numerically stable.** For a discussion on scaling procedure, see Kenney *et al.* (1989), and Benner (1997). See also Pandey (1993).

Benner (1997) has given an extensive discussion on scaling. Based on several existing scaling strategies and considering the practical difficulties with these strategies, he has proposed a mixture of these procedures for scaling the CARE. Benner's strategy is as follows:

Write the CARE:

$$XA + A^{\mathrm{T}}X - XSX + Q = 0$$

in the form:

$$X_\rho A_\rho + A_\rho^{\mathrm{T}} X_\rho - X_\rho S_\rho X_\rho + Q = 0,$$

where $A_\rho = \rho A$, $A_\rho^{\mathrm{T}} = (\rho A)^{\mathrm{T}}$, $X_\rho = X/\rho$, and $S_\rho = \rho^2 S$, ρ being a positive scalar.

Choose ρ as

$$\rho = \begin{cases} \dfrac{\|S\|_2}{\|Q\|_2}, & \text{if } \|Q\|_2 > \|S\|_2 \\ \dfrac{\|A\|_2}{\|S\|_2}, & \text{if } \|Q\|_2 \le \|S\|_2 \text{ and } \|Q\|_2\|S\|_2 < \|A\|_2^2 \\ 1, & \text{otherwise.} \end{cases}$$

For a rationale of choosing ρ this way, see Benner (1997).

Note: Note that the relative condition number of the CARE remains invariant under the above scaling.

The Schur Method for the DARE

The Schur method for the DARE:

$$A^{\mathrm{T}}XA - X - A^{\mathrm{T}}XB(R + B^{\mathrm{T}}XB)^{-1}B^{\mathrm{T}}XA + Q = 0$$

is analogous. Form the symplectic matrix:

$$M = \begin{pmatrix} A + S(A^{-1})^{\mathrm{T}}Q & -S(A^{-1})^{\mathrm{T}} \\ -(A^{-1})^{\mathrm{T}}Q & (A^{-1})^{\mathrm{T}} \end{pmatrix},$$

where $S = BR^{-1}B^{\mathrm{T}}$.

Let M be transformed to an ordered RSF such that the eigenvalues with moduli less than 1 appear in the first block, that is, an orthogonal matrix U can be constructed such that

$$U^{\mathrm{T}}MU = \begin{pmatrix} S_{11} & S_{12} \\ 0 & S_{22} \end{pmatrix},$$

where each eigenvalue of S_{11} is inside the unit circle. Partition U conformably:

$$U = \begin{pmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{pmatrix}.$$

Then $X = U_{21}U_{11}^{-1}$ is the unique stabilizing solution of the DARE.

Remarks

- Since one needs to form A^{-1} explicitly to compute M, the Schur method for the DARE is not applicable if A is singular. Even if A is theoretically nonsingular, **the method is expected to give an inaccurate answer in case A is ill-conditioned with respect to inversion.**
- A slightly faster method (Sima (1996, p. 244)) forms the matrix M^{-1} and orders the RSF so that the eigenvalues with moduli less than 1 appear in the first block.

MATCONTROL note: The Schur method for the DARE has been implemented in MATCONTROL function **ricschd**.

The Hamiltonian–Schur Methods for the CARE

The Schur methods for the AREs are based on orthogonal similarity transformations of the associated Hamiltonian and symplectic matrices to RSFs. The rich structures of these matrices are, however, not exploited in these methods. The Hamiltonian and the symplectic matrices are treated just as $2n \times 2n$ general matrices in these methods. **It would be useful if methods could be developed that could take advantage of Hamiltonian and Symplectic structures.** Such structure-preserving methods, besides reflecting physical structures, are often faster.

Theorem 13.5.1 below shows that developments of such structure-preserving methods are possible.

Definition 13.5.1. *If a matrix U is both symplectic and unitary, it is called a* **symplectic–unitary** *matrix. A* ***symplectic–orthogonal*** *matrix can be similarly defined.*

From the above definition, it follows that a $2n \times 2n$ symplectic–unitary matrix U can be written as:

$$U = \begin{pmatrix} U_{11} & U_{12} \\ -U_{12} & U_{11} \end{pmatrix},$$

where U_{11} and U_{12} are $n \times n$. If $\hat{U}$ is $n \times n$ unitary, then

$$U = \begin{pmatrix} \hat{U} & 0_{n\times n} \\ 0_{n\times n} & \hat{U} \end{pmatrix}$$

is symplectic–unitary.

Theorem 13.5.1. *The Hamiltonian–Schur Decomposition (HSD) Theorem. (Paige and Van Loan 1981). If the real parts of all the eigenvalues of a Hamiltonian matrix H are nonzero, then there exists a symplectic–orthogonal matrix U and a Hamiltonian matrix T such that*

$$U^{\mathrm{T}}HU = T = \begin{pmatrix} T_1 & T_2 \\ 0_{n\times n} & -T_1^{\mathrm{T}} \end{pmatrix},$$

where T_1 is an $n \times n$ upper triangular, and T_2 is an $n \times n$ symmetric matrix. Furthermore, U and T can be chosen so that the eigenvalues of T_1 have negative real parts.

Definition 13.5.2. *The Hamiltonian matrix T in Theorem* 13.5.1 *is called a* **Hamiltonian–Schur matrix** *and the decomposition itself is called the* **HSD.**

Note: The first n columns of U in the above HSD span the invariant subspace corresponding to the stabilizing solution of the CARE.

Symplectic-Schur Decomposition (SSD)

For a symplectic matrix, we have the following theorem.

Theorem 13.5.2. *The SSD Theorem. If M is symplectic and has no eigenvalues on the unit circle, then there exists a symplectic–orthogonal matrix U such that*

$$U^{\mathrm{T}}MU = R = \begin{pmatrix} R_1 & R_2 \\ 0_{n\times n} & R_1^{-\mathrm{T}} \end{pmatrix},$$

where R_1 is $n \times n$ upper triangular. Moreover, R_2R_1 is symmetric.

Definition 13.5.3. *The above decomposition is called an* **SSD.**

The existence of the HSD and the SSD naturally lead to the following problem: **How to obtain these decompositions in a numerically effective way by exploiting the structures of the Hamiltonian and the symplectic matrices?**

Byers (1983, 1986a) first developed such a structure-preserving method for the HSD in the case the matrix S in the Hamiltonian matrix:

$$H = \begin{pmatrix} A & -S \\ -Q & -A^{\mathrm{T}} \end{pmatrix},$$

has rank 1. (For example, a single-input problem).

Definition 13.5.4. *A Hamiltonian matrix H has* ***Hamiltonian-Hessenberg*** *form, if it has the zero structure of a $2n \times 2n$ upper Hessenberg matrix with the order of the last n rows and columns reversed.*

As in the standard QR iteration algorithm for the RSF of a matrix A, Byers' method also comes in two stages:

Stage I. The matrix H is reduced to a **Hamiltonian-Hessenberg matrix** H_{H} by an orthogonal–symplectic transformation.

Stage II. The Hamiltonian-Hessenberg matrix H_{H} is further reduced to Hamiltonian–Schur form using Hamiltonian QR iterations.

Of course, once such a reduction is done, this can immediately be used to solve the CARE.

For a complete description of the method and details of numerical implementations, see Byers (1986a).

Unfortunately, in spite of several attempts, such a reduction in the general case of a Hamiltonian matrix remained a difficult problem, until the recent paper of Benner *et al.* (1997c).

A Hamiltonian–Schur Method for the CARE (rank $S \geq 1$)

We next outline briefly the Hamiltonian–Schur method of Benner *et al.* (1997c) for solving the CARE in the multi-input case. The method also uses symplectic–orthogonal transformations in the reduction to the Hamiltonian–Schur form of the matrix H_{E} defined below.

The method is based on an interesting relationship between the invariant subspaces of the Hamiltonian matrix H and the extended matrix

$$\begin{pmatrix} 0 & H \\ H & 0 \end{pmatrix}.$$

It makes use of the **symplectic URV-like decomposition** that was also introduced by the authors (Benner *et al.* 1999c).

Theorem 13.5.3. *Symplectic-URV Decomposition. Given a* $2n \times 2n$ *Hamiltonian matrix* H, *there exist symplectic–orthogonal matrices* U_1 *and* U_2 *such that*

$$H = U_2 \begin{pmatrix} H_{\mathrm{t}} & H_r \\ 0 & -H_{\mathrm{b}}^{\mathrm{T}} \end{pmatrix} U_1^{\mathrm{T}},$$

where H_r *is an* $n \times n$ *matrix,* H_{t} *is an* $n \times n$ *upper triangular matrix and* H_{b} *is an* $n \times n$ *real Schur matrix.*

Furthermore, the positive and negative square roots of the eigenvalues of $H_{\mathrm{t}} H_{\mathrm{b}}$ *are the eigenvalues of* H.

The basis of the Hamiltonian–Schur method is the following result.

Theorem 13.5.4. *Extended HSD Theorem. Suppose that the Hamiltonian matrix*

$$H = \begin{pmatrix} A & -S \\ -Q & -A^{\mathrm{T}} \end{pmatrix}$$

has no purely imaginary eigenvalues. Define

$$H_{\mathrm{E}} = \begin{pmatrix} 0 & H \\ H & 0 \end{pmatrix}.$$

Then there exists an orthogonal matrix U *of order* $4n$ *such that*

$$U^{\mathrm{T}} H_{\mathrm{E}} U = T = \begin{pmatrix} T_1 & T_2 \\ 0 & -T_1^{\mathrm{T}} \end{pmatrix}$$

is in Hamiltonian–Schur form and no eigenvalues of T_1 *have negative real parts.*

Remark

- Note that the transforming matrix U in Theorem 13.5.4 is not symplectic–orthogonal. But this non-symplectic transformation can be computed without rounding errors!

Solution of the CARE using the Extended HSD

Let H have no eigenvalue on the imaginary axis. Let the matrix U in Theorem 13.5.4 be partitioned as

$$U = \begin{pmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{pmatrix},$$

where each U_{ij} is of order $2n \times 2n$. Define the matrix $\hat{Y}$ as

$$\hat{Y} = \frac{\sqrt{2}}{2}(U_{11} - U_{21}).$$

Let Y be an orthogonal basis of Range($\hat{Y}$). Then it has been shown (Benner *et al.* 1997c) that

$$\text{Range}(Y) = \text{Inv}(H),$$

where Inv(H) is the invariant subspace associated with the eigenvalues of H with negative real parts.

Furthermore, if

$$\hat{Y} = \begin{pmatrix} \hat{Y}_1 \\ \hat{Y}_2 \end{pmatrix},$$

where $\hat{Y}_1$ and $\hat{Y}_2$ are of order $n \times 2n$, then the stabilizing solution X of the CARE is given by

$$X\hat{Y}_1 = -\hat{Y}_2.$$

Note that the above equations represent an overdetermined consistent set of linear equations.

The symplectic-URV decomposition is used to compute the matrix U to achieve the Hamiltonian–Schur matrix T. Note also that it is not necessary to explicitly compute Y, if only the stabilizing solution of the CARE is sought.

The details are rather involved and we refer the readers to the paper of Benner *et al.* 1997c).

Efficiency and stability: The method based on the above discussion is more efficient than the Schur method. It has also been shown that the method computes the Hamiltonian–Schur form of a Hamiltonian matrix close to $\tilde{H}_E$, where $\tilde{H}_E$ is permutationally similar to H_E, that is, there exists a permutation matrix P such that $PH_EP^T = \tilde{H}_E$.

13.5.2 The Generalized Eigenvector and Schur Vector Methods

The deflating subspace methods are generalizations of the invariant subspace methods in the sense that the solutions of the Riccati equations are now computed by finding the bases for the stable deflating subspaces of certain matrix pencils rather than finding those of the Hamiltonian and the symplectic matrices. As of the invariant subspace methods, it will be assumed that for solving the CARE (DARE) with deflating subspace methods, the pair (A, B) is stabilizable (discrete stabilizable) and the associated Hamiltonian (symplectic) matrix pencil does not have an imaginary eigenvalue (an eigenvalue on the unit circle).

For the CARE, the pencil is $P_{\text{CARE}} - \lambda N_{\text{CARE}}$, where

$$P_{\text{CARE}} = \begin{pmatrix} A & -S \\ -Q & -A^{\text{T}} \end{pmatrix}, \qquad N_{\text{CARE}} = \begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix}. \tag{13.5.1}$$

For the DARE, the pencil is $P_{\text{DARE}} - \lambda N_{\text{DARE}}$, where

$$P_{\text{DARE}} = \begin{pmatrix} A & 0 \\ -Q & I \end{pmatrix}, \qquad N_{\text{DARE}} = \begin{pmatrix} I & S \\ 0 & A^{\text{T}} \end{pmatrix}. \tag{13.5.2}$$

Since no inversion of A is required to form the above pencils, **this generalization is significant for the DARE,** because, as we have seen, the eigenvector and the Schur methods cannot be applied to the DARE when A is singular.

As in the case of an invariant subspace method, a basis for a deflating subspace of a pencil can be constructed either by using the generalized eigendecomposition or the generalized Schur decomposition of the pencil. As before, an eigenvector method will have numerical difficulties in case the pencil has a multiple or near-multiple eigenvalue. **We will thus skip the descriptions of the generalized eigenvector methods and describe here only the generalized Schur method for the DARE.** We leave the description of the generalized Schur method for the CARE as an exercise **(Exercise 13.18)**.

The following results form a mathematical foundation for a deflating subspace method for the DARE. The results are due to Pappas *et al.* (1980).

Theorem 13.5.5. *Suppose that (A, B) is discrete-stabilizable and (A, Q) is discrete-detectable. Then the symplectic pencil $P_{\text{DARE}} - \lambda N_{\text{DARE}}$ does not have any eigenvalue λ with $|\lambda| = 1$.*

Proof. The proof is by contradiction.

Let $|\lambda| = 1$ be an eigenvalue of the pencil $P_{\text{DARE}} - \lambda N_{\text{DARE}}$ with the eigenvector

$$z = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} \neq 0.$$

Then we can write:

$$\begin{pmatrix} A & 0 \\ -Q & I \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \lambda \begin{pmatrix} I & S \\ 0 & A^{\mathrm{T}} \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \end{pmatrix}.$$

This means that

$$Az_1 = \lambda z_1 + \lambda S z_2; \tag{13.5.3}$$

$$-Qz_1 + z_2 = \lambda A^{\mathrm{T}} z_2. \tag{13.5.4}$$

Premultiplying the first equation by $\bar{\lambda} z_2^*$ and postmultiplying the conjugate transpose of the second by z_1, we have

$$\bar{\lambda} z_2^* A z_1 = |\lambda|^2 z_2^* z_1 + |\lambda|^2 z_2^* S z_2 \tag{13.5.5}$$

and

$$z_2^* z_1 = z_1^* Q z_1 + \bar{\lambda} z_2^* A z_1. \tag{13.5.6}$$

Substituting (13.5.5) into (13.5.6), we obtain

$$z_2^* z_1 = z_1^* Q z_1 + |\lambda|^2 z_2^* z_1 + |\lambda|^2 z_2^* S z_2 \tag{13.5.7}$$

or

$$z_2^* S z_2 + z_1^* Q z_1 = 0 \ \text{(since } |\lambda|^2 = 1). \tag{13.5.8}$$

Since $S = BR^{-1}B^{\mathrm{T}}$, Eq. (13.5.8) can be written as:

$$(z_2^* B) R^{-1} (B^{\mathrm{T}} z_2) + z_1^* Q z_1 = 0. \tag{13.5.9}$$

Since R is positive definite, this implies that

$$B^{\mathrm{T}} z_2 = 0 \quad \text{and} \quad Q z_1 = 0. \tag{13.5.10}$$

Therefore, from (13.5.3) and (13.5.4), we have $Az_1 = \lambda z_1$ and $A^{\mathrm{T}} z_2 = (1/\lambda) z_2$. (Note that since $|\lambda| = 1$, $\lambda \neq 0$).

Thus, from (13.5.10) and from the last equation, we have $z_2^* B = 0$ and $z_2^* A = (1/\bar{\lambda}) z_2^*$.

This means that for any F, $z_2^*(A - BF) = (1/\bar{\lambda}) z_2^*$, that is, $(1/\bar{\lambda})$ is an eigenvalue of $A - BF$ for every F. Since (A, B) is discrete-stabilizable, this means that $z_2 = 0$. Similarly, since (A, Q) is detectable, it can be shown **(Exercise 13.17)** that $z_1 = 0$. Therefore,

$$z = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix}$$

is a zero vector, which is a contradiction. ■

Theorem 13.5.5, together with the fact that if $\lambda \neq 0$ is an eigenvalue with multiplicity r of the pencil $P_{\text{DARE}} - \lambda N_{\text{DARE}}$, so is $1/\bar{\lambda}$ with the same multiplicity, allows us to state the following theorem:

Theorem 13.5.6. *Suppose that (A, B) is discrete-stabilizable and (A, Q) is discrete-detectable. Let $\lambda = 0$ be an eigenvalue of multiplicity r. Then the eigenvalues of the pencil $P_{\text{DARE}} - \lambda N_{\text{DARE}}$ can be arranged as follows (adopting the convention that the reciprocal of a zero is infinity):*

$$\underbrace{0, \ldots, 0}_{r};\quad \underbrace{\lambda_{r+1}, \ldots, \lambda_n}_{n-r};\quad \underbrace{\frac{1}{\lambda_n}, \ldots, \frac{1}{\lambda_{r+1}}}_{n-r};\quad \underbrace{\infty, \infty, \ldots, \infty}_{r}.$$

with $0 < |\lambda_i| < 1, i = r + 1, \ldots, n$.

MATCONTROL note: The generalized eigenvector method for the DARE has been implemented in MATCONTROL function **ricgeigd**.

