SLICOT Working Note 1999-6

CTLEX – a Collection of Benchmark Examples for Continuous-Time Lyapunov Equations[1]

Daniel Kreßner[2], Volker Mehrmann[3], Thilo Penzl[4]

May 1999

[1]This paper describes Version 1.0 of the benchmark collection CTLEX. It presents research results of the European Community BRITE-EURAM III Thematic Networks Programme NICONET (contract number BRRT-CT97-5040) and is distributed by the Working Group on Software WGS. *WGS secretariat*: Mrs. Ida Tassens, ESAT - Katholieke Universiteit Leuven, K. Mercierlaan 94, 3001-Leuven-Heverlee, BELGIUM. This report is also available by anonymous ftp from *wgs.esat.kuleuven.ac.be/pub/WGS/REPORTS/SLWN1999-6.ps.Z*

[2]Fakultät für Mathematik, TU Chemnitz, D–09107 Chemnitz, Germany. Email: `kreda@mathematik.tu-chemnitz.de`

[3]Fakultät für Mathematik, TU Chemnitz, D–09107 Chemnitz, Germany. Email: `volker.mehrmann@mathematik.tu-chemnitz.de`

[4]Fakultät für Mathematik, TU Chemnitz, D–09107 Chemnitz, Germany. Email: `tpenzl@mathematik.tu-chemnitz.de`

Contents

1 Introduction

In the analysis of numerical methods and their implementation as numerical software it is extremely important to test the correctness of the implementation as well as the performance of the method. This validation is one of the major steps in the construction of a software library, in particular, if this library is used in practical applications.

In order to carry out such tests it is necessary to have tools that yield an evaluation of the performance of the method as well as the implementation with respect to correctness, accuracy, and speed. Similar tools are needed to compare different numerical methods, to test their robustness, and also to analyze the behaviour of the methods in extreme situations, where the limit of the possible accuracy is reached.

In many application areas benchmark collections have been created that can partially serve for this purpose. Such collections are heavily used. In order to have a fair evaluation and a comparison of methods and software, there should be a standardized set of examples, which are freely available and on which newly developed methods and their implementations can be tested. Moreover, public benchmark collections can be used by developers of algorithms and software as a reference when reporting the results of numerical experiments in publications.

In order to make such collections useful it is important that they cover a wide range of problems. Two kinds of test problems are of particular interest. First, benchmark collections should contain so-called 'real world' examples, i.e., examples reflecting current problems in applications. Second, they must contain test examples which drive numerical methods and their implementations to a limit. These are ideal test cases because errors and failures usually occur only in extreme cases and these are often not covered by standard software validation procedures.

Whereas plenty of test problems for basic linear algebra problems (e.g., systems of linear equations, eigenvalue problems) are collected in several collections, e.g., [8], benchmark examples for problems in control theory are harder to find. In this paper, we describe the benchmark collection CTLEX, which provides test examples of continuous-time *Lyapunov equations*. The standard form of these linear matrix equations is

$$A^T X + XA = Y.$$

However, we consider the more general form

$$A^T XE + E^T XA = Y, \tag{1}$$

to which we refer as *generalized* Lyapunov equation. Here, $A, E, Y \in \mathbb{R}^{n,n}$ are given matrices and $X \in \mathbb{R}^{n,n}$ is the sought solution matrix. The present version of CTLEX contains several examples of such equations or, to be more precise, several sets of corresponding matrices A, E, and Y. For some examples we provide the solution matrix X as well. This is the case, when the exact solution matrix is known or if it can be computed "directly" (see, e.g., Example 4.1 in §6). Solution matrices which have been computed numerically (e.g., by the standard methods [1, 6]) are not provided in CTLEX.

Many applications in control theory lead to Lyapunov equations with right hand side matrices of type $Y = -B^T B$, where $B \in \mathbb{R}^{m,n}$. Moreover, these equations are mostly *stable*. We call the generalized Lyapunov equation (1) stable if Y is symmetric and positive semidefinite, E is invertible, and $\sigma(AE^{-1}) \subset \mathbb{C}_-$, where $\sigma(AE^{-1})$ is the spectrum of AE^{-1} and $\mathbb{C}_-$ is the open left half of the complex plane. Note that the nonsingularity of both A and E is a necessary condition for the nonsingularity of the continuous-time Lyapunov equation (1). It is well-known that there exists a unique, symmetric, positive semidefinite solution to stable Lyapunov equations. Consequently, the solution matrix can be represented as a product of Cholesky factors $X = U^T U$. Several numerical methods (e.g., [6, 10]) compute the Cholesky factor U without determining the solution X itself and without forming the right hand side $Y = -B^T B$. For this reason, our benchmark collection is intended to provide not only the matrices E, A, Y, and X but also the factors B and U if these data are available for the respective example.

