Condition Monitoring of Control Loops

Alexander Horch

Stockholm 2000

Submitted to the School of Electrical Engineering, Royal Institute of Technology, in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Process Control
Department of Signals, Sensors and Systems
Royal Institute of Technology
SE-100 44Stockholm, Sweden
http://www.s3.kth.se

TRITA–S3–REG–0002
ISSN 1404–2150
ISBN 91–7170–638–0

Printed by Universtitetsservice US AB
Stockholm December 2000

Labor voluptasque, dissimillima natura, societate
quadam inter se naturali sunt iuncta

Livius. Ab urbe condita 5,4,4

Abstract

The main concern of this work is the development of methods for automatic condition monitoring of control loops with application to the process industry. By condition monitoring both detection and diagnosis of malfunctioning control loops is understood, using normal operating data and a minimum amount of process knowledge.

The use of indices for quantifying loop performance is dealt with in the first part of the thesis. The starting point is an index proposed by Harris (1989). This index has been modified in order to cover a larger range of processes. The same concept is then used to assess the sampling rate in control loops. Other index-based monitoring methods where some amount of process knowledge is available are discussed.

The evaluation of the performance indices discussed requires knowledge of the process dead-time. Therefore a concept called *event-triggered estimation* is introduced in the second part of the thesis. Both automatic data selection and dead-time estimation methods are proposed and conditions for successful estimation are discussed.

The last part of the thesis deals with the diagnosis of oscillations. A method to automatically diagnose static friction (stiction) in the actuator is presented. Furthermore, two methods are proposed which allow automatic distinction of externally and internally generated oscillations. All described methods have been implemented in a MATLAB™-based graphical user interface which is briefly described.

Acknowledgements

Being a PhD student in Sweden still feels like a privilege. I am grateful for the opportunity to join the young and dynamic control group at KTH. Undergraduate teaching, graduate courses, conference participation and the work with my research project made the last 51 months very eventful and enjoyable.

Mainly responsible for that are Professors Bo Wahlberg and Alf Isaksson who encouraged me to leave Germany for five more years after having done my Master's thesis. I have not regretted that decision.

One of the main ingredients in a PhD student's daily life is the supervisor. There, I could not have made a better choice. Alf has the ability (and the patience) to scrutinize problems – both large and small – with the same enthusiasm. This in connection with the never-ending stream of energy and positive ideas made the work very fruitful. I appreciated the balance between individual freedom and "supervisory control".

The work reported here was mainly conducted within the KRIM-project (*Condition monitoring of control loops in pulp and paper processes*) together with Asea Brown Bovery (ABB) and AssiDomän. The financial support from the Swedish National Board for Industrial and Technical Development (NUTEK) is gratefully acknowledged. Special thanks to Krister Forsman, Carl-Fredrik Lindberg, Stefan Rönnbäck, Andreas Stattin and Inge Åkesson for sending me data, testing my algorithms, answering and asking questions and for being nice people in general. The project was a successful and very pleasant cooperation.

Professor Tore Hägglund's help such as providing some of the data, focusing my view to control valves and discussing ideas, has been very helpful and is gratefully acknowledged.

There *was* a life outside KTH. The endless number of choir rehearsals resulted in some moments out of the ordinary: meeting the King and the Queen, singing at the Nobel banquet, in Jerusalem and Spain. Thank you for singing with me Sofia KK and Royal Philharmonic Choir. Many other people made us feel at home in the beautiful city of Stockholm. Great thanks to our Dutch-French-Irish-Swedish friends for all the nice weekend trips and so much more.

Love is the greatest refreshment in life[1]. Thank you to my wonderful family, Ulrike and Jonathan and •. I am very happy to share my life with you. Finally, my parents deserve a special acknowledgement having supported me for five years of undergraduate studies – and so much more.

[1] Picasso

Contents

Chapter 1

Introduction

1.1 Control performance monitoring

Performance in control loops is an important issue. A large part of the literature concerning control performance deals with the performance during the *control design stage*. A process model, specifications and results from experiments are often available to the designer. Many of the available performance measures are therefore defined from this background.

The literature is much more sparse when it comes to the *control performance assessment stage*. Typically, in such a situation one does not have the same information as during the design stage. The reasons for this are many: lost information, bad documentation, trial-and-error tuning, no-tuning at all etc.

A simple illustration may help to understand what is meant here by condition monitoring, see Figure 1.1. The two main problems one has to face when desiring to assess the performance of a control loop are

- to obtain information despite the required passiveness of an automatic monitoring tool,
- to quantify what kind of behaviour is considered to be acceptable.

It is obvious that the more information about the process that is available, the better one can assess the control performance. However, the amount of available information is restricted since a monitoring tool is subject to certain operating constraints. The most important ones posed by the industry are

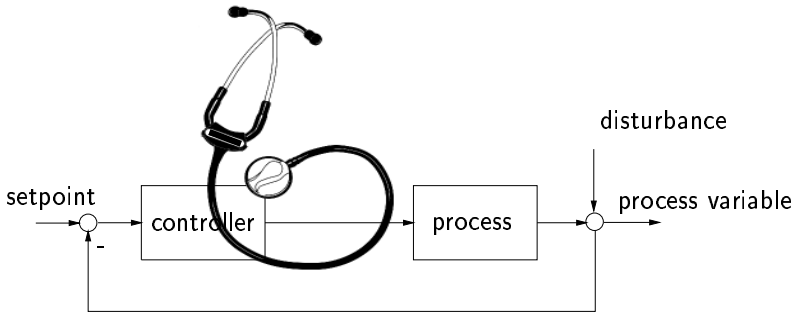


Figure 1.1: A simple illustration of condition monitoring. Using non-invasive tools ("stethoscope") the aim is to monitor the performance of the control loop.

1. Non-invasiveness. The tool has to be a passive observer.
2. No new sensors. Only available measurement signals may be used.
3. Minimal process knowledge. Ideally, one should not assume any knowledge about the loop in question[1].
4. Simple and non-complex algorithms. Since the aim is to monitor many (in the order of $10^2 - 10^3$) control loops, it is important that the complexity of the used algorithms is modest.

On the other hand, there are requirements on the tasks which a monitoring tool has to be able to perform. The following list is a selection of properties which the author thinks are most important.

1. Detection of malfunctioning loops, i.e. loops with large variability.
2. Diagnosis of malfunctioning loops.
3. Suggestion of suitable measures in order to remove the cause of the performance deterioration.
4. Present the results to the user in an intelligible way.

[1] This assumption will be qualified to some extent in later chapters

1.2 Previous work

A brief review of related work within the area of control performance monitoring will be presented here. More detailed reviews will be given later on in this thesis. Most of the research has been concentrated around the minimum-variance based performance index, first suggested by Harris (1989). The ideas go back to Åström (1970) who settled the theory necessary for minimum-variance control and DeVries and Wu (1978) who used these ideas for performance assessment. The concept has also been extended to feed-forward loops, see (Desborough and Harris, 1992, 1993). Many others took up the idea and applied it to real processes, mostly within the pulp & paper industry, see for example (Stanfelj *et al.*, 1991), (Perrier and Roche, 1992), (Jofriet *et al.*, 1995), (Owen *et al.*, 1996), (Thornhill *et al.*, 1999). Modifications and extensions of the original performance index have been reported by Tyler and Morari (1995, 1996) who extended the concept to unstable and non-minimum phase processes and introduced statistical likelihood ratio tests. Lynch and Dumont (1996) used Laguerre networks to evaluate the performance index. Also multivariable extensions of the Harris performance index have been studied, see for example (Harris *et al.*, 1996*b*), (Huang *et al.*, 1997), (Huang and Shah, 1999) or (Ettaleb, 1999).

Furthermore, conceptually different performance indices have been proposed. Eriksson and Isaksson (1994) and Ko and Edgar (1998) considered constraints on the control structure, Hägglund (1995) proposed a methodology to detect oscillations, Swanda and Seborg (1997) suggested a performance index which normalises the closed-loop settling time with the process time-delay. Hägglund (1999) presented a method to detect sluggish control loops, Bezergianni and Georgakis (2000) defined a performance index which compares actual control to both minimum-variance and open-loop control. Rhinehart (1995) and Venkataramanan *et al.* (1997) proposed and applied a statistical test which detects deviations from setpoint.

Commercial tools

The demand in the process industry for control performance monitoring tools is considerable. Therefore, many suppliers have either started their own implementations of published results, started collaborations with groups who do research in the area or purchased commercially available software tools.

Future control systems will most likely have monitoring tools as a part of the system. Nowadays, supervision tools are often used as stand-alone units. Except for single company implementations and development tools used by universities, the following commercial tools are known to the author

- ABB: `Loop optimizer suite`™
- Honeywell: `Loop scout`™, `@`*sset*.**MAX**™
- Pulp & Paper Research Inst. of Canada (PAPRICAN): `LoopMD`™
- Invensys, Foxboro: `LoopAnalyst`™
- Matrikon: `ProcessDoc`™

1.3 Contributions

Results which can be considered as the main contributions of the thesis are:

- A modification of an existing minimum-variance based performance index. The modification is shown to be significant for a certain class of processes. Furthermore, the modified method does not require more knowledge than the original one, but permits the incorporation additionally available process knowledge.
- Using the minimum-variance based performance index, a method is developed which allows control performance assessment for sampling rates higher than the actual one used for data collection. By that, the benefit of faster sampling can be quantified.
- A new method for the detection of static friction (stiction) in valves in self-regulating processes.
- A new method for detection of stiction in integrating processes.
- Modification of a method for dead-time estimation from normal operating data.
- An automatic procedure for data segmentation for use in closed-loop estimation.

- A new method for distinction between internally and externally caused oscillations based on a simple process model.

- A modified method for evaluation of the Harris index for oscillating loops. This method can also be used to distinguish between internally and externally caused oscillations.

The work reported in this thesis has thus far led to the following publications and submissions:

A patent application has been filed for the method proposed in Chapter 11.

The material in Chapter 3 has been published in

- A. Horch and A.J. Isaksson. "A modified index for control performance assessment". In *Journal of Process Control*, 9(6):475-483, December 1999. A shorter version has also been published in *1998 American Control Conference*, Philadelphia, PA, pp. 3430-3434, June 1998.

The material in Chapter 4 has been published in

- A. Horch and A.J. Isaksson. "Assessment of the sampling rate in control systems". Proc. *Control Systems 98*. Porvoo, Finland, pp.267-274, September 1998. An extended version has been accepted for publication in *Control Engineering Practice*, 2001.

The method proposed in Chapter 10 has been published in

- A. Horch. A simple method for detection of stiction in process control loops. In *Control Engineering Practice*, 7(10):1221-1231, October 1999. A shorter version has also been published in *IEEE Int. Conference on Control Applications*, Hawai'i, USA, 1284-1289, August 1999.

Parts of Chapters 6, 7 and 8 have been submitted as

- A.J. Isaksson, A. Horch and G.A. Dumont. Event-triggered deadtime estimation from closed-loop data. In *American Control Conference*, Arlington, Virginia, USA, 2001.

A short version of Chapter 11 has been submitted as

- A. Horch and A.J. Isaksson. Detection of valve stiction in integrating processes. In *European Control Conference*, Seminário de Vilar, Porto, Portugal, September 2001.

Work that has not been explicitly included in this thesis but was performed during the Ph.D. studies was published in

- A. Horch and A.J. Isaksson. "A method for detection of stiction in control valves". In Proc. *IFAC Workshop on On-Line-Fault Detection and Supervision in the Chemical Process Industry*, Session 4B, Lyon, France, June 1998.

- A.J. Isaksson, A. Horch and G.A. Dumont. Event-triggered deadtime estimation – comparison of methods. In *Control Systems 2000*, Victoria, B.C., Canada, pp.209-215, May 2000.

