Hence, y_k can be found by solving

$$(s_{kk}H+I)y_k=\left[f_k-H\sum_{j=k+1}^{n}s_{kj}y_j\right].$$

The presence of 2×2 bumps on the diagonal of T can be handled in a fashion similar to what is done in the Hessenberg–Schur method.

This algorithm which we have sketched should be 30–70 percent faster than the Bartels–Stewart type technique in which both A and M are reduced to triangular form via the QR algorithm. (See [5].)

The second matrix equation problem we wish to consider involves finding $X\in R^{m\times n}$ such that

$$AXM+LXB=C \tag{7.3}$$

where $A,L\in R^{m\times m}$, $M,B\in R^{n\times n}$, and $C\in R^{m\times n}$. For a discussion of these and more general problems, see [7] and [13]. If M and L are nonsingular, then (7.3) can be put into "standard" $AX+XB=C$ form,

$$(L^{-1}A)X+X(BM^{-1})=L^{-1}CM^{-1}.$$

If M and/or L is poorly conditioned, it may make more numerical sense to apply the QZ algorithm of Moler and Stewart [8] to effect a stable transformation of (7.3). In particular, their techniques allow us to compute orthogonal U, V, Q, and Z such that

$$\begin{aligned}Q^TAU&=P \quad \text{(quasi-upper triangular)}\\ Q^TLU&=R \quad \text{(upper triangular)}\\ Z^TB^TV&=S \quad \text{(quasi-upper triangular)}\\ Z^TM^TV&=T \quad \text{(upper triangular).}\end{aligned}$$

If $Y=U^TXV$ and $F=Q^TCZ$, then (7.3) transforms to

$$PYT^T+RYS^T=F.$$

Comparing kth columns and assuming $s_{k,k-1}=T_{k,k-1}=0$ we find

$$P\sum_{j=k}^{n}t_{kj}y_j+R\sum_{j=k}^{n}s_{kj}y_j=f_k$$

and so

$$(t_{kk}P+s_{kk}R)y_k=f_k-P\sum_{j=k+1}^{n}t_{kj}y_j-R\sum_{j=k+1}^{n}s_{kj}y_j \tag{7.4}$$

This quasi-triangular system can then be solved for y_k once the right-hand side is known and under the assumption that the matrix $(t_{kk}P+s_{kk}R)$ is nonsingular. (Note that T, P, S, and R can all be singular without $t_{kk}P+s_{kk}R$ being singular.)

Now, as in the Hessenberg–Schur algorithm, significant economies can be made if A is only reduced to Hessenberg form. This is easily accomplished for when applied to the matrix pair (A,L), the QZ algorithm first computes orthogonal Q and U such that $Q^TAU=H$ is upper Hessenberg and $Q^TLU=R$ is upper triangular. The systems in (7.4) are now Hessenberg form and can consequently be solved very quickly. Again, we leave it to the reader to verify that the presence of 2×2 bumps on the diagonal of S pose no serious difficulties.

VIII. Conclusions

We have presented a new algorithm for solving the matrix equation $AX+XB=C$. The technique relies upon orthogonal matrix transformations and is not only extremely stable, but considerably faster than its nearest competitor, the Bartels–Stewart algorithm. We have included perturbation and roundoff analyses for the purpose of justifying the favorable performance of our method. Although these analyses are quite tedious, they are critical to the development of reliable software for this important computational problem.

References

[1] R. H. Bartels and G. W. Stewart, "A solution of the equation $AX+XB=C$," *Commun. ACM*, vol. 15, pp. 820–826, 1972.

[2] P. R. Belanger and T. P. McGillivray, "Computational experience with the solution of the matrix Lyapunov equations," *IEEE Trans. Automat. Contr.*, vol. AC-21, pp. 799–800, 1976.

[3] G. E. Forsythe and C. B. Moler, *Computer Solution of Linear Algebraic Systems*. Englewood Cliffs, NJ: Prentice-Hall, 1967.

[4] P. Hagander, "Numerical solution of $A^TS+SA+Q=0$," *Inform. Sci.*, vol. 4, pp. 35–40, 1972.

[5] G. Kitagawa, "An algorithm for solving the matrix equation $X=FXF^T+S$," *Int. J. Contr.*, vol. 25, pp. 745–753, 1977.

[6] G. Kreisselmeier, "A solution of the bilinear matrix equation $AY+YB=-Q$," *SIAM J. Appl. Math.*, vol. 23, pp. 334–338, 1973.

[7] P. Lancaster, "Explicit solutions of linear matrix equations," *SIAM Rev.*, vol. 12, pp. 544–566, 1970.

[8] C. B. Moler and G. W. Stewart, "An algorithm for generalized matrix eigenvalue problems," *SIAM J. Numer. Anal.*, vol. 10, pp. 241–256, 1973.

[9] B. P. Molinari, "Algebraic solution of matrix linear equations in control theory," *Proc. Inst. Elec. Eng.*, vol. 116, pp. 1748–1754, 1969.

[10] D. Rothschild and A. Jameson, "Comparison of four numerical algorithms for solving the Lyapunov matrix equation," *Int. J. Contr.*, vol. 11, pp. 181–198, 1970.

[11] B. T. Smith *et al.*, *Matrix Eigensystem Routines—EISPACK Guide* (Lecture Notes in Computer Science). New York: Springer-Verlag, 1970.

[12] J. H. Wilkinson, *The Algebraic Eigenvalue Problem*. Oxford, England: Oxford Univ. Press, 1965.

[13] H. Wimmer and A. D. Ziebur, "Solving the matrix equation $\sum_{p=1}^{r}f_p(A)Xg_p(B)=C$," *SIAM Rev.*, vol. 14, pp. 318–323, 1972.

[14] A. K. Cline, C. B. Moler, G. W. Stewart, and J. H. Wilkinson, "An estimate for the condition number of a matrix," *SIAM J. Numer. Anal.*, vol. 16, pp. 368–375, 1979.

A Schur Method for Solving Algebraic Riccati Equations

ALAN J. LAUB, MEMBER, IEEE

***Abstract*—In this paper a new algorithm for solving algebraic Riccati equations (both continuous-time and discrete-time versions) is presented. The method studied is a variant of the classical eigenvector approach and uses instead an appropriate set of Schur vectors, thereby gaining substantial numerical advantages. Considerable discussion is devoted to a number of numerical issues. The method is apparently quite numerically stable and performs reliably on systems with dense matrices up to order 100 or so, storage being the main limiting factor.**

I. Introduction

In this paper a new algorithm for solving algebraic Riccati equations (both continuous-time and discrete-time versions) is presented. These equations play fundamental roles in the analysis, synthesis, and design of linear-quadratic-Gaussian control and estimation systems as well as in many other branches of applied mathematics. It is not the purpose of this paper to survey the extensive literature available for these equations but, rather, we refer the reader to, for example, [1]–[5] for references. Nor is it our intention to investigate any but the unique (under suitable hypotheses) symmetric, nonnegative definite solution of an algebraic Riccati equation even though the algorithm to be presented does also have the potential to produce other solutions. For further reference to the "geometry" of the Riccati equation we refer to [3], [6], and [7].

The method studied here is a variant of the classical eigenvector approach to Riccati equations, the essentials of which date back to at least von Escherich in 1898 [8]. The approach has also found its way into the control literature in papers by, for example, MacFarlane [9], Potter [10], and Vaughn [11]. Its use in that literature is often associated with

Manuscript received November 21, 1978; revised July 9, 1979. Paper recommended by E. Polak, Past Chairman of the Computational Methods and Discrete Systems Committee. This work was supported by the U.S. Department of Energy under Contract ERDA-E(49-18)-2087.

The author was with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA 02139. He is now with the Department of Electrical Engineering–Systems, University of Southern California, Los Angeles, CA 90007.

0018-9286/79/1200-0913$00.75

the name of Potter. However, the use of eigenvectors is often highly unsatisfactory from a numerical point of view and the present method uses the so-called and much more numerically attractive Schur vectors to get a basis for a certain subspace of interest in the problem.

Other authors such as Fath [12] and Willems [3], to name two, have also noted that any basis of the subspace would suffice but the specific use of Schur vectors was inhibited by a not entirely straightforward problem of ordering triangular canonical forms—a problem which is discussed briefly in the sequel. The paper by Fath is very much in the spirit of the work presented here and is one of the very few in the literature which seriously addresses numerical issues.

