CR-IDENT: A MATLAB Toolbox for Multivariable Control-Relevant System Identification

Hyunjin Lee [1] and Daniel E. Rivera[2]

Control Systems Engineering Laboratory
Department of Chemical Engineering
Ira A. Fulton School of Engineering
Arizona State University, Tempe, Arizona 85287-6006

January, 2008

[1] Currently with the Department of Chemical and Biological Engineering, Rensselaer Polytechnic Institute, Troy, NY. Email: hlee6@rpi.edu or peacemaple@hotmail.com

[2] To whom all correspondence should be addressed. Phone: (480) 965-9476, Fax: (480) 965-0037, E-mail: daniel.rivera@asu.edu

Contents

1 CR-IDENT Overview

This document describes CR-IDENT, a Matlab-based toolbox that implements a comprehensive framework for multivariable control-relevant system identification aimed primarily at process system applications. CR-IDENT consists of a series of graphic user interface (GUI) modules that accomplish (1) multivariable input signal design for multisine and pseudo-random binary sequences (PRBS), (2) frequency response estimation, (3) control-relevant frequency response curvefitting, and (4) robust loopshaping. The toolbox is intended to generate efficient models whose end use is the design of high-performance model-based control systems (Rivera *et al.*, 2003; Lee, 2006). An important component in the implementation of this design procedure is its reliance on *a priori* knowledge of the system of interest to design input signals meeting both theoretical and practical user requirements. Data from identification testing using these signals are the basis for the subsequent steps of frequency-response estimation, and control-relevant parameter estimation. A resultant representation is a discrete-time state-space model that can be used as the nominal model for Model Predictive Control or other forms of multivariable control design.

Improving the link between system identification and control design has been a subject of great interest in the control systems literature for nearly two decades (Hjalmarsson, 2005). There is a continuing need for control-relevant identification methodologies focused on multivariable problems that appeal to both academic and industrial practitioners. In principle, such methodologies should be comprehensive in nature, take full advantage of *a priori* knowledge of a system to be identified, be as short and non-invasive as possible to the process (i.e., “plant-friendly”), and not make substantial demands on user-skill levels in its implementation (Rivera *et al.*, 2003). Based on recent research activities in the Control Systems Engineering Laboratory at Arizona State University, CR-IDENT has been developed with these goals in mind (Rivera *et al.*, 2003; Lee, 2006).

The functionality implemented in the CR-IDENT is summarized in Figure 1. Although aimed primarily at process system applications, the methodology is broadly-applicable and can be useful in multiple application domains. These modules can be used independently or as part of an integrated procedure, as shown in Figure 1. The functionality of each module is accessed with a graphical user interface (GUI) to provide flexibility and convenience to the user. The theoretical background behind CR-IDENT is described in various papers by Lee *et al.* (2003*a*; 2003*b*), Lee and Rivera (2004; 2005*a*; 2005*b*), Rivera *et al.* (2007) and the Ph.D. dissertation by Lee (2006).

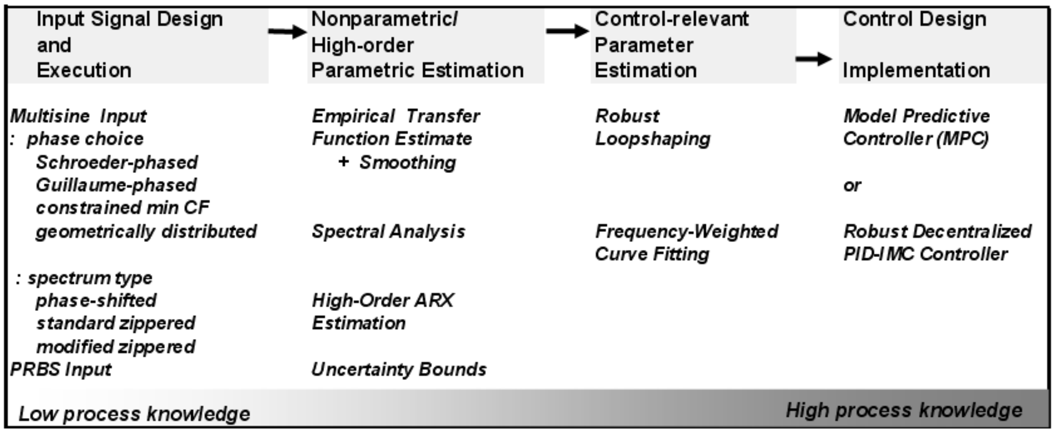
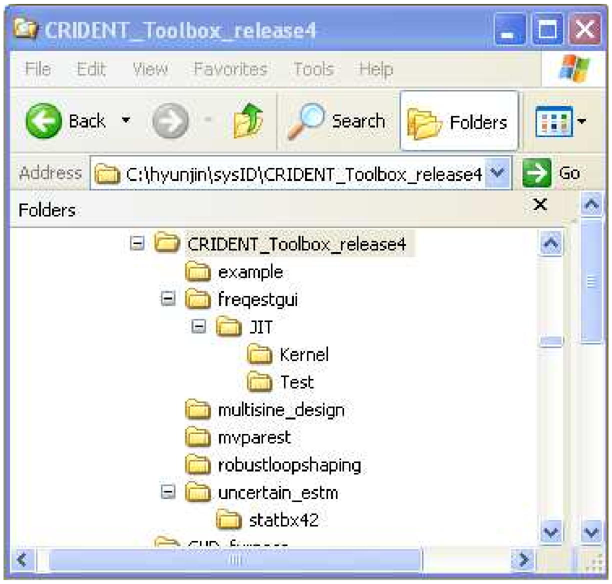

Input Signal Design and Execution
Nonparametric/ High-order Parametric Estimation
Control-relevant Parameter Estimation
Control Design Implementation
Multisine Input
: phase choice
Schroeder-phased
Guillaume-phased
constrained min CF
geometrically distributed
: spectrum type
phase-shifted
standard zippered
modified zippered
PRBS Input
Empirical Transfer Function Estimate
+ Smoothing
Spectral Analysis
High-Order ARX Estimation
Uncertainty Bounds
Robust Loopshaping
Frequency-Weighted Curve Fitting
Model Predictive Controller (MPC)
or
Robust Decentralized PID-IMC Controller
Low process knowledge
High process knowledge

Figure 1: Summary of the design procedure and functionality available in CR-IDENT, a comprehensive framework for multivariable control-relevant system identification.

This document presents the general features of the CR-IDENT and helps the users get started and acquire the experience of using this toolbox. In the following, we will introduce the four main GUIs in the CR-IDENT with step-by-step guidelines for the user. Moreover, two illustrative example cases are provided based on the Shell Heavy-Oil Fractionator Problem (Prett and García, 1988) and Jacobsen-Skogestad high-purity distillation columns (Jacobsen and Skogestad, 1994). These examples can be accessed directly from the GUI. A variety of multisine input signal designs supported by the toolbox is demonstrated and the data arising from these designs are used to obtain models which are evaluated through the closed-loop Model Predictive Control application. This document is organized as follows: Section 2 describes the installation of the CR-IDENT release and Section 3 shows how to get started with the CR-IDENT. Section 4 explains the Multivariable Input Signal Design GUI and Section 5 introduces examples of Matlab Simulink for the open-loop experiment simulation. Sections 6 and 7 present the Frequency Response Estimation GUI and Control-Relevant Curvefitting GUI, respectively. Section 8 presents the Robust Loopshaping GUI that involves uncertainty estimation, performance weight specification, and μ-analysis. The illustrative example cases are given with step-by-step procedures and discussed in Section 9. Section 9.3 presents an illustrative example that follows the identification test monitoring procedure taking advantage of interactive and iterative usage for control-relevant modeling, targeted on Model Predictive Control application. Finally, Section 10 presents remarks and conclusions.

2 Installing CR-IDENT and System Requirements

The directory structure for the CR-IDENT toolbox is shown in Figure 2



Address: C:\hyunjin\sysID\CRIDENT_Toolbox_release4

```
⊟ CRIDENT_Toolbox_release4
     example
   ⊟ freqestgui
      ⊟ JIT
           Kernel
           Test
     multisine_design
     mvparest
     robustloopshaping
   ⊟ uncertain_estm
        statbx42
```

Figure 2: Directory structure for CR-IDENT Toolbox

The toolbox is designed using Matlab R14 with Service Pack 3 (Version 7.1) and requires the Signal Processing, Control System, System Identification, and Model Predictive Control Toolboxes as well as Simulink.

Installation of the CR-IDENT Toolbox is accomplished by the following steps:

1. Copy “CRIDENT_Toolbox_releasexx.zip” at a directory.

2. Uncompress “CRIDENT_Toolbox_releasexx.zip” from the current directory. If it is properly uncompressed, the main directory contains the same sub-directories as Figure 2.

3. Start Matlab on your computer and open “Set Path” window from “File” menu on the Matlab command window.

4. Click “Add with Subfolders” and select “CRIDENT_Toolbox_releasexx” directory. Then, it will automatically include the sub-directories.

5. Hit the “Save” button and close the “Set Path” window.

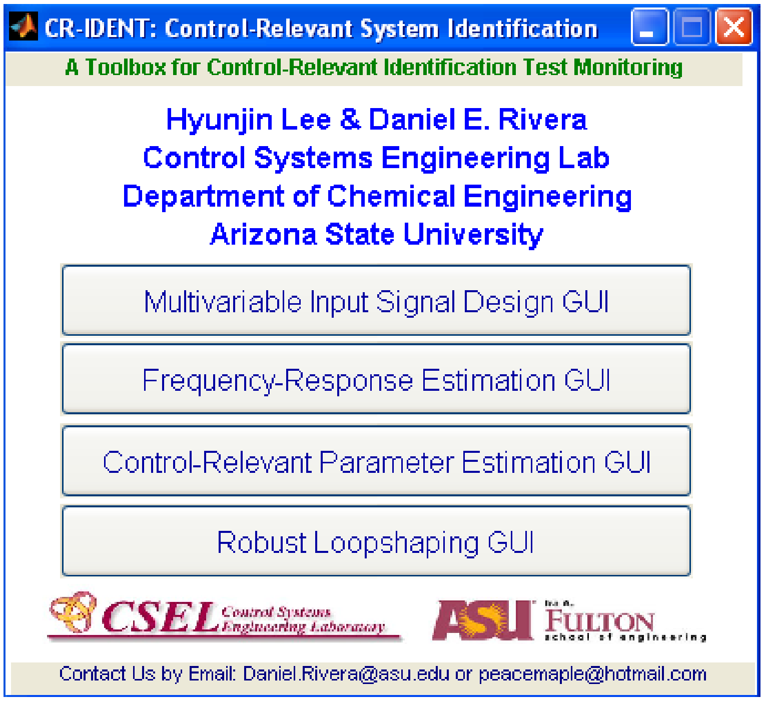
Although the CR-IDENT toolbox should work in all Matlab plaforms, its development has been primarily done using PC-Windows.

3 Getting Started with the CR-IDENT Toolbox

Now, CR-IDENT is ready to be started by the user. Invoking the command

```
>> crident
```

calls a main menu (see Figure 3) with options corresponding to each of the four GUIs that comprise CR-IDENT. The toolbox consists of modules for multi-channel input signal design (multisine and PRBS), frequency-response estimation, control-relevant frequency-response curvefitting, and robust loopshaping.



CR-IDENT: Control-Relevant System Identification

A Toolbox for Control-Relevant Identification Test Monitoring

Hyunjin Lee & Daniel E. Rivera
Control Systems Engineering Lab
Department of Chemical Engineering
Arizona State University

Multivariable Input Signal Design GUI
Frequency-Response Estimation GUI
Control-Relevant Parameter Estimation GUI
Robust Loopshaping GUI
CSEL Control Systems Engineering Laboratory
ASU Ira A. Fulton school of engineering

Contact Us by Email: Daniel.Rivera@asu.edu or peacemaple@hotmail.com

Figure 3: The main GUI for CR-IDENT, a toolbox for multivariable control-relevant identification.

To open each individual GUI, the user needs to hit the buttons on the CR-IDENT GUI, which will then open the corresponding GUI window. Currently, it is not possible to run multiple instances of the same GUI. Salient aspects for each GUI module are described in the ensuing sections.

4 Multivariable Input Signal Design GUI

4.1 Background in Multisine Input Signal Design

By enabling direct specification of the power spectrum, multisine signals represent a versatile class of inputs for accomplishing system identification (see Figure 5). A multisine input $u_j(k)$ for the j-th channel of a multivariable system with m inputs is defined as

$$\begin{aligned} u_j(k) &= \sum_{i=1}^{m\delta} \hat{\delta}_{ji} \cos(\omega_i kT + \phi_{ji}^{\delta}) + \sum_{i=m\delta+1}^{m(\delta+n_s)} \sqrt{2\alpha_{ji}} \cos(\omega_i kT + \phi_{ji}) \\ &+ \sum_{i=m(\delta+n_s)+1}^{m(\delta+n_s+n_a)} \hat{a}_{ji} \cos(\omega_i kT + \phi_{ji}^{a}), \;\; j = 1, \cdots, m \end{aligned} \tag{1}$$

where m is the number of channels, δ, n_s, n_a are the number of sinusoids per channel ($m(\delta + n_s + n_a) = N_s/2$), $\phi_{ji}^{\delta}, \phi_{ji}, \phi_{ji}^{a}$ are the phase angles, and $\hat{\delta}_{ji}$, α_{ji}, & $\hat{a}_{ji}$ represents the Fourier coefficients defined by the user. $\hat{\delta}_{ji}, \hat{a}_{ji}$ are the "snow effect" Fourier coefficients, and $\omega_i = 2\pi i/N_sT$ is the frequency grid. The "snow effect" (Guillaume *et al.*, 1991) refers to sinusoids where both Fourier coefficients and phases are selected by the optimizer, which can serve as an aid for achieving plant-friendliness (Lee *et al.*, 2003*b*). The snow effect is not implemented in the current version of CR-IDENT; instead the user can specify lf and hf which are Fourier coefficients in the low and high frequency grids. Figure 4 shows a general zippered input power spectrum for a three-channel MIMO case as implemented in CR-IDENT.

For orthogonal input signals, the power spectrum of channels should not be excited at the same frequencies; only a single input channel is excited at a specific frequency, while the other channels are suppressed. To achieve a zippered spectrum, we define the Fourier coefficients α_{ji} for $u_j(k)$ as

$$\alpha_{ji} = \begin{cases} \neq 0, & i = m\delta + j, m(\delta+1) + j, ..., m(\delta + n_s - 1) + j \\ = 0, & \text{for all other } i \text{ up to } m(\delta + n_s) \end{cases} \tag{2}$$

A frequency range of interest containing the bandwidth of the primary power spectrum should fall within the range specified by the inequality of (3) (Braun *et al.*, 2002; Rivera *et al.*, 2002).

$$\frac{1}{\beta\, \tau_{dom}^{H}} = \omega_* \leq \;\omega\; \leq \omega^* = \frac{\alpha}{\tau_{dom}^{L}} \tag{3}$$

Within the primary bandwidth, sinusoidal harmonics perturb the dynamics of a system of

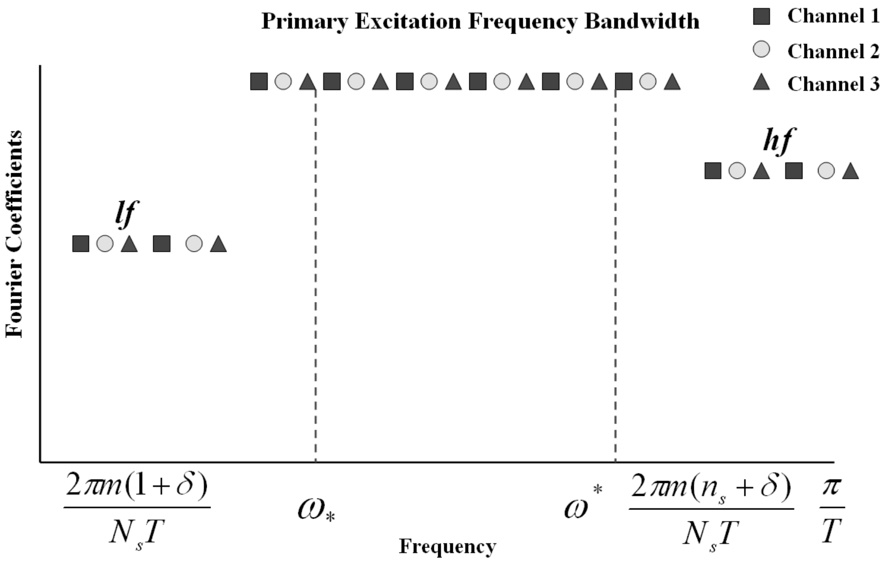

Primary Excitation Frequency Bandwidth
■ Channel 1
○ Channel 2
▲ Channel 3
Fourier Coefficients
lf
hf
$\frac{2\pi m(1+\delta)}{N_s T}$
ω_*
ω^*
$\frac{2\pi m(n_s+\delta)}{N_s T}$
$\frac{\pi}{T}$
Frequency

Figure 4: Conceptual design of a standard “zippered” spectrum for a three-channel signal.

interest to produce informative datasets for system identification. Consequently,

$$\frac{2\pi m(1+\delta)}{N_s T} \leq \omega_* \leq \omega \leq \omega^* \leq \frac{2\pi m(n_s+\delta)}{N_s T} \leq \frac{\pi}{T} \tag{4}$$

which in turn translates into the following inequalities for the number of sinusoids, the sampling time, and the sequence length (n_s, T, andN_s, respectively)

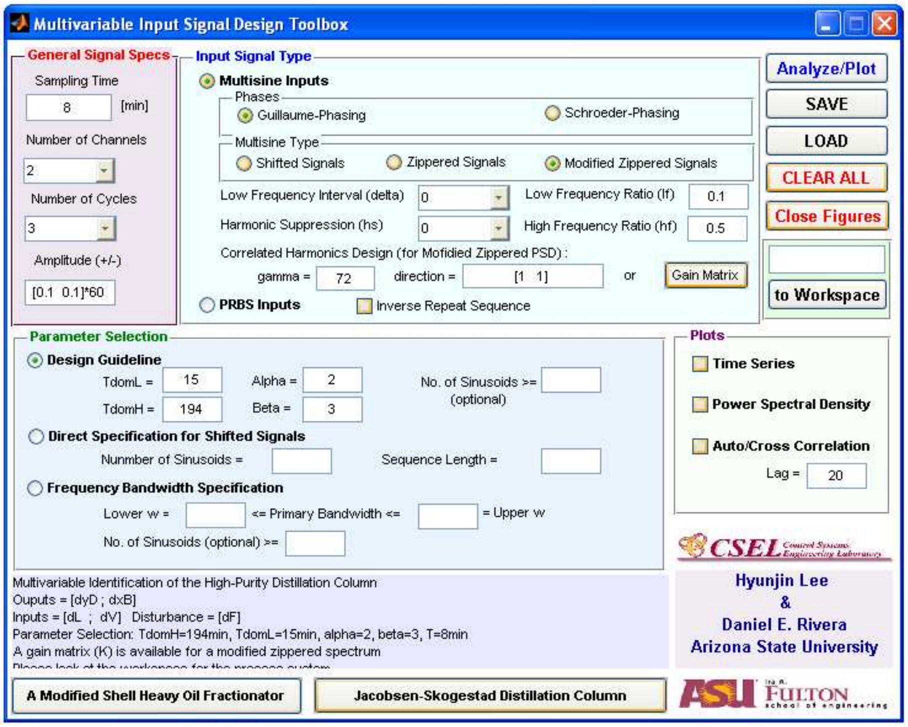
$$n_s + \delta \geq (1+\delta)\frac{\omega^*}{\omega_*} \tag{5}$$

$$T \leq \min\left(\frac{\pi}{\omega^*}, \ \frac{\pi}{\omega^* - \omega_*}\frac{n_s - 1}{n_s + \delta}\right) \tag{6}$$

$$\max\left(2m(n_s+\delta), \ \frac{2\pi m(1+\delta)}{\omega_* T}\right) \leq N_s \leq \frac{2\pi m(n_s+\delta)}{\omega^* T} \tag{7}$$

The principal design guideline implemented in the CR-IDENT uses *a priori* knowledge of dominant time constants of the system and speed of the response specifications to define a primary bandwidth for excitation in the signal (Lee *et al.*, 2003*b*; Lee and Rivera, 2005*b*). For users not wishing to use these guidelines, the input GUI supports direct parameter specification with a verification routine to insure proper implementation.