The examples of CTLEX are subdivided into four different *groups*:

- Group 1 — parameter-free problems of fixed size,
- Group 2 — parameter-dependent problems of fixed size,
- Group 3 — parameter-free examples of scalable size,
- Group 4 — parameter-dependent examples of scalable size.

The examples in Groups 2 and 4 depend on several parameters, which have a direct impact on the algebraic properties of the problem. Groups 3 and 4 contain examples which are freely scalable by a so-called scaling parameter. Of course, this parameter can have an (indirect) influence on the algebraic system properties, too. In some examples, the parameters can be restricted to certain ranges. These are indicated in the description. Moreover, default values are provided for each parameter.

The benchmark collection CTLEX has been implemented in FORTRAN and MATLAB. The FORTRAN codes conform to the implementation and documentation standards of the software library SLICOT [11]. See also [9] for general guidelines for the implementation of benchmark libraries in SLICOT.

We intend to augment the benchmark collection in the future. Contributions to forthcoming releases are highly appreciated.

2 Deriving further benchmark examples from other collections

Lyapunov equations arise from different problems in control theory. Perhaps the most important sources are balancing and balanced truncation model reduction of dynamical systems as well as the Newton method for algebraic Riccati equations. Dynamical systems and Riccati equations are covered by the benchmark collections CTDSX [7] and CAREX [2, 3], respectively. As a consequence, each of the benchmark examples in CTDSX and CAREX could be used to derive benchmark examples for Lyapunov equations. At first glance, adopting all the examples in CTDSX and CAREX to create a large number of Lyapunov benchmark examples seems to be the most straightforward way when implementing a benchmark collection such as CTLEX. However, we did not pursue this approach for several reasons. Avoiding the duplication of data and code is perhaps the most important of them. CTLEX is a benchmark collection which mainly focuses on “pure” Lyapunov examples. These are quite rare. However, a relatively large number of further benchmark examples can be obtained from the collections CTDSX and CAREX. This is very easy and straightforward with CTDSX but rather complicated with CAREX because here an implementation of the Newton method must be involved. Therefore, we describe only the first approach.

The major part of CTDSX consists of dynamical systems of type

$$\begin{aligned} \dot{x}(\tau) &= Ax(\tau) + Bu(\tau) \\ y(\tau) &= Cx(\tau). \end{aligned} \tag{2}$$

A pair of Lyapunov equations closely related to such systems are the following

$$AP + PA^T = -BB^T \tag{3}$$

$$A^TQ + QA = -C^TC. \tag{4}$$

The solution matrices P and Q are called controllability and observability Gramians, respectively. They play an important role in balancing and model reduction of systems (2). Since the matrices A, B, and C, which determine the Lyapunov equations (3) and (4), are explicitly given by CTDSX, it is possible to derive a pair of Lyapunov benchmarks from CTDSX benchmarks of type (2) with negligible implementational effort.

3 Parameter-free problems of fixed size (Group 1)

The current version of CTLEX contains no examples of this type. See §2 for how to obtain further benchmark examples.

4 Parameter-dependent problems of fixed size (Group 2)

The current version of CTLEX contains no examples of this type. See §2 for how to obtain further benchmark examples.

5 Parameter-free examples of scalable size (Group 3)

The current version of CTLEX contains no examples of this type. See §2 for how to obtain further benchmark examples.

6 Parameter-dependent examples of scalable size (Group 4)

Example 4.1 Standard stable Lyapunov equation. Matrices delivered: A, B, Y, X.

n	m	scaling parameter	default value	parameter	default value
n	1	$n \in \{2, 3, \ldots\}$	10	$r > 1$	1.5
				$s > 1$	1.5

To construct the matrices of this example we start with the auxiliary Lyapunov equation

$$A_0^T X_0 + X_0 A_0 = -B_0^T B_0. \tag{5}$$

Here, we choose

$$\begin{aligned} A_0 &= \operatorname{diag}(-1, -r, -r^2, \ldots, -r^{n-1}), \quad r > 1 \\ B_0 &= \begin{bmatrix} 1 & 2 & \ldots & n \end{bmatrix}. \end{aligned}$$