- A. Horch, A.J. Isaksson, B.J. Allison, A. Karlström and L. Nilsson. "Dynamic simulation of a thermomechanical pulp refiner". *Nordic Pulp and Paper Res. J.*, Vol.12, No.4, pp.270-275, December 1997. Also presented at the *4th PIRA International Refining Conference*, Fiuggi, Italy, March 1997.

- A.J. Isaksson, A. Horch, B.J. Allison, A. Karlström and L. Nilsson. "Modelling of Mechanical Thrust in TMP Refiners". Presented at the *International Mechanical Pulping Conference 1997*, Stockholm, Sweden, June 1997.

- K.H. Johansson, A. Horch, O. Wijk and A. Hansson. "Teaching multivariable control using the Quadruple-tank process". In *38th IEEE Conference on Decision and Control*, Phoenix, AZ, December 1999.

1.4 Topics for further research

Even though the research in the area of control performance monitoring is quite active, there are only few commercially available tools which are able to cover the whole range of requirements mentioned before. Unfortunately, many of the commercially used algorithms are not available in the literature. There are, however scattered results published which suggest solutions for

some of the problems in connection with condition monitoring. In addition, a lot of practical applications have been reported.

There is, however, still a lot of research to do even though it is not clear how far one can get while adhering the restrictive practical constraints given by the industry. It is more than likely that one cannot perform a detailed (and error-free!) performance assessment of a complicated industrial process simply by passively collecting data.

Future work may include some of the following topics:

Learning supervision tool. Learning here means the *automatic* construction of a database for each loop. For example the storage of important loop information such as re-tuning, operating range, valve characteristics, process model (if available) etc. Such information – if available – may then be used by different algorithms and then enable more detailed analysis.

Smart equipment. Modern actuator elements (e.g. control valves) and sensor equipment are nowadays operated digitally and have the ability to perform computations and to communicate with the distributed control system. These features should of course be used for self-diagnosis in actuators. A supervision system could for example compare the self-diagnosis to other available data.

Model-based diagnostics. This thesis deals mainly with model-free approaches, i.e. no process model has to be available when commissioning the monitoring tool. When using mechanistic process models, one could use the large quantity of results from the fault detection area, see (Isermann and Ballé, 1997) for a recent survey. This could be interesting for example for the petro-chemical industry, where (at least static) models for most processes are usually available.

Large-scale tests. Many different approaches for performance assessment have been proposed during the last decade. However, no large scale comparison of all methods has been published so far. It would be an interesting experiment to design a database of benchmark data which can be used by all researchers and suppliers within the area.

Part I

Performance assessment using indices

Chapter 2

Control performance indices

One of the starting points for the work presented in this thesis is a control performance index proposed by Harris (1989). This method will briefly be reviewed in Section 2.1. Ingredients in Harris' method are the fitting of a time-series model to the process output and the calculation of the output variance. These topics are discussed separately since they can be done in several ways, see Sections 2.2 and 2.3 respectively. Strengths and weaknesses of the Harris index are then discussed in Section 2.4. How a similar performance index has been defined by Eriksson and Isaksson (1994) if a process model is available is described in Section 2.5.

The Harris index has started an intensive research activity which has resulted in a number of similar and new concepts for performance assessment using indices. Some of these approaches will briefly be reviewed in Section 2.6.

2.1 A minimum-variance based index

An appealing method for control performance assessment was first described by Harris (1989) and compares actual (process) output variance σ_y^2 to the output variance σ_{MV}^2 as obtained using a minimum variance controller. A

performance index is defined as

$$I_p = \frac{\sigma_y^2}{\sigma_{MV}^2}. \tag{2.1}$$

The index will of course always be larger than or equal to one, where values close to one indicate good control with respect to the theoretically achievable output variance.

All the information needed to compute the index (2.1) is one set of normal operating output data, $y(t)$, and knowledge of the process dead-time d. As pointed out by Harris, the minimum achievable variance, σ_{MV}^2, can indeed be calculated, regardless of what the current controller is. To calculate σ_{MV}^2, estimate a time-series model from the measured output

$$y(t) = \frac{C(q^{-1})}{A(q^{-1})} e(t). \tag{2.2}$$

A series expansion (i.e. the pulse response) of this time-series model can be written as

$$\begin{aligned} y(t) &= \left(h_0 + h_1 q^{-1} + h_2 q^{-2} + \dots\right) e(t) \\ &= \sum_{i=0}^{\infty} \left(h_i q^{-i}\right) e(t) \end{aligned} \tag{2.3}$$

where h_i, $i = 0, \dots$ are the Markov-parameters of the time-series model. It can be shown that the first d elements of this series expansion coincide with the coefficients of the series expansion of the noise process, see (Åström, 1970). The reason is that no controller can influence the process output before the time-delay d has elapsed.

Theoretically, a minimum-variance controller can completely remove all influence from the noise after the time-delay. Hence, the theoretically optimal output is a moving average process

$$y(t) = \left(h_0 + h_1 q^{-1} + \dots + h_{d-1} q^{-(d-1)}\right) e(t), \tag{2.4}$$

and as a consequence

$$\sigma_{MV}^2 = (h_0^2 + h_1^2 + h_2^2 + \dots + h_{d-1}^2)\sigma_e^2. \tag{2.5}$$

The actual output variance can of course directly be estimated from the collected samples of $y(t)$. However, it is suggested to use the estimated time-series model also for evaluating the current variance σ_y^2, see Section 2.3 for a more detailed discussion. It is therefore proposed to compute the performance index using

$$I_p = \frac{\sum_{i=0}^{\infty} h_i^2}{\sum_{i=0}^{d-1} h_i^2}. \tag{2.6}$$

Note also that this form of the index guarantees that $I_p \geq 1$ and that the noise variance estimate σ_e^2 is not needed since it is cancelled.

To summarize, the complete algorithm to evaluate the performance index (2.1) contains the following steps:

1. Estimate a time-series model (2.2) for the measured output $y(t)$.
2. Calculate the series-expansion (2.3) of the estimated time-series model.
3. Calculate the minimum achievable variance (2.5) based on the series expansion.
4. Calculate the actual output variance σ_y^2 (Section 2.3).
5. Compute the performance index (2.1).

2.2 Time-series modelling

Even though the evaluation of the Harris index is straight forward, one has to make certain decisions. One of them is the choice of the time-series model structure. Several different structures can be chosen from:

Auto-regressive moving average (ARMA) model. The process variable $y(t)$ is described by

$$A(q^{-1})\,y(t) = C(q^{-1})\,e(t). \tag{2.7}$$

As will been shown in Chapter 3.1 (Equation 3.2), the process variable $y(t)$ is theoretically a real ARMA process. In order to estimate the parameters in (2.7), one has to solve a nonlinear optimisation problem.

Moving average (MA) model. If $A(q^{-1}) \equiv 1$ in (2.7), the process variable is described by

$$y(t) = C(q^{-1})\,e(t). \tag{2.8}$$

This model has finite memory. In order to model closed-loop systems with slow decay a large number of parameters will be required. Still the parameter estimation has to be done by solving a nonlinear estimation problem. As a consequence it seems reasonable to estimate a full ARMA-model instead, thereby being able to reduce the number of parameters.

Auto-regressive (AR) model. If $C(q^{-1}) \equiv 1$ in (2.7), the process variable is described by

$$A(q^{-1})\,y(t) = e(t). \tag{2.9}$$

The estimation of the parameters in (2.9) is simple and the problem can be formulated as a linear regression with a closed-form solution. Theoretically, one needs an AR-model of infinite order to completely describe an ARMA model. This can be seen using a simple example:

$$\begin{aligned} \mathrm{ARMA}(1,1) = \frac{1+cq^{-1}}{1+aq^{-1}} &= \frac{1}{(1+aq^{-1})\frac{1}{1+cq^{-1}}} \\ &= \frac{1}{(1+aq^{-1})(1-cq^{-1}+c^2q^{-2}+\cdots)} = \mathrm{AR}(\infty). \end{aligned}$$

It is usually sufficient to use a high-order AR model in order to approximate a process which is really ARMA, see for example (Wahlberg, 1990) for a detailed discussion. Ogawa (1998) proposed the use of the partial autocorrelation function (PACF) in order to determine the AR model order required to

describe a certain signal sufficiently. The author makes use of the fact that the PACF of an AR(n)-model vanishes after lag n. The PACF measures the correlation between $y(t)$ and $y(t-j)$ after the correlation due to intermediate values $y(t-1), \ldots, y(t-j+1)$ has been eliminated. The PACF can be calculated using the Levinson-Durbin algorithm or as a by-product from a lattice filter estimator.

A simple way of determining the necessary AR model-order is hence to calculate the PACF up to an order which is almost surely larger than the required model order. Then a sufficiently high model AR model-order is the lag from which on the PACF stays within its 95% confidence limit.

The estimation of the parameters in $A(q^{-1})$ can be done in may ways. Some commonly used algorithms, see for example Ljung (1999), are

1. *Least-squares method.* The sum of squared forward prediction errors is minimised.

2. *Forward-backward method.* The sum of a least-squares criterion for a forward model and the analogous criterion for a time-reversed model is minimised.

3. *Yule-Walker method.* Solving the Yule-Walker equations, formed from sample covariances (using e.g the Levinson-Durbin algorithm).

4. *Lattice filters* The lattice filter equations are solved, using the harmonic (Burg's method) or the geometric mean of forward and backward squared prediction errors.

Note that for some of the methods the estimated model is also guaranteed to be stable (Methods 3 and 4). Performing comparative tests on real data it has been found that the estimated time-series model does not change significantly when using different estimation methods for the AR-parameters.

Laguerre network. The use of a Laguerre network for evaluation of the Harris index has been proposed by Lynch and Dumont (1996). Laguerre models will also be used for dead-time estimation in Chapter 8. A Laguerre network can be used either in a moving average or an auto-regressive sense,

i.e.

$$y(t) = \left(1 + \sum_{i=1}^{n} c_i L_i(q,\alpha)\right) e(t), \tag{2.10}$$

$$y(t) = -\left(\sum_{i=1}^{n} c_i L_i(q,\alpha)\right) y(t) + e(t), \tag{2.11}$$

where the Laguerre filters are given by

$$L_i(q,\alpha) = \frac{\sqrt{1-\alpha^2}}{q-\alpha}\left(\frac{1-\alpha q}{q-\alpha}\right)^{i-1}, \quad i \geq 1.$$

Note that if the filter pole is chosen as $\alpha = 0$, models (2.10) and (2.11) reduce to MA- and AR-models respectively. See also Figure 2.1 for a block diagram of (2.11). The estimation of the parameters in the Laguerre model

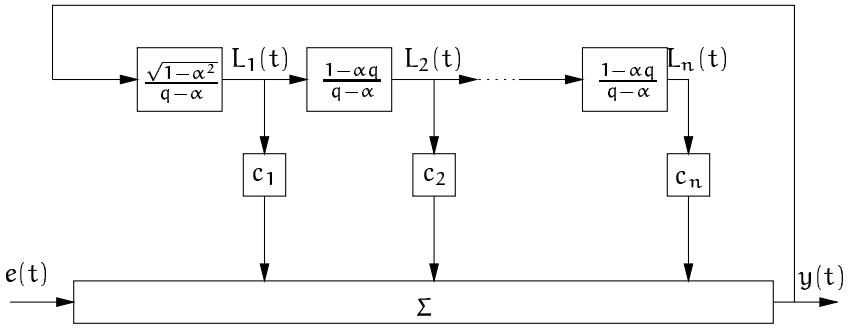


Figure 2.1: Block diagram of the Laguerre network (2.11).

(2.11) can be done easily since the estimation problem can be formulated as a linear regression. A thorough treatment of parametric signal modelling with Laguerre filters has been presented by Wahlberg and Hannan (1993). They conclude that – having chosen the Laguerre filter pole appropriately – the number of parameters needed to obtain useful approximations can be considerably reduced compared to AR modelling. Also, several results on AR parameter estimation could be generalised to Laguerre models (e.g. asymptotic statistical properties and the Levinson algorithm).