The input design GUI (Figure 5) offers various options for multi-channel implementation such as phase-shifted, orthogonal (“zippered”) spectrum, and modified zippered spectrum designs (Lee and Rivera, 2004). The phase-shifted multisine input design adopts a technique well-known in the literature for pseudo-random signals in which multiple channels


Multivariable Input Signal Design Toolbox
General Signal Specs
Input Signal Type
Analyze/Plot
SAVE
LOAD
CLEAR ALL
Close Figures
to Workspace
Gain Matrix
Parameter Selection
Plots
CSEL Control Systems Engineering Laboratory
Hyunjin Lee
&
Daniel E. Rivera
Arizona State University
A Modified Shell Heavy Oil Fractionator
Jacobsen-Skogestad Distillation Column
ASU
Ira A. Fulton school of engineering

Sampling Time 8 [min]

Number of Channels 2

Number of Cycles 3

Amplitude (+/-) [0.1 0.1]*60

● Multisine Inputs

Phases: ● Guillaume-Phasing ○ Schroeder-Phasing

Multisine Type: ○ Shifted Signals ○ Zippered Signals ● Modified Zippered Signals

Low Frequency Interval (delta) 0 Low Frequency Ratio (lf) 0.1

Harmonic Suppression (hs) 0 High Frequency Ratio (hf) 0.5

Correlated Harmonics Design (for Mofidied Zippered PSD) :

gamma = 72 direction = [1 1] or

○ PRBS Inputs □ Inverse Repeat Sequence

● Design Guideline

TdomL = 15 Alpha = 2 No. of Sinusoids >= (optional)

TdomH = 194 Beta = 3

○ Direct Specification for Shifted Signals

Nunmber of Sinusoids = Sequence Length =

○ Frequency Bandwidth Specification

Lower w = <= Primary Bandwidth <= = Upper w

No. of Sinusoids (optional) >=

□ Time Series

□ Power Spectral Density

□ Auto/Cross Correlation

Lag = 20

Multivariable Identification of the High-Purity Distillation Column
Ouputs = [dyD ; dxB]
Inputs = [dL ; dV] Disturbance = [dF]
Parameter Selection: TdomH=194min, TdomL=15min, alpha=2, beta=3, T=8min
A gain matrix (K) is available for a modified zippered spectrum
Please look at the workspace for the process system

Figure 5: Multivariable Input Signal Design GUI in CR-IDENT.

are shifted relative to each other in order to reduce the interactions between channels. A similar feature is available for PRBS signals. A zippered power spectrum design gives orthogonality to individual input channels so that only one input channel is excited at each frequency grid. The result of this approach is longer signal lengths relative to shifted signals, but with lower levels of cross-correlation.

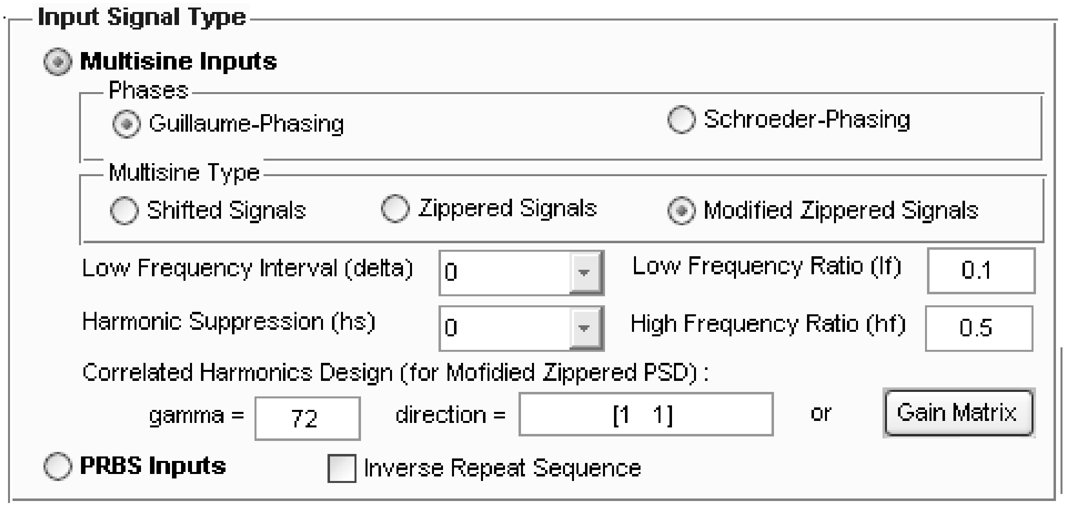
For the case of strongly interactive systems, CR-IDENT offers the multisine signal with a modified zippered spectrum that contains both correlated and uncorrelated harmonics over the frequency bandwidth. A modified zippered spectrum enables a directional adjustment that can be used to emphasize specific gain directions, i.e., the low-gain direction. This is a critical consideration in the identification of highly-interactive systems (Lee *et al.*, 2003*b*; Lee and Rivera, 2005*b*; Rivera *et al.*, 2007). If users provide a gain matrix, the GUI computes the system gain directions and adjusts the signal's input directions and power amplitudes to produce a more balanced distribution in the output state-space (Lee and Rivera, 2005*b*).

For all multisine signal choices shown in Figure 5, the phases can be obtained through either the closed-form formula by Schroeder (1970) or the iterative *p*-norm optimization approach that minimizes crest factor developed by Guillaume *et al.* (1991). Signals can be validated in both time and frequency domains and be exported in “struct” format. A modified Shell Heavy Oil Fractionator (Prett and García, 1988) and Jacobsen-Skogestad high purity distillation column (Jacobsen and Skogestad, 1994) problems are provided as examples with the CR-IDENT. When the user selects the example buttons on the GUI, these bring up default parameters that represent sensible choices of design variables for each of the two example systems.

4.2 Generating Multi-Channel Input Signals

The detailed procedure of how to generate input signals using the Multivariable Input Signal Design GUI is given here.

4.2.1 Input Signal Type



Input Signal Type

● Multisine Inputs

Phases

● Guillaume-Phasing ○ Schroeder-Phasing

Multisine Type

○ Shifted Signals ○ Zippered Signals ● Modified Zippered Signals

Low Frequency Interval (delta) 0

Low Frequency Ratio (lf) 0.1

Harmonic Suppression (hs) 0

High Frequency Ratio (hf) 0.5

Correlated Harmonics Design (for Mofidied Zippered PSD) :

gamma = 72 direction = [1 1] or

○ PRBS Inputs □ Inverse Repeat Sequence

Figure 6: Input signal type selection on the Input Design GUI

First, the user determines an input signal type between “Multisine” or “PRBS.” The GUI will disable some components depending on the user selection.

Multisine Inputs:
“Phases” should be either Guillaume-Phasing (Guillaume *et al.*, 1991) or Schroeder-Phasing (Schroeder, 1970). The crest factor minimization algorithm using Guillaume-Phasing takes more time to compute than Schroeder-Phasing.

“Multisine Type” should be given among Shifted, Zippered, or Modified Zippered design that specifies an input power spectrum for multiple input channels. If the number of input channels is one, they are all identical.

“Low Frequency Interval (δ)” creates the given number of channel groups in the input power spectrum at low frequency grids using *“Low Frequency Ratio (lf)”* (≤ 1). The coefficients in the lower frequency grids in Figure 4 that correspond to lf.

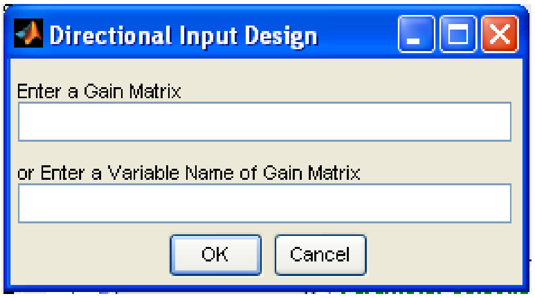
“High Frequency Ratio (hf)” (≤ 1) defines Fourier coefficients at higher frequencies outside the primary bandwidth. The coefficients in the higher frequency grid in Figure 4 have value hf.

“Harmonic Suppression (hs)” suppresses sinusoidal harmonics at the frequency grids of the multiples of prime number (2,3,5,...) in the input power spectrum.

“Correlated Harmonics Design” is for a modified zippered spectrum that requires two additional design factors.

- “*gamma* (γ)” applies its ratio to the correlated harmonics.
- *“direction”* specifies a particular input directional vector (in real or complex values) of interest to the correlated harmonics.

This can be given directly by the user or can be computed from a steady-state gain by hitting on the **Gain Matrix** button. Then, the following window will show up.


Directional Input Design

Enter a Gain Matrix

or Enter a Variable Name of Gain Matrix

Figure 7: Entering a gain matrix for design a modified zippered spectrum

The user can directly enter a gain matrix or a variable name that exists in the Matlab workspace. A general scheme of the modified zippered spectrum is illustrated in Figure 8.

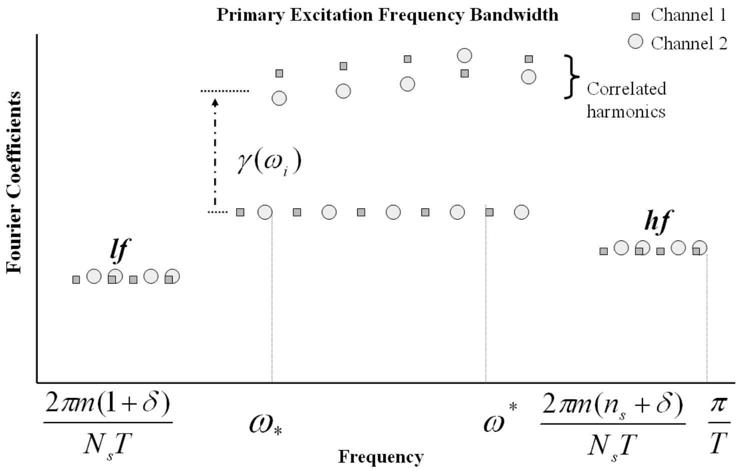


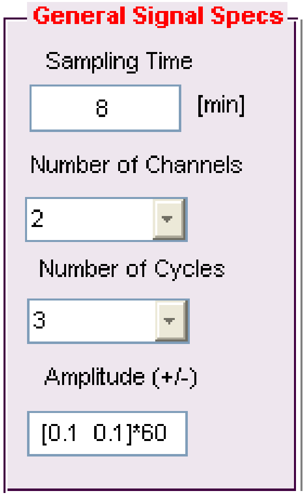
Figure 8: Conceptual design of a modified zippered power spectrum for 2-channel signal

PRBS Inputs:
PRBS inputs rely on the phase-shifting method for multiple input implementation.

"Inverse Repeat Sequence" results in an even frequency grid harmonic suppression effect.

4.2.2 General Signal Specs

Although the user can determine general signal specs, more often these parameters should be based on the system's physical characteristics.



General Signal Specs

Sampling Time

8 [min]

Number of Channels

2

Number of Cycles

3

Amplitude (+/-)

[0.1 0.1]*60

Figure 9: General input signal specs on the Input Design GUI

"Sampling Time" is the minimum hold time of change for the input sequence, which should

agree with the measurement sampling time in the output.

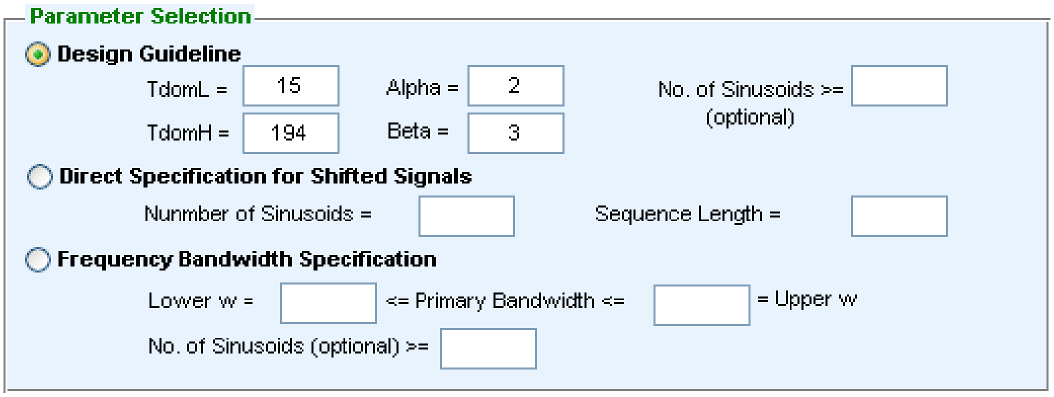
"Number of Channels" corresponds to the number of input channels in a given system. The user can select a value between 1 and 20 on the current GUI, or numerically specify a number larger than 20.

"Number of Cycles" repeats one period of the input signal for the specified number of cycles.

"Amplitude" is the input magnitude that can be given in several ways for multiple channels. For example when the user has two input channels, the amplitude can be given either in string or number format,

- $1 \equiv [1 \quad 1] \equiv 1 * ones(2,1)$
- $[1 \quad 0.5]$

4.2.3 Parameter Selection for Signal Bandwidth



Parameter Selection

● Design Guideline

TdomL = 15

TdomH = 194

Alpha = 2

Beta = 3

No. of Sinusoids >= (optional)

○ Direct Specification for Shifted Signals

Nunmber of Sinusoids =

Sequence Length =

○ Frequency Bandwidth Specification

Lower w = <= Primary Bandwidth <= = Upper w

No. of Sinusoids (optional) >=

Figure 10: Design parameter selection on the Input Design GUI

Design Guideline:
a priori process information is utilized for this option, which specifies the primary excitation bandwidth as described before from (3) to (7)

"$TdomL$" and "$TdomH$" indicate the low and high dominant time constants, i.e., τ^L_{dom} and τ^H_{dom}.

"$Alpha$" and "$Beta$" represent α_s and β_s for the speed of closed-loop response and settling time requirement, respectively.

"No. of Sinusoids" is an optional parameter allows the user to specify the number of sinusoids in the primary excitation bandwidth.

Direct Specification for Shifted Signals:
The options are changed based on Multisine and PRBS, respectively, such that

1. Multisine

   - *"Number of Sinusoids"* directly specifies the number of sinusoids for the primary bandwidth.
   - *"Sequence Length"* directly specifies the sequence length for multisine input signals.

2. PRBS

   - *"Switching intervals (nsw)"* indicates that the input magnitude will remain unchanged for at least this time interval.
   - *"No. of shift registers"* specifies for the number of registers for the pseudo-random binary sequence generator.

Frequency Bandwidth Specification:
This is a simple specification relative to the previous selections. It allows the user to specify a primary bandwidth (ω_* - ω^*) for both multisine and PRBS inputs (see Equation (3)).

"Lower w" indicates ω_* for the lower bound of a primary bandwidth.

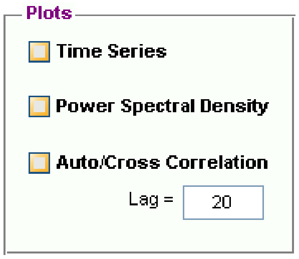
"Upper w" indicates ω^* for the upper bound of a primary bandwidth.

"No. of Sinusoids" is only available with the Multisine signal option, which generates that amount of sinusoids within the specified bandwidth.

4.2.4 Plots

The GUI provides several plots for the input signal.

- *"Time Series"* plots one cycle of the generated input signals.
- *"Input Power Spectrum"* draws the input power spectrum based on one cycle.
- *"Auto/Cross Correlation"* takes the correlation analysis for input signals. The user can specify the lag size.



Lag = 20

Figure 11: Plotting options on the Input Design GUI

4.2.5 Action Buttons

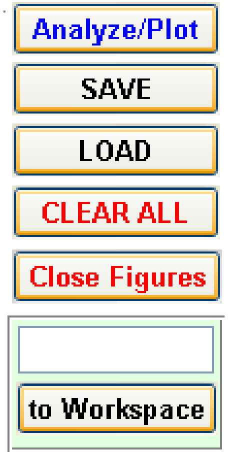


Figure 12: Action buttons on the Input Design GUI

The GUI provides a series of the action buttons for signal generation and data management.