Loosely speaking, the parameter r regulates the spread of the eigenvalues of A_0. Because A_0 is diagonal, the entries of the solution can be computed easily as

$$(X_0)_{ij} = -\frac{(B_0)_i (B_0)_j}{(A_0)_{ii} + (A_0)_{jj}}$$

for $i, j = 1, \ldots, n$. Next we construct a transformation matrix $T \in \mathbb{R}^{n,n}$ as

$$T = H_2 S H_1,$$

where

$$\begin{aligned} H_1 &= I_n - \frac{2}{n} e e^T, \quad e = \begin{bmatrix} 1 & 1 & \ldots & 1 \end{bmatrix}^T & (6) \\ H_2 &= I_n - \frac{2}{n} f f^T, \quad f = \begin{bmatrix} 1 & -1 & \ldots & (-1)^{n-1} \end{bmatrix}^T & (7) \\ S &= \operatorname{diag}(1, s, \ldots, s^{n-1}), \quad s > 1. & (8) \end{aligned}$$

The parameter s determines the condition number of the transformation matrix. Note that

$$H_i = H_i^T = H_i^{-1} \tag{9}$$

for $i = 1, 2$. Transforming the matrices A_0, B_0, and X_0 as

$$\begin{aligned} A &= T A_0 T^{-1} \\ B &= B_0 T^{-1} \\ X &= T^{-T} X_0 T^{-1} \end{aligned}$$

we obtain the Lyapunov equation

$$A^T X + X A = -B^T B,$$

which is equivalent to (5). Taking (9) into account the desired matrices can be generated as

$$\begin{aligned} A &= H_2 S H_1 A_0 H_1 S^{-1} H_2 \\ B &= B_0 H_1 S^{-1} H_2 \\ X &= H_2 S^{-1} H_1 X_0 H_1 S^{-1} H_2 \\ Y &= -B^T B. \end{aligned}$$

Note that

$$\operatorname{cond}(A_0) = r^{n-1}$$

and

$$\mathrm{cond}(T) = \mathrm{cond}(S) = s^{n-1}.$$

As a consequence, increasing values of r and s tend to worsen the conditioning of the Lyapunov equation. Some caution is needed when choosing the parameters r and s. Even for moderate dimensions n values $r \gg 1$ and $s \gg 1$ will lead to numerically singular Lyapunov equations.

Example 4.2 Standard stable Lyapunov equation. Matrices delivered: A, B, Y.

n	m	scaling parameter	default value	parameter	default value
n	1	$n \in \{2, 3, \ldots\}$	10	$\lambda < 0$	$-.5$
				$s > 1$	1.5

This example is similar to Example 4.1 but here the matrices A_0 and B_0 are chosen as

$$\begin{aligned} A_0 &= \begin{bmatrix} \lambda & 1 & & & \\ & \ddots & \ddots & & \\ & & \ddots & \ddots & \\ & & & \ddots & 1 \\ & & & & \lambda \end{bmatrix} \\ B_0 &= \begin{bmatrix} 1 & 0 & \ldots & 0 \end{bmatrix}. \end{aligned}$$

The matrix A_0 consists of a Jordan block of size n. Note that the eigenvalue λ of A_0 is very sensitive with respect to perturbations. The matrices A, B, and Y are the result of the transformation

$$\begin{aligned} A &= H_2 S H_1 A_0 H_1 S^{-1} H_2 \\ B &= B_0 H_1 S^{-1} H_2 \\ Y &= -B^T B, \end{aligned}$$

where H_1, H_2, and S are given by (6), (7), and (8). Again, S is dependent on the scalar parameter s and $\mathrm{cond}(S) = s^{n-1}$. Increasing values of s tend to worsen the condition number of the Lyapunov equation.

Example 4.3 [4, 5, 10] Generalized Lyapunov equation. Matrices delivered: A, E, Y, X.

n	m	scaling parameter	default value	parameter	default value
n	-	$n \in \{2, 3, \ldots\}$	10	$t \geq 0$	10

This is an example of a generalized Lyapunov equation, where the exact solution matrix is provided. A and E are defined as

$$\begin{aligned} A &= (2^{-t} - 1) I_n + \mathrm{diag}(1, 2, \ldots, n) + U^T \\ E &= I_n + 2^{-t} U \end{aligned}$$

with

$$U = \begin{bmatrix} 0 & \cdots & \cdots & 0 \\ 1 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ 1 & \cdots & 1 & 0 \end{bmatrix}.$$

The solution is "chosen" as

$$X = ee^T, \quad e = \begin{bmatrix} 1 & 1 & \ldots & 1 \end{bmatrix}^T$$

and the corresponding right hand side matrix is computed as

$$Y = A^T XE + E^T XA.$$

Increasing values of t tend to lower the accuracy of numerically computed solution matrices.

