ANDECS
Model Order Reduction Modules

A. Varga

DLR
Deutsche Forschungsanstalt für Luft- und Raumfahrt e.V.
Institut für Robotik & Systemdynamik
Entwurfsorientierte Regelungstechnik (Prof. G. Grübel)
D–82234 Oberpfaffenhofen
## CONTENTS

**LSRED - Linear Systems Reduction 3**

**Modules for Reduction of Stable Linear Systems**
- **BTA**: Balanced truncation approximation of stable linear systems 9
- **SPA**: Singular perturbation approximation of stable linear systems 14
- **HNA**: Hankel-norm approximation of stable linear systems 18
- **BST**: Balanced stochastic truncation of stable linear systems 22
- **SPF**: Singular perturbation formulas for linear systems 27
- **BIL**: Bilinear transformation of linear systems 30

**Modules for Reduction of Unstable Linear Systems**
- **ASD**: Additive spectral decomposition of linear systems 33
- **LCF**: Left coprime factorization of transfer-function matrices 38
- **RCF**: Right coprime factorization of transfer-function matrices 42
- **LCFID**: Left coprime factorization with inner denominator 46
- **RCFID**: Right coprime factorization with inner denominator 50
- **LCF2LS**: Left coprime factorization to linear system transformation 54
- **RCF2LS**: Right coprime factorization to linear system transformation 57

**Module for Frequency Weighted Model Reduction**
- **PFWMR**: Projections for frequency-weighted model reduction 60

**Modules for Norms Evaluations**
- **HANKNORM**: Hankel-norm of a transfer function matrix 64
- **L2NORM**: L2-norm of a transfer function matrix 68
- **LINFNORM**: L-infinity norm of a transfer function matrix 72
LSRED - Linear Systems REDuction

Chapter purpose:
This chapter describes modules for order reduction of linear time-invariant systems. Several basic modules can be used directly to perform model reduction of stable systems and can serve occasionally as minimal realization procedures too. Other modules, used in conjunction with the basic modules, represent tools which aid in performing more complex model reductions as for instance the reduction of unstable systems or the frequency-weighted model reduction.

I. Reduction of Stable Systems

The basic model reduction modules (BTA, SPA, HNA, BST) are based on numerically reliable accuracy enhancing algorithms for reduction of stable systems and possess the following special features:
1. They employ exclusively square-root information (Cholesky factors instead of gramians) to compute the reduced order models and therefore have better accuracy than methods working with full gramians.
2. They compute the reduced models by using projection formulas and thus are applicable regardless the original system is minimal or not. Excepting the module HNA, the computation of projections in all other modules can avoid systems balancing. These latter methods are called square-root balancing-free methods and have better accuracy for highly unbalanced systems than the pure square-root (balancing based) methods.
3. They can be used to reduce both continuous-time and discrete-time systems.
4. They have an H-infinity approximation error bound. Either the infinity norm of the absolute error or the relative error of the reduced order model is bounded by a precomputable positive real number for all frequency.

Additive Model Reduction Methods

If \( G \) and \( G_r \) are the transfer-function matrices of the original and reduced order systems, respectively, then the additive methods try to compute \( G_r \) such that the norm of the additive modelling error

\[
\Delta_n = G - G_r
\]

is minimized. Three basic methods are available to do additive error model reduction:
- Balance & Truncate Approximation (B & T) /1/;
- Singular Perturbation Approximation (SPA) /2/;
- Optimal Hankel-Norm Approximation (HNA) /3/.

Each of the above methods possess the same infinity-norm error bounds for an \( r \)-th order reduced order model \( G_r \) of an \( n \)-th order system \( G \):
\[ \sigma_{r+1} \leq \| G - G_r \|_\infty \leq 2 \sum_{i=r+1}^{n} \frac{\sigma_i}{1 - \sigma_i}, \]

where \( \sigma_i \), \( i = 1, 2, \ldots, n \) are the Hankel-singular values of the system. The Hankel-singular values are the positive square-roots of the eigenvalues of the product of systems gramians.

The following modules are available for additive model reduction:

- **BTA** computes a reduced order model by using the square-root or square-root balancing-free version of the B & T method /4,5/. This module can also be used for computing balanced minimal state-space realizations of continuous-time or discrete-time systems.

- **SPA** computes a reduced order model by using the square-root or square-root balancing-free version of the SPA method /6/.

- **HNA** computes a reduced order model by using the optimal HNA method of /3/ based on the square-root balancing method.

Relative Model Reduction Method

If \( G \) and \( G_r \) are the transfer-function matrices of the original and of the reduced order systems, respectively, then the relative methods try to compute \( G_r \) such that the norm of the relative modelling error \( \Delta_r \), defined by the expression

\[ G_r = G(I - \Delta_r) \]

is minimized. The implemented method is the Balanced Stochastic Truncation (BST) method /7/. This method possesses the following bound for the relative error for an \( r \)-th order reduced order model \( G_r \) of an \( n \)-th order system \( G \):

\[ \sigma_{r+1} \leq \| \Delta_r \|_\infty \leq 2 \sum_{i=r+1}^{n} \frac{\sigma_i}{1 - \sigma_i}, \]

where \( \sigma_i \), \( i = 1, 2, \ldots, n \) are the Hankel-singular values of an all-pass phase matrix.

The following module is available for relative model reduction:

- **BST** computes a reduced order model by using the square-root or square-root balancing-free version of the BST method /8,9/. This module can also be used for computing stochastically balanced minimal state-space realizations of continuous-time or discrete-time systems.
In the BST module, a parameter $\alpha$ can be used as a weight between the absolute and relative errors. For $\alpha > 0$, the BST method is performed on a modified system with the transfer-function matrix $[G \, \alpha I]$. A zero value of $\alpha$ means a pure relative error minimization. Large positive values of $\alpha$ produce approximations which minimize the absolute approximation error. When $\alpha$ tends to infinite, the BST method produces identical results with the B & T method.

Auxiliary tools

Two modules are provided to be used in conjunction with the model reduction subroutines for stable systems:

SPF computes a reduced order model by using state residualization (singular perturbation) formulas. The reduced model has the same steady-state gain as the original system. This module can be also used to evaluate the steady-state gain of a stable system.

BIL performs a multivariable two-parameters bilinear transform. This module can be used for continuous-to-discrete or discrete-to-continuous mapping of linear systems.

II. Order Reduction of Unstable Systems

Two alternative basic approaches can be used for reducing unstable models:

Reduction of Stable Projections

If $G$ is the transfer-function matrix of a $n$-th order (not necessarily stable) system, then the following procedure can be used to reduce the order of $G$:

1) By using the ASD module decompose additively $G$ as

$$G = G_1 + G_2$$

such that $G_1$ has only stable poles and $G_2$ has only unstable poles.

2) Determine $G_{1r}$, a reduced order approximation of the stable part $G_1$, by using any of modules BTA, SPA, HNA or BST.

3) By using the PARALLEL module, assemble the reduced model $G_r$ as

$$G_r = G_{1r} + G_2$$
Reduction of Stable Coprime Factors /10,11/

The following procedure can be used to compute an r-th order approximation \( G_r \) of an n-th order (not necessarily stable) system \( G \):

1) By using one of modules LCF or LCFID, compute a left coprime factorization of the transfer-function matrix \( G \) in the form

\[
G = R^{-1}Q,
\]

where \( R \) and \( Q \) are stable transfer-function matrices of degree \( n \).

2) Approximate the transfer-function matrix \([[Q, R]]\) of degree \( n \) with \([[Q_r, R_r]]\) of degree \( r \) by using a model reduction method for stable systems, by using by using any of modules BTA, SPA, HNA or EST.

3) By using the module LCF2LS, form the r-th order approximation of \( G \) as

\[
G_r = R_r^{-1}Q_r.
\]

A similar procedure can be given for a right coprime factorization of \( G \) in the form

\[
G = QR^{-1},
\]

by using the modules RCF or RCFID, and RCF2LS.

III. Frequency-Weighted Order Reduction

If \( G \) is a given \( p \times m \) stable transfer-function matrix of degree \( n \), and \( W_1 \) and \( W_2 \) are \( p \times p \) and \( m \times m \) stable, invertible and minimum-phase transfer-function matrices of degrees \( n_1 \) and \( n_2 \), respectively, then the following procedure can be used to compute an r-th order approximation \( G_r \) of \( G \) by employing the frequency-weighting approach proposed in /12/:

1) By using the module PFWMR, compute \( G_1 \), the n-th order stable projection of \((W_1^*)^{-1}G(W_2^*)^{-1} \).

2) Determine \( G_{r1} \), an r-th degree approximation of \( G_1 \) by using any of modules BTA, SPA, HNA or BST.

3) By using the module PFWMR, compute \( G_r \) as the r-th order stable projection of \( W_1^*G_{r1}W_2^* \).

In the above expressions \( W^* \) represents either \( W^T(-s) \) for a continuous-time system or \( W^T(1/z) \) for a discrete-time system. The stable projections at step 2 and 3 are computed by using the
explicit formulas derived in /13/.

If the optimal Hankel-norm approximation method (HNA module) is used at step 2 of the above procedure the optimal weighted approximation error satisfies

\[ \| (W_1^*)^{-1}(G-G_r)(W_2^*)^{-1}\|_H = \sigma_{r+1}, \]

where \( \sigma_{r+1} \) is the \((r+1)\)-th Hankel singular values of \( G_1 \), the transfer-function matrix computed at step 1 of the above procedure, and \( \|G\|_H \) is the Hankel-norm of \( G \).

Literature:

/1/ Moore B.C.
Principal Component Analysis in Linear Systems: Controllability, Observability, and Model Reduction.

/2/ Liu Y. and Anderson B.D.O.
Singular Perturbation Approximation of Balanced Systems,

/3/ Glover, K.
All Optimal Hankel Norm Approximation of Linear Multivariable Systems and Their L-Infinity Error Bounds,

/4/ Tombs M.S. and Postlethwaite I.
Truncated Balanced Realization of Stable, Non-minimal State-Space Systems.

/5/ Varga A.
Efficient Minimal Realization Procedure Based on Balancing.

/6/ Varga A.

/7/ Desai U.B. and Pal D.
A Transformation Approach to Stochastic Model Reduction.

/8/ Safonov M.G. and Chiang R.Y.
Model Reduction for Robust Control: a Schur Relative-Error Method.

/9/ Varga A. and Fasol K.H.
A New Square-Root Balancing-Free Stochastic Truncation Model Reduction Algorithm.

/10/ Varga A.
Coprime Factors Model Reduction Based on Accuracy Enhancing Techniques,

/11/ Varga A.

/12/ Latham, G.A., Anderson, B.D.O.

/13/ Varga A.

Chapter structure:

**Modules for Reduction of Stable Linear Systems**
- **RTA**: Balanced truncation approximation of stable linear systems
- **SPA**: Singular perturbation approximation of stable linear systems
- **ENA**: Hankel-norm approximation of stable linear systems
- **BST**: Balanced stochastic truncation of stable linear systems
- **SPF**: Singular perturbation formulas for linear systems
- **BIL**: Bilinear transformation of linear systems

**Modules for Reduction of Unstable Linear Systems**
- **ASD**: Additive spectral decomposition of linear systems
- **LCF**: Left coprime factorization of transfer-function matrices
- **RCF**: Right coprime factorization of transfer-function matrices
- **LCFID**: Left coprime factorization with inner denominator
- **RCFID**: Right coprime factorization with inner denominator
- **LCF2LS**: Left coprime factorization to linear system transformation
- **RCF2LS**: Right coprime factorization to linear system transformation

**Module for Frequency Weighted Model Reduction**
- **PFWMR**: Projections for frequency-weighted model reduction

**Modules for Norms Evaluations**
- **HANKNORM**: Hankel-norm of a transfer function matrix
- **L2NORM**: L2-norm of a transfer function matrix
- **LINFNORM**: L-infinity norm of a transfer-function matrix
MONITORMODULE

BTA - Balanced truncation approximation of stable linear systems

Monitormodule purpose:
Given a linear state-space model \((A,B,C,D)\) of order \(n\),
the balanced truncation approximation (BTA) method is used
to compute a reduced order model \((A_r,B_r,C_r,D)\) of order \(r\)
by using either:
(1) the square-root BTA method /1/, or
(2) the square-root balancing-free BTA method /2/.
The module can be also used for computing balanced minimal order
state-space realizations.

The Hankel singular values and the error bound for the maximum
approximation error can be optionally displayed.

Optional outputs on database are:
- reduced order system  \(\text{(def. output)}\)
- Hankel singular values  \(\text{(def. no output)}\)
- Error bound for the maximum approximation error  \(\text{(def. no output)}\)

The Hankel singular values are the square-roots of eigenvalues of the
product of controllability and observability gramians. The largest
singular value is the Hankel-norm of the system transfer-function
matrix.

If \(G\) and \(G_r\) are the transfer-function matrices of the systems
\((A,B,C,D)\) and \((A_r,B_r,C_r,D)\), respectively, then the approximation error
\(G-G_r\) satisfies the inequalities:

\[
\text{HSV}(r+1) \leq \text{INFNORM}(G-G_r) \leq 2^{1+} \left( \text{HSV}(r+1) + \ldots + \text{HSV}(r) \right),
\]

where HSV(i), \(i = 1, \ldots, n\) are the Hankel-singular values of the \(n\)-th
order system \((A,B,C,D)\) and \text{INFNORM}(G) is the infinity-norm of G.
The right-hand side above is the optionally computed error bound for
the maximum approximation error.