"Analyze/Plot" generates input signals with selected user's choices and plot options. A small bar window shows up for the progress status.

"SAVE" opens a window to save the current signal data to a file.

"LOAD" opens a window to load a signal data file to the current GUI.

"CLEAR ALL" clears all the data information on the GUI.

"Close Figures" closes all the open Matlab figures.

"toWorkspace" transfers the input signal data to the Matlab workspace with a variable name on the text box given by the user.

4.3 Data Structure from Input Signal Design GUI

The data structure from the Input Signal Design GUI has a number of components that are needed for the internal operation of the GUI. The essential components for executing an identification testing experiment, as well as being useful for subsequent portions of CR-IDENT, are as follows

Components	Description
u	Inputs
T_s	Sampling time
N_s	One cycle sequence length
n_s	Number of sinusoids in the primary bandwidth

Table 1: Essential components of the data in "struct" format from Multivariable Input Signal Design GUI

4.4 Examples

The input design GUI provides two example cases based on the Shell Heavy Oil Fractionator (Prett and García, 1988) and the Jacobsen-Skogestad (Jacobsen and Skogestad, 1994) high-purity distillation column problems. Preset values are loaded on the GUI when buttons for these examples are selected; the user can then analyze these examples within the GUI as desired.

A Modified Shell Heavy Oil Fractionator
Jacobsen-Skogestad Distillation Column

Figure 13: Illustrative examples on the Input Signal Design GUI

5 Running Experiment from Input Design GUI

The user can use the designed input data in variety of ways for simulation or experiment for identification testing. To use the input signal on the Matlab workspace, it can be available directly from the GUI using "toWorkspace" or using a command "load filename" on the Matlab command window. With a designed input signal, u, N_s, and T_s are used as essential information for running the testing experiment. The user has to decide a number of cycles for the total test duration. N_s should not be modified and will be the basis for calculating the test cycles.

The simulation or experiment styles may depend on the user. However, the input and output data should be collected properly for the further identification analysis. Figures 14 and 15 show typical examples of the simulation set using Matlab Simulink.

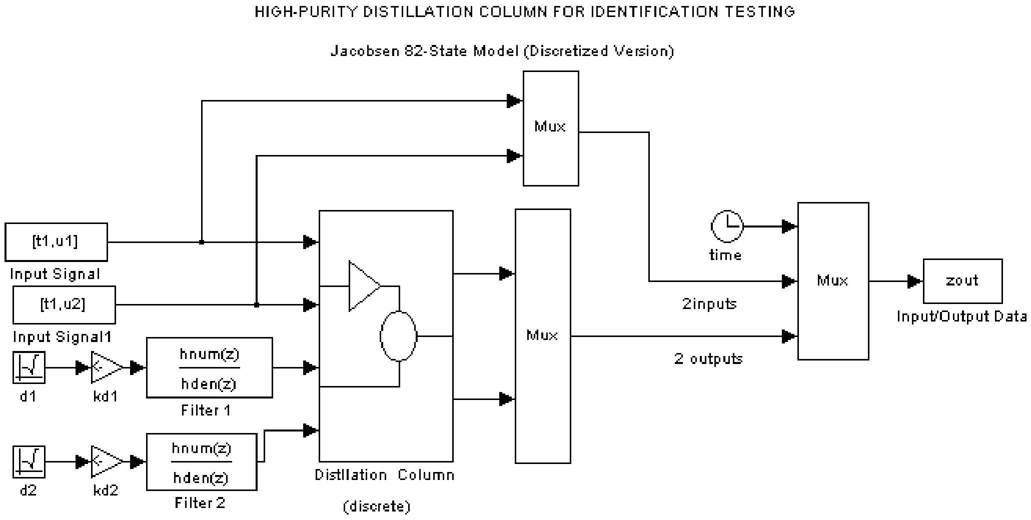


Figure 14: A Matlab Simulink simulation for Jacobsen-Skogestad Distillation Column

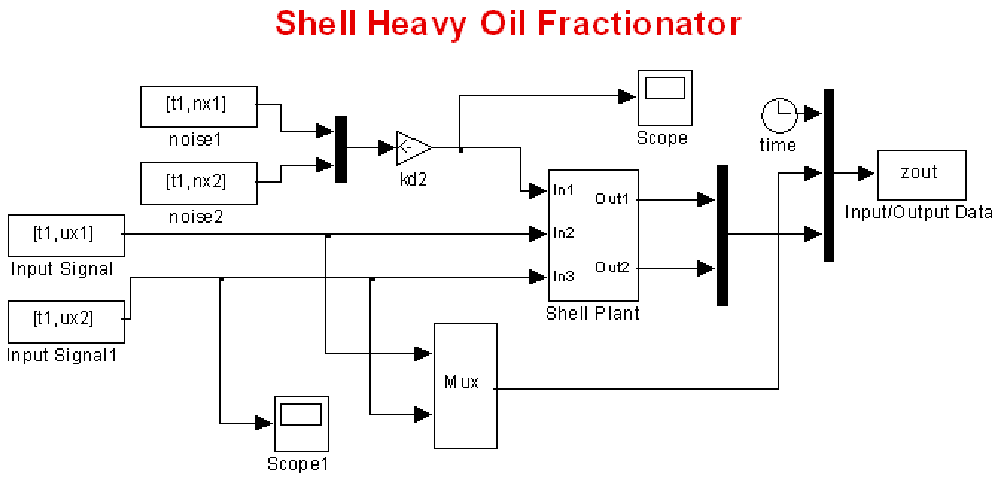


Figure 15: A Matlab Simulink simulation for a modified Shell Heavy-Oil Fractionator Problem

6 Frequency Response Estimation GUI

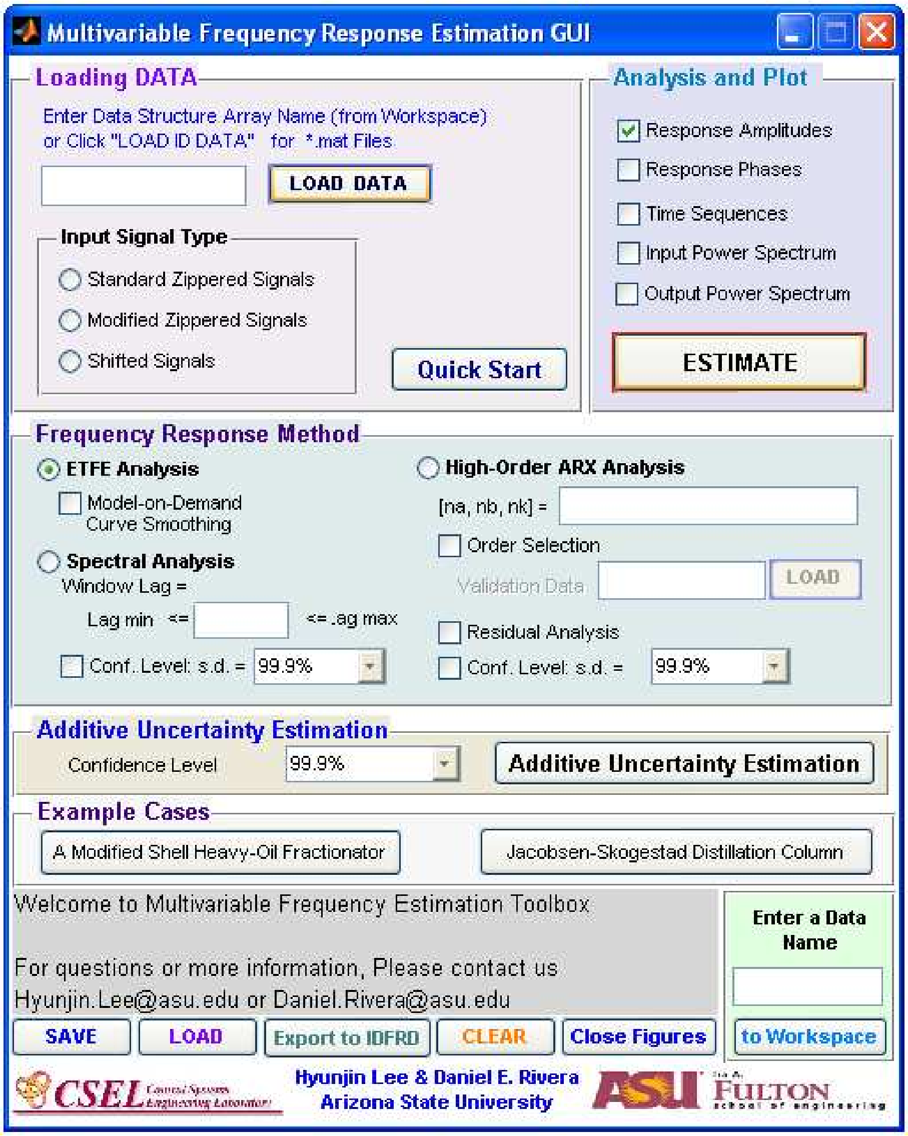
6.1 Background in Frequency Response Estimation

The Frequency-Response Estimation GUI (Figure 16) enables estimating frequency responses from multisine-generated data using both parametric and non-parametric approaches.

The Empirical Transfer Function Estimate (ETFE) and Spectral Analysis (SA) methods are used for non-parametric estimation, while high-order ARX models are utilized for parametric frequency-response estimation. For noisy ETFE responses, a Model-on-Demand curve smoothing algorithm by Stenman *et al.* (2000) is available. The specific requirements created by zippered frequency grids are considered for orthogonal ETFE computation; this also includes the treatment of signals that involve harmonic suppression. Since multisine signals with the modified zippered spectrum contain correlated harmonics, ETFEs cannot be computed; however, SA and ARX estimation options are available, instead.

6.2 Entry Data Format for Frequency Response Estimation GUI

Having completed experimental testing, the generated dataset can be used for further analysis in CR-IDENT. The first cycle of the data is usually not used for the identification analysis in order to reduce the impact of transient effects.


Multivariable Frequency Response Estimation GUI
Loading DATA
Enter Data Structure Array Name (from Workspace)
or Click "LOAD ID DATA" for *.mat Files
LOAD DATA
Input Signal Type
○ Standard Zippered Signals
○ Modified Zippered Signals
○ Shifted Signals
Quick Start
Analysis and Plot
Response Amplitudes
□ Response Phases
□ Time Sequences
□ Input Power Spectrum
□ Output Power Spectrum
ESTIMATE
Frequency Response Method
ETFE Analysis
□ Model-on-Demand Curve Smoothing
○ Spectral Analysis
Window Lag =
Lag min <= <= Lag max
□ Conf. Level: s.d. =
○ High-Order ARX Analysis
[na, nb, nk] =
□ Order Selection
Validation Data
LOAD
□ Residual Analysis
□ Conf. Level: s.d. =
Additive Uncertainty Estimation
Confidence Level
Additive Uncertainty Estimation
Example Cases
A Modified Shell Heavy-Oil Fractionator
Jacobsen-Skogestad Distillation Column
Enter a Data Name
SAVE
LOAD
Export to IDFRD
CLEAR
Close Figures
to Workspace
CSEL Control Systems Engineering Laboratory
Hyunjin Lee & Daniel E. Rivera
Arizona State University
ASU
FULTON school of engineering

☑ Response Amplitudes

● ETFE Analysis

Conf. Level: s.d. = 99.9%

Conf. Level: s.d. = 99.9%

Confidence Level 99.9%

Welcome to Multivariable Frequency Estimation Toolbox

For questions or more information, Please contact us
Hyunjin.Lee@asu.edu or Daniel.Rivera@asu.edu

Figure 16: Frequency-Response Estimation GUI.

The GUI takes “iddata” or “struct” format as input data whose essential components include the following: input and output time series, sampling time, the sequence length for one cycle, and the number of sinusoids (only for “struct”; see Table 2). Once the input data is loaded, the GUI analyzes an input signal type and blocks the choices that are not available with the given input type.

Users can export the estimated responses in “idfrd” or “struct” format. These are used in the ensuing modules in support of advanced control system design, e.g., control-relevant parameter estimation and robust loopshaping, among other uses.

Components	Description
y	Outputs
u	Inputs
T_s	Sampling Time
N_s	One cycle sequency length
n_s	Number of sinusoids (in “struct”)

Table 2: Components of the experimental data, entry structure for Frequency Response Estimation GUI

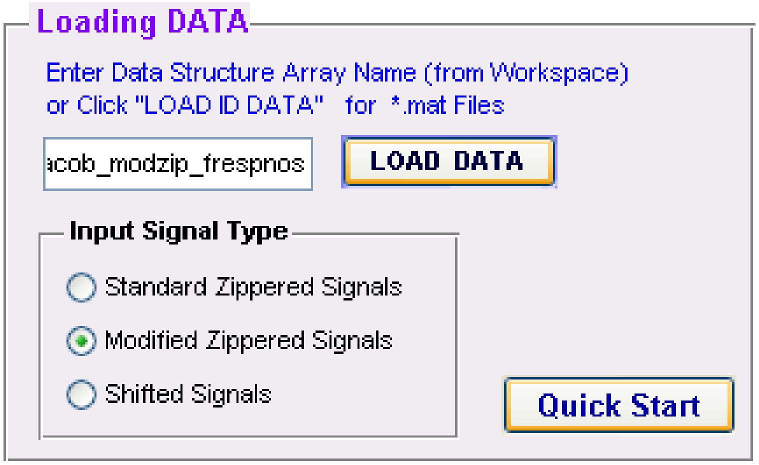
6.3 Start Estimating Frequency Reponses

6.3.1 Loading DATA

An experimental dataset is loaded using the block in Figure 17. The user can load the data from the Matlab workspace or a *.mat file of either “struct” or “iddata” object.

From the workspace, the user enters a data name on the edit block and hit the *“LOAD DATA”* button. If the entered data is not available on the workspace, an error message will appear.

From a *.mat file, the user hits the *“LOAD DATA”* button with the empty edit box. Then, it prompts the file open window for selecting a data file.



Enter Data Structure Array Name (from Workspace)
or Click "LOAD ID DATA" for *.mat Files

acob_modzip_frespnos

○ Standard Zippered Signals
● Modified Zippered Signals
○ Shifted Signals

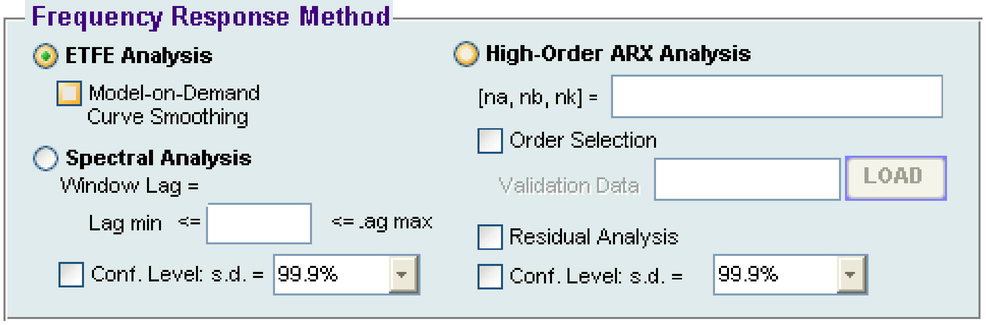
Figure 17: Load an input/output dataset to Frequency Response Estimation GUI

Once the experiment dataset is loaded, the GUI determines the input signal type based on its input power spectrum. The user has no control over the *“Input Signal Type”*.

“Quick Start” runs a series of frequency response estimations with basic options available on the GUI.

6.3.2 Freqeuncy Response Methods

The user decides a method of frequency response estimation with the loaded input/output data using the selection options in Figure 18.



Frequency Response Method

● ETFE Analysis

☐ Model-on-Demand Curve Smoothing

○ Spectral Analysis

Window Lag =

Lag min <= [ ] <= .ag max

☐ Conf. Level: s.d. = 99.9%

○ High-Order ARX Analysis

[na, nb, nk] = [ ]

☐ Order Selection

Validation Data [ ] LOAD

☐ Residual Analysis

☐ Conf. Level: s.d. = 99.9%

Figure 18: Methods of frequency response estimation option

ETFE Analysis computes non-parametric Empirical Transfer Function Estimates. *“Model-on-Demand Curve Smoothing”* is available for smoothing noisy frequency responses. The ETFE of the $h\ell$-th element of G is obtained from a fraction of output and input DFT sequences (Ljung and Glad, 1994) such that

$$g_{h\ell}(\omega_i) = \frac{\frac{1}{r}\sum_{s=1}^{r} Y_h^s(\omega_i)}{U_\ell(\omega_i)} \tag{8}$$

$Y_h^s(\omega_i)$ and $U_\ell(\omega_i)$ represent the DFT sequences of $y_h^s(k)$ and $u_\ell(k)$ at frequency ω_i, respectively, and

$$y_h^s(k) = y_h(k+(s-1)N_s) \tag{9}$$

for $k = 1,\ldots,N_s$ and $s = 1,\ldots,r$.

Spectral Analysis uses the “spa” command of the Matlab. Alternatively, a frequency response estimator can be obtained as follows

$$g_{h\ell}(\omega_i) = \frac{S_{u_\ell\, y_h}(\omega_i)}{S_{u_\ell\, u_\ell}(\omega_i)} \tag{10}$$

where $S_{u_\ell\, y_h}(\omega_i)$ and $S_{u_\ell\, u_\ell(\omega_i)}$ represent power spectra of corresponding signal sequence.

"Lag Window" is determined from the number of sinusoids (n_s) and the sequence length (N_s).

"Conf. Level" (confidence level) is optional for the standard deviation analysis with the frequency response. With a selected confidence level, the GUI generates shaded areas associated with the frequency responses. The user can consult the "spa" command in Matlab for details.

High-Order ARX Analysis utilizes a parametric ARX (auto-regressive exogenous) estimation. From a high-order ARX model, frequency responses are approximated as

$$G(\omega_i) \;\approx\; (1+A_1 z^1+\ldots+A_{na} z^{na})^{-1}(B_0+B_1\, z^1\ldots+B_{nb}\, z^{nb})\, z^{nk} \tag{11}$$

where $z = e^{-j\omega_i T}$ for a discrete-time model. The user can run either single-input multiple-output (SIMO) or multiple-input multiple-output (MIMO) ARX analysis depending on the order in **"[na, nb, nk]"**.

