APPENDIX **B**

MATCONTROL AND LISTING OF MATCONTROL FILES

B.1 ABOUT MATCONTROL

What is the MATCONTROL library?
The MATCONTROL library is a set of M-files implementing the majority of algorithms of the book:

Numerical Methods for Linear Control Systems Design and Analysis by B.N. Datta.

Who wrote the MATCONTROL library?
The MATCONTROL library was written by several graduate students of Professor Datta. The most contributions were made by Joao Carvalho and Daniil Sarkissian.

How can I get the MATCONTROL library?
The MATCONTROL library is distributed with the book mentioned above.

What to do if a routine is suspected to give wrong answers?
Please let us know immediately. Send an email to dattab@math.niu.edu and, if possible, include a MATLAB diary file that calls the routine and produces the wrong answer.

How to install MATCONTROL
The MATCONTROL library is distributed in a subdirectory called "Matcontrol." This directory must be copied from the media that accompanies the book into anywhere in your system.

After the "Matcontrol" directory has been copied, you just have to let MATLAB know where MATCONTROL is located. In order to do that, you must include it in MATLAB's path.

The easiest way to do so is by including the proper MATLAB commands in your MATLAB startup file (startup.m). If you do not have this file already, please create it.

Using your preferred text editor, open (or create) startup.m and add the following line:

Unix/Linux systems:
matlabpath([matlabpath,'path_of_Matcontrol']);
MS-Windows* systems:
path(path,'path_of_Matcontrol');

Examples: Here, "Mfiles" is the working directory of MATLAB.
On Linux PC:
matlabpath([matlabpath,':/home/carvalho/Mfiles/Matcontrol']);
On Unix-Solaris Workstation:
matlabpath([matlabpath,':/export/home/grad/carvalho/Mfiles/Matcontrol']);
On MS-Windows PC:
path(path,'C:\Mfiles\Matcontrol');

Once you have done that, you can use MATCONTROL in the next MATLAB session. Please issue the command **"help Matcontrol"** to see if MATCONTROL was properly included in MATLAB's path. You should see a list of all MATCONTROL M-files.

*Disclaimer: MATLAB and Windows are trademarks of their respective owners.

B.2 CHAPTERWISE LISTING OF MATCONTROL FILES

Here is the chapterwise listing of MATCONTROL files.

Reference: ***Numerical Methods for Linear Control Systems Design and Analysis* by B.N. Datta.**

Chapter 5: Linear State-Space Models and Solutions of the State Equations

EXPMPADE	The Padé approximation to the exponential of a matrix
EXPMSCHR	Computing the exponential of a matrix using Schur decomposition
FREQRESH	Computing the frequency response matrix using Hessenberg decomposition
INTMEXP	Computing an integral involving matrix exponentials

Chapter 6: Controllability, Observability, and Distance to Uncontrollability

CNTRLHS	Finding the controller-Hessenberg form
CNTRLHST	Finding the controller-Hessenberg form with triangular sub-diagonal blocks.
OBSERHS	Finding the observer-Hessenberg form

CNTRLC	Finding the controller canonical form (Lower Companion)
DISCNTRL	Distance to controllability using the Wicks–DeCarlo algorithm

Chapter 7: Stability, Inertia, and Robust Stability

INERTIA	Determining the inertia and stability of a matrix without solving a matrix equation or computing eigenvalues
H2NRMCG	Finding H_2-norm using the controllability Grammians
H2NRMOG	Finding H_2-norm using the observability Grammian
DISSTABC	Determining the distance to the continuous-time stability
DISSTABD	Determining the distance to the discrete-time stability

Chapter 8: Numerical Solutions and Conditioning of Lyapunov and Sylvester Equations

CONDSYLVC	Finding the condition number of the Sylvester equation problem
LYAPCHLC	Finding the Cholesky factor of the positive definite solution of the continuous-time Lyapunov equation
LYAPCHLD	Finding the Cholesky factor of the positive definite solution of the discrete-time Lyapunov equation
LYAPCSD	Solving the discrete-time Lyapunov equation using complex Schur decomposition of A
LYAPFNS	Solving the continuous-time Lyapunov equation via finite series method
LYAPHESS	Solving the continuous-time Lyapunov equation via Hessenberg decomposition
LYAPRSC	Solving the continuous-time Lyapunov equation via real Schur decomposition
LYAPRSD	Solving the discrete-time Lyapunov equation via real Schur decomposition
SEPEST	Estimating the *sep* function with triangular matrices
SEPKR	Computing the *sep* function using Kronecker product
SYLVHCSC	Solving the Sylvester equation using Hessenberg and complex Schur decompositions
SYLVHCSD	Solving the discrete-time Sylvester equation using Hessenberg and complex Schur decompositions
SYLVHESS	Solving the Sylvester equation via Hessenberg decomposition

SYLVHRSC	Solving the Sylvester equation using Hessenberg and real Schur decompositions
SYLVHUTC	Solving an upper triangular Sylvester equation

Chapter 9: Realization and Subspace Identification

MINRESVD	Finding minimal realization using singular value decomposition of the Hankel matrix of Markov parameters
MINREMSVD	Finding minimal realization using singular value decomposition of a Hankel matrix of lower order

Chapter 10: Feedback Stabilization, Eigenvalue Assignment, and Optimal Control

STABLYAPC	Feedback stabilization of continuous-time system using Lyapunov equation
STABLYAPD	Feedback stabilization of discrete-time system using Lyupunov equation
STABRADC	Finding the complex stability radius using the bisection method
HINFNRM	Computing H_∞-norm using the bisection method

Chapter 11: Numerical Methods and Conditioning of the Eigenvalue Assignment Problems

POLERCS	Single-input pole placement using the recursive algorithm
POLEQRS	Single-input pole placement using the QR version of the recursive algorithm
POLERQS	Single-input pole placement using RQ version of the recursive algorithm
POLERCM	Multi-input pole placement using the recursive algorithm
POLERCX	Multi-input pole placement using the modified recursive algorithm that avoids complex arithmetic and complex feedback
POLEQRM	Multi-input pole placement using the explicit QR algorithm
POLESCH	Multi-input pole placement using the Schur decomposition
POLEROB	Robust pole placement

Chapter 12:	**State Estimation: Observer and the Kalman Filter**
SYLVOBSC	Solving the constrained multi-output Sylvester-observer equation
SYLVOBSM	Solving the multi-output Sylvester-observer equation
SYLVOBSMB	Block triangular algorithm for the multi-output Sylvester-observer equation
Chapter 13:	**Numerical Solutions and Conditioning of the Algebraic Riccati Equations**
RICEIGC	The eigenvector method for the continuous-time Riccati equation
RICSCHC	The Schur method for the continuous-time Riccati equation
RICSCHD	The Schur method for the discrete-time Riccati equation
RICGEIGD	The generalized eigenvector method for the discrete-time Riccati equation
RICNWTNC	Newton's method for the continuous-time Riccati equation
RICNWTND	Newton's method for the discrete-time Riccati equatioin
RICSGNC	The matrix sign-function method for the continuous-time Riccati equation
RICSGND	The matrix sign-function method for the discrete-time Riccati equation
RICNWLSC	Newton's method with line search for the continuous-time Riccati equation
RICNWLSD	Newton's method with line search for the discrete-time Riccati equation
Chapter 14:	**Internal Balancing and Model Reduction**
BALSVD	Internal balancing using the singular value decomposition
BALSQT	Internal balancing using the square-root algorithm
MODREDS	Model reduction using the Schur method
HNAPRX	Hankel-norm approximation