Selected Topics in Identification, Modelling and Control
Vol. 9, December 1996

# CLOSID - A closed-loop system identification toolbox for Matlab[‡§]

**Paul Van den Hof, Raymond de Callafon[♯] and Edwin van Donkelaar[♭]**

*Mechanical Engineering Systems and Control Group*
*Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands.*
*E-mail: p.m.j.vandenhof@wbmt.tudelft.nl*

Abstract. A closed-loop system identification toolbox for MATLAB is presented, including a user-friendly graphical user interface, that communicates with MathWork's System Identification Toolbox (SITB), version 4.0. With the CLOSID toolbox it is possible to identify linear (parametric) models on the basis of experimental data obtained from a plant that is operating under the presence of a controller. The toolbox is designed partially as a shell around the SITB, and has been given a similar setup. It comprises several closed-loop identification methods (both classical and more recent ones), and includes tools for evaluation of closed-loop model properties.

## 1 Introduction

Nowadays there are well-supported and user friendly tools available for the identification of (linear) systems on the basis of experimental data. See in particular the Mathwork's System Identification Toolbox SITB, version 4.0, which is equipped with a graphical user interface. This enables the user to identify and validate models in different types of model structures by mouse-clicking, rather than by entering (complex) commands. Additionally there is users' support in terms of graphical tools for model evaluation as well as support for e.g. bookkeeping of identified models.
In the tools that are currently available, there are only limited possibilities to identify models on the basis of data that is obtained under closed-loop experimental conditions. This particular experimental situation - which often occurs in practical situations - requires a special treatment, in the sense that besides input and output signals of a plant, measured external excitation signals can be involved, as well as some (possibly known) controller that is implemented on the system.
The current toolbox CLOSID offers an extension to the open-loop toolbox SITB, in the sense that

- It provides a graphical user interface supported tool for identification of models from closed-loop observations;
- It enables the use of external excitation signals as well as a (possibly) known controller in the loop;
- It communicates with the SITB, meaning that for the actual estimation part of the closed-loop identification methods, SITB is automatically opened and applied, while in the CLOSID tool the data processing and the (closed-loop) model processing is performed. Therefore full performance and flexibility of the estimation methods in SITB is retained.
- It provides evaluation of models in terms of their closed-loop properties, as e.g. sensitivity functions, complementary sensitivities, closed-loop poles, etcetera.

[‡]MATLAB is a registered trademark of the Mathworks, Inc.
[§]The software described in this paper is available through anonymous ftp at: `ftp-mesc.wbmt.tudelft.nl`, directory `/pub/matlab/closid`.
[♯]The work of Raymond de Callafon is financially supported by the Dutch Institute of Systems and Control (DISC).
[♭]The work of Edwin van Donkelaar is financially supported by the Dutch Technology Foundation (STW) under contract DWT55.3618