"na" determines the number of discrete output terms in the ARX model. For a 2×2 system, "na" can be given "10" or "10 10" for SIMO and MIMO, respectively.

"nb" indicates the number of discrete input terms in ARX structure, which should be given with respect to each input.

"nk" represents the order of discrete input delay in ARX structure, which should be given with respect to each input.

"Order Selection" is an option available for SIMO-ARX analysis. The user should specify ranges of the orders, e.g., [[1:10] [1:10] [1:1]] for "na", "nb", and "nk", respectively. It also requires a validation dataset with the same data dimension (data length can be different) as the loaded estimation dataset. To load a validation dataset, it is the same for loading the estimation dataset by hitting *"LOAD"* button.

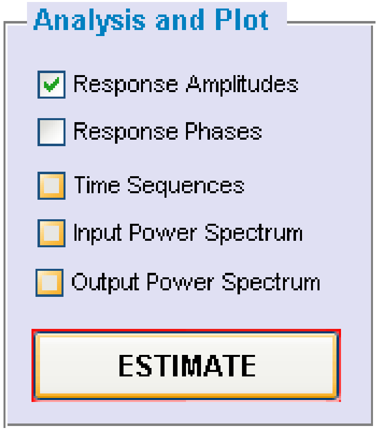
"Residual Analysis" is available with the ARX model and residual data.

"Conf. Level" (confidence level) generates the standard deviation analysis with the pa-

rameter values in the ARX model.

6.3.3 Plots

With the selected options, the user can start analyzing the input/output data into frequency responses interactively on the GUI (Figure 19).



Analysis and Plot

☑ Response Amplitudes

☐ Response Phases

☐ Time Sequences

☐ Input Power Spectrum

☐ Output Power Spectrum

Figure 19: Analysis and Plot for frequency response estimation

"Response Amplitudes" generates the frequency response amplitudes for individual transfer functions.

"Response Phases" generates the frequency response phases for individual transfer functions.

"Time Sequences" draws the input and output time sequence over the total length.

"Input Power Spectrum" provides an input power spectrum.

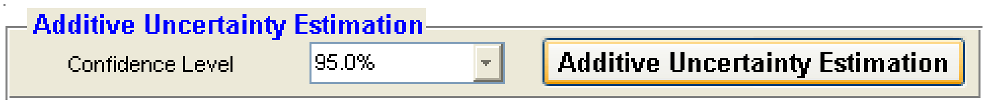
"Output Power Spectrum" provides an output power spectrum using the last cycle from the estimation data.

6.3.4 Additive Uncertainty Estimation

Additive uncertainty estimation is included in the GUI. One can assume that each element p_{ij} in the plant P is independent, but confined to a disk with radius ℓ_{ij} at $\tilde{p}_{ij}$ in the Nyquist plane (Morari and Zafiriou, 1988)

$$|p_{ij} - \tilde{p}_{ij}| \leq \ell_{ij} \tag{12}$$

Therefore, a set of models are defined as they reside within the distance ℓ. For more information for the additive uncertainty description, the user can refer to Lee (2006).

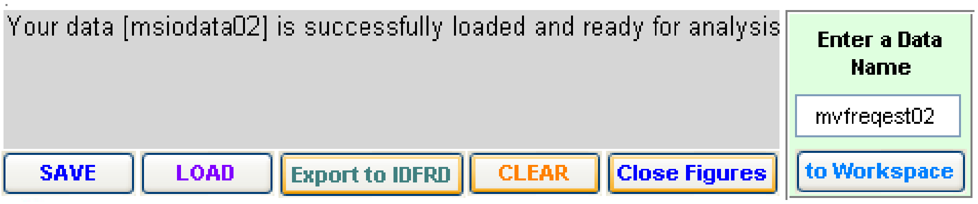

Additive Uncertainty Estimation
Additive Uncertainty Estimation

Confidence Level 95.0%

Figure 20: Additive uncertainty estimation on the Frequency Response Estimation GUI

6.3.5 Action Buttons

The user can manipulate the estimated frequency responses for further analysis using the action buttons, as shown in Figure 21. The estimation data can be in either "IDFRD" or "struct" format for further analysis in the CR-IDENT.



Your data [msiodata02] is successfully loaded and ready for analysis

Enter a Data Name

mvfreqest02

Figure 21: Action buttons on the Frequency Response Estimation GUI

"SAVE" stores the estimation data into a *.mat file.

"LOAD" opens a *.mat data file and uploads it on the GUI.

"Export to IDFRD" transforms the frequency response data into an IDFRD object to the Matlab workspace. The IDFRD object can be used for the control-relevant curvefitting; however, currently it cannot contain the uncertainty description and other information from the GUI.

"toWorkspace" puts the current data information into the Matlab workspace (in "struct" format) with a given variable name.

"CLEAR" empties all the information and data on the GUI.

"Close Figures" closes all the Matlab figures.

6.4 Examples

By the use of the Input Signal Design GUI, two examples of open-loop testing experiment are performed in Matlab Simulink. Their Simulink MDL files are available in the "example" directory. The input/output dataset are uploaded for frequency response estimation by hitting the buttons in Figure 22

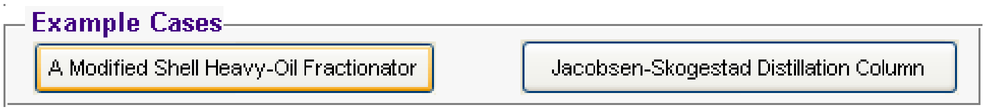

Example Cases
A Modified Shell Heavy-Oil Fractionator
Jacobsen-Skogestad Distillation Column

Figure 22: Example cases on the Frequency Estimation GUI

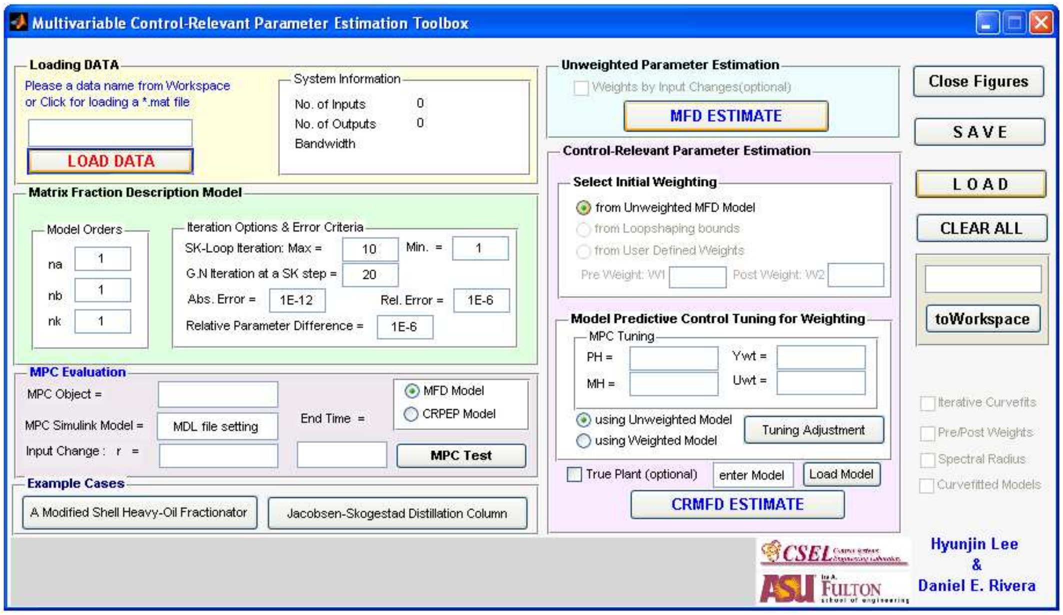
A Modified Shell Heavy-Oil Fractionator provides the two cases using phase-shifted and standard zippered multisine input designs. In addition the model is slightly modified from the Shell Heavy-Oil Fractionator with longer time delays in transfer functions.

Jacobsen-Skogestad Distillation Column provides the three cases using phase-shifted, standard zippered, and modified zippered multisine input designs. In particular, the modified zippered spectrum is considered for the directional adjustment to enhance the low gain directionality in the outputs, meaningful for such a highly interactive systems.

7 Control-Relevant Parameter Estimation GUI

The goal of the control-relevant frequency-response curvefitting GUI (Figure 23) is to optimally estimate a useful parametric model with satisfactory control-relevance which can be used as a nominal for Model Predictive Control (MPC).

The approach used here follows according to the analysis described in Lee and Rivera (2005*a*; 2005*b*). Users can rely on frequency responses from the previous GUI, or import frequency responses in "idfrd" or "struct" format with the proper syntax. The control-relevant curvefitting algorithm approximates frequency-responses into parsimonious discrete-time state-space model representations based on linear Matrix Fractional Descriptions (MFD).



No. of Inputs 0
No. of Outputs 0
Bandwidth

na 1
nb 1
nk 1

SK-Loop Iteration: Max = 10 Min. = 1
G.N Iteration at a SK step = 20
Abs. Error = 1E-12 Rel. Error = 1E-6
Relative Parameter Difference = 1E-6

MPC Object =
MPC Simulink Model = MDL file setting
Input Change : r =
End Time =
● MFD Model ○ CRPEP Model

● from Unweighted MFD Model
○ from Loopshaping bounds
○ from User Defined Weights

PH = Ywt =
MH = Uwt =
● using Unweighted Model
○ using Weighted Model
□ True Plant (optional)

Figure 23: Control-Relevant Parameter Estimation GUI.

The curvefitting relies on frequency-dependent pre- and post- weight functions meaningful to MPC control (Rivera and Gaikwad, 1995). The control-relevant weighting functions systematically shift the model error into regions that are less significant to closed-loop control performance. Sanathanan-Koerner iteration and Gauss-Newton minimization are applied to successively improve the model estimate; robustness criteria such as the Small Gain Theorem can be specified from the GUI to monitor the control-relevancy of an MFD model. If a closed-loop Simulink model is available, users can test the estimated model with respect to setpoint tracking directly from the GUI.

7.1 Entry Data Format for Control-Relevant Parameter Estimation GUI

The necessary components for the entry data structure to the control-relevant curvefitting GUI are given as follows

Components	Description
G_w	Frequency response, in complex valued-form
w	Frequency grids
T_s	Sampling time [in minute]

Table 3: Essential components for Control-Relevant Frequency Response Curvefitting GUI

The components in Table 3 can be contained either “IDFRD”or “struct” format.

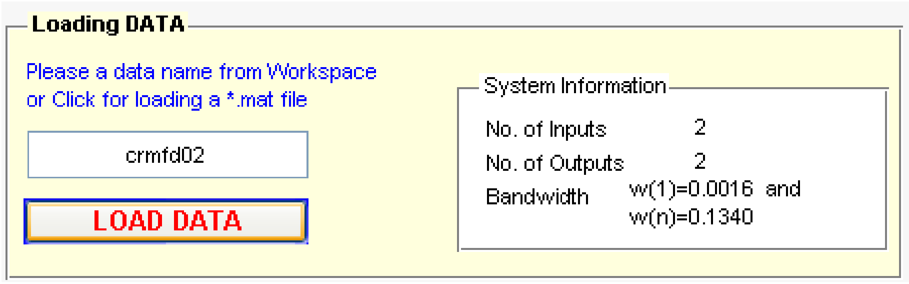
7.2 Getting Started with Control-Relevant Curvefitting

7.2.1 Loading DATA

An estimated data is loaded using the block in Figure 24. The user can load the data from the Matlab workspace or a *.mat file.

From the workspace the user enters a data structure name on the edit block and hit the *“LOAD DATA”* button. If the entered data is not available on the workspace, an error message will appear.

From a *.mat file the user just hits the *“LOAD DATA”* button with the empty edit box. Then, it prompts the file open window for selecting a data file.



Loading DATA

Please a data name from Workspace
or Click for loading a *.mat file

crmfd02

System Information

No. of Inputs 2

No. of Outputs 2

Bandwidth w(1)=0.0016 and w(n)=0.1340

Figure 24: Loading the dataset to the Control-Relevant Curvefitting GUI

Once the dataset is loaded without errors, its information is analyzed and printed in *“System Information”* box, e.g., the number of inputs and outputs and frequency bandwidth range.

7.2.2 Specification for Matrix Fraction Description

Model Orders:

The MFD model consists of a matrix polynomial fraction. A model $\tilde{P}$ is approximated into a linear parametric, real rational transfer function, formed by a left-side fractional matrix polynomial description.

$$\text{Left MFD} \quad \tilde{P}(\xi^{-1},\theta) \;=\; A(\xi^{-1},\theta)^{-1}B(\xi^{-1},\theta) \qquad (13)$$

where A and B denote parameterized polynomial matrices in the indeterminate ξ^{-1}. $\xi(\omega_i) = j\omega_i$ represents a continuous time model, whereas $\xi(\omega_i) = e^{j\omega_i T}$ represents the shift operator. It requires the model orders for the number of matrix polynomial in the B (*"nb"*) and A (*"na"*), respectively. In addition, the time delay (*"nk"*) can be specified.

Iteration Options & Error Criteria:

"SK-Loop Iteration" gives the minimum and maximum iteration numbers for the Sanathanan and Koerner (1963) method that updates the weighting functions utilizing the previous model parameter.

"G.N. Iteration at a SK step" specifies the number of Gauss-Newton error minimization iteration at each SK-step. Extended from the original SK-method, an optimal model parameter set is considered simultaneously with the SK-iteration.

"Abs. Error" sets the absolute weighted error criterion for the SK and GN loop termination.

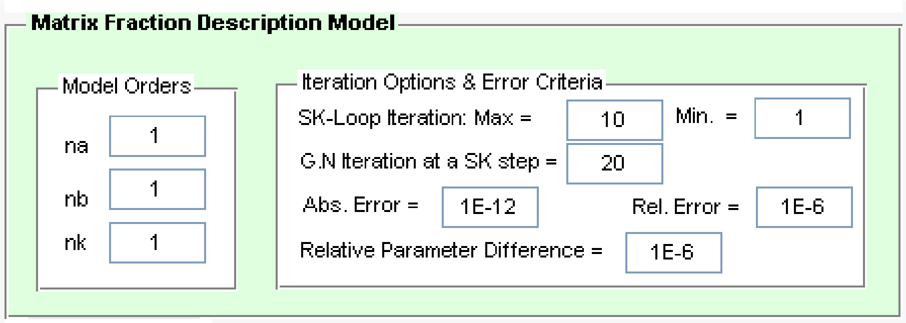
"Rel. Error" sets the relative weighted error criterion for the SK and GN loop termination.

"Relative Parameter Difference" sets the relative parameter difference criterion with the previous and current model parameters for the SK and GN loop termination.

The curvefitting loop terminates only when all three user error specifications are met. If the loop is not numerically converging properly, the curvefitter will terminate abruptly. Curvefitting noisy frequency responses requires a more relaxed set of error criteria to keep it from abrupt termination. Over-parameterization could result in crashes in the loop iteration. On the other hand, the user can try setting some relaxed error criteria while monitoring model performance with the Small Gain Theorem analysis and closed-loop MPC evaluation.

7.2.3 Parameter Estimation by Unweighted Curvefitting

Model estimation error is minimized iteratively without using weighting functions. Currently, the input change weighting is not implemented in the routine. This will considered for the future release version.


Matrix Fraction Description Model
Model Orders
Iteration Options & Error Criteria

na = 1

nb = 1

nk = 1

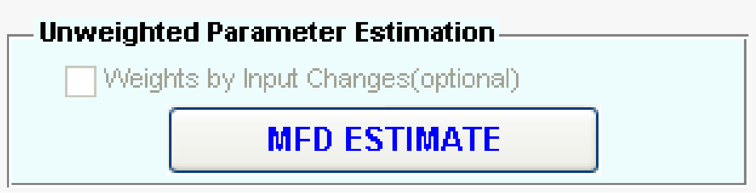
SK-Loop Iteration: Max = 10 Min. = 1

G.N Iteration at a SK step = 20

Abs. Error = 1E-12 Rel. Error = 1E-6

Relative Parameter Difference = 1E-6

Figure 25: Specification options for MFD model on the curvefitting GUI



□ Weights by Input Changes(optional)

Figure 26: Unweighted curvefitting into MFD model

7.2.4 Parameter Estimation by Weighted Curvefitting

The control-relevant weighting is considered for curvefitting the frequency response into a parametric MFD model. In particular, this requires more information than the unweighted curvefitting. The weighted curvefitting needs an initial parameter model to obtain the pre/post weighting functions. Since the closed-loop response is reflected on the weighting functions, the user should provide an appropriate tuning set for MPC.

Select Initial Weighting:
The current release version is only based on the initial model *“from Unweighted MFD Model”* for initial weights. The future release will consider *“from Loopshaping bounds”* and *“from User Defined Weights.”*

Model Predictive Control Tuning for Weighting:
MPC (Model Predictive Control) takes a set of tuning parameters: PH (prediction horizon), MH (moving horizon), Ywt (output weighting), and Uwt (move suppression). The detailed information can referred to the Matlab MPC Toolbox manual. Moreover, the weighting function also needs the *“Input Changes”* which is located in the *“MPC Evaluation”* panel.

“Tuning Adjustment” is a feature that draws the sensitivity and complementary sensitivity

functions using a parametric model and the given MPC tuning set for the user convenience.

"True Plant" is an option for the user to provide the true plant to the curvefitting GUI, in the state-space model or a Matlab transfer function object. If provided, true plant is used for the spectral radius analysis and the comparison with frequency responses and parametric models.