The Generalized Schur-Vector Method for the DARE

Assume that the generalized Schur form of the pencil $P_{\text{DARE}} - \lambda N_{\text{DARE}}$ has been ordered such that the generalized eigenvalues of the pencil with moduli less than 1 can be obtained from the first quarters of the matrices, that is, the orthogonal matrices Q' and Z have been computed such that:

$$Q'(P_{\text{DARE}} - \lambda N_{\text{DARE}})Z = P_1 = \begin{pmatrix} P_{11} & P_{12} \\ 0 & P_{22} \end{pmatrix}$$

and

$$Q'(P_{\text{DARE}} - \lambda N_{\text{DARE}})Z = N_1 = \begin{pmatrix} N_{11} & N_{12} \\ 0 & N_{22} \end{pmatrix}$$

and the generalized eigenvalues of the pencil $P_{11} - \lambda N_{11}$ have modulii less than 1 (see below for details of how to do this).

Let

$$Z = \begin{pmatrix} Z_{11} & Z_{12} \\ Z_{21} & Z_{22} \end{pmatrix}.$$

Then the columns of

$$\begin{pmatrix} Z_{11} \\ Z_{21} \end{pmatrix}$$

form a basis for the discrete stable (deflating) subspace and the matrix $X = Z_{21}Z_{11}^{-1}$ is a unique symmetric positive semidefinite stabilizing solution of the DARE. We leave the details as an exercise **(Exercise 13.18)**.

Algorithm 13.5.2. *The Generalized Schur Algorithm for the DARE*

Inputs.

A—An $n \times n$ matrix

B—An $n \times m$ matrix

Q—An $n \times n$ symmetric matrix

R—An $m \times m$ symmetric matrix

Output. *X—The unique stabilizing solution of the DARE:* $A^TXA + Q - X - A^TXB(R + B^TXB)^{-1}B^TXA = 0$. **Step 1.** *Form* $P_{\text{DARE}} = \begin{pmatrix} A & 0 \\ -Q & I \end{pmatrix}$, $N_{\text{DARE}} = \begin{pmatrix} I & S \\ 0 & A^T \end{pmatrix}$.

Step 2. *Transform the pencil* $P_{\text{DARE}} - \lambda N_{\text{DARE}}$ *to the* **generalized RSF** *using the QZ algorithm, that is, find orthogonal matrices* Q_1 *and* Z_1 *such that:*

$$Q_1 P_{\text{DARE}} Z_1 = P_1 = \begin{pmatrix} P_{11} & P_{12} \\ 0 & P_{22} \end{pmatrix}$$

and

$$Q_1 N_{\text{DARE}} Z_1 = N_1 = \begin{pmatrix} N_{11} & N_{12} \\ 0 & N_{22} \end{pmatrix},$$

where P_1 *is quasi-upper triangular and* N_1 *is upper triangular.*

Step 3. *Reorder the above generalized RSF by using an orthogonal transformation, so that the pencil* $P_{11} - \lambda N_{11}$ *has all its eigenvalues with moduli less than 1. That is, find orthogonal matrices* Q_2 *and* Z_2 *such that* $Q_2 Q_1 P_{\text{DARE}} Z_1 Z_2$ *is quasi-upper triangular and* $Q_2 Q_1 N_{\text{DARE}} Z_1 Z_2$ *is upper triangular, and moreover, the diagonal blocks corresponding to the eigenvalues with moduli less than 1 are in the upper left quarter of these matrices.*

Step 4. *Form*

$$Z = Z_1 Z_2 = \begin{pmatrix} Z_{11} & Z_{12} \\ Z_{21} & Z_{22} \end{pmatrix}.$$

Step 5. *Compute* $X = Z_{21} Z_{11}^{-1}$, *that is, solve for X:* $XZ_{11} = Z_{21}$.

Example 13.5.2. Consider solving the DARE with

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}, \qquad B = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \qquad Q = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \qquad R = 1.$$

Step 1. $P_{\text{DARE}} = \begin{pmatrix} 1 & 2 & 0 & 0 \\ 3 & 4 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{pmatrix}$, $N_{\text{DARE}} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 3 \\ 0 & 0 & 2 & 4 \end{pmatrix}$.

Step 2. The generalized RSF of the pencil $P_{\text{DARE}} - \lambda N_{\text{DARE}}$ is given by: $Q_1(P_{\text{DARE}} - \lambda N_{\text{DARE}})Z_1 = P_1 - \lambda N_1$, where

$$P_1 = \begin{pmatrix} -5.5038 & 0.3093 & 0.7060 & 0.0488 \\ 0 & 1.4308 & 0.1222 & 0.0903 \\ 0 & 0 & 0.2665 & 0.2493 \\ 0 & 0 & 0 & 0.9530 \end{pmatrix},$$

$$N_1 = \begin{pmatrix} -0.9912 & -0.3540 & 0.2965 & -0.8012 \\ 0 & -0.2842 & 0.8565 & -0.5442 \\ 0 & 0 & -1.3416 & 0.9885 \\ 0 & 0 & 0 & 5.2920 \end{pmatrix}.$$

Step 3. The eigenvalues with moduli less than 1 are:

$$\frac{P_1(3,3)}{N_1(3,3)} = -0.1986 \quad \text{and} \quad \frac{P_1(4,4)}{N_1(4,4)} = 0.1801.$$

Step 4. The matrix

$$\begin{pmatrix} Z_{11} \\ Z_{21} \end{pmatrix}$$

is given by

$$\begin{pmatrix} Z_{11} \\ Z_{21} \end{pmatrix} = \begin{pmatrix} 0.5518 & -0.1074 \\ -0.3942 & 0.0847 \\ 0.6400 & 0.4499 \\ -0.3614 & 0.8825 \end{pmatrix}.$$

Step 5.

$$X = Z_{22}Z_{11}^{-1} = \begin{pmatrix} 54.9092 & 75.2247 \\ 75.2247 & 106.1970 \end{pmatrix}$$

is the stabilizing solution.

Implementational Details

The reduction to the generalized RSF can be achieved using the QZ algorithm, as described in Chapter 4.

Unfortunately, however, the eigenvalues might appear in any arbitrary order. Some reordering needs to be done. A systematic way to do this is as follows:

First, check if the last eigenvalue in the upper left quarter has modulus less than 1, if not, move it to the last position in the lower right quarter. Check the next eigenvalue now in the upper left quarter, if it does not have modulus less than 1, move it to the next position in the lower right quarter.

Note that each move is equivalent to finding a pair of orthogonal matrices such that pre- and postmultiplications by these matrices perform the necessary change.

The process can be continued until all the n eigenvalues with moduli greater than 1 have been moved to the lower right quarter and the upper left quarter contains only the eigenvalues with moduli less than 1.

There is also a slightly more efficient algorithm (Sima 1996, pp. 262–264) for ordering the eigenvalues of the pencil $P_{\text{DARE}} - \lambda N_{\text{DARE}}$.

There exists FORTRAN routines, developed by Van Dooren (1982) to compute deflating subspaces with specified spectrum. These subroutines are available as Algorithm 590–DSUBSP and EXCHQZ in ACM software library. Also, the LAPACK package (Anderson *et al.* 1999) includes the routine STGSEN, which performs a specified reordering of the eigenvalues of the generalized RSF.

Numerical stability and scaling: It can be shown (see Petkov *et al.* 1989) that the generalized Schur method may yield inaccurate results if the DARE is not properly scaled. For a scaling strategy that can be used to overcome this problem, see Gudmundsson *et al.* (1992) and Benner (1997).

The Generalized Schur Methods Without Explicit Computation of the Inverse of the Control Weighting Matrix R

All the methods we have considered so far require the explicit computation of the inverse of the control weighting matrix R. **These methods, therefore, may not yield accurate solutions when R is severely ill-conditioned.**

For example, consider the following example from Arnold and Laub (1984):

$$A = \begin{pmatrix} -0.1 & 0 \\ 0 & -0.02 \end{pmatrix}, \qquad B = \begin{pmatrix} 0.1 & 0 \\ 0.001 & 0.01 \end{pmatrix},$$
$$Q = \begin{pmatrix} 100 & 1000 \\ 1000 & 10{,}000 \end{pmatrix}, \qquad R = \begin{pmatrix} 1+\epsilon & 1 \\ 1 & 1 \end{pmatrix}.$$

The pair (A, B) is controllable. The matrix R becomes progressively ill-conditioned as $\epsilon \to 0$. The CARE with the above data was solved by Arnold and Laub, using RICPACK, **a software package especially designed for solving Riccati equations.** It was shown that the accuracy of the solution deteriorated as R became more and more ill-conditioned. For $\epsilon = 10^{-16}$, the relative accuracy was of order 10^{-1} only.

In this case, an **inverse-free generalized Schur method**, that avoids computations of R^{-1} is useful.

The Continuous-Time Case

First, we observe that the Hamiltonian eigenvalue problem $Hx = \lambda x$ associated with the CARE, can be replaced by the eigenvalue problem for the extended

$(2n + m) \times (2n + m)$ pencil:

$$P^{\mathrm{E}}_{\mathrm{CARE}} - \lambda N^{\mathrm{E}}_{\mathrm{CARE}},$$

where $P^{\mathrm{E}}_{\mathrm{CARE}} = \begin{pmatrix} A & 0 & B \\ -Q & -A^{\mathrm{T}} & 0 \\ 0 & B^{\mathrm{T}} & R \end{pmatrix}$, and $N^{\mathrm{E}}_{\mathrm{CARE}} = \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{pmatrix}$.

(Note that this pencil does not involve R^{-1}.) The solution of the CARE can now be obtained by constructing a basis of the stable deflating subspace of this pencil. It was further observed by Van Dooren (1981) that this $(2n+m) \times (2n+m)$ pencil can be compressed, using an orthogonal factorization of the matrix

$$\begin{pmatrix} R \\ B \end{pmatrix},$$

into a $2n \times 2n$ pencil, without affecting the deflating subspaces. Thus, if

$$\begin{pmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{pmatrix} \begin{pmatrix} R \\ B \end{pmatrix} = \begin{pmatrix} \hat{R} \\ 0 \end{pmatrix},$$

then instead of considering the $(2n + m) \times (2n + m)$ pencil $P^{\mathrm{E}}_{\mathrm{CARE}} - \lambda N^{\mathrm{E}}_{\mathrm{CARE}}$, we consider the $2n \times 2n$ compressed pencil $P^{\mathrm{EC}}_{\mathrm{CARE}} - \lambda N^{\mathrm{EC}}_{\mathrm{CARE}}$, where

$$P^{\mathrm{EC}}_{\mathrm{CARE}} = \begin{pmatrix} W_{22}A & W_{21}B^{\mathrm{T}} \\ -Q & -A^{\mathrm{T}} \end{pmatrix} \quad \text{and} \quad N^{\mathrm{EC}}_{\mathrm{CARE}} = \begin{pmatrix} W_{22} & 0 \\ 0 & I \end{pmatrix}.$$

This leads to the following algorithm:

Algorithm 13.5.3. *Inverse-Free Generalized Schur Algorithm for the CARE.*

Inputs.

A—An $n \times n$ matrix

B—An $n \times m$ matrix ($m \leq n$)

Q—An $n \times n$ symmetric matrix

R—An $m \times m$ symmetric matrix

Output.

X—The unique stabilizing solution of the CARE

Step 1. *Find the QR factorization of the matrix*

$$\begin{pmatrix} R \\ B \end{pmatrix} : W \begin{pmatrix} R \\ B \end{pmatrix} = \begin{pmatrix} \hat{R} \\ 0 \end{pmatrix}.$$

Partition

$$W = \begin{pmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{pmatrix},$$

where W_{22} is an $n \times n$ matrix.

Step 2. *Form $P^{\mathrm{EC}}_{\mathrm{CARE}}$ and $N^{\mathrm{EC}}_{\mathrm{CARE}}$ as shown above.*

Step 3. *Find the* **ordered generalized Schur form** *of the pencil $P^{\mathrm{EC}}_{\mathrm{CARE}} - \lambda N^{\mathrm{EC}}_{\mathrm{CARE}}$ using the QZ algorithm, that is, find orthogonal matrices Q_1 and Z*

such that $Q_1(P_{\text{CARE}}^{\text{EC}} - \lambda N_{\text{CARE}}^{\text{EC}})Z = \tilde{M} - \lambda\tilde{N}$; *where* $\tilde{M}$ *and* $\tilde{N}$ *are, respectively, quasi-upper and upper triangular matrices, and the n eigenvalues with negative real parts appear first.*

Step 4. *Compute* $X = Z_{21}Z_{11}^{-1}$, *where*

$$Z = \begin{pmatrix} Z_{11} & Z_{12} \\ Z_{21} & Z_{22} \end{pmatrix}.$$

Remark

- In his paper, Van Dooren (1981) described the compression technique by using an orthogonal factorization of the matrix

$$\begin{pmatrix} B \\ 0 \\ R \end{pmatrix}.$$

Instead, here we have used (an equivalent) factorization of $\begin{pmatrix} R \\ B \end{pmatrix}$ in the form $\begin{pmatrix} \hat{R} \\ 0 \end{pmatrix}$, so that a standard QR factorization algorithm can be used to achieve this factorization.

Example 13.5.3.

$$A = \begin{pmatrix} 2 & -1 \\ 1 & 0 \end{pmatrix}, \qquad B = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \qquad Q = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \qquad R = 10^{-10}.$$

Step 1.

$$W = \left(\begin{array}{c|cc} -0.0000 & -1.0000 & 0 \\ \hline -1.0000 & 0.0000 & 0 \\ 0 & 0 & 1.0000 \end{array}\right) = \left(\begin{array}{c|c} W_{11} & W_{12} \\ \hline W_{21} & W_{22} \end{array}\right).$$

Step 2.

$$P_{\text{CARE}}^{\text{EC}} = \begin{pmatrix} 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 \\ -1 & 0 & -2 & -1 \\ 0 & -1 & 1 & 0 \end{pmatrix}, \qquad N_{\text{CARE}}^{\text{EC}} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Step 3.

$$Z = \begin{pmatrix} -1.0000 - 0.0028i & 0.0000 + 0.0000i & -0.0000 + 0.0000i & -0.0000 + 0.0000i \\ 0.0000 + 0.0000i & 0.7071 + 0.0025i & -0.7071 + 0.0044i & 0.0000 - 0.0000i \\ -0.0000 - 0.0000i & 0.0000 + 0.0000i & 0.0000 - 0.0000i & 1.0000 - 0.0000i \\ 0.0000 + 0.0000i & 0.7071 + 0.0025i & 0.7071 - 0.0044i & -0.0000 + 0.0000i \end{pmatrix}.$$

Step 4.

$$X(\textbf{in Long Format}) = \begin{pmatrix} 0.00001000030018 & 0.00000999990018 \\ 0.00000999990018 & 1.00001000029721 \end{pmatrix}.$$

Verify: The residual norm= 7.357×10^{-8}.

The Discrete-Time Case

The discrete problem is analogous. Here we consider the $(2n + m) \times (2n + m)$ pencil $P^{\mathrm{E}}_{\mathrm{DARE}} - \lambda N^{\mathrm{E}}_{\mathrm{DARE}}$, where

$$P^{\mathrm{E}}_{\mathrm{DARE}} = \begin{pmatrix} A & 0 & -B \\ -Q & -I & 0 \\ 0 & 0 & R \end{pmatrix}, \quad \text{and} \quad N^{\mathrm{E}}_{\mathrm{DARE}} = \begin{pmatrix} I & 0 & 0 \\ 0 & A^{\mathrm{T}} & 0 \\ 0 & B^{\mathrm{T}} & 0 \end{pmatrix}.$$

This pencil is then compressed into the $2n \times 2n$ pencil $P^{\mathrm{EC}}_{\mathrm{DARE}} - \lambda N^{\mathrm{EC}}_{\mathrm{DARE}}$, where

$$P^{\mathrm{EC}}_{\mathrm{DARE}} = \begin{pmatrix} W_{22}A & 0 \\ -Q & -I \end{pmatrix}, \quad \text{and} \quad N^{\mathrm{EC}}_{\mathrm{DARE}} = \begin{pmatrix} W_{22} & W_{21}B^{\mathrm{T}} \\ 0 & A^{\mathrm{T}} \end{pmatrix},$$

by taking the QR factorization of the matrix $\begin{pmatrix} R \\ -B \end{pmatrix}$:

$$W \begin{pmatrix} R \\ -B \end{pmatrix} = \begin{pmatrix} \tilde{R} \\ 0 \end{pmatrix}, \quad \text{where } W = \begin{pmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{pmatrix}.$$

This leads to the following algorithm:

Algorithm 13.5.4. *Inverse-free Generalized Schur Method for the DARE.*

Inputs.

A—An $n \times n$ matrix

B—An $n \times m$ matrix ($m \leq n$)

Q—An $n \times n$ symmetric matrix

R—An $m \times m$ symmetric matrix.

Output.

X—The unique stabilizing solution of the DARE.

Step 1. *Find the QR factorization of* $\begin{pmatrix} R \\ -B \end{pmatrix}$, *that is, find an orthogonal matrix W such that*

$$W \begin{pmatrix} R \\ -B \end{pmatrix} = \begin{pmatrix} \tilde{R} \\ 0 \end{pmatrix}.$$

Partition

$$W = \begin{pmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{pmatrix}.$$

Step 2. *Form* $P^{\mathrm{EC}}_{\mathrm{DARE}} = \begin{pmatrix} W_{22}A & 0 \\ -Q & -I \end{pmatrix}$, $N^{\mathrm{EC}}_{\mathrm{DARE}} = \begin{pmatrix} W_{22} & W_{21}B^{\mathrm{T}} \\ 0 & A^{\mathrm{T}} \end{pmatrix}$.