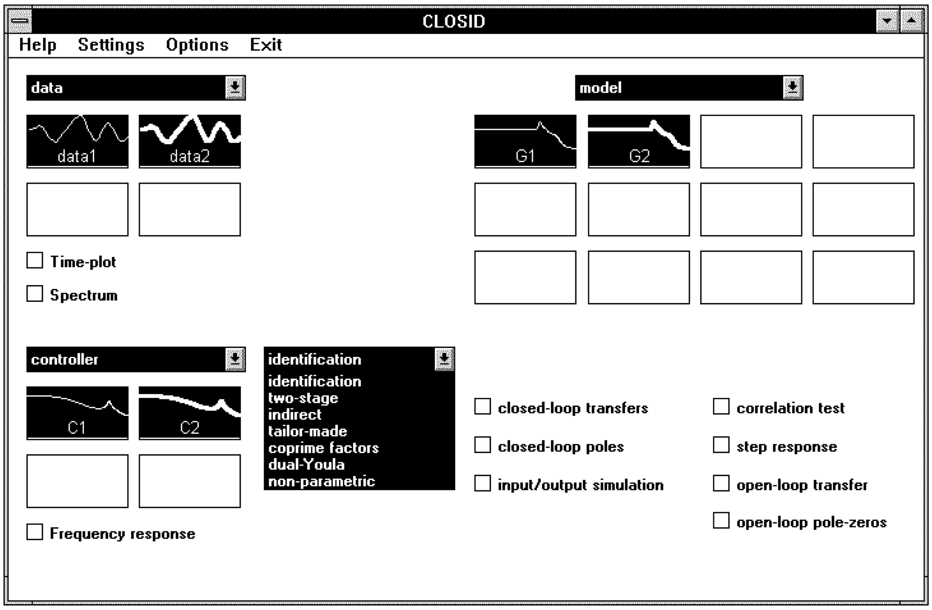


Fig. 1: CLOSID main window

In the current version, the graphical user interface of CLOSID is able to deal with SISO models only.

## 2 Main CLOSID-window

The graphical user interface of the CLOSID toolbox is opened by entering `closid` in the MATLAB command window. This opens the main window as shown in Figure 1.
The main window shows the following basic parts:

- a **data board** on the left upper part, where imported data sets are represented by colored line-icons, that can be selected by a mouse action.
- a **controller board** on the left lower part, where imported controllers are represented by colored line-icons, with similar selection options.
- an **identification menu** in the middle; this pop-up menu provides the user with a list of identification methods that can be applied.
- a **model board** on the right upper part, showing identified or imported models of the plant to be identified.
- a **model evaluation** area, containing check boxes for the application of several (closed-loop) evaluation procedures for the models on the model board.

Besides the controller board, the composition of the CLOSID main window is very similar to the main window of the SITB. This controller board is required, as some of the closed-loop identification methods need the a priori knowledge of the controller.
Additionally, this enables the user to evaluate the models in the presence of a (user-chosen) feedback controller.
Data sets, controllers and models can be imported from the MATLAB workspace, through selecting the respective pop-up menus for data, controller and model.
The closed-loop configuration that is considered all through the toolbox is depicted in Figure 2. It is also displayed in the **data import** window.

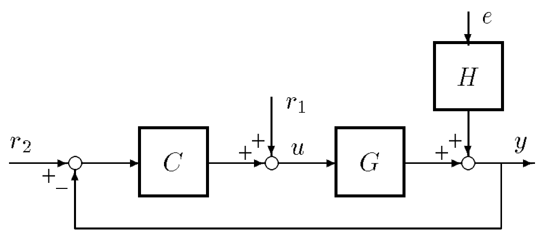


Fig. 2: Closed-loop system configuration

A data set is composed of experimental data $\{u, y\}$ over a given time horizon, together with either one of the external excitation signals $r_1$ and/or $r_2$. Data sets can be viewed on screen in terms of time sequences and power spectra, by clicking on the corresponding check boxes under the data board.
Models, as well as controllers, can be imported from and exported to the MATLAB workspace, in different formats:

- $[num; den]$: polynomial coefficients of numerator and denominator, in descending powers of $z$, stacked in a matrix with height 2.
- $[A\ \ B; C\ \ D]$: state space matrices (A,B,C,D) placed in a system matrix.
- *theta*: *theta*-format as used in the SITB.

The particular model import window is depicted in Figure 3.

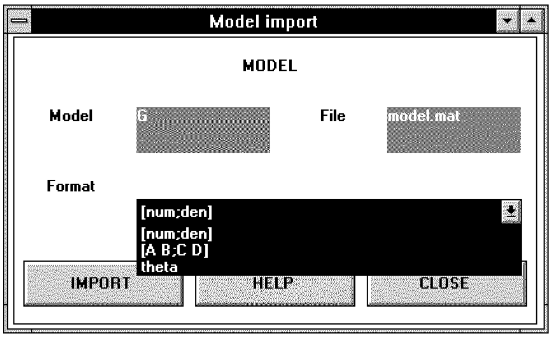


Fig. 3: Model import window

## 3 Closed-loop identification

The CLOSID toolbox contains five identification methods for parametric model identification, and one nonparametric method. The methods are denoted by

1. **two-stage** method,
2. **indirect** identification,
3. identification with a **tailor-made** parametrization,
4. **coprime factor** identification,
5. identification in the **dual-Youla** parametrization,
6. **non-parametric** (spectral) estimation.