When using (2.10), the estimation problem is more difficult since the noise (which has to be estimated) is filtered. The benefit for the price paid is that one obtains the Markov-parameters directly from the time-series model. The use of Laguerre models for industrial problems has been advocated by Dumont *et al.* (1990) and by Wang and Cluett (2000)

Recommended model orders. Whatever model structure is chosen for the estimation, one has to choose a certain model order. It can easily be verified that the performance index may vary considerably if the model order is chosen too low. As a consequence, the model order is recommended to be chosen rather too high than too low. Reasonable values which have been tested in practice are shown in Table 2.1. The orders for AR and Laguerre

model structure	model order
AR	15-25
ARMA	8-12
Laguerre	10

Table 2.1: Recommended model orders for time-series modelling.

modelling have been proposed by Ogawa (1998) and Lynch and Dumont (1996) respectively.

2.3 How to compute the actual variance

The simplest way of estimating the actual output variance σ_y^2 is directly from measurement data, i.e.

$$\sigma_y^2 = \frac{1}{N-1}\sum_{i=1}^{N}[y(i)-\bar{y}]^2, \qquad \bar{y} = \frac{1}{N}\sum_{i=1}^{N} y(i). \tag{2.12}$$

However, since a time-series model for $y(t)$ has already been estimated, one can make use of that for estimation of σ_y^2 also. Using the series expansion of the time-series model, we have

$$\sigma_y^2 = \left(\sum_{i=0}^{\infty} h_i^2\right)\sigma_e^2. \tag{2.13}$$

Note that using (2.13) has the nice effect that the noise variance (estimated using the residual variance from the identification procedure) will be cancelled when computing the performance index (2.1). Since the residuals contain all unmodelled dynamics, this is an advantage.

Of course, the evaluation of (2.13) should not be done by computing the sum directly. One can instead convert the time-series model to state-space form $(\mathbf{F}, G, H)$ and use

$$\begin{aligned} \mathbf{P} &= \mathbf{F}\mathbf{P}\mathbf{F}^T + GG^T \\ \sigma_y^2 &= Ey(t)^2 = \left(H\mathbf{P}H^T\right)\sigma_e^2 \end{aligned} \tag{2.14}$$

which only requires the solution of a Lyapunov equation.

2.4 Pros and Cons

The index as described in the previous section has become popular in the process industry, especially in the paper industry where many applications are reported, see e.g. (Fu and Dumont, 1995), (Harris *et al.*, 1996*a*), (Kozub, 1996), (Lin *et al.*, 1998) or (Ogawa, 1998). The advantages are obvious:

+ easy to implement
+ easy to interpret
+ non-invasive
+ modest process knowledge required (dead-time only)

These properties are perfectly in line with the requirements for an industrial control performance tool. There are however some weaknesses which have to be taken care of, namely

– comparison to perfect control, i.e. possibly overly optimistic indices,
– dead-time often unknown or time-varying,
– assesses only dead-time as performance limiting factor,
– data pre-treatment necessary in order to avoid use on "bad" data (for example outliers, linear trends etc.),

- based on stochastic control (i.e. stochastic disturbances) whereas the control task is often rejection of deterministic disturbances or setpoint tracking,
- even though the Harris index is an absolute measure on loop performance, one has to set reasonable alarm limits. Malfunctioning loops shall be detected but no unnecessary alarms should occur. (A limit of two or three is often used in practical applications).

An example

Consider a simple example taken from (Åström and Wittenmark, 1997). The ARMAX process is given by

$$(1 - q^{-1})(1 - 0.7\,q^{-1})\,y(t) = (1 + 0.5\,q^{-1})\,u(t-2) + (1 - 0.9\,q^{-1})\,e(t)$$

where $e(t)$ is zero mean white noise with variance $\sigma_e^2 = 1$. Consider first a minimum-variance controller for that process, which can be calculated as

$$(1 + 0.5\,q^{-1})(1 + 0.8\,q^{-1})\,u(t) = -(0.66 - 0.56\,q^{-1})\,y(t).$$

Closed-loop data using this controller can be seen on the top left side of Figure 2.2. The bottom left plot shows the impulse response of the (estimated) time-series model. It can clearly be seen that the controller succeeds in removing the stochastic disturbances after the dead-time (dashed-line) has elapsed. Since the process has a time-delay of two samples, the theoretical minimum variance is $\sigma_{MV}^2 = (1 + 0.8^2) * 1 = 1.64$. The actual output variance when using the minimum-variance controller is $\sigma_y^2 = 1.61$. The minimum achievable variance as computed from the estimated model is $\sigma_{MV}^2 = (1 + h_1^2) * 1 = 1.52$ yielding an index of $I_p = 1.06$.

As a comparison consider a P-controller with gain 0.06 instead. The results using this controller are shown on the right side of Figure 2.2. The actual output variance is $\sigma_y^2 = 3.29$, the minimum-achievable variance is estimated to be $\sigma_{MV}^2 = 1.55$ yielding an index of $I_p = 2.12$.

It is interesting to take a look at the control signals which were used in both simulations, see Figure 2.3. The control signal variances are 1.87 and 0.12 respectively. The minimum-variance controller requires hence a more than 15 times higher control signal variance compared to P-control. On the

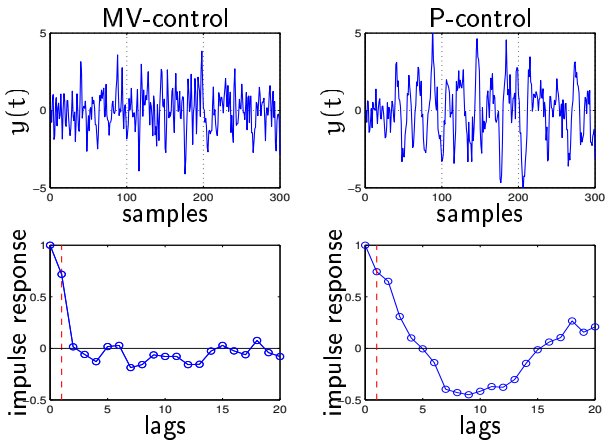


Figure 2.2: Example for minimum-variance control (left) and proportional control (right).

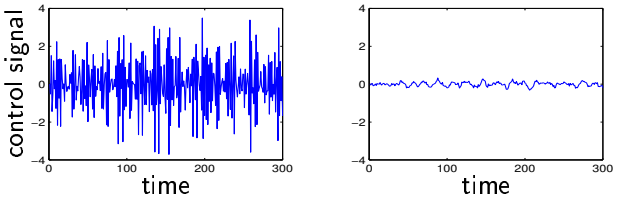


Figure 2.3: Control signals for MV-control (left) and P-control (right).

other hand, the output variance is only about two times lower than for P-control. From this it is understandable why a performance index lower than three often is considered as satisfactory due to practical constraints in the actuator.

2.5 Performance index using a process model

As stated in Chapter 1, a monitoring tool should not require knowledge of a process model in order to perform its task. However, simple process models are often available. For example if model-based controller tuning

was performed, such as the λ-tuning method which is widely spread in the pulp and paper industry nowadays. Such models are typically first order plus dead-time and are mostly fitted to experimental data. A natural question is then in what way the concept of a performance index can be used to assess control performance when such a simple process model is indeed available.

Furthermore, often the controller parameters are also known. If this is not the case, they can easily be estimated from normal operating data as will be shown in Chapter 6.

The use of a process model and the controller parameters has been proposed in (Eriksson and Isaksson, 1994) and (Isaksson, 1996). Such a knowledge allows the evaluation of performance indices for different control tasks and/or for different controller structures. Below a brief review of this concept is presented.

Another interesting model-based performance assessment approach using frequency performance measures has been proposed by Kendra and Cinar (1997). However, their approach requires closed-loop experiments and given design specifications for each loop.

The index for different control tasks

The original Harris index evaluates control performance when the control task is stochastic control. As shown by Eriksson and Isaksson (1994), the same concept can be used to assess the performance when the control task is setpoint tracking or input disturbance rejection. If the setpoint change or the input disturbance are assumed to be steps, the noise model is replaced by $1/(1-q^{-1})$. The actual output variance σ_y^2 can then no longer be calculated directly from data since a step disturbance may not have been present in the actual data set. The variance can instead be calculated from

$$\sigma_y^2 = \sum_{t=1}^{\infty} y^2(t) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \left|H(e^{i\omega})\right|^2 \, d\omega \qquad (2.15)$$

where $H(q^{-1})$ is the closed-loop transfer function from input disturbance or setpoint to the process output depending on what the control task in question is. The denominator of the performance index is calculated as usual by series expansion of $H(q^{-1})$ as described in Chapter 2. In order to do that, process, controller and noise model are needed. Note the interesting fact that if the noise model *is* assumed to be $1/(1-q^{-1})$, the minimum-variance term is

trivially given by $\sigma^2_{mv} = d$ since the series expansion of $1/(1 - q^{-1})$ is $1 + q^{-1} + q^{-2} + \ldots$.

The index for different control structures

The comparison of the actual output variance to minimum-variance may be overly pessimistic. What one rather would like to assess is how much the output variability can be decreased using the *current* control structure. For a PI-controller one would like to compute an index as

$$I_{PI} = \frac{\sigma^2_y}{\sigma^2_{PI}}, \tag{2.16}$$

where σ^2_{PI} is the minimum achievable output variance using a PI-controller. Here, in order to compute (2.16), process, controller and noise model have to be available too. The index (2.16) is obtained by minimising the expression for σ^2_{PI} with respect to the controller parameters. For a PI-controller one has to evaluate

$$\sigma^2_{PI} = \min_{K_c, T_i} \sigma^2_y = \min_{K_c, T_i} \frac{1}{2\pi} \int_{-\pi}^{\pi} \left| H(e^{i\omega}) \right|^2 d\omega \tag{2.17}$$

where $H(q^{-1})$ is the closed-loop transfer function as described above.

Note that one is not restricted to the current control structure. One can of course compare the current controller with another structure, for example to assess the possible benefit of adding a derivative part to a PI-controller.

A matrix of indices

Combining the indices from the two previous sections one can obtain a whole matrix of performance indices. In (Eriksson and Isaksson, 1994) and (Isaksson, 1996) three different control tasks and three different controller structures are proposed yielding the following index matrix:

	stoch. control	setp. track.	inp. dist. rej.
free contr. struct.			
restr. to PID			
restr. to PI			

From such a table it is obvious that the additional information leads to a more detailed assessment. The simplicity of a single performance index has disappeared somewhat. However, this is not really a disadvantage since the described method cannot be used as a first-level assessment on all loops since a process model is typically only available for a limited amount of loops. For such loops, a more detailed performance assessment should be considered.

It is interesting to note that the procedure here also – at least implicitly – delivers the optimal controller parameters in a least squares sense. Two remarks shall be made on this fact. Firstly, since optimal controller parameters are computed implicitly, a monitoring system should save these and on request be able to suggest them in case re-tuning was decided.

Secondly, for the calculation of the index denominator, some control design has to be performed implicitly. In (2.17) the controller parameters are optimised using a quadratic criterion. As an alternative, one could use any control design method which can be automised. Having achieved a certain design, the theoretical minimal variance can be computed and the index be evaluated. By that, one could create an individual performance index which takes a certain design method into account. If then controller re-tuning is requested, the new parameters are already available.

Assessment using other quantities

With the process and the controller available, it is also possible to compute other quantities which can be used for performance assessment. Examples are design variables as, for example, gain and phase margin, cross-over frequency, bandwidth or sensitivity function. Such information may be of great use for an engineer when scrutinising a certain loop. It is however more difficult to use it for automatic performance assessment since none of these values represents an absolute measure as the Harris index is. The estimation of the phase margin from routine operating data for performance assessment has been proposed by Müller (1986).

2.6 Other approaches based on indices

As already mentioned in Chapter 1, there are other performance indices proposed in the literature. Some of them will be reviewed very briefly for completeness.