Example 4.4 [10] Generalized stable Lyapunov equation. Matrices delivered: A, E, B, Y.

n	m	scaling parameter	default value	parameter	default value
$3q$	1	$q \in \{1, 2, \ldots\}$	10	$t \geq 1$	1.5

This is an example of a stable generalized Lyapunov equation, where the matrices A and E are defined as

$$\begin{aligned} A &= V \operatorname{diag}(A_1, A_2 \ldots, A_q) W \\ E &= VW \end{aligned} \tag{10}$$

with

$$\begin{aligned} A_i &= \begin{bmatrix} -t^i & 0 & 0 \\ 0 & -t^i & -t^i \\ 0 & t^i & -t^i \end{bmatrix} \\ V &= \begin{bmatrix} 0 & \cdots & 0 & 1 \\ \vdots & \cdot & \cdot & \vdots \\ 0 & \cdot & & \vdots \\ 1 & \cdots & \cdots & 1 \end{bmatrix} \\ W &= \begin{bmatrix} 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \vdots \\ 1 & \cdots & \cdots & 1 \end{bmatrix}. \end{aligned}$$

Note that this guarantees that the spectrum of AE^{-1} contains both real and complex eigenvalues. This is important because components with respect to real and complex eigenvalues are often treated in different branches in numerical software for Lyapunov equations.

The matrix B is chosen as

$$B = \begin{bmatrix} 1 & 2 & \dots & n \end{bmatrix}.$$

Increased values of t result in an increasing condition number of the equation. Some caution is needed when choosing the parameter t. Even for moderate dimensions n values $t \gg 1$ will lead to numerically singular Lyapunov equations.

Acknowledgments

The authors wish to thank N. Christov, P. Petkov, and M. Konstantinov, who provided Example 4.1.

References

[1] R.H. Bartels and G.W. Stewart. Solution of the matrix equation $AX + XB = C$: Algorithm 432. *Comm. ACM*, 15:820–826, 1972.

[2] P. Benner, A.J. Laub, and V. Mehrmann. A collection of benchmark examples for the numerical solution of algebraic Riccati equations I: Continuous-time case. Preprint SPC95-22, Technische Universität Chemnitz, Chemnitz, Germany, 1995.

[3] P. Benner, A.J. Laub, and V. Mehrmann. Benchmarks for the numerical solution of algebraic Riccati equations. *IEEE Control Systems Magazine*, 7:18–28, 1997.

[4] J.D. Gardiner, A.J. Laub, J.J. Amato, and C.B. Moler. Solution of the Sylvester matrix equation $AXB + CXD = E$. *ACM Trans. Math. Software*, 18:223–231, 1992.

[5] J.D. Gardiner, M.R. Wette, A.J. Laub, J.J. Amato, and C.B. Moler. Algorithm 705: A Fortran-77 software package for solving the Sylvester matrix equation $AXB^T + CXD^T = E$. *ACM Trans. Math. Software*, 18:232–238, 1992.

[6] S.J. Hammarling. Numerical solution of the stable, non–negative definite Lyapunov equation. *IMA J. Numer. Anal.*, 2:303–323, 1982.

[7] D. Kreßner, V. Mehrmann, and T. Penzl. CTDSX – a collection of benchmark examples for state-space realizations of continuous-time dynamical systems. SLICOT Working Note 1998-9.

[8] Matrix Market. URL: `http://math.nist.gov/MatrixMarket/`.

[9] V. Mehrmann and T. Penzl. Benchmark collections in SLICOT. SLICOT Working Note 1998-5.

[10] T. Penzl. Numerical solution of generalized Lyapunov equations. *Advances in Comp. Math.*, 8:33–48, 1998.

[11] SLICOT Implementation and Documentation Standards. URL: `http://www.win.tue.nl/wgs/stand.html`.