Step 3. *Compute the* **ordered generalized Schur form** *of the pencil* $P^{EC}_{DARE} - \lambda N^{EC}_{DARE}$, *using the QZ algorithm followed by some ordering procedure so that the eigenvalues of moduli less than 1 appear in the first quarter, that is, find orthogonal matrices* Q_1 *and* Z *such that* $Q_1(P^{EC}_{DARE} - \lambda N^{EC}_{DARE})Z = \tilde{P} - \lambda\tilde{N}$ *and the n eigenvalues with moduli less than 1 appear first.*

Step 4. *Form* $X = Z_{21}Z_{11}^{-1}$, *where*

$$Z = \begin{pmatrix} Z_{11} & Z_{12} \\ Z_{21} & Z_{22} \end{pmatrix}.$$

Example 13.5.4. Consider solving the DARE with **a singular matrix** A:

$$A = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad Q = \begin{pmatrix} 1 & 2 \\ 2 & 4 \end{pmatrix}, \quad R = 1.$$

Step 1.

$$W = \left(\begin{array}{c|cc} -0.7071 & 0 & 0.7071 \\ \hline 0 & 1.0000 & 0 \\ 0.7071 & 0 & 0.7071 \end{array}\right) = \left(\begin{array}{c|c} W_{11} & W_{12} \\ \hline W_{21} & W_{22} \end{array}\right).$$

Step 2.

$$P^{EC}_{DARE} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & -2 & -1 & 0 \\ -2 & -4 & 0 & -1 \end{pmatrix},$$

$$N^{EC}_{DARE} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.7071 & 0 & 0.7071 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

Step 3.

$$Z = \begin{pmatrix} 0.8615 & -0.2781 & 0.3731 & -0.2034 \\ -0.3290 & 0.3256 & 0.8231 & -0.3290 \\ 0.2034 & 0.3731 & 0.2781 & 0.8615 \\ 0.3290 & 0.8231 & -0.3256 & 0.9329 \end{pmatrix}.$$

Step 4.

$$X = \begin{pmatrix} 1.0000 & 2.0000 \\ 2.0000 & 4.5000 \end{pmatrix}.$$

Verify: The residual norm= 7.772×10^{-16}.

MATLAB note: MATLAB functions **care** and **dare** solve the CARE and DARE, respectively, using generalized Schur methods, when R is well-conditioned and inverse free methods when R is ill-conditioned or singular.

13.5.3 The Matrix Sign Function Methods

Let A be an $n \times n$ matrix with **no zero** or **purely imaginary eigenvalues**. Let

$$J = X^{-1}AX = D + N,$$

be the Jordan canonical form (JCF) of A, where $D = \text{diag}(d_1, \ldots, d_n)$ and N is nilpotent and commutes with D. Then the matrix sign function of A is defined as: $\text{Sign}(A) = X \,\text{diag}\,(\text{sign}(d_1), \text{sign}(d_2), \ldots, \text{sign}(d_n))\, X^{-1}$, where

$$\text{sign}(d_i) = \begin{cases} 1 & \text{if } \text{Re}(d_i) > 0, \\ -1 & \text{if } \text{Re}(d_i) < 0. \end{cases}$$

Some important properties of $\text{Sign}(A)$ **are (Exercise 13.16)**:

1. $\text{Sign}(A)$ has the same stable invariant subspace as A.
2. The eigenvalues of $\text{Sign}(A)$ are ± 1, depending upon the sign of the corresponding eigenvalues of A.
3. The range of $\text{Sign}(A) - I$ is the stable invariant subspace of A.
4. The eigenvectors of $\text{Sign}(A)$ are the eigenvectors and principal vectors of A.
5. $\text{Sign}(TAT^{-1}) = T\text{Sign}(A)T^{-1}$.

We will now show how sign function can be used to solve the CARE and DARE. Before doing so, let's first describe an algorithm for computing $\text{Sign}(A)$.

The basic sign function algorithm is:

$$Z_0 = A,$$
$$Z_{k+1} = \tfrac{1}{2}\left(Z_k + Z_k^{-1}\right), \quad k = 0, 1, \ldots$$

It can be shown that the sequence $\{Z_k\}$ converges to $\text{Sign}(A)$ quadratically.

The initial convergence can, however, be very slow. Byers (1987) has shown that the convergence can be accelerated if Z_k is scaled by $|\det(Z_k)|^{1/n}$. For a discussion of scaling, see Kenney and Laub (1992).

Thus, a **practical algorithm** for computing $\text{Sign}(A)$ is:

Algorithm 13.5.5. *Computing Sign(A)*
Input. *An $n \times n$ matrix A.*
Output. *Sign(A), the matrix sign function of A.*
Step 1. *Set $Z_0 = A$.*
Step 2. *For $k = 0, 1, 2, \ldots$, do until convergence*
Compute $c = |\det Z_k|^{1/n}$.
Compute $Z_{k+1} = (1/2c)\left(Z_k + c^2 Z_k^{-1}\right)$.
End

Stopping criteria: The algorithm can be terminated if

- the norm of the difference between two successive iterates is small enough or
- the number of iterations exceeds the maximum number prescribed.

The Matrix Sign Function Method for the CARE

The mathematical basis for the matrix sign function method for the CARE is the following theorem.

Theorem 13.5.7. *Roberts (1971). Let H be the Hamiltonian matrix* (13.2.1) *associated with the CARE: $XA + A^{\mathrm{T}}X + Q - XSX = 0$.*

Let (A, B) be stabilizable and let (A, Q) be detectable.

Let

$$\operatorname{Sign}(H) = \begin{pmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{pmatrix},$$

where W_{ij} are $n \times n$ real matrices.

Then a stabilizing solution X of the CARE is a solution of the following ***overdetermined*** *consistent linear systems:*

$$\begin{pmatrix} W_{12} \\ W_{22} + I \end{pmatrix} X = -\begin{pmatrix} W_{11} + I \\ W_{21} \end{pmatrix}.$$

Proof. Define

$$T = \begin{pmatrix} I & Y \\ 0 & I \end{pmatrix} \begin{pmatrix} I & 0 \\ -X & I \end{pmatrix} = \begin{pmatrix} I - YX & Y \\ -X & I \end{pmatrix},$$

where Y satisfies

$$(A - SX)Y + Y(A - SX)^{\mathrm{T}} = -S.$$

An easy computation then shows that

$$THT^{-1} = \begin{pmatrix} A - SX & 0 \\ 0 & -(A - SX)^{\mathrm{T}} \end{pmatrix}.$$

Note that $T^{-1} = \begin{pmatrix} I & -Y \\ X & I - XY \end{pmatrix}$.

Then, using Property 5 of the sign function matrix, we obtain

$$\begin{aligned}\text{Sign}(H) &= \mathrm{T}^{-1}\text{Sign}\begin{pmatrix} A-SX & 0 \\ 0 & -(A-SX)^{\mathrm{T}}\end{pmatrix}\mathrm{T}, \\ &= \mathrm{T}^{-1}\begin{pmatrix} -I & 0 \\ 0 & I\end{pmatrix}\mathrm{T} \text{ (since } A-SX \text{ is asymptotically stable)}, \\ &= \begin{pmatrix} 2YX-I & -2Y \\ 2XYX-2X & I-2XY\end{pmatrix}.\end{aligned}$$

Thus, $\text{Sign}(H)+I_{2n} = \begin{pmatrix} 2YX & -2Y \\ 2XYX-2X & 2I-2XY\end{pmatrix}$

or $\begin{pmatrix} W_{11}+I & W_{12} \\ W_{21} & W_{22}+I\end{pmatrix} = \left(\begin{pmatrix} 2Y \\ 2(XY-I)\end{pmatrix}X, -\begin{pmatrix} 2Y \\ 2(XY-I)\end{pmatrix}\right).$

Now comparing both sides of the equation, we see that X must satisfy:

$$\begin{pmatrix} W_{12} \\ W_{22}+I\end{pmatrix}X = -\begin{pmatrix} W_{11}+I \\ W_{21}\end{pmatrix}. \quad \blacksquare$$

Symmetric Version of the Matrix Sign Function Algorithm

Theorem 13.5.7 yields a computational method to solve the CARE. However, the convergence can be painfully slow. The method can be made more efficient by using the following trick (Bierman 1984; Byers 1987) in which one works only with symmetric matrices.

Define

$$W_0 = JH = \begin{pmatrix} 0 & I \\ -I & 0\end{pmatrix}\begin{pmatrix} A & -S \\ -Q & -A^{\mathrm{T}}\end{pmatrix} = \begin{pmatrix} -Q & -A^{\mathrm{T}} \\ -A & S\end{pmatrix}.$$

The matrix W_0 is symmetric.

Now compute $\text{Sign}(H)$ by performing the following iterations:

$$W_{k+1} = \frac{1}{2c_k}\left(W_k + c_k^2 J W_k^{-1} J\right), \quad k = 0, 1, 2, \ldots$$

Then each W_k is symmetric and $\lim_{k\to\infty} W_k = J\text{sign}(H)$.

The parameter c_k is chosen to enhance the rate of convergence, as before.

Let

$$J\ \mathrm{Sign}(H) = Y = \begin{pmatrix} Y_{11} & Y_{12} \\ Y_{21} & Y_{22} \end{pmatrix}.$$

Then

$$\mathrm{Sign}(H) = \begin{pmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{pmatrix} = J^{\mathrm{T}}Y,$$

The equation:

$$\begin{pmatrix} W_{12} \\ W_{22} + I \end{pmatrix} X = -\begin{pmatrix} W_{11} + I \\ W_{21} \end{pmatrix}$$

then becomes

$$\begin{pmatrix} Y_{22} \\ Y_{12} + I \end{pmatrix} X = \begin{pmatrix} I - Y_{21} \\ -Y_{11} \end{pmatrix}.$$

This leads to the following symmetric version of the matrix sign function algorithm for the CARE:

Algorithm 13.5.6. *The Matrix Sign Function Algorithm for the CARE.*

Inputs.

A—An $n \times n$ matrix

B—An $n \times m$ matrix

Q—An $n \times n$ symmetric matrix

R—An $m \times m$ symmetric matrix

ϵ—Error tolerance.

Output.

X—The unique stabilizing solution of the CARE:

$A^{\mathrm{T}}X + XA - XBR^{-1}B^{\mathrm{T}}X + Q = 0$

Step 1.

1.1 *Form* $S = BR^{-1}B^{\mathrm{T}}$

1.2 *Define* $J = \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix}$. *Form* $W = JH = \begin{pmatrix} -Q & -A^{\mathrm{T}} \\ -A & S \end{pmatrix}$.

Step 2. *For* $k = 1, 2, \ldots$ *do until convergence with the given tolerance* ϵ

$$c = |\det W|^{1/2n}$$

$$W = \frac{1}{2c}(W + c^2 J W^{-1} J),$$

Step 3. *Partition* $W = \begin{pmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{pmatrix}$, *where each* W_{ij} *is of order n.*

Step 4. *Form* $M = \begin{pmatrix} W_{22} \\ W_{12} + I_n \end{pmatrix}$, $N = \begin{pmatrix} I - W_{21} \\ -W_{11} \end{pmatrix}$.

Step 5. *Solve for* $X : MX = N$.

Example 13.5.5. Consider solving the CARE using Algorithm 13.5.6 with

$$A = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad Q = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad R = 1.$$

Step 1. $S = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$, $H = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & -1 & 0 \end{pmatrix}$,

$$W_0 = JH = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & -1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Step 2. $W_1 = \frac{1}{2}\left(W_0 + JW_0^{-1}J\right) = \begin{pmatrix} -1 & 0 & 0 & -0.5 \\ 0 & -1 & -0.5 & 0 \\ 0 & -0.5 & 0.5 & 0 \\ -0.5 & 0 & 0 & 0.5 \end{pmatrix}$,

$$c = |\det(W_1)|^{1/4} = 0.8660.$$

$$W_2 = \frac{1}{2c}(W_1 + c^2 JW_1^{-1}J) = \begin{pmatrix} -1.1547 & 0 & 0 & -0.5774 \\ 0 & -1.1548 & -0.5774 & 0 \\ 0 & -0.5774 & 0.5774 & 0 \\ -0.5774 & 0 & 0 & 0.5774 \end{pmatrix}.$$

(Note that each W_i, $i = 0, 1, 2$ is symmetric.)

Step 3. $J\text{Sign}(H) = W_2 = W = \begin{pmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{pmatrix}$.

Step 5. $X = \begin{pmatrix} 1.7321 & 1 \\ 1 & 1.7321 \end{pmatrix}$.

Verify: The residual norm $= 9.9301 \times 10^{-16}$.

Example 13.5.6. Now consider solving the CARE using Algorithm 13.5.6 with the following data:

$$A = \begin{pmatrix} -1 & 1 & 1 \\ 0 & -2 & 0 \\ 0 & 0 & -3 \end{pmatrix}, \quad B = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \quad Q = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad R = 1.$$

Step 1.

$$W_0 = \begin{pmatrix} -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & -1 & 2 & 0 \\ 0 & 0 & -1 & -1 & 0 & 3 \\ 1 & -1 & -1 & 1 & 1 & 1 \\ 0 & 2 & 0 & 1 & 1 & 1 \\ 0 & 0 & 3 & 1 & 1 & 1 \end{pmatrix}.$$

Step 2. After five iterations, $\|W_5 - W_4\|/\|W_4\| = 6.4200 \times 10^{-15}$ (The readers are asked to verify this by carrying out 5 iterations).

Step 3.

$$W \equiv W_5.$$

Step 5.

$$X = \begin{pmatrix} 0.3732 & 0.0683 & 0.0620 \\ 0.0683 & 0.2563 & 0.0095 \\ 0.0620 & 0.0095 & 0.1770 \end{pmatrix}.$$

Verify: The residual norm $= 3.1602 \times 10^{-16}$.

Flop-count and stability: It can be shown that Algorithm 13.5.6 requires about $4n^3$ flops per iteration. **The algorithm is not stable in general (Byers 1986b)**, unless used with an iterative refinement technique such as Newton's method (see **Section 13.5.4**).

MATCONTROL note: Algorithm 13.5.6 has been implemented in MATCONTROL function **ricsgnc**.

The Matrix Sign Function Method for the DARE

The matrix sign function method for the CARE described in the previous section can now be applied to solve the DARE by converting the symplectic matrix M to the Hamiltonian matrix H using the bilinear transformation:

$$H = (M + I)^{-1}(M - I).$$

Because A needs to be nonsingular, the method is not applicable if A is singular, and is not numerically effective when A is ill-conditioned.

Avoiding Explicit Inversion of A

The explicit inversion of A, however, may be avoided, by using the following simple trick (Gardiner and Laub 1986).

Write

$$M = N^{-1}P,$$

where

$$N = \begin{pmatrix} I & S \\ 0 & A^{\mathrm{T}} \end{pmatrix} \quad \text{and} \quad P = \begin{pmatrix} A & 0 \\ -Q & I \end{pmatrix}.$$

Then it can be shown that even if A is singular, the matrix $(P + N)$ is invertible and the matrix H can be expressed as $H = (P + N)^{-1}(P - N)$.

Algorithm 13.5.7. *The Matrix Sign Function Algorithm for the DARE.*

Inputs.

A—An $n \times n$ matrix

B—An $n \times m$ matrix

Q—An $n \times n$ symmetric matrix

R—An $m \times m$ symmetric matrix

Output.

X—The unique stabilizing solution X of the DARE:
$A^TXA - X + Q - A^TXB(R + B^TXB)^{-1}B^TXA = 0$.

Step 1. *Form* $S = BR^{-1}B^T$,

$$N = \begin{pmatrix} I & S \\ 0 & A^T \end{pmatrix}, \qquad P = \begin{pmatrix} A & 0 \\ -Q & I \end{pmatrix}.$$

Step 2. *Form* $H = (P + N)^{-1}(P - N)$.

Step 3. *Apply the matrix sign function algorithm for the CARE (Algorithm* 13.5.6*) with H in Step* 2.

Example 13.5.7. Consider solving the DARE using Algorithm 13.5.7 with

$$A = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \qquad B = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \qquad Q = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \qquad R = 1.$$

Step 1. $S = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$, $N = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$, $P = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{pmatrix}$.

Step 2. $H = \begin{pmatrix} -0.3333 & 0.6667 & -0.6667 & 0.6667 \\ -0.6667 & 0.3333 & 0.6667 & -0.6667 \\ -1.3333 & 0.6667 & 0.3333 & 0.6667 \\ 0.6667 & -1.3333 & -0.6667 & -0.3333 \end{pmatrix}$.

Step 3. $X = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}$.

Verify: The residual norm $= 6.7195 \times 10^{-16}$.

MATCONTROL note: Algorithm 13.5.7 has been implemented in MATCONTROL function **ricsgnd**.

13.5.4 Newton's Methods

Recall that the classical Newton's method for finding a root x of $f(x) = 0$ can be stated as follows:

- Choose x_0, an initial approximation to x.

- Generate a sequence of approximations $\{x_i\}$ defined by

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}, \quad i = 0, 1, 2, \ldots \tag{13.5.11}$$

Then, *whenever x_0 is chosen close enough to x*, the sequence $\{x_i\}$ converges to the root x and the convergence is *quadratic* if $f'(x) \neq 0$. Newton's methods for the CARE and DARE can similarly be developed.

Newton's Method for the CARE

Consider first the CARE: $XA + A^{\mathrm{T}}X - XBR^{-1}B^{\mathrm{T}}X + Q = 0$.