Options:
Next case

Database Structure:
input:
linear system, state space
output:
<Result S-Name>
 _LS  linear system, state space
 _MAXERR  matrix for error bound
 _HSV  matrix for Hankel singular values,

Dialog:
------ requested input
LS RS-NAME:
-> linear system name
    or carriage return to end the module
INPUT SIGNALS - U,V,ALLIN ?
-> select inputs, default is U
OUTPUT SIGNALS - YM,YA,ALLOUT ?
-> select outputs, default is YM
------ end

1. Level commands:
-----------------
BTA >C>
PARAM : set parameter values for model reduction
PLIST : set parameter values for listing of results
POUT : set parameter values for output on database
COMPUTE : perform model reduction
    If one of outputting parameters is set then:
        ------ requested input
        <Result S-NAME> ( <Default S-NAME> ):
            -> If a blank is input, the <Default S-Name> is used.
            Otherwise the user must input a new S-Name under
            which the result will be stored according to the
            parameters selected through POUT.
            The <Default S-name> is the S-name of the input
            linear system appended by _BTA.index, where index
            is the next free value. If one does not use the
            <Default S-Name>, there could already exist other
            objects under the chosen < Result S-Name>. Then
            the action taken depends on a status variable:
            /SDEVERIF is set: It is asked whether all objects
            can be or not deleted. If objects
            should be not deleted, then a new
            <Result S-Name> is asked for.
            /SDENOVER is set: All objects under the given
            structure are deleted.
        If the resulting reduced system is stored:
            ------ requested input
            Text for dataobject LS:
            ------ end
            ------ end
END : terminate module

2. Level Commands:
-------------------
BTA_PARAM >P>
BALANCE : 0/1 (INTEGER def: 0)
        0 : use the square-root balancing-free BTA method
        1 : use the square-root BTA method
ORDER : desired order of reduced model (INTEGER def: 0)
        0 : order chosen in accordance with the value of TOLHSV
k : desired order of the reduced model (k ≥ 0)
TOLHSV : tolerance on Hankel singular values (DOUBLE def: 0.0)
If ORDER = 0 and TOLHSV > 0, then the order of the computed reduced system is equal to the number of Hankel-singular values less than or equal to TOLHSV.
When TOLHSV .LE. 0, an internally computed default value TOLHSV = n*EPS*HNORM(A,B,C) is used, where n is the order of the original system, EPS is the machine precision and HNORM(A,B,C) is the Hankel-norm of the original model (largest Hankel singular value)

BTA_PLIST >>P>
HSV : 0/1 (INTEGER def: 0)
  0 : do not list the Hankel singular values
  1 : list the Hankel singular values
MAXERR : 0/1 (INTEGER def: 0)
  0 : do not list the maximum approximation error
  1 : list the maximum approximation error

BTA_POUT >>P>
REDSYS : 0/1 (INTEGER def: 1)
  0 : do not store the resulting reduced order system
  1 : store the resulting reduced order system
HSV : 0/1 (INTEGER def: 0)
  0 : do not store the Hankel singular values
  1 : store the Hankel singular values
MAXERR : 0/1 (INTEGER def: 0)
  0 : do not store the maximum approximation error
  1 : store the maximum approximation error

Monitor module initialization:
module name : BTA
module group : LSRED
version number : 1.0
FORTRAN name : PEBTA

File input/output:
none

Method:

Literature
/1/ Tombs M.S. and Postlethwaite I.
   Truncated balanced realization of stable, non-minimal
   state-space systems, Int. J. Control, Vol. 46,

/2/ Varga A.
   Efficient minimal realization procedure based on balancing
   Proc. of IMACS/IFAC Symp. MCTS, Lille, France, May 7-10, 1991,

Remarks:
- The reduced system is stable and minimal if HSV(r) > HSV(r+1), where r is the order of the reduced system.
- The reduced system is not balanced if BALANCE is set to 0.
- A balanced minimal realization can be computed by setting BALANCE = 1, ORDER = 0 and TOLHSV = 0.

Life cycle:
1993 Aug A. Varga DLR FF-DF: coded

Packages required:
none

Libraries required:
RASP, LAPACK, BLAS (1,2,3)

Example:
Given the continuous-time system (A,B,C) with the following matrices:

\[
A = \begin{pmatrix}
-0.04165 & 0 & 4.92 & -4.92 & 0 & 0 & 0 \\
-5.21 & -12.5 & 0 & 0 & 0 & 0 & 0 \\
0 & 3.33 & -3.33 & 0 & 0 & 0 & 0
\end{pmatrix}
\]

\[
B = \begin{pmatrix}
0 & 0 \\
12.5 & 0 \\
0 & 0
\end{pmatrix}
\]

\[
C = \begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}
\]

the following balanced state-space realization can be computed by setting: BALANCE = 1, ORDER = 0, TOLHSV = 0:

\[
A_r = \begin{pmatrix}
-0.3967 & 3.0279 & 0 & 0 & 0.9557 & 0.4547 & 0 \\
-3.0871 & -2.524 & 0 & 0 & -2.601 & -3.651 & 0 \\
0 & 0 & -0.5849 & -2.2166 & 0 & 0 & -0.6258
\end{pmatrix}
\]

\[
B_r = \begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}
\]

\[
C_r = \begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}
\]
\[
\begin{pmatrix}
-0.9987 & 0.9987 \\
-0.7254 & 0.7254 \\
-1.0591 & -1.0591 \\
\end{pmatrix}
\]
\[
\begin{pmatrix}
1.2046 & 1.2046 \\
-0.9716 & 0.9716 \\
0.5840 & -0.5840 \\
-0.5738 & -0.5738 \\
\end{pmatrix}
\]
\[
\begin{pmatrix}
-0.9922 & 0.6335 & -1.0591 & -1.2046 & 0.8978 & 0.5838 & -0.5738 \\
0.1612 & 0.4997 & 0 & 0 & -0.5253 & 0.0224 & 0 \\
-0.9922 & -0.6335 & -1.0591 & -1.2046 & -0.8978 & -0.5838 & -0.5738 \\
\end{pmatrix}
\]

The computed Hankel singular values are:

\[
2.5139 & 2.0846 & 1.9178 & 0.7666 & 0.5473 & 0.0253 & 0.0246
\]

The following reduced order system is obtained by setting BALANCE = 0, ORDER = 0, TOLHSV = 0.1:

\[
\begin{pmatrix}
1.3451 & -5.0399 & 0 & 0 & -4.5315 \\
4.0214 & -3.6604 & 0 & 0 & -0.9056 \\
0 & 0 & 0.5124 & -1.7910 & 0 \\
0 & 0 & 4.2167 & -2.9900 & 0 \\
-1.2402 & 1.6416 & 0 & 0 & -0.0586 \\
\end{pmatrix}
\]

\[
\begin{pmatrix}
-0.3857 & 0.3857 \\
3.1753 & -3.1753 \\
-3.6872 & -3.6872 \\
-1.8197 & 1.8197 \\
\end{pmatrix}
\]

\[
\begin{pmatrix}
0.1089 & -0.4867 & 0 & 0 & -0.8651 \\
0.6704 & 0.1828 & 0.6582 & 0.2222 & -0.0104 \\
\end{pmatrix}
\]
MONITORMODULE

SPA - Singular perturbation approximation of stable linear systems

Monitor module purpose:

Given a linear state-space model \((A, B, C, D)\) of order \(n\), the square-root balancing-free singular perturbation approximation (SPA) method is used to compute a reduced order model \((A_r, B_r, C_r, D_r)\) of order \(r \leq 1\). The module can be also used to compute reduced order balanced state-space realizations.

The Hankel singular values and the error bound for the maximum approximation error can be optionally displayed.

Optional outputs on database are:
- reduced order system (def. output)
- Hankel singular values (def. no output)
- Error bound for the maximum approximation error (def. no output)

The Hankel singular values are the square-roots of the eigenvalues of the product of controllability and observability gramians. The largest singular value is the Hankel-norm of the system transfer-function matrix.

If \(G\) and \(Gr\) are the transfer-function matrices of the systems \((A, B, C, D)\) and \((A_r, B_r, C_r, D_r)\), respectively, then the approximation error \(G-Gr\) satisfies the inequalities

\[
\text{HSV}(r+1) \leq \text{INFNORM}(G-Gr) \leq 2 \times (\text{HSV}(r+1) + \ldots + \text{HSV}(n)),
\]

where \(\text{HSV}(i),\ i = 1, \ldots, n\) are the Hankel-singular values of the system \((A, B, C, D)\) and \(\text{INFNORM}(G)\) is the infinity-norm of \(G\).

The right-hand side above is the optionally computed error bound for the maximum approximation error.

Options:

Next case

Database Structure:

input:
linear system, state space

output:
<Result S-Name>
  _LS  linear system, state space
  _MAXERR  matrix for error bound
  _HSV  matrix for Hankel singular values,

Dialog:

-------- requested input
LS RS-NAME:
  -> linear system name
or carriage return to end the module

INPUT SIGNALS - U,V,ALLIN ?
-> select inputs, default is U
OUTPUT SIGNALS - YM,YA,ALLOUT ?
-> select outputs, default is YM

----- end

1. Level commands:

------------------

SPA >C>
PARAM : set parameter values for model reduction
PLIST  : set parameter values for listing of results
POUT  : set parameter values for output on database
COMPUTE : perform model reduction
          If one of outputting parameters is set then:
          ------- requested input
          <Result S-NAME> ( <Default S-NAME> ):
          -> If a blank is input, the <Default S-Name> is used.
             Otherwise the user must input a new S-Name under
             which the result will be stored according to the
             parameters selected through POUT.
             The <Default S-name> is the S-name of the input
             linear system appended by _SPA.index, where index
             is the next free value. If one does not use the
             <Default S-Name>, there could already exist other
             objects under the chosen < Result S-Name>. Then
             the action taken depends on a status variable:
             /SDEVERIF is set: It is asked whether all objects
             can be or not deleted. If objects
             should be not deleted, then a new
             <Result S-Name> is asked for.
             
             /SDENOVER is set: All objects under the given
             structure are deleted.

If the resulting reduced system is stored:
          ------- requested input
          Text for dataobject LS:
          ------- end
          ------- end
END : terminate module

2. Level Commands:

------------------

SPA_PARAM >P>
BALANCE : 0/1 (INTEGER def: 0)
  0 : use the square-root balancing-free SPA method
  1 : use the square-root SPA method
ORDER : desired order of reduced model (INTEGER def: 0)
  0 : order chosen in accordance with the value of TOLHSV
  r : desired order of the reduced model (r >= 0)
TOLHSV : tolerance on Hankel singular values (DOUBLE def: 0.0)
  If ORDER = 0 and TOLHSV > 0, then the order of the computed
reduced system is equal to the number of Hankel-singular values less than or equal to TOLHSV.
When TOLHSV .LE. 0, an internally computed default value TOLHSV = n*EPS*HNORM(A,B,C) is used, where EPS is the machine precision and HNORM(A,B,C) is the Hankel-norm of the original model (largest Hankel singular value).

MRTOL : tolerance for zero Hankel singular values
        (DOUBLE def: 0.0)
This tolerance is used to compute the order of the minimal realization of the original system.
When MRTOL .LE. 0, an internally computed default value MRTOL = n*EPS*HNORM(A,B,C) is used.

SPA_PLIST >P>
HSV : 0/1
0 : do not list the Hankel singular values
1 : list the Hankel singular values
MAXERR : 0/1
0 : do not list the maximum approximation error
1 : list the maximum approximation error

SPA_POUT >P>
REDSYS : 0/1
0 : do not store the resulting reduced order system
1 : store the resulting reduced order system
HSV : 0/1
0 : do not store the Hankel singular values
1 : store the Hankel singular values
MAXERR : 0/1
0 : do not list the maximum approximation error
1 : list the maximum approximation error

Monitor module initialization:
module name : SPA
module group : LSRED
version number : 1.0
FORTRAN name : PESPA

File input/output:
none

Method:
Literature
/1/ Varga A.
Balancing-free square-root algorithm for computing singular perturbation approximations. Proc. 30-th IEEE CDC, Brighton,

Remarks:
- The reduced model is stable and minimal if HSV(r) > HSV(r+1).
- The computed SPA preserves the DC-gain of the original system.
- The reduced model is not balanced if BALANCE = 0. Balanced reduced models can be computed by setting BALANCE = 1.
- If TOLHSV = MRTOL > 0, then the performed reduction is equivalent to the square-root B & T method, if BALANCE = 1, or to the balancing-free square root B & T method, if BALANCE = 0.

Life cycle:
1993 Aug A. Varga DLR FF-DF: coded

Packages required:
none

Libraries required:
RASP, LAPACK, BLAS (1,2,3)

Example:
Given the continuous-time system example used by the BTA module, the following fifth order reduced model is obtained by setting BALANCE = 0, ORDER = 0, TOLHSV = 0.1, MRTOL = 1.E-14:

\[
\begin{pmatrix}
1.3960 & -5.1248 & 0 & 0 & -4.4331 \\
4.1411 & -3.8605 & 0 & 0 & -0.6738 \\
0 & 0 & 0.5847 & -1.9230 & 0 \\
0 & 0 & 4.3823 & -3.2922 & 0 \\
-1.3261 & 1.7851 & 0 & 0 & -0.2249
\end{pmatrix}
\]

\[
\begin{pmatrix}
-0.2901 & 0.2901 \\
3.4004 & -3.4004 \\
0.6379 & 0.6379 \\
-3.9315 & -3.9315 \\
-1.9813 & 1.9813
\end{pmatrix}
\]

\[
\begin{pmatrix}
-0.6570 & -0.2053 & 0.6416 & 0.2526 & 0.0364 \\
0.1094 & -0.4875 & 0 & 0 & -0.8641 \\
0.6570 & 0.2053 & 0.6416 & 0.2526 & 0.0364 \\
0.0498 & -0.0007 \\
0.0010 & -0.0010 \\
-0.0007 & 0.0498
\end{pmatrix}
\]

The computed Hankel singular values are:

\[
2.5139 \quad 2.0846 \quad 1.9178 \quad 0.7666 \quad 0.5473 \quad 0.0253 \quad 0.0246
\]
HANKEL - Hankel-norm approximation of stable linear systems

Monitormodule purpose:
Given an original stable state-space model \((A, B, C, D)\) of order \(n\),
the Hankel-norm approximation (HNA) method is used to compute a
reduced order model \((\bar{A}, \bar{B}, \bar{C}, \bar{D})\) of order \(r\) by using the optimal
Hankel-norm approximation method /1/, in conjunction with square-root
balancing /2/.

Optionally displayed results:
- Hankel singular values
- Error bound for the maximum approximation error

Optional outputs on database are:
- reduced order system
- Hankel singular values
- Error bound for the maximum approximation error

The Hankel singular values are the square-roots of the eigenvalues of
product of controllability and observability gramians. The largest
singular value is the Hankel-norm of the system transfer-function
matrix.

If \(G\) and \(G_r\) are the transfer-function matrices of the systems
\((A, B, C, D)\) and \((\bar{A}, \bar{B}, \bar{C}, \bar{D})\), respectively, then the approximation
error \(G-G_r\) satisfies the inequalities

\[
HSV(r+1) \leq \text{INFNORM}(G-G_r) \leq 2^*( HSV(r+1) + \ldots + HSV(n) ),
\]

where \(HSV(i), i = 1, \ldots, n\) are the Hankel-singular values of the
system \((A, B, C, D)\) and \(\text{INFNORM}(G)\) is the infinity-norm of \(G\).
Moreover, the computed reduced model is optimal for the Hankel-norm,
that is, the approximation error \(G-G_r\) satisfies \(\text{HNORM}(G-G_r) = HSV(r+1)\),
where \(\text{HNORM}(G)\) is the Hankel-norm of \(G\).
The right-hand side above is the optionally computed error bound for
the maximum approximation error.