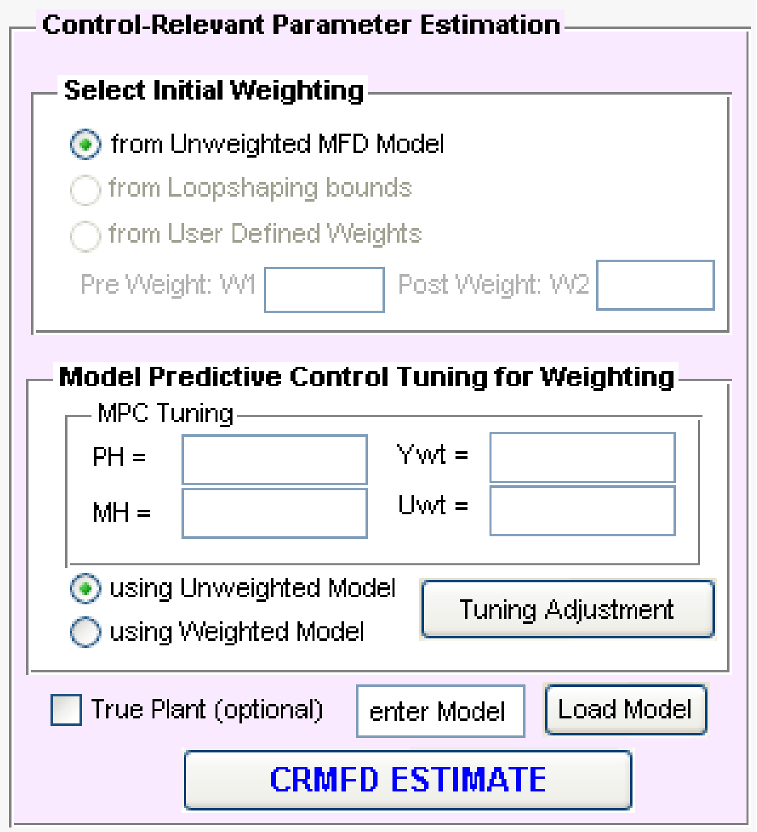
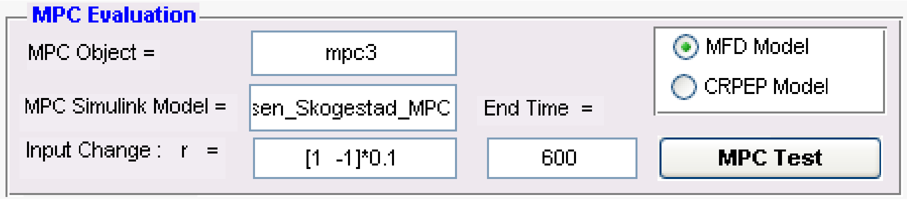


Figure 27: Control-relevant weighted curvefitting into MFD model

7.2.5 MPC Evaluation

Taking advantage of the estimated parametric model, a closed-loop evaluation with the MPC tuning parameters used for the weighting function that can be performed. It requires a Matlab Simulink Model (*.MDL) file in *"MPC Simulink Model"*, and the user is required to assign a name for an MPC object in *"MPC Object."* In addition, the user should specify *"End Time"* in the same unit with the sampling time.

"Input Change" is considered for generating the weighting function and a reference change



MPC Evaluation

MPC Object = mpc3

MPC Simulink Model = sen_Skogestad_MPC

Input Change : r = [1 -1]*0.1

End Time = 600

● MFD Model
○ CRPEP Model

Figure 28: Preparation for MPC closed-loop evaluation

in the closed-loop evaluation with MPC. For example, it can be given for a two-output system as

$$r = [1 \quad -1]$$

7.2.6 Action Buttons

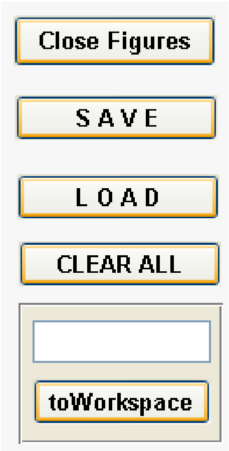


Figure 29: Action buttons on the Curvefitting GUI

"Close Figures" closes all the Matlab figures.

"SAVE" stores all the information from the GUI into a file.

"LOAD" opens a file and uploads it onto the GUI.

"CLEAR ALL" empties all the information on the GUI.

"toWorkspace" exports all the information to the Matlab workspace.

7.3 Examples

By the use of the Input Signal Design and Frequency Estimation GUIs, two example cases for the curvefitting procedure available directly from the GUI. The frequency responses can be uploaded by hitting the buttons in Figure 30

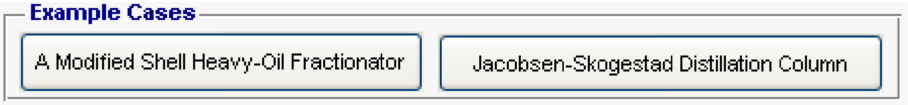

Example Cases
A Modified Shell Heavy-Oil Fractionator
Jacobsen-Skogestad Distillation Column

Figure 30: Illustrative example cases on the curvefitting GUI

A Modified Shell Heavy-Oil Fractionator provides the two cases using phase-shifted and standard zippered multisine input designs. In addition, the model is slightly modified from the Shell Heavy-Oil Fractionator with longer time delays in transfer functions.

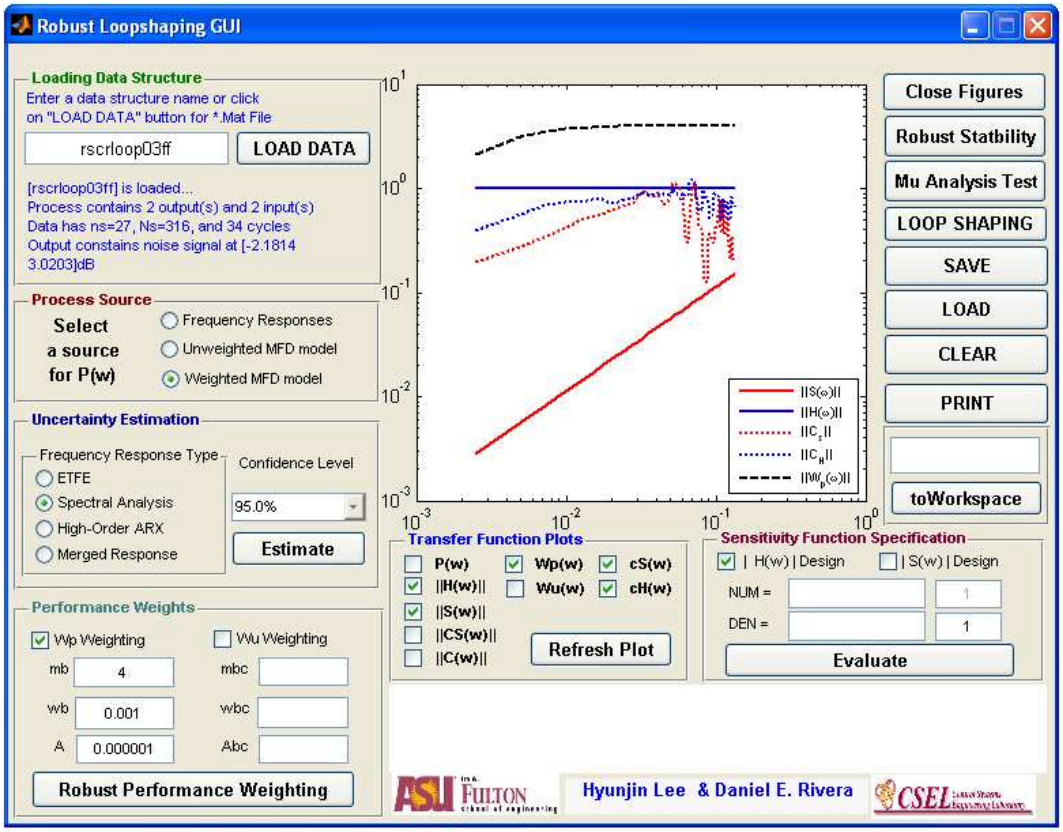
Jacobsen-Skogestad Distillation Column provides the three cases using phase-shifted, standard zippered, and modified zippered multisine input designs. In particular, the modified zippered spectrum is considered for the directional adjustment to enhance the low gain direction in the distillation column.

The frequency responses are estimated based on the ETFE for a standard zippered spectrum signal and Spectral Analysis for phase-shifted and modified zippered spectrum signals.

8 Robust Loopshaping GUI

The goal of the Robust Loopshaping GUI is to design a set of models that can meet the requirements of robust stability and performance. Figure 31 shows the necessary components for this robust loopshaping procedure. The robustness bounds for the sensitivity and complementary sensitivity functions involve performance weight(s) and uncertainty bound. Those conditions of robust stability and performance are computed using Structured Singular Value analysis.

For successful robust control system design, it is necessary to iterate between control-relevant parameter estimation and robust loopshaping; this follows a procedure we call identification test monitoring (Rivera *et al.*, 2003). Initially, frequency responses with uncertainty bounds are considered to compute robust loop bounds that are used as a preliminary guideline for controller tuning in the control-relevant curvefitting. The evaluation of



rscrloop03ff

[rscrloop03ff] is loaded...
Process contains 2 output(s) and 2 input(s)
Data has ns=27, Ns=316, and 34 cycles
Output constains noise signal at [-2.1814 3.0203]dB

○ Frequency Responses
○ Unweighted MFD model
● Weighted MFD model

○ ETFE
● Spectral Analysis
○ High-Order ARX
○ Merged Response

95.0%

☑ Wp Weighting
mb 4
wb 0.001
A 0.000001

□ Wu Weighting
mbc
wbc
Abc

□ P(w) ☑ Wp(w) ☑ cS(w)
☑ ||H(w)|| □ Wu(w) ☑ cH(w)
☑ ||S(w)||
□ ||CS(w)||
□ ||C(w)||

☑ | H(w) | Design □ | S(w) | Design
NUM = 1
DEN = 1

Figure 31: Robust Loopshaping GUI.

a parametric model is performed using robust loopshaping considering both uncertainty bounds and model estimation error. If the model satisfies the loop bounds, the procedure can be terminated; otherwise, the user needs to change performance weights and otherwise iterate on the control-relevant parameter estimation and robust loopshaping procedures.

8.1 Entry Data Format for Robust Loopshaping GUI

The Robust Loopshaping GUI supports data formats from the Frequency Response Estimation GUI and the Control-Relevant Parameter Estimation GUI. Specifically, the Robust Loopshaping GUI needs input and output data, frequency responses, or a parametric model to evaluate robustness conditions. In addition, the data structure from the Robust Loopshaping GUI can work with the Control-Relevant Parameter Estimation GUI supporting iterative procedure during identification test monitoring.

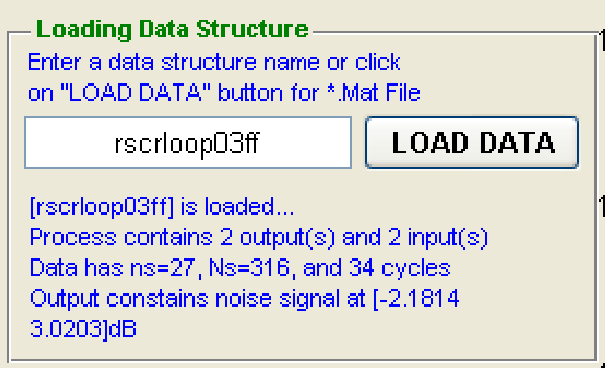
8.2 Getting Started with Robust Loopshaping

8.2.1 Loading DATA

A dataset from Frequency Response Estimation GUI and Control-Relevant Parameter Estimation GUI is loaded using the block in Figure 32. The user can load the data from the Matlab workspace or a *.mat file.

From the workspace the user enters a data structure name on the edit box and hit the *"LOAD DATA"* button. If the entered data is not available on the workspace, an error message will appear.

From a *.mat file the user just hits the *"LOAD DATA"* button with the empty edit box. Then, it prompts the file open window for selecting a data file.



Enter a data structure name or click on "LOAD DATA" button for *.Mat File

rscrloop03ff

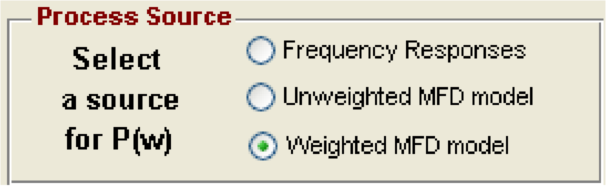
[rscrloop03ff] is loaded...
Process contains 2 output(s) and 2 input(s)
Data has ns=27, Ns=316, and 34 cycles
Output constains noise signal at [-2.1814 3.0203]dB

Figure 32: Loading the dataset to the Robust Loopshaping GUI

Once the dataset is loaded without errors, data information is analyzed and printed in the box, e.g., the number of inputs and outputs, the number of sinusoids, the sequence length, and the sampling time. If available, it indicates the signal-to-noise ratio in dB.

8.2.2 Process Source

The Robust Loopshaping GUI computes robust loop bounds based on frequency responses or a parametric models. The user can select a process source in Figure 33. If a parametric MFD model is selected, the loopshaping procedure will consider model estimation error in addition to uncertainty bounds.



Process Source

Select a source for P(w)

○ Frequency Responses

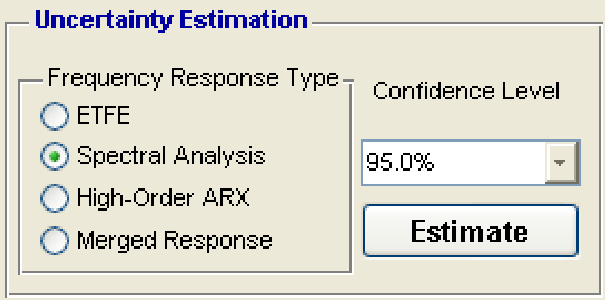
○ Unweighted MFD model

● Weighted MFD model

Figure 33: Process Source for the Robust Loopshaping GUI

8.2.3 Uncertainty Estimation

Uncertainty estimation relies on the same procedure implemented in the Frequency Response Estimation GUI. The user may change the confidence level during the iterative procedure of designing a robust control system.



Uncertainty Estimation

Frequency Response Type

○ ETFE

● Spectral Analysis

○ High-Order ARX

○ Merged Response

Confidence Level: 95.0%

Figure 34: Uncertainty Estimation on the Robust Loopshaping GUI

8.2.4 Performance Weights

The two performance weights are considered for the robust control system: (1) W_p performance weight (Figure 35) on the sensitivity function $\tilde{S}$ such that

$$||W_p \, \tilde{S}|| \;\; < 1 \tag{14}$$

$$W_p = \frac{s/M_s + \omega_b}{s + \omega_b \varepsilon} \tag{15}$$

where M_s is the peak sensitivity, ω_b is the bandwidth, and ε is given as $|\tilde{S}(0)| \leq \varepsilon$, and (2) W_u on $\tilde{P}^{-1}\tilde{H} = C\tilde{S}$ (Figure 36) is given as

$$||W_u \, \tilde{P}^{-1}\tilde{H}|| \;\; < 1 \tag{16}$$

$$W_u = \frac{s + \omega_{bc}/M_u}{\varepsilon_1 \; s + \omega_{bc}} \tag{17}$$

where M_u is the maximum gain of $C\tilde{S}$, ω_{bc} is the controller bandwidth, and $\varepsilon_1 \geq |C\tilde{S}|$ at high frequencies. The parameters for W_p and W_u can be specified on Figure 37.

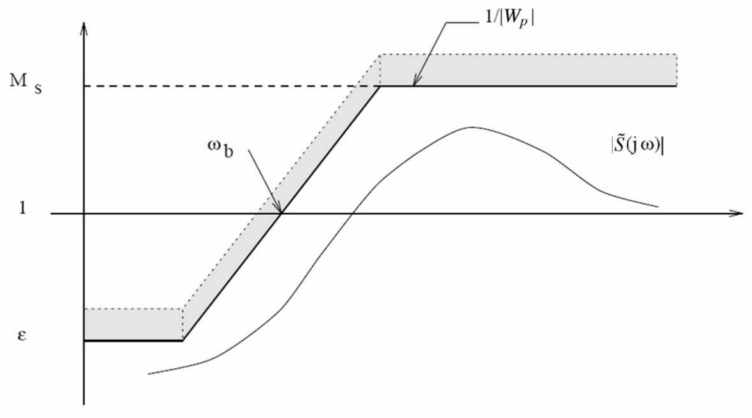


Figure 35: Performance weight W_p on $\tilde{S}$

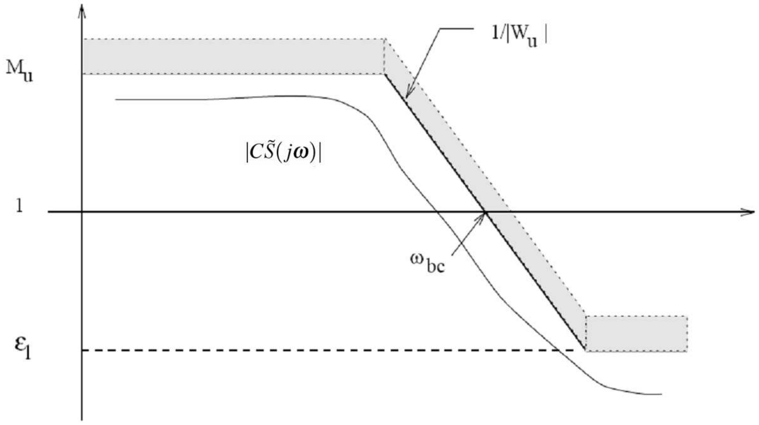


Figure 36: Performance weight W_u on $C\tilde{S}$

8.2.5 Transfer Function Plots

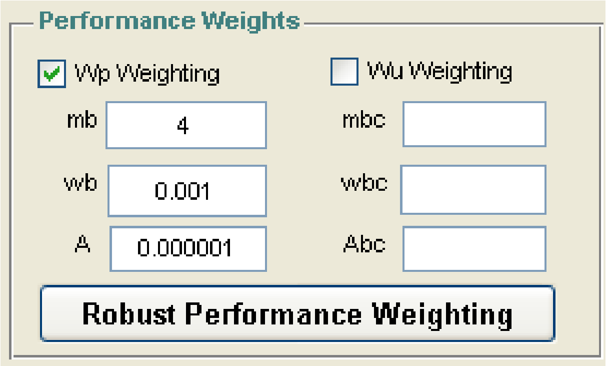
Robust loopshaping involves a number of transfer functions and the user can plot them on the GUI. Select check boxes of transfer functions and hit “Refresh Plot” button on Figure 38, it will update the figure window, as shown in Figure 39.

8.2.6 Action Buttons

“Close Figures” closes all the Matlab figures.

“Robust Stability” analyzes robust stability on a set of models, i.e., $\mu(M_{11})$.

“Mu Analysis Test” performs μ-analysis test on a set of models, i.e., $\mu(M)$.