For details on the different methods, one is referred to the references, in particular the survey paper Van den Hof and Schrama (1995).
The methods are all characterized by three steps, focussed on a specific closed-loop object that is going to be identified. E.g. in the indirect method, this closed-loop object is the plant-times-sensitivity $G/(1+CG)$, that is identified on the basis of measured signals $r_1$ and $y$. The three steps are clearly indicated in the several identification windows and are characterized as follows.

- **Construction of auxiliary i/o signals**.
  A first step of choosing/constructing appropriate auxiliary input and output signals, that are going to be used to identify a particular transfer function object.
- **Identification**.
  A second step of actual identification of the considered object, by estimating parameters through a least-squares identification criterion.
- **Calculate plant model**.
  From the identified object a plant model is constructed and this plant model is copied to the CLOSID model board.

By choosing one of the identification methods from the **identification** pop-up menu, a particular window is opened, displaying the three steps mentioned above.
The first step is trivial for some methods, but requires a separate identification for some others, as e.g. the identificaiton of the sensitivity function for the two-stage method. In these latter cases, quick-start options provide a simple means to construct the appropriate signals.
Apart from the "tailor-made" approach, all identification methods will perform the second step by opening MATLAB's SITB automatically, copying the appropriate signals from the CLOSID tool to the data board of SITB, allowing the user to identify the required transfer function object in the open-loop toolbox. In all of these situations, the second step is an identification problem that can be handled by the (open-loop) tools in hte SITB.
When an appropriate model is identified and validated in SITB, it can be copied to the CLOSID tool, by pushing the **Calculate and copy plant model** in the CLOSID identification window. This third step then transfers the plant model to the CLOSID model board.
As an illustration the **coprime factor** identification window is shown in Figure 4
The nonparametric identification method identifies spectral models for the one input, two output transfer from $r$ to $col(y, u)$, and constructs a plant model by taking the quotient of the two scalar nonparametric estimates.

## 4 Parametric methods

A brief overview is given of the characteristics of the different parametric methods. In the descriptions it is specified which data and priors are required (measured signal and/or knowledge of the controller), and which auxiliary information needs to be specified before the actual identification in step 2 can be performed.

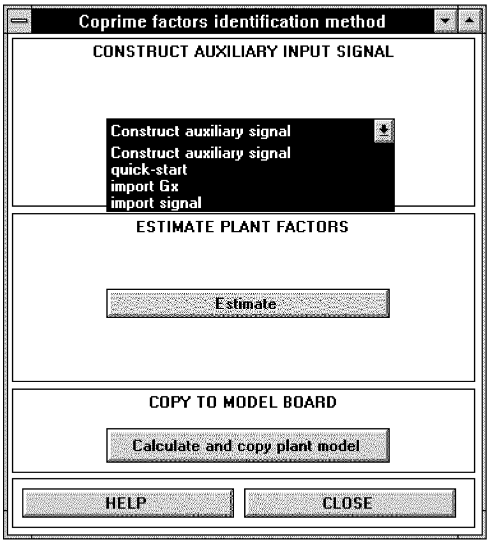


Fig. 4: Window for coprime factor identification

### 4.1 Two-stage method

**Approach**
In the first stage the transfer function between reference signal $r1$ and input signal $u$ (sensitivity function) is estimated, possibly with a high-order model. With this estimate a noise-free input signal is simulated, which is used in the second stage together with the measured output signal, to identify a plant model.