Relative variance index (RVI). Bezergianni and Georgakis (2000) proposed a performance assessment index where the closed-loop performance is compared with minimum-variance control *and* open-loop control. The RVI is defined as

$$\mathrm{RVI} = \frac{\sigma^2_{\mathrm{OL}} - \sigma^2_{\mathrm{y}}}{\sigma^2_{\mathrm{OL}} - \sigma^2_{\mathrm{MV}}},$$

where σ^2_{OL} is the output variance if the controller is removed. The RVI is equal to zero if the current performance is equal to the open-loop performance and equal to one if the performance is equal to minimum-variance control. The price paid for the more sophisticated assessment is that one needs to know the process, controller and noise model. The authors suggest closed-loop identification using subspace identification techniques.

Idle index. Hägglund (1999) proposed a performance index which detects sluggish control loops. The idle index describes the relation between times of positive and negative correlation between the control signal and the process output increments, Δu and Δy respectively. It is defined as

$$\mathrm{I_i} = \frac{\mathrm{t_{pos}} - \mathrm{t_{neg}}}{\mathrm{t_{pos}} + \mathrm{t_{neg}}},$$

where $\mathrm{t_{pos}}$ and $\mathrm{t_{neg}}$ are updated every sampling instant (sampling period $\mathrm{T_s}$):

$$\mathrm{t_{pos}} = \begin{cases} \mathrm{t_{pos}} + \mathrm{T_s} & \text{if } \Delta u \Delta y > 0 \\ \mathrm{t_{pos}} & \text{if } \Delta u \Delta y \leq 0 \end{cases} \qquad \mathrm{t_{neg}} = \begin{cases} \mathrm{t_{neg}} + \mathrm{T_s} & \text{if } \Delta u \Delta y < 0 \\ \mathrm{t_{neg}} & \text{if } \Delta u \Delta y \geq 0 \end{cases}$$

Then, positive values of $\mathrm{I_i} > 0.4$ indicate sluggish control. The idle index should be evaluated when load disturbances occur in the loop.

A performance watchdog. Rhinehart (1995) proposed a very simple performance index which is the ratio of the process variance, calculated in two different ways,

$$\mathrm{I}_w = \frac{\sigma^2_{\mathrm{y}1}}{\sigma^2_{\mathrm{y}2}}.$$

The variance $\sigma^2_{\mathrm{y}1}$ is obtained from $\sigma^2_{\mathrm{y}1} = \frac{1}{\mathrm{N}-1}\sum_{\mathrm{i}=1}^{\mathrm{N}} \mathrm{d}_1^2(\mathrm{i})$ where $\mathrm{d}_1(\mathrm{i})$ is the deviation of sample i from the setpoint value. The variance $\sigma^2_{\mathrm{y}2}$

is obtained from $\sigma^2_{y1} = \frac{1}{2(N-1)} \sum_{i=1}^{N} d_2^2(i)$ where $d_2(i)$ is the distance between two consecutive samples. A performance problem is then indicated if both variance estimates differ such that the ratio is larger than $I_w = 3$.

Normalised settling time. Swanda and Seborg (1997) proposed to use the settling time T_s of a setpoint response, normalised by the apparent time-delay θ of the process. The index is defined as

$$I_{T_s} = \frac{T_s}{\theta}.$$

The authors define intervals of I_{T_s} which indicate satisfactory and non-satisfactory control respectively. They also show that the value of I_{T_s} is rather insensitive to the model order and model type for a wide range of process models.

PI-control performance index. Ko and Edgar (1998) suggested an index similar to the one described in Section 2.5. They compute the ratio of the actual variance and the minimum achievable one using a PI-controller. It is assumed that a process model is available. The noise model is then estimated using stochastic disturbance model realisation.

Chapter 3

A modified control performance index

The Harris approach as introduced in the last chapter has been shown to indicate actual control performance correctly in many reported cases. For an overview see (Harris *et al.*, 1999). The main principle of the minimum variance based index is to relate the actual performance to the best achievable performance when time-delay limitations are considered. Hence, if the time-delay is the main factor limiting performance in a control loop, the Harris index will give good results.

Another situation arises when the system time-delay is very short. In that case there may be other limitations which have a larger influence on the achievable performance. These other limitations can be taken into account by integrating their contributions to output variance when calculating a performance index. Tyler and Morari (1996) proposed a modification of the Harris index which takes performance limitations due to non-minimum phase zeros and unstable poles into account. They showed that neglecting these performance limitations can lead to an incorrect performance assessment. This modification requires knowledge of the unstable poles and zeros.

In order to accomplish a realistic performance bound that takes other limitations than time-delay into account, Kozub and Garcia (1993) suggested a performance index which allows an exponential decay of the closed-loop response after the dead time has run out. They consider this an important extension of the original method. The minimum variance benchmark was

found to be "not practical in many circumstances" (Kozub, 1996). This extension was also used and described by Huang and Shah (1997). However, a problem of the "user-defined benchmark" in (Huang and Shah, 1997) is that it requires knowledge of the disturbance model. Huang assumes that the disturbance model is either an integrator or has to be estimated from input-output data. The use of an integrator noise model for deterministic disturbances has also been discussed in (Eriksson and Isaksson, 1994).

We will here introduce an alternative problem formulation which allows both the specification of a *user-defined benchmark* and the calculation of the modified performance index using solely the measured output and the process time-delay. The concept of the *user-defined benchmark* implies that not all closed-loop system poles[1] are placed in the origin – as done when using minimum variance control – but one of them is placed arbitrarily by the user. This means that we do not assume a dead-beat system but allow an exponential decay which depends on where the free pole was placed. This modified minimum achievable variance will always be larger than the variance achieved by a minimum variance controller. Hence, our modified index will always be smaller than the Harris index.

The chapter is organised as follows. In Section 3.1, the Harris approach is re-derived in the pole-placement framework. The same notion is then used to formulate the proposed modified closed-loop system in Section 3.2. Also, a possible implementation is outlined. The choice of the closed-loop pole is discussed in Section 3.3. In order to determine the accuracy of the calculated index, it would also be desirable to compute its standard deviation. A method to do this is presented in Section 3.4. Examples from industrial processes are used to compare the old and the new index in Section 3.5. Finally, conclusions are given in Section 5.2.

3.1 An alternative derivation of the Harris index

In this section the performance index presented by Harris (1989) is re-formulated using a pole placement notion. The key-point in the approach

[1] Note that closed-loop system here refers to the transfer function from noise to process output. The poles of the transfer function from setpoint to output are not all located in the origin.

presented by Harris is that the controller cannot influence the process output before the dead-time has run out. Consider the block diagram in Figure 3.1. Let the process be described as

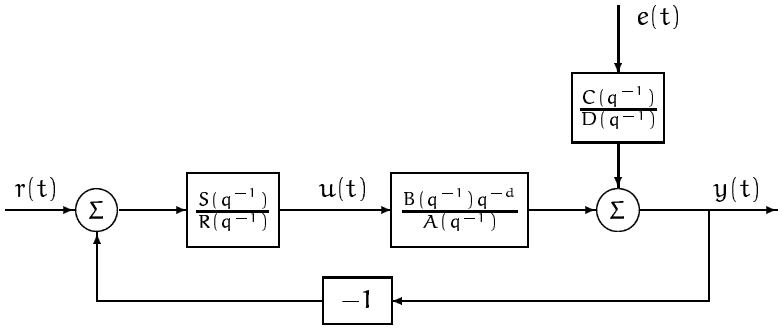


Figure 3.1: Closed-loop control loop.

$$y(t) = \frac{B(q^{-1})}{A(q^{-1})}q^{-d}u(t) + \frac{C(q^{-1})}{D(q^{-1})}e(t). \tag{3.1}$$

where $y(t)$ and $u(t)$ are the process output and the control signal and $e(t)$ is zero mean white noise with variance σ_e^2. Denote the degrees of the polynomials A, B, C and D as n_A, n_B, n_C and n_D respectively. We will assume that $n_B < n_A$ and $n_C < n_D$.

Closing the loop with a polynomial controller $F(q^{-1}) = \frac{S(q^{-1})}{R(q^{-1})}$ yields the closed-loop system

$$y(t) = \frac{ARC}{D[AR + BSq^{-d}]}e(t). \tag{3.2}$$

We assume here that the process is stable and minimum phase. Now, all poles shall be placed in the origin, which equals to applying a minimum variance controller. To achieve this, the controller denominator has to contain both the noise model poles and the process zeros $R = DB\tilde{R}_1$ where $\tilde{R}_1$ is the remainder of R. Inserting this controller into (3.1) yields

$$y(t) = \frac{A\tilde{R}_1C}{AD\tilde{R}_1 + Sq^{-d}}e(t). \tag{3.3}$$

The minimum variance closed-loop system,

$$y(t) = \tilde{R}_1\, e(t), \tag{3.4}$$

will be obtained if we require

$$AD\tilde{R}_1 + Sq^{-d} = CA. \tag{3.5}$$

Note that the polynomial $A(q^{-1})$ appears on both sides of the Diophantine equation (3.5). That means also $S(q^{-1})$ has to contain $A(q^{-1})$, i.e. $S = A\tilde{S}$. Hence the controller cancels all (stable) process poles. Inserting $S(q^{-1})$ yields the Diophantine equation

$$D\tilde{R}_1 + \tilde{S}q^{-d} = C. \tag{3.6}$$

With the orders $n_{\tilde{R}_1} = d - 1$ and $n_{\tilde{S}} = n_D - 1$, this equation has a unique solution, (Åström and Wittenmark, 1997), $\tilde{R}_1(q^{-1})$ and $\tilde{S}(q^{-1})$. This follows since $n_C \leq n_D + d - 1$ and q^{-d} and $D(q^{-1})$ do not have common factors. Note that the time-delay d is at least one sample in all discrete-time processes originating from sampled continuous-time ones. Since $\tilde{R}_1$ is of order $d - 1$, its parameters can easily be determined as the first $d - 1$ samples of the the series expansion of

$$\frac{C}{D} = \tilde{R}_1 + q^{-d}\frac{\tilde{S}}{D}. \tag{3.7}$$

The minimum achievable variance can be calculated as

$$\sigma^2_{mv} = \left(h_0^2 + h_1^2 + \ldots + h_{d-1}^2\right)\sigma_e^2 \tag{3.8}$$

where h_i are the coefficients of $\tilde{R}_1$. The performance index will then be obtained as the ratio of the actual output variance σ_y^2 and the minimum achievable variance

$$I_p = \frac{\sigma_y^2}{\sigma^2_{mv}}. \tag{3.9}$$

The key-point in the Harris approach is that the process noise model does not have to be known. This implies that the first $d - 1$ coefficients of the impulse response of the closed-loop transfer function from disturbance $e(t)$ to output $y(t)$,

$$y(t) = \frac{T(q^{-1})}{N(q^{-1})}e(t), \tag{3.10}$$

are exactly equal to the first $d-1$ coefficients of the impulse response of the noise model C/D. The closed-loop transfer function in (3.10) can be estimated as a time-series model for the measured process output $y(t)$.

3.2 The modified performance index

Derivation

Let us now assume that we want to place one pole at $q = \mu$ and the other poles in the origin. That means we require the following closed-loop system

$$y(t) = \frac{\tilde{R}_2(q^{-1})}{(1-\mu q^{-1})} e(t). \tag{3.11}$$

With the same assumptions for $S(q^{-1})$ and $R(q^{-1})$ as in the previous section, i.e. $R = DB\tilde{R}_2$ and $S = A\tilde{S}$, we have to solve the Diophantine equation

$$D\tilde{R}_2 + \tilde{S}q^{-d} = C(1-\mu q^{-1}). \tag{3.12}$$

Since $n_C + 1 \leq n_D + d - 1$ is satisfied for $n_D > n_C$ and $D(q^{-1})$ and q^{-d} do not have any common factors, Equation (3.7) has a unique solution

$$\begin{aligned} &\tilde{R}_2(q^{-1}) \quad \text{with} \quad n_{\tilde{R}_2} = d-1 \\ &\tilde{S}(q^{-1}) \quad \text{with} \quad n_{\tilde{S}} = n_D - 1. \end{aligned} \tag{3.13}$$

It remains to show that $\tilde{R}_2$ still can be calculated from a time series model. Consider therefore Equation (3.12) when divided by $D(q^{-1})$

$$\frac{C}{D}(1-\mu q^{-1}) = \tilde{R}_2 + \frac{\tilde{S}}{D} q^{-d}. \tag{3.14}$$

As shown above, $\tilde{R}_2$ is of degree $d-1$ and can thus be calculated as the first $d-1$ elements of the series expansion of the left side of (3.14).