Options:
- Next case

Database Structure:
- input:
  - linear system, state space
- output:
  - <Result S-Name>
    - _LS
    - _MAXERR
    - _HSV
      - linear system, state space
      - matrix for error bound
      - matrix for Hankel singular values,
Dialog:
------ requested input
LS RS-NAME:
  -> linear system name
      or carriage return to end the module
INPUT SIGNALS - U,V,ALLIN ?
  -> select inputs, default is U
OUTPUT SIGNALS - YM,YA,ALLOUT ?
  -> select outputs, default is YM
------ end

1. Level commands:
---------------------

HNA >C>

PARAM : set parameter values for model reduction
PLIST : set parameter values for listing of results
POUT : set parameter values for output on database
COMPUTE : perform model reduction

If one of outputting parameters is set then:
------ requested input
<Result S-NAME> ( <Default S-NAME> ):
  -> If a blank is input, the <Default S-Name> is used.
     Otherwise the user must input a new S-Name under
     which the result will be stored according to the
     parameters selected through POUT.
     The <Default S-name> is the S-name of the input
     linear system appended by _HNA.index, where index
     is the next free value. If one does not use the
     <Default S-Name>, there could already exist other
     objects under the chosen < Result S-Name>. Then
     the action taken depends on a status variable:
     /SDEVERIF is set: It is asked whether all objects
     can be or not deleted. If objects
     should be not deleted, then a new
     <Result S-Name> is asked for.
     /SDENOVER is set: All objects under the given
     structure are deleted.

If the resulting reduced system is stored:
------ requested input
      Text for dataobject LS:
------ end
------ end

END : terminate module

2. Level Commands:
---------------------

HNA_PARAM >P>

ORDER : desired order of reduced model       (INTEGER def: 0)
-1 : order chosen in accordance with the value of TOLHSV
r  : desired order of the reduced model (r >= 0).
      The resulting order is max(0,r-k+1), where k is the
multiplicity \((r+1)\)-th Hankel singular value.

\(TOLHSV\) : tolerance on Hankel singular values \((\text{DOUBLE def: 0.0})\)

If \(\text{ORDER = 0}\) and \(TOLHSV > 0\), then the order of the computed reduced system is equal to the number of Hankel-singular values less than or equal to \(TOLHSV\).

When \(TOLHSV \leq 0\), an internally computed default value \(TOLHSV = \sqrt{\text{eps}} \times HNORM(A,B,C)\) is used, where \(\text{eps}\) is the machine precision and \(HNORM(A,B,C)\) is the Hankel-norm of the original model (largest Hankel singular value).

\(\text{MRTOL}\) : tolerance for zero Hankel singular values \((\text{DOUBLE def: 0.0})\)

This tolerance is used to compute the order of the minimal realization of the original system.

When \(\text{MRTOL} \leq 0\), an internally computed default value \(\text{MRTOL} = \text{eps} \times HNORM(A,B,C)\) is used.

\(\text{HNA\_PLIST} >P<\)

\(HSV\) : 0/1 \((\text{INTEGER def: 0})\)

0 : do not list the Hankel singular values
1 : list the Hankel singular values

\(\text{MAXERR}\) : 0/1 \((\text{INTEGER def: 0})\)

0 : do not list the maximum approximation error
1 : list the maximum approximation error

\(\text{HNA\_POUT} >P<\)

\(\text{REDSYS}\) : 0/1 \((\text{INTEGER def: 1})\)

0 : do not store the resulting reduced order system
1 : store the resulting reduced order system

\(\text{HSV}\) : 0/1 \((\text{INTEGER def: 0})\)

0 : do not store the Hankel singular values
1 : store the Hankel singular values

\(\text{MAXERR}\) : 0/1 \((\text{INTEGER def: 0})\)

0 : do not list the maximum approximation error
1 : list the maximum approximation error

Monitor module initialization:

\text{module name} : \text{HNA}
\text{module group} : \text{LSRED}
\text{version number} : 1.0
\text{FORTRAN name} : \text{PEHNA}

File input/output:

\text{none}

Method:

\text{Literature}

\(1/\) Glover, K.

All optimal Hankel norm approximation of linear multivariable systems and their \(L\)-infinity error bounds,

\text{Int. J. Control, Vol. 35, pp. 1145-1193, 1984.}
/2/ Tombs M.S. and Postlethwaite I.
Truncated balanced realization of stable, non-minimal
state-space systems,

Remarks:
- The resulting reduced model (Ar,Br,Cr,Dr) is stable and minimal
  if HSV(r) > HSV(r+1).

Life cycle:
1993 Aug  A. Varga DLR FF-DF: coded

Packages required:
none

Libraries required:
RASP, LAPACK, BLAS (1,2,3)

Example:
Given the continuous-time system example used by the BTA module,
the following fifth order reduced model is obtained by setting
ORDER = -1, TCLHSV = 0.1, MRTOL = 1.E-14:

\[
\begin{bmatrix}
-0.5038 & -1.8355 & -2.6289 & 0 & 0 \\
5.3070 & -0.5038 & 3.2250 & 0 & 0 \\
0 & 0 & -1.5171 & 0 & 0 \\
0 & 0 & 0 & -1.2925 & 9.0718 \\
0 & 0 & 0 & -0.5047 & -1.2925 \\
\end{bmatrix}
\]

\[
\begin{bmatrix}
-0.3614 & 0.3614 \\
1.5343 & -1.5343 \\
-1.1096 & 1.1096 \\
-4.5325 & -4.5325 \\
0.7396 & 0.7396 \\
\end{bmatrix}
\]

\[
\begin{bmatrix}
-0.3055 & -1.8971 & -2.1124 & 0.4421 & 2.1023 \\
1.1112 & 0.0394 & -0.3119 & 0 & 0 \\
0.3055 & 1.8971 & 2.1124 & 0.4421 & 2.1023 \\
0.0126 & -0.0126 \\
0.0005 & -0.0005 \\
-0.0126 & 0.0126 \\
\end{bmatrix}
\]

The computed Hankel singular values are:

\[
2.5139 \quad 2.0846 \quad 1.9178 \quad 0.7666 \quad 0.5473 \quad 0.0253 \quad 0.0246
\]
MOnitormodule

BST - Balanced stochastic truncation of stable linear systems

Monitormodule purpose:
Given a linear stable state-space model \((A,B,C,D)\) of order \(n\),
the balanced stochastic truncation (BST) method is used
to compute a reduced order model \((Ar,Br,Cr,Dr)\) of order \(r\)
by using either:
(1) the square-root BST method /1/, or
(2) the square-root balancing-free BST method /2/.
The module can be also used for computing stochastically balanced
minimal state-space realizations of continuous-time or discrete-
time systems.

For the applicability of the BST method, the given system must have
the number of systems outputs less than or equal to the number of
systems inputs and the transfer-function matrix of the system \(G\),
must have no zeros on the imaginary axis for a continuous-time system
or on the unit circle for a discrete-time system.
In particular, the feedthrough matrix \(D\) must have maximal row rank.
If \(D\) has maximal column rank, the BST method can be employed
on the dual system with the transfer-function matrix \(G'\).

A parameter \(\text{ALPHA}\) can be used as a weight between the absolute and
relative errors. For \(\text{ALPHA} <= 0\), the BST method is performed on the
original system. If \(\text{ALPHA} > 0\), the BST method is performed on a
modified system with the transfer-function matrix \((G \text{ ALPH}A*\text{I})\).
This is the recommended approach to be used when the conditions on
the number inputs and outputs and/or on the rank of \(D\) are not
fulfilled by a given system.

Optionally displayed results:
- phase Hankel singular values (def. display)
- Error bound for the maximum relative error (def. display)

Optional outputs on database are:
- reduced order system (def. output)
- phase Hankel singular values (def. no output)
- Error bound for the maximum relative error (def. no output)

Let \(G\) be the transfer-function matrix of the system \((A,B,C,D)\) and
let \(W\) be the square minimal-phase stable spectral factor
of \(G*\text{CONJ}(G)\) satisfying
\[
\text{CONJ}(W)*W = G*\text{CONJ}(G),
\]
where \(\text{CONJ}(G(s)) = G'(-s)\) for a continuous-time system and
\(\text{CONJ}(G(z)) = G'(1/z)\) for a discrete-time system. The system with
the all-pass transfer-function matrix \(PH = \text{INV}(\text{CONJ}(W))*G\) is called
the phase-system associated with \(G\).
The phase Hankel singular values are the square-roots of the eigenvalues of the product of controllability and observability gramians of the stable projection of the phase system PH. The largest singular value is always less than or equal to 1.

If Gr is the transfer-function matrix of the reduced system (Ar,Br,Cr,Dr), then the relative approximation error satisfies the inequalities

\[ HSV(r+1) \leq \text{INFNORM}(\text{relerr}) \leq 2 \sum_{i=r+1}^{n} \frac{HSV(i)}{1-HSV(i)} \]

where INFNORM(G) is the infinity-norm of G. The relative error relerr is defined by the expression

\[ Gr = G*(I-\text{relerr}). \]

The right-hand side above is the optionally computed error bound for the maximum relative approximation error.

Options:
Next case

Database Structure:
input:
linear system, state space
output:
<Result S-Name>
  _LS linear system, state space
  _MAXERR matrix for relative error bound
  _PHSV matrix for phase Hankel singular values,

Dialog:
------ requested input
LS RS-NAME:
  \( \rightarrow \) linear system name
  \( \rightarrow \) carriage return to end the module
INPUT SIGNALS - U,V,ALLIN ?
  \( \rightarrow \) select inputs, default is \( U \)
OUTPUT SIGNALS - YM,YA,ALLOUT ?
  \( \rightarrow \) select outputs, default is YM
------ end

1. Level commands:
-------------
BST >C>

PARAM : set parameter values for model reduction
PLIST : set parameter values for listing of results
POUT : set parameter values for output on database
COMPUTE : perform model reduction
  If one of outputting parameters is set then:
  ------ requested input
<Result S-NAME> ( <Default S-NAME> ):
  -> If a blank is input, the <Default S-NAME> is used.
  Otherwise the user must input a new S-Name under
  which the result will be stored according to the
  parameters selected through POUT.
  The <Default S-name> is the S-name of the input
  linear system appended by _BST.index, where index
  is the next free value. If one does not use the
  <Default S-Name>, there could already exist other
  objects under the chosen <Result S-Name>. Then
  the action taken depends on a status variable:
  /SDEVERIF is set: It is asked whether all objects
  can be or not deleted. If objects
  should be not deleted, then a new
  <Result S-Name> is asked for.
  /SDENOVER is set: All objects under the given
  structure are deleted.

  If the resulting reduced system is stored:
    ------ requested input
    Text for dataobject LS:
      ------ end
    ------ end
  END : terminate module

2. Level Commands:

--------------------------

BST_PARAM >>P>

BALANCE : 0/1 (INTEGER def: 0)
  0 : use the square-root balancing-free BST method
  1 : use the square-root BST method

ORDER : desired order of reduced model (INTEGER def: 0)
  0 : order chosen in accordance with the value of TOLPHSV
  k : desired order of the reduced model (k >= 0)

TOLPHSV : tolerance on Hankel singular values (DOUBLE def: 0.0)
  If ORDER = 0 and TOLPHSV > 0, then the order of the
  computed reduced system is equal to the number of phase
  Hankel singular values less than or equal to TOLPHSV.
  When TOLPHSV .LE. 0, an internally computed default value
  TOLPHSV = n*EPS is used, where EPS is the machine
  precision.

ALPHA : absolute/relative error weight (DOUBLE def: 0.0)
  ALPHA > 0 specifies the absolute/relative error weighting
  parameter. A large positive value of ALPHA favours the
  minimization of the absolute approximation error, while a
  small value of ALPHA is appropriate for the minimization
  of the relative error.
  ALPHA <= 0 means pure relative error method and can be
  used only if D has full row rank.

BST_LIST >>P>

PHSV : 0/1 (INTEGER def: 1)
0: do not list the phase Hankel singular values
1: list the phase Hankel singular values

MAXRERR : 0/1
0: do not list the maximum relative approximation error
1: list the maximum relative approximation error

BST_POUT >>P>
REDSYS : 0/1
0: do not store the resulting reduced order system
1: store the resulting reduced order system
PHSV : 0/1
0: do not store the phase Hankel singular values
1: store the phase Hankel singular values
MAXRERR : 0/1
0: do not list the maximum relative approximation error
1: list the maximum relative approximation error

Monitor module initialization:
module name : BST
module group : LSRED
version number : 1.0
FORTRAN name : PEBST

File input/output:
none

Method:

Literature
/1/ Safonov M.G. and Chiang R.Y.
Model reduction for robust control: a Schur relative
error method,
/2/ Varga A. and Fasol K.H.
A New Square-Root Balancing-Free Stochastic Truncation Model
Reduction Algorithm,

Remarks:
- The reduced model is stable and minimal if HSV(r) > HSV(r+1).
- The reduced model is not stochastically balanced if BALANCE
  is set to 0.
- A stochastically balanced minimal realization can be computed
  by setting BALANCE = 1, ORDER = 0 and TOLPHSV = 0.

Life cycle:
1993 Aug  A. Varga DLR FF-DF: coded

Packages required:
none

Libraries required:
RASP, LAPACK, BLAS (1,2,3)

Example:
Given the continuous-time system used by the BTA module, the matrices corresponding to the stochastically balanced system $(G \ I)$ computed by setting $\text{BALANCE} = 1$, $\text{ORDER} = 0$, $\text{TOLPHSV} = 0$, $\text{ALPHA} = 1$ are:

$$\begin{pmatrix}
-0.1996 & 0 & -2.8560 & 0 & 1.4913 & -0.5003 & 0 \\
0 & -0.2774 & 0 & -2.3059 & 0 & 0 & -0.5931 \\
2.9100 & 0 & -0.1252 & 0 & -0.0070 & -0.3191 & 0 \\
\end{pmatrix}$$

$$\begin{pmatrix}
-1.5039 & 0 & 0.6922 & 0 & -1.9158 & 2.4031 & 0 \\
-0.5023 & 0 & 0.4349 & 0 & -2.5019 & -13.6311 & 0 \\
0 & -0.5931 & 0 & 2.3648 & 0 & 0 & -13.5759 \\
\end{pmatrix}$$

$$\begin{pmatrix}
0.4191 & -0.4191 \\
-0.4857 & -0.4857 \\
-0.3173 & 0.3173 \\
\end{pmatrix}$$

$$\begin{pmatrix}
0.9524 & 0.9524 \\
0.8724 & -0.8724 \\
0.5402 & -0.5402 \\
-0.5320 & -0.5320 \\
\end{pmatrix}$$

$$\begin{pmatrix}
1.8572 & -1.7570 & 0.6352 & -1.1934 & -0.9958 & 0.5404 & -0.5323 \\
-0.1816 & 0 & 0.8944 & 0 & 0.4040 & 0.0137 & 0 \\
-1.8572 & -1.7570 & -0.6352 & -1.1934 & 0.9958 & -0.5404 & -0.5323 \\
\end{pmatrix}$$

The computed singular values are:

0.8803 0.8506 0.8038 0.4494 0.3973 0.0214 0.0209

The following reduced order system is obtained by setting $\text{BALANCE} = 0$, $\text{ORDER} = 0$, $\text{TOLPHSV} = 0.1$, $\text{ALPHA} = 1$:

$$\begin{pmatrix}
1.2729 & 0 & -6.5947 & 0 & -3.4229 \\
0 & 0.8169 & 0 & 2.4821 & 0 \\
2.9889 & 0 & -2.9028 & 0 & 0.3692 \\
0 & -3.3921 & 0 & -3.1126 & 0 \\
-1.4767 & 0 & 2.0339 & 0 & -0.6107 \\
\end{pmatrix}$$

$$\begin{pmatrix}
0.1331 & -0.1331 \\
-0.0862 & -0.0862 \\
2.6777 & -2.6777 \\
-3.5767 & -3.5767 \\
-2.3033 & 2.3033 \\
\end{pmatrix}$$

$$\begin{pmatrix}
-0.6907 & -0.6882 & -0.0779 & 0.958 & -0.0038 \\
0.0676 & 0 & -0.6532 & 0 & -0.7522 \\
0.6907 & -0.6882 & 0.0779 & 0.958 & 0.0038 \\
\end{pmatrix}$$
MONITOR MODULE

SPF - Singular perturbation formulas for linear systems

Monitor module purpose:
Given a linear state-space model \((A,B,C,D)\) with matrices
\(A, B\) and \(C\) partitioned conformally as

\[
\begin{pmatrix}
A_{11} & A_{12} \\
A_{21} & A_{22}
\end{pmatrix},
\begin{pmatrix}
B_1 \\
B_2
\end{pmatrix},
\begin{pmatrix}
C_1 & C_2
\end{pmatrix},
\]

the singular perturbation formulas are used to compute the matrices of the reduced order system \((A_r,B_r,C_r,D_r)\):

\[
A_r = A_{11} + A_{12}*(g*I-A_{22})^{-1}A_{21},
B_r = B_1 + A_{12}*(g*I-A_{22})^{-1}B_2
\]

\[
C_r = C_1 + C_2*(g*I-A_{22})^{-1}A_{21},
D_r = D + C_2*(g*I-A_{22})^{-1}B_2
\]

where \(g = 0\) for a continuous-time system and \(g = 1\) for a discrete-time system.