One of the best summaries of the eigenvector approach to solving algebraic Riccati equations is the work of Martensson [13]. This work extends [10] to the case of "multiple closed-loop eigenvalues." It will be shown in the sequel how the present approach recovers all the theoretical results of [10] and [13] while providing significant numerical advantages.

Most numerical comparisons of Riccati algorithms tend to favor the standard eigenvector approach—its numerical difficulties notwithstanding—over other approaches such as Newton's method [14] or methods based on integrating a Riccati differential equation. Typical of such comparisons are [7], [15], and [16]. It will be demonstrated in this paper that if you previously liked the eigenvector approach, you must prefer, almost by definition, the Schur vector approach. This statement, while somewhat simplistic, is based on the fact that a Schur vector approach provides a substantially more efficient, useful, and reliable technique for numerically solving algebraic Riccati equations. The method is intended primarily for the solution of dense, moderate-sized equations (say, order < 100) rather than large, sparse equations. While the algorithm in its present state offers much scope for improvement, it still represents a substantial improvement over current direct methods for solving algebraic Riccati equations.

Briefly, the rest of the paper is organized as follows. This section is concluded with some notation and linear algebra review. In Sections II and III the continuous-time and discrete-time Riccati equations, respectively, are treated. In Section IV numerical issues such as algorithm implementation, balancing, scaling, operation counts, timing, storage, stability, and conditioning are considered. In Section V we emphasize the advantages of the Schur vector approach and make some further general remarks. Six examples are given in Section VI and some concluding remarks are made in Section VII.

A. Notation

Throughout the paper $A \in \mathbb{F}^{m\times n}$ will denote an $m\times n$ matrix with coefficients in a field $\mathbb{F}$. The field will usually be the real numbers $\mathbb{R}$ or the complex numbers $\mathbb{C}$. The notations A^T and A^H will denote transpose and conjugate transpose, respectively, while A^{-T} will denote $(A^T)^{-1} = (A^{-1})^T$. The notation A^+ will denote the Moore–Penrose pseudoinverse of the matrix A. For $A \in \mathbb{R}^{n\times n}$ its spectrum (set of n eigenvalues) will be denoted by $\sigma(A)$. When a matrix $A \in \mathbb{R}^{2n\times 2n}$ is partitioned into four $n\times n$ blocks as

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}$$

we shall frequently refer to the individual blocks A_{ij} without further discussion.

B. Linear Algebra Review

Definition 1: $A \in \mathbb{R}^{n\times n}$ is *orthogonal* if $A^T = A^{-1}$.
Definition 2: $A \in \mathbb{C}^{n\times n}$ is *unitary* if $A^H = A^{-1}$.

Let $J = \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix} \in \mathbb{R}^{2n\times 2n}$ where I denotes the nth order identity matrix. Note that $J^T = J^{-1} = -J$.

Definition 3: $A \in \mathbb{R}^{2n\times 2n}$ is *Hamiltonian* if $J^{-1}A^TJ = -A$.
Definition 4: $A \in \mathbb{R}^{2n\times 2n}$ is *symplectic* if $J^{-1}A^TJ = A^{-1}$.

Hamiltonian and symplectic matrices are obviously closely related. For a discussion of this relationship and a review of "symplectic algebra" see [17], [18]. We will use the following two theorems from symplectic algebra. Their proofs (see [18]) are trivial (and hence will be omitted).

Theorem 1: 1) Let $A \in \mathbb{R}^{2n\times 2n}$ be Hamiltonian. Then $\lambda \in \sigma(A)$ implies $-\lambda \in \sigma(A)$ with the same multiplicity. 2) Let $A \in \mathbb{R}^{2n\times 2n}$ be symplectic. Then $\lambda \in \sigma(A)$ implies $1/\lambda \in \sigma(A)$ with the same multiplicity.

There is a relationship between the right and left eigenvectors of these symplectically associated eigenvalues. See [18] for details.

Theorem 2: Let $A \in \mathbb{R}^{2n\times 2n}$ be Hamiltonian (or symplectic). Let $U \in \mathbb{R}^{2n\times 2n}$ be symplectic. Then $U^{-1}AU$ is Hamiltonian (or symplectic).

Finally, we need two theorems from classical similarity theory which form the theoretical cornerstone of modern numerical linear algebra. See [19], for example, for a textbook treatment.

Theorem 3 (Schur Canonical Form): Let $A \in \mathbb{R}^{n\times n}$ have eigenvalues $\lambda_1, \cdots, \lambda_n$. Then there exists a unitary similarity transformation U such that U^HAU is upper triangular with diagonal elements $\lambda_1, \cdots, \lambda_n$ in that order.

In fact, it is possible to work only over $\mathbb{R}$ by reducing to quasi-upper-triangular form with 2×2 blocks on the (block) diagonal corresponding to complex conjugate eigenvalues and 1×1 blocks corresponding to the real eigenvalues. We refer to this canonical form as the real Schur form (RSF) or the Murnaghan–Wintner [20] canonical form.

Theorem 4 (RSF): Let $A \in \mathbb{R}^{n\times n}$. Then there exists an orthogonal similarity transformation U such that U^TAU is quasi-upper-triangular. Moreover, U can be chosen so that the 2×2 and 1×1 diagonal blocks appear in any desired order.

If in Theorem 4 we partition U^TAU into $\begin{pmatrix} S_{11} & S_{12} \\ 0 & S_{22} \end{pmatrix}$ where $S_{11} \in \mathbb{R}^{k\times k}$, $0 < k < n$, we shall refer to the first k vectors of U as the *Schur vectors* corresponding to $\sigma(S_{11}) \subseteq \sigma(A)$. The Schur vectors corresponding to the eigenvalues of S_{11} span the eigenspace corresponding to those eigenvalues even when some of the eigenvalues are multiple (see [21]). We shall use this property heavily in the sequel.

II. The Continuous-Time Algebraic Riccati Equation

In this section we shall present a method for using a certain set of Schur vectors to solve (for X) the continuous-time algebraic Riccati equation

$$F^TX + XF - XGX + H = 0. \tag{1}$$

All matrices are in $\mathbb{R}^{n\times n}$ and $G = G^T \geq 0$, $H = H^T \geq 0$.

It is assumed that (F, B) is a stabilizable pair [1] where B is a full-rank factorization (FRF) of G [i.e., $BB^T = G$ and $\operatorname{rank}(B) = \operatorname{rank}(G)$] and (C, F) is a detectable pair [1] where C is a FRF of H [i.e., $C^TC = H$ and $\operatorname{rank}(C) = \operatorname{rank}(H)$]. Under these assumptions, (1) is known to have a unique nonnegative definite solution [1]. There are, of course, other solutions to (1) but for the algorithm presented here the emphasis will be on computing the nonnegative definite one.

Now consider the Hamiltonian matrix

$$Z = \begin{pmatrix} F & -G \\ -H & -F^T \end{pmatrix} \in \mathbb{R}^{2n\times 2n}. \tag{2}$$

Our assumptions guarantee that Z has no pure imaginary eigenvalues. Thus, by Theorem 4 we can find an orthogonal transformation $U \in \mathbb{R}^{2n\times 2n}$ which puts Z in RSF:

$$U^TZU = S = \begin{pmatrix} S_{11} & S_{12} \\ 0 & S_{22} \end{pmatrix} \tag{3}$$

where $S_{ij} \in \mathbb{R}^{n\times n}$. It is possible to arrange, moreover, that the real parts of the spectrum of S_{11} are negative while the real parts of the spectrum of S_{22} are positive. U is conformably partitioned into four $n\times n$ blocks:

$$U = \begin{pmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{pmatrix}. \tag{4}$$

We then have the following theorem.

Theorem 5: With respect to the notation and assumptions above,
1) U_{11} is invertible and $X = U_{21}U_{11}^{-1}$ solves (1) with $X = X^T \geq 0$;
2) $\sigma(S_{11}) = \sigma(F - GX)$ = the "closed-loop" spectrum.