Starting from an initial approximate solution X_0, the computed solutions are iteratively refined until convergence occurs; this is done by solving a Lyapunov equation at each iteration. The way how the Lyapunov equations arise can be explained as follows. Write $X = X_0 + (X - X_0)$. Substituting this into the CARE, we have

$$\begin{aligned}(A - BR^{-1}B^{\mathrm{T}}X_0)^{\mathrm{T}}X + X(A - BR^{-1}B^{\mathrm{T}}X_0) \\ = -X_0BR^{-1}B^{\mathrm{T}}X_0 - Q + (X - X_0)BR^{-1}B^{\mathrm{T}}(X - X_0).\end{aligned}$$

Assuming that $X - X_0$ is small (i.e., the initial approximate solution is good), we can neglect the last term on the right-hand side of the above equation. Thus we obtain the following Lyapunov equation for the next approximation X_1:

$$(A - BR^{-1}B^{\mathrm{T}}X_0)^{\mathrm{T}}X_1 + X_1(A - BR^{-1}B^{\mathrm{T}}X_0) = -X_0BR^{-1}B^{\mathrm{T}}X_0 - Q.$$

Assuming that X_1 is a better approximation than X_0 (i.e., $\| X - X_1 \| < \| X - X_0 \|$), the process can be continued until the convergence occurs, if there is convergence.

The above discussion immediately suggests the following **Newton method for the CARE**: (Kleinman 1968):

Step 1. Choose an initial approximation X_0.

Step 2. Compute $\{X_k\}$ iteratively by solving the Lyapunov equation:

$$(A - SX_k)^{\mathrm{T}}X_{k+1} + X_{k+1}(A - SX_k) = -X_kSX_k - Q, \quad k = 0, 1, 2, \ldots,$$

where $S = BR^{-1}B^{\mathrm{T}}$.

Step 3. Continue until and if convergence occurs.

Newton's method, as stated above, is not in the familiar form. However, the above steps can be easily reorganized to obtain Newton's method in the familiar form (see Benner (1997), Hammarling (1982) and Lancaster and Rodman (1995) for details).

To do this, let's define

$$R_{\mathrm{C}}(X) = XA + A^{\mathrm{T}}X - XSX + Q,$$

where $S = BR^{-1}B^{\mathrm{T}}$.

Now, the Fréchet derivative of $R_C(X)$ is given by

$$R'_X(Z) :\equiv (A - SX)^T Z + Z(A - SX).$$

Thus, Newton's method for $R_C(X) = 0$ is

$$R'_{X_i}(\Delta_i) + R_C(X_i) = 0, \quad i = 0, 1, 2, \ldots$$

$$X_{i+1} = X_i + \Delta_i.$$

The above observation leads to the following Newton algorithm for the CARE.

Algorithm 13.5.8. *Newton's Method for the CARE*

Inputs.

A—An $n \times n$ matrix

B—An $n \times m$ matrix

Q—An $n \times n$ symmetric matrix

R—An $m \times m$ symmetric matrix

Output. *The set $\{X_k\}$ converging to an approximate stabilizing solution matrix X of the CARE.*

Assumptions. *(A, B) is stabilizable, $R > 0$ and the CARE has a stabilizing solution X, and is unique.*

Step 1. *Set $S = BR^{-1}B^T$.*

Step 2. *Choose an initial approximate solution $X_0 = X_0^T$ such that $A - SX_0$ is stable.*

Step 3. *Construct the sequence of solutions $\{X_i\}$ as follows:*

For $i = 0, 1, 2, \ldots$ do until convergence occurs

3.1. *Compute $A_i = A - SX_i$*

3.2. *Compute $R_C(X_i) = A^T X_i + X_i A + Q - X_i S X_i$*

3.3. *Solve the Lyapunov equation for Δ_i: $A_i^T \Delta_i + \Delta_i A_i + R_C(X_i) = 0$.*

3.4. *Compute $X_{i+1} = X_i + \Delta_i$.*

End

Remark

- The above form of Newton's method is usually known as **Newton's Method in incremental form.** This form has some computational advantages over that presented in the beginning of this section in the sense that, in general, more accurate answers can be expected. This is because, in the incremental form algorithm, we solve the Lyapunov equation for the increment Δ_i and not for the solution directly and therefore, the solution X_i will have more correct digits.

The proof of the following theorem can be found in Lancaster and Rodman (1995, pp. 232–233). It gives conditions under which the above iterates converge.

Theorem 13.5.8. *Convergence of Newton's Method for the CARE. Let the assumptions for Algorithm* 13.5.8 *hold. Let X_0 be an approximate stabilizing solution and let X be a unique stabilizing solution X of the CARE. Then the matrices A_i and X_i, $i = 0, 1, \ldots$, constructed by the above algorithm are such that*

(i) *All A_i are stable; that is, all iterates X_i are stabilizing.*
(ii) $X \leq \cdots \leq X_{i+1} \leq X_i \leq \cdots \leq X_1$.
(iii) $\text{Lim}_{i\to\infty} X_i = X$, *where X is the unique symmetric positive-semidefinite stabilizing solution of the CARE.*
(iv) *There exists a constant $c > 0$ such that $\|X_{i+1} - X\| \leq c\|X_i - X\|^2$, for $i \geq 1$; that is, the sequence $\{X_i\}$ converges quadratically.*

Stopping criterion: The following can be used as a stopping criterion. Stop the iteration if

I. for a certain value of k and the prescribed *tolerance* ϵ

$$\frac{\|X_{k+1} - X_k\|_F}{\|X_k\|_F} \leq \epsilon,$$

or

II. the number of iterations k exceeds a prescribed number N.

If a condition-number estimator for the CARE is available, then Criterion I can be replaced by the following more appropriate stopping criterion: Stop the iteration if

$$\frac{\|X_{k+1} - X_k\|_F}{\|X_k\|_F} \leq \mu\kappa^E_{\text{CARE}},$$

where κ^E_{CARE} denotes an estimate of the κ_{CARE} and μ is the machine precision.

Example 13.5.8. Consider solving the CARE using Newton's method (**Algorithm 13.5.8**) with

$$A = \begin{pmatrix} -1 & 1 & 1 \\ 0 & -2 & 0 \\ 0 & 0 & -3 \end{pmatrix}, \qquad B = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \qquad Q = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \qquad R = 1.$$

Step 1. $S = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$.

Step 2. $X_0 = \begin{pmatrix} 0.4 & 0.1 & 0.1 \\ 0.1 & 0.3 & 0.0 \\ 0.1 & 0 & 0.2 \end{pmatrix}$

Step 3.

$$i = 0$$

$$\Delta_0 = \begin{pmatrix} -0.0248 & -0.0302 & -0.0369 \\ -0.0302 & -0.0426 & 0.0103 \\ -0.0369 & 0.0103 & -0.0224 \end{pmatrix},$$

$$X_1 = X_0 + \Delta_0 = \begin{pmatrix} 0.3752 & 0.0698 & 0.0631 \\ 0.0698 & 0.2574 & 0.0103 \\ 0.0631 & 0.0103 & 0.1776 \end{pmatrix}.$$

Relative Change: $\dfrac{\|X_1 - X_0\|}{\|X_0\|} = 0.1465.$

$$i = 1.$$

$$\Delta_1 = \begin{pmatrix} -0.0020 & -0.0015 & -0.0010 \\ -0.0015 & -0.0011 & -0.0008 \\ -0.0010 & -0.0008 & -0.0005 \end{pmatrix},$$

$$X_2 = X_1 + \Delta_1 = \begin{pmatrix} 0.3732 & 0.0683 & 0.0620 \\ 0.0683 & 0.2563 & 0.0095 \\ 0.0620 & 0.0095 & 0.1770 \end{pmatrix}.$$

Relative Change: $\dfrac{\|X_2 - X_1\|}{\|X_1\|} = 0.0086.$

$$i = 2$$

$$\Delta_2 = 10^{-5} \begin{pmatrix} -0.4561 & -0.3864 & -0.2402 \\ -0.3864 & -0.3311 & -0.2034 \\ -0.2402 & -0.2034 & -0.1265 \end{pmatrix},$$

$$X_3 = X_2 + \Delta_2 = \begin{pmatrix} 0.3732 & 0.0683 & 0.0620 \\ 0.0683 & 0.2563 & 0.0095 \\ 0.0620 & 0.0095 & 0.1770 \end{pmatrix}.$$

Relative Change: $\dfrac{\|X_3 - X_2\|}{\|X_2\|} = 2.1709 \times 10^{-5}.$

MATHCONTROL note: Algorithm 13.5.8 has been implemented in MATCONTROL function **ricnwtnc**.

Convergence: We know that there exist infinitely many X_0 for which $A - SX_0$ is stable. *The choice of proper X_0 is crucial.* If the initial solution matrix X_0 is not close enough to the exact solution X, then, as in the case of scalar Newton's method, the convergence can be painfully slow. *The method might even converge*

to a nonstabilizing solution in the presence of round-off errors. Things might go wrong even at the first step. To see this, let's consider the following example from Kenney *et al.* (1990):

$$A = 0, \qquad B = Q = I, \qquad R = I.$$

The exact solution is $X = I$. Let $X_0 = \epsilon I$, where $\epsilon > 0$ is a small positive number. Then,

$$A - BB^{\mathrm{T}}X_0 = -\epsilon I$$

is stable for all $\epsilon > 0$ and the initial error is $\|X - X_0\| = 1 - \epsilon \cong 1$ for small ϵ. However,

$$X_1 = \frac{1+\epsilon^2}{2\epsilon} I \quad \text{and} \quad \|X - X_1\| \simeq \frac{1}{2\epsilon},$$

which is quite large. Thus, even though the errors at subsequent steps decrease, a large number of steps will be needed for the error made at the first step to damp out.

Some conditions guaranteeing convergence from the first step on have been given by Kenney *et al.* (1990). This is stated in the following Theorem (**assuming that $R = I_{m\times m}$**).

Theorem 13.5.9. *Let X_0 be an initial approximation such that $A - BB^{\mathrm{T}}X_0$ is stable and assume that $\|X - X_0\| < 1/(3\|B\|^2\|\Omega^{-1}\|)$, where $\Omega(Z) = (A - BB^{\mathrm{T}}X)^{\mathrm{T}}Z + Z(A - BB^{\mathrm{T}}X)$, then $\|X - X_1\| \leq \|X - X_0\|$, with equality only when $X_0 = X$.*

Flop-count: Newton's method is iterative; therefore, an exact flop count cannot be given. However, if the Schur method is used to solve the Lyapunov equations at each iteration, then about $40n^3$ flops are needed per iteration.

Stability: Since the principal computational task in Newton's method is the solution of a Lyapunov matrix equation at each iteration, **the method can be shown to be stable if a numerically stable method such as the Schur method is used to solve the Lyapunov equation.** Specifically, if $\hat{X}$ is the computed solution obtained by Newton's method, then it can be shown (Petkov *et al.* 1991) that

$$\frac{\|\hat{X} - X\|_{\mathrm{F}}}{\|X\|_{\mathrm{F}}} \leq \mu\kappa_{\mathrm{CARE}},$$

where κ_{care} is the condition number of the CARE. **That is, the method does not introduce more errors than what is already inherent in the problem.**

Modified Newton's Methods

Several modifications of Newton's methods for the AREs have been obtained in recent years (Benner 1990; Benner and Byers 1998; Guo 1998; Guo and Lancaster

1998; Guo and Laub 2000; etc.). We just state in the following the line search modification of Newton's method by Benner and Byers (1998).

Newton's Method with Line Search

The performance of Newton's method can be improved by using an optimization technique called **line search.**

The idea is to take a Newton step at each iteration in the direction so that $\|R_C(X_{i+1})\|_F^2$ is minimized. Thus the iteration:

$$X_{i+1} = X_i + \Delta_i$$

in Step 3 of Newton's method will be replaced by

$$X_{i+1} = X_i + t_i \Delta_i,$$

where t_i is a real scalar to be chosen so that $\|R_C(X_i + t_i\Delta_i)\|_F^2$ will be minimized.

This is equivalent to minimizing

$$f_i(t) = \text{Trace}(R_C(X_i + t\Delta_i)^T R_C(X_i + t\Delta_i)) = \text{Trace}(R_C(X_i + t\Delta_i)^2),$$
$$= \alpha_i(1-t)^2 - 2\beta_i(1-t)t^2 + \nu_i t^4,$$

where

$$\alpha_i = \text{Trace}(R_C(X_i)^2), \qquad \beta_i = \text{Trace}(R_C(X_i)V_i),$$
$$\nu_i = \text{Trace}(V_i^2), \quad V_i = \Delta_i S \Delta_i.$$

It can be shown (see Benner 1997; Benner and Byers 1998) that the function $f_i(t)$ has a local minimum at some value $t_i \in [0, 2]$.

We thus have the following modified Newton's algorithm.

Algorithm 13.5.9. *Newton's Method with Line Search for the CARE*

Inputs. *Same as in Algorithm* 13.5.8.

Output. *Same as in Algorithm* 13.5.8.

Assumptions. *Same as in Algorithm* 13.5.8.

Step 1. *Same as in Algorithm* 13.5.8.

Step 2. *Same as in Algorithm* 13.5.8.

Step 3. *For* $i = 0, 1, 2, \ldots$ *do until convergence occurs*

3.1 *Same as in Algorithm* 13.5.8.

3.2 *Same as in Algorithm* 13.5.8.

3.3 *Same as in Algorithm* 13.5.8.

3.4 *Compute* $V_i = \Delta_i S \Delta_i$

3.5 *Compute* α_i, β_i, *and* ν_i *of* f_i *as given above.*

Step 3.6 *Compute* $t_i \in [0, 2]$ *such that* $f_i(t_i) = \min_{t\in[0,2]} f_i(t)$.

Step 3.7 *Compute* $X_{i+1} = X_i + t_i\Delta_i$.

End.

Example 13.5.9. The input matrices A, B, Q, and R are the same as in Example 13.5.8.

Step 1. $S = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$.

Step 2. $X_0 = \begin{pmatrix} 0.4 & 0.1 & 0.1 \\ 0.1 & 0.3 & 0 \\ 0.1 & 0 & 0.2 \end{pmatrix}$.

Step 3. $i = 0$: $\Delta_0 = \begin{pmatrix} -0.0248 & -0.0302 & -0.0369 \\ -0.0302 & -0.0426 & 0.0103 \\ -0.0369 & 0.0103 & -0.0224 \end{pmatrix}$.

$\alpha_0 = 0.1761, \quad \beta_0 = -0.0049, \quad \gamma_0 = 2.1827 \times 10^{-4}, \quad t_0 = 1.0286.$

$$X_1 = X_0 + t_0\Delta_0 = \begin{pmatrix} 0.3745 & 0.0690 & 0.0620 \\ 0.0690 & 0.2562 & 0.0105 \\ 0.0620 & 0.0105 & 0.1770 \end{pmatrix}.$$

Relative change: $\|X_1 - X_0\|/\|X_0\| = 0.1507$.

$$\boldsymbol{i = 1}\text{: } \Delta_1 = \begin{pmatrix} -0.0012 & -0.0006 & 0.0000 \\ -0.0006 & 0.0001 & -0.0011 \\ 0.0000 & -0.0011 & 0.0001 \end{pmatrix}$$

$\alpha_1 = 8.9482 \times 10^{-5}, \quad \beta_1 = -4.2495 \times 10^{-8}, \quad \gamma_1 = 4.9519 \times 10^{-11}, \quad t_1 = 1.0005.$

$$X_2 = X_1 + t_1\Delta_1 = \begin{pmatrix} 0.3732 & 0.0683 & 0.0620 \\ 0.0683 & 0.2563 & 0.0095 \\ 0.0620 & 0.0095 & 0.1770 \end{pmatrix}.$$

Relative change: $\|X_2 - X_1\|/\|X_1\| = 0.0038587$.

$$\boldsymbol{i = 2}\text{: } \Delta_2 = 10^{-6}\begin{pmatrix} -0.1677 & -0.4428 & -0.4062 \\ -0.4428 & -0.7620 & 0.1277 \\ -0.4062 & 0.1277 & -0.2505 \end{pmatrix}.$$

$\alpha_2 = -2.9393 \times 10^{-10}, \quad \beta_2 = -1.0425 \times 10^{-17}, \quad \gamma_2 = 6.1179 \times 10^{-24}, \quad t_2 = 1.0000.$

$$X_3 = X_2 + t_2\Delta_2 = \begin{pmatrix} 0.3732 & 0.0683 & 0.0620 \\ 0.0683 & 0.2563 & 0.0095 \\ 0.0620 & 0.0095 & 0.1770 \end{pmatrix}.$$

Relative change: $\|X_3 - X_2\|/\|X_2\| = 2.4025 \times 10^{-6}$.

$$\boldsymbol{i = 3}\text{: } \Delta_3 = 10^{-12}\begin{pmatrix} -0.1593 & -0.0972 & 0.0319 \\ -0.0972 & -0.0286 & -0.1791 \\ 0.00319 & -0.1791 & 0.0308 \end{pmatrix}.$$

$\alpha_3 = 2.4210 \times 10^{-24}$, $\beta_3 = -1.4550 \times 10^{-37}$, $\gamma_3 = 2.4612 \times 10^{-50}$, $t_3 = 1.0000$.

$$X_4 = X_3 + t_3\Delta_3 = \begin{pmatrix} 0.3732 & 0.0683 & 0.0620 \\ 0.0683 & 0.2563 & 0.0095 \\ 0.0620 & 0.0095 & 0.1770 \end{pmatrix}.$$

Relative change: $\|X_4 - X_3\|/\|X_3\| = 5.5392 \times 10^{-13}$.

Theorem 13.5.10. *Convergence of Newton's Method with Line Search for the CARE. If (A, B) is a controllable pair, and if the step sizes t_i are bounded away from zero, then Newton's method with the line search* **(Algorithm 13.5.9)** *converges to the stabilizing solution.*

Proof. See Benner and Byers (1998), Guo and Laub (2000). ■

Flop-count: Algorithm 13.5.9 is slightly more expensive (about 8% to the cost of one Newton step) than Algorithm 13.5.8. **However, one saves about one iteration step out of 15;** often much more, but seldom less.

MATCONTROL note: Algorithm 13.5.9 has been implemented in MATCONTROL function **ricnwlsc**.