Implementation

Since the control has no influence before time $t = d$, the impulse response of $\tilde{R}_2/(1-\mu q^{-1})$ has to coincide with the impulse response of $\tilde{R}_1$ for $t < d$. For

$t \geq d$ we have the impulse response $h(t) = h(d-1)\mu^{t-d+1}$. The output variance of $y(t)$ can now be calculated as

$$\begin{aligned}
\sigma^2_{mod} &= \left(\sum_{i=0}^{d-1} h_i^2 + h_{d-1}^2 \mu^2 \sum_{i=0}^{\infty} \mu^{2i} \right) \sigma_e^2 \\
&= \left(\sum_{i=0}^{d-1} h_i^2 + h_{d-1}^2 \frac{\mu^2}{1-\mu^2} \right) \sigma_e^2 \\
&= \sigma^2_{MV} + h_{d-1}^2 \frac{\mu^2 \sigma_e^2}{1-\mu^2} = \sigma^2_{MV} + \sigma^2_{\mu}. \qquad (3.15)
\end{aligned}$$

Note that this variance has two parts, the minimum variance and the contribution from the first-order decay. Thus the implementation of the modified index I_{mod} can be done using the following steps:

1. Estimate a time-series model for the closed-loop system $T(q^{-1})/N(q^{-1})$. The series expansion of this is equal to the one of $C(q^{-1})/D(q^{-1})$ for the first $k-1$ samples.
2. Equation (3.15) showed that only $\tilde{R}_1$ is needed for the calculations. Determine $\tilde{R}_1$ as the first $k-1$ coefficients of the series expansion of the identified time series model.
3. Calculate the minimum variance σ^2_{MV} as in (3.8).
4. Calculate the modified minimum variance as in (3.15).
5. Calculate the actual output variance σ^2_y.
6. Calculate the modified performance index as

$$I_{mod} = \frac{\sigma^2_y}{\sigma^2_{mod}}.$$

Connection to the original performance index

An interesting property of the presented modified index and the original Harris index is their ratio. Consider therefore

$$r = \frac{I_p}{I_{mod}} = \frac{\frac{\sigma_y^2}{\sigma_{MV}^2}}{\frac{\sigma_y^2}{\sigma_{mod}^2}} = \frac{\sigma_{mod}^2}{\sigma_{MV}^2} = \frac{\sigma_{MV}^2 + \sigma_\mu^2}{\sigma_{MV}^2}. \tag{3.16}$$

If we assume the smallest possible time-delay $k = 1$, this ratio can be calculated analytically from (3.15) and depends only on the choice of the closed-loop pole μ:

$$r^* = \frac{1}{1-\mu^2}. \tag{3.17}$$

It reveals that the Harris index is always 33% larger than the modified index if we choose $\mu = 0.5$. For $\mu = 0.7$ the Harris index is 96% larger. From Equation (3.16) it can also be seen that the ratio of the indices decreases (towards one) if the process time-delay increases. Hence we can conclude that the modified index becomes more significant for systems with short time-delays, as desired in the introduction.

Note that the modified index may in fact be smaller than one, thereby indicating that the current controller is doing better than required. This is due to the fact that a more realistic benchmark than perfect control (minimum-variance) is used.

3.3 Choice of closed-loop pole

A modified performance index was introduced which requires the specification of a closed-loop pole. In this section, different approaches for the choice of this closed-loop pole μ are discussed. As there are many possibilities to choose μ, no final solution can be given and some ideas will be outlined. The first one does not require any extra knowledge and is based on robustness considerations only. The second approach makes use of available additional process knowledge. Note again that the original minimum-variance based index corresponds to $\mu = 0$.

Choice of closed-loop pole based on robustness margins

We will here derive a lower bound on the closed-loop pole μ. It is based on an imaginary controller design using robustness arguments. Recall now that the time-delay plays a central role in the concept of the Harris index. Assume that the correct process time-delay is only known with an accuracy of ± 1 sample, i.e. let the real process be described as

$$G(q^{-1}) = \frac{B(q^{-1})}{A(q^{-1})} q^{-d}, \tag{3.18}$$

and the process model as

$$\hat{G}(q^{-1}) = \frac{B(q^{-1})}{A(q^{-1})} q^{-\hat{d}}, \tag{3.19}$$

where $d = \hat{d} \pm 1$. The following theorem gives a lower bound on the closed-loop pole.

Theorem 3.1. *Let the real process and its model description be given by (3.18) and (3.19). The controller is designed (using $\hat{G}$) such that one closed-loop pole is located in $q = \mu$ and the others in the origin. Then, if the closed-loop pole μ is chosen as*

$$\mu \geq 0.5, \tag{3.20}$$

the closed-loop system, now using $G(q^{-1})$ instead of the model, has a guaranteed gain margin $GM \geq 2$. □

Proof: See Appendix A.

The assumption that the process time-delay is not known exactly is reasonable. As a matter of fact, the assumed variation of ± 1 sample is rather modest. The main reason being that the delay is often inaccurate since it has to be estimated in most cases. Furthermore the physical time-delay is seldom an exact multiple of the sampling interval, hence rounding errors are introduced when determining the number of samples of dead-time.

Use of additional process knowledge

Another reasoning can enlighten what the lower bound for μ derived in the previous section means for the case of a correctly estimated time delay.

Assume that the system is sampled with a reasonable sampling interval, i.e. approximately 5 to 10 times the slowest process time-constant. When determining the discrete-time pole λ from the continuous-time counterpart we have to calculate

$$\lambda = \exp\left(-\frac{T_s}{\tau}\right) \tag{3.21}$$

where T_s is the sampling time and τ is the slowest process time-constant. Inserting $\tau \in [5 \ldots 10] \cdot T_s$ into (3.21) yields $\lambda \in [0.81 \ldots 0.90]$. Using $\mu = 0.5$ as proposed above would mean to compare the current controller to a controller which speeds up the system $[3.5 \ldots 7]$ times. As this is rather much, one may choose μ to be even larger than 0.5. Clearly, there are processes which may be speeded up that much but a reasonable choice which often gives good results in process industry is to speed up the system not more than two to four times.

As a consequence, a constraint on μ could be formulated if it was known to what extent the process can be speeded up. This would result in a benchmark which is quite realistic for that specific loop. Assume that the slowest process time-constant, τ, is known approximately and one allows to speed up the system three times. Then a reasonable choice of the closed-loop pole would be $\mu \geq \exp\left(-\frac{3T_s}{\tau}\right)$. As mentioned in the introduction, here the question of the available process knowledge arises. In our opinion it is reasonable to assume that knowledge of the open-loop dominant process time-constant is not less likely than knowledge of the time-delay. When such a knowledge is available, there are several guidelines how to place the dominant closed-loop pole. See (Panagopoulos *et al.*, 1997) for a discussion on this issue.

It would also be highly desirable to take the control action into account. A drawback of minimum variance control is that it yields rather excessive control actions. These may not be able to apply due to constraints in the manipulated variable. In this case, the input constraints may limit control performance rather than the system time-delay. Hence the actual process is compared to a sometimes over-optimistic benchmark. The incorporation of the control action, however, would require additional process knowledge.

3.4 How accurate is the index?

An estimate of the closed-loop performance using an index is of course of little value without some indication of its accuracy. Therefore the standard deviation of the estimated index should be presented along with the index.

This issue was also discussed by Desborough and Harris (1992) where approximate statistical moments of a normalised performance index are calculated using Taylor series approximation. We will here use a similar approach.

Generally speaking, the calculated index is a function of the estimated time-series model parameters $\theta = (\theta_1, \ldots, \theta_n)^T$, (e.g. the parameters of $T(q^{-1})$ and $N(q^{-1})$ in (3.10)); i.e.

$$I = f(\theta). \tag{3.22}$$

The function $f(\theta)$ is nonlinear and the exact calculation of the standard deviation is therefore difficult. However, the variance of (3.22) can be approximated using the Gauss approximation formula

$$\mathrm{Var}(I) \approx f'(\theta)\mathrm{Cov}(\theta)[f'(\theta)]^T \tag{3.23}$$

where

$$f'(\theta) = \left(\frac{\partial f(\theta)}{\partial \theta_1}, \frac{\partial f(\theta)}{\partial \theta_2}, \ldots, \frac{\partial f(\theta)}{\partial \theta_n}\right)$$

and $\mathrm{Cov}(\theta)$ is the covariance matrix of the estimated parameters. In most system identification software packages the covariance matrix $\mathrm{Cov}(\theta)$ is delivered along with the estimated parameters θ. For some cases it should be possible to calculate $f'(\theta)$ analytically. However, in this chapter the standard deviation will be calculated using a numerical approximation of $f'(\theta)$. Each $f'(\theta_i)$ is calculated by perturbing a certain parameter θ_i and then using the approximation

$$f'(\theta_i) \approx \frac{f(\theta_i + \frac{\Delta}{2}) - f(\theta_i - \frac{\Delta}{2})}{\Delta},$$

where Δ is the size of the perturbation. In (Eriksson, 1995) it it shown that it is rather complicated to calculate the covariance for σ_y^2 and that it still remains to calculate some cross-correlations. A solution to this problem is to calculate σ_y^2 using the Markov parameters $\{h_i\}, i = 0, \ldots, \infty$. The performance index can then be calculated from

$$I = \frac{\sigma_y^2}{\sigma_{MV}^2} = \frac{\frac{1}{N-1}\sum_{i=1}^{N}(y_i - \bar{y})^2}{(\sum_{i=0}^{k-1} h_i^2)\sigma_v^2} \approx \frac{(\sum_{i=0}^{\infty} h_i^2)\sigma_v^2}{(\sum_{i=0}^{k-1} h_i^2)\sigma_v^2} = \frac{\sum_{i=0}^{\infty} h_i^2}{\sum_{i=0}^{k-1} h_i^2}. \tag{3.24}$$

where $\bar{y}$ is the signal mean. Note that this expression does not involve the estimated noise variance σ_e^2 any more. The approximation is used to indicate that both expressions are theoretically equal. They will normally differ and none of them can be claimed to be the "true" one. The problem of how to calculate the actual output variance using (3.24) has already been discussed in Chapter 2. Using the Lyapunov approach the value of the output variance and hence the performance index only depend on the estimated model, i.e. θ. Note that the Harris index has to be calculated again and again for every parameter perturbation. However, the computations involved are simple and fast such that the additional computational effort is not severe.

Note that it may be misleading to compare the relative uncertainty of the Harris index and the modified index. This is because the value of the modified index is always lower than the Harris index. What matters is the absolute value of the uncertainty, giving a measure of the reliability of the performance index.

Another important remark has to be made here. It is easy to see that the original performance index, as described in Section 3.1 may vary considerably in some cases. If there is a short dead-time, then the performance index will be rather sensitive to an uncertain estimate of the number of samples of time-delay. Also, the time-delay may be time-varying which is often the case in many processes, e.g. paper machines. This sensitivity will be smaller when the proposed modified index is used. This is because the additional exponential decay of the impulse response decreases the impact of time-delay variations on the index value. This effect is illustrated in Figure 3.2 where the performance index is evaluated for a process with uncertain time-delay. It can be seen that the index becomes less sensitive to uncertain time-delay with increasing value of the closed-loop pole μ.

