Initializing Newton's method for discrete-time algebraic Riccati equations using the butterfly SZ algorithm

Heike Faßbender
Zentrum für Technomathematik
Fachbereich 3 - Mathematik und Informatik
Universität Bremen
D-28334 Bremen, Germany
E-mail: heike@math.uni-bremen.de

Peter Benner
Zentrum für Technomathematik
Fachbereich 3 - Mathematik und Informatik
Universität Bremen
D-28334 Bremen, Germany
E-mail: benner@math.uni-bremen.de

Abstract

The numerical solution of discrete-time algebraic Riccati equations is discussed. We propose to compute an approximate solution of the discrete-time algebraic Riccati equation by the (butterfly) SZ algorithm. This solution is then refined by a defect correction method based on Newton's method. The resulting method is very efficient and produces highly accurate results.

Keywords. discrete-time algebraic Riccati equation, Newton's method, SZ algorithm, symplectic matrix pencil, defect correction

1 Introduction

The standard (discrete-time) linear-quadratic optimization problem consists in finding a control trajectory $\{u(t), t \geq 0\}$, minimizing the cost functional

$$\mathcal{J}(x_0, u) = \sum_{t=0}^{\infty} [x(t)^T Q x(t) + u(t)^T R u(t)]$$

in terms of u subject to the dynamical constraint

$$x(t+1) = Ax(t) + Bu(t), \qquad x(0) := x_0.$$

Under certain conditions there is a unique control law,

$$u(t) = Hx(t), \qquad H := -(R + B^T X B)^{-1} B^T X A,$$

minimizing $\mathcal{J}$ in terms of u subject to the dynamical constraint. The matrix X is the unique symmetric stabilizing solution of the algebraic matrix Riccati equation

$$0 = \mathcal{DR}(X) = Q - X + A^T X A - A^T X B (R + B^T X B)^{-1} B^T X A. \quad (1)$$

The last equation is usually referred to as discrete-time algebraic Riccati equation (DARE). It appears not only in the context presented, but also in numerous procedures for analysis, synthesis, and design of control and estimation systems with H_2 or H_∞ performance criteria, as well as in other branches of applied mathematics.

The DARE (1) can be considered as a nonlinear set of equations. Therefore, Newton's method has been one of the first methods proposed to solve DAREs [8]. Given a symmetric matrix X_0, the method can be given in algorithmic form as follows:

```
FOR k = 0, 1, 2, ...
    A_k ← A − B(R + B^T X_k B)^{-1} B^T X_k A.
    Solve for N_k in the Stein equation
        A_k^T N_k A_k − N_k = −DR(X_k).
    X_{k+1} ← X_k + N_k
END FOR
```

It is well known that (under certain reasonable assumptions) if X_0 is a stabilizing starting guess, then all iterates are stabilizing and converge globally quadratic to the desired solution X (see, e.g., [8, 9, 11]). Despite the ultimate rapid convergence, the iteration may initially converge slowly. This can be due to a large initial error $||X - X_0||$ or a disastrously large first Newton step resulting in a large error $||X - X_1||$. In both cases, it is possible that many iterations are required to find the region of rapid convergence. An ill-conditioned Stein equation makes it difficult to compute an accurate Newton step. An inaccurately computed Newton step can cause the usual convergence theory to break down in practise. Sometimes rounding errors or a poor choice of X_0 cause Newton's method to converge to a non-stabilizing solution.

For these reasons, Newton's method is usually not used by itself to solve DAREs. Usually it is used as a defect correction method or for iterative refinement of an approximate solution obtained by a more robust method. A defect correction method can be based on the following result of Mehrmann and Tan [12, Theorem 2.7] (see also [11, Chapter 10]).

Theorem 1 *Let X be a symmetric solution of $\mathcal{DR}(X) = 0$. Let $\widehat{X}$ be a symmetric approximation to X and set $V := X - \widehat{X}$. If $\widehat{R} = R + B^T\widehat{X}B$ and $I + VB\widehat{R}^{-1}B^T$ are nonsingular, then V satisfies the DARE*

$$\widehat{A}^T V \widehat{A} - V - \widehat{A}^T VB(\widehat{R} + B^T VB)^{-1}B^T V\widehat{A} + \widehat{Q} = 0$$

where $\widehat{Q} = \mathcal{DR}(\widehat{X})$ and $\widehat{A} = (I - B\widehat{R}^{-1}B^T\widehat{X})A$.