**Required data and priors**

- $r_1, u, y$

**Auxiliary information**
An estimate is required of the sensitivity function $S_0$, i.e. the transfer between $r_1$ and $u$. This is obtained in the first stage of the identification procedure, by

$$\hat{\beta}_N = \arg\min \frac{1}{N}\sum_{t=1}^{N}[u(t) - S(q,\beta)r(t)]^2$$

An accurate (high-order) model is obtained and denoted as

$$\hat{S}(q) = S(q, \hat{\beta}_N).$$

A quick-start option for this estimation is available.

**Signal construction**
The input and output signal for final estimation are constructed by

$$\begin{aligned} x(t) &= \hat{S}(q)r(t) \\ z(t) &= y(t) \end{aligned}$$

**Estimation** (in SITB)
Parameters are estimated according to (e.g.)

$$\hat{\theta}_N = \arg\min_{\theta} \frac{1}{N}\sum_{t=1}^{N}[z(t) - G(q,\theta)x(t)]^2.$$

**Plant model**
A model of the plant is obtained as

$$\hat{G}(q) = G(q, \hat{\theta}_N).$$

**Comments**
This method will generally not be able to provide unstable models of an unstable plant.

### 4.2 Indirect method

**Approach**
The closed-loop transfer function between $r_1$ and $y$ is estimated, and by using information on the implemented controller $C$, an open-loop plant model is reconstructed from this estimate.

**Required data and priors**

- $r_1, y$
- $C$

**Auxiliary information**
none.

**Signal construction**
The input and output signal for final estimation are constructed by

$$\begin{aligned} x(t) &= r_1(t) \\ z(t) &= y(t) \end{aligned}$$

**Estimation** (in SITB)
The exact transfer function between $x$ and $z$, i.e. the object of identification, is given by

$$R_o = \frac{G_o}{1 + CG_o}$$

Parameters are estimated according to e.g.

$$\hat{\theta}_N = \arg\min_{\theta} \frac{1}{N}\sum_{t=1}^{N}[z(t) - R(q,\theta)x(t)]^2,$$

leading to the identified transfer function

$$\hat{R}(q) = R(q, \hat{\theta}_N).$$

**Plant model**
A model of the plant is obtained as

$$\hat{G}(q) = \frac{\hat{R}}{1 - C\hat{R}}.$$

**Comments**
If the controller is stable, then $\hat{G}$ is guaranteed to be stabilized by $C$. The model order of $\hat{G}$ will generically be equal to model order of $\hat{R}$ plus order of $C$.

### 4.3 Identification with tailor-made parametrization

**Approach**
The closed-loop transfer function between $r_1$ and $y$ is estimated, using a dedicated parametrization in terms of the parameters of the open-loop plant model and the known controller $C$.

**Required data and priors**

- $r_1, y$
- $C$

**Auxiliary information**
none.

**Signal construction**
The input and output signal for final estimation are constructed by

$$x(t) = r_1(t)$$
$$z(t) = y(t)$$

**Estimation** (in CLOSID)
The exact transfer function between $x$ and $z$, i.e. the object of identification, is given by

$$R_o = \frac{G_o}{1 + CG_o}$$

Parameters are estimated according to

$$\hat{\theta}_N = \arg\min_{\theta} \frac{1}{N} \sum_{t=1}^{N} [z(t) - \frac{G(q,\theta)}{1 + C(q)G(q,\theta)} x(t)]^2,$$

leading to the identified transfer function

$$\hat{G}(q) = G(q, \hat{\theta}_N).$$

**Plant model**
A model of the plant is obtained as

$$\hat{G}(q).$$

**Comments**
The parameter set that corresponds to stable closed-loop systems may be disconnected in the case that the model order of $G(q,\theta)$ is smaller than the order of $C$. In this case inaccurate models can result.

### 4.4 Coprime factor method

**Approach**
The closed-loop transfer functions between $r$ (as input) and $(y, u)$ are estimated, and an open-loop plant model is obtained by taking the quotient of the two estimates.