Optional output on database is:
- reduced order system (def. output)

Options:
Next case

Database Structure:
input:
linear system, state space
output:
<Result S-Name>
\_LS linear system, state space

Dialog:

-------- requested input
LS RS-NAME:
--> linear system name
or carriage return to end the module
INPUT SIGNALS - U,V,ALLIN ?
--> select inputs, default is U
OUTPUT SIGNALS - YM,YA,ALLOUT ?
--> select outputs, default is YM
-------- end

1. Level commands:
---------------
SPF >C
PARAM : set parameter value for model reduction
POUT : set parameter value for output on database
COMPUTE : perform model reduction
If the outputing parameter is set then:

----- requested input
<Result S-NAME> ( <Default S-NAME> ):
   -> If a blank is input, the <Default S-Name> is used.
   Otherwise the user must input a new S-Name under
   which the result will be stored.
The <Default S-name> is the S-name of the input
system appended by _SPF.index, where index is the
next free value. If one does not use the
<Default S-Name>, there could already exist other
objects under the chosen < Result S-Name >.
Then the action taken depends on a status variable:
/SDEVERIF is set: It is asked whether all objects
can be or not deleted. If objects
should be not deleted, then a new
<Result S-Name> is asked for.
/SDENOVER is set: All objects under the given
structure are deleted.

If the resulting reduced system is stored:

----- requested input
Text for dataobject LS:

----- end

END : terminate module

2. Level Commands:

-------------
SPF_PARAM >P>
ORDER : desired order of reduced model (INTEGER def: 0)
    0 : order chosen in accordance with the value of TOLHSV
    r : desired order of the reduced model (r \geq 0)

SPF_POUT >P>
REDSYS : 0/1 (INTEGER def: 1)
    0 : do not store the resulting reduced order system
    1 : store the resulting reduced order system

Monitor module initialization:
module name : SPF
module group : LSRED
version number : 1.0
FORTRAN name : PESPF

File input/output:
none

Method:

Literature
1/ Liu Y. and Anderson B.D.O.
Singular perturbation approximation of balanced systems,
Remarks:
- For a stable system \((A,B,C,D)\), the resulting reduced system
  \((Ar,Br,Cr,Dr)\) has the same steady-state gain as the given
  full order system.
- If for a stable system \((A,B,C,D)\), the order of the reduced model
  is set to \(r = 0\), the resulting \(Dr\) is the steady-state gain matrix
  of the given system.

Life cycle:
1993 Aug  A. Varga DLR FF-DF: coded

Packages required:
none

Libraries required:
RASP, LAPACK, BLAS (1,2,3)

Example:
Given the balanced continuous-time system resulted by using the
BTA module, the following fifth order reduced model represents
its balanced singular perturbation approximation /1/:

\[
Ar = \begin{pmatrix}
-0.3813 & 3.0418 & 0 & 0 & 1.0286 \\
-3.0996 & -2.635 & 0 & 0 & -0.3187 \\
0 & 0 & -0.5557 & -2.3036 & 0 \\
0 & 0 & 2.3036 & -2.1518 & 0 \\
-1.0479 & -0.6685 & 0 & 0 & -2.0447 \\
\end{pmatrix}
\]

\[
Br = \begin{pmatrix}
-0.9790 & 0.9790 \\
-0.7412 & 0.7412 \\
-1.0323 & -1.0323 \\
1.2844 & 1.2844 \\
-1.0578 & 1.0578 \\
\end{pmatrix}
\]

\[
Cr = \begin{pmatrix}
-0.9723 & 0.6513 & -1.0323 & -1.2844 & 0.9914 \\
0.1620 & 0.5004 & 0 & 0 & -0.5217 \\
0.9723 & -0.6513 & -1.0323 & -1.2844 & -0.9914 \\
0.0498 & -0.0007 \\
0.0010 & -0.0010 \\
-0.0007 & 0.0498 \\
\end{pmatrix}
\]
MONITORMODULE
BIL - Bilinear transformation of linear systems

Monitor module purpose:
Given a linear state-space model (A,B,C,D), a transformation on
the parameters of the system is performed which is equivalent to
a bilinear transformation of the corresponding transfer function
matrix /1/.

For a continuous-time system, the resulting matrices correspond to the
continuous-to-discrete bilinear transformation

\[
\begin{align*}
\beta + s \\
\beta - s
\end{align*}
\]

For a discrete-time system, the resulting matrices correspond to the
discrete-to-continuous bilinear transformation

\[
\begin{align*}
z - \alpha \\
z + \alpha
\end{align*}
\]

Optional output on database is:
- transformed system (def. output)

Options:
Next case

Database Structure:
input:
linear system, state space
output:
<Result S-Name>
  _LS  linear system, state space

Dialog:
------ requested input
LS RS-NAME:
  -> linear system name
  or carriage return to end the module
INPUT SIGNALS - U,V,ALLIN ?
  -> select inputs, default is U
OUTPUT SIGNALS - YM,YA,ALLOUT ?
  -> select outputs, default is YM
------ end

1. Level commands:
-----------------
BIL >C>
PARAM : set parameter values for model reduction
POUT : set parameter value for output on database
COMPUTE : perform model reduction

If the output parameter is set then:
------ requested input
<Result S-NAME> ( <Default S-NAME> ) :

--> If a blank is input, the <Default S-Name> is used.
Otherwise the user must input a new S-Name under which the result will be stored.
The <Default S-Name> is the S-name of the input linear system appended by _BIL.index, where index is the next free value. If one does not use the <Default S-Name>, there could already exist other objects under the chosen < Result S-Name>. Then the action taken depends on a status variable:
/SDEVIF is set: It is asked whether all objects can be or not deleted. If objects should be not deleted, then a new <Result S-Name> is asked for.

/SDENIVER is set: All objects under the given structure are deleted.

If the resulting transformed system is stored:
------ requested input
Text for dataobject LS:
------ end
------ end

END : terminate module

2. Level Commands:
-----------------------

BIL_PARAM >P>

ALPHA : parameter of bilinear transformation (DOUBLE def: 1.0)
Specifies a non-zero value for parameter alpha.

BETA : parameter of bilinear transformation (DOUBLE def: 1.0)
Specifies a non-zero value for parameter beta.

BIL_POUT >P>

TRSYS : 0/1 (INTEGER def: 1)
0 : do not store the resulting transformed system
1 : store the resulting transformed system

Monitor module initialization:
module name : BIL
module group : LSRED
version number : 1.0
FORTRAN name : PEBIL

File input/output:
none

Method:
Literature

/1/ Al-Saggaf U.M. and Franklin G.F.
Model reduction via balanced realizations: a extension and

Remarks:
- For stable systems, the recommended values of the bilinear
  transformation parameters are: alpha = 1 and beta = 1.
  By using these values, to left half-plane poles of continuous-time
  systems correspond poles of the resulting discrete-time systems
  inside the unit circle and vice versa.
- For an unstable continuous-time system, beta should be not a pole
  of the original system.
- For an unstable discrete-time system, -alpha should be not a pole
  of the original system.

Life cycle:
1993 Aug  A. Varga DLR FF-DF: coded

Packages required:
none

Libraries required:
RASP, LAPACK, BLAS (1,2,3)

Example:
none
MONITORMODULE

ASD - Additive spectral decomposition of linear systems

Monitor module purpose:
Given a state-space model \((A,B,C,D)\) corresponding to a transfer-function matrix \(G\), an additive spectral decomposition of \(G\) is computed as

\[ G = G_1 + G_2 \]

where \(G_1\) is the stable projection of \(G\) having only poles in a stability region and \(G_2\) is the unstable projection of \(G\) having only poles outside of this region. The stability region is defined by a parameter ALPHA specifying its boundary. For continuous-time systems the stability region is the open left half plane \(\text{Real}(s) < \text{ALPHA}\), while for discrete-time systems the stability region is the interior of the circle in origin of radius \(\text{ALPHA}\) ( \(\text{abs}(z) < \text{ALPHA}\) ). The state-space representations of \(G_1\) and \(G_2\) are computed as

\[ G_1 = (A_1,B_1,C_1,D_1), \quad G_2 = (A_2,B_2,C_2,D_2), \]

where \(A_1\) has only eigenvalues in the stability region and \(A_2\) has only eigenvalues outside of this region. The feedthrough matrices \(D_1\) and \(D_2\) can be computed as \(D_1 = \text{BETA}*D\) and \(D_2 = (1-\text{BETA})*D\), where \(\text{BETA}\) can be arbitrarily chosen.

This module can be used to compute stable and antistable projections in order to perform further order reduction of the stable part. It can be also used for determining fast and slow modes decompositions in order to remove the fast modes of a given system.

Optional outputs on database are:
- stable projection (def. output)
- unstable projection (def. output)

Options:
Next case

Database Structure:
input:
linear system, state space
output:
<Result S-Name>
  _STABLE_LS linear system, state space
  _UNSTABLE_LS linear system, state space

Dialog:
------- requested input
LS RS-NAME:
-> linear system name
or carriage return to end the module
INPUT SIGNALS - U, V, ALLIN?
  -> select inputs, default is U
OUTPUT SIGNALS - YM, YA, ALLOUT?
  -> select outputs, default is YM
------ end

1. Level commands:
---------------------

ASD >C>
PARAM : set parameter values for model reduction
POUT : set parameter values for output on database
COMPUTE : compute the additive decomposition

If one of outputting parameters is set then:
------ requested input

<Result S-NAME> ( <Default S-NAME> ) :
  -> If a blank is input, the <Default S-Name> is used.
Otherwise the user must input a new S-Name under
which the result will be stored according to the
parameters selected through POUT.
The <Default S-name> is the S-name of the input
linear system appended by .ASD.index, where index
is the next free value. If one does not use the
<Default S-Name>, there could already exist other
objects under the chosen < Result S-Name >. Then
the action taken depends on a status variable:
/SDEVERIF is set: It is asked whether all objects
can be or not deleted. If objects
should be not deleted, then a new
<Result S-Name> is asked for.

/SDENOVER is set: All objects under the given
structure are deleted.

If the stable projection is stored:
------ requested input
  Text for dataobject LS:
------ end
If the unstable projection is stored:
------ requested input
  Text for dataobject LS:
------ end
------ end
END : terminate module

2. Level Commands:
-------------------

ASD_PARAM >P>
UNSTABLE : 0/1 (INTEGER def: 0)
  0 : the poles of G1 belong to the stable region
  1 : the poles of G1 belong to the unstable region
UNSTABLE = 1 can be used to compute conveniently a
slow/fast decomposition of the given G. In this case
the unstable region corresponds to the domain of
dominant system poles.

**ALPHA** : boundary of stability region (DOUBLE def: stpar)
Specifies the boundary of the stability region for the eigenvalues of A:
- for continuous-time systems ALPHA represents the maximum admissible value for the real parts of the eigenvalues.
  Default value of stpar is 0.
- for discrete-time systems ALPHA represents the maximum admissible value for the moduli of the eigenvalues.
  Default value of stpar is 1.0.

**BETA** : splitting coefficient for D matrix (DOUBLE def: 1.0)
The matrices D1 and D2 satisfy \( D = D_1 + D_2 \), and are computed as \( D_1 = \beta \cdot D \) and \( D_2 = (1-\beta) \cdot D \).

**ASD_POUT >P>
**
**SYS1** : 0/1 (INTEGER def: 1)
- 0 : do not store the projection G1
- 1 : store the projection G1

**SYS2** : 0/1 (INTEGER def: 1)
- 0 : do not store the projection G2
- 1 : store the projection G2

Monitor module initialization:
- **module name** : ASD
- **module group** : LSRED
- **version number** : 1.0
- **FORTRAN name** : PEASD

File input/output:
- none

Method:
A similarity transformation matrix \( T \) is computed such that the transformed system

\[
\begin{pmatrix}
  - & - & - & 1 \\
  A, B, C, D & = & T & A T, T & B, C T, D
\end{pmatrix}
\]

has the state-matrix \( A \) in a block diagonal form. If we partition the transformed systems matrices conformally with the structure of the state-matrix

\[
A = \begin{pmatrix}
  A_1 & 0 \\
  0 & A_2
\end{pmatrix}, \quad B = \begin{pmatrix}
  B_1 \\
  B_2
\end{pmatrix}, \quad C = \begin{pmatrix}
  C_1 & C_2
\end{pmatrix},
\]

then \( A_1 \) has eigenvalues in a region of interest and \( A_2 \) outside of that region. The region of interest is the stability region if \( \text{UNSTABLE} = 0 \) and the instability region if \( \text{UNSTABLE} = 1 \).

The decomposition (2) is used to define the additive decomposition of the transfer-function matrix \( G \) as
\[ G = G_1 + G_2 \]

where \( G_1 = (A_1, B_1, C_1, D_1) \) and \( G_2 = (A_2, B_2, C_2, D_2) \).