Hence, the error V fulfills a discrete time algebraic Riccati equation just like the desired solution X. The defect discrete-time algebraic Riccati equation may be solved by any method for discrete-time algebraic Riccati equations. Most frequently, in situations like this, Newton's method is used [8, 11, 9, 2]. Then V can be used to correct $\widehat{X}$. Iterating this process until $\mathcal{DR}(X)$ is suitably small is called a defect correction method. Using this approach, it is often possible to squeeze out the maximum possible accuracy [11] after only a few iterations.

Here we propose to compute an approximate solution $\widehat{X}$ of $\mathcal{DR}(X)$ by the (butterfly) SZ algorithm. Combined with the defect correction method, the resulting method for solving (1) is a very efficient method and produces highly accurate results.

Now assume R to be positive definite and define

$$K - \lambda N = \begin{bmatrix} A & 0 \\ Q & I \end{bmatrix} - \lambda \begin{bmatrix} I & -BR^{-1}B^T \\ 0 & A^T \end{bmatrix}. \quad (2)$$

Using the standard control-theoretic assumptions that

- (A, B) is stabilizable,
- (Q, A) is detectable,
- Q is positive semidefinite,

then $K - \lambda N$ has no eigenvalues on the unit circle and there exists a unique stabilizing solution X of the DARE (1); see, e.g., [9]. It is then easily seen that $K-\lambda N$ has precisely n eigenvalues in the open unit disk and n outside. Moreover, the Riccati solution X can be given in terms of the deflating subspace of $K-\lambda N$ corresponding to the n eigenvalues $\lambda_1, \ldots, \lambda_n$ inside the unit circle using the relation

$$\begin{bmatrix} A & 0 \\ Q & I \end{bmatrix} \begin{bmatrix} I \\ -X \end{bmatrix} = \begin{bmatrix} I & -BR^{-1}B^T \\ 0 & A^T \end{bmatrix} \begin{bmatrix} I \\ -X \end{bmatrix} \Lambda$$

where $\Lambda \in \mathbb{R}^{n\times n}$, $\sigma(\Lambda) = \{\lambda_1, \ldots, \lambda_n\}$. Therefore, if we can compute $Y, Z \in \mathbb{R}^{n\times n}$ such that the columns of $\left[\begin{smallmatrix} Y \\ Z \end{smallmatrix}\right]$ span the desired deflating subspace of $K - \lambda N$, then $X = -ZY^{-1}$ is the desired solution of the Riccati equation (1). See, e.g., [9, 10, 11], and the references therein.

It is worthwhile to note that $K - \lambda N$ of the form (2) is a symplectic matrix pencil. A *symplectic matrix pencil* $K - \lambda N, K, N \in \mathbb{R}^{2n\times 2n}$, is defined by the property

$$KJK^T = NJN^T,$$

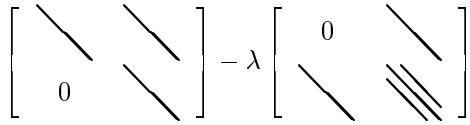
where

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

and I_n is the $n \times n$ identity matrix. The nonzero eigenvalues of a symplectic matrix pencil occur in reciprocal pairs: If λ is an eigenvalue of $K - \lambda N$ with left eigenvector x, then λ^{-1} is an eigenvalue of $K - \lambda N$ with right eigenvector $(Jx)^T$.

The numerical computation of a deflating subspace of a (symplectic) matrix pencil $K - \lambda N$ is usually carried out by an iterative procedure like the QZ algorithm. The QZ algorithm is numerically backward stable but it ignores the symplectic structure. Applying the QZ algorithm to a symplectic matrix pencil results in a general $2n \times 2n$ matrix pencil in generalized Schur form from which the eigenvalues and deflating subspaces can be read off. Due to roundoff errors unavoidable in finite-precision arithmetic, the computed eigenvalues will in general not come in pairs $\{\lambda, \lambda^{-1}\}$, although the exact eigenvalues have this property. Even worse, small perturbations may cause eigenvalues close to the unit circle to cross the unit circle such that the number of true and computed eigenvalues inside the open unit disk may differ. Moreover, the application of the QZ algorithm to $K-\lambda N$ is computationally quite expensive. The usual initial reduction to Hessenberg-triangular form requires about $70n^3$ flops plus $24n^3$ for accumulating the Z matrix; each iteration step requires about $88n^2$ flops for the transformations and $136n^2$ flops for accumulating Z; see, e.g., [13]. An estimated $40n^3$ flops are necessary for ordering the generalized Schur form. This results in a total cost of roughly $415n^3$ flops for solving a DARE using the QZ algorithm, using standard assumptions about convergence of the QZ iteration (see, e.g., [7]).