**Required data and priors**

- $u, y$
- $C, r_1$ and/or $r_2$

**Auxiliary information**
Any auxiliary system $G_x$ with a factorization

$$G_x = \frac{N_x}{D_x}$$

that is stabilized by $C$.

**Signal construction**
The input and output signals for final estimation are constructed by

$$x(t) = r_1(t) + C(q)r_2(t)$$
$$z(t) = \begin{pmatrix} y(t) \\ u(t) \end{pmatrix}$$

**Estimation** (in SITB)
The exact transfer function between $x$ and $z$, i.e. the object of identification, is given by

$$\begin{pmatrix} N_o \\ D_o \end{pmatrix} = \begin{pmatrix} \dfrac{G_o F^{-1}}{1 + CG_o} \\ \dfrac{F^{-1}}{1 + CG_o} \end{pmatrix}$$

with $F^{-1} = D_x + CN_x$.
Parameters are estimated according to

$$\hat{\theta}_N = \arg\min_{\theta} \frac{1}{N} \sum_{t=1}^{N} \text{tr } [z(t) - \begin{pmatrix} N(q,\theta) \\ D(q,\theta) \end{pmatrix} x(t)][\cdot]^T$$

leading to the identified transfer functions

$$\begin{pmatrix} \hat{N} \\ \hat{D} \end{pmatrix} = \begin{pmatrix} N(q, \hat{\theta}_N) \\ D(q, \hat{\theta}_N) \end{pmatrix}.$$

**Plant model**
A model of the plant is obtained as

$$\hat{G}(q) = \frac{\hat{N}(q)}{\hat{D}(q)}.$$

**Comments**
By using a normalization procedure, and a common denominator parametrization in the identification, the model order of $\hat{G}$ will be equal to the maximum model order of $\hat{N}$ and $\hat{D}$.

| Identification method | Data | Auxiliary information | Signals for estimation $(x, z)$ | Estimated object $x \to z$ | Exported model |
|---|---|---|---|---|---|
| Two-step | $r_1, u, y$ | $\hat{S}$ | $x = \hat{S}r$<br>$z = y$ | $G_o$ | $\hat{G}$ |
| Indirect | $r_1, y$<br>$C$ | | $x = r$<br>$z = y$ | $R_o = \frac{G_o}{1 + CG_o}$ | $\hat{G} = \frac{\hat{R}}{1 - C\hat{R}}$ |
| Tailor-made | $r_1, y$<br>$C$ | | $x = S(\theta)r$<br>$z = y$ | $G_o$ | $\hat{G}$ |
| Coprime factors | $r, u, y$<br>$C$ | $G_x = N_x D_x^{-1}$ | $x = \frac{r}{D_x + CN_x}$<br>$z = (y, u)$ | $(N_o, D_o)$ | $\hat{G} = \hat{N}\hat{D}^{-1}$ |
| Dual Youla | $r, u, y$<br>$C$ | $C = N_c D_c^{-1}$<br>$G_x = N_x D_x^{-1}$ | $x = \frac{r}{D_x + CN_x}$<br>$z = \frac{y - G_x u}{D_c + G_x N_c}$ | $R_o$ | $\hat{G} = \frac{N_x + D_c\hat{R}}{D_x - N_c\hat{R}}$ |

Table 1: Synopsis of closed-loop identification methods

### 4.5 Dual-Youla method

**Approach**
A particular closed-loop transfer function is estimated, and by using knowledge of the controller an open-loop plant model is reconstructed. The plant model is guaranteed to be stabilized by the implemented controller. This method is a generalization of the *Indirect method.*

**Required data and priors**

- $u, y$
- $C, r_1$ and/or $r_2$

**Auxiliary information**
The controller $C$ is required to be known in a coprime factor representation

$$C = \frac{N_c}{D_c},$$

as well as any auxiliary system $G_x$ with a factorization

$$G_x = \frac{N_x}{D_x}$$

that is stabilized by $C$.