Proof: A direct proof of this theorem may be found in [22] but will be omitted here. An alternate proof is suggested in Remark 1 below. ■

Remark 1: As an alternative to the direct proofs provided in [22] one could simply appeal to the proofs given for the eigenvector approach and note that the Schur vectors are related to the eigenvectors by a nonsingular transformation. Specifically, with Z, U, and S as above, let $V \in \mathbb{R}_{2n}^{2n\times 2n}$ put Z in real Jordan form

$$V^{-1}ZV = \begin{pmatrix} -\Lambda & 0 \\ 0 & \Lambda \end{pmatrix} \tag{5}$$

($\mathbb{R}_{2n}^{2n\times 2n}$ denotes the set of $2n\times 2n$ matrices of rank $2n$, i.e., invertible) where $-\Lambda$ is the real Jordan form of the eigenvalues of Z with negative real parts. Furthermore, let $T \in \mathbb{R}_n^{n\times n}$ transform S_{11} to the real Jordan form $-\Lambda$. Then

$$Z\begin{pmatrix} V_{11} \\ V_{21} \end{pmatrix} = \begin{pmatrix} V_{11} \\ V_{21} \end{pmatrix}(-\Lambda) \tag{6}$$

and

$$Z\begin{pmatrix} U_{11} \\ U_{21} \end{pmatrix} = \begin{pmatrix} U_{11} \\ U_{21} \end{pmatrix} S_{11}.$$

We thus have

$$Z\begin{pmatrix} U_{11} \\ U_{21} \end{pmatrix} T = \begin{pmatrix} U_{11} \\ U_{21} \end{pmatrix} TT^{-1}S_{11}T = \begin{pmatrix} U_{11} \\ U_{21} \end{pmatrix} T(-\Lambda). \tag{7}$$

Since eigenvectors are unique up to nonzero scalar multiple, comparing (6) and (7) we must have

$$\begin{pmatrix} U_{11} \\ U_{21} \end{pmatrix} T = \begin{pmatrix} V_{11} \\ V_{21} \end{pmatrix} D$$

where D is diagonal and invertible. Thus,

$$\begin{pmatrix} U_{11} \\ U_{21} \end{pmatrix} = \begin{pmatrix} V_{11} \\ V_{21} \end{pmatrix} DT^{-1}$$

and since $V_{21}V_{11}^{-1}$ solves (1), $U_{21}U_{11}^{-1}$ must also solve (1) since

$$U_{21}U_{11}^{-1} = V_{21}DT^{-1}(V_{11}DT^{-1})^{-1} = V_{21}V_{11}^{-1}.$$

Further discussion of Theorem 5 and computational considerations are deferred until Section IV.

III. The Discrete-Time Algebraic Riccati Equation

In this section we shall present an analogous method using certain Schur vectors to solve the discrete-time algebraic Riccati equation

$$F^TXF - X - F^TXG_1(G_2 + G_1^TXG_1)^{-1}G_1^TXF + H = 0. \tag{8}^1$$

Here $F, H, X \in \mathbb{R}^{n\times n}$, $G_1 \in \mathbb{R}^{n\times m}$, $G_2 \in \mathbb{R}^{m\times m}$, and $H = H^T \geq 0$, $G_2 = G_2^T > 0$. Also, $m < n$. The details of the method for this equation are sufficiently different from the continuous-time case that we shall explicitly present most of them.

It is assumed that (F, G_1) is a stabilizable pair and that (C, F) is a detectable pair where C is a FRF of H [i.e., $C^TC = H$ and rank(C) = rank(H)]. We also assume that F is invertible—a common assumption on the open-loop dynamics of a discrete-time system [23]. The details for the case when F is singular or ill-conditioned with respect to inversion can be found in [24].

Under the above assumptions (8) is known to have a unique nonnegative definite solution [25] and the method proposed below will be directed towards finding that solution.

Setting $G = G_1G_2^{-1}G_1^T$ we consider this time the symplectic matrix

$$Z = \begin{pmatrix} F + GF^{-T}H & -GF^{-T} \\ -F^{-T}H & F^{-T} \end{pmatrix}. \tag{9}$$

Our assumptions guarantee that Z has no eigenvalues on the unit circle. By Theorem 4 we can find an orthogonal transformation $U \in \mathbb{R}^{2n\times 2n}$ which puts Z in RSF:

$$U^TZU = S = \begin{pmatrix} S_{11} & S_{12} \\ 0 & S_{22} \end{pmatrix} \tag{10}$$

where $S_{ij} \in \mathbb{R}^{n\times n}$.

It is possible to arrange, moreover, that the spectrum of S_{11} lies inside the unit circle while the spectrum of S_{22} lies outside the unit circle. Again U is partitioned conformably. We then have the following theorem.

Theorem 6: With respect to the notation and assumptions above,
1) U_{11} is invertible and $X = U_{21}U_{11}^{-1}$ solves (8) with $X = X^T > 0$;

2) $$\begin{aligned} \sigma(S_{11}) &= \sigma\left(F - G_1(G_2 + G_1^TXG_1)^{-1}G_1^TXF\right) \\ &= \sigma(F - GF^{-T}(X - H)) \\ &= \sigma\left(F - G(X^{-1} + G)^{-1}F\right) \qquad \text{when } X \text{ is invertible} \\ &= \text{the "closed-loop" spectrum.} \end{aligned}$$

Proof: As for Theorem 5 we shall omit a direct proof of this theorem and refer the interested reader instead to [22] or [24]. Again an alternate proof is possible as in Remark 1. ■

We now turn to some general numerical considerations regarding the Schur vector approach.

IV. Numerical Considerations

There are two steps to the Schur vector approach. The first is reduction of a $2n\times 2n$ matrix to an ordered real Schur form; the second is the solution of an nth order linear matrix equation. We shall discuss these in the context of the continuous-time case noting differences for the discrete-time case where appropriate.

A. Algorithm Implementation

It is well known (see [21], for example) that the double Francis QR algorithm applied to a real general matrix does not guarantee any special order for the eigenvalues on the diagonal of the Schur form. However, it is also known how the real Schur form can be arbitrarily reordered via orthogonal similarities; see [21] for details. Thus, any further orthogonal similarities required to ensure that $\sigma(S_{11})$ in (3) lies in the left-half complex plane can be combined with the U initially used to get a RSF to get a final orthogonal matrix which effects the desired ordered RSF.

Stewart has recently published Fortran subroutines for calculating and ordering the RSF of a real upper Hessenberg matrix [26]. The 1×1 or 2×2 blocks are ordered so that the eigenvalues appear in descending order of magnitude along the diagonal. Stewart's software (HQR3) may thus be used directly if one is willing to first apply to the Z of (2) an appropriate bilinear transformation which maps the left-half plane to the exterior of the unit circle. Since the transformed Z is an analytic function of Z, the U that reduces it to an ordered RSF—with half the eigenvalues outside the unit circle—is the desired U from which the solution of (1) may be constructed. Alternatively, Stewart's software can be modified to directly reorder a RSF by algebraic sign.

In the discrete-time case, HQR3 can be used directly by working with

$$Z^{-1} = \begin{pmatrix} F^{-1} & F^{-1}G \\ HF^{-1} & F^T + HF^{-1}G \end{pmatrix}. \tag{11}$$

The U which puts $\sigma(S_{11})$ outside the unit circle is thus the same U which

[1]Note that an alternate equivalent form of (8) when X is invertible is $F^T(X^{-1} + G_1G_2^{-1}G_1^T)^{-1}F - X + H = 0$.

puts the upper left $n \times n$ block of the RSF of Z inside the unit circle.

In summary then, to use HQR3 we would recommend using the following sequence of subroutines (or their equivalents):

BALANC	to balance a real general matrix
ORTHES	to reduce the balanced matrix to upper Hessenberg form using orthogonal transformations
ORTRAN	to accumulate the transformations from the Hessenberg reduction
HQR3	to determine an ordered RSF from the Hessenberg matrix
BALBAK	to backtransform the orthogonal matrix to a nonsingular matrix corresponding to the original matrix.

The subroutines BALANC, ORTHES, ORTRAN, and BALBAK are all available in EISPACK [27].

The second step to be implemented is the solution of an nth order linear matrix equation

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

to find $X = U_{21}U_{11}^{-1}$. For this step we would recommend a good linear equation solver such as DECOMP and SOLVE available in [28] or the appropriate routines available in LINPACK [29]. A routine such as DECOMP computes the LU-factorization of U_{11} and SOLVE performs the forward and backward substitutions. A good estimate of the condition number of U_{11} with respect to inversion is available with good linear equation software and this estimate should be inspected. A badly conditioned U_{11} usually results from a "badly conditioned Riccati equation." This matter will be discussed further in Section IV-D. While we have no analytical proof at this time, we have observed empirically that a condition number estimate on the order of 10^t for U_{11} usually results in a loss of about t digits of accuracy in X.