☑ Wp Weighting

mb: 4

wb: 0.001

A: 0.000001

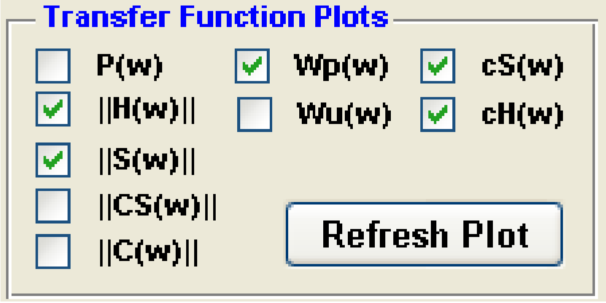
□ Wu Weighting

mbc:

wbc:

Abc:

Figure 37: Performance weight specification on the Robust Loopshaping GUI



□ P(w)
☑ ||H(w)||
☑ ||S(w)||
□ ||CS(w)||
□ ||C(w)||
☑ Wp(w)
□ Wu(w)
☑ cS(w)
☑ cH(w)

Figure 38: Plotting options on the Robust Loopshaping GUI

"LOOP SHAPING" computes robust loop bounds taking advantage of μ-analysis, i.e., $\bar{\sigma}(\tilde{S})$ and $\bar{\sigma}(\tilde{H})$.

"SAVE" stores all the information from the GUI into a file.

"LOAD" opens a file and uploads it onto the GUI.

"CLEAR" empties all the information on the GUI.

"toWorkspace" exports all the information to the Matlab workspace.

8.2.7 Sensitivity Function Specification

With computed robust loop bounds, the user can optionally specify a nominal model that can meet robustness conditions. The specified nominal model and computed loop bounds are used a guideline for controller tuning in control-relevant curvefitting.

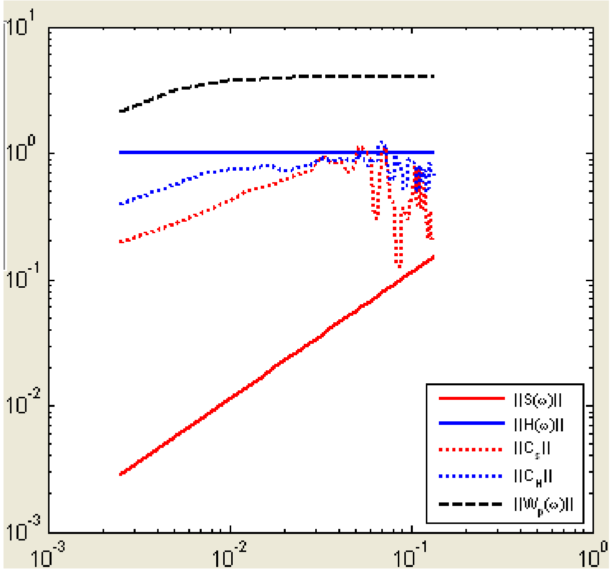


Figure 39: Plotting window on the Robust Loopshaping GUI

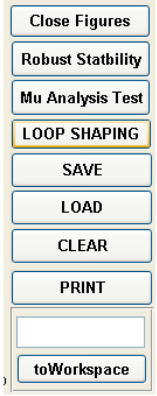
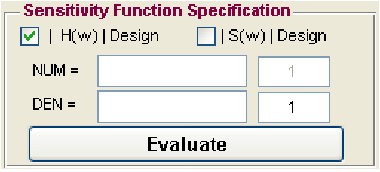


Figure 40: Action buttons on the Robust Loopshaping GUI

9 Illustrative Examples of CR-IDENT

CR-IDENT includes two built-in illustrative example cases to help the user with appropriate parameter specification on the GUIs. In this section, we demonstrate step-by-step procedures for using CR-IDENT on two example cases: a modified version of the Shell Heavy-Oil Fractionator Problem (Prett and García, 1988) and Jacobsen-Skogestad distillation column (Jacobsen and Skogestad, 1994). The results and discussion for the Jacobsen-Skogestad distillation column are presented later in this section.



Sensitivity Function Specification

☑ | H(w) | Design □ | S(w) | Design

NUM = 1

DEN = 1

Figure 41: Nominal model specification on the Robust Loopshaping GUI

9.1 Illustration with Shell Heavy-Oil Fractionator Problem

1. Open the Multivariable Input Signal Design GUI

2. Hit *“Heavy-Oil Fractionator”* button, and it brings the preset parameters on the GUI. The user may change some design options or multisine signal type.

3. Choose *“Time Series”*, *“Power Spectral Density”*, and *“Auto/Cross Correlation”*.

4. Hit *“Analyze/Plot”* button.

5. Put “test01” on the edit box above the *“toWorkspace”* button.

6. Hit *“toWorkspace”* button.

7. Go to the Matlab command window. Run **“run_shell_idtest”** and follow the instructions from the program such that

```
Enter Multisine Input Data Structure (e.g., contains ###.u): test01

Please make input signals as us[i,:], i=channel index

Please enter a number of cycles for running [nc=3]: 21

Please enter a name for saving Input/Output data to a structure: idtest01

idtest01 =
          u: [7000x2 double]
          y: [7000x2 double]
         Ts: 4
         Ns: 350
         ns: 10
    ncycles: 21
```

   with “vard=1.0” for the noise option on the popup menu selection. Lowering the value of vard will require less input signal cycles for successful estimation. The user may change noise realization inside the script program of “run_shell_idtest.m”

8. Open the Frequency Response Estimation GUI

9. Put “idtest01” on the data load box and Hit *“LOAD DATA”* button.

10. Select *“Spectral Analysis”* since the input signal is designed by the phase-shifting method. Put “256” in the *“Lag Window”* box.

11. Select *“Response Amplitudes”* and Hit *“ESTIMATE”* button (the user can choose different options).

12. Put “fresptest01” on the edit box in the *“toWorkspace”* box and Hit the button.

13. Open the Control-Relevant Curvefitting GUI

14. Before loading “fresptest01” on the GUI, please Hit *“Heavy-Oil Fractionator”* and Choose “Phase-Shifted Case” because by doing this the user can take advantage of preset curvefitting options for “A Modified Shell Heavy-Oil Fractionator” case on the GUI. Then proceed to load “fresptest01’.’

15. Hit *“MFD ESTIMATE”* button for the unweighted curvefitting. If it is successful, the user can run the closed-loop evaluation with MPC by hitting the *“MPC Test”* button with *“MFD Model”* selected. The user may change the parameters in *“Matrix Fraction Description Method”* to improve the numerical convergence for the loop iteration. Model orders are [na=1, nb=1, nk=1], and Iteration Options & Error Criteria are “SK-Loop Iteration: Max=30, Min=2”, “GN Iteration at a SK step=50”, “Abs. Error=1e-12”, “Rel. Error=0.0001”, “Relative Parameter Difference=1e-10.”

16. Hit *“CRMFD ESTIMATE”* button for the weighted curvefitting. If it is successful, the user can run the closed-loop evaluation with MPC by hitting the *“MPC Test”* button with *“CRPEP Model”* selected. The user may change the parameters in *“Matrix Fraction Description Method”* to improve the numerical convergence for the loop iteration. To use the true model during the curvefitting, hit “Load Model” with putting “shell_ss” in the edit box and select “True Plant(optional).” For MPC-relevant weights, “PH=35”, “MH=10”, “Ywt=[1 1]”, and “Uwt=[15 22]*1.605.” MPC Evaluation is set as “MPC Object=mpc3”, “MPC Simulink Model=Shell_HOFP_MPC”, “Input Change: r=[0.1 -0.1]”, and “End Time=1500.”

17. The use may iterate the last two steps until the estimated model shows desirable performance in the closed-loop evaluation. Otherwise, the user may need to refine the frequency responses or to redo the input signal design and testing experiment.

9.2 Illustration with Jacobsen-Skogestad Distillation Column

Utilizing the step-wise procedures given in the previous section, a modified zippered spectrum input design is utilized for the Jacobsen-Skogestad Distillation Column. The open-loop Matlab Simulink simulation is provided at “run_jacobsen_skogestad_idtest” in the “Example” directory.

1. Open the Multivariable Input Signal Design GUI
2. Hit *“Jacobsen-Skogestad Distillation Column”* button, and it brings the preset parameters on the GUI. The user may change some design options or multisine signal type.
3. Choose *“Time Series”*, *“Power Spectral Density”*, and *“Auto/Cross Correlation”*.
4. Hit *“Analyze/Plot”* button.
5. Put “test02” on the edit box above the *“toWorkspace”* button.
6. Hit *“toWorkspace”* button.
7. Go to the Matlab command window. Run **“run_jacobsen_skogestad_idtest”** and follow the instructions from the program such that

```
Enter Multisine Input Data Structure (e.g., contains ###.u): test02

Defining the Full-Order High-Purity Column...

MULTIVARIABLE IDENTIFICATION OF THE HIGH-PURITY DISTILLATION COLUMN

The 82-state high-purity column is described as:

y(t) = [P_LV(s)] u(t) + [P_F(s)] d(t)

where
        |dyD|             |dL |
y(t) =  |   | ;    u(t) = |   | ;      d(t)=dF
        |dxB|             |dV |

Enter the variance of noise added to Output 1(0=no noise) [4.7e-005]:
Enter the variance of noise added to Output 2(0=no noise) [8.7e-005]:

Please make input signals as us[i,:], i=channel index

Please enter a number of cycles for running [nc=3]:6

Please enter a name for saving Input/Output data to a structure: idtest02
```

```
idtest02 =
          u: [4580x2 double]
          y: [4580x2 double]
         Ts: 8
         Ns: 916
         ns: 78
    ncycles: 6
```

Inside the script program “run_jacobsen_skogestad_idtest.m”, the user may change noise realization.

8. Open the Frequency Response Estimation GUI
9. Put “idtest02” on the data load box and Hit *“LOAD DATA”* button.
10. Select *“Spectral Analysis”* since the input signal is designed by a modified zippered power spectrum design. Put “256” or “512” in the *“Window Lag”* box.
11. Select *“Response Amplitudes”* and Hit *“ESTIMATE”* button (the user can choose different options).
12. Put “fresptest02” on the edit box in the *“toWorkspace”* box and Hit the button.
13. Open the Robust Loopshaping GUI
14. Put “fresptest02” on the data load box and Hit *“LOAD DATA”* button.
15. Choose “Frequency Responses” on the *“Process Source”*.
16. Select a “Confidence Level” for the *Uncertainty Estimation* and then press the “Estimate” button.
17. Select a *“Performance Weight”* and put appropriate parameters. Once such case is “W_p” with $mb = 4$, $wb = 0.001$, and $A = 10^{-6}$.
18. Hit the *“LOOP SHAPING”* and it computes the robustness bounds with respect to the nonparametric frequency responses.
19. Write “rsloop02” in the edit box and Hit the *“toWorkspace”*.
20. Open the Control-Relevant Curvefitting GUI
21. Before loading “rsloop02” on the GUI, please Hit *“Jacobsen-Skogestad Distillation Column”* and Choose “Modified Zippered Case” because by doing this the user can take advantage of preset curvefitting options for “Jacobsen-Skogestad Distillation Column” case on the GUI.

22. Hit *"MFD ESTIMATE"* button for the unweighted curvefitting. If it is successful, the user can run the closed-loop evaluation with MPC by hitting the *"MPC Test"* button with *"MFD Model"* selected. The user may change the parameters in *"Matrix Fraction Description Method"* to improve the numerical convergence for the loop iteration. Model orders are [na=1, nb=1, nk=1], and Iteration Options & Error Criteria are "SK-Loop Iteration: Max=20, Min=2", "GN Iteration at a SK step=100", "Abs. Error=1e-12", "Rel. Error=0.001", "Relatvie Parameter Difference=0.001."

23. Hit *"CRMFD ESTIMATE"* button for the weighted curvefitting. If it is successful, the user can run the closed-loop evaluation with MPC by hitting the *"MPC Test"* button with *"CRPEP Model"* selected. The user may change the parameters in *"Matrix Fraction Description Method"* to improve the numerical convergence for the loop iteration. To use the true model during the curvefitting, hit "Load Model" with putting "jacob_ss" in the edit box and select "True Plant(optional)." For MPC-relevant weights, "PH=35", "MH=5", "Ywt=[1 1]", and "Uwt=[0.05 0.03]*0.13." MPC Evaluation is set as "MPC Object=mpc3",
"MPC Simulink Model=Jacobsen_Skogestad_MPC", "Input Change: r=[0.1 -0.1]", and "End Time=500."

24. The user may iterate the last two steps until the estimated model satisfies the robustness conditions and shows desirable nominal performance in the closed-loop evaluation.

25. Once the weighted model is obtained, repeat the procedure with the Robust Loopshaping GUI and compute the robustness conditions based on the parametric model. For a weighting fuction in the loopshaping, "W_p" is used with $mb = 4$, $wb = 0.001$, and $A = 10^{-6}$, and "Confidence Level" is 95%.

26. When new robustness condition bounds are computed, repeat the procedure in the Control-Relevant Curvefitting GUI. Find a controller tuning parameter set that can satisfy robustness conditions. If a weighted model satisfies robust conditions in both Control-Relevant Curvefitting and Robust Loopshaping GUIs, the user can stop.

27. At last, the obtained set of models based on the weighted model and uncertainty bounds may possess sufficient model adequacy for robust control system.

9.3 Example Study of Jacobsen-Skogestad Distillation Column

The Jacobsen-Skogestad column represents a highly interactive system, and as a result a directional input signal design is considered to achieve balanced gain directionality in the data. From *a priori* model information, a steady-state gain is obtained such that

$$K = \begin{bmatrix} 0.785 & -0.771 \\ 0.966 & -0.978 \end{bmatrix} \tag{18}$$

and SVD analysis on K gives

$$U = \begin{bmatrix} -0.6249 & -0.7807 \\ -0.7807 & 0.6249 \end{bmatrix} \Sigma = \begin{bmatrix} 1.7609 & 0 \\ 0 & 0.0130 \end{bmatrix} V^H = \begin{bmatrix} -0.7072 & -0.7070 \\ 0.7070 & -0.7072 \end{bmatrix} \tag{19}$$

A γ range is obtained using the above information as

$$\gamma_{min} = 54.25 \leq \gamma \leq \gamma_{max} = 84.65 \tag{20}$$

As a result, $\gamma_{avg} = 69.45$ and $v_2 = [-0.7070 \ -0.7072]$ are selected for designing a modified zippered power spectrum, as shown in Figure 42a. The input magnitude is set to $u = [\pm 14 \ \pm 14]$. The high magnitude helps in reducing the uncertainty norm bound by applying a higher signal-to-noise ratio, as indicated by the following equation

$$\bar{\ell}_a \propto \frac{n}{N} \frac{\Phi_v}{\Phi_u} \tag{21}$$

where n is the number of model parameters and N is the data length. The nature of the distillation process requires information at the lower frequencies for accurate gain directionality of steady-state, while the closed-loop dynamics requires information at the higher frequencies. Considering such characteristics, we can decide on design parameters $\tau^H_{dom} = 67$, $\tau^L_{dom} = 15$, $\alpha = 2$, and $\beta = 3$ that lead to a much shorter bandwidth, i.e., a sequence length $N_s = 316$.

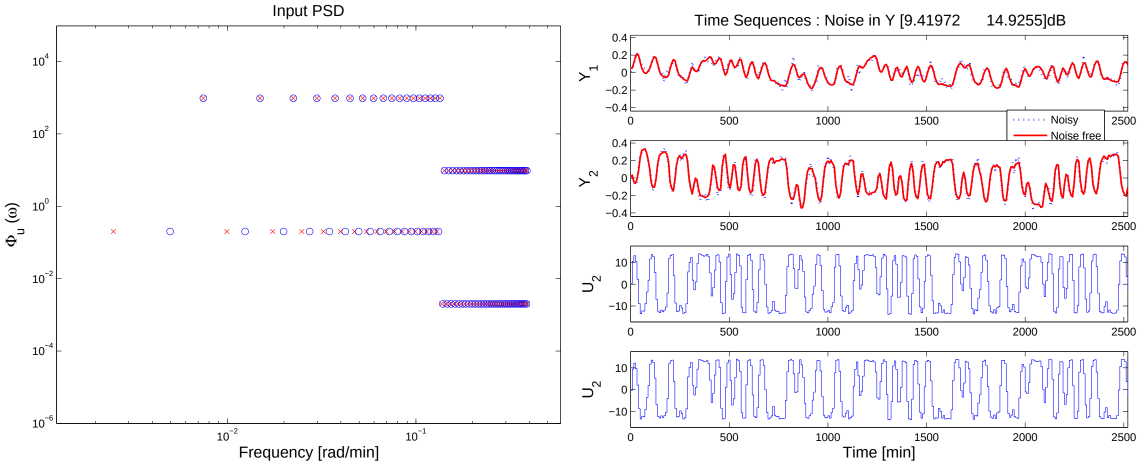
The higher correlated harmonics increase the corresponding output span in the [1 -1] direction (see Figure 43), which ultimately enhances the low gain directionality. Spectral analysis is applied to compute nonparametric frequency responses ($\tilde{P}$) using a Hamming window with 128 lags. An uncertainty description bound is estimated to define a set of models, as shown in Figure 45. As a consequence, robust loopshaping can be applied.

In this experiment, robust stability is satisfied over all frequencies, as shown in Figure 46. Robust stability and performance conditions from robust loopshaping can be used to assess control adequacy of the models estimated from the data, and hence can influence the decision to start or stop experimental testing. The experiment is tested for 14 cycles to reduce the norm bound of model uncertainty to a suitable level for robust loopshaping analysis. Initially, robust loopshaping on $\tilde{P}$ is performed, providing provides $\bar{\sigma}(\tilde{H})$ and $\bar{\sigma}(\tilde{S})$ as a preliminary tuning guideline to properly tune the MPC controller in control-relevant curvefitting that are selected to satisfy $\bar{\sigma}(\tilde{H})$ and $\bar{\sigma}(\tilde{S})$ bounds (Figure 44a). Relying on previously weighted models, we can iteratively tune the weighted curvefitting algorithm until a weighted model meets the loopshaping bounds. When a weighted model satisfies the loopshaping bounds, its Small Gain condition shows sufficient levels of nominal stability, as seen in Figure 44b. The weighted model $(\hat{P})$ is then taken into account again for robust loopshaping including the model error $\tilde{P}-\hat{P}$. The model estimation error and model uncertainty are incorporated into $N^{\hat{H}}$ and $N^{\hat{S}}$ structures for loopshaping. As the bounds of $\bar{\sigma}(\hat{H})$ and $\bar{\sigma}(\hat{S})$ are satisfied with the weighted model $\hat{P}$, a finalized model can be obtained that meets both control relevance and robustness requirements with an MPC tuning set of PH=35, MH=5, Ywt=[1 1], and Uwt=[0.05 0.03]*0.13 (Figure 47).

