Appendix G
The CAPTAIN Toolbox for Matlab: an Overview

The **C***omputer-***A***ided* **P***rogram for* **T***ime Series* **A***nalysis and* **I***dentification of* **N***oisy Systems* (CAPTAIN) Toolbox provides access to novel algorithms for various important aspects of identification, estimation, nonstationary time series analysis and signal processing, adaptive forecasting and automatic control system design. These have been developed between 1981 and the present at Lancaster University, UK, based on research carried out by the author at the University of Cambridge, UK (1965-1975) and the Australian National University, Canberra, Australia (1975-1981). Although it has its origins in the CAPTAIN (Young and Shellswell, 1973; Young and Jakeman, 1979a) and micro-CAPTAIN (Young and Benner, 1991) packages, the CAPTAIN Toolbox for Matlab is much more flexible and provides access to many more algorithms that have been added over the 1990s and continue to be developed up to the present day, most of them only available in this Toolbox. The Toolbox Website is at `http://captaintoolbox.co.uk/` and the fully functional CAPTAIN Toolbox can be downloaded for a three month long trial from a linked website at `http://www.es.lancs.ac.uk/cres/captain/`. Further usage requires payment of a licence fee, which entitles the licensee to a very useful e-Handbook (Pedregal et al., 2005) and the provision of all future additions and enhancements. The Handbook can be considered as a companion volme to the present book: it contains a description of each routine available in the Toolbox, as well as numerous examples that include verbatim snapshots of command line operations and explanations of how the various program parameters are specified. The modest, non-profit license fee, available to legitimate students, is required to cover the overhead costs required for continuing development of the Toolbox.

Toolbox Overview

Essentially, the CAPTAIN toolbox is a collection of routines (Matlab m-file scripts) for the identification and estimation of the various model types discussed in the main body of the present book; and their subsequent use with various application routines available in the Toolbox. These are organized around the core of the *Recursive Least Squares* (RLS), *Kalman Filter* (KF), *Fixed Interval Smoothing* (FIS) and *Refined Instrumental Variable* (RIV) recursive algorithms. However, in order to allow for straightforward user access, CAPTAIN consists of numerous 'shells' i.e. top level functions that automatically generate the necessary model structures, with default options based on the experience of the developers and the extensive practical experience with the algorithms (but user-adjustable to override the default settings). In this regard, the main areas of functionality are listed below.

Unobserved Component (UC) Models

- The *Dynamic Harmonic Regression* (DHR) model in the form of a harmonic regression with time variable parameters (Young et al, 1999) . This is estimated using the hyper-parameter optimization function dhropt followed by the DHR algorithm dhr primed with these optimized hyper-parameters. It is a particularly useful algorithm for signal extraction, forecasting and back-casting of periodic or quasi-periodic series. The same function allows for the estimation of other, related models, such the *Basic Structural Model* (BSM: Harvey et al, 1989).
- Other related TVP models are: *Dynamic Linear Regression* (DLR), *Dynamic Auto-Regression* (DAR), *Dynamic AR with eXogenous inputs* (DARX), and *Dynamic Transfer Function* (DTF). These are accessible via the dlr, dar, darx and dtfm routines; and the associated dlropt, daropt, darxopt optimization routines. dtfmopt.

As this book shows, these UC routines provide a general resource for nonstationary time series analysis, forecasting and signal extraction, as well as time-frequency analysis; and they have been applied to a wide range of engineering, environmental, biological and socio-economic systems, as discussed in numerous earlier publications.

Transfer Function Model Identification and Estimation

- The functional pairs rivid/riv (for discrete-time SISO and MISO TF models) and rivcid/rivc (for continuous-time SISO and MISO TF models estimated from discrete-time sampled data) are provided for order/structure identification (rivid, rivcid) and parameter estimation (riv, rivc) in the case of constant parameter, linear TF models. These routines include, as options, both recursive and *en-bloc* versions of the optimal RIV and SRIV algorithms, in addition to conventional least squares-based approaches. Recent enhancements are the introduction of the rivbj/rivcbj and rivbjid/rivcbjid routines for full BJ models with ARMA additive noise. These also produce the estimation results in an 'object' coded form that is compatible with routines in the Matlab SID Toolbox.
- In all of the order/structure identification routines (the above routines ending in 'id'), the model listing, for the selection of models chosen, can be reported in the order of various statistics, such as the coefficient of determination R_T^2 and various identification statistics, including the AIC, BIC and our own YIC (see main text).