**Signal construction**
The input and output signals for final estimation are constructed by

$$x(t) = r_1(t) + C(q)r_2(t)$$
$$z(t) = \frac{1}{D_c + G_x N_c}[y(t) - G_x(q)u(t)].$$

**Estimation** (in SITB)
The exact transfer function between $x$ and $z$, i.e. the object of identification, is given by

$$R_o = \frac{(G_o - G_x)D_x}{D_c(1 + CG_0)}.$$

Parameters are estimated according to

$$\hat{\theta}_N = \arg\min_\theta \frac{1}{N}\sum_{t=1}^{N}[z(t) - R(q,\theta)x(t)]^2,$$

leading to the identified transfer function

$$\hat{R}(q) = R(q, \hat{\theta}_N).$$

**Plant model**
A model of the plant is obtained as

$$\hat{G}(q) = \frac{N_x + D_c\hat{R}}{D_x - N_c\hat{R}}$$

**Comments**
The model order of $\hat{G}$ will generically be equal to the sum of the model orders of $G_x$, $C$ and $\hat{R}$.

## Synopsis of parametric methods

In Table 1 a synopsis is given of the parametric identification methods. In this table the signal $r$ is used as an abbreviation for $r_1 + Cr_2$.

# 5 Model evaluation

Once a model is estimated and made available on the model board, several open-loop and closed-loop model properties can be evaluated. This is done using the seven **Model evaluation** options at the bottom of the main **Closid** window:

1. **closed-loop transfer** functions. The frequency responses of the four transfer functions from $col(r_2, r_1)$ to $col(y, u)$, are shown in a separate window, using the current models from the model board and the current controller $C$ from the controller board. In the window the amplitude of the frequency responses are shown, see Figure 5.

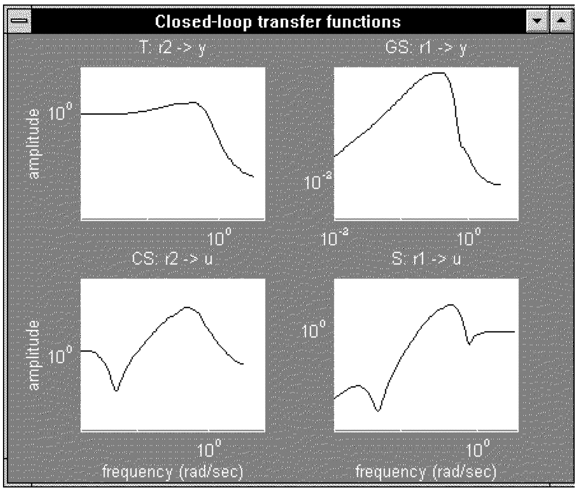


Fig. 5: Closed-loop frequency responses

2. **closed-loop poles**. When clicking this option, the poles of the closed-loop transfer functions are plotted in a separate window, also showing the stability region (unit circle). Thus a simple check is executed showing the (in)stability of the closed-loop system.

3. **input/output simulation**. Using the available reference signal(s) in the current data set, a plant input signal $u$ and plant output signal $y$ are simulated (noisefree), employing the current model and controller. These simulated signals are plotted together with the actual (measured) input and output signals from the current data set.

4. **correlation test**. The sample cross-covariance function is shown between the external reference signal $r$ in the current data set, and the output simulation error (top) and the input simulation error (bottom). This test indicates whether there is still reference signal information in the output and/or input residual, see Figure 6.

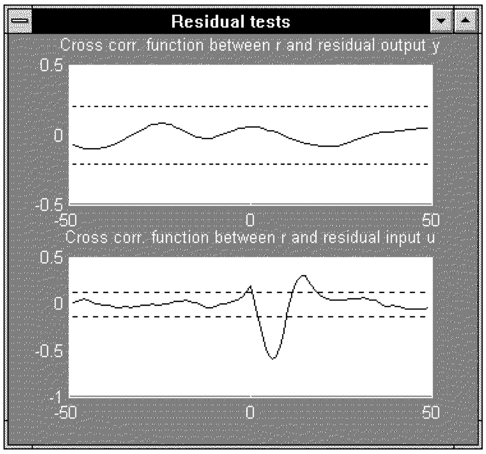


Fig. 6: Closed-loop correlation test

5. **step responses**. This option displays the step responses of the four closed-loop transfer functions from $col(r_2, r_1)$ to $col(y, u)$, for the current models on the model board and the current controller on the controller board.