3.5 Application to industrial examples

In this section the described indices along with their standard deviations will be evaluated on industrial data sets. Data from two different processes are investigated.

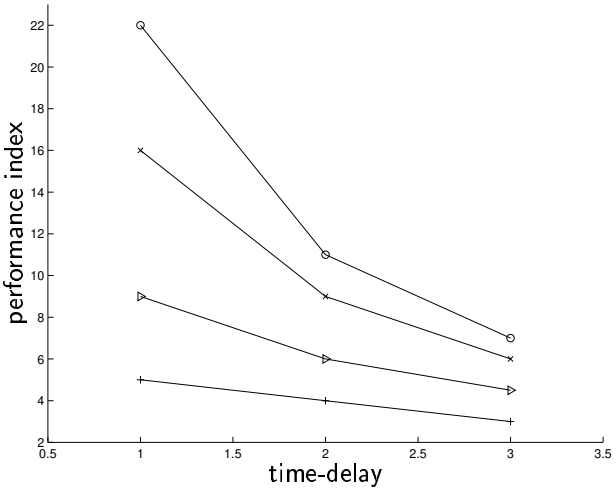


Figure 3.2: Performance index for a process with varying time-delay and different values of the closed-loop pole μ. $\mu = 0$(o), $\mu = 0.5$(x), $\mu = 0.77$(▷), $\mu = 0.88$(+)

Frontend-temperature control of a lime kiln

The data analysed in this section was collected at a Swedish pulp mill. It describes the frontend temperature in a lime kiln. The system which is controlled by a model-predictive controller has a sampling time of 2 minutes, an open-loop dominant time-constant of about 30 minutes and no dead-time. Since the process is controlled by a digital controller and the calculations of the model-predictive controller are time-consuming, there is, however, a process time-delay of two samples. Note that if the sampling rate is changed, there will still be a time-delay of two samples. This sampling-independent behaviour of the time-delay for systems with no or very short dead-time is important when applying the described indices. In this case, the proposed modification will become significant.

The measured control error of the frontend-temperature of the lime kiln is shown in Figure 3.3. The different calculated indices for this example are shown in Table 3.1. The Harris index for this system is rather high, indicating that the output variance can theoretically be decreased significantly. The

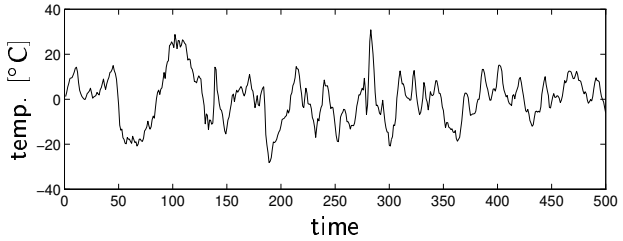


Figure 3.3: Measured control error of the frontend-temperature control in a lime kiln. Sampling time $T_s = 2$ [min].

pole μ	T_s [min]	Harris	mod. index
0.5	2	4.99 ± 0.25	3.95 ± 0.21
0.77	2	4.99 ± 0.25	2.31 ± 0.13
0.88	2	4.99 ± 0.25	1.34 ± 0.08
0.5	1	10.92 ± 0.38	9.27 ± 0.36
0.88	1	10.92 ± 0.38	3.84 ± 0.21
0.94	1	10.92 ± 0.38	2.15 ± 0.13

Table 3.1: Performance indices (given with their estimated standard deviations) for the control of the lime kiln process for different choices of the closed-loop pole μ and the sampling time T_s.

modified performance index using $\mu = 0.5$ results in a smaller but still high value. From practical experience it is known that the process cannot be speeded up too much. Since the open-loop dominant time-constant of the process is approximately 30 minutes, the sampling of 2 minutes is rather fast. This constellation corresponds to a discrete-time pole of 0.94. Requiring $\mu = 0.5$ means thus that the system has to become more than 10 times faster. For this process it is not likely to speed up the system more than two times resulting in $\mu \geq 0.88$. The resulting modified index is significantly smaller than the Harris index. If one would like to keep the open-loop time-constant, the modified index would become smaller than one which means that the actual performance is better than the required one. The different impulse responses for $\mu = 0.88$ are plotted in Figure 3.4. Even if we allow

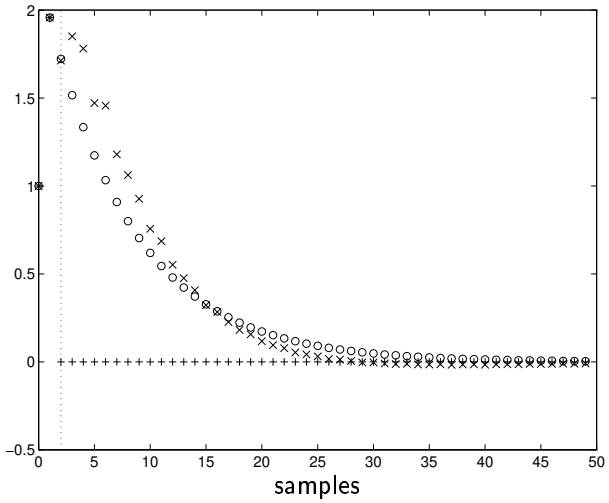


Figure 3.4: Impulse responses of the frontend-temperature control loop. Time-series model ($\times$), minimum variance benchmark ($+$) and modified benchmark (o). Closed-loop pole $\mu = 0.88$, sampling time $T_s = 2$ [min].

to speed up the system four times, which corresponds to a closed-loop pole at $\mu = 0.77$, it can be seen that the modified index is significantly smaller than the Harris index.

The actual data set was originally collected with a sampling interval of one minute. If we consider the performance of the system when applying this faster sampling rate, the indices differ even more. For the open-loop dominant time-constant of about 30 minutes, we get an open-loop pole at 0.97. A closed-loop pole at $\mu = 0.5$ corresponds to speeding up the system more than 20 times. Table 3.1 shows the resulting indices when allowing to speed up the process two or four times which corresponds to closed-loop poles at 0.94 and 0.88 respectively.

For an autonomous control performance tool one has to specify an alarm threshold on the calculated performance index. This choice is a trade-off between unnecessary false alarms and the detection of control loops which perform badly. For industrial acceptance, the threshold must not be too low. Let us here for instance assume a value of three. Then the results

from Table 3.1 show that only the modified index will correctly indicate good control performance if the sampling rate and the open-loop dominant time-constant are taken into account.

Tray temperatures in a distillation column

This example deals with data originating from three different distillation columns at Shell. Kozub and Garcia (1993) offered this data to the academic community in order to stimulate research on performance monitoring. Part of this data was also analysed in connection with control performance monitoring in (Vishnubhotla *et al.*, 1997). We will here consider two of the available data sets which are referred to as Column 1 and Column 3. Part of the measured tray temperatures are plotted in Figure 3.5. As can be

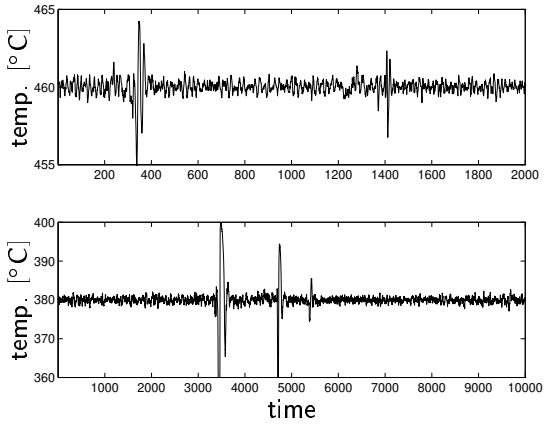


Figure 3.5: Tray temperature in a Shell distillation column. Column 1 (top) and Column 3 (bottom).

seen in the figure, there are some rather large temperature deviations in the data. These are mainly caused by disturbances. The sampling time here is one minute, and the open-loop dominant time-constant is about three minutes. The process time-delay is determined to two samples. All calculated indices are shown in Table 3.2. The data of Column 1 is sampled very slowly compared to the open-loop dominant time-constant. For that reason, the modification does not give significantly different results. The sampled

Data	pole μ	Harris	mod. index
Col.1	0.5	2.60 $\pm$ 0.07	2.21 $\pm$ 0.07
Col.3	0.5	4.51 $\pm$ 0.15	4.24 $\pm$ 0.14
Col.3	0.73	4.51 $\pm$ 0.15	3.71 $\pm$ 0.13

Table 3.2: Performance indices of the distillation columns.

open-loop pole is located at 0.72 which means that $\mu = 0.5$ corresponds to speeding up the process two times. This is a reasonable assumption.

In Column 3 we have a significantly larger dead-time. The sampling time is one minute and the open-loop dominant time-constant is about 16 minutes. The time-delay was determined to be seven samples. The numerous disturbances result in rather different indices dependent on the chosen data batch. The choice is necessary since the data batch contains about 90,000 samples. Here, 10,000 samples were used to calculate the indices. The open-loop pole of this fast-sampled loop is located at 0.94. Hence, a closed-loop pole at $\mu = 0.5$ corresponds to speeding up the process about 11 times. When we assume that the process should be speeded up at most five times, which means $\mu = 0.73$, then the modified index is still not significantly lower than the Harris index as shown in Table 3.2. In this case the dead-time is the main performance limiting factor, hence the Harris index does give a reasonable result and will be close to the modified index as mentioned in Section 3.2. The impulse responses for $\mu = 0.73$ are shown in Figure 3.6.

To summarise, Column 1 has a short time-delay but too slow sampling for the modified index to be significantly lower than the Harris index. Column 3 has relatively speaking faster sampling but here the dead-time is long enough to completely dominate the achievable performance. Hence for both these examples the Harris index gives a sufficiently good assessment of achievable control performance.

Summary

The control performance index described in Chapter 2 may be overly pessimistic when used for processes with short or no dead-time. A modified benchmark has therefore been applied, allowing for one closed-loop pole outside the origin. The idea has been inspired by Kozub and Garcia (1993)

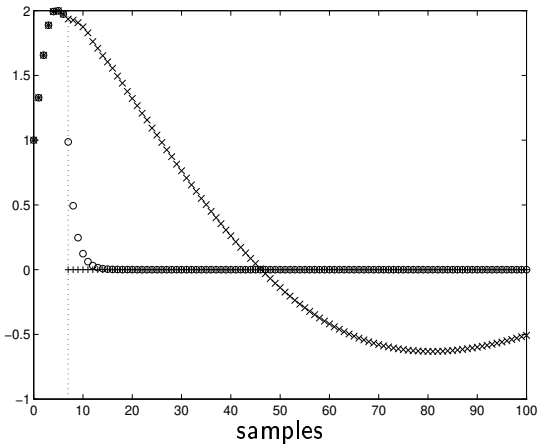


Figure 3.6: Impulse responses of the tray temperature for Column 3. Time-series model ($\times$), minimum variance benchmark ($+$) and modified benchmark (o).Closed-loop pole $\mu = 0.73$.

and was further derived and applied to industrial data. It was shown that such a modified performance index can be computed based on measurements of the process output and knowledge of the time-delay only (i.e. the same information as for the original Harris index). No knowledge about the noise model is required. Also a practical algorithm for industrial application has been discussed and several other important practical issues have been addressed.

Of course it also remains to choose the free closed-loop pole. Some discussion on this matter was presented. Firstly, it was argued that for many plants in the process industry it is inconceivable to speed up the response more than, say two to four times. Approximate knowledge of the open-loop time-constant is then enough to determine a lower bound on the discrete-time closed-loop pole. Secondly, a robustness bound on the sampled closed-loop pole was referred to with respect to modelling errors in the time-delay and the process gain. According to this bound, the sampled closed-loop pole should always be larger than 0.5.