Newton's method for the DARE

Newton's method for the DARE:

$$A^{\mathrm{T}}XA - X + Q - A^{\mathrm{T}}XB(R + B^{\mathrm{T}}XB)^{-1}B^{\mathrm{T}}XA = 0$$

is analogous. It is based on successive solutions of **Stein equations (discrete-time Lyapunov equations)** associated with the discrete-time system. We state the algorithm below without detailed discussions. The algorithm was originally developed by Hewer (1971). See also Kleinman (1974).

Algorithm 13.5.10. *Newton's Method for the DARE*

Inputs. *A—An $n \times n$ matrix*
B—An $n \times m$ matrix
Q—An $n \times n$ symmetric matrix
R—An $m \times m$ symmetric matrix

Output. *The set* $\{X_k\}$ *converging to the unique stabilizing solution* X *of the DARE:*

$$R_D(X) = A^T XA - X + Q - A^T XB(R + B^T XB)^{-1} B^T XA = 0.$$

Assumptions. *(i)* (A, B) *is discrete-stabilizable (ii)* $R \geq 0$, *(iii) A stabilizing solution* X *exists and is unique, and (iv)* $R + B^T XB > 0$.

Step 1. *Choose* $X_0 = X_0^T$ *such that* $A - B(R + B^T X_0 B)^{-1} B^T X_0 A$ *is a discrete-stable matrix, that is, it has all its eigenvalues inside the unit circle.*

Step 2. *For* $i = 0, 1, 2, \ldots$ *do until convergence.*

2.1 *Compute* $K_i = (R + B^T X_i B)^{-1} B^T X_i A$

2.2 *Compute* $A_i = A - BK_i$

2.3 *Compute* $R_D(X_i) = A^T X_i A - X_i + Q - A^T X_i B(R + B^T X_i B)^{-1} B^T X_i A$

2.4 *Solve the discrete-time Lyapunov equation (Stein equation) for* Δ_i: $A_i^T \Delta_i A_i - \Delta_i + R_D(X_i) = 0$

2.5 *Compute* $X_{i+1} = X_i + \Delta_i$.

End

The following theorem gives conditions under which the sequence $\{X_i\}$ converges. The proof of this theorem can be found in Lancaster and Rodman (1995, pp. 308–310), in case R is nonsingular. See also Benner (1997), Mehrmann (1991).

Theorem 13.5.11. *Convergence of Newton's Method for the DARE. Let the assumptions of Algorithm* 13.5.10 *hold. Let* X_0 *be a stabilizing approximate solution of the DARE. Then the matrices* A_i *and* X_i, *constructed by the above algorithm, are such that*

(i) All A_i are discrete-stable,

(ii) $\lim_{i \to \infty} X_i = X$, where X is the unique symmetric positive semidefinite discrete-stabilizing solution of the DARE.

(iii) $X \leq \cdots \leq X_{i+1} \leq X_i \leq \cdots \leq X_1$

(iv) There exists a constant $c > 0$ such that $\|X_{i+1} - X\| \leq c\|X_i - X\|^2$, $i \geq 1$, that is, the sequence $\{X_i\}$ converges quadratically.

Stopping criterion: The same stopping criteria as in the case of Newton's method for the CARE can be used.

Example 13.5.10. Consider solving the DARE using Algorithm 13.5.10 with

$$A = \begin{pmatrix} -1 & 1 & 1 \\ 0 & -2 & 0 \\ 0 & 0 & -3 \end{pmatrix}, \quad B = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \quad R = 1, \quad Q = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Step 1. $X_0 = \begin{pmatrix} 1 & -5 & 10 \\ -5 & 1600 & -2000 \\ 10 & -2000 & 2700 \end{pmatrix}$.

Step 2. $i = 0$

The eigenvalues of $A - B(R + B^{\mathrm{T}} X_0 B)^{-1} B^{\mathrm{T}} X_0 A$ are $-0.8831 \pm j0.2910$, -0.0222. Then X_0 is a discrete-stabilizing approximate solution of the DARE.

$$K_0 = \begin{pmatrix} -0.0192 & 2.6154 & -6.8077 \end{pmatrix},$$

$$A_0 = \begin{pmatrix} -0.9808 & -1.6154 & 7.8077 \\ 0.0192 & -4.6154 & 6.8077 \\ 0.0192 & -2.6154 & 3.8077 \end{pmatrix},$$

$$X_1 = 10^4 \begin{pmatrix} 0.0008 & -0.0137 & 0.0167 \\ -0.0137 & 0.6808 & -0.9486 \\ 0.0165 & -0.9486 & 1.3364 \end{pmatrix}.$$

Relative change: $\dfrac{\|X_1 - X_0\|}{\|X_0\|} = 3.7654.$

$$i = 1.$$

$$K_1 = \begin{pmatrix} -0.0301 & 4.4699 & -9.5368 \end{pmatrix},$$

$$A_1 = \begin{pmatrix} -0.9699 & -3.4699 & 10.5368 \\ 0.0301 & -6.4699 & 9.5368 \\ 0.0301 & -4.4699 & 6.5368 \end{pmatrix},$$

$$X_2 = 10^3 \begin{pmatrix} 0.0067 & -0.0893 & 0.1029 \\ -0.0893 & 2.0297 & -2.5658 \\ 0.1029 & -2.5658 & 3.3125 \end{pmatrix}.$$

Relative change: $\dfrac{\|X_2 - X_1\|}{\|X_1\|} = 0.7364.$

$$i = 2.$$

$$K_2 = \begin{pmatrix} -0.0826 & 5.1737 & -10.2938 \end{pmatrix},$$

$$A_2 = \begin{pmatrix} -0.9174 & -4.1737 & 11.2938 \\ 0.0826 & -7.1737 & 10.2938 \\ 0.0826 & -5.1737 & 7.2938 \end{pmatrix},$$

$$X_3 = 10^3 \begin{pmatrix} 0.0054 & -0.0670 & 0.0767 \\ -0.0670 & 1.6234 & -2.0796 \\ 0.0767 & -2.0796 & 2.7283 \end{pmatrix}.$$

Relative change: $\dfrac{\|X_3 - X_2\|}{\|X_2\|} = 0.1862.$

The relative changes continue to decrease from this step onwards.

$$X_7 = 10^3 \begin{bmatrix} 0.0053 & -0.0658 & 0.0751 \\ -0.0658 & 1.5943 & -2.0428 \\ 0.0751 & -2.0428 & 2.6817 \end{bmatrix}.$$

For $i = 6$, relative change: $\|X_7 - X_6\| / \|X_6\|$ is 2.3723×10^{-15}.

MATCONTROL note: Algorithm 13.5.10 has been implemented in MATCONTROL function **ricnwtnd**.

Newton's Method with Line Search for the DARE

Algorithm 13.5.10 can be modified in a similar way as in case of the CARE to include the line search.

The function $f_i(t)$ to be minimized in this case is given by:

$$f_i(t) = \alpha_i(1-t)^2 - 2\beta_i(1-t)t^2 + \gamma_i t^4,$$

where $\alpha_i = \text{Trace}(R_d(X_i)^2)$, $\beta_i = \text{Trace}(R_d(X_i)V_i)$, $\gamma_i = \text{Trace}(V_i^2)$, and $V_i = A_i^T \triangle_i B(R + B^T X_i B)^{-1} B^T \triangle_i A_i$

For details, see Benner (1997).

Algorithm 13.5.11. *Newton's Method with Line Search for the DARE*

Inputs. *Same as in Algorithm* 13.5.10.

Output. *Same as in Algorithm* 13.5.10.

Assumptions. *Same as in Algorithm* 13.5.10.

Step 1. *Same as in Algorithm* 13.5.10.

Step 2. *For* $k = 0, 1, 2, \ldots$ *do until convergence*

2.1 *Same as in Algorithm* 13.5.10

2.2 *Same as in Algorithm* 13.5.10

2.3 *Same as in Algorithm* 13.5.10

2.4 *Same as in Algorithm* 13.5.10

2.5 *Compute* $S_i = B(R + B^T X_i B)^{-1} B^T$

2.6 *Compute* $V_i = A_i^T \triangle_i S_i \triangle_i A_i$

2.7 *Compute the coefficients* α_i, β_i, *and* γ_i *of* $f_i(t)$ *as above*

2.8 *Compute* $t_i \in [0, 2]$ *such that* $f_i(t_i) = \min_{t\in[0,2]} f_i(t)$

2.9 $X_{i+1} = X_i + t_i \triangle_i$.

End

Flop-count: The algorithm is again just slightly more expensive than Algorithm 13.5.10. The additional cost of forming V_i, the coefficients of f_i, a local minimizer t_i of f_i and scaling $\triangle_i$ by t_i is cheap as compared to $O(n^3)$ flops required for other computations.

Convergence: The line search procedure can sometimes significantly improve the convergence behavior of Newton's method. For details, see Benner (1997).

Example 13.5.11. Consider solving the DARE using Algorithm 13.5.11 with

$$A = \begin{pmatrix} -1 & 1 & 1 \\ 0 & -2 & 0 \\ 0 & 0 & -3 \end{pmatrix}, \qquad B = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \qquad Q = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

$$R = 1.$$

Step 1. $X_0 = \begin{pmatrix} 1 & -5 & 10 \\ -5 & 1600 & -2000 \\ 10 & -2000 & 2700 \end{pmatrix}$.

Step 2. $i = 0$, $\Delta_0 = 10^4 \begin{pmatrix} 0.0007 & -0.0132 & 0.0157 \\ -0.0132 & 0.5208 & -0.7486 \\ 0.0157 & -0.7486 & 1.0664 \end{pmatrix}$,

$\alpha_0 = 9.7240 \times 10^7$, $\beta_0 = 5.5267 \times 10^8$, $\gamma_0 = 3.1518 \times 10^9$, $t_0 = 0.3402$.

$X_1 = X_0 + t_0 D_0 = 10^3 \begin{pmatrix} 0.0034 & -0.0500 & 0.0635 \\ -0.0500 & 3.3718 & -4.5471 \\ 0.0635 & -4.5471 & 6.3283 \end{pmatrix}$.

Relative change: $\|X_1 - X_0\| / \|X_0\| = 1.2812$.

Step 3. $i = 1$, $\Delta_1 = 10^3 \begin{pmatrix} 0.0029 & -0.0405 & 0.0431 \\ -0.0405 & -1.1655 & 1.7233 \\ 0.0431 & 1.7233 & -2.6498 \end{pmatrix}$,

$\alpha_1 = 1.1123 \times 10^7$, $\beta_1 = 1.7963 \times 10^6$, $\gamma_1 = 3.0428 \times 10^5$, $t_1 = 0.8750$.

$X_2 = X_1 + t_1 \Delta_1 = 10^3 \begin{pmatrix} 0.0059 & -0.0854 & 0.1012 \\ -0.0854 & 2.3520 & -3.0392 \\ 0.1012 & -3.0392 & 4.0097 \end{pmatrix}$.

Relative change: $\|X_2 - X_1\| / \|X_1\| = 0.3438$.

$i = 2$, $\Delta_2 = 10^{-3} \begin{pmatrix} -0.0006 & 0.0196 & -0.0261 \\ 0.0196 & -0.7570 & 0.9955 \\ -0.0261 & 0.9955 & -1.3267 \end{pmatrix}$,

$\alpha_2 = 1.9251 \times 10^5$, $\beta_2 = -157.2798$, $\gamma_2 = 0.1551$, $t_2 = 1.0008$.

$X_3 = X_2 + t_2 \Delta_2 = 10^3 \begin{pmatrix} 0.0053 & -0.0658 & 0.0751 \\ -0.0658 & 1.5944 & -2.0429 \\ 0.0751 & -2.0429 & 2.6819 \end{pmatrix}$.

Relative change: $\|X_3 - X_2\|/\|X_2\| = 0.3283.$

$$i = 3, \qquad \Delta_3 = \begin{pmatrix} -0.0003 & 0.0024 & -0.0011 \\ 0.0024 & -0.0481 & 0.1094 \\ -0.0011 & 0.1094 & -0.2202 \end{pmatrix},$$

$$\alpha_3 = 0.0912, \quad \beta_3 = -2.8785 \times 10^{-5}, \quad \gamma_3 = 1.6525 \times 10^{-8}, \quad t_3 = 1.0003.$$

$$X_4 = X_3 + t_3\Delta_3 = 10^3 \begin{pmatrix} 0.0053 & -0.0658 & 0.0751 \\ -0.0658 & 1.5943 & -2.0428 \\ 0.0751 & -2.0428 & 2.6817 \end{pmatrix}.$$

Relative change: $\|X_4 - X_3\|/\|X_3\| = 6.4273 \times 10^{-5}.$

The relative changes continue to decrease after each iteration. For example, for $i = 5$, we have

Relative change: $\|X_6 - X_5\|/\|X_5\| = 1.0906 \times 10^{-13}$, and **Relative Residual** = 3.2312×10^{-11}.

MATCONTROL note: Algorithm 13.5.11 has been implemented in MATCONTROL function **ricnwlsd**.

Newton's Method as an Iterative Refinement Technique

Newton's method is often used as an **iterative refinement technique.** First, a direct robust method such as the Schur method or the matrix sign function method is applied to obtain an approximate solution and this approximate solution is then refined by using a few iterative steps of Newton's method. **For higher efficiency, Newton's method with the line search (Algorithm 13.5.9 for the CARE and Algorithm 13.5.11 for the DARE) should be preferred over Newton's method.**

13.6 THE SCHUR AND INVERSE-FREE GENERALIZED SCHUR METHODS FOR THE DESCRIPTOR RICCATI EQUATIONS

We have seen in Chapter 5 that several practical applications give rise to the descriptor systems:

$$E\dot{x}(t) = Ax(t) + Bu(t) \quad \text{(Continuous-time)}, \tag{13.6.1}$$

$$Ex_{k+1} = Ax_k + Bu_k \quad \text{(Discrete-time)}. \tag{13.6.2}$$

The AREs **associated** with the above systems, respectively, are:

$$A^{\mathrm{T}}XE + E^{\mathrm{T}}XA - E^{\mathrm{T}}XBR^{-1}B^{\mathrm{T}}XE + Q = 0, \tag{13.6.3}$$

and

$$E^{\mathrm{T}}XE = A^{\mathrm{T}}XA - A^{\mathrm{T}}XB(B^{\mathrm{T}}XB + R)^{-1}B^{\mathrm{T}}XA + Q. \tag{13.6.4}$$

The Riccati equations (13.6.3) and (13.6.4) will be, respectively, called as the **descriptor continuous-time algebraic Riccati equation** (DCARE) and the **descriptor discrete-time algebraic Riccati equation** (DDARE).

Most of the methods, such as the Schur method, the matrix sign function method, and Newton's method, can be easily extended to solve DCARE and DDARE.

Below we state how the generalized Schur methods and the inverse-free generalized Schur methods can be extended to solve these equations. The derivations of the others are left as **Exercises.** See Bender and Laub (1985, 1987), Benner (1997), Laub (1991), Mehrmann (1988), Benner *et al.* (1999a) etc. in this context. For descriptor discrete-time Lyapunov and Riccati equations, see Zhang *et al.* (1999).

13.6.1 The Generalized Schur Method for the DCARE

The matrix pencil associated with the DCARE is

$$P_{\mathrm{DCARE}} - \lambda N_{\mathrm{DCARE}} = \begin{pmatrix} A & -S \\ -Q & -A^{\mathrm{T}} \end{pmatrix} - \lambda \begin{pmatrix} E & O \\ O & E^{\mathrm{T}} \end{pmatrix},$$

where $S = BR^{-1}B^{\mathrm{T}}$.

The Schur method for the DCARE, then, can be easily developed by transforming the above pencil to the **Ordered** RSF using the QZ iteration algorithm (Chapter 4). Thus, if Q_1 and Z_1 are orthogonal matrices such that

$$Q_1 P_{\mathrm{DCARE}} Z_1 = \begin{pmatrix} L_{11} & L_{12} \\ O & L_{22} \end{pmatrix}, \qquad Q_1 N_{\mathrm{DCARE}} Z_1 = \begin{pmatrix} N_{11} & N_{12} \\ O & N_{22} \end{pmatrix},$$

where $Q_1 P_{\mathrm{DCARE}} Z_1$ is upper quasi-triangular, $Q_1 N_{\mathrm{DCARE}} Z_1$ is upper triangular, and $L_{11} - \lambda N_{11}$ is stable, then the columns of $\begin{pmatrix} Z_{11} \\ Z_{21} \end{pmatrix}$, where

$$Z_1 = \begin{pmatrix} Z_{11} & Z_{12} \\ Z_{21} & Z_{22} \end{pmatrix},$$

span the stable deflating subspace. So, the matrix $X = Z_{21}Z_{11}^{-1}$ is a solution of the DCARE.

MATLAB note: MATLAB function **care** in the form:

$$[X, L, G, rr] = \text{care}(A, B, Q, R, E)$$

solves the DCARE.

Here $G = R^{-1}(B^{\mathrm{T}}XE)$; the gain matrix, $L = \text{eig}(A - BG, E)$, and rr = the Frobenius norm of the relative residual matrix.

13.6.2 The Inverse-Free Generalized Schur Method for the DCARE

In case R is singular or nearly singular, one needs to use the inverse-free generalized Schur method. The extended pencil to be considered in this case is

$$\begin{pmatrix} A & 0 & B \\ -Q & -A^{\mathrm{T}} & 0 \\ 0 & B^{\mathrm{T}} & R \end{pmatrix} - \lambda \begin{pmatrix} E & 0 & 0 \\ 0 & E^{\mathrm{T}} & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

This extended pencil is then compressed into a $2n \times 2n$ pencil in the same way as in Algorithm 13.5.3 and the rest of the procedure is the same as that algorithm.