A The FORTRAN 77 subroutine BB03AD

The FORTRAN 77 subroutine BB03AD has the following calling sequence and in-line documentation.

```
      SUBROUTINE BB03AD(DEF, NR, DPAR, IPAR, VEC, N, M, E, LDE, A, LDA,
     1                  Y, LDY, B, LDB, X, LDX, U, LDU, NOTE, DWORK,
     2                  LDWORK, INFO)
C
C     PURPOSE
C
C     This routine is an implementation of the benchmark library
C     CTLEX (Version 1.0) described in [1].
C
C     It generates benchmark examples of (generalized) continuous-time
C     Lyapunov equations
C
C        T           T
C       A  X  E  +  E  X A  =  Y .
C
C     In some examples, the right hand side has the form
C
C                  T
C       Y  =  - B  B
C
C     and the solution can be represented as a product of Cholesky
C     factors
C
C                T
C       X  =  U  U .
C
C     E, A, Y, X, and U are real N-by-N matrices, and B is M-by-N. Note
C     that E can be the identity matrix. For some examples, B, X, or U
C     are no provided.
C
C     ARGUMENTS
C
C     Mode Parameters
C
C     DEF     CHARACTER*1
C             Specifies the kind of values used as parameters when
C             generating parameter-dependent and scalable examples
C             (i.e., examples with NR(1) = 2, 3, or 4):
C             DEF = 'D' or 'd': Default values are used.
C             DEF = 'N' or 'n': Values set in DPAR and IPAR are used.
```

```
C             This parameter is not referenced if NR(1) = 1.
C             Note that the scaling parameter of examples with
C             NR(1) = 3 or 4 is considered as a regular parameter in
C             this context.
C
C     Input/Output Parameters
C
C     NR      (input) INTEGER array, dimension 2
C             Specifies the index of the desired example according
C             to [1].
C             NR(1) defines the group:
C                   1 : parameter-free problems of fixed size
C                   2 : parameter-dependent problems of fixed size
C                   3 : parameter-free problems of scalable size
C                   4 : parameter-dependent problems of scalable size
C             NR(2) defines the number of the benchmark example
C             within a certain group according to [1].
C
C     DPAR    (input/output) DOUBLE PRECISION array, dimension 2
C             On entry, if DEF = 'N' or 'n' and the desired example
C             depends on real parameters, then the array DPAR must
C             contain the values for these parameters.
C             For an explanation of the parameters see [1].
C             For Example 4.1, DPAR(1) and DPAR(2) define 'r' and 's',
C             respectively.
C             For Example 4.2, DPAR(1) and DPAR(2) define 'lambda' and
C             's', respectively.
C             For Examples 4.3 and 4.4, DPAR(1) defines the parameter
C             't'.
C             On exit, if DEF = 'D' or 'd' and the desired example
C             depends on real parameters, then the array DPAR is
C             overwritten by the default values given in [1].
C
C     IPAR    (input/output) INTEGER array of DIMENSION at least 1
C             On entry, if DEF = 'N' or 'n' and the desired example
C             depends on integer parameters, then the array IPAR must
C             contain the values for these parameters.
C             For an explanation of the parameters see [1].
C             For Examples 4.1, 4.2, and 4.3, IPAR(1) defines 'n'.
C             For Example 4.4, IPAR(1) defines 'q'.
C             On exit, if DEF = 'D' or 'd' and the desired example
C             depends on integer parameters, then the array IPAR is
C             overwritten by the default values given in [1].
C
C     VEC     (output) LOGICAL array, dimension 8
```

```
C             Flag vector which displays the availability of the output
C             data:
C             VEC(1) and VEC(2) refer to N and M, respectively, and are
C             always .TRUE.
C             VEC(3) is .TRUE. iff E is NOT the identity matrix.
C             VEC(4) and VEC(5) refer to A and Y, respectively, and are
C             always .TRUE.
C             VEC(6) is .TRUE. iff B is provided.
C             VEC(7) is .TRUE. iff the solution matrix X is provided.
C             VEC(8) is .TRUE. iff the Cholesky factor U is provided.
C
C     N       (output) INTEGER
C             The actual state dimension, i.e., the order of the
C             matrices E and A.
C
C     M       (output) INTEGER
C             The number of rows in the matrix B. If B is not provided
C             for the desired example, M = 0 is returned.
C
C     E       (output) DOUBLE PRECISION array, dimension (LDE,N)
C             The leading N-by-N part of this array contains the
C             matrix E.
C             NOTE that this array is overwritten (by the identity
C             matrix), if VEC(3) = .FALSE.
C
C     LDE     INTEGER
C             The leading dimension of array E. LDE >= N
C
C     A       (output) DOUBLE PRECISION array, dimension (LDA,N)
C             The leading N-by-N part of this array contains the
C             matrix A.
C
C     LDA     INTEGER
C             The leading dimension of array A. LDA >= N
C
C     Y       (output) DOUBLE PRECISION array, dimension (LDY,N)
C             The leading N-by-N part of this array contains the
C             matrix Y.
C
C     LDY     INTEGER
C             The leading dimension of array Y. LDY >= N
C
C     B       (output) DOUBLE PRECISION array, dimension (LDB,N)
C             The leading M-by-N part of this array contains the
C             matrix B.
```

```
C
C     LDB     INTEGER
C             The leading dimension of array B. LDB >= M
C
C     X       (output) DOUBLE PRECISION array, dimension (LDX,N)
C             The leading N-by-N part of this array contains the
C             matrix X.
C
C     LDX     INTEGER
C             The leading dimension of array X. LDX >= N
C
C     U       (output) DOUBLE PRECISION array, dimension (LDU,N)
C             The leading N-by-N part of this array contains the
C             matrix U.
C
C     LDU     INTEGER
C             The leading dimension of array A. LDU >= N
C
C     NOTE    (output) CHARACTER*70
C             String containing short information about the chosen
C             example.
C
C     Workspace
C
C     DWORK   DOUBLE PRECISION array, dimension (LDWORK)
C
C     LDWORK  INTEGER
C             The length of the array DWORK.
C             For Examples 4.1 and 4.2., LDWORK >= 2*IPAR(1) is
C             required.
C             For the other examples, no workspace is needed, i.e.,
C             LDWORK >= 1.
C
C     Error Indicator
C
C     INFO    INTEGER
C             = 0:  successful exit;
C             < 0:  if INFO = -i, the i-th argument had an illegal
C                   value; in particular, INFO = -3 or -4 indicates
C                   that at least one of the parameters in DPAR or
C                   IPAR, respectively, has an illegal value.
C
C     REFERENCES
C
C     [1]  D. Kressner, V. Mehrmann, and T. Penzl.
```

```
C           CTLEX - a Collection of Benchmark Examples for Continuous-
C           Time Lyapunov Equations.
C           SLICOT Working Note 1999-6, 1999.
C
```