The described modified performance index is less sensitive to uncertain

or time-varying time-delays. This feature makes the proposed modification interesting when thinking of industrial implementation. In order to increase the reliability of an automatic performance measure, it was shown how to calculate a standard deviation along with the index. The additional computational effort needed is modest.

Chapter 4

Assessment of the sampling rate in control systems

A modification of the minimum-variance based performance index was described in the last chapter. It was found that the performance index is very sensitive to the actual sampling interval in cases where there is no or only a very short physical dead-time. Therefore, it would be desirable to assess automatically – by computing a performance index – if the present controller uses an appropriate sampling frequency. Too slow sampling may cause an unnecessarily large output variance. This will, however, typically not be picked up by the performance index, unless it is computed using data sampled faster than the controller sampling rate. Notice that also too fast sampling may be undesirable (e.g. due to wear of actuators), but is less likely to occur in process industry.

Hence, one would like to be able to evaluate the performance index at sampling rates faster than the one used for data collection. Thereby, it is possible to assess the benefit of increasing the sampling rate, without disturbing the process itself. As a byproduct one also can detect if the process is controlled unnecessarily fast which may imply excessive input actions.

The calculation of the original and modified Harris index requires a time-series model of the process output. In this chapter a method is described which calculates a re-sampled time-series model for the new sampling rate based on the available model estimated from data. Wahlberg *et al.* (1993) proposed a method which allows the direct determination of the closed-loop

system for a decreased sampling rate. A similar result was presented by MacGregor (1976). Here modifications necessary to handle also increased sampling rate are described.

The following information will be needed in order to be able to use the methods presented here. First of all, the measured process output (under routine operation) has to be available with the same sampling rate as used by the discrete-time controller. Secondly, one needs to know the dead-time of the process. This information is sufficient to be able to compute the Harris index. In order to apply the modification suggested in Chapter 2, one may use some additional information, for example an approximate estimate of the dominant time constant of the process.

The remainder of the chapter is organised as follows. Two methods for re-sampling of an ARMA model, one exact and one approximate, are proposed in Sections 4.1 and 4.2. Both methods are illustrated on two simple numerical examples in Section 4.3. The method for calculating the control performance index for different sampling rates is presented in Section 4.4. In Section 4.5 the described approach is evaluated on two data sets from Swedish pulp and paper industry. Finally, a conclusion section finishes the chapter.

4.1 Exact re-sampling of an ARMA model

This section deals with the problem of calculating a time-series model for a new sampling rate. Even though the interpolation problem is of main interest here, both, interpolation and decimation can be handled with the same algorithm.

The problem of calculating the continuous-time counterpart of a discrete-time ARMA model has also been dealt with in (Söderström, 1991), where several exact and some approximate methods are described. Here, it is not necessary to go via a continuous-time model in order to calculate the re-sampled ARMA model. Given an ARMA model with the autocorrelation function (ACF) r_{y1}, an algorithm is sought which delivers a re-sampled ARMA model with a new ACF, r_{y2}. In common lags, the two ACFs are required to be equal.

Consider now the following (original) ARMA model:

$$\begin{aligned} y(kh_1) &= \frac{E(q^{-1})}{D(q^{-1})}e(kh_1) \\ &= \frac{1+e_1q^{-1}+\ldots+e_{n-1}q^{-(n-1)}}{1+d_1q^{-1}+\ldots+d_nq^{-n}}e(kh_1) \\ & E\{e^2(kh_1)\} = \sigma_e^2, \end{aligned} \tag{4.1}$$

where it is assumed that $\deg E < \deg D$. Let the sampling interval be denoted by h_1. Then a possible state-space representation is the controller canonical form:

$$\begin{aligned} x(kh_1+h_1) &= \mathbf{F}_{d1}x(kh_1)+\varepsilon(kh_1+h_1) \\ &= \mathbf{F}_{d1}x(kh_1)+G_{d1}w_1(kh_1) \\ y(kh_1) &= Hx(kh_1) \end{aligned} \tag{4.2}$$

where

$$\begin{aligned} \mathbf{F}_{d1} &= \begin{bmatrix} -d_1 & -d_2 & \cdots & -d_n \\ 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 1 & 0 \end{bmatrix} \\ \varepsilon(kh_1+h_1) &= \begin{bmatrix} e(kh_1+h_1) & 0 & \ldots & 0 \end{bmatrix}^T \\ G_{d1} &= \begin{bmatrix} 1 & 0 & \ldots & 0 \end{bmatrix}^T \\ H &= \begin{bmatrix} 1 & e_1 & \ldots & e_{n-1} \end{bmatrix} x(kh_1) \end{aligned}$$

and the noise covariance

$$\mathbf{R}_{d1} = E\left\{G_{d1}w_1^2(kh_1)G_{d1}^T\right\}$$

contains zeros except for $\mathbf{R}_{d1}(1,1) = \sigma_e^2$. In (4.2) the substitution $w_1(kh_1) = \varepsilon(kh_1+h_1)$ is made, which does not change the statistical properties of $y(kh_1)$.

The system (4.2) shall now be re-sampled with the new sampling rate $h_2 = h_1/\kappa$ where $\kappa > 1$. The decimation case ($\kappa < 1$) is easily handled using an algorithm described in (Wahlberg *et al.*, 1993). A similar result was

also presented by MacGregor (1976). However, even the decimation case can be handled by the method proposed here.

The new state-space model is now given by

$$\begin{aligned} x(kh_2 + h_2) &= \mathbf{F}_{d2}x(kh_2) + w_2(kh_2) \\ y(kh_2) &= Hx(kh_2), \end{aligned} \tag{4.3}$$

As mentioned in the beginning of this section, the requirement on the re-sampled model is that its ACF coincides with that of the original one for common lags. To calculate the ACF of the process output $y(t)$ it is first necessary to compute the state covariance matrix $\mathbf{P} = E\{x(kh)x(kh)^T\}$ for the respective models. This can be done solving the Lyapunov equation

$$\mathbf{P} = \mathbf{F}_d\mathbf{P}(\mathbf{F}_d)^T + \mathbf{R}_d. \tag{4.4}$$

Then the ACFs for (4.2) and (4.3) can be obtained (Söderström, 1994) as

$$r_{y1}(\nu) = H\mathbf{F}_{d1}^{\nu}\mathbf{P}_{d1}H^T;\ r_{y2}(\mu) = H\mathbf{F}_{d2}^{\mu}\mathbf{P}_{d2}H^T.$$

Require then ACFs to be equal in common sampling points, i.e. $r_{y1}(\nu) = r_{y2}(\kappa\nu)$ – where $\kappa\nu$ integer. This is obviously satisfied if $\mathbf{P}_{d1} = \mathbf{P}_{d2}$ and $\mathbf{F}_{d1} = \mathbf{F}_{d2}^{\kappa}$. The latter condition yields

$$\mathbf{F}_{d2} = \exp\left(h_2\left[1/h_1\log(\mathbf{F}_{d1})\right]\right) = \left[\mathbf{F}_{d1}\right]^{1/\kappa}. \tag{4.5}$$

For $r_{y1}(\nu) = r_{y2}(\kappa\nu)$ to hold, it remains to make sure that the state covariances $\mathbf{P}_{d1}$ and $\mathbf{P}_{d2}$ are equal. This is here done by adjusting the covariance matrix $\mathbf{R}_{d2} = E\left\{w_2(kh_2)w_2(kh_2)^T\right\}$ for the re-sampled noise $w_2(kh_2)$. If the Lyapunov equation (4.4) is first solved for (4.2) yielding $\mathbf{P}$, the noise covariance matrix for (4.3) can be calculated from $\mathbf{R}_{d2} = \mathbf{P} - \mathbf{F}_{d2}\mathbf{P}\left(\mathbf{F}_{d2}\right)^T$. Notice that, for very short sampling periods, numerical errors may be introduced here because $\mathbf{R}_{d2}$ is calculated as the difference of two almost equal quantities, see (Söderström, 1994).

Unfortunately, the noise covariance matrix $\mathbf{R}_{d2}$ is typically a full-rank matrix, why (4.3) does not describe an ARMA model. Therefore it is necessary to use spectral factorisation in order to obtain the desired new ARMA-model description. This is equivalent to calculating the innovations form of (4.3), which is given by

$$\begin{aligned} x(kh_2 + h_2) &= \mathbf{F}_{d2}x(kh_2) + Kv(kh_2), \\ y(kh_2) &= Hx(kh_2) + v(kh_2). \end{aligned} \tag{4.6}$$

Note that the state-vector x in (4.6) is no longer the same as in (4.2). The output of system (4.6) has the same spectral density as the output of (4.3) if the Kalman gain K is calculated as

$$K = \mathbf{F}_{d2}\mathbf{X}H^T\left(H\mathbf{X}H^T\right)^{-1}$$

where $\mathbf{X}$ solves the following Riccati Equation

$$\begin{aligned}\mathbf{X} &= \mathbf{F}_{d2}\mathbf{X}(\mathbf{F}_{d2})^T + \mathbf{R}_{d2} \\ &\quad - \mathbf{F}_{d2}\mathbf{X}H^T\left(H\mathbf{X}H^T\right)^{-1}H\mathbf{X}(\mathbf{F}_{d2})^T.\end{aligned} \tag{4.7}$$

Finally one gets $Ev(kh_2)^2 = H\mathbf{X}H^T$ for the innovations $v(kh_2) = y(kh_2) - Hx(kh_2)$ and the re-sampled ARMA model is given as

$$\begin{aligned}y(kh_2) &= \frac{\tilde{E}(q^{-1})}{\tilde{D}(q^{-1})}v(kh_2) \\ &= \left[1 + H\left(q\mathbf{I} - \mathbf{F}_{d2}\right)^{-1}K\right]v(kh_2).\end{aligned}$$

An interesting phenomenon is that, even though both processes (4.2) and (4.3) now describe the same stochastic process $y(t)$, there is generally no guarantee that the new noise covariance $\mathbf{R}_{d2}$ is positive definite. However, the indefinite noise covariance matrix does not prevent the Riccati Equation from having a solution for certain cases. In particular, this appears to be the case when the underlying continuous-time model exists. See (Söderström, 1991) for a discussion in the context of calculating the continuous-time counterpart of an ARMA model.

An extensive treatment of the problem of the existence of solutions of the discrete-time algebraic Riccati Equation (DARE) was presented by Hassibi (1996). The author proves that there is a unique, stabilising solution to the DARE for all initial conditions that ensure that the output $y(t)$ behaves like a stochastic process.

Hence, even though the method proposed here does not involve the underlying continuous-time system, there will typically be no solution in cases where there is no continuous-time counterpart to the discrete system in question. Unfortunately, there is no guarantee that the data collected originates from a sampled continuous-time process. On the contrary, a continuous-time plant controlled by a discrete-time controller may well constitute a closed-loop system that has no continuous-time counterpart. It may still be

interesting, however, to evaluate the achievable minimum variance at another sampling interval than the one used for data collection. In the next section a re-sampling procedure is described which yields a result even though an underlying continuous-time process is missing.

Another drawback is that the proposed method cannot handle poles on the unit circle. However, methods for this case are described in (Söderström, 1991).

4.2 Using deterministic re-sampling

Even though the numerical tools used in the previous section are well-known, it would be advantageous to have a method which avoids the solution of Riccati and Lyapunov equations. Note that the method presented above is *exact* in the sense that the auto-covariance functions of the original system and the re-sampled system are exactly equal at the lags of the slower sampled system.

A simpler but approximate way of approaching the re-sampling problem would be to treat the system as re-sampled with a zero-order hold circuit, i.e. neglecting the stochastic nature of the system. This method, used to obtain a continuous-time ARMA model from a sampled one, is also considered in (Söderström, 1991). In order to compensate for the error made by neglecting the stochastic nature of the system, one has to adjust the noise-covariance matrix of the re-sampled system.