13.6.3 The Inverse-Free Generalized Schur Method for the DDARE

The matrix pencil associated with the DDARE is

$$\begin{pmatrix} A & 0 \\ -Q & E^{\mathrm{T}} \end{pmatrix} - \lambda \begin{pmatrix} E & S \\ 0 & A^{\mathrm{T}} \end{pmatrix}, \text{ where } S = BR^{-1}B^{\mathrm{T}}.$$

The extended pencil for the **Inverse-free generalized Schur method for the DDARE** is

$$\begin{pmatrix} A & 0 & -B \\ -Q & E^{\mathrm{T}} & 0 \\ 0 & 0 & R \end{pmatrix} - \lambda \begin{pmatrix} E & 0 & 0 \\ 0 & A^{\mathrm{T}} & 0 \\ 0 & B^{\mathrm{T}} & 0 \end{pmatrix}.$$

The pencil is now compressed into a $2n \times 2n$ pencil as in Algorithm 13.5.4 and the rest of the steps of Algorithm 13.5.4 is then followed.

MATLAB note: The MATLAB function **dare** in the form $[X, L, G, rr] = \text{dare}(A, B, Q, R, E)$ solves the DDARE. Here $G = (B^{\mathrm{T}}XB + R)^{-1}B^{\mathrm{T}}XA$, $L = \text{eig}(A - BG, E)$, and rr = the Frobenius norm of the relative residual matrix.

13.7 CONCLUSIONS AND TABLE OF COMPARISONS

In this section, we present Table 13.1 which compares the different methods discussed in this chapter and gives a guideline for practical uses of these methods, based on this comparative study. We only present data for the CARE. A similar table can be set up for the DARE as well. However, the comments made about the Schur method for the CARE are not valid for the DARE, because *the Schur method*

Table 13.1: A table of comparisons of different methods for the CARE

Method	Efficiency, stability, and convergence properties	Remarks
The Eigenvector and the Generalized Eigenvector Methods	The methods are in general **not numerically stable** (they become unstable when the Hamiltonian matrix has nearly multiple eigenvalues).	**Not recommended to be used in practice.**
The Schur Method	**Stable in practice.**	**Widely used.**
The Symplectic Hamiltonian–Schur Method	**Stable and structure-preserving.** Requires less computations and storage for problems of size greater than 20.	**Works in the single-input.**
The Extended Hamiltonian–Schur Method	**Stable and structure-preserving. More-efficient** than the **Schur-method.**	**Works in the multi-input case.**
Newton's Method	Convergence is ultimately quadratic if the initial approximation is close to the solution slow initial convergence can be improved by using Newton's methods with line search.	Usually used as an **iterative refinement procedure.**
The Matrix Sign Function Method	**Not stable** in general. Though iterative in nature; unlike Newton's method, it does not require the knowledge of a stabilizing initial guess.	Simple to use and is structure preserving. **Recommended to be used in conjunction with Newton's method, with line search.**
The Generalized Schur Method	**Stable in practice.**	Does not work if the control weighting matrix R is singular. Even if R is theoretically nonsingular, the method should not be used if it is ill-conditioned.
The Inverse-Free Generalized Schur Method	**Stable in practice**	**The best way to solve the CARE is when R is nearly singular.**

for the DARE does not work when A is singular and is expected to give inaccurate results when A is theoretically nonsingular, but is computationally nearly singular.

Conclusions and Recommendations

In conclusion, the following recommendations are made: **For the CARE:** *The Schur method* (***Algorithm* 13.5.1**), *the extended Hamiltonian–Schur method or the matrix sign function* (***Algorithm* 13.5.6**) *method followed by Newton's iteration with line search* (***Algorithm* 13.5.9**) *is recommended. If R is singular or nearly singular, then the inverse-free generalized Schur method* (***Algorithm* 13.5.3**) *should be used in place of the Schur method or the matrix sign function method.*

For the DARE: *The inverse-free generalized Schur method* (***Algorithm* 13.5.4**) *or the matrix sign function method* (***Algorithm* 13.5.7**) *followed by Newton's method with line search* (***Algorithm* 13.5.11**) *is recommended. However, the matrix sign function method should be avoided if R is nearly singular.*

13.8 SOME SELECTED SOFTWARE

13.8.1 MATLAB Control System Toolbox

Matrix equation solvers.

care	Solve continuous algebraic Riccati equations
dare	Solve discrete algebraic Riccati equations.

13.8.2 MATCONTROL

RICEIGC	The eigenvector method for the continuous-time Riccati equation
RICSCHC	The Schur method for the continuous-time Riccati equation
RICSCHD	The Schur method for the discrete-time Riccati equation
RICGEIGD	The generalized eigenvector method for the discrete-time Riccati equation
RICNWTNC	Newton's method for the continuous-time Riccati equation
RICNWTND	Newton's method for the discrete-time Riccati equation
RICSGNC	The matrix sign function method for the continuous-time Riccati equation
RICSGND	The matrix sign function method for the discrete-time Riccati equation
RICNWLSC	Newton's method with line search for the continuous-time Riccati equation
RICNWLSD	Newton's method with line search for the discrete-time Riccati equation.

13.8.3 CSP-ANM

Solutions of the AREs:

- The Schur method is implemented as `RiccatiSolve` $[a, b, q, r,$ `SolveMethod` $\rightarrow$ `SchurDecomposition`] (continuous-time case) and `DiscreteRiccatiSolve` $[a, b, q, r,$ `SolveMethod` $\rightarrow$ `SchurDecomposition`] **(discrete-time case)**.
- Newton's method is implemented as `RiccatiSolve` $[a, b, q, r,$ `SolveMethod` $\rightarrow$ `Newton, InitialGuess` $\rightarrow w_0$] (discrete-time case).
- The matrix sign function method is implemented as `RiccatiSolve` $[a, b, q, r,$ `SolveMethod` $\rightarrow$ `MatrixSign`] (continuous-time case) and `DiscreteRiccatiSolve` $[a, b, q, r,$ `SolveMethod` $\rightarrow$ `MatrixSign`] (discrete-time case).
- The inverse-free method based on generalized eigenvectors is implemented as `RiccatiSolve` $[a, b, q, r,$ `SolveMethod` $\rightarrow$ `Generalized Eigendecomposition`] (continuous-time case) and `DiscreteRiccatiSolve` $[a, b, q, r,$ `SolveMethod` $\rightarrow$ `GeneralizedEigendecomposition`] (discrete-time case).
- The inverse-free method based on generalized Schur decomposition is implemented as `RiccatiSolve` $[a, b, q, r,$ `SolveMethod` $\rightarrow$ `GeneralizedSchurDecomposition`] (continuous-time case) and `DiscreteRiccatiSolve` $[a, b, q, r,$ `Solvemethod` $\rightarrow$ `GeneralizedSchurDecomposition`] (discrete-time case).

13.8.4 SLICOT

Riccati equations

SB02MD	Solution of AREs (Schur vectors method)
SB02MT	Conversion of problems with coupling terms to standard problems
SB02ND	Optimal state feedback matrix for an optimal control problem
SB02OD	Solution of AREs (generalized Schur method)
SB02PD	Solution of continuous algebraic Riccati equations (matrix sign function method) with condition and forward error bound estimates
SB02QD	Condition and forward error for continuous Riccati equation solution
SB02RD	Solution of AREs (refined Schur vectors method) with condition and forward error bound estimates
SB02SD	Condition and forward error for discrete Riccati equation solution

13.8.5 MATRIX$_X$

Purpose: Solve Riccati equation. Using the option "DISC" solves the discrete Riccati equation.

Syntax: [EV, KC] = RICCATI (S, Q, NS, 'DISC')
[EV, KC, P] = RICCATI (S, Q, NS, 'DISC')

Purpose: Solves the indefinite ARE: $A'P + PA - PRP + Q = 0$

Syntax: [P, SOLSTAT] = SINGRICCATI (A, Q, R { ,TYPE})

13.9 SUMMARY AND REVIEW

As we have seen in Chapters 10 and 12 that the AREs (13.1.1) and (13.1.2.) and their variations arise in many areas of control systems design and analysis, such as:

- The LQR and LQG designs
- Optimal state estimation (Kalman filter)
- H_∞-control
- Spectral factorizations (not described in this book, see Van Dooren 1981).

Existence and Uniqueness of Stabilizing Solution

Let $Q \geq 0$ and $R > 0$. If (A, B) is stabilizable and (A, Q) is detectable, then the CARE admits a unique symmetric positive semidefinite stabilizing solution (**Theorem 13.2.6**).

Conditioning of the Riccati Equations

The absolute and the relative condition numbers of the CARE have been identified using a perturbation result (**Theorem 13.4.1**).

An approximate condition number of the CARE, using a first-order estimate is Byers' condition number (in **Frobenius norm**):

$$\kappa^{\mathrm{B}}_{\mathrm{CARE}} = \frac{\|\Omega^{-1}\|\|Q\| + \|\Theta\|\|A\| + \|\Pi\|\|S\|}{\|X\|},$$

where X is the stabilizing solution of the CARE and Ω, Π, and Θ are defined by (13.4.4)–(13.4.6), and $\|\Omega^{-1}\|_{\mathrm{F}} = 1/\mathrm{sep}(A_{\mathrm{C}}^{\mathrm{T}}, -A_{\mathrm{C}})$, where $A_{\mathrm{C}} = A - SX$, $S = BR^{-1}B^{\mathrm{T}}$.

The quantities $\|\Omega^{-1}\|$, $\|\Theta\|$, and $\|\Pi\|$ are computationally intensive. Upper bounds of $\kappa^{\mathrm{B}}_{\mathrm{CARE}}$ can be obtained by solving the following Lyapunov equations:

$$(A - SX)^{\mathrm{T}} H_k + H_k(A - SX) = -X^k, \quad k = 0, 1, 2.$$

The large norms of these matrices (relative to the stabilizing solution X), in general, indicate that the CARE is ill-conditioned.

The condition number of the DARE is given by (13.4.16).

A first-order estimator for the condition number of the DARE is

$$\kappa^{\mathrm{E}}_{\mathrm{DARE}} = \frac{2\|A\|_{\mathrm{F}}^2 \|Q\|_{\mathrm{F}}/\|X\|_{\mathrm{F}} + \|A\|_{\mathrm{F}}^2 \|S\|_{\mathrm{F}} \|X\|_{\mathrm{F}}}{\mathrm{sep}(A_{\mathrm{d}}^{\mathrm{T}}, A_{\mathrm{d}})},$$

where $A_{\mathrm{d}} = A - B(R + B^{\mathrm{T}}XB)^{-1}B^{\mathrm{T}}XA$, $S = BR^{-1}B^{\mathrm{T}}$. The quantity $\mathrm{sep}(A_{\mathrm{d}}^{\mathrm{T}}, A_{\mathrm{d}})$ can be determined as the minimum singular value of the matrix $A_{\mathrm{d}}^{\mathrm{T}} \otimes A_{\mathrm{d}}^{\mathrm{T}} - I_n^2$.

Numerical Methods for the Riccati Equations

The numerical methods for the Riccati equations discussed here can be broadly classified into the following three classes:

- Invariant subspace methods
- Deflating subspace methods
- Newton's methods.

A basic idea of finding a stabilizing solution of the CARE (DARE), using eigenvector and Schur methods is to construct a basis for the stable invariant subspace of the Hamiltonian matrix H (symplectic matrix M). Such a basis can be constructed using the eigenvectors or the Schur vectors of the Hamiltonian matrix H (the symplectic matrix M). The eigenvector matrix can be ill-conditioned if the matrix H (the matrix M) is nearly defective and, therefore, **the eigenvector approach is not recommended to be used in practice**. The Schur method is preferable to the eigenvector method. If

$$U^{\mathrm{T}} H U = \begin{pmatrix} T_{11} & T_{12} \\ 0 & T_{22} \end{pmatrix}$$

is the **ordered** RSF of H, and the eigenvalues with negative real parts are contained in T_{11}, then $X = U_{21}U_{11}^{-1}$ is the unique stabilizing solution of the CARE, where

$$U = \begin{pmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{pmatrix}.$$

The Schur method for the DARE can be similarly developed by finding an ordered RSF of the symplectic matrix M. However, since computation of the matrix M

requires the explicit inversion of A, **the Schur method for the DARE does not work if A is singular or can be problematic if A is theoretically nonsingular, but is computationally singular.** In such cases, a deflating subspace method should be used.

The idea behind the generalized eigenvector and Schur vector methods is basically the same as that of an invariant subspace method except that the solution of the Riccati equation is now found by computing a basis for the deflating subspace of a matrix pencil. For the CARE, the pencil is $P_{\text{CARE}} - \lambda N_{\text{CARE}}$, where

$$P_{\text{CARE}} = \begin{pmatrix} A & -S \\ -Q & A^{\text{T}} \end{pmatrix}, \qquad N_{\text{CARE}} = \begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix}.$$

For the DARE, the matrices of the pencil are

$$P_{\text{DARE}} = \begin{pmatrix} A & 0 \\ -Q & I \end{pmatrix}, \qquad N_{\text{DARE}} = \begin{pmatrix} I & S \\ 0 & A^{\text{T}} \end{pmatrix}.$$

Again, for reasons stated above, the **generalized Schur decomposition using the QZ algorithm should be used to compute such a basis. See Section 13.5.2 for details. The eigenvector approach should be avoided.**

Both the Schur methods and the generalized Schur methods require an explicit inversion of the matrix R. In case R is ill-conditioned with respect to matrix inversion, these methods may not give accurate solutions. The difficulties can be overcome by using an extended $(2n + m) \times (2n + m)$ pencil.

For the CARE, the extended pencil is $P^{\text{E}}_{\text{CARE}} - \lambda N^{\text{E}}_{\text{CARE}}$, where

$$P^{\text{E}}_{\text{CARE}} = \begin{pmatrix} A & 0 & B \\ -Q & -A^{\text{T}} & 0 \\ 0 & B^{\text{T}} & R \end{pmatrix}, \qquad N^{\text{E}}_{\text{CARE}} = \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

This extended $(2n + m) \times (2n + m)$ pencil can then be compressed into a $2n \times 2n$ pencil by finding the QR factorization of $\begin{pmatrix} R \\ B \end{pmatrix}$, without affecting the deflating subspace. The solution of the CARE can then be obtained by finding the ordered generalized Schur form of the compressed pencil.

For the DARE, the extended pencil is $P^{\text{E}}_{\text{DARE}} - \lambda N^{\text{E}}_{\text{DARE}}$, where

$$P^{\text{E}}_{\text{DARE}} = \begin{pmatrix} A & 0 & -B \\ -Q & -I & 0 \\ 0 & 0 & R \end{pmatrix}, \qquad N^{\text{E}}_{\text{DARE}} = \begin{pmatrix} I & 0 & 0 \\ 0 & A^{\text{T}} & 0 \\ 0 & B^{\text{T}} & 0 \end{pmatrix}.$$

This $(2n + m) \times (2n + m)$ can be compressed into a $2n \times 2n$ pencil by using the QR factorization of $\begin{pmatrix} R \\ -B \end{pmatrix}$. For details, see **Section 13.5.2.**

Again, the required basis should be constructed by finding the generalized RSF of the pencil using the QZ algorithm.

13.10 CHAPTER NOTES AND FURTHER READING

The AREs have been very well studied in the literatures of mathematics and control and filter theory.

For an excellent account of up-to-date theoretical developments, see the recent book of Lancaster and Rodman (1995). Some of the earlier theoretical developments are contained in Kučera (1972, 1979), Coppel (1974), and Singer and Hammarling (1983), Willems (1971), Wimmer (1984, 1994), Lancaster and Rodman (1980). The books by Anderson and Moore (1990), Ando (1988), Kwakernaak and Sivan (1972), Kimura (1997), Zhou *et al.* (1996) also contain a fair amount of theory of AREs. The existence of maximal solutions for generalized AREs arising in stochastic control has been discussed in DeSouza and Fragoso (1990). The paper by DeSouza *et al.* (1986) deals with Riccati equations arising in optimal filtering of nonstabilizable systems having singular state transition matrices. For some application of Riccati equations in general forms to dynamic games, see Basar (1991).

Important numerical methods have been dealt with in details in the books by Sima (1996) and Mehrmann (1991). Benner (1999) has given an up-to-date review with special attention to structure-preserving methods. An extensive bibliography on numerical methods appear in Laub (1991) and Benner (1997). See Jamshidi (1980) for an earlier review.

For a review of periodic Riccati equations see the article of Bittanti *et al.* and the references therein in the book "**The Riccati Equation**" edited by Bittanti *et al.* (1991). The latter contains several important papers on Riccati equations and the paper by Bittanti gives a brief life history of Count Jacopo Riccati (1676–1754), which is certainly worth reading.

The sensitivity of the continuous-time Riccati equations has been studied by several people: Byers (1985), Kenney and Hewer (1990), Chen (1988), Konstantinov *et al.* (1990), Xu (1996), Sun (1998), and Ghavimi and Laub (1995), etc. **Theorem 13.4.1** is due to Sun (1998). The bound (13.4.14) is due to Kenney and Hewer (1990). The residual of an approximate stabilizing solution (**Theorem 13.4.3**) is due to Sun (1997a). The sensitivity of the DARE has been studied in Gudmundsson *et al.* (1992), Konstantinov *et al.* (1993), Sun (1998), and Gahinet *et al.* (1990). The paper by Ghavimi and Laub (1995) relates backward error and sensitivity to accuracy and discusses techniques for refinement of computed solutions of the AREs. For recent results, see Petkov *et al.* (2000). For results on the upper and lower bounds of the solutions of CARE and DARE, see Lee (1997a, 1997b).