Literature

/1/ Safonov, M.G., Jonckheere, E.A., Verma, M. and Limebeer, D.J.
Synthesis of positive real multivariable feedback systems,

Remarks:
none

Life cycle:
1993 Aug  A. Varga DLR FF-DF: coded

Packages required:
none

Libraries required:
RASP, LAPACK, BLAS (1,2,3)

Example:
Given the unstable continuous-time system \((A,B,C,D)\) with the following matrices:

\[
\begin{pmatrix}
-0.04165 & 0 & 4.92 & 0.492 & 0 & 0 & 0 \\
-5.21 & -12.5 & 0 & 0 & 0 & 0 & 0 \\
0 & 3.33 & -3.33 & 0 & 0 & 0 & 0 \\
\end{pmatrix}
\]

\[
A = \begin{pmatrix}
0.545 & 0 & 0 & 0 & 0.0545 & 0 & 0 \\
0 & 0 & 0 & -0.492 & 0.004165 & 0 & 4.92 \\
0 & 0 & 0 & 0 & 0.521 & -12.5 & 0 \\
0 & 0 & 0 & 0 & 0 & 3.33 & -3.33 \\
\end{pmatrix}
\]

\[
B = \begin{pmatrix}
0 & 0 \\
0 & 0 \\
0 & 0 \\
0 & 0 \\
0 & 0 \\
12.5 & 0 \\
0 & 0 \\
\end{pmatrix}
\]

\[
C = \begin{pmatrix}
0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 \\
\end{pmatrix}
\]

\[ D = 0 \]

A slow/fast decomposition can be computed which can serve as basis to compute a modal approximation of the original system.

By setting \textsc{unstable} = 1 and \textsc{beta} = 1, the matrices of the slow subsystem \( G_1 \) and of the fast subsystem \( G_2 \) are:

\[
\begin{pmatrix}
-1.4178 & -5.1682 & 0.5481 & 2.5944 & 0.0695 \\
0.9109 & -1.4178 & 0.1460 & 0.4718 & 0.3044 \\
\end{pmatrix}
\]

\[
A_1 = \begin{pmatrix}
0 & 0 & 0.1605 & -0.4948 & 4.7654 \\
0 & 0 & 0.0474 & 0.1605 & -0.3769 \\
0 & 0 & 0 & 0 & -3.5957 \\
\end{pmatrix}
\]
\[
B_1 = \begin{pmatrix}
2.8493 & 0.0351 \\
2.9533 & 0.0993 \\
-0.3203 & 1.6526 \\
-1.0775 & -0.1609 \\
0.0089 & -4.7125 \\
\end{pmatrix}
\]

\[
C_1 = \begin{pmatrix}
-0.8659 & 0.2787 & -0.0185 & -0.2005 & -0.0002 \\
0.0797 & -0.3951 & -0.0427 & -0.9141 & 0.0068 \\
-0.0165 & -0.0645 & -0.9935 & 0.0732 & 0.0376 \\
\end{pmatrix}
\]

\[
A_2 = \begin{pmatrix}
-13.1627 & 0 \\
0 & -12.4245 \\
\end{pmatrix}
\]

\[
B_2 = \begin{pmatrix}
-11.4205 & -0.0015 \\
-0.0020 & 12.4858 \\
\end{pmatrix}
\]

\[
C_2 = \begin{pmatrix}
-0.1245 & 0 \\
0.0052 & -0.0005 \\
0.0002 & 0.1472 \\
\end{pmatrix}
\]

The slow subsystem of order can be used as a dominant modes approximation of the original system.
MONITORMODULE
LCF - Left coprime factorization of transfer-function matrices

Monitor module purpose:
Given an n-th order state-space model \((A,B,C,D)\) corresponding to a pxm transfer-function matrix \(G\), a state-space representation \((AQR, BQR, CQR, DQR)\) of the augmented transfer-function matrix \((Q R)\)

\[-1\]
is computed defining a left coprime factorization \((LCF) G = R \ast Q\).
The state matrix AQR can be determined such that its eigenvalues are inside a specified stability region of the complex plane.
The computation of the factorization is based on a pole assignment method described in /1/.

Optional outputs on database are:
- left coprime factorization (in augmented form) (def. output)

The q-th order state-space representations of the factors \(Q\) and \(R\) are constructed in the forms

\[Q = (Z'*(A+H*C)*Z, Z'*(B+H*D), C*Z, D)\]

and

\[R = (Z'*(A+H*B)*Z, Z'*H, C*Z, I)\]

where \(Z\) is an \(n \times q\) matrix with orthonormal columns and \(H\) is an \(n \times p\) output injection matrix assigning the eigenvalues of \(A\) in the specified stability region. If the given state-space representation is detectable, the order \(q\) of the resulting state-space representation of \((Q R)\) is equal to \(n\). If the given state-space representation is not detectable, the undetectable part of the original system is automatically deflated and the resulting \(q\) is less than \(n\).
The matrices AQR, BQR, CQR and DQR are computed as:

\[AQR = Z'*(A+H*C)*Z,\quad BQR = (Z'*B+Z'*H*D, Z'*H),\]
\[CQR = C*Z,\quad DQR = (D I).\]

Options:
Next case

Database Structure:
input:
linear system, state space
output:
<Result S-Name>
   _LS  linear system, state space

Dialog:
-------- requested input
LS RS-NAME:
   -> linear system name
or carriage return to end the module

INPUT SIGNALS - U,V,ALLIN ?
   -> select inputs, default is U
OUTPUT SIGNALS - YM,YA,ALLOUT ?
   -> select outputs, default is YM

---- end

1. Level commands:

-------------------

LCF >C>

PARAM : set parameter values for model reduction
POUT : set parameter values for output on database
COMPUTE : compute the factorization

If the outputing parameter is set then:
------ requested input

<Result S-NAME> ( <Default S-NAME> ):
   -> If a blank is input, the <Default S-NAME> is used.
   Otherwise the user must input a new S-Name under
   which the result will be stored according to the
   parameters selected through POUT.
   The <Default S-name> is the S-name of the input
   linear system appended by _LCF.index, where index
   is the next free value. If one does not use the
   <Default S-Name>, there could already exist other
   objects under the chosen < Result S-Name>. Then the
   action taken depends on a status variable:
   /SDEVERIF is set: It is asked whether all objects
   can be or not deleted. If objects
   should be not deleted, then a new
   <Result S-Name> is asked for.
   /SDENOVER is set: All objects under the given
   structure are deleted.

If the augmented factorization is stored:
------ requested input

   Text for dataobject LS:
   ------ end
   ------ end
END : terminate module

2. Level Commands:

-------------------

LCF_PARAM >P>

ALPHA : boundary of stability region  (DOUBLE def: stpar)
   Specifies the boundary of the stability region for
   the eigenvalues of A:
   - for continuous-time systems ALPHA represents the maximum
     admissible value for the real parts of the eigenvalues.
     Default value of stpar is 0.
   - for discrete-time systemy ALPHA represents the maximum
     admissible value for the moduli of the eigenvalues.
     Default value of stpar is 1.0.
The eigenvalues lying inside the stability region will be not modified.

**STABDEC**: desired stability degree  
(DOUBLE def: sdpar)

The eigenvalues of A, lying outside the stability region defined by ALPHA, will be moved to locations corresponding to a stability degree STABDEC, namely:
- for a continuous-time system, the modified eigenvalues will have real parts equal to STABDEC and unmodified imaginary parts. Default value of sdpar is -0.2.
- for a discrete-time system, the modified eigenvalues will have moduli equal to STABDEC.
Default value of sdpar is 0.8.

**TOL**: tolerance for observability tests  
(DOUBLE def: tolpar)

Specifies the absolute tolerance level below which the elements of C are considered zero.

Default value of TOL is tolpar = n*EPS*NORM(C), where NORM(C) denotes the infinity-norm of C and EPS is the machine precision.

**LCP_POUT**>

**LCP**: 0/1  
(INTEGER def: 1)

0 : do not store the augmented LCF
1 : store the augmented LCF

Monitor module initialization:

- **module name**: LCF
- **module group**: LSRED
- **version number**: 1.0
- **FORTRAN name**: PELCF

File input/output:

none

Method:

**Literature**

/1/ Varga A.

Coprime Factors Model Reduction Method based on Square-Root Balancing-Free Techniques,  

Remarks:

none

Life cycle:

1993 Aug A. Varga DLR FF-DF: coded

Packages required:

none

Libraries required:
RASP, LAPACK, BLAS (1,2,3)

Example:

Given the continuous-time unstable system example (A,B,C,D) used by the ASD module (D = 0), the matrices of the left coprime factorization computed with ALPHA = 0, STABDEG = -1, TOL = 1.0D-10 are:

\[
\begin{pmatrix}
-1.0000 & -0.4465 & 4.8212 & 0.2260 & 0.0062 & -0.1813 & -0.0895 \\
0.0526 & -1.0000 & -0.3364 & -0.0166 & 0.4199 & -0.2408 & -1.7274 \\
0 & 0 & -3.5957 & -3.5463 & -0.0163 & 0.0175 & 0.0592 \\
\end{pmatrix}
\]

\[
\begin{pmatrix}
0 & 0 & 0 & -12.4245 & 0.0004 & -0.0344 & -0.0180 \\
0 & 0 & 0 & 0 & -13.1627 & 1.9835 & 3.6182 \\
0 & 0 & 0 & 0 & 0 & -1.4178 & 5.6218 \\
0 & 0 & 0 & 0 & 0 & -0.8374 & -1.4178 \\
\end{pmatrix}
\]

\[
\begin{pmatrix}
0.0631 & 0.5122 & 0.0155 & 0.0753 & 1.1676 \\
1.1544 & 0.0159 & 0.2623 & 1.1297 & 0.0763 \\
0.0476 & 0.3029 & 0 & 0 & 0 \\
\end{pmatrix}
\]

\[
\begin{pmatrix}
0.0130 & 12.4858 & 0 & 0 & 0 \\
11.7198 & 0.0038 & 0 & 0 & 0 \\
-2.8173 & 0.0308 & 0 & 0 & 0 \\
3.1018 & -0.0009 & 0 & 0 & 0 \\
\end{pmatrix}
\]

\[
\begin{pmatrix}
-0.0132 & -0.2238 & 0.0079 & -0.0026 & -0.1279 & 0.8797 & 0.3994 \\
-0.0643 & -0.9639 & 0.0040 & -0.0009 & -0.0305 & -0.2562 & 0.0122 \\
-0.9962 & 0.0660 & 0.0377 & -0.0419 & 0 & 0.0022 & -0.0017 \\
\end{pmatrix}
\]

\[
\begin{pmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 \\
\end{pmatrix}
\]
MONITORMODULE

RCF - Right coprime factorization of transfer-function matrices

Monitormodule purpose:
Given an n-th order state-space model (A,B,C,D) corresponding to
a pxm transfer-function matrix G, a state-space representation
(AQR, BQR, CQR, DQR) of the augmented transfer-function matrix
\[
\begin{bmatrix}
Q \\
R
\end{bmatrix}
\]
is computed defining a right coprime factorization (RCF) \( G = Q \cdot R \). The state matrix AQR can be determined such that its eigenvalues
are inside a specified stability region of the complex plane.
The computation of the factorization is based on a pole assignment
method described in /1/.

Optional output on database is:
- right coprime factorization (in augmented form) (def. output)

The q-th order state-space representations of the factors Q and R are
constructed in the forms
\[
Q = (Z'*(A+B*F)*Z, Z'*B, (C+D*F)*Z, D)
\]
and
\[
R = (Z'*(A+B*F)*Z, Z'*B, F*Z, I)
\]

where Z is an n x q matrix with orthonormal columns and F is an m x n
state-feedback matrix assigning the eigenvalues of A in a specified
stability region. If the given state-space representation is
stabilizable, then q = n. If the given state-space representation is
not stabilizable, then the unstabilizable part of the original system
is automatically deflated and the resulting q is less than r.
The matrices AQR, BQR, CQR and DQR of augmented RCF are computed as:
\[
\begin{bmatrix}
C*Z+D*F*Z \\
D
\end{bmatrix}
\]

Options:
Next case

Database Structure:
input:
linear system, state space
output:
<Result S-Name>
  _LS  linear system, state space

Dialog:
------- requested input
LS RS-NAME:
-> linear system name
or carriage return to end the module
INPUT SIGNALS - U,V,ALLIN ?
-> select inputs, default is U
OUTPUT SIGNALS - YM,YA,ALLOUT ?
-> select outputs, default is YM

----- end

1. Level commands:
---------------------
RCF >C>
PARAM : set parameter values for model reduction
POUT : set parameter value for output on database
COMPUTE : compute the factorization
If the outputing parameter is set then:
----- requested input
<Result S-NAME> ( <Default S-NAME> ):
-> If a blank is input, the <Default S-Name> is used.
   Otherwise the user must input a new S-Name under
   which the result will be stored.
The <Default S-name> is the S-name of the input
linear system appended by _RCF.index, where index
is the next free value. If one does not use the
<Default S-Name>, there could already exist other
objects under the chosen < Result S-Name>. Then
the action taken depends on a status variable:
/SDEVERIF is set: It is asked whether all objects
can be or not deleted. If objects
should be not deleted, then a new
<Result S-Name> is asked for.
/SDENOVER is set: All objects under the given
structure are deleted.

   If the augmented factorization is stored:
----- requested input
   Text for dataobject LS:
----- end
----- end
END : terminate module

2. Level Commands:
---------------------
RCF_PARAM >P>
ALPHA : boundary of stability region (DOUBLE def: stpar)
Specifies the boundary of the stability region for
the eigenvalues of A:
- for continuous-time systems ALPHA represents the maximum
  admissible value for the real parts of the eigenvalues.
  Default value of stpar is 0.
- for discrete-time systems ALPHA represents the maximum
  admissible value for the moduli of the eigenvalues.
  Default value of stpar is 1.0.
The eigenvalues lying inside the stability region will be
not modified.

**STABDEG**: desired stability degree  
(DOUBLE def: sdpar)

The eigenvalues of A, lying outside the stability region defined by ALPHA, will be moved to locations corresponding to a stability degree STABDEG, namely:
- for a continuous-time system, the modified eigenvalues will have real parts equal to STABDEG and unmodified imaginary parts. Default value of sdpar is -0.2.
- for a discrete-time system, the modified eigenvalues will have moduli equal to STABDEG.

Default value of sdpar is 0.8.

**TOL**: tolerance for controllability tests  
(DOUBLE def: tolpar)

Specifies the absolute tolerance level below which the elements of B are considered zero.

Default value of TOL is tolpar = n*EPS*NORM(B), where NORM(B) denotes the 1-norm of B and EPS is the machine precision.

RCF_POUT >P>

**RCF**:  
(INTEGER def: 1)

0 : do not store the augmented RCF
1 : store the augmented RCF

Monitor module initialization:

- **module name**: RCF
- **module group**: LSRED
- **version number**: 1.0
- **FORTRAN name**: PERCF

File input/output:

- **none**

Method:

- **Literature**

  /1/ Varga A.