Here we propose to use the butterfly SZ algorithm for computing the deflating subspace of $K-\lambda N$. The butterfly SZ algorithm [4, 6] is a structure-preserving algorithm. It makes use of the fact that any symplectic matrix pencil can be reduced to a matrix pencil of the form



which is determined by just $4n-1$ parameters. By exploiting this special reduced form, the SZ algorithm is

fast and efficient; in each iteration step only $\mathcal{O}(n)$ arithmetic operations are required instead of $\mathcal{O}(n^2)$ arithmetic operations for a QZ step. We thus save a significant amount of work. Of course, the accumulation of the Z matrix is $\mathcal{O}(n^2)$ as in the QZ step. Moreover, by forcing the symplectic structure the above mentioned problems of the QZ algorithm are avoided. Combined with a defect correction method, the resulting method for solving discrete-time algebraic Riccati equations is a very efficient method and produces highly accurate results.

2 The butterfly SZ algorithm

For simplicity let us assume at the moment that A is nonsingular. Premultiplying $K - \lambda N$ by $\left[\begin{smallmatrix} I & 0 \\ 0 & A^{-T} \end{smallmatrix}\right]$ results in a symplectic matrix pencil

$$K' - \lambda N' = \begin{bmatrix} A & 0 \\ A^{-T}Q & A^{-T} \end{bmatrix} - \lambda \begin{bmatrix} I & -BR^{-1}B^T \\ 0 & I \end{bmatrix},$$

where K', N' are both symplectic. In [4, 6] it is shown that for every symplectic matrix pencil $L - \lambda M$ with $LJL^T = MJM^T = J$ there exist numerous symplectic matrices Z and nonsingular matrices S such that

$$S(L - \lambda M)Z = \begin{bmatrix} C & F \\ 0 & C^{-1} \end{bmatrix} - \lambda \begin{bmatrix} 0 & -I \\ I & T \end{bmatrix},$$

where C and F are diagonal matrices, and T is a symmetric tridiagonal matrix. Such a symplectic matrix pencil is called a *symplectic butterfly pencil*. If T is an unreduced tridiagonal matrix, then the butterfly pencil is called *unreduced*. If any of the $n-1$ subdiagonal elements of T are zero, the problem can be split into at least two problems of smaller dimension, but with the same symplectic butterfly structure.

Once the reduction to a symplectic butterfly pencil is achieved, the SZ algorithm is a suitable tool for computing the eigenvalues/deflating subspaces of the symplectic pencil. The SZ algorithm preserves the symplectic butterfly form in its iterations. It is the analogue of the SR algorithm for the generalized eigenproblem, just as the QZ algorithm is the analogue of the QR algorithm for the generalized eigenproblem. Both are instances of the GZ algorithm [14].

Each iteration step begins with L and M such that the butterfly pencil $L - \lambda M$ is unreduced. Choose a spectral transformation function q and compute a symplectic matrix Z_1 such that

$$Z_1^{-1} q(M^{-1}N) e_1 = \alpha e_1$$

for some scalar α. Then transform the pencil to

$$\widetilde{M} - \lambda \widetilde{N} = (M - \lambda N) Z_1.$$

This introduces a bulge into the matrices $\widetilde{M}$ and $\widetilde{N}$. Now transform the pencil to

$$\widehat{M} - \lambda \widehat{N} = S^{-1}(\widetilde{M} - \lambda \widetilde{N})\widetilde{Z},$$

where $\widehat{M}$ and $\widehat{N}$ are of symplectic butterfly form. S and $\widetilde{Z}$ are symplectic, and $\widetilde{Z}e_1 = e_1$. This concludes the iteration. Under certain assumption, it can be shown that the butterfly SZ algorithm converges cubically. For a detailed discussion of the butterfly SZ algorithm see [4, 6].

Given a symplectic pencil $L - \lambda M$, where $LJL^T = MJM^T = J$, first symplectic matrices Z_0 and S_0 are computed such that

$$\widehat{L} - \lambda \widehat{M} := S_0^{-1} L Z_0 - \lambda S_0^{-1} M Z_0$$

is a symplectic butterfly pencil. Using the butterfly SZ algorithm, symplectic matrices Z_1 and S_1 are computed such that

$$S_1^{-1}\widehat{L}Z_1 - \lambda S_1^{-1}\widehat{M}Z_1$$

is a symplectic butterfly pencil and the symmetric tridiagonal matrix $\widehat{T}$ in the lower right block of $S_1^{-1}\widehat{M}Z_1$ is reduced to quasi-diagonal form with 1×1 and 2×2 blocks on the diagonal. The eigenproblem decouples into a number of simple 2×2 or 4×4 generalized symplectic eigenproblems. Solving these subproblems, finally symplectic matrices Z_2, S_2 are computed such that

