Contents

Chapter 1

User guide to SMI 2.0

1.1 List of functions in SMI 2.0

The following table gives an overview of the functions in the SMI toolbox. Some function are new to this toolbox. Others appeared in the previous version of the toolbox under an other name. These names have been changed to prevent conflicts with other toolboxes and to provide a more consistent naming. This makes the functions easier to recognize and remember.

function	description	old name
Discrete time moesp		
dordom	Ordinary moesp preprocessor	dordom
dordpi	Ppast input moesp preprocessor	dordpi
dordpo	Past output moesp preprocessor	dordpo
dordeiv	Eiv moesp preprocessor	-
dordrs	Reconstructed state moesp preprocessor	dordrs
destac	Estimate A,C	dmodpi,dmodpo
destbd	Estimate B,D	dac2bd,destb
destk	Estimate Kalman gain	dmodpo
destx	Estimate initial state	dinit
dmoesp	Frontend for DT moesp	-
Continuous time moesp		
cordom	Ordinary moesp preprocessor	-
cordpi	Past input moesp preprocessor	-
cordpo	Past output moesp preprocessor	-
cestac	Estimate A,C	-
cestbd	Estimate B,D	-
cestx	Estimate initial state	-
Recursive moesp		
drpi	Recursive PI moesp	-
drpo	Recursive PO moesp	-
SLS optimization		
dss2th	Parameterization of state space system	ss2thon
dth2ss	Reconstruction of state space system	th2sson
dslslin	Optimize DT linear model using SLS	gnlisls
dslswie	Optimize DT wiener model using SLS	gnwisls
dfunlin	Cost-function for dslslin	-
dfunwie	Cost-function for dslswie	-
drslslin	Recursive optimization of DT model using SLS	-
clslin	Optimize CT linear model using SLS	-
cfunlin	Cost-function for dslslin	-
crslslin	Recursive optimization of CT model using SLS	-
Non causal models		
ncdlsim	Simulate non causal model	-
ncdestac	Estimate A and C for non-causal model	-
ncdestbd	Estimate B and D for non-causal model	-
kroneckf	Calculate Kronecker canonical form	-
Nonlinear models		
chebest	Estimate MIMO nonlinear model	tchebest
chebsim	Simulate MIMO nonlinear model	tchebest
Miscellaneous		
prbn	Pseudo random binary sequence	prbn
vaf	Variance accounted for	vaf
shave	Remove peaks and outliers	shave

1.2 Function descriptions

cestac

Purpose

Estimates the matrices A and C of a LTI state space model using the result of the preprocessor routines `cordxx` (`cordom`, `cordpo`, etc.). General model structure:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) + w(t) \\ y(t) &= Cx(t) + Du(t) + v(t) \end{aligned}$$

For more information about the disturbance properties see the help pages for the preprocessor `cordxx` functions.

Syntax

```
[A,C]=cestac(R,n);
```

Inputs

`R`	Data structure obtained from `cordxx`, containing the triangular factor and additional information (such as i/o dimension etc.).
`n`	Order of system to be estimated.

Outputs

`A,C`	Estimated system matrices.

See also

`cordom, cordpi, cordpo, cestbd`

cestbd

Purpose

Estimates the matrices B and D of the state space model

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) + w(t) \\ y(t) &= Cx(t) + Du(t) + v(t) \\ x(0) &= x_0 \end{aligned}$$

using the knowledge of the pair A, C. This function can concatenate different input-output data batches, through the matrix R and Rold). B and D and x_0 are calculated by solving a linear least squares problem.

Syntax

```
[B,D,x0,R]=cestbd(u,y,A,C,[fB fD fx],Rold);
[B,D]=cestbd(u,y,A,C);
```

Inputs

`u,y`	The input and output data of the system to be identified.
`A,C`	The estimated system matrices A and C.
`model`	Three element flag vector $\begin{bmatrix} fB & fD & fx \end{bmatrix}$ indicating whether B, D and x_0 should be estimated. The default value is $\begin{bmatrix} 1 & 1 & 0 \end{bmatrix}$. The matrix B or D can be assumed zero by setting fB or fD to zero. The calculation of x_0 can be omitted by setting fx to zero. However x_0 will not be assumed zero then. It's influence will still be taken into account for the computation of B and D.
`Rold`	R matrix obtained from previous data batch. This variable can be used to process data in batches, or to combine data from different experiments.

Outputs

`B,D`	The estimated system matrices B and D.
`x0`	The estimated initial state of the system.
`R`	Compressed data matrix, storing information on the calculation of the matrices B and D in following `cestbd`. Used when analyzing multiple input-output data sequences.

See also

`cordxx, cestac, cestx`

cestx

Purpose

Estimate the initial state, given the estimated system matrices and a set of input/output data.

Syntax

```
x0=cestx(u,y,Ts,A,B,C,D);
```

Inputs

`u,y`	The input and output data of the system to be identified.
`Ts`	Sampling period of the measured data.
`A,B,C,D`	System matrices.

Outputs

`x0`	Estimated initial state.

See also

`cestbd`

chebest

Purpose

This function estimates a MIMO static nonlinear function between the signals y and z. The function is estimated on the basis of Chebychev polynomials. Before estimating the coefficients of the polynomials the input signal is shifted and scaled to fall within the region [-1,1]. The shifting and scaling factors are included in the parameter vector.

Syntax

```
[thl,ze,Phi]=chebest(y,z,nn);
```

Inputs

`y,z`	Input and output of the nonlinearity.
`nn`	Order of the Chebychev polynomials in the nonlinear function.

Outputs

`thl`	Vector with the parameters of the static nonlinearity.
`ze`	Estimated output, on basis of the model that is obtained.
`Phi`	matrix with the Chebychev functions of y, such that ze = Phi × thl.

See also

`chebsim, dslswie`

chebsim

Purpose

Simulates a static nonlinear function on the basis of Chebychev polynomials with input y. The coefficients of the Chebychev polynomials are given with the vector thl, and commonly estimated with either `dslswie` or `chebest`.

Syntax

```
[ze,Phi]=chebsim(y,thl)
```

Inputs

`y`	The input to the nonlinearity.
`thl`	Parameter matrix, with coefficients of the nonlinear function.

Outputs

`ze`	Estimated output.
`Phi`	matrix with the Chebychev functions of y, such that ze = Phi × thl.

See also

`chebest, dslswie`