  Coprime Factors Model Reduction Based on Accuracy Enhancing Techniques,


Remarks:

- **none**

Life cycle:

1993 Aug  A. Varga DLR FF-DF: coded

Packages required:

- **none**

Libraries required:

- RASP, LAPACK, BLAS (1,2,3)
Example:
Given the continuous-time unstable system example \((A,B,C,D)\) used by the ASD module \((D = 0)\), the matrices of the right coprime factorization computed with \(\text{ALPHA} = 0, \text{STABDEG} = -1, \text{TUL} = 1.0D-10\) are:

\[
\begin{bmatrix}
-1.4178 & -5.1682 & 3.2450 & .0241 & .2232 & 5.4114 & .0201 \\
.9109 & -1.4178 & -2.1262 & -.1188 & -.0867 & .5383 & .2376 \\
0 & 0 & -13.1627 & -.0011 & -.0646 & -6.8045 & .0315 \\
\end{bmatrix}
\]

\[
\text{AQR} = \begin{bmatrix}
0 & 0 & 0 & -12.4245 & 3.3373 & -33.5354 & -6.4031 \\
0 & 0 & 0 & 0 & -3.5957 & 10.6960 & 6.6253 \\
0 & 0 & 0 & 0 & 0 & -1.0000 & .0030 \\
0 & 0 & 0 & 0 & 0 & -7.7653 & -1.0000 \\
\end{bmatrix}
\]

\[
\begin{bmatrix}
5.0302 & -.0063 \\
.7078 & -.0409 \\
-11.3663 & .0051 \\
\end{bmatrix}
\]

\[
\text{BQR} = \begin{bmatrix}
-.0375 & -11.6309 \\
-.1740 & 3.7681 \\
-.1040 & -.1956 \\
-.0472 & -.25948 \\
\end{bmatrix}
\]

\[
\begin{bmatrix}
-.8659 & .2787 & -.3432 & -.0007 & -.0019 & -.2335 & .0152 \\
.0797 & -.3951 & .0976 & .0045 & .0295 & -.9043 & .0968 \\
-.0165 & -.0645 & .0097 & -.1341 & -.8080 & -.0599 & -.5666 \\
0 & 0 & 0 & 0 & 0 & .4915 & -.0036 \\
\end{bmatrix}
\]

\[
\text{CQR} = \begin{bmatrix}
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
\end{bmatrix}
\]

\[
\text{DQR} = \begin{bmatrix}
0 & 0 \\
1 & 0 \\
0 & 1 \\
\end{bmatrix}
\]
MONITORMODULE

LCFID - Left coprime factorization with inner denominator

Monitor module purpose:

Given an n-th order state-space model \((A,B,C,D)\) corresponding to a \(p \times m\) transfer-function matrix \(G\), a state-space representation \((AQ, BR, CR, DR)\) of the augmented transfer-function matrix \(\begin{bmatrix} Q & R \end{bmatrix}\) is computed defining a left coprime factorization with inner \(-1\).

Denominator \((LCFID)\) \(G = R \ast Q\).

\(G\) must not have poles on the imaginary axis for a continuous-time system or on the unit circle for a discrete-time system.

The computation of the factorization is based on the method described in /1/.

Optional outputs on database are:

- left coprime factorization (in augmented form) \(\) (def. output)

The \(q\)-th order state-space representations of the factors \(Q\) and \(R\) are constructed in the forms

\[
Q = (Z' \ast (A + H \ast C) \ast Z, Z' \ast (B + H \ast D), V \ast C \ast Z, V \ast D)
\]

and

\[
R = (Z' \ast (A + H \ast B) \ast Z, Z' \ast H, V \ast C \ast Z, V)
\]

where \(Z\) is an \(n \times q\) matrix with orthonormal columns, \(H\) is an \(n \times p\) output injection matrix reflecting the unstable eigenvalues of \(A\) in the stability region, and \(V\) is a \(p \times p\) gain matrix. The resulting \(R\) is generally non-minimal and co-inner, that is \(R(s) \ast R'(\ast s) = I\) in the continuous-time case or \(R(z) \ast R'(1/z) = I\) in the discrete-time case.

If the given state-space representation is detectable, the order \(q\) of the resulting state-space representation of \(\begin{bmatrix} Q & R \end{bmatrix}\) is equal to \(n\). If the given state-space representation is not detectable, the undetectable part of the original system is automatically deflated and the resulting \(q\) is less than \(n\). The matrices \(AQR, BQR, CQR\) and \(DQR\) of augmented LCFID are computed as:

\[
AQR = Z' \ast (A + H \ast C) \ast Z, \quad BQR = (Z' \ast B + Z' \ast H \ast D, Z' \ast H),
\]

\[
CQR = V \ast C \ast Z, \quad DQR = (V \ast D \quad V).
\]

Options:

Next case

Database Structure:

input:
linear system, state space
output:
<Result S=Name>
\_LS linear system, state space
Dialog:

------ requested input

LS RS-NAME:
  -> linear system name
  or carriage return to end the module

INPUT SIGNALS - U,V,ALLIN ?
  -> select inputs, default is U

OUTPUT SIGNALS - YM,YA,ALLOUT ?
  -> select outputs, default is YM

------ end

1. Level commands:

-------------------

LCFID >C>

PARAM : set parameter values for model reduction
POUT : set parameter values for output on database
COMPUTE : compute the factorization

If the outputing parameter is set then:

------ requested input

<Result S-NAME> ( <Default S-NAME> ):
  -> If a blank is input, the <Default S-Name> is used.
     Otherwise the user must input a new S-Name under
     which the result will be stored according to the
     parameters selected through POUT.
     The <Default S-name> is the S-name of the input
     linear system appended by _LCFID.index, where
     index is the next free value. If one does not use
     the <Default S-Name>, there could already exist
     other objects under the chosen < Result S-Name>.
     Then the action taken depends on a status variable:

/SDEVERIF is set: It is asked whether all objects
  can be or not deleted. If objects
  should be not deleted, then a new
<Result S-Name> is asked for.

/SDENOVER is set: All objects under the given
  structure are deleted.

If the augmented factorization is stored:

------ requested input

Text for dataobject LS:

------ end

------ end

END : terminate module

2. Level Commands:

---------------

LCFID_PARAM >P>

TOL : tolerance for observability tests (DOUBLE def: tolpar)
  Specifies the absolute tolerance level below which the
  elements of C are considered zero.
  Default value of TOL is tolpar = n*EPS*NORM(C),
  where NORM(C) denotes the infinity-norm of C and
EPS is the machine precision.

LCFID_POUT >P>
LCFID  : 0/1  (INTEGER def: 1)
  0 : do not store the augmented LCFID
  1 : store the augmented LCFID

Monitor module initialization:
module name  : LCFID
module group  : LSRED
version number  : 1.0
FORTRAN name  : PELCFI

File input/output:
  none

Method:

Literature
/1/ Varga A.
  A Schur method for computing coprime factorizations
  with inner denominators and applications in model reduction,

Remarks:
  none

Life cycle:
  1993 Aug  A. Varga DLR FF-DF: coded

Packages required:
  none

Libraries required:
  RASP, LAPACK, BLAS (1,2,3)

Example:
Given the continuous-time unstable system example (A,B,C,D) used by
the ASD module (D = 0), the matrices of the left coprime factorization
computed with TOL = 1.0D-10 are:

\[
\begin{pmatrix}
-0.1605 & -0.4489 & 4.2621 & 0.2229 & -0.2394 & -0.0491 & 0.8740 \\
0.0523 & -0.1605 & 2.2250 & 0.1217 & 0.4166 & -0.2518 & -1.6140 \\
0 & 0 & -3.5967 & -3.5463 & -0.0163 & 0.0175 & 0.0592 \\
\end{pmatrix}
\]

\[
AQR = \begin{pmatrix}
0 & 0 & 0 & 0 & -12.4245 & 0.0004 & -0.0344 & -0.0180 \\
0 & 0 & 0 & 0 & -13.1627 & 1.9835 & 3.6182 \\
0 & 0 & 0 & 0 & 0 & -1.4178 & 5.6218 \\
0 & 0 & 0 & 0 & 0 & -0.8374 & -1.4178 \\
\end{pmatrix}
\]

\[
\begin{pmatrix}
-0.5523 & -0.4443 & 0.0306 & -0.1281 & 0.4984 \\
1.0157 & -0.2554 & 0.0158 & 0.0692 & 0.1688 \\
-0.0476 & 0.3029 & 0 & 0 & 0 \\
\end{pmatrix}
\]
\[
BQR = \begin{pmatrix}
0.0130 & 12.4858 & 0 & 0 & 0 \\
-11.7198 & 0.0038 & 0 & 0 & 0 \\
-2.8173 & 0.0308 & 0 & 0 & 0 \\
3.1018 & -0.0009 & 0 & 0 & 0 \\
\end{pmatrix}
\]
\[
CQR = \begin{pmatrix}
0.1063 & -0.1975 & 0.0079 & -0.0026 & -0.1279 & 0.8797 & 0.3994 \\
0.4513 & -0.8541 & 0.0040 & -0.0009 & -0.0305 & -0.2562 & 0.0122 \\
-0.8826 & -0.4668 & 0.0377 & -0.0419 & 0 & 0.0022 & -0.0017 \\
\end{pmatrix}
\]
\[
DQR = \begin{pmatrix}
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
\end{pmatrix}
\]

The minimal state-space realization of the inner denominator $R$ has the following matrices:

\[
AR = \begin{pmatrix}
-0.1605 & -0.4489 \\
0.0523 & -1.605 \\
\end{pmatrix}
\]
\[
BR = \begin{pmatrix}
-0.0306 & -0.1281 & 0.4984 \\
0.0158 & 0.0692 & 0.1688 \\
\end{pmatrix}
\]
\[
CR = \begin{pmatrix}
0.1063 & -0.1975 \\
0.4513 & -0.8541 \\
-0.8826 & -0.4668 \\
\end{pmatrix}
\]
\[
DR = \begin{pmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1 \\
\end{pmatrix}
\]
MONITORMODULE

RCFID - Right coprime factorization with inner denominator

Monitor module purpose:

Given an n-th order state-space model \((A, B, C, D)\) corresponding to a \(p \times m\) transfer-function matrix \(G\), a state-space representation \((AQR, BQR, CQR, DQR)\) of the augmented transfer-function matrix
\[
\begin{pmatrix}
Q \\
R
\end{pmatrix}
\]
is computed defining a right coprime factorization with inner 
-1
denominator \((RCFID) G = Q \cdot R\).

\(G\) must not have poles on the imaginary axis for a continuous-time system or on the unit circle for a discrete-time system.
The computation of the factorization is based on the method described in 
/1/.

Optional outputs on database are:
- right coprime factorization (in augmented form) (def. output)

The \(q\)-th order state-space representations of the factors \(Q\) and \(R\) are constructed in the forms
\[
Q = (Z' \cdot (A+B \cdot F) \cdot Z, Z' \cdot B+V, (C+D \cdot F) \cdot Z, D \cdot V)
\]
and
\[
R = (Z' \cdot (A+B \cdot F) \cdot Z, Z' \cdot B+V, F \cdot Z, V)
\]

where \(Z\) is an \(n \times q\) matrix with orthonormal columns, \(F\) is an \(m \times n\) state-feedback matrix reflecting the unstable eigenvalues of \(A\) in the stability region and \(V\) is a \(p \times p\) gain matrix. The resulting \(R\) is inner, that is \(R' \cdot (-s) \cdot R(s) = I\) in the continuous-time case or \(R' \cdot (1/z) \cdot R(z) = I\) in the discrete-time case.

If the given state-space representation is stabilizable, the order \(q\) of the resulting state-space representation
\[
\begin{pmatrix}
Q \\
R
\end{pmatrix}
\]
is equal to \(n\). If the given state-space representation is not stabilizable, the unstabilizable part of the original system is automatically deflated and the resulting \(q\) is less than \(n\).
The matrices \(AQR, BQR, CQR\) and \(DQR\) of augmented RCFID are computed as:
\[
AQR = Z' \cdot (A+B \cdot F) \cdot Z, BQR = Z' \cdot B+V, CQR = (F \cdot Z), DQR = (V).
\]

Options:
Next case

Database Structure:
input:
linear system, state space
output:
<Result S-Name>
   _LS    linear system, state space

Dialog:
----- requested input
LS RS-NAME:
   -> linear system name
    or carriage return to end the module
INPUT SIGNALS - U,V,ALLIN ?
   -> select inputs, default is U
OUTPUT SIGNALS - YM,YA,ALLOUT ?
   -> select outputs, default is YM
----- end

1. Level commands:
------------------
RCFID >C>
PARAM    : set parameter values for model reduction
POUT     : set parameter values for output on database
COMPUTE  : compute the factorization
    If the outputting parameter is set then:
       ----- requested input
       <Result S-NAME> ( <Default S-NAME> ):
       -> If a blank is input, the <Default S-Name> is used.
          Otherwise the user must input a new S-Name under
          which the result will be stored according to the
          parameters selected through POUT.
       The <Default S-Name> is the S-name of the input
      linear system appended by _RCFID.index, where
       index is the next free value. If one does not use
       the <Default S-Name>, there could already exist
       other objects under the chosen < Result S-Name >.
       Then the action taken depends on a status variable:
      /SDEVERIF is set: It is asked whether all objects
         can be or not deleted. If objects
         should be not deleted, then a new
         <Result S-Name> is asked for.
      /SDENOVER is set: All objects under the given
         structure are deleted.

       If the augmented factorization is stored:
       ------ requested input
       ----- Text for dataobject LS:
       ----- end
       ----- end
END      : terminate module

2. Level Commands:
------------------
RCFID_PARAM >P>
TOL      : tolerance for controllability tests    (DOUBLE def: tolpar)
       Specifies the absolute tolerance level below which the
elements of B are considered zero.
Default value of TOL is tolpar = n*EPS*NORM(B),
where NORM(B) denotes the 1-norm of B and EPS is the
machine precision.