$$\begin{aligned} S_2^{-1}S_1^{-1}\widehat{L}Z_1Z_2 &= \begin{bmatrix} \phi_{11} & \phi_{12} \\ 0 & \phi_{22} \end{bmatrix}, \\ S_2^{-1}S_1^{-1}\widehat{M}Z_1Z_2 &= \begin{bmatrix} \psi_{11} & \psi_{12} \\ 0 & \psi_{22} \end{bmatrix}, \end{aligned}$$

where the eigenvalues of the matrix pencil $\phi_{11} - \lambda\psi_{11}$ are precisely the n stable generalized eigenvalues. Let $Z = Z_0Z_1Z_2$. Partitioning Z conformably,

$$Z = \begin{bmatrix} Z_{11} & Z_{12} \\ Z_{21} & Z_{22} \end{bmatrix}, \tag{3}$$

the Riccati solution X is found by solving a system of linear equations:

$$X = -Z_{21}Z_{11}^{-1}. \tag{4}$$

Instead of generating the symplectic matrix Z as in (3), one can work with $n \times n$ matrices X, Y and T such that finally $X = -Z_{21}Z_{11}^{-1}, Y = Z_{11}^{-1}Z_{12}$, and $T = Z_{11}^{-1}$. Starting from $X = Y = 0, T = I$, this can be implemented without accumulating the intermediate symplectic transformations used in the butterfly SZ algorithm, just using the parameters that determine these transformations. As for every symplectic matrix Z written in the form (3), $Z_{21}Z_{11}^{-1}$ is symmetric, this approach guarantees that all intermediate (and

the final) X are symmetric. Such an approach, called *symmetric updating* was first proposed by Byers and Mehrmann [5] in the context of solving continuous-time algebraic Riccati equations via the Hamiltonian SR algorithm and has also been proposed for solving DAREs with an SR algorithm in [1].

If the so computed approximate solution of the DARE is refined using Newton's method, usually the same number of iterations is required as when refining an approximation computed by the QZ algorithm. Even if one or two iterations more are necessary due to the loss of accuracy caused by using non-orthogonal transformations, this is well compensated by the cheaper SZ iteration.

3 A hybrid method for DAREs

Combined with a strategy to deflate zero and infinity eigenvalues from the symplectic pencil in order to deal with discrete-time algebraic Riccati equations with singular A matrix, the hybrid method consisting of the SZ algorithm followed by a few Newton iteration steps results in an efficient and accurate method.

Altogether, we propose the following algorithm to solve the DARE (1).

Algorithm 1
Input: *The coefficient matrices* $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $Q = Q^T \in \mathbb{R}^{n\times n}$, *and* $R \in \mathbb{R}^{m\times m}$.
Output: *An approximation* $\tilde{X} = \tilde{X}^T \in \mathbb{R}^{n\times n}$ *to the stabilizing solution of the DARE.*

1. Form the symplectic pencil $K - \lambda N$ as in (2).
2. Use Algorithm 15.16 of [11] to deflate all zero and infinite eigenvalues of $K - \lambda N$. That is, compute a nonsingular transformation matrix T_1 and a symplectic matrix S_1 such that

$$T_1(K - \lambda N)S_1 = \begin{bmatrix} 0 & \bar{A}_1 & 0 & 0 \\ 0 & \bar{A} & 0 & 0 \\ 0 & 0 & I_{n-k} & 0 \\ 0 & \bar{Q} & 0 & I_k \end{bmatrix} - \lambda \begin{bmatrix} I_{n-k} & 0 & -\bar{G}_{11} & -\bar{G}_{12} \\ 0 & I_{n-k} & -\bar{G}_{12}^T & -\bar{G}_{22} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -\bar{A}_1^T & -\bar{A}^T \end{bmatrix}$$

   and first $n-k$ columns of S_1 span the deflating subspace of $K - \lambda N$ corresponding to the zero eigenvalues.
3. Apply the butterfly SZ algorithm described Section 2 (and in detail in [4]) to the symplectic pencil

$$\tilde{K} - \lambda\tilde{N} := \begin{bmatrix} I_k & 0 \\ 0 & \tilde{A}^{-T} \end{bmatrix} \begin{bmatrix} \tilde{A} & 0 \\ \tilde{Q} & I_k \end{bmatrix} - \lambda \begin{bmatrix} I_k & -\tilde{G}_{22} \\ 0 & -\tilde{A}^T \end{bmatrix}$$