6. **open-loop transfer**. The (open-loop) Bode diagram is displayed of the current plant models on the model board. This reflects the estimated transfer function between plant input $u$ and output $y$.

7. **pole-zero plot** of the estimated transfer function between the plant input $u$ and output $y$.

Selecting one or several of these evaluation tools will open a figure with a plot of the evaluation result for the current models from the model board; where appropriate the current data and current controller will also be employed. A zoom option is available in each figure. By selecting multiple models from the model board, evaluation results of several models can be compared in one figure.

# 6 Summary

A MATLAB toolbox has been presented for closed-loop system identification on the basis of time domain data. It has been designed as a "shell" around Mathworks' "open-loop" System Identification Toolbox (SITB). A graphical user interface constructed similar to the SITB supports the user, and facilitates exchange of models between the SITB and the current tool. In its current version the graphical user interface supports the identification of SISO models; the provided MATLAB m-files are implemented to handle also multivariable models.


## Acknowledgements

The authors acknowledge contributions of Peter Bongers, Ruud Schrama and Douwe de Vries to the research and the software implementations involved in this project.


## References


System identification:

Ljung, L. (1987). *System Identification: Theory for the User.* Prentice-Hall, Englewood Cliffs, NJ.

Ljung, L. (1995). *System Identificaiton Toolbox User's Guide*, Version 4.0, The Mathworks, Inc., Natick, Mass., 1995.

Survey papers on closed-loop identification:

Gevers, M. (1993). Towards a joint design of identification and control? In: *Essays on control: Perspectives on the Theory and its Applications*, Birkhäuser, Boston, pp. 111-151.

Van den Hof, P.M.J. and R.J.P. Schrama (1995). Identification and control - closed-loop issues. *Automatica, 31*, 1751-1770.

Two-stage method:

Van den Hof, P.M.J. and R.J.P. Schrama (1993). An indirect method for transfer function estimation from closed-loop data. *Automatica, 29*, 1523-1527.

Indirect method:

Söderström, T. and P. Stoïca (1989). *System Identification.* Prentice-Hall, Hemel Hempstead, U.K.

Van den Hof, P.M.J. and R.A. de Callafon (1996). Multivariable closed-loop identification: from indirect identification to dual-Youla parametrization. *Proc. 35th IEEE Conf. on Dec. and Control*, Kobe, Japan. (See also this issue, pp. 1–7).

Tailor-made approach:

Van Donkelaar, E.T. and P.M.J. Van den Hof (1996). Analysis of closed-loop identification with a tailor-made parametrization. In: O.H. Bosgra et al. (Eds.), *Selected Topics in Identif., Modelling and Control*, Vol. 9, Delft University Press, 1996, pp. 17-24.

Coprime factor identification:

Van den Hof, P.M.J. R.J.P. Schrama, R.A. de Callafon and O.H. Bosgra (1995). Identification of normalised coprime plant factors from closed-loop experimental data. *Europ. J. Control, 1*, 62-74.

Dual-Youla identification:

Lee, W.S., B.D.O. Anderdson, R.L. Kosut and I.M.Y. Mareels (1993). On robust performance improvement through the windsurfer approach to adaptive robust control. *Proc. 32nd IEEE Conf. Decision and Control*, San Antonio, TX, pp. 2821-2827.

Schrama, R.J.P. (1992). *Approximate Identification and Control Design.* Ph.D. Thesis, Mech.Eng. Sys. and Contr. Group, Delft Univ. of Techn., The Netherlands.