RCFID_POUT >P>
  RCFI : 0/1 (INTEGER def: 1)
    0 : do not store the augmented RCFID
    1 : store the augmented RCFID

Monitor module initialization:
  module name : RCFID
  module group : LSRED
  version number : 1.0
  FORTRAN name : PERCFI

File input/output:
  none

Method:
Literature
  /1/ Varga A.
    A Schur method for computing coprime factorizations
    with inner denominators and applications in model reduction,

Remarks:
  none

Life cycle:
  1993 Aug A. Varga DLR FF-DF: coded

Packages required:
  none

Libraries required:
  RASP, LAPACK, BLAS (1,2,3)

Example:
Given the continuous-time unstable system example (A,B,C,D) used by
the ASD module (D = 0), the matrices of the right coprime
factorization computed with TOL = 1.0D-10 are:

\[
\begin{pmatrix}
-1.4178 & -5.1682 & 3.2450 & .0241 & .2232 & 4.1066 & -.2336 \\
.9109 & -1.4178 & -2.1262 & -.1188 & -.0867 & .4816 & .2196 \\
0 & 0 & -13.1627 & -.0011 & -.0646 & -3.8320 & .3429 \\
\end{pmatrix}
\]

AQR =
\[
\begin{pmatrix}
0 & 0 & 0 & -12.4245 & 3.3373 & -.2642 & -.6816 \\
0 & 0 & 0 & 0 & 0 & -3.5957 & .1871 & 5.4221 \\
0 & 0 & 0 & 0 & 0 & 0 & -.1605 & .0772 \\
0 & 0 & 0 & 0 & 0 & 0 & -.3040 & -.1605 \\
\end{pmatrix}
\]
$$\begin{bmatrix}
5.0302 & -0.0063 \\
0.7078 & -0.0409 \\
-11.3663 & 0.0051
\end{bmatrix}$$

$$\begin{bmatrix}
-0.0375 & -11.6309 \\
-0.1740 & 3.7681 \\
-1.1050 & -0.3215 \\
0.0066 & -2.5822
\end{bmatrix}$$

$$\begin{bmatrix}
-0.8659 & 0.2787 & -0.3432 & -0.0007 & -0.0019 & -0.2325 & 0.0265 \\
0.0797 & -0.3951 & -0.976 & 0.0045 & 0.0295 & -0.8985 & 0.1406
\end{bmatrix}$$

$$\begin{bmatrix}
-0.0165 & -0.0645 & 0.0097 & -0.1341 & -0.8080 & -0.0874 & -0.5630 \\
0 & 0 & 0 & 0 & 0 & 0.2288 & -0.0259
\end{bmatrix}$$

$$\begin{bmatrix}
0 & 0 \\
0 & 0
\end{bmatrix}$$

$$\begin{bmatrix}
0 & 0 \\
1 & 0 \\
0 & 1
\end{bmatrix}$$

The minimal state-space realization of the inner denominator $R$ has the following matrices:

$$\begin{bmatrix}
-0.1605 & 0.0772 \\
-0.3040 & -0.1605
\end{bmatrix}$$

$$\begin{bmatrix}
-1.1050 & -0.3215 \\
0.0066 & -2.5822
\end{bmatrix}$$

$$\begin{bmatrix}
0.2288 & -0.0259 \\
0.0070 & 0.1497
\end{bmatrix}$$

$$\begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix}$$
MONITOR MODULE

LCF2LS - Left coprime factorization to linear system transformation

Monitormodule purpose:
Computes the state-space representation of the system \( G = (A,B,C,D) \)
from the factors \( Q = (AQR,BQ,CQR,DQ) \) and \( R = (AQR, BR, CQR, DR) \) of
its left coprime factorization
\[ G = R \ast Q , \]

where \( G, Q \) and \( R \) are the corresponding transfer-function matrices.
The matrices of the state-space representations of \( Q \) and \( R \) should be
contained in a state-space representation of the augmented system
\( (Q R) \) in the form \((AQR, (EQ BR), CQR, (DQ DR))\) (as computed for
example by the LCF or LCFID modules).

LCF2LS is used as a post-processing module in the coprime factors
model reduction method /1/ in conjunction with the left coprime
factorizations modules LCF or LCFID and the model reduction modules
for stable systems.

Optional outputs on database are:
- left coprime factorization (in augmented form) (def. output)

The matrices of the state-space representation \( G = (A,B,C,D) \) are
computed by using the formulas:
\[
A = AQR - BR \ast DR \ast CQR, \quad B = BQ - BR \ast DR \ast DQ, \\
C = DR \ast CQR, \quad D = DR \ast DQ .
\]

Options:
Next case

Database Structure:
input:
linear system, state space
output:
\(<Result S-Name> \_LS \)
linear system, state space

Dialog:
-------- requested input
LS RS-NAME:
-> linear system name
or carriage return to end the module
INPUT SIGNALS - U,V,ALLIN ?
-> select inputs, default is U
OUTPUT SIGNALS - YM,YA,ALLOUT ?
-> select outputs, default is YM
1. Level commands:

LCF2LS >C>

POUT : set parameter values for output on database
COMPUTE : compute the factorization

If the outputing parameter is set then:

----- requested input

<Result S-NAME> ( <Default S-NAME> ) :

-> If a blank is input, the <Default S-NAME> is used.
Otherwise the user must input a new S-Name under which the result will be stored.
The <Default S-name> is the S-name of the input
linear system appended by _LCF2LS.index, where
index is the next free value. If one does not use
the <Default S-Name>, there could already exist
other objects under the chosen <Result S-Name>.
Then the action taken depends on a status variable:
/SDEVERIF is set: It is asked whether all objects
can be or not deleted. If objects
should be not deleted, then a new
<Result S-Name> is asked for.

/SDENOVER is set: All objects under the given
structure are deleted.

If the augmented factorization is stored:

----- requested input

Text for dataobject LS:

----- end

----- end

END : terminate module

2. Level Commands:

LCF2LS_POUT >P>

SYSTEM : 0/1

(INTEGER def: 1)

0 : do not store the computed state-space representation
1 : store the computed state-space representation

Monitor module initialization:

module name : LCF2LS
module group : LSRED
version number : 1.0
FORTTRAN name : PEBLCF

File input/output:

none

Method:

Literature

/1/ Varga A.
Coprime Factors Model Reduction Based on Accuracy Enhancing Techniques,

Remarks:
none

Life cycle:
1993 Aug A. Varga DLR FF-DF: coded

Packages required:
none

Libraries required:
RASP, LAPACK, BLAS (1,2,3)

Example:
none
MONITORMODULE

RCF2LS - Right coprime factorization to linear system transformation

Monitormodule purpose:
Computes the state-space representation of the system \( G = (A,B,C,D) \)
from the factors \( Q = (AQR,BQR,CQ,DQ) \) and \( R = (AQR,BQR,CR,DR) \)
of its right coprime factorization
\[
G = Q \ast R^{-1},
\]

where \( G, Q \) and \( R \) are the corresponding transfer-function matrices.
The matrices of the state-space representations of \( Q \) and \( R \) should be
contained in a state-space representation of the augmented system
\[
\begin{pmatrix}
Q \\
CQ \\
DR \\
R
\end{pmatrix}
\]
in the form \( (AQR, BQR, (CR), (DR)) \) (as computed for
example by the RCF or RCFID modules).

RCF2LS is used as a post-processing module in the coprime factors
model reduction method /1/ in conjunction with the right coprime
factorizations modules RCF or RCFID and the model reduction modules
for stable systems.

Optional output on database are:
- right coprime factorization (in augmented form) (def. output)

The matrices of the state-space representation \( G = (A,B,C,D) \) are
computed by using the formulas:
\[
\begin{align*}
A &= AQR - BQR \ast DR \ast CR, & B &= BQR \ast DR^{-1}, \\
C &= CQ - DQ \ast DR \ast CR, & D &= DQ \ast DR^{-1}.
\end{align*}
\]

Options:
Next case

Database Structure:
input:
linear system, state space
output:
<Result S-Name>
_LS linear system, state space

Dialog:
-------- requested input
LS RS-NAME:
-> linear system name
or carriage return to end the module
INPUT SIGNALS - U,V,ALLIN ?
-> select inputs, default is U
OUTPUT SIGNALS - YM,YA,ALLOUT ?
-> select outputs, default is YM

------ end

1. Level commands:
-------------------

RCF2LS >C>

POUT: set parameter value for output on database
COMPUTE: compute the factorization

If the outputting parameter is set then:

------ requested input

<Result S-NAME> ( <Default S-NAME> ):

-> If a blank is input, the <Default S-Name> is used.
   Otherwise the user must input a new S-Name under
   which the result will be stored.
   The <Default S-name> is the S-name of the input
   linear system appended by _RCF2LS.index, where
   index is the next free value. If one does not use
   the <Default S-Name>, there could already exist
   other objects under the chosen < Result S-Name>.

Then the action taken depends on a status variable:
/SDEVERIF is set: It is asked whether all objects
   can be or not deleted. If objects
   should be not deleted, then a new
   <Result S-Name> is asked for.

/SDENOVER is set: All objects under the given
   structure are deleted.

If the augmented factorization is stored:

------ requested input

Text for dataobject LS:

------ end

END: terminate module

2. Level Commands:
-------------------

RCF2LS_POUT >P>

SYSTEM: 0/1 (INTEGER def: 1)

0: do not store the computed state-space representation
1: store the computed state-space representation

Monitor module initialization:

module name: RCF2LS
module group: LSRED
version number: 1.0
FORTRAN name: PEBRCF

File input/output:

none

Method:

Literature
/1/ Varga A.
Coprime Factors Model Reduction Based on Accuracy
Enhancing Techniques,

Remarks:
none

Life cycle:
1993 Aug  A. Varga DLR FF-DF: coded

Packages required:
none

Libraries required:
RASP, LAPACK, BLAS (1,2,3)

Example:
none
MONITORMODULE
PFWMR - Projections for frequency-weighted model reduction

Monitoromodule purpose:
Constructs a state-space representation (AS,BS,CS,DS) of the
* -1    * -1    *
stable projection of either (W1) *G*(W2) or W1 *G*W2
from the state-space representations (A,B,C,D), (AW1,BW1,CW1,DW1)
and (AW2,BW2,CW2,DW2) of the transfer-function matrices G, W1 and W2,
respectively. G, W1, W2 should be stable, W1 and W2 should be
invertible and minimum-phase. In the discrete-time case W1 and W2
should have no poles in the origine.
*
Note. W denotes the conjugate of W given by either W'(-s) for a
continuous-time system or W'(1/z) for a discrete-time system.

Optional output on database is:
- computed system projection (def. output)

The matrices of the stable projection are computed using the
explicit formulas established in /1/.

Options:
Next case

Database Structure:
input:
linear system, state space
output:
<Result S-Name>
    _LS    linear system, state space

Dialog:
------ requested input
LS RS-NAME:
    -> linear system name
    or carriage return to end the module
    INPUT SIGNALS - U,V,ALLIN ?
    -> select inputs, default is U
    OUTPUT SIGNALS - YM,YA,ALLOUT ?
    -> select outputs, default is YM
------ end

1. Level commands:
---------------------
PFWMR >C>

WEIGHTS : set option parameters for weights
If the input weighting parameter is set then:
------ requested input
    LS RS-NAME:
linear system name
  or carriage return to end the module
INPUT SIGNALS - U,V,ALLIN ?
  -> select inputs, default is U
OUTPUT SIGNALS - YM,YA,ALLOUT ?
  -> select outputs, default is YM
----- end

If the output weighting parameter is set then:
----- requested input
LS R3-NAME:
  -> linear system name
  or carriage return to end the module
INPUT SIGNALS - U,V,ALLIN ?
  -> select inputs, default is U
OUTPUT SIGNALS - YM,YA,ALLOUT ?
  -> select outputs, default is YM
----- end

POUT : set parameter value for output on database
COMPUTE : compute the stable projection

If the outputting parameter is set then:
----- requested input
<Result S-NAME> ( <Default S-NAME> ):
  -> If a blank is input, the <Default S-Name> is used.
Otherwise the user must input a new S-Name under
which the result will be stored according to the
parameters selected through POUT.
The <Default S-name> is the S-name of the input
linear system appended by _PFWMR.index, where index
is the next free value. If one does not use the
<Default S-Name>, there could already exist other
objects under the chosen < Result S-Name >. Then
the action taken depends on a status variable:
/SDEVERIF is set: It is asked whether all objects
can be or not deleted. If objects
should be not deleted, then a new
<Result S-Name> is asked for.

/SDENOVER is set: All objects under the given
  structure are deleted.

If the resulting reduced system is stored:
----- requested input
  Text for dataobject LS:
----- end
----- end

END : terminate module

2. Level Commands:

------------------
PFWMR_WEIGHTS >>
OUTPUT : 0/1 (INTEGER def: 1)
  0 : output weight is not used
  1 : output weight is used
INPUT : 0/1  (INTEGER def: 0)
   0 : input weight is not used
   1 : input weight is used

INVERSE : 0/1  (INTEGER def: 1)
   0 : conjugated weights are used
   1 : inverses of conjugated weights are used

WITHD : 0/1  (INTEGER def: 0)
   0 : the feedthrough matrix D is not used in computing the stable projection
   1 : the feedthrough matrix D is used in computing the stable projection

PFWMR_POUT >P>
PRSYS : 0/1  (INTEGER def: 1)
   0 : do not store the computed projection
   1 : store the computed projection

Monitor module initialization:
module name : PFWMR
module group : LSRED
version number : 1.0
FORTRAN name : PFWMR

File input/output:
none

Method:

Literature
/1/ Varga A.
   Explicit formulas for an efficient implementation of the frequency-weighting model reduction approach,

Remarks:
- No explicit singularity checks are performed in order to verify the conditions to be fulfilled by the matrices of the weights W1 and W2. Error messages are issued only if exact singularity is detected. If the respective matrices are nearly singular, the computed results may be inaccurate.

Life cycle:
1993 Aug A. Varga DLR FF-DF: coded

Packages required:
none

Libraries required:
RASP, LAPACK, BLAS (1,2,3)
Example:

Given the continuous-time system \((A,B,C,D)\) with the following matrices:

\[
\begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0
\end{pmatrix} \quad \begin{pmatrix} 1 \end{pmatrix} \\
A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad B = \begin{pmatrix} 0 \end{pmatrix}
\]

\[C = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad D = 0\]

and the output frequency-weighting \(W_1\) with the matrices:

\[
AW_1 = \begin{pmatrix} -2 & -1 \end{pmatrix} \quad BW_1 = \begin{pmatrix} 1 \end{pmatrix} \quad CW_1 = \begin{pmatrix} -1.8 & 0 \end{pmatrix} \quad DW_1 = 1,
\]

\[\quad \star -1\]

the matrices of the stable projection of \((W_1) \star G\) computed with the parameter setting OUTPUT = 1, INPUT = 0, INVERSE = 1, WITHD = 0 are:

\[
\begin{pmatrix}
-.2588 & 1.1498 & -.4160 & .4591 & 1.7533 & -7.7212 \\
-1.8114 & -.2588 & -.0281 & .3703 & -.5606 & 1.1293 \\
.0 & .0 & -.7071 & -.5287 & -.8349 & 2.0301 \\
0 & 0 & .9457 & -.7071 & -2.6789 & 9.9802 \\
0 & 0 & 0 & 0 & -.9659 & 6.9494 \\
0 & 0 & 0 & 0 & -.0096 & -.9659
\end{pmatrix}
\]

\[
\begin{pmatrix}
-.5286 \\
.0829 \\
.1458 \\
.6908 \\
.4464 \\
-.1263
\end{pmatrix}
\]

\[
CS = \begin{pmatrix} 1.8545 & .2907 & -.9917 & 1.5564 & 2.0202 & 8.0920 \end{pmatrix}
\]

\[DS = 0\]
MONITORMODULE
HANKNORM - Hankel-norm of a transfer function matrix

Monitormodule purpose:
Computes the Hankel-norm and the Hankel-singular values of the stable
projection of the transfer-function matrix of a linear state-space
model (A,B,C,D).