A closed-loop model evaluation of a satisfactory curvefit model using setpoint tracking is shown in Figure 48. With a faster set of MPC tuning parameters, only the weighted model shows fast and over-damped responses without offset; the unweighted model suffers numerical oscillations in y_2 and overshoot in y_1. A synergistic effect of using the directional input signal design and weighted curvefitting clearly results in the accurate estimate of the low singular values (Figure 49). In spite of the large number of testing cycles, the unweighted curvefitted model does not show a reasonable estimate of the low singular values. This is clear evidence that the weighted curvefitting is effectively able to efficiently capture the low gain directional information which is designed to be emphasized by the use of a modified zippered spectrum.

The performance weight and tuning parameters can provide flexibility to adjust the robust loop bounds during the iterative procedure between the curvefitting and loopshaping. The success of control-relevant and robust models can be guaranteed when a sufficient number of testing cycles are tested so that the uncertainty norm bound is reduced to a suitable level. A control-relevant model satisfying robustness conditions is obtained after 14 cycles, at which time experimental testing is halted.

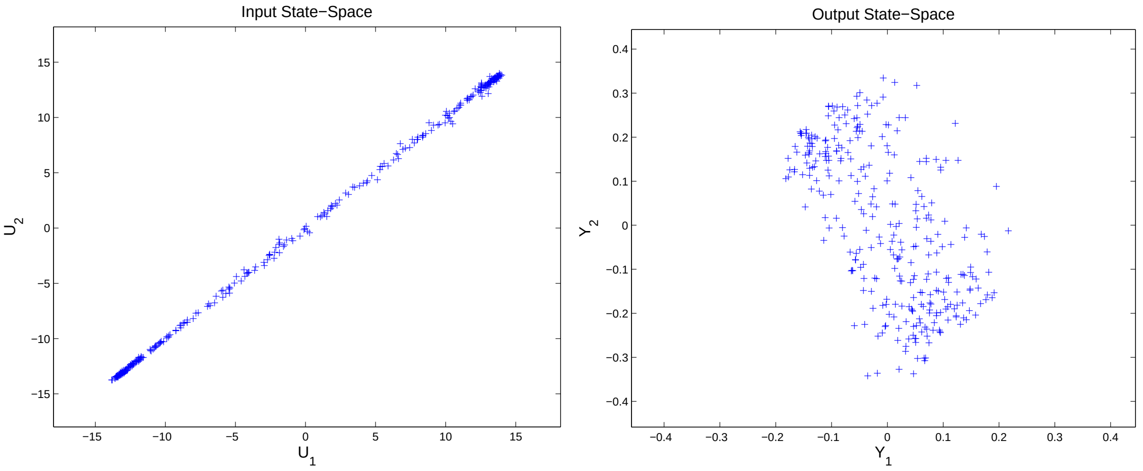
At each cycle interval, the identification test monitoring framework evaluates updated robust loopshaping bounds and whether these are applicable for weighted curvefitting; if so, we can halt identification testing. Figures 51 to 52 show the intermediate identification test monitoring analysis at 5 and 10 cycles, respectively. Their Small Gain conditions



(a) Input PSD

(b) Time Sequences

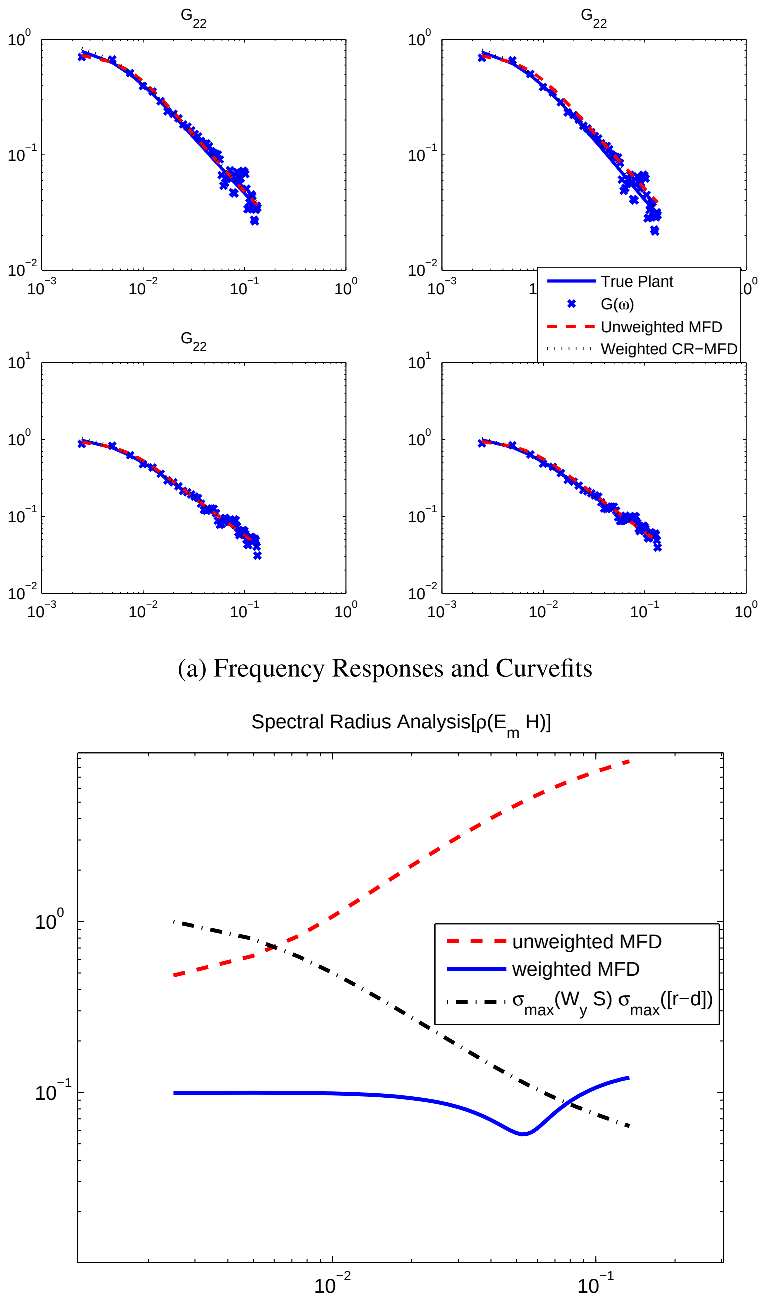
Figure 42: Experiment of identification test monitoring for the Jacobsen-Skogestad high purity distillation column: a standard zippered power spectrum design (a) and time series of input and output data (b)



(a) Input State-Space

(b) Output State-Space

Figure 43: Experiment of identification test monitoring for the Jacobsen-Skogestad high purity distillation column: input (a) and output (b) state-space plots



(a) Frequency Responses and Curvefits

(b) Small Gain Condition $\rho(E_m\hat{H})$

Figure 44: Experiment of identification test monitoring for the Jacobsen-Skogestad high purity distillation column:: frequency responses and curvefitting (a) with MPC tuning set: PH=35, MH=5, Ywt=[1 1], and Uwt=[0.05 0.03]*0.13 (a) and Small Gain condition for the unweighted and weighted curvefit models (b)

are all under the unity (around 10^{-1}) over the bandwidth (Figures 51a to 52a), which is much lower than those of unweighted models. With the increasing number of cycles, the levels of uncertainty bound are lower and have very similar values at 10 and 14 cycles (see Figure 50). The robust loopshaping bounds are changed dynamically along testing cycles. $\bar{\sigma}(H)$ is always lower than the unity in this case; therefore, $\bar{\sigma}(S)$ is only considered for applicable robustness bound in the identification test monitoring procedure. $\bar{\sigma}(S)$ after 5 cycles is only available for the high frequencies (Figure 51b). After 10 cycles, $\bar{\sigma}(S)$ is computed for a wider range but is partly missing in the low frequencies (Figures 52b). All the low singular value plots show good estimation to the true plant (Figures 51d to 52d). Robust stability analysis is improved with increasing test cycles from 5 to 10 cycles (see Figures 53 and 54).

10 Remarks and Conclusions

CR-IDENT represents a software implementation of a comprehensive control-relevant identification methodology that is motivated by the needs and requirements of process systems, particularly strongly interactive ones such as high purity distillation. The toolbox reduces the background and skill level required to implement this procedure, since only *a priori* knowledge of a system in terms of time constants and steady-state gains (if available) is required to initiate this toolbox. Following identification testing, the frequency-response estimation and control-relevant curvefitting modules work interactively (with necessary iterations between the GUIs) to produce a useful model leading to a high performance Model Predictive Controller. This was demonstrated for a strongly interactive distillation column simulation. As a result, CR-IDENT is able to provide an integrated framework that is highly demanded to achieve a systematic identification test monitoring procedure. The most updated CR-IDENT toolbox and its documents can be accessed from the ASU-CSEL website using the link: **http://www.fulton.asu.edu/˜csel/Software.html**.

11 Acknowledgements

Funding for this work was provided by grant number PRF-37610-AC9 from the American Chemical Society-Petroleum Research Fund, and general support from the Honeywell Foundation to the Control Systems Engineering Laboratory.

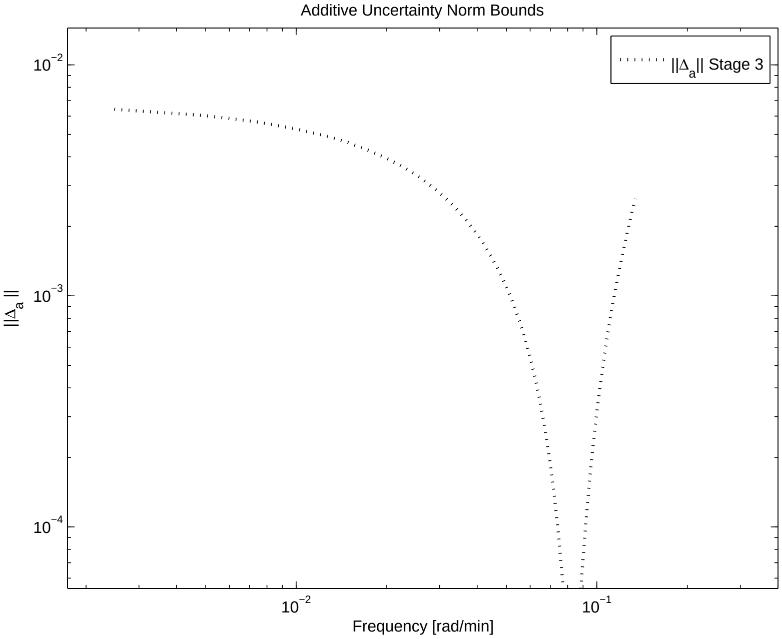


Figure 45: Experiment of identification test monitoring for the Jacobsen-Skogestad high purity distillation column: additive model uncertainty norm bound with 35 cycles

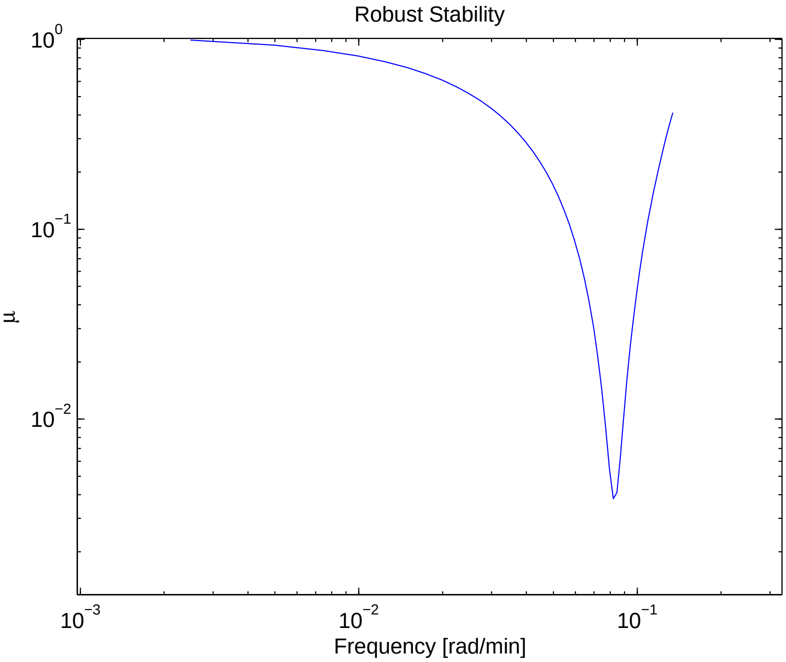


Figure 46: Experiment (at 14 cycles) of identification test monitoring for the Jacobsen-Skogestad high purity distillation column: robust stability analysis

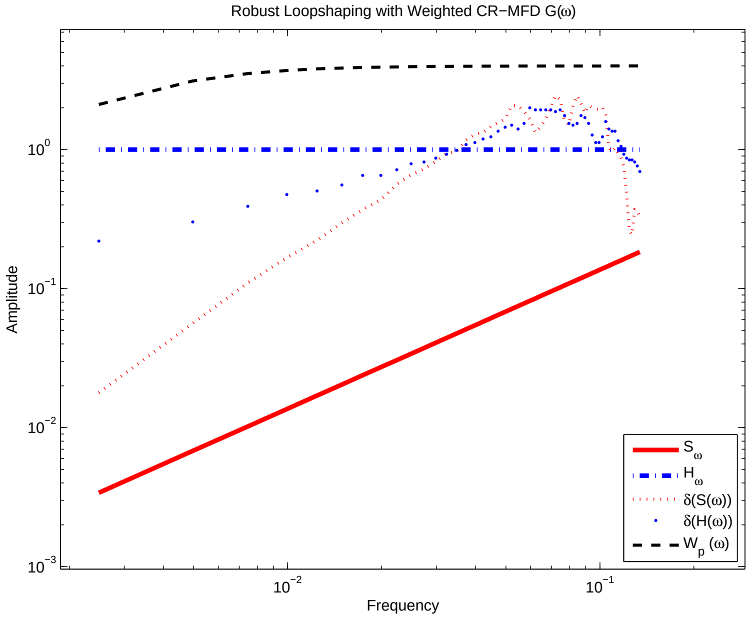


Figure 47: Experiment of identification test monitoring for the Jacobsen-Skogestad high purity distillation column: robust loopshaping bounds on $\hat{P}$ with MPC tuning set: PH=35, MH=5, Ywt=[1 1], and Uwt=[0.05 0.03]*0.13

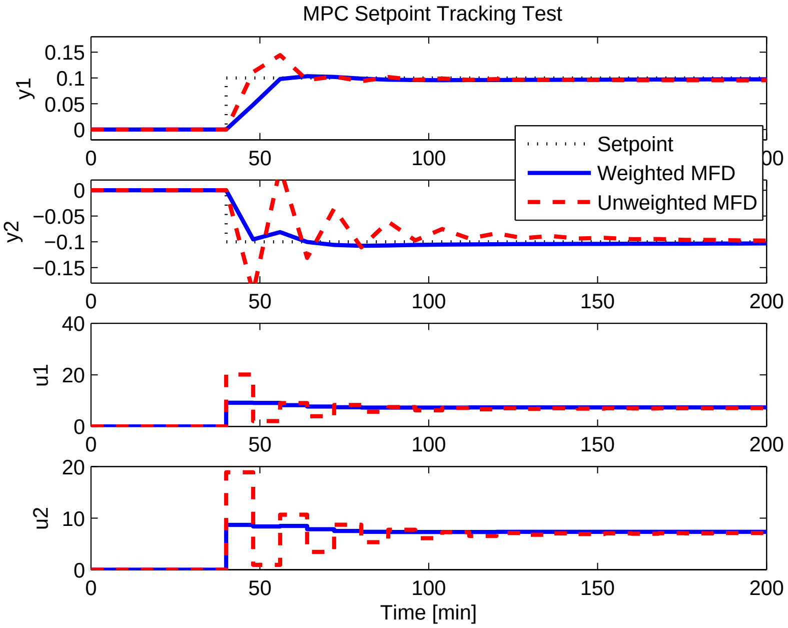


Figure 48: Experiment of identification test monitoring for the Jacobsen-Skogestad high purity distillation column: setpoint (r=[0.1 -0.1]) tracking test with MPC, MPC tuning set: PH=35, MH=5, Ywt=[1 1], and Uwt=[0.05 0.03]*0.13

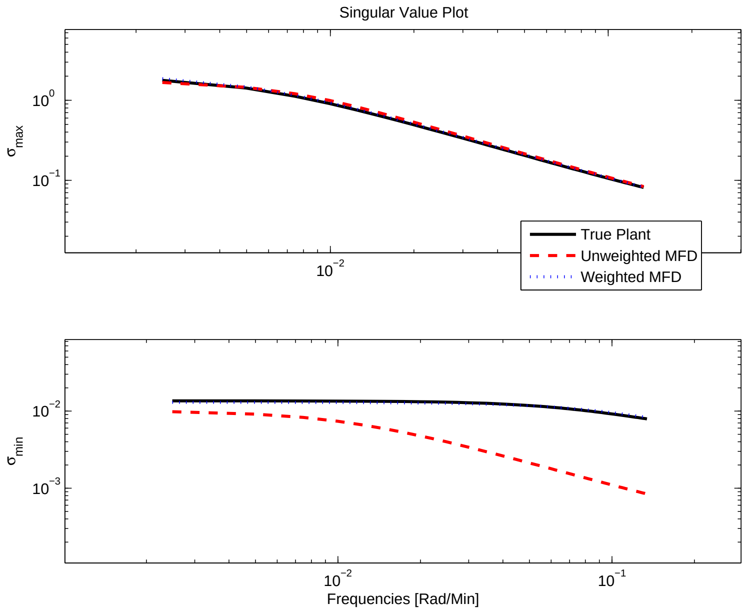


Figure 49: Experiment of identification test monitoring for the Jacobsen-Skogestad high purity distillation column: singular values of true plant vs. estimated models in frequency domain, MPC tuning set: PH=35, MH=5, Ywt=[1 1], and Uwt=[0.05 0.03]*0.13