Consider the state-space model for the original sampling rate h_1:

$$\begin{aligned} x(kh_1 + h_1) &= \mathbf{F}_{d1}x(kh_1) + G_{d1}w_1(kh_1) \\ y(kh_1) &= Hx(kh_1). \end{aligned} \tag{4.8}$$

The re-sampled system is given by

$$\begin{aligned} x(kh_2 + h_2) &= \mathbf{F}_{d2}x(kh_2) + G_{d2}w_2(kh_2) \\ y(kh_2) &= Hx(kh_2), \end{aligned} \tag{4.9}$$

where $\mathbf{F}_{d2}$ is obtained from (4.5). An expression for the vector G_{d2} can be found by considering the underlying continuous-time system $\dot{x}(t) = \mathbf{A}x(t) + G_c v(t)$ using

$$G_{d2} = \int_0^{h_2} e^{\mathbf{A}s} G_c \, ds$$

and

$$G_c = \mathbf{A}(\mathbf{F}_{d1} - \mathbf{I})^{-1} G_{d1},$$

which – using $\mathbf{A} = \frac{1}{h_2} \log \mathbf{F}_{d2}$ – leads to

$$G_{d2} = (\mathbf{F}_{d2} - \mathbf{I})(\mathbf{F}_{d1} - \mathbf{I})^{-1} G_{d1}. \tag{4.10}$$

As shown in the Appendix, the approximation of the "correct" noise covariance of the re-sampled system can than be achieved by substituting

$$G_{d2} \quad \longleftarrow \quad \sqrt{\frac{h_1}{h_2}}\, G_{d2}\,. \tag{4.11}$$

Using deterministic sampling does not involve intensive computations and the error made is rather modest, see (Söderström, 1991). Furthermore, this method does not require the underlying continuous-time counterpart to exist.

Another remark about numerical problems has to be made here. The time-series models used in the performance index framework are usually relatively high-order. This may in some cases lead to numerical problems when using the exact method which involves solution of Lyapunov and Riccati equations. The simpler calculations involved in the approximate method appear to result in a numerically more stable computation.

4.3 Numerical examples

Some simple examples for the re-sampling of an ARMA model are presented in this section, illustrating the methods described in the previous section. Assume for all examples that the original sampling rate is $h_1 = 2$ and one wants to re-sample the system with the new sampling rate $h_2 = 1$.

Re-sampling of an ARMA model

Consider first an ARMA model which is known to have a continuous-time counterpart. Consider the ARMA model

$$\begin{aligned} y(kh_1) &= \frac{1 - 0.5q^{-1}}{1 - 1.5q^{-1} + 0.7q^{-2}} e(kh_1) \\ \sigma_e^2 &= E\{e^2(kh_1)\} = 3\,. \end{aligned}$$

In state-space form one gets, defining $w_1(kh_1) = e(kh_1 + h_1)$,

$$\begin{aligned} x(kh_1+h_1) &= \begin{bmatrix} 1.5 & -0.7 \\ 1 & 0 \end{bmatrix} x(kh_1) + \begin{bmatrix} 1 \\ 0 \end{bmatrix} w_1(kh_1) \\ y(kh_1) &= \begin{bmatrix} 1 & -0.5 \end{bmatrix} x(kh_1) \\ \mathbf{R}_{d1} &= E\left\{ G_{d1} w_1^2(kh_1) G_{d1}^T \right\} = \begin{bmatrix} 3 & 0 \\ 0 & 0 \end{bmatrix}. \end{aligned}$$

The new dynamics matrix (4.5) and the state-covariance matrix (4.4) become

$$\begin{aligned} \mathbf{F}_{d2} &= \begin{bmatrix} 1.3117 & -0.3930 \\ 0.5614 & 0.4697 \end{bmatrix} \\ \mathbf{P} &= \begin{bmatrix} 26.5625 & 23.4375 \\ 23.4375 & 26.5626 \end{bmatrix}. \end{aligned}$$

The new noise covariance is calculated to be

$$\mathbf{R}_{d2} = \begin{bmatrix} 0.9191 & -0.4885 \\ -0.4885 & -0.0263 \end{bmatrix}.$$

Note that this matrix is indefinite. The solution of the Riccati Equation (4.7) and the Kalman gain K are obtained as

$$\mathbf{X} = \begin{bmatrix} 0.8927 & -0.5638 \\ -0.5638 & -0.2413 \end{bmatrix}, \quad K = \begin{bmatrix} 1.2283 \\ 0.3232 \end{bmatrix},$$

yielding the innovations form (4.6)

$$\begin{aligned} x(kh_2+h_2) &= \begin{bmatrix} 1.3117 & -0.3930 \\ 0.5614 & 0.4697 \end{bmatrix} x(kh_2) \\ &\quad + \begin{bmatrix} 1.2283 \\ 0.3232 \end{bmatrix} \nu(kh_2) \\ y(kh_2) &= \begin{bmatrix} 1 & -0.5 \end{bmatrix} x(kh_2) + \nu(kh_2) \\ \sigma_\nu^2 &= E\{\nu^2(kh_2)\} = 1.40. \end{aligned}$$

Hence, the exact re-sampled ARMA model is given by

$$\begin{aligned} y(kh_2) &= \frac{1 - 0.7147 q^{-1}}{1 - 1.7810 q^{-1} + 0.8367 q^{-2}} \nu(kh_2) \\ \sigma_\nu^2 &= E\{\nu^2(kh_2)\} = 1.40. \end{aligned}$$

This can be re-written using a new noise $e_1(kh_2)$ with the same variance as the original noise $e(kh_1)$, i.e. in this case $\sigma^2_{e_1} = 3$. The system with the new noise is obtained by multiplying the numerator of the ARMA process with σ_v/σ_e. This yields

$$\begin{aligned} y(kh_2) &= \frac{0.6822 - 0.4876q^{-1}}{1 - 1.7810q^{-1} + 0.8367q^{-2}} e_1(kh_2) \\ \sigma^2_{e_1} &= E\{e_1^2(kh_2)\} = 3. \end{aligned}$$

Using deterministic re-sampling, together with the proposed compensation (4.11) for the error made, the following system is obtained:

$$\begin{aligned} x(kh_2 + h_2) &= \begin{bmatrix} 1.3113 & -0.3930 \\ 0.5614 & 0.4697 \end{bmatrix} x(kh_2) \\ &\quad + \begin{bmatrix} 0.5745 \\ -0.2194 \end{bmatrix} e_2(kh_2) \\ y(kh_2) &= \begin{bmatrix} 1 & -0.5 \end{bmatrix} x(kh_2) \\ \sigma^2_{e_2} &= E\{e_2^2(kh_2)\} = 3. \end{aligned}$$

Then the approximated re-sampled ARMA model becomes

$$\begin{aligned} y(kh_2) &= \frac{0.6842 - 0.4887q^{-1}}{1 - 1.7810q^{-1} + 0.8367q^{-2}} e_2(kh_2) \\ \sigma^2_{e_2} &= E\{e_2^2(kh_2)\} = 3. \end{aligned}$$

The impulse responses for the three different models and the auto-covariance functions (ACF) are shown in Figure 4.1.

Re-sampling of an AR-model

The next example shows that the re-sampling of an AR model does not generally yield an AR model but an ARMA model. Consider the AR model

$$\begin{aligned} y(kh_1) &= \frac{1}{1 - 1.5q^{-1} + 0.7q^{-2}} e(kh_1) \\ \sigma^2_e &= E\{e(kh_1)^2\} = 3. \end{aligned}$$

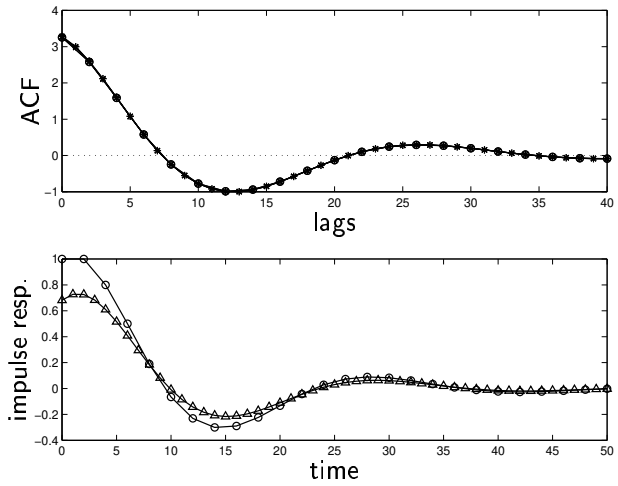


Figure 4.1: Re-sampling of an ARMA model. Top: Autocorrelation function original, exact and approximated re-sampled; bottom: impulse responses, original (o), exact (△). The impulse response for the approximative method equals almost perfectly the exact one and is hence omitted.

Using the same steps as in the previous example yields the (exact) ARMA model

$$y(kh_2) = \frac{1 - 0.2751q^{-1}}{1 - 1.7810q^{-1} + 0.8367q^{-2}} v(kh_2)$$
$$\sigma_v^2 = E\{v(kh_2)^2\} = 0.87.$$

As in the previous example, this can be re-written using a new noise $e_1(kh_2)$ with the same variance as the original noise $e(kh_1)$, yielding

$$y(kh_2) = \frac{0.5390 - 0.1483q^{-1}}{1 - 1.7810q^{-1} + 0.8367q^{-2}} e_1(kh_2)$$
$$\sigma_{e_1}^2 = E\{e_1(kh_2)^2\} = 3.$$

The approximate method using deterministic sampling yields the model

$$y(kh_2) = \frac{0.5745 - 0.1836q^{-1}}{1 - 1.7810q^{-1} + 0.8367q^{-2}} e_2(kh_2)$$
$$\sigma_{e_2}^2 = E\{e_2(kh_2)^2\} = 3.$$

Another situation arises when the discrete-time ARMA model does not have a continuous-time counterpart. In that case, the exact method in general does not work and one has two choices. Either to use an approximate method or modify the original time-series model and thereby hoping to receive a model which has a continuous-time counterpart.

4.4 Index for different sampling intervals

The evaluation of a performance index requires the calculation of the actual output variance σ_y^2. As briefly discussed in Section 2.1, the direct estimation of σ_y^2 will not be used here. Instead, since a state-space representation of the time-series model is available, it will be used to calculate σ_y^2 by solving a Lyapunov equation and using the available state-space model $(\mathbf{F}, G, H)$:

$$\begin{aligned} \mathbf{P} &= \mathbf{F}\mathbf{P}\mathbf{F}^T + GG^T \\ \sigma_y^2 &= Ey(t)^2 = \left(H\mathbf{P}H^T\right)\sigma_e^2 \end{aligned} \tag{4.12}$$

Notice that the variance expression σ_y^2 in the index for the re-sampled system has to be the same as for the original system since it was required that the two ACFs are equal in equal sampling points. Then of course it holds for lag zero, i.e.

$$\sigma_y^2 = r_{y1}(0) = r_{y2}(0).$$

Hence, in order to calculate a performance index for other than the original sampling rate, one only has to calculate the minimum-variance expression σ_{MV}^2 using the re-sampled time-series model.

A remark on the time-delay

When evaluating the re-sampled performance indices, one has to express the system dead-time in terms of the new sampling intervals. Assume that the physical dead-time is T_d time units, and that the sample-and-hold and controllers introduce a fixed number of delays d_s. Then the resulting total time-delay is

$$d = d_s + \mathrm{floor}(T_d/h),$$

where $\mathrm{floor}(x)$ rounds x to the nearest integer towards zero.

Obviously, if the system is re-sampled with a faster sampling rate, the number of samples of time-delay increases if there really is a physical dead-time in the process. However, if there is no physical dead-time in the process, the number of samples of time-delay (e.g. as may be introduced by the controller) remains constant at $\mathrm{d} = \mathrm{d_s}$, regardless what the sampling rate is.

Implementation

The scheme for sampling rate assessment can be implemented as shown in the following algorithm.

1. Calculate the minimum-variance based (Harris) index for the original system (Section 2.1).
2. Calculate a modified minimum-