Optionally displayed results:
- Computed Hankel norm (def. display)
- Computed Hankel singular values (def. display)
- Order of an approximate reduced model
  corresponding to a specified absolute error (def. display)
- Error bound for the maximum absolute
  approximation error (def. display)

Optional outputs on database are:
- Computed Hankel norm (def. output)
- Computed Hankel singular values (def. no output)
- Order of an approximate reduced model
  corresponding to a specified absolute error (def. no output)
- Error bound for the maximum absolute
  approximation error (def. no output)

If the transfer-function matrix G is unstable, then the stable
projection G1 of G is first computed from the additive
stable/unstable decomposition of G

G = G1 + G2

where G1 is the stable projection and G2 is defined as G2 = G - G1.
If G is stable, then G1 = G and G2 = 0.
The computation of the stable projection is based on the
algorithm presented in /1/.

Let (A1,B1,C1) be the state-space representation of G1. Then,
the Hankel-norm of G is computed as the maximum Hankel singular
value of the system (A1,B1,C1,0). The computation of the Hankel
singular values is performed by using the square-root method
of /2/.

Options:
Next case

Database Structure:
input:
linear system, state space
output:
<Result S-Name>
  _HANKNORM  value of Hankel-norm
Dialog:

-------- requested input
LS RS-NAME:
   -> linear system name
      or carriage return to end the module
INPUT SIGNALS - U,V,ALLIN ?
   -> select inputs, default is U
OUTPUT SIGNALS - YM,YA,ALLOUT ?
   -> select outputs, default is YM
-------- end

1. Level commands:

-------------
HANKNORM >C>
PARAM : set parameter value for order estimation
PLIST : set parameter values for listing of results
POUT : set parameter values for output on database
COMPUTE : compute Hankel norm and Hankel-singular values
If one of outputting parameters is set then:
-------- requested input
<Result S-NAME> ( <Default S-NAME> ):
   -> If a blank is input, the <Default S-Name> is used.
      Otherwise the user must input a new S-Name under
      which the result will be stored according to the
      parameters selected through POUT.
      The <Default S-name> is the S-name of the input
      linear system appended by _HANKNORM.index, where
      index is the next free value. If one does not use
      the <Default S-Name>, there could already exist
      other objects under the chosen < Result S-Name >.
      Then the action taken depends on a status variable:
      /SDEVERIF is set: It is asked whether all objects
      can be or not deleted. If objects
      should be not deleted, then a new
      <Result S-Name> is asked for.
      /SDENOVER is set: All objects under the given
      structure are deleted.
-------- end
END : terminate module

2. Level Commands:

-------------
HANKNORM_PARAM >P>
ABERR : admissible absolute approximation error (DOUBLE def: 0.0)

HANKNORM_PLIST >P>
HANKNORM : 0/1 (INTEGER def: 1)
0 : do not list the value of the Hankel norm
1 : list the value of the Hankel norm

HSV : 0/1 (INTEGER def: 1)
0 : do not list the Hankel singular values
1 : list the Hankel singular values

ORDER : 0/1 (INTEGER def: 1)
0 : do not list the determined order
1 : list the determined order

MAXRERR : 0/1 (INTEGER def: 1)
0 : do not list the maximum absolute approximation error
1 : list the maximum absolute approximation error

HANKNORM_POUT >>P>

HANKNORM : 0/1 (INTEGER def: 1)
0 : do not store the value of the Hankel norm
1 : store the value of the Hankel norm

HSV : 0/1 (INTEGER def: 0)
0 : do not store the Hankel singular values
1 : store the Hankel singular values

ORDER : 0/1 (INTEGER def: 0)
0 : do not store the determined order
1 : store the determined order

MAXRERR : 0/1 (INTEGER def: 0)
0 : do not store the maximum absolute approximation error
1 : store the maximum absolute approximation error

Monitormodule initialization:

module name : HANKNORM
module group : LSRED
version number : 1.0
FORTRAN name : PEHNRM

File input/output:
none

Method:

Literature
/1/ Safonov, M.G., Jonckheere, E.A., Verma, M. and Limebeer, D.J.
Synthesis of positive real multivariable feedback systems,
/2/ Tombs M.S. and Postlethwaite I.
Truncated balanced realization of stable, non-minimal
state-space systems, Int. J. Control, Vol. 46,

Remarks:
- The Hankel-norm is only a seminorm when the transfer-function
matrix is not strictly proper or not causal.

Life cycle:
1993 Aug A. Varga DLR FF-DF: coded
Packages required:
none

Libraries required:
RASP, LAPACK, BLAS (1,2,3)

Example:
Given the continuous-time unstable system \((A,B,C,D)\) used by the module ASD, (the stable projection has order \(s = 5\)), the computed Hankel-norm and Hankel-singular values are:

- the Hankel-norm of the stable projection:

\[
\text{HANORM}(G) = 1.8198
\]

- the Hankel-singular values of the stable projection:

\[
1.8198 \quad .8047 \quad .7382 \quad .0242 \quad .0238
\]
MONITORMODULE
L2NORM - L2-norm of a transfer function matrix

Monitormodule purpose:
Computes the L2- or the l2-norm of the transfer-function matrix of a linear state-space model (A,B,C,D). For a continuous-time system, the feedthrough matrix D is assumed to be null.

Optionally displayed result:
- Computed L2- or 12-norm (def. display)

Optional output on database:
- Computed L2- or 12-norm (def. output)

Options:
Next case

Database Structure:
input:
linear system, state space
output:
<Result S-Name>
_L2NORM value of L2- or 12-norm

Dialog:
------- requested input
LS RS-NAME:
  -> linear system name
  or carriage return to end the module
INPUT SIGNALS - U,V,ALLIN ?
  -> select inputs, default is U
OUTPUT SIGNALS - YM,YA,ALLOUT ?
  -> select outputs, default is YM
------- end

1. Level commands:
-------------------
L2NORM >C>
PARAM : set parameter value for norm computation
PLIST : set parameter values for listing of results
POUT : set parameter values for output on database
GLOBAL : output the L2- or 12-norm as a global variable
  ------- requested input
  Global variable name?
  -> name for the global variable
-------------------
COMPUTE : compute the L2- or 12-norm
If the outputting parameter is set then:
  ------- requested input
<Result S-NAME> ( <Default S-NAME> ):
If a blank is input, the <Default S-Name> is used. Otherwise the user must input a new S-Name under which the result will be stored according to the parameters selected through POUT. The <Default S-name> is the S-name of the input linear system appended by _L2NORM.index, where index is the next free value. If one does not use the <Default S-Name>, there could already exist other objects under the chosen <Result S-Name>. Then the action taken depends on a status variable: 
/SDEVERIF is set: It is asked whether all objects can be or not deleted. If objects should be not deleted, then a new <Result S-Name> is asked for. 
/SDENOVER is set: All objects under the given structure are deleted.

------- end
END : terminate module

2. Level Commands:
-----------------

L2NORM_PARAM >P>
TOL : tolerance for controllability tests (DOUBLE def: tolpar)
     Specifies the absolute tolerance level below which the elements of B are considered zero.
     Default value of TOL is tolpar = n*EPS*NORM(B), where NORM(B) denotes the 1-norm of B and EPS is the machine precision.

L2NORM_PLIST >P>
L2NORM : 0/1 (INTEGER def: 1)
   0 : do not list the value of the L2- or 12-norm
   1 : list the value of the L2- or 12-norm

L2NORM_POUT >P>
L2NORM : 0/1 (INTEGER def: 1)
   0 : do not store the value of the L2- or 12-norm
   1 : store the value of the L2- or 12-norm

Monitor module initialization:
module name : L2NORM
module group : LSRED
version number : 1.0
FORTRAN name : PEL2NM

File input/output:
none

Method:
The module is based on the algorithms proposed in /1/ and /2/.
If the given transfer-function matrix \( G \) is unstable, then a right coprime factorization with inner denominator of \( G \) is first computed

\[
G = Q^{-1} R
\]

where \( Q \) and \( R \) are stable transfer-function matrices and \( R \) is inner. If \( G \) is stable, then \( Q = G \) and \( R = I \).

Let \((AQ,BQ,CQ,DQ)\) be the resulting state-space representation of \( Q \).

For a continuous-time system the L2-norm of \( G \) is computed as

\[
\text{NORM2}(G) = \text{NORM2}(Q) = \text{SQRT}(\text{TRACE}(BQ'\times X \times BQ))
\]

where \( X \) satisfies the continuous-time Lyapunov equation

\[
AQ'\times X + X \times AQ + CQ'\times CQ = 0.
\]

For a discrete-time system the L2-norm of \( G \) is computed as

\[
\text{NORM2}(G) = \text{NORM2}(Q) = \text{SQRT}(\text{TRACE}(BQ'\times X \times BQ + DQ' \times DQ))
\]

where \( X \) satisfies the discrete-time Lyapunov equation

\[
AQ'\times X \times AQ - X + CQ'\times CQ = 0.
\]

Literature
/1/  Varga A.,
On computing 2-norms of transfer-function matrices,

/2/  Varga A.
A Schur method for computing coprime factorizations
with inner denominators and applications in model reduction,

Remarks:
none

Life cycle:
1993 Aug  A. Varga DLR FF-DF: coded

Packages required:
none

Libraries required:
RASP, LAPACK, BLAS (1,2,3)

Example:
For the stable continuous-time system \((A,B,C,D)\) used by the module BTA the computed L2-norm is 3.8474.
For the unstable continuous-time system \((A, B, C, D)\) used by the module ASD the computed L2-norm is 7.9395.
MONITORMODULE

LINFNORM - L-infinity norm of a transfer-function matrix

Monitormodule purpose:
Computes the L-infinity norm of the transfer-function matrix G of a linear state-space model (A,B,C,D) by using the bisection method proposed in /1/ or /2/ with optional extrapolation.

Optionally displayed results:
- Computed L-infinity norm (def. display)
- Frequency where the L-infinity norm is attained (def. display)
- Statistical information on: relative and absolute accuracy attained; number of performed iterations (def. display)

Optional outputs on database are:
- Computed L-infinity norm (def. output)
- Frequency where the L-infinity norm is attained (def. output)

Options:
Next case

Database Structure:
input:
linear system, state space
output:
<Result S-Name>
  _LINFNORM value of L-infinity norm
  _FREQ frequency where the L-infinity norm is attained

Dialog:
-------- requested input
LS RS-NAME:
  -> linear system name
    or carriage return to end the module
INPUT SIGNALS - U,V,ALLIN ?
  -> select inputs, default is U
OUTPUT SIGNALS - YM,YA,ALLOUT ?
  -> select outputs, default is YM
-------- end

1. Level commands:
-------------------

LINFNORM >>

  PARAM : set options and accuracy parameter for norm computation
  PLIST : set parameter values for listing of results
  POUT : set parameter values for output on database
  GLOBAL : output the L-infinity norm as a global variable
    ---- requested input
    Global variable name?
      -> name for the global variable
--------------

**COMPUTE** : compute the L-infinity norm

If one of outputting parameters is set then:

----- requested input

<Result S-NAME> ( <Default S-NAME> ):

-> If a blank is input, the <Default S-Name> is used. Otherwise the user must input a new S-Name under which the result will be stored according to the parameters selected through POUT. The <Default S-name> is the S-name of the input linear system appended by _LINFNORM.index, where index is the next free value. If one does not use the <Default S-Name>, there could already exist other objects under the chosen <Result S-Name>. Then the action taken depends on a status variable:

/SDEVERIF is set: It is asked whether all objects can be or not deleted. If objects should be not deleted, then a new <Result S-Name> is asked for.

/SDENOVER is set: All objects under the given structure are deleted.

----- end
END : terminate module

2. Level Commands:
---------------

**LINFNORM_PARAM >P>**

**EXTRAPOL** : 0/1
0 : use bisection without extrapolation
1 : use bisection with extrapolation

**FACTOR** : 0/1
0 : apply bisection on the original system
1 : apply bisection on the denominator factor of a left coprime factorization with inner denominator of G (necessary if the original system is unstable)

**ACCUACY** : desired relative accuracy
(DOUBLE def: 1.0D-5)

**LINFNORM_PLIST >P>**

**LINFNORM** : 0/1
0 : do not list the value of the L-infinity norm
1 : list the value of the L-infinity norm

**FREQ.** : 0/1
0 : do not list the value of frequency at L-infinity norm
1 : list the value of frequency at L-infinity norm

**STATINF** : 0/1
0 : do not list statistical information
1 : list statistical information on: relative and absolute accuracy attained; number of performed iterations.

**LINFNORM_POUT >P>**

**LINFNORM** : 0/1
(DOUBLE def: 1)
0 : do not store the L-infinity norm
1 : store the L-infinity norm

FREQ. : 0/1  (INTEGER def: 1)
0 : do not store the value of frequency at L-infinity norm
1 : store the value of frequency at L-infinity norm

Monitor module initialization:
module name : LINFORM
module group : LSRED
version number : 1.0
FORTRAN name : PELINM

File input/output:
none

Method:
If the given transfer-function matrix G is unstable, then
a left coprime factorization with inner denominator of G
is first computed

\[-1\]

\[G = R \times Q\]

where Q and R are stable transfer-function matrices and R is
inner. If G is stable, then Q = G and R = I.
Thus, the L-infinity norm of G equals H-infinity norm of Q.
The factorization is performed only if FACTOR is set to 1. The
algorithm to compute the factorization was proposed in /3/.

The calculation of the H-infinity norm of Q requires repeated checking
for pure imaginary eigenvalues of a Hamiltonian matrix H(q) depending
on a real parameter q. Successive values of q approach the H-infinity
norm of Q and lie always between a lower and upper bounds on the
H-infinity norm. The bounds are updated after each iteration
by bisection. The iteration stops when

\[(\text{the current upper bound}) \leq (1 + \text{tol}) \times (\text{the current lower bound})\]

where tol is the desired accuracy specified through the ACCURACY
parameter.

The bisection with extrapolation is quadratically convergent
and is usually faster than the simple bisection. The updating
of lower and upper bound is however more involved. It requires
additionally the evaluation, for each pair of imaginary eigenvalues,
of the maximal singular value of the transfer function matrix
evaluated at one of that eigenvalues. Two pairs of imaginary
eigenvalues which produce the two largest maximum singular values,
are used to update the lower and upper bounds for the next
bisection step.

For a discrete-time system the same method is used after applying to
Q a bilinear transformation which preserves its H-infinity norm /4/.

Literature
/1/ Boyd S., Balkrishnan V., and Kabamba P.
A bisection method for computing the H-Infinity norm of a transfer matrix and related problems,
/2/ Bruinsma N.A. and Steinbuch M.
A fast algorithm to compute the H-infinity norm of a transfer function matrix,
/3/ A. Varga
A Schur method for computing coprime factorizations with inner denominators and applications in model reduction,
/4/ Al-Saggaf U.M. and Franklin G.F.

Remarks:
none

Life cycle:
1993 Aug  A. Varga DLR FF-DF: coded

Packages required:
none

Libraries required:
RASP, LAPACK, BLAS (1,2,3)

Example:
For the stable continuous-time system (A,B,C,D) used by the module BTA the computed infinity-norm is 4.2333 attained at frequency 3.093.

For the unstable continuous-time system (A,B,C,D) used by the module ASD the computed infinity-norm is 20.06 attained at frequency 0.16.