- The ivarmaid and ivarma routines that implement the IVARMA algorithm for the identification and estimation of ARMA noise models. The ivarma routine is part of the rivbj/rivcbj routines but is made available separately for univariate noise model estimation.
- New rivbjdd and rivcbjdd for estimating parameters in multi-input models with different denominators are being developed and it is hoped that they will be available to all users during the next year.
- New clrivbj and clrivcbj routines for closed loop system identification and estimation (see chapter 9) are being developed and it is hoped that these, as well as rriv and rrivc routines for real-time recursive TVP estimation (see chapter 10), will be available sometime during the next two years (only my colleague Dr. Wlodek Tych and I are involved in these 'CAPTAINization' tasks and preparing foolproof routines for CAPTAIN is a time-consumptive operation).

Nonlinear State Dependent Parameter Model Identification

- State-dependent parameter (SDP) nonlinear modelling is based around the sdp routine, which yields non-parametric (graphical) estimates of state-dependent parameters. If required, the user can parameterize these graphically defined nonlinearities using specified nonlinear functions (*e.g.* exponential, power law, radial basis functions etc.) that can then be optimized using standard Matlab functions (see chapter 11 in the main text).

Proportional-Integral-Plus (PIP) Control System Design Routines

- The PIP control system design routines include the pip algorithm for pole assignment design and the pipopt for PIP-LQ design, together with all other required support routines. These are not discussed in this book but details can be found in Taylor et al. (2000) and the prior references therein. Also a book, *True Digital Control* on this topic is currently in preparation. These very powerful, multivariable PIP control system design routines are based on TF models identified and estimated using the above routines in CAPTAIN, which are then converted to the *Non-Minimum State Space* (NMSS) form that is the basis for the design of the PIP non-minimal state variable feedback control laws. They are also integrated into Simulink objects that can be used in simulation studies and, potentially, for on-line use.

Other Routines

- Various conventional models, identification tools and auxiliary functions, too numerous to list individually here. Of these, the largest is the kalmanfis routine, which provides a shell to the KF/FIS algorithms for general state space filtering, smoothing and forecasting purposes. System identification is inherent to the modelling approach utilized by most of the functions already discussed. Other routines include: acf to determine the sample and partial autocorrelation function; ccf for the sample cross-correlation; period to estimate the periodogram;

and statist for some sample descriptive statistics. Additional statistical diagnostics include: boxcox (optimal Box-Cox transformation for homoscedasticity); cusum (cusum recursive test for time varying mean and cusumsq recursive test for time varying variance); and histon (histogram over normal distribution and Bera-Jarque statistical test); while useful general routines are: del for generating a matrix of delayed variables; irwsm for smoothing, decimation or for fitting a simple trend to a time series; prepz to prepare data for TF modelling (e.g. baseline removal and input scaling); scaleb to rescale the estimated TF model numerator polynomial following initial prepz use; stand to standardise or de-standardise a matrix by columns; and reconst to reconstruct a time series by removing any dramatic jumps in the trend.

Finally, note that almost all of the recursive estimation and smoothing procedures outlined above will automatically handle missing data in the time series, represented in Matlab by special Not-a-Number (NaN) variables. Indeed, by appending or prepending such variables to the data set using the fcast function, the UC routines will forecast, interpolate or back-cast as appropriate, without requiring further user intervention. Finally, the toolbox is supported by on-line help and demonstration examples (see below).

Demonstration examples

- The demonstration examples are invoked by typing the instruction captdemo in the Matlab command line, which displays a GUI Menu with various 'button' options. In addition to a Captain Overview, these include: the DHR show, DAR show and RIV show options for free running demonstration examples in a special graphics window, as well as other 'command line' options for TVP command line demos; TF and PIP command line demos; Upgraded (BJ) TF demos and Handbook demos. These various command line demos will be the most useful for readers of this book and some their options are referred to in various of the exercises at the end of the book chapters. Each of the command line demo buttons accesses another GUI sub-menu that has self-explanatory options.