The eigenvector methods for the Riccati equations were proposed by McFarlane (1963) and Potter (1966). The Schur method for the Riccati equations originally appeared in the celebrated paper by Laub (1979). Petkov *et al.* (1987) studied the numerical properties of the Schur method and concluded that the Schur method

can be unstable in some cases and the solutions may be inaccurate. A further analysis by Kenney *et al.* (1989) attributed such inaccuracy to poor scaling. For an excellent account of scaling of the Schur methods, see Benner (1997). The structure-preserving Hamiltonian–Schur method was first proposed by Byers in his Householder-prize winning Ph.D. thesis (1983) in the case of a single-input system (rank$(B) = 1$). See Byers (1986a) for details of the method. The theoretical foundation of this method is contained in the well-known paper by Paige and Van Loan (1981). Their result was later extended to the case when the Hamiltonian matrix has eigenvalues on the imaginary axis by Lin and Ho (1990). Patel *et al.* (1994) have discussed computation of stable invariant subspaces of Hamiltonian matrices. Another method, called the multishift method to compute the invariant subspace of the Hamiltonian matrix corresponding to the stable eigenvalues was developed by Ammar and Mehrmann (1993). The algorithm is called the multishift algorithm because n stable eigenvalues of the Hamiltonian matrix are used as shifts to isolate the desired invariant subspace. The multishift method sometimes has convergence problems, particularly for large n. The Hamiltonian–Schur algorithm in the multi-input case is due to Benner *et al.* (1997c). For structure-preserving eigenvalue methods see Benner *et al.* (1999c) and Bunse-Gerstner *et al.* (1992). Mehrmann (1988) has given a structure-preserving method for the discrete-time Riccati equation with single-input and single-output. The non-orthogonal symplectic methods have been discussed by Bunse-Gerstner and Mehrmann (1986) and Bunse-Gerstner *et al.* (1989) for the CARE, and by Benner *et al.* (1999b), Fassbender and Benner (2001) for the DARE. The details of these methods and other references can be found in the recent book by Fassbender (2000). Interesting connection between structure-preserving HR and SR algorithms appears in Benner *et al.* (1997a).

The generalized eigenvalue problem approach leading to deflating subspace method for the discrete-time Riccati equation was proposed by Pappas *et al.* (1980). See also Arnold and Laub (1984), Emami-Naeini and Franklin (1979, 1982). The inverse-free methods (the extended pencil approach (**Algorithm 13.5.3** and **Algorithm 13.5.4**)) and the associated compressed techniques were proposed by Van Dooren (1981).

The idea of using matrix sign function to solve the CARE was first introduced by Roberts (1980, [1971]). Byers (1986b, 1987) discussed numerical stability of the method and studied the computational aspects in details. See also Bierman (1984) and Bai and Demmel (1998). A generalization of the matrix sign function method to a matrix pencil and its application to the solutions of DCARE and DDARE was proposed by Gardiner and Laub (1986). For an account of the matrix sign function, see the recent paper of Kenney and Laub (1995). For a perturbation analysis of the matrix sign function, see Sun (1997c). Howland (1983) relates matrix sign function to separation of matrix eigenvalues.

For details of Newton's algorithm for the CARE (**Algorithm 13.5.8**) and that of the DARE (**Algorithm 13.5.10**), as presented here, see Benner (1997), Lancaster and Rodman (1995). The correct proof of convergence of Newton's method (**Theorem 13.5.8**) seemed to appear for the first time in Lancaster and Rodman (1995).

Kenney *et al.* (1990) gave results on error bounds for Newton's method, where it was first pointed out that if the initial solution X_0 is not chosen carefully, the error on the first step can be disastrous. They also gave conditions which guarantee monotone convergence from the first step on (**Theorem 13.5.9**). Several modifications of Newton's methods have appeared in recent years (Guo 1998; Guo and Lancaster 1998; Guo and Laub 2000; etc.). The line search modification proposed by Benner and Byers (1998) is extremely useful in practice. In general, it improves the convergence behavior of Newton's method and avoids the problem of a disastrously large first step. For acceleration techniques of the DARE, see Benner (1998).

Ghavimi *et al.* (1992) have discussed the local convergence analysis of conjugate gradient methods for solving the AREs.

For an account of parallel algorithms for AREs, see Bai and Qian (1994), Gardiner and Laub (1991), and Laub (1991) and references therein, Quintana and Hernández (1996a, 1996b, 1996c), etc.

For large-scale solutions of the AREs see Ferng *et al.* (1997), Lu and Lin (1993), Jaimoukha and Kasenally (1994) and Benner and Fassbender (1997). The recent book by Ionescu *et al.* (1999) gives a nice treatment of AREs for the indefinite sign and singular cases. See also Campbell (1980). For least-squares solutions of stationary optimal control using the AREs, see Willems (1971).

Some discussions on finding the Cholesky factor of the solution to an ARE without first computing the solution itself appears in Singer and Hammarling (1983). Lin (1987) has given a numerical method for computing the closed-loop eigenvalues of a discrete-time Riccati equation. Patel (1993) has given a numerical method for computing the eigenvalues of a simplectic matrix. For numerical algorithms for descriptor Riccati equations, see Benner (1999), Mehrmann (1991), Bender and Laub (1985, 1987), Benner *et al.* (1999a), etc. A description of discrete-time descriptor Riccati equations also appears in Zhang *et al.* (1999). A comparative study with respect to efficiency and accuracy of most of the methods described in this chapter for the CARE (the **eigenvector, Schur, inverse-free generalized Schur, Hamiltonian–Schur and Newton's Methods**) has been made in the recent M.Sc. Thesis of Ho (2000), using MATLAB and FORTRAN-77 codes. (In particular, this thesis contains MATLAB codes for ordered **Real Schur** and **Generalized Real Schur** decompositions). Numerical experiments were performed on 12 benchmark examples taken from the collection of Benner *et al.* (1995a, 1997b). The conclusions drawn in this thesis are almost identical to those mentioned in **Section 13.7**. For a recent collection of benchmark examples for Riccati equations, see Abels and Benner (1999a, 1999b).

Exercises

13.1 Derive necessary and sufficient conditions for the CARE (13.1.1) to have a unique symmetric positive definite stabilizing solution X.

13.2 Construct an example to show that the observability of (A, Q) is not necessary for the solution X of the CARE (13.1.1) to be positive definite.

13.3 Prove that the matrix defined in (13.3.1) is symplectic, and that if λ is a nonzero eigenvalue of M, so is $1/\bar{\lambda}$.

13.4 Establish the relation (13.2.16).

13.5 (a) Prove the discrete counterpart of Theorem 13.2.4, that is, prove an analogous theorem for the DARE.

(b) Using the results of Problem 13.3, and those of 13.5(a), prove Theorem 13.3.2.

13.6 Prove that the homogeneous CARE: $XA + A^{\mathrm{T}}X + XSX = 0$ has a stabilizing solution if A has no eigenvalues on the imaginary axis. Prove or disprove a discrete-counterpart of this result.

13.7 Prove that the quantity (13.4.22) serves as an approximate condition number of the DARE (13.1.2). Construct an example of an ill-conditioned DARE using this quantity.

13.8 Find an example to illustrate that a small relative residual in a computed solution of the CARE does not guarantee a small error in the solution.

13.9 Prove that if Ω is singular, then $\mathrm{sep}((A - SX), -(A - SX)^{\mathrm{T}})$ is zero.

13.10 Give a proof of Algorithm 13.5.1, making necessary assumptions.

13.11 Construct an example to show that the solution of the CARE, obtained by Algorithm 13.5.1, might be inaccurate, even though the problem is not ill-conditioned. (**Hint:** Construct an example for which U_{11} is ill-conditioned, but the CARE is well-conditioned.)

13.12 Give an example to demonstrate the superiority of the Schur algorithm for the CARE over the eigenvector algorithm, in case the associated Hamiltonian matrix is nearly defective.

13.13 Using Theorem 13.5.1 and the transformation

$$H = (M + I_{2n})(M - I_{2n})^{-1},$$

prove Theorem 13.5.2.

13.14 Construct an example to demonstrate the numerical difficulties of the Schur algorithm for the DARE in case the matrix A is nearly singular.

13.15 Write down an algorithm for solving the discrete algebraic Riccati equation, using the eigenvectors of the symplectic matrix. Discuss the computational drawbacks of the algorithm. Construct an example to illustrate the computational drawbacks.

13.16 Prove the properties 1 through 5 of the matrix sign function $\mathrm{Sign}(A)$ stated in Section 13.5.3.

13.17 Prove that if $|\lambda| = 1$ is an eigenvalue of the pencil $P_{\mathrm{DARE}} - \lambda N_{\mathrm{DARE}}$ with the eigenvector $z = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix}$, where P_{DARE} and N_{DARE} are the same as given in Theorem 13.5.5, then the detectability of (A, Q) implies that $z_1 = 0$.

13.18 Formulate the generalized Schur method for the CARE and develop that for the DARE in details.

13.19 Why is the generalized Schur method not preferable over the Schur method for the CARE if R is not nearly singular?

13.20 Construct an example to demonstrate the poor accuracy of the generalized eigenvector method for the DARE in case the pencil $P_{\text{DARE}} - \lambda N_{\text{DARE}}$ has near multiple eigenvalues.

Apply the generalized Schur algorithm (Algorithm 13.5.2) to the same example and verify the improvement in the accuracy of the solution.

13.21 Work out the details of how the pencil $P^{\text{E}}_{\text{CARE}} - \lambda N^{\text{E}}_{\text{CARE}}$ can be transformed to the compressed pencil $P^{\text{EC}}_{\text{CARE}} - \lambda N^{\text{EC}}_{\text{CARE}}$ using the QR factorization of the matrix $\begin{pmatrix} R \\ B \end{pmatrix}$.

13.22 Repeat the previous exercise for the DARE, that is, work out the details of the transformation to the pencil $P^{\text{EC}}_{\text{DARE}} - \lambda N^{\text{EC}}_{\text{DARE}}$ using the QR factorization of the matrix $\begin{pmatrix} R \\ -B \end{pmatrix}$.

13.23 Prove that the pencil $P^{\text{E}}_{\text{CARE}} - \lambda N^{\text{E}}_{\text{CARE}}$ and the pencil $P^{\text{EC}}_{\text{CARE}} - \lambda N^{\text{EC}}_{\text{CARE}}$ as defined in Section 13.5.2 for the CARE have the same deflating subspaces, and similarly for the DARE.

13.24 Develop the following algorithms in detail for both the DCARE and DDARE (consult Laub (1991) and Benner (1997)):

The Schur algorithms, the generalized Schur algorithms, the inverse-free generalized Schur algorithms, the matrix sign function algorithms, and Newton's algorithms.

Construct a simple example to illustrate each of the above algorithms.

13.25 Perform numerical experiments to compare Newton's methods with Newton's methods with line search, both for the CARE and DARE, by using several examples from the Benchmark collections in Benner *et al.* (1995a, 1995b, 1997b). Display your results on number of iterations and norms of the residual matrices using tables and graphs.

13.26 Construct an example to demonstrate the superiority of the inverse-free generalized Schur algorithm over the Schur algorithm for the CARE, in case the control weighting matrix R is positive definite but nearly singular.

13.27 Carry out a numerical experiment with a 150×150 randomly generated problem to make a comparative study with respect to computer-time and accuracy of the solution to the CARE with the following methods: *the eigenvector method, the Schur method, inverse-free generalized Schur method, the matrix sign function method,* and *the Hamiltonian structure-preserving Schur method.* Write down your observations and conclusions with tables and graph.

13.28 Repeat the previous exercise with the DARE using the following methods: The *eigenvector method, the generalized eigenvector method, the Schur method, the generalized Schur method, inverse-free generalized Schur method, and the matrix sign function method.*

13.29 (Kenney and Hewer 1990). Study the sensitivity of the solution of the CARE with the following data, for $\epsilon = 10^0, 10^{-1}, 10^{-2}, 10^{-3}$. Present your results with tables

and graphs.

$$A = \begin{pmatrix} -\epsilon & 1 & 0 & 0 \\ -1 & -\epsilon & 0 & 0 \\ 0 & 0 & \epsilon & 1 \\ 0 & 0 & -1 & \epsilon \end{pmatrix}, \qquad B = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \qquad R = 1, \qquad Q = BB^{\mathrm{T}}.$$

Research Problems

13.1 Develop a structure-preserving method to compute the symplectic Schur decomposition and apply the method to solve the DARE, thus obtaining a symplectic structure-preserving method for the DARE.

References

Abels J. and Benner P. "CAREX – a collection of benchmark examples for continuous-time algebraic Riccati equations (version 2.0)". *SLICOT Working Note* 1999-14, November 1999a. (Available at the NICONET Website: http://www.win.tue.ne/niconet/niconet.html).

Abels J. and Benner P. "DAREX–a collection of benchmark examples for discrete-time algebraic Riccati equations (version 2.0)". *SLICOT Working Note* 1999-15, November 1999b. (Available at the NICONET Website: http://www.win.tue.ne/niconet/niconet.html).

Ammar G. and Mehrmann V. "A multishift algorithm for the numerical solution of algebraic Riccati equations," *Electr. Trans. Num. Anal.*, Vol. 1, pp. 33–48, 1993.

Anderson B.D.O. and Moore J.B. *Optimal Control: Linear Quadratic Methods*, Prentice Hall, Englewood Cliffs, NJ, 1990.

Anderson E., Bai Z., Bischof C., Blockford S., Demmel J., Dongarra J., DuCroz J., Greenbaum A., Hammarling S., McKenney A., and Sorensen D. *LAPACK Users' Guide*, 3rd edn, SIAM, Philadelphia, 1999.

Ando T. *Matrix Quadratic Equations*, Hokkaido University, Research Institute of Applied Electricity, Division of Applied Mathematics, Sapporo, Japan, 1988.

Arnold W. III and Laub A. "Generalized eigenproblem algorithms and software for algebraic Riccati equations," *Proc. IEEE*, Vol. 72, pp. 1746–1754, 1984.

Bai Z. and Demmel J. "Using the matrix sign function to compute invariant subspaces," *SIAM J. Matrix Anal. Appl.*, Vol. 19, pp. 205–225, 1998.

Bai Z. and Qian Q. "Inverse free parallel method for the numerical solution of algebraic Riccati equations," in *Proc. Fifth SIAM Conf. Appl. Lin. Alg.* (Lewis J. ed.) June, pp. 167–171, Snowbird, UT, 1994.

Basar T. "Generalized Riccati equations in dynamic game," in *The Riccati Equation*, (Bittanti S., Alan Laub and Williams J.C. eds.), Springer-Verlag, Berlin, 1991.

Bender D. and Laub A. "The linear-quadratic optimal regulator problem for descriptor systems," *Proc.* 24th *IEEE Conf. Dec. Control*, Ft. Lauderdale, Florida, December, pp. 957–962, 1985.

Bender D. and Laub A. "The linear-quadratic optimal regular for descriptor systems: discrete-time case," *Automatica*, Vol. 23, pp. 71–85, 1987.

Benner P. *Contributions to the Numerical Solution of Algebraic Riccati Equations and Related Eigenvalue Problems*, Dissertation for Dipl.-Math., Technischen Universität Chemnitz-Zwickau, Germany 1997.

Benner P. "Acclerating Newton's method for discrete-time Riccati equations," in *Proc. MTNS'98*, (Beghi A., Finesso L., and Picci G. eds.) pp. 569–572, Padova, Italy, 1998.

Benner P. "Computational methods for linear-quadratic optimization," *Rendiconti del Circulo Matematico di Palermo*, Supplemento, Serie II, no. 58, pp. 21–56, 1999.

Benner P. and Byers R. "An exact line search method for solving generalized continuous-time algebraic Riccati equations," *IEEE Trans. Autom. Control*, Vol. 43, pp. 101–107, 1998.

Benner P., Byers R., Mehrmann V., and Xu H. "Numerical solution of linear–quadratic control problems for descriptor systems," in *Proc. 1999 IEEE Intl. Symp. CACSD*, (Gonzalez O. ed.) Kohala Coast-Island of Hawaii, Hawaii, USA, pp. 46–51, August 22–27, 1999a.

Benner P. and Fassbender H. "An implicitly restarted symplectic Lanczos method for the Hamiltonian eigenvalue problem," *Lin. Alg. Appl.*, Vol. 263, pp. 75–111, 1997.

Benner P., Fassbender H., and Watkins D., "Two connections between the SR and HR eigenvalue algorithms," *Lin. Alg. Appl.*, Vol. 272, pp. 17–32, 1997a.

Benner P., Fassbender H. and Watkins D. "SR and SZ algorithms for the symplectic (butterfly) eigenproblem," *Lin. Alg. Appl.*, Vol. 287, pp. 41–76, 1999b.

Benner P., Laub A., and Mehrmann V. "A collection of benchmark examples for the numerical solution of algebraic Riccati equations I: continuous-time case," *Tech. Report SPC 95-22, Fak. f. Mathematik, TU Chemnitz-Zwickau, 09107 Chemnitz, FRG, 1995a.*

Benner P., Laub A., and Mehrmann V. "A collection of benchmark examples for the numerical solution of algebraic Riccati equations II: discrete-time case," *Tech. Report SPC 95-23, Fak. f. Mathematik, TU Chemnitz-Zwickau*, 09107 *Chemnitz, FRG, 1995b.*

Benner P., Laub A., and Mehrmann V. "Benchmarks for the numerical solution of algebraic Riccati equations," *IEEE Control Syst. Mag.*, Vol. 7(5), pp. 18–28, 1997b.

Benner P., Mehrmann V., and Xu H. "A new method for computing the stable invariant subspace of a real Hamiltonian matrix," *J. Comput. Appl. Math.*, Vol. 86, pp. 17–43, 1997c.

Benner P., Mehrmann V., and Xu H. "A numerically stable, structure preserving method for computing the eigenvalues of real Hamiltonian or symplectic pencils," *Numer. Math.*, Vol. 78, pp. 329–358, 1999c.