One final note on implementation. Since X is symmetric it is usually more convenient, with standard linear equation software, to solve the equation

$$U_{11}^T X = U_{21}^T$$

to find $X = U_{11}^{-T}U_{21}^T = U_{21}U_{11}^{-1}$.

B. Balancing and Scaling

Note that the use of balancing in the above implementation results in a nonsingular (but not necessarily orthogonal) matrix which reduces Z to RSF. More specifically, suppose P is a permutation matrix and D is a diagonal matrix such that PD balances Z, i.e.,

$$D^{-1}P^TZPD = Z_b$$

where Z_b is the balanced matrix; see [30] for details. We then find an orthogonal matrix U which reduces Z_b to ordered RSF:

$$U^TZ_bU = S.$$

Then PDU (produced by BALBAK) is clearly a nonsingular matrix which reduces Z to ordered RSF. The first n columns of PDU span the eigenspace corresponding to eigenvalues of Z with negative real parts and that is the only property we require of the transformation. For simplicity in the sequel, we shall speak of the transformation reducing Z to RSF as simply an orthogonal matrix U with the understanding that the more computationally attractive transformation is of the form PDU.

An alternative approach to direct balancing of Z is to attempt some sort of scaling in the problem which generates the Riccati equation. To illustrate, consider the linear optimal control problem of finding a feedback controller $u(t) = Kx(t)$ which minimizes the performance index

$$J(u) = \int_0^{+\infty} [x^T(t)Hx(t) + u^T(t)Ru(t)]\,dt$$

with plant constraint dynamics given by

$$\dot{x}(t) = Fx(t) + Bu(t); \qquad x(0) = x_0.$$

We assume $H = H^T \geq 0$, $R = R^T > 0$, and (F, B) controllable, (F, C) observable where $C^TC = H$ and $\operatorname{rank}(C) = \operatorname{rank}(H)$. Then the optimal solution is well known to be

$$u(t) = -R^{-1}B^TXx$$

where X solves the Riccati equation

$$F^TX + XF - XBR^{-1}B^TX + H = 0.$$

Now suppose we change coordinates via a nonsingular transformation $x(t) = Tw(t)$. Then in terms of the new state w our problem is to minimize

$$\int_0^{+\infty} [w^T(t)(T^THT)w(t) + u^T(t)Ru(t)]\,dt$$

subject to

$$\dot{w}(t) = (T^{-1}FT)w(t) + (T^{-1}B)u(t).$$

The Hamiltonian matrix Z for this transformed system is now given by

$$Z_w = \begin{pmatrix} T^{-1}FT & -T^{-1}BR^{-1}B^TT^{-T} \\ -T^THT & -T^TF^TT^{-T} \end{pmatrix}$$

and the associated solution X_w of the transformed Riccati equation is related to the original X by $X = T^{-T}X_wT^{-1}$. One interpretation of T then is as a scaling transformation, a diagonal matrix, for example, in an attempt to "balance" the elements of Z_w. Applying such a procedure, even in an ad hoc way, is frequently very useful from a computational point of view.

Another way to look at the above procedure is that Z_w is symplectically similar to Z via the transformation $\begin{pmatrix} T & 0 \\ 0 & T^{-T} \end{pmatrix}$, i.e.,

$$Z_w = \begin{pmatrix} T & 0 \\ 0 & T^{-T} \end{pmatrix}^{-1} Z \begin{pmatrix} T & 0 \\ 0 & T^{-T} \end{pmatrix}.$$

It is well known that Z_w is again Hamiltonian (or symplectic in the discrete-time case) since the similarity transformation is symplectic. One can then pose the problem of transforming Z by other symplectic similarities, say orthogonal, so as to achieve various desirable numerical properties or canonical forms. This topic for further research is presently being investigated.

C. Operation Counts, Timing, and Storage

We shall give approximate operation counts for the solution of nth order algebraic Riccati equations of the form (1) or (8). Each operation is assumed to be roughly equivalent to forming $a + (b \times c)$ where a, b, c are floating-point numbers. It is almost impossible to give an accurate operation count for the algorithm described above since so many factors are variable such as the ordering of the RSF. We shall indicate only a ballpark $0(n^3)$ figure.

Let us assume then that we already have at hand the $2n \times 2n$ matrix Z of the form (2) or (9). Note, however, that unlike forming Z in (2), Z in (9) requires approximately $4n^3$ additional operations to construct, given only F, G, and H. This will turn out to be fairly negligible compared to the counts for the overall process. Furthermore, we shall give only order of n^3 counts for these rough estimates. The three main steps are:

	Operations
i) reduction of Z to upper Hessenberg from	$\frac{5}{3}(2n)^3$
ii) reduction of upper Hessenberg form to RSF	$> 4k(2n)^3$
iii) solution of $XU_{11} = U_{21}$	$\frac{4}{3}n^3$.

The number k represents the average number of QR steps required per eigenvalue and is usually overestimated by 1.5. We write $> 4k(2n)^3$ since, in general, the reduction may need more operations if ordering is required. Using $k = 1.5$ we see that the total number of operations required is at least $63n^3$. Should the ordering of the RSF require, say, 25

percent more operations than the unordered RSF, we have a ballpark estimate of about 75 n^3 for the entire process.

Timing estimates for steps i) and ii) may be obtained from [27] for a variety of computing environments. The additional time for balancing and for step iii) would then add no more than about 5 percent to those times while the additional time for ordering the RSF is variable, but typically adds no more than about 15 percent. For example, adding 20 percent to the published figures [27] for an IBM 370/165 (a typical medium speed machine) under OS/360 at the University of Toronto using Fortran H Extended with Opt. = 2 and double precision arithmetic, we can construct the following table:

Riccati equation order $n=$	10	20	30	40
CPU time (s)	0.2	1.3	4.0	9.0

In fact, these times are in fairly close agreement with actual observed times for randomly chosen test examples of these orders. Note the approximately cubic behavior of time versus order.

Extrapolating these figures for a 64th-order equation (see Example 5 in Section VI) one might expect a CPU time in the neighborhood of 38 s. In fact, for that particular example the time was approximately 34 s.

It must be reemphasized here that timing estimates derived as above are very approximate and depend on numerous factors in the actual computing environment as well as the particular input data. However, such estimates can provide very useful and quite reliable information if interpreted as providing essentially order of magnitude figures.

With respect to storage considerations the algorithm requires $8n^2 + cn$(c = a small constant) storage locations. This fairly large figure limits applicability of the algorithm to Riccati equations on the order of about 100 or less in many common computing environments. Of course, CPU time becomes a significant factor for $n > 100$, also.

D. Stability and Conditioning

This section will be largely speculative in nature as very few hard results are presently available. A number of areas of continuing research will be described.

With respect to stability, the implementation discussed in Section IV-A consists of two effectively stable steps. The crucial step is the QR step and the present algorithm is probably essentially as stable as QR. The overall two step process is apparently quite stable numerically but we have no proof of that statement.

Concerning the conditioning of (1) [or (8)] almost no analytical results are known. The study of (1) is obviously more complex than the study of even the Lyapunov equation

$$F^TX + XF + H = 0 \tag{12}$$

where $H = H^T \geq 0$. And yet very little numerical analysis is known for (12). In case F is normal, a condition number with respect to inversion of the Lyapunov operator $\mathcal{F}X = F^TX + XF$ is easily shown to be given by

$$\frac{\max_{i,j} |\lambda_i(F) + \lambda_j(F)|}{\min_{i,j} |\lambda_i(F) + \lambda_j(F)|}.$$

But in the general case, a condition number in terms of F rather than $F^T \otimes I + I \otimes F^T$ ($\otimes$ denotes Kronecker product) has not been determined. Some empirical observations on the accuracy of solutions of certain instances of (12) suggest that one factor influencing conditioning of (12) is the proximity of the spectrum of F to the imaginary axis. To be more specific, suppose F has an eigenvalue at $a \pm jb$ with $\left|\frac{b}{a}\right| \gg 1$ (typically $a < 0$ is very small). If $\left|\frac{b}{a}\right| = 0(10^t)$ we lose approximately t digits of accuracy and we might expect a condition number for the solution of (12) to also be $0(10^t)$ in this situation.