B The MATLAB function CTLEX

The MATLAB function `ctlex` has the following calling sequence and in-line documentation.

```
function [E,A,Y,B,X,U] = ctlex(nr,parin)
%CTLEX
%
% Usage:  [E,A,Y,B,X,U] = ctlex(nr,parin)
%         [E,A,Y,B,X,U] = ctlex(nr)
%
% Main routine of the benchmark library CTLEX (Version 1.0) described
% in [1]. It generates benchmark examples of (generalized) continuous-time
% Lyapunov equations
%
%        T           T
%       A  X  E  +  E  X A  =  Y .                               (1)
%
% In some examples, the right hand side has the form
%
%                T
%       Y  =  - B  B
%
% and the solution can be represented as a product of Cholesky factors
%
%              T
%       X  =  U  U .
%
% E, A, Y, X, and U are real n-by-n matrices, and B is m-by-n. Note
% that E can be the identity matrix. For some examples, B, X, or U are
% not provided.
%
% Input:
%  - nr    : index of the desired example according to [1];
%            nr is a 1-by-2 matrix;
%            nr(1) defines the group:
%             = 1 : parameter-free problems of fixed size
%             = 2 : parameter-dependent problems of fixed size
%             = 3 : parameter-free problems of scalable size
%             = 4 : parameter-dependent problems of scalable size
%            nr(2) defines the number of the benchmark example within
%            a certain group.
%  - parin : parameters of the chosen example;
%            referring to [1], the entries in parin have the following
%            meaning:
%            Ex. 4.1 : parin(1:3) = [n r s]
```

```
%             Ex. 4.2 : parin(1:3) = [n lambda s]
%             Ex. 4.3 : parin(1:2) = [n t]
%             Ex. 4.4 : parin(1:2) = [q t]
%             parin is optional; default values as defined in [1] are
%             used as example parameters if parin is ommited. Note that
%             parin is not referenced if nr(1) = 1.
%
% Output:
%  - E, A, Y, B, X, U :  matrices of the Lyapunov equation (1).
%
% References:
%
% [1]   D. Kressner, V. Mehrmann, and T. Penzl.
%       CTLEX - a Collection of Benchmark Examples for Continuous-Time
%       Lyapunov Equations.
%       SLICOT Working Note 1999-6, 1999.
```