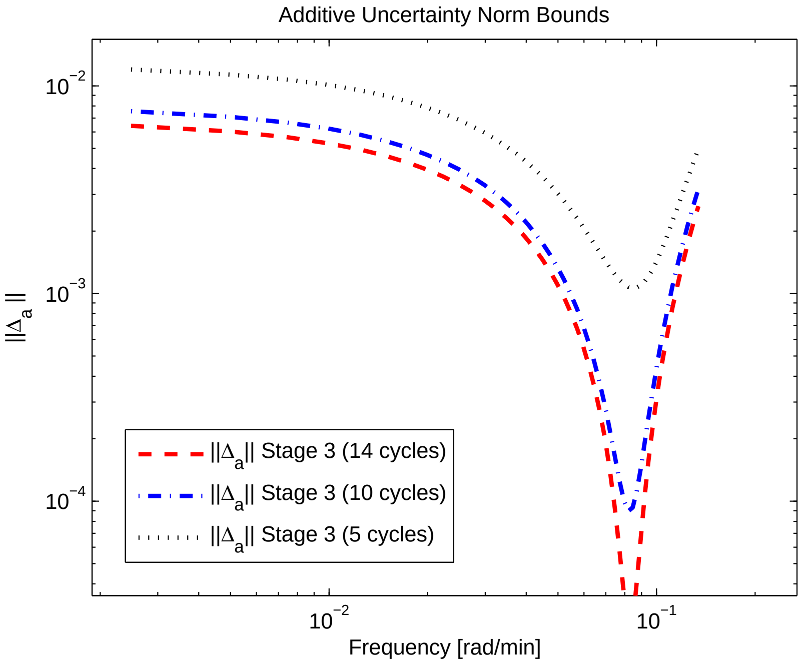
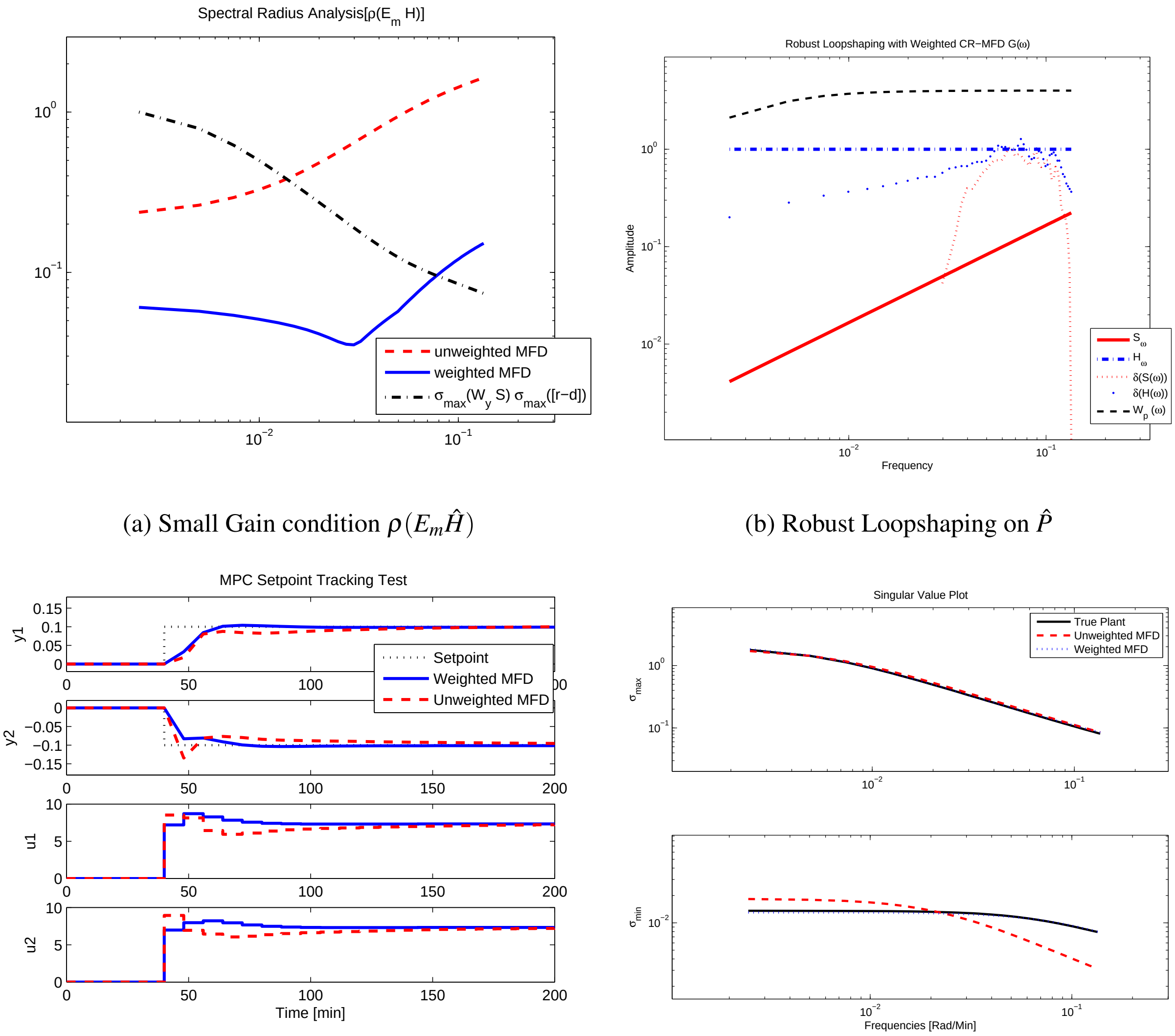


Figure 50: Experiment of identification test monitoring for the Jacobsen-Skogestad high purity distillation column: additive uncertainty norm bounds at 5, 10, and 14 cycles



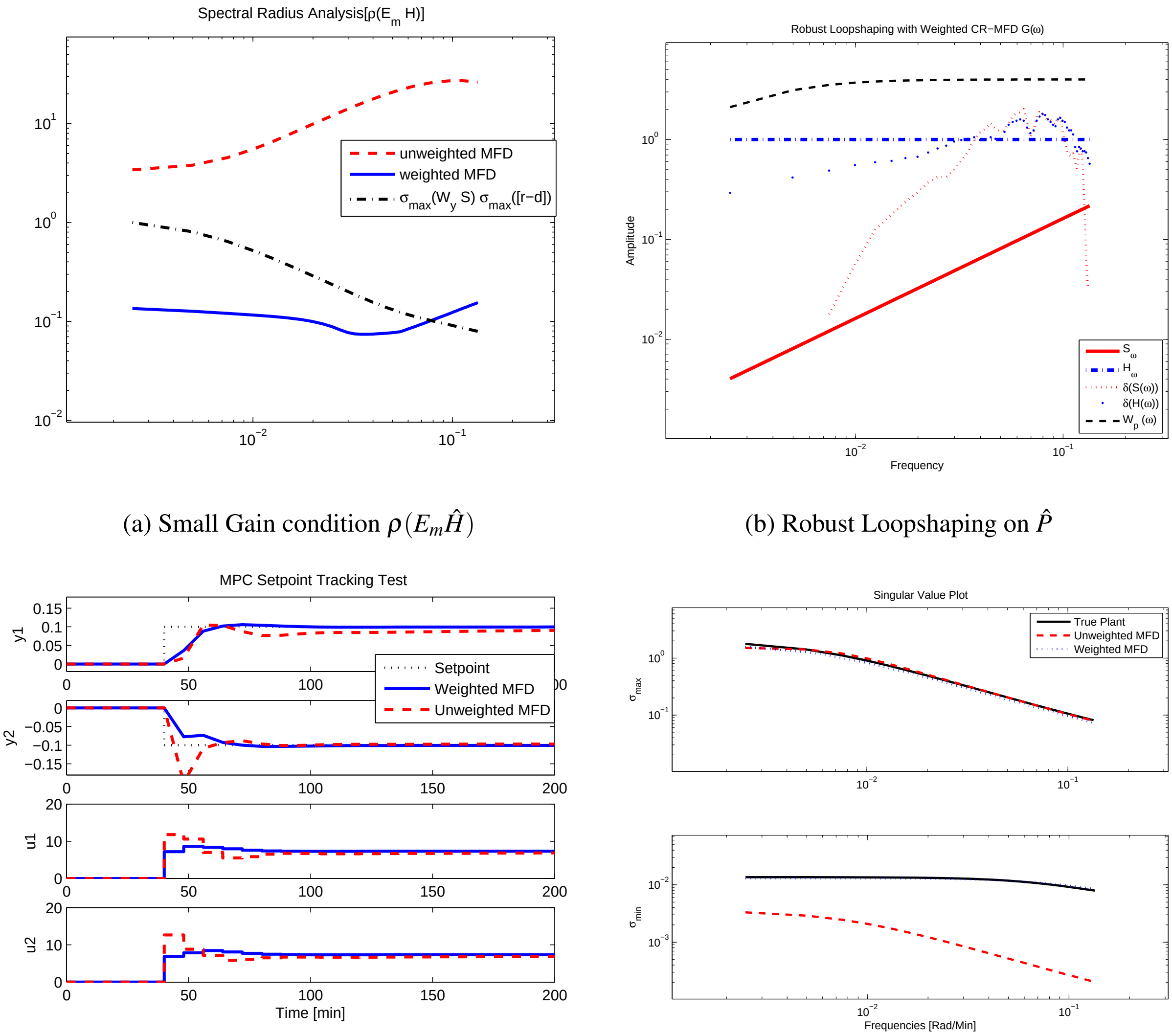
(a) Small Gain condition $\rho(E_m\hat{H})$

(b) Robust Loopshaping on $\hat{P}$

(c) Setpoint Test

(d) Singular Values

Figure 51: Experiment of identification test monitoring (at $n_{cycles} = 5$) for the Jacobsen-Skogestad high purity distillation column: Small Gain condition (a), robust loopshaping (b), MPC setpoint test (c), and singular values (d) with MPC tuning set: PH=35, MH=5, Ywt=[1 1], and Uwt=[0.05 0.03]*0.2



(a) Small Gain condition $\rho(E_m\hat{H})$

(b) Robust Loopshaping on $\hat{P}$

(c) Setpoint Test

(d) Singular Values

Figure 52: Experiment of identification test monitoring (at $n_{cycles} = 10$) for the Jacobsen-Skogestad high purity distillation column: Small Gain condition (a), robust loopshaping (b), MPC setpoint tracking test (c), and singular values (d) with MPC tuning set: PH=35, MH=5, Ywt=[1 1], and Uwt=[0.05 0.03]*0.2

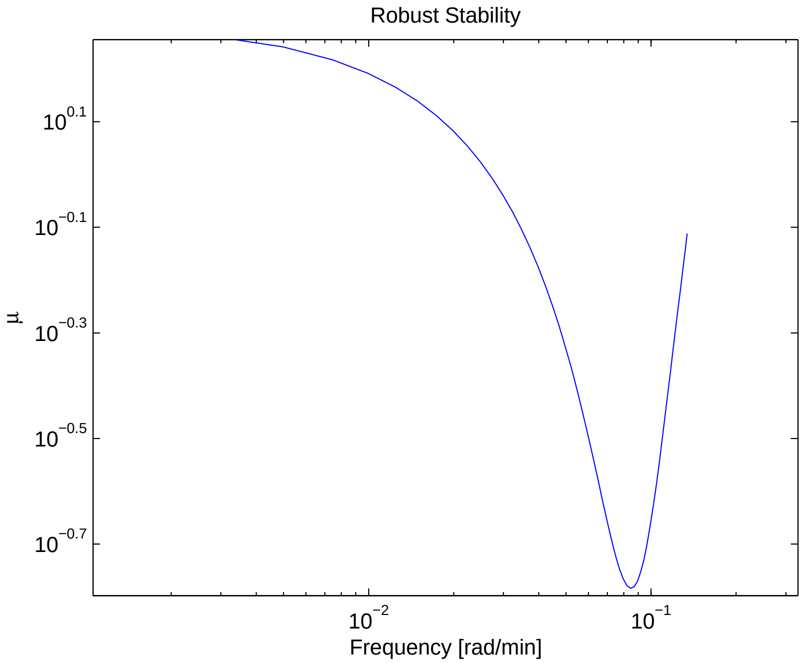


Figure 53: Experiment (at 5 cycles) of identification test monitoring for the Jacobsen-Skogestad high purity distillation column: robust stability analysis

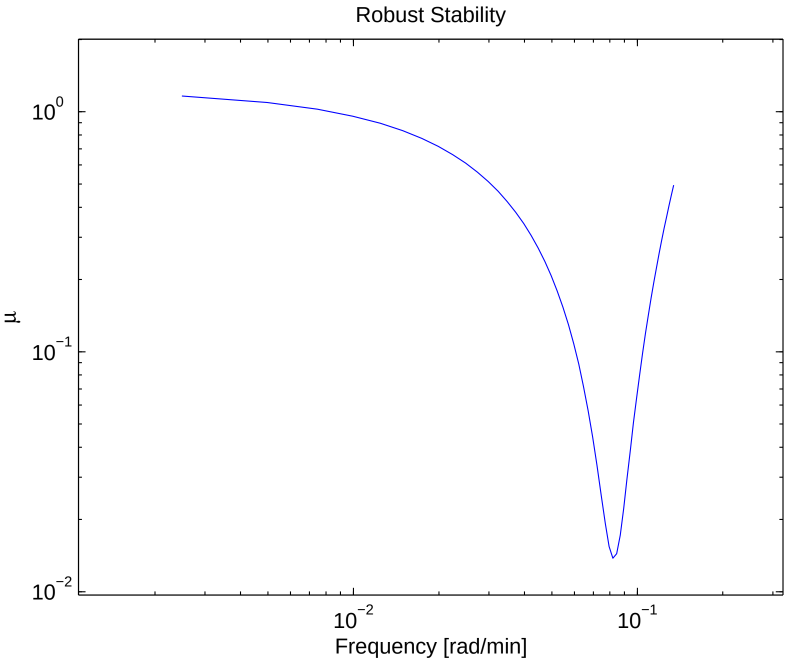


Figure 54: Experiment (at 10 cycles) of identification test monitoring for the Jacobsen-Skogestad high purity distillation column: robust stability analysis

References

Braun, M.W., R. Ortiz-Mojica and D.E. Rivera (2002). Application of minimum crest factor multisinusoidal signals for "plant-friendly" identification of nonlinear process systems. *Control Engineering Practice* **10**, 301.

Guillaume, P., J. Schoukens, R. Pintelon and I. Kollár (1991). Crest-factor minimization using nonlinear Chebyshev approximation methods. *IEEE Trans. on Inst. and Meas.* **40**(6), 982–989.

Hjalmarsson, H. (2005). From experiment design to closed-loop control. *Automatica* **41**, 393–438.

Jacobsen, E.W. and S. Skogestad (1994). Inconsistencies in dynamic models for ill-conditioned plants application to low-order models of distillation columns. *Ind. Eng. Chem. Res.* **33**, 631–640.

Lee, H. (2006). A plant-friendly multivariable system identification framework based on identification test monitoring. PhD thesis. Dept. of Chemical Engineering, Arizona State University. Tempe, AZ, U.S.A.

Lee, H. and D.E. Rivera (2004). A control-relevant, plant-friendly system identification methodology using shifted and "zippered" input signals. In: *2004 AIChE Annual Meeting*. Austin, TX. paper 414r.

Lee, H. and D.E. Rivera (2005*a*). Control relevant curvefitting for plant-friendly multivariable system identification. In: *2005 American Control Conference*. Portland, OR. pp. 1431–1436.

Lee, H. and D.E. Rivera (2005*b*). An integrated methodology for plant-friendly input signal design and control-relevant estimation of highly interactive processes. In: *Annual AIChE 2005 Meeting*. Cincinnati, OH. paper 520b.

Lee, H., D.E. Rivera and H. D. Mittelmann (2003*a*). A novel approach to plant-friendly multivariable identification of highly interactive systems. In: *Annual AIChE 2003 Meeting*. San Francisco, CA. paper 436a.

Lee, H., D.E. Rivera and H. Mittelmann (2003*b*). Constrained minimum crest factor multisine signals for plant-friendly identification of highly interactive systems. In: *13th IFAC Symp. on System Identification*. Rotterdam. pp. 947–952.

Ljung, L. and T. Glad (1994). *Modeling of Dynamic Systems*. Prentice-Hall. New Jersey.

Morari, M. and E. Zafiriou (1988). *Robust Process Control*. Prentice-Hall. Englewood Cliffs, N.J.

Prett, D.M. and C.E. García (1988). *Fundamental Process Control*. Butterworth. Stoneham, M.A.

Rivera, D.E. and S.V. Gaikwad (1995). Systematic techniques for determining modeling requirements for SISO and MIMO feedback control problems. *Journal of Process Control* **5**(4), 213–224.

Rivera, D.E., H. Lee, H.D. Mittelmann and M.W. Braun (2007). High purity distillation: Using plant-friendly multisine signals to identify a strongly interactive process. *IEEE Control Systems Magazine* **27**, 72 – 89. Special section on "Applications of System Identification.

Rivera, D.E., H. Lee, M.W. Braun and H.D. Mittelmann (2003). Plant-friendly system identification: a challenge for the process industries. In: *13th IFAC Symposium on System Identification (SYSID 2003)*. Rotterdam, Netherlands. pp. 917–922.

Rivera, D.E., M.W. Braun and H.D. Mittelmann (2002). Constrained multisine inputs for plant-friendly identification of chemical process. In: *15th IFAC World Congress, Barcelona, Spain.* paper T-We-A11.

Sanathanan, C.K. and J. Koerner (1963). Transfer function synthesis as a ratio of two complex polynomials. *IEEE Trans. Autom. Control* **9**, 56–58.

Schroeder, M.R. (1970). Synthesis of low-peak-factor signals and binary sequences with low autocorrelation. *IEEE Trans. Info. Theory* **IT-16**, 85–89.

Stenman, A., F. Gustafsson, D.E. Rivera, L. Ljung and T. McKelvey (2000). On adaptive smoothing of empirical transfer function estimates. *Control Engineering Practice* **8**(2), 1309–1315.