There are some close connections between (12) and (1) (and the respective discrete-time versions) and we shall indicate some preliminary observations here. A perturbation analysis or the notion of a condition number for (1) is intimately related to the condition of an associated Lyapunov equation, namely one whose "F-matrix" approximates the closed-loop matrix $F - GX$ where X solves (1). To illustrate, suppose $X = Y + E$ where $Y = Y^T$ may be interpreted as an approximation of X. Then

$$\begin{aligned} 0 &= F^T(Y+E) + (Y+E)F - (Y+E)G(Y+E) + H \\ &\approx (F - GY)^TE + E(F - GY) + (F^TY + YF - YGY + H) \\ &= \hat{F}^TE + E\hat{F} + \hat{H} \end{aligned}$$

where we have neglected the second-order term EGE. Thus, conditioning of (1) should be closely related to nearness of the closed-loop spectrum $[\sigma(F - GX)]$ to the imaginary axis. Observations similar to these have been made elsewhere; see, for example, Bucy [31] where the problem is posed as one of structural stability. A condition number might, in some sense, be thought of as a quantitative measure of the degree of structural stability.

Another factor involved in the conditioning of (1) relates to the assumptions of stabilizability of (F, B) and the detectability of (C, F). For example, near-unstabilizability of (F, B) in either a parametric sense or in a control energy sense (i.e., near-singular controllability Gramian) definitely causes (1) to become badly conditioned. Our experience has been that the ill-conditioning manifests itself in the algorithm by a badly conditioned U_{11}.

Work related to the conditioning of (1) and (8) is under continuing investigation and will be the subject of another paper. Such analysis is, of course, independent of the particular algorithm used to solve (1) or (8), but is useful to understand how ill-conditioning can be expected to manifest itself in a given algorithm.

V. Advantages of the Schur Vector Approach and Further General Remarks

A. Advantages and Disadvantages of the Schur Vector Approach

The advantages of this algorithm over others using eigenvectors (such as Potter's approach [10] and its extensions) are obvious. Firstly, the reduction to RSF is an intermediate step in computing eigenvectors anyway (using the double Francis QR algorithm) so the Schur approach must, by definition, be faster. Secondly, and more importantly, this algorithm will not suffer as severely from the numerical hazards inherent in computing eigenvectors associated with multiple or near-multiple eigenvalues. The computation of eigenvectors is fraught with difficulties (see, e.g., [21] for a cogent discussion) and the eigenvectors themselves are simply not needed. All that is needed is a basis for the eigenspace spanned by the eigenvalues of Z with negative real parts (with an analogous statement for the discrete-time case). As good a basis as is possible (in the presence of rounding error) for this subspace can be found from the Schur vectors comprising the matrix $\begin{pmatrix} U_{11} \\ U_{21} \end{pmatrix}$, independently of individual eigenvalue multiplicities. The reader is strongly urged to consult [32] and [21] (especially pp. 609–610) for further numerical details.

The fact that any basis for the stable eigenspace can be used to construct the Riccati equation solution has been noted by many people; see [12] or [3] among others. The main stumbling block with using the Schur vectors was the ordering problem with the RSF but once that is handled satisfactorily the algorithm is easy.

The Schur vector approach derives its desirable numerical properties from the underlying QR-type process. To summarize: if you like the eigenvector approach for solving the algebraic Riccati equation you must like the Schur vector approach better.

Like the eigenvector approach, the Schur vector approach has the advantage of producing the close-loop eigenvalues (or whatever is appropriate to the particular application from which the Riccati equation arises) essentially for free. And finally, an important advantage of the Schur vector approach, in addition to its general reliability for engineering applications, is its speed in comparison with other methods. We have already mentioned the advantage, by definition, over previous eigenvec-

tor approaches but there is also generally an even more significant speed advantage over iterative methods. This advantage is particularly apparent in poorly conditioned problems and in cases in which the iterative method has a bad starting value. Of course, it is impossible to make the comparison between a direct versus iterative method any more precise for general problems but we have found it not at all uncommon for an iterative method, such as straightforward Newton [14], to take ten to thirty times as long—if, indeed, there was convergence at all.

As mentioned above, a possible disadvantage of the method is the storage requirement of at least two $2n \times 2n$ arrays. Another disadvantage is the fact that for the computed $X = U_{21}U_{11}^{-1}$ there is no guarantee of symmetry. In practice, we have found the deviation from symmetry to be only slight for most problems, becoming more pronounced only when the Riccati equation was known to be "ill-conditioned." This phenomenon might be used advantageously if the deviation from perfect symmetry could be used to reliably monitor conditioning.

B. Miscellaneous General Remarks

Remark 2: An nth order algebraic Riccati equation has a finite number of solutions if the characteristic and minimal polynomials of $F-GX$ are equal where X is the unique nonnegative definite solution. In that case there are still as many as $\binom{2n}{n}$ solutions corresponding to as many as $\binom{2n}{n}$ choices of n of the $2n$ eigenvalues of Z. Any of these solutions may also be generated by the Schur approach, as for the eigenvector approach, by an appropriate reordering of the RSF. For most control and filtering applications we are interested in the unique nonnegative definite solution and we have thus concentrated the exposition on that particular case.

Remark 3: One of the most complete sources for an eigenvector-oriented proof of Theorem 5 for the general case of multiple eigenvalues is Martensson [13]. But even a casual glance at that proof exposes the awkwardness of fussing with eigenvectors and principal vectors. The proof using Schur vectors is extremely clean and easy by comparison and neatly avoids any difficulties with multiple eigenvalues. This observation is but one instance of the more general observation that *Schur vectors can probably always replace principal vectors (or generalized eigenvectors) corresponding to multiple eigenvalues throughout linear control/systems theory.* Principal vectors are not generally reliably computable in the presence of roundoff error anyway (see [21]) and a basis for an eigenspace—but not the particular one corresponding to the principal vectors—is all that is normally needed. Use of Schur vectors will not only frequently provide cleaner proofs by is also numerically much more attractive.

Remark 4: The same Schur vector approach employed in this paper can also be used instead of the eigenvector approach for the nonsymmetric matrix quadratic equation

$$XEX + FX + XG + H = 0$$

where $E \in \mathbb{R}^{m\times n}$, $F \in \mathbb{R}^{n\times n}$, $G \in \mathbb{R}^{m\times m}$, $H \in \mathbb{R}^{n\times m}$, and $X \in \mathbb{R}^{n\times m}$. In this case, we work with the $(m+n)\times(m+n)$ matrix

$$Z = \begin{pmatrix} -G & -E \\ H & F \end{pmatrix} \tag{13}$$

and various solutions of (13) are determined by generating appropriate combinations of m eigenvalues of Z along the diagonal of the RSF of Z. The corresponding m Schur vectors give the solution $X = U_{21}U_{11}^{-1}$ as before where $U_{11} \in \mathbb{R}^{m\times m}$, $U_{21} \in \mathbb{R}^{n\times m}$. The analogous remarks apply for the corresponding nonsymmetric "discrete-time equation." Proofs are essentially the same in both cases. Further details on the eigenvector approach can be found in [33], [34].

Remark 5: Special cases of the matrix quadratic equations such as (1), (8), or (13) include the Lyapunov equation (12) (or its discrete-time counterpart $F^TXF - X + H = 0$) and the Sylvester equation

$$FX + XG + H = 0 \tag{14}$$

(or its discrete-time counterpart $FXG - X + H = 0$).

Thus, setting an appropriate block of the Z matrix equal to 0 provides a method of solving such "linear equations" and, in fact, this method has even been proposed in the literature [35]. However, the approach probably has little to recommend it from a numerical point of view as compared to applying the Bartels–Stewart algorithm [39] and we mention it only in passing.

VI. Examples

In this section we give a few examples both to illustrate various points discussed previously and to provide some numerical results for comparison with other approaches. All computations were done at the Massachusetts Institute of Technology on an IBM 370/168 using Fortran H Extended (Opt. = 2) and double precision arithmetic.