   such that

$$\tilde{T}_2(\tilde{K} - \lambda\tilde{N})\tilde{S}_2 = \begin{bmatrix} \phi_{11} & \phi_{12} \\ 0 & \phi_{22} \end{bmatrix} - \lambda \begin{bmatrix} \psi_{11} & \psi_{12} \\ 0 & \psi_{22} \end{bmatrix},$$

   where the eigenvalues of $\phi_{11} - \lambda\psi_{11}$ are the stable nonzero eigenvalues of $K - \lambda N$.
4. Partition $\tilde{S}_2 = \left[\begin{smallmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{smallmatrix}\right]$ where $S_{jj} \in \mathbb{R}^{k\times k}$, $j = 1, 2$. Set

$$Z := S_1 \begin{bmatrix} I_{n-k} & 0 & 0 & 0 \\ 0 & S_{11} & 0 & S_{12} \\ 0 & 0 & I_{n-k} & 0 \\ 0 & S_{21} & 0 & S_{22} \end{bmatrix}.$$

   Then the first n columns of Z span the stable deflating subspace of $K - \lambda N$ and an approximate solution $\widehat{X}$ of the DARE can be computed as in (4).
5. Use Newton's method endowed with a line search strategy as proposed in [2] and starting guess $X_0 = \widehat{X}$ in order to iteratively refine the solution of the DARE to the highest achievable accuracy.

Note that all left transformation matrices need not be accumulated. The accumulation of the right transformation matrices and the computation of $\widehat{X}$ via (4) can be avoided using the symmetric updating technique as mentioned at the end of Section 2.

More details of the algorithm and its implementation as well as a thorough numerical study regarding performance and accuracy will be reported in [3].

4 Concluding remarks

We have discussed the numerical solution of discrete-time Riccati equations. By initializing Newton's method with a starting guess computed by applying the butterfly SZ algorithm combined with a method for deflating zero and infinite eigenvalues to the corresponding symplectic matrix pencil, an efficient and accurate hybrid method is derived.

References

[1] G. Banse. *Symplektische Eigenwertverfahren zur Lösung zeitdiskreter optimaler Steuerungsprobleme.* Dissertation, Fachbereich 3 – Mathematik und Informatik, Universität Bremen, Bremen, FRG, June 1995. In German.

[2] P. Benner. Accelerating Newton's method for discrete-time algebraic Riccati equations. In *Proc. MTNS 98*, Padova, Italy, 1998. To appear.

[3] P. Benner and H. Faßbender. A hybrid method for the numerical solution of discrete-time algebraic Riccati equations. In preparation.

[4] P. Benner, H. Faßbender, and D.S. Watkins. *SR* and *SZ* algorithms for the symplectic (butterfly) eigenproblem. *Linear Algebra Appl.*, 287:41–76, 1999.

[5] R. Byers and V. Mehrmann. Symmetric updating of the solution of the algebraic Riccati equation. *Methods of Operations Research*, 54:117–125, 1985.

[6] H. Faßbender. *Symplectic Methods for Symplectic Eigenproblems.* Habilitationsschrift, Fachbereich 3 – Mathematik und Informatik, Universität Bremen, 28334 Bremen, (Germany), 1998.

[7] G.H. Golub and C.F. Van Loan. *Matrix Computations.* Johns Hopkins University Press, Baltimore, third edition, 1996.

[8] G.A. Hewer. An iterative technique for the computation of steady state gains for the discrete optimal regulator. *IEEE Trans. Automat. Control*, AC-16:382–384, 1971.

[9] P. Lancaster and L. Rodman. *The Algebraic Riccati Equation.* Oxford University Press, Oxford, 1995.

[10] A.J. Laub. Algebraic aspects of generalized eigenvalue problems for solving Riccati equations. In C.I. Byrnes and A. Lindquist, editors, *Computational and Combinatorial Methods in Systems Theory*, pages 213–227. Elsevier (North-Holland), 1986.

[11] V. Mehrmann. *The Autonomous Linear Quadratic Control Problem, Theory and Numerical Solution.* Number 163 in Lecture Notes in Control and Information Sciences. Springer-Verlag, Heidelberg, July 1991.

[12] V. Mehrmann and E. Tan. Defect correction methods for the solution of algebraic Riccati equations. *IEEE Trans. Automat. Control*, 33:695–698, 1988.

[13] V. Sima. *Algorithms for Linear-Quadratic Optimization*, volume 200 of *Pure and Applied Mathematics.* Marcel Dekker, Inc., New York, NY, 1996.

[14] D.S. Watkins and L. Elsner. Theory of decomposition and bulge chasing algorithms for the generalized eigenvalue problem. *SIAM J. Matrix Anal. Appl.*, 15:943–967, 1994.