Bierman G.J. "Computational aspects of the matrix sign function solution to the ARE," *Proc. 23rd IEEE Conf. Dec. Contr.*, Las Vegas, Nevada, pp. 514–519, 1984.

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

Bunse-Gerstner A., Mehrmann V., and Watkins D. "An SR algorithm for Hamiltonian matrices based on Gaussian elimination," *Meth. Oper. Res.*, Vol. 58, pp. 339–358, 1989.

Bunse-Gerstner A., Byers R., and Mehrmann V. "A chart of numerical methods for structured eigenvalue problems," *SIAM J. Matrix Anal. Appl.*, Vol. 13(2), pp. 419–453, 1992.

Bunse-Gerstner A. and Mehrmann V. "A symplectic QR-like algorithm for the solution of the real algebraic Riccati equation," *IEEE Trans. Autom. Control*, Vol. AC-31, pp. 1104–1113, 1986.

Byers R. *Hamiltonian and Symplectic Algorithms for the Algebraic Riccati Equation*, Ph.D. thesis, Cornell University, Department of Computer Science, Ithaca, NY, 1983.

Byers R. "Numerical condition of the algebraic Riccati equation," in *Contemporary Mathematics*, (Brualdi R. *et al.*, ed.) Vol. 47, pp. 35–49, American Mathematical Society, Providence, RI, 1985.

Byers R. "A Hamiltonian QR-algorithm," *SIAM J. Sci. Statist. Comput.*, Vol. 7, pp. 212–229, 1986a.

Byers R. "Numerical stability and instability in matrix sign function based algorithms," in *Computational and Combinatorial Methods in Systems Theory* (Byrnes C.I. and Lindquist A. eds.) pp. 185–200, North Holland, New York, 1986b.

Byers R. "Solving the algebraic Riccati equation with the matrix sign function," *Lin. Alg. Appl.*, Vol. 85, pp. 267–279, 1987.

Byers R. "A Hamiltonian Jacobi Algorithm," *IEEE Trans. Autom. Control*, Vol. 35(5), pp. 566–570, 1990.

Campbell S.L. *Singular Systems of Differential Equations*, Pitman, Marshfield, MA, 1980.

Chen C.-H. "Perturbation analysis for solutions of algebraic Riccati equations," *J. Comput. Math.*, Vol. 6, pp. 336–347, 1988.

Coppel W.A. "Matrix quadratic equations," *Bull. Australian Math. Soc.*, Vol. 10, pp. 327–401, 1974.

Denman E.D. and Beavers A.N. "The matrix sign function and computations in systems," *Appl. Math. Comput.*, Vol. 2, pp. 63–94, 1976.

DeSouza C.E. and Fragoso M.D. "On the existence of maximal solution for generalized algebraic Riccati equations arising in stochastic control," *Syst. Control Lett.*, Vol. 14, pp. 223–239, 1990.

DeSouza C.E., Gevers M.R., and Goodwin G.C. "Riccati equations in optimal filtering of nonstabilizable systems having singular state transition matrices," *IEEE Trans. Autom. Control*, Vol. AC-31, pp. 831–838, 1986.

Emami-Naeini A. and Franklin G.F. "Design of steady-state quadratic-loss optimal digital controls for systems with a singular system matrix, Proc. 13th Astimor Conf. Circuits, Systems and Computers, pp. 370–374, 1979.

Emami-Naeini A. and Franklin G.F. "Deadbeat control and tracking of discrete-time systems," *IEEE Trans. Autom. Control*, Vol. AC-27, pp. 176–181, 1982.

Fassbender H. and Benner P. "A hybrid method for the numerical solution of discrete-time algebraic Riccati equations," in *Contemporary Mathematics on Structured Matrices in Mathematics, Computer Science, and Engineering* (Olshevsky V. ed.) Vol. 280, pp. 255–269, American Mathematical Society, Providence, RI, 2001.

Fassbender H. *Symplectic Method for the Symplectic Eigenproblem,* Kluwer Academic/Plenum Publishers, New York, 2000.

Ferng W.R., Lin W.-W., and Wang C.-S. "The shift-inverted J-Lanczos algorithm for the numerical solutions of large sparse algebraic Riccati equations," *Comput. Math. Appl.*, Vol. 33(10), pp. 23–40, 1997.

Gahinet P.M., Laub A.J., Kenney C.S., and Hewer G.A. "Sensitivity of the stable discrete-time Lyapunov equation," *IEEE Trans. Autom. Control*, Vol. 35, pp. 1209–1217, 1990.

Gardiner J.D. and Laub A.J. "A generalization of the matrix-sign-function solution for algebraic Riccati equations," *Int. J. Control*, Vol. 44, pp. 823–832, 1986.

Gardiner J.D. and Laub A.J. "Parallel algorithms for algebraic Riccati equations," *Int. J. Control*, Vol. 54, pp. 1317–1333, 1991.

Ghavimi A., Kenney C., and Laub A.J. "Local convergence analysis of conjugate gradient methods for solving algebraic Riccati equations," *IEEE Trans. Autom. Control*, Vol. AC-37, pp. 1062–1067, 1992.

Ghavimi A.R. and Laub A.J. "Backward error, sensitivity, and refinement of computed solutions of algebraic Riccati equations," *Num. Lin. Alg. Appl.*, Vol. 2, pp. 29–49, 1995.

Gudmundsson T., Kenney C., and Laub A.J. "Scaling of the discrete-time algebraic Riccati equation to enhance stability of the Schur method," *IEEE Trans. Autom. Control*, Vol. AC-37, pp. 513–518, 1992.

Guo C.-H. "Newton's method for discrete algebraic Riccati equations when the closed-loop matrix has eigenvalues on the unit circle," *SIAM J. Matrix Anal. Appl.*, Vol. 20, pp. 279–294, 1998.

Guo C.-H. and Lancaster P. "Analysis and modification of Newton's method for algebraic Riccati equations," *Math. Comp.*, Vol. 67, pp. 1089–1105, 1998.

Guo C.-H. and Laub A.J. "On a Newton-like method for solving algebraic Riccati equations," *SIAM J. Matrix Anal. Appl.*, Vol. 21, pp. 694–698, 2000.

Hammarling S.J. *Newton's Method for Solving the Algebraic Riccati Equation*, NPL Report DITC 12/82, National Physical Laboratory, Teddington, Middlesex TW11 OLW, U.K., 1982.

Hewer G.A. "An iterative technique for the computation of steady state gains for the discrete optimal controller," *IEEE Trans. Autom. Control*, Vol. AC-16, pp. 382–384, 1971.

Ho T. *A Study of Computational Methods for the Continuous-time Algebraic Riccati Equation*, M.Sc. Thesis, Northern Illinois University, DeKalb, Illinois, 2000.

Howland J.L. "The sign matrix and the separation of matrix eigenvalues," *Lin. Alg. Appl.*, Vol. 49, pp. 221–232, 1983.

Ionescu V., Oara C., and Weiss M. *Generalized Riccati Theory and Robust Control*, John Wiley, New York, 1999.

Jaimoukha I.M. and Kasenally E.M. "Krylov subspace methods for solving large Lyapunov equations," *SIAM J. Numer. Anal.* Vol. 31, 227–251, 1994.

Jamshidi M. "An overview on the solutions of the algebraic Riccati equation and related problems," *Large-Scale Syst.*, Vol. 1, pp. 167–192, 1980.

Kenney C.S. and Hewer G. "The sensitivity of the algebraic and differential Riccati equations," *SIAM J. Control Optimiz.*, Vol. 28, pp. 50–69, 1990.

Kenney C.S. and Laub A.J. "On scaling Newton's method for polar decomposition and the matrix sign function," *SIAM J. Matrix Anal. Appl.*, Vol. 13, pp. 688–706, 1992.

Kenney C.S. and Laub A.J. "The matrix sign function," *IEEE Trans. Autom. Control*, Vol. 40, pp. 1330–1348, 1995.

Kenney C.S., Laub A.J., and Wette M. "A stability-enhancing scaling procedure for Schur-Riccati solvers," *Syst. Control Lett.*, Vol. 12, pp. 241–250, 1989.

Kenney C.S., Laub A.J., and Wette M. "Error bounds for Newton refinement of solutions to algebraic Riccati equations," *Math. Control, Signals, Syst.*, Vol. 3, pp. 211–224, 1990.

Kimura, H. *Chain-Scattering Approach to H∞-Control*, Birkhäuser, Boston, 1997.

Kleinman D.L. "On an iterative technique for Riccati equation computations," *IEEE Trans. Autom. Control*, Vol. AC-13, pp. 114–115, 1968.

Kleinman D.L. "Stabilizing a discrete, constant linear system with application to iterative methods for solving the Riccati equation, *IEEE Trans. Autom. Control*, Vol. AC-19, pp. 252–254, 1974.

Konstantinov M.M., Petkov P., and Christov N.D. "Perturbation analysis of matrix quadratic equations," *SIAM J. Sci. Stat. Comput.*, Vol. 11, pp. 1159–1163, 1990.

Konstantinov M.M., Petkov P., and Christov N.D. "Perturbation analysis of the discrete Riccati equation," *Kybernetika*, Vol. 29, pp. 18–29, 1993.

Konstantinov M.M., Petkov P., Gu D.W., and Postlethwaite I. *Perturbation Techniques for Linear Control Problems*, Report 95-7, Control Systems Research, Department of Engineering, Leicester University, UK, 1995.

Kučera V. "A contribution to matrix quadratic equations," *IEEE Trans. Autom. Control*, Vol. 17, pp. 344–347, 1972.

Kučera, V. *Discrete Linear Control*, John Wiley & Sons, New York, 1979.

Kwakernaak H. and Sivan R. *Linear Optimal Control Systems*, Wiley-Interscience, New York, 1972.

Lancaster P. and Rodman L. "Existence and uniqueness theorems for algebraic Riccati equations," *Int. J. Control*, Vol. 32, pp. 285–309, 1980.

Lancaster P. and Rodman L. *Algebraic Riccati Equations*, Oxford University Press, Oxford, 1995.

Lee C.-H. "On the upper and lower bounds of the solution of the continuous algebraic Riccati matrix equation," *Int. J. Control*, Vol. 66, pp. 105–118, 1997a.

Lee C.-H. "Upper and lower bounds of the solutions of the discrete algebraic Riccati and Lyupunov matrix equations," *Int. J. Control*, Vol. 68, pp. 579–598, 1997b.

Laub A.J. "A Schur method for solving algebraic Riccati equations," *IEEE Trans. Autom. Control*, Vol. AC-24, pp. 913–921, 1979.

Laub, A.J. "Invariant subspace methods for the numerical solution of Riccati equations," in *The Riccati Equation* (Bittanti S. *et al.*, eds.) pp. 163–196, Springer-Verlag, Berlin 1991.

Lin W.-W. "A new method for computing the closed-loop eigenvalues of a discrete-time algebraic Riccati equation," *Lin. Alg. Appl.*, Vol. 6, pp. 157–180, 1987.

Lin W.-W. and Ho T.-C. *On Schur type decompositions of Hamiltonian and Symplectic Pencils*, Tech. Report, Institute of Applied Mathematics, National Tsing Hua University Taiwan, 1990.

Lu L. and Lin W.-W. "An iterative algorithm for the solution of the discrete-time algebraic Riccati equation," *Lin. Alg. Appl.*, Vol. 188/189, pp. 465–488, 1993.

McFarlane A. "An eigenvector solution of the optimal linear regulator problem," *J. Electron. Control*, Vol. 14, pp. 643–654, 1963.

Mehrmann V. *The Autonomous Linear Quadratic Control Problem.* Lecture Notes in Control and Information Sciences, Vol. 163, Springer-Verlag, Berlin, 1991.

Mehrmann V. "A symplectic orthogonal method for single-input single-output discrete-time optimal linear quadratic control problems," *SIAM J. Matrix Anal. Appl.*, Vol. 9, pp. 221–247, 1988.

Paige C. and Van Loan C. "A Schur decomposition for Hamiltonian matrices," *Lin. Alg. Appl.*, pp. 11–32, 1981.

Pandey P. "On scaling an algebraic Riccati equation," *Proc. Amer. Control Conf.*, Vol. 9, pp. 1583–1587, June, 1993.

Pappas T., Laub A.J., and Sandell N. "On the numerical solution of the discrete-time algebraic Riccati equation," *IEEE Trans. Autom. Control*, Vol. AC-25, pp. 631–641, 1980.

Patel R.V. "On computing the eigenvalues of a symplectic pencil," *Lin. Alg. Appl.*, Vol. 188/189, pp. 591–611, 1993.

Patel R.V., Lin Z., and Misra P. "Computation of stable invariant subspaces of Hamiltonian matrices," *SIAM J. Matrix Anal. Appl.*, Vol. 15, pp. 284–298, 1994.

Petkov P., Christov N.D., and Konstantinov M.M. "Numerical properties of the generalized Schur approach for solving the discrete matrix Riccati equation," *Proc. 18th Spring Conference of the Union of Bulgarian Mathematicians*, Albena, pp. 452–457, 1989.

Petkov P., Christov N.D., and Konstantinov M.M. "On the numerical properties of the Schur approach for solving the matrix Riccati equation," *Syst. Control Lett.*, Vol. 9, pp. 197–201, 1987.

Petkov P., Christov N.D., and Konstantinov M.M. *Computational Methods for Linear Control Systems*, Prentice Hall, London, 1991.

Petkov P., Konstantinov M.M., and Mehrmann V. DGRSVX and DMSRIC: *Fortran 77 Subroutines for Solving Continuous-time Matrix Algebraic Riccati Equations with Condition and Accuracy Estimates*. Tech. Report SFB393/98-116, Fakultät für Mathematik, TU Chemnitz, 09107 Chemnitz, FRG, 1998.

Petkov P., Gu D., Konstantinou M.M., and Mehrmann V. "Condition and Error Estimates in the Solution of Lyapunov and Riccati equations, *SLICOT Working Note*, 2000-1.

Potter J.E. "Matrix quadratic solutions," *SIAM J. Appl. Math.*, Vol. 14, pp. 496–501, 1966.

Quintana E. and Hernández V. *Algoritmos por bloques y paralelos para resolver ecuaciones matriciales de Riccati mediante el método de Newton*, Tech. Report DSIC-II/6/96, Dpto. de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia, Valencia, Spain, 1996a.

Quintana E. and Hernández V. *Algoritmos por bloques y paralelos para resolver ecuaciones matriciales de Riccati mediante el método de Schur*, Tech. Report DSIC-II/7/96, Dpto. de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia, Valencia, Spain, 1996b.

Quintana E. and Hernández V. *Algoritmos por bloques y paralelos para resolver ecuaciones matriciales de Riccati mediante la división espectral*, Tech. Report DSIC-II/6/96, Dpto de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia, Valencia, Spain, 1996c.

Roberts J. "Linear model reduction and solution of the algebraic Riccati equation by use of the sign function," *Int. J. Control*, Vol. 32, pp. 677–687, 1980 (reprint of a technical report form Cambridge University in 1971).

Sima V. *Algorithms for Linear-Quadratic Optimization*, Marcel Dekker, New York, 1996.

Sima V., Petkov P. and VanHuffel S. "Efficient and reliable algorithms for condition estimation of Lyapunov and Riccati equations," *Proc. Mathematical Theory of Networks and Systems* (MTNS – 2000), 2000.

Singer M.A. and Hammarling S.J. *The Algebraic Riccati Equation*, National Physical Laboratory Report, DITC 23/83, January, 1983.

Stewart G.W. "Algorithm 506-HQR3 and EXCHNG: Fortran subroutines for calculating and ordering the eigenvalues of a real upper Hessenberg matrix," *ACM Trans. Math. Soft.*, Vol. 2, pp. 275–280, 1976.

Sun J.-G. "Residual bounds of approximate solutions of the algebraic Riccati equations," *Numer. Math.*, Vol. 76, pp. 249–263, 1997a.

Sun J.-G. "Backward error for the discrete-time algebraic Riccati equation," *Lin. Alg. Appl.*, Vol. 25, pp. 183–208, 1997b.

Sun J.-G. "Perturbation analysis of the matrix sign function," *Lin. Alg. Appl.*, Vol. 250, pp. 177–206, 1997c.

Sun J.-G. "Perturbation theory for algebraic Riccati equation," *SIAM J. Matrix Anal. Appl.*, Vol. 19(1), pp. 39–65, 1998.

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

Van Dooren P. "Algorithm 590-DSUBSP and EXCHQZ: Fortran subroutines for computing deflating subspaces with specified spectrum," *ACM Trans. Math. Soft.*, Vol. 8, pp. 376–382, 1982.

Van Loan C.F. "A symplectic method for approximating all the eigenvalues of a Hamiltonian matrix," *Lin. Alg. Appl.*, Vol. 16, pp. 233–251, 1984.

Willems J.C. "Least squares stationary optimal control and the algebraic Riccati equation," *IEEE Trans. Autom. Control*, Vol. AC-16, pp. 621–634, 1971.

Wimmer H.K. "The algebraic Riccati equation: Conditions for the existence and uniqueness of solutions," *Lin. Alg. Appl.*, Vol. 58, pp. 441–452, 1984.

Wimmer H.K. "Existence of positive-definite and semidefinite solutions of discrete-time algebraic Riccati equations," *Int. J. Control*, Vol. 59, pp. 463–471, 1994.

Xu S.-F. "Sensitivity analysis of the algebraic Riccati equations," *Numer. Math.*, Vol. 75, pp. 121–134, 1996.

Zhang L.Q., Lam J., and Zhang Q.L. "Lyapunov and Riccati equations of discrete-time descriptor systems," *IEEE Trans. Autom. Control*, Vol. 44(1), pp. 2134–2139, 1999.

Zhou K. (with Doyle J.C. and Glover K.), *Robust and Optimal Control*, Prentice Hall, Upper Saddle River, NJ, 1996.