Example 1: The Schur vector approach is obviously not well-suited to hand computation—which partly explains its desirable numerical properties. However, to pacify a certain segment of the population a "hand example" is provided in complete detail. Consider the equation

$$A^TX + XA - XBR^{-1}B^TX + Q = 0 \tag{15}$$

which arises in a linear-quadratic optimal control context with

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

Then

$$Z = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ -1 & 0 & 0 & 0 \\ 0 & -2 & -1 & 0 \end{bmatrix}$$

and the matrix

$$U = \begin{bmatrix} \frac{1}{2} & -\frac{\sqrt{5}}{10} & -\frac{3\sqrt{5}}{10} & \frac{1}{2} \\ -\frac{1}{2} & -\frac{\sqrt{5}}{10} & -\frac{3\sqrt{5}}{10} & -\frac{1}{2} \\ \frac{1}{2} & -\frac{3\sqrt{5}}{10} & \frac{\sqrt{5}}{10} & -\frac{1}{2} \\ -\frac{1}{2} & -\frac{3\sqrt{5}}{10} & \frac{\sqrt{5}}{10} & \frac{1}{2} \end{bmatrix}$$

is an orthogonal matrix which reduces Z to RSF

$$S = U^TZU = \begin{bmatrix} -1 & 0 & 1 & -\frac{1}{2} \\ 0 & -1 & -1 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Then the unique positive definite solution of (15) is given by the solution of the linear matrix equation

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

or

$$\begin{bmatrix} x_{11} & x_{12} \\ x_{12} & x_{22} \end{bmatrix} \begin{bmatrix} \frac{1}{2} & -\frac{\sqrt{5}}{10} \\ -\frac{1}{2} & -\frac{\sqrt{5}}{10} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & -\frac{3\sqrt{5}}{10} \\ -\frac{1}{2} & -\frac{3\sqrt{5}}{10} \end{bmatrix}.$$

Thus, $X = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$ and it can quickly be checked that the spectrum of the "closed-loop matrix" $(A - BR^{-1}B^TX) = \begin{pmatrix} 0 & 1 \\ -1 & -2 \end{pmatrix}$ is $\{-1, -1\}$ as was evident from S_{11}.

Example 2: For checking purposes consider the solution of (15) with the following uncontrollable but stabilizable, and unobservable but detectable data:

$$A = \begin{pmatrix} 4 & 3 \\ -\frac{9}{2} & -\frac{7}{2} \end{pmatrix}, \quad B = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \quad R = 1, \quad Q = \begin{pmatrix} 9 & 6 \\ 6 & 4 \end{pmatrix}.$$

The solution of (15) is $X = \begin{pmatrix} 9c & 6c \\ 6c & 4c \end{pmatrix}$ where $c = 1 + \sqrt{2}$ and the closed-

loop spectrum is $\{-1/2, -\sqrt{2}\}$. These values were all obtained correctly to at least 14 significant figures as were the values for the corresponding discrete-time problem

$$A^TXA - X - A^TXB(R+B^TXB)^{-1}B^TXA + Q = 0 \qquad (16)$$

the solution of which is

$$X = \begin{pmatrix} 9d & 6d \\ 6d & 4d \end{pmatrix}$$

where $d = (1+\sqrt{5})/2$ and the closed-loop spectrum is $\{-1/2, (3-\sqrt{5})/2\}$.

Example 3: For further comparison purposes consider the discrete-time system of Example 6.15 in [36] where

$$A = \begin{pmatrix} 0.9512 & 0 \\ 0 & 0.9048 \end{pmatrix}, \quad B = \begin{pmatrix} 4.877 & 4.877 \\ -1.1895 & 3.569 \end{pmatrix}.$$

$$R = \begin{pmatrix} \frac{1}{3} & 0 \\ 0 & 3 \end{pmatrix}, \quad Q = \begin{pmatrix} 0.005 & 0 \\ 0 & 0.02 \end{pmatrix}.$$

The solution of (16) is given by

$$\begin{pmatrix} 1.36302 & 2.61722 & -0.705427 & 0.936860 & -0.293666 & 0.477354 & -0.197375 & 0.211212 & -0.166552 \\ & 7.59255 & -1.68036 & 1.47522 & -0.459506 & 0.665147 & -0.266142 & 0.280654 & -0.211212 \\ & & 1.77478 & 2.15771 & -0.609136 & 0.670717 & -0.262843 & 0.266142 & -0.197375 \\ & & & 8.25770 & -1.94650 & 1.75587 & -0.670717 & 0.665147 & -0.477354 \\ & & & & 1.80560 & 1.94650 & -0.609136 & 0.459506 & -0.293666 \\ & & & & & 8.25770 & -2.15771 & 1.47522 & -0.936860 \\ & & & & & & 1.77478 & 1.68036 & -0.705427 \\ & [\text{Symmetric}] & & & & & & 7.59255 & -2.61722 \\ & & & & & & & & 1.36302 \end{pmatrix}.$$

$$X = \begin{pmatrix} 0.010459082320970 & 0.003224644477419 \\ 0.003224644477419 & 0.050397741135643 \end{pmatrix}$$

and the feedback gain $\bar{F} = (R+B^TXB)^{-1}B^TXA$ is given by

$$\bar{F} = \begin{pmatrix} 0.071251660724426 & -0.070287376494153 \\ 0.013569839235296 & 0.045479287667006 \end{pmatrix}.$$

Note the typographical error in the (1,2)-element of $\bar{F}$ in [36]. The closed-loop eigenvalues are given by

0.508333461684191 and 0.688069670988913.

These are definitely different from [36] but have the same sum. Our numbers do appear to be the correct ones.

Example 4: We now consider somewhat higher order Riccati equations arising from position and velocity control for a string of high-speed vehicles. The matrices are taken from a paper by Athans, Levine, and Levis [37]. For a string of N vehicles it is necessary to solve the Riccati equation

$$A_N^TX_N + X_NA_N - X_NB_NR_N^{-1}B_N^TX_N + Q_N = 0$$

where all matrices are of order $n = 2N-1$ and are given by

$$A_N = \begin{pmatrix} A_{11} & A_{12} & & & & \\ & A_{22} & A_{23} & & & \\ & & \ddots & \ddots & & \\ & & & A_{N-2,N-2} & A_{N-2,N-1} & \\ & & & & A_{N-1,N-1} & 0 \\ & & & & 0 \; 0 & -1 \end{pmatrix}$$

where

$$A_{k,k} = \begin{pmatrix} -1 & 0 \\ 1 & 0 \end{pmatrix}, \quad A_{k,k+1} = \begin{pmatrix} 0 & 0 \\ -1 & 0 \end{pmatrix}$$

and

$$B_NR_N^{-1}B_N^T = \operatorname{diag}\{1,0,1,0,\cdots,0,1\}$$
$$Q_N = \operatorname{diag}\{0,10,0,10,\cdots,10,0\}.$$

For the case of five vehicles we repeated the calculations presented in [37]. The correct values for X rounded to six significant figures are

While 4 or 5 decimal places are published in [37], it can be seen that, surprisingly, only the first and sometimes the second were correct. Substitution of our full 16 decimal place solution into the Riccati equation gives a residual of norm on the order of 10^{-14} (consistent with a condition estimate of U_{11} of 26.3) while the residual for the solution in [37] has a large norm on the order of 10^{-1}. The closed-loop eigenvalues for the above problem (again rounded to six significant figures) are

$$\begin{array}{ll} -1.00000 & \\ -1.10779 \pm 0.852759 & j \\ -1.45215 \pm 1.26836 & j \\ -1.67581 \pm 1.51932 & j \\ -1.80486 \pm 1.66057 & j. \end{array}$$

We also computed the Riccati solution and closed-loop eigenvalues for the cases of 10 and 20 vehicles. This involved the solutions of 19th and 39th order Riccati equations, respectively, and rather than reproduce all the numbers here we give only the first five and last five elements of the first row (or column) of X and the fastest and slowest closed-loop modes. Again all values are rounded to just six significant figures; the complete numerical solutions are available from the author.

First row (column) of Riccati Solution		Fastest and Slowest Closed-Loop Modes	
$N=10$	$N=20$	$N=10$	$N=20$
$n=19$	$n=39$	$n=19$	$n=39$
1.40826	1.42021	-1.83667	-1.84459
2.66762	2.68008	$\pm 1.69509j$	$\pm 1.70368j$
-0.658219	-0.646127	$\vdots$	$\vdots$
1.04031	1.06539	-0.862954	-0.662288
-0.242133	-0.229761	$\pm 0.494661j$	
$\vdots$	$\vdots$		
-0.0515334	-0.0123718		
0.103453	0.0250824		
-0.0472086	-0.0120915		
0.0504036	0.0124632		
-0.0452352	-0.0119545		

The closed-loop eigenvalues for the case of, say, ten vehicles interlace and include, as a subset, those of five vehicles. Similarly, those for 20 vehicles interlace and include, as a subset, those of ten (and hence five) vehicles. It appears evident that both the elements of the Riccati solution and the closed-loop eigenvalues are converging to values in some finite region.

Example 5: This example involves circulant matrices. We wish to solve (15) with

$$A=\begin{bmatrix} -2 & 1 & 0 & \cdots & 0 & 1 \\ 1 & & & & & 0 \\ 0 & & & & & \cdot \\ \cdot & & & & & \cdot \\ \cdot & & & & & \cdot \\ \cdot & & & & & 0 \\ 0 & & & & & 1 \\ 1 & 0 & \cdots & & 0 & 1 & -2 \end{bmatrix}$$

and $BR^{-1}B^T=I$, $Q=I$. The matrices A, $BR^{-1}B^T$, Q are all circulant so the Riccati solution $X\in\mathbb{R}^{n\times n}$ is known to be circulant of the form

$$X=\begin{bmatrix} x_0 & x_{n-1} & x_{n-2} & \cdots & x_1 \\ x_1 & x_0 & x_{n-1} & \cdots & x_2 \\ x_2 & x_1 & x_0 & & \cdot \\ \cdot & \cdot & & \cdot & \cdot \\ \cdot & \cdot & & \cdot & \cdot \\ \cdot & \cdot & & \cdot & \cdot \\ x_{n-1} & x_{n-2} & \cdots & & x_0 \end{bmatrix}.$$

In fact, there is a simple transformation which "diagonalizes" the Riccati equation and allows the solution of (15) to be recovered via the solution of n scalar quadratic equations and an inverse discrete Fourier transform. The details of this procedure and related analysis of circulant systems can be found in the work of Wall [38]. For this example, we have $n=64$ and the x_i are given by

$$x_i=\frac{1}{64}\sum_{k=0}^{63}\left\{-2+2\cos\left(\frac{2\pi k}{64}\right)+\sqrt{5-4\cos\left(\frac{2\pi k}{64}\right)+4\cos^2\left(\frac{2\pi k}{64}\right)}\right\}\omega_{64}^{ik}$$

where ω_{64}^i is a 64th root of unity. The solution was computed by the Schur vector approach and checked by means of the circulant analysis of Wall. Our computed Riccati solution had at least 13 significant figures. For reference purposes we list

$$\begin{aligned} x_{11}&=0.37884325313566 \\ x_{12}&=0.18581947375535 \\ &\vdots \\ x_{44}&=0.37884325313567 \\ x_{45}&=0.18581947375536 \\ &\vdots \end{aligned}$$

The closed-loop eigenvalues are all real and are arranged as follows:

$$\left.\begin{aligned} &-4.1231056256177 \\ &-4.1137632861146 \\ &-4.1137632861146 \\ &\quad\vdots \end{aligned}\right\}\ \text{31 eigenvalues of multiplicity 2.}$$
$$-0.99999999999991$$

This 64th order example required approximately 50 s of CPU time on the 370/168 at M.I.T. and approximately 34 s on the 370/165 at the University of Toronto—both using Fortran H Extended (Opt.=2), double precision.

Example 6: This example is one which would be expected to cause problems on physical grounds and which appears to give rise to an "ill-conditioned Riccati equation." Consider the solution of (15) with

$$A=\begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ \cdot & & & & \cdot \\ \cdot & & & & \cdot \\ \cdot & & & & 0 \\ 0 & & & & 1 \\ 0 & & \cdots & & 0 \end{bmatrix},\quad B=\begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}$$

$$Q=\operatorname{diag}\{q,0,\cdots,0\},\quad R=r.$$

Here we have a system of n integrators connected in series. It is desired to apply a feedback controller to the nth system (which is to be integrated n times) so as to achieve overall asymptotic stability. Only deviations of x_1 (the nth integral of the constant system) from 0 are penalized. The reachability Gramian

$$W_t=\int_0^t e^{sA}BB^Te^{sA^T}ds,$$

while positive definite for all $t>0$, becomes more nearly singular as n increases. The system is "hard to control" in the sense of requiring a large amount of control energy (as measured by $\|W_t^{-1}\|$).

The closed-loop eigenvalues are easily seen to be the roots of

$$\lambda^{2n}+(-1)^n\frac{q}{r}=0$$

with negative real parts. These eigenvalues lie in a classic Butterworth pattern. It can also be easily verified that

$$x_{1n}=\sqrt{\frac{q}{r}}$$
$$=\text{product of the closed-loop eigenvalues.}$$

We attempted the solution of (15) with the above matrices and $q=r=1$. While the closed-loop eigenvalues were determined quite accurately as expected (approximately 14 decimal places using IBM double precision), the Riccati solution was increasingly less accurate as n increased due to the increasingly ill-conditioned nature of U_{11}. For example, for $n=21$ there was already a loss of ten digits of accuracy (consistent with a condition estimate of $0(10^{10})$ for U_{11}) in $x_{1n}(=1)$. Other computed elements of X were as large as $0(10^9)$ in magnitude.

Repeating the calculations with $q=10^4$, $r=1$ there was a loss of approximately 12 digits of accuracy in $x_{1n}(=100)$ for $n=21$. In this case other elements of X were as large as $0(10^{11})$ in magnitude. Again, the closed-loop eigenvalues were determined very accurately.

Our attempts to get Newton's method to converge on the above problem were unsuccessful.

Obviously, there is more that can be said analytically about this problem. Our interest here has been only to highlight some of the numerical difficulties.

VII. Concluding Remarks

We have discussed in considerable detail a new algorithm for solving algebraic Riccati equations. A number of numerical issues have been addressed and various examples given. The method is apparently quite numerically stable and performs reliably on systems with dense matrices of up to order 100 or so, storage being the main limiting factor.

For some reason, numerical analysts have never really studied algebraic Riccati equations. The algorithm presented here can undoubtedly be refined considerably from a numerical point of view but it nonetheless represents an immense improvement over algorithms heretofore proposed.

Some topics of continuing research in this area will include:

i) conditioning of Riccati equations,

ii) use of software to sort blocks of the RSF diagonal into just the two appropriate groups rather than within the two groups as well,

iii) making numerically viable the use of symplectic transformations such as in [17] to reduce the Hamiltonian or symplectic matrix Z to a convenient canonical form.

Each of these topics is of research interest in its own right in addition to the application to Riccati equations.

References

[1] W. M. Wonham, "On a matrix Riccati equation of stochastic control," *SIAM J. Contr.*, vol. 6, pp. 681–697, 1968.

[2] W. T. Reid, *Riccati Differential Equations,* New York: Academic, 1972.

[3] J. C. Willems, "Least squares stationary optimal control and the algebraic Riccati equation," *IEEE Trans. Automat. Contr.*, vol. AC-16, pp. 621–634, 1971.

[4] L. M. Silverman, "Discrete Riccati equations: Alternative algorithms, asymptotic properties, and system theory interpretations," in *Advances in Control Systems,* vol. 12, Leondes, Ed. New York: Academic, 1976, pp. 313–386.

[5] D. G. Lainiotis, "Partitioned Riccati solutions and integration-free doubling algorithms," *IEEE Trans. Automat. Contr.*, vol. AC-21, pp. 677–689, 1976.

[6] J. Rodriguez-Canabal, "The geometry of the Riccati equation," *Stochastics,* vol. 1, pp. 129–149, 1973.

[7] M. Pachter and T. E. Bullock, "Ordering and stability properties of the Riccati equation," Nat. Res. Inst. for Math. Sci., Rep. WISK 264, Pretoria, June 1977.

[8] G. Von Escherich, "Die Zweite Variation der Einfachen Integrale," *Wiener Sitzungsberichte,* vol. 8, pp. 1191–1250, 1898.

[9] A. G. J. MacFarlane, "An eigenvector solution of the optimal linear regulator problem," *J. Electron. Contr.*, vol. 14, pp. 643–654, 1963.

[10] J. E. Potter, "Matrix quadratic solutions," *SIAM J. Appl. Math.*, vol. 14, pp. 496–501, 1966.

[11] D. R. Vaughn, "A nonrecursive algebraic solution for the discrete Riccati equation," *IEEE Trans. Automat. Contr.*, vol. AC-15, pp. 597–599, 1970.

[12] A. F. Fath, "Computational aspects of the linear optimal regulator problem," *IEEE Trans. Automat. Contr.*, vol. AC-14, pp. 547–550, 1969.

[13] K. Mårtensson, "Approaches to the numerical solution of optimal control problems," Lund Inst. of Tech., Div. of Automat. Contr., Lund, Sweden, Rep. 7206, Mar. 1972.

[14] D. L. Kleinman, "On an iterative technique for Riccati equation computations," *IEEE Trans. Automat. Contr.*, vol. AC-13, pp. 114–115, 1968.

[15] F. A. Farrar and R. C. DiPietro, "Comparative evaluation of numerical methods for solving the algebraic matrix Riccati equation," United Technologies Res. Center, East Hartford, CT, Rep. R76-140268-1, Dec. 1976.

[16] G. A. Hewer and G. Nazaroff, "A survey of numerical methods for the solution of algebraic Riccati equations," Naval Weapons Center Rep., China Lake, CA.

[17] A. J. Laub and K. R. Meyer, "Canonical forms for Hamiltonian and symplectic matrices," *Celestial Mechan.*, vol. 9, pp. 213–238, 1974.

[18] A. J. Laub, "Canonical forms for σ-symplectic matrices," M.S. thesis, School of Math., Univ. of Minnesota, Dec. 1972.

[19] G. W. Stewart, *Introduction to Matrix Computations.* New York: Academic, 1973.

[20] F. D. Murnaghan and A. Wintner, "A canonical form for real matrices under orthogonal transformations," *Proc. Nat. Acad. Sci.*, vol. 17, pp. 417–420, 1931.

[21] G. H. Golub and J. H. Wilkinson, "Ill-conditioned eigensystems and the computation of the Jordan canonical form," *SIAM Rev.*, vol. 18, pp. 578–619, 1976.

[22] A. J. Laub, "A Schur method for solving algebraic Riccati equations," Lab. for Inform. and Decision Syst., Massachusetts Inst. of Tech., Cambridge, LIDS Rep. LIDS-R-859.

[23] P. Dorato and A. Levis, "Optimal linear regulators: The discrete-time case," *IEEE Trans. Automat. Contr.*, vol. AC-16, pp. 613–620, 1971.

[24] T. Pappas, A. J. Laub, and N. R. Sandell, "On the numerical solution of the discrete-time algebraic Riccati equation," Lab. for Inform. and Decision Systems, Massachusetts Inst. of Tech., Cambridge, LIDS Rep. LIDS-P-908.

[25] B. D. O. Anderson and J. B. Moore, *Optimal Filtering.* Englewood Cliffs, NJ: Prentice-Hall, 1979.

[26] G. W. Stewart, "HQR3 and EXCHNG: Fortran subroutines for calculating and ordering the eigenvalues of a real upper Hessenberg matrix," *ACM Trans. Math. Software,* vol. 2, pp. 275–280, 1976.

[27] B. T. Smith *et al., Matrix Eigensystems Routines—EISPACK Guide,* 2nd ed. *(Lecture Notes in Computer Science),* vol. 6. New York: Springer-Verlag, 1976.

[28] G. E. Forsythe, M. A. Malcolm, and C. B. Moler, *Computer Methods for Mathematical Computations.* Englewood Cliffs, NJ: Prentice-Hall, 1977.

[29] J. J. Dongarra, J. R. Bunch, C. B. Moler, and G. W. Stewart, *LINPACK User's Guide.* Philadelphia, PA: SIAM, 1979.

[30] B. N. Parlett and C. Reinsch, "Balancing a matrix for calculation of eigenvalues and eigenvectors," *Numer. Math.*, vol. 13, pp. 296–304, 1969.

[31] R. S. Bucy, "Structural stability for the Riccati equation," *SIAM J. Contr.*, vol. 13, pp. 749–753, 1975.

[32] J. H. Wilkinson, *The Algebraic Eigenvalue Problem.* London, England: Oxford Univ. Press, 1965.

[33] W. A. Coppel, "Matrix quadratic equations," *Bull. Austral. Math. Soc.*, vol. 10, pp. 377–401, 1974.

[34] H. -B. Meyer, "The matrix equation $AZ+B-ZCZ-ZD=0$," *SIAM J. Appl. Math.*, vol. 30, pp. 136–142, 1976.

[35] Y. Bar-Ness and G. Langholz, "The solution of the matrix equation $XC-BX=D$ as an eigenvalue problem," *Int. J. Syst. Sci.*, vol. 8, pp. 385–392, 1977.

[36] H. Kwakernaak and R. Sivan, *Linear Optimal Control Systems.* New York: Wiley, 1972.

[37] M. Athans, W. S. Levine, and A. Levis, "A system for the optimal and suboptimal position and velocity control for a string of high-speed vehicles," in *Proc. 5th Int. Analogue Computation Meetings,* Lausanne, Switzerland, Sept. 1967.

[38] J. E. Wall, "Control and estimation for large-scale systems having spatial symmetry," Ph.D. dissertation, Massachusetts Inst. of Tech., Electron. Syst. Lab. Rep. ESL-TH-842, Aug. 1978.

[39] R. H. Bartels and G. W. Stewart, "Solution of the matrix equation $AX+XB=C$," *Commun. ACM,* vol. 15, pp. 820–826, 1972.

Robust Solution of Perturbed Dynamical Equations from Within a Convex Restraint Set

B. ROSS BARMISH

***Abstract*—In [1] and [2], the notion of "*robust solution*" was defined for a system of perturbed linear dynamical equations. Heuristically speaking, a robust solution (input) $x(\cdot)$ is one for which the dependent variable (output) can be guaranteed to lie in a certain interval—independently of nature's choice of perturbation within given bounds. In this paper, we go beyond the results of [1] and [2] by developing criteria for robustness in the presence of a *solution restraint set* X, a compact-convex subset of R^m from which $x(t)$ must be chosen. Our new results are shown to degenerate into those of [1] and [2] as X becomes "large."**

I. Introduction and Formulation

The motivation for this paper comes from the problem of controlling a linear dynamical system whose description includes uncertain parameters. In particular, we shall be concerned with the case which arises when the range of possible parameter variations is sufficiently large to preclude the so-called sensitivity or series expansion approaches. Instead, we shall examine this problem from the *guaranteed performance* point of view, i.e., we seek an input (solution) vector $x(\cdot)$ which guarantees a certain *interval of solution* (range of outputs)—for *all* admissible parameter variations within given bounds. No *a priori* statistics are assumed.

In [1] and [2], a new notion of *robustness* was introduced which captures some of the salient features for the class of problems described above. The criteria for robustness given in both of these papers is *only valid when the input vector* $x(\cdot)$ is *unconstrained.* In this paper, we develop a more general theory of robustness which can handle the constrained input case. Our new results are shown to degenerate into those of [1] and [2] as the input restraint set becomes sufficiently large.

Our definition of robustness is comparable with the notion of target set reachability considered in the context of perturbed linear control systems (see [3]–[5], [13]), the major difference being as follows: the uncertain parameters here enter the dynamical equations multiplicatively rather than additively with respect to the input vector $x(\cdot)$, i.e., our uncertain parameters are to be thought of as model uncertainty rather than as additive noise. Loosely speaking, we can envision a dynamical system having an impulse response matrix $A(t,\tau)$ whose entries $a_{ij}(t,\tau)$ are uncertain, known only within given bounds.[1] As a consequence of the uncertainty in $A(t,\tau)$, the system output $y(t)$ is also uncertain. All we can assume *a priori* is that for a given input $x(\cdot)$, the output $y(t)$ is a member of the set

$$\left\{ \int_0^t A(t,\tau)x(\tau)\,d\tau : A(t,\tau) \text{ admissible} \right\}.$$

This is what is meant when we talk about *multiplicative uncertainty* with

Manuscript received November 10, 1978; revised June 20, 1979 and July 16, 1979. Paper recommended by A. Manitius, Chairman of the Optimal Systems Committee.

The author is with the Department of Electrical Engineering, University of Rochester, Rochester, NY 14627.

[1]Perhaps these uncertainties are a result of other *uncertainties* in the parameters of some underlying state equation.

0018-9286/79/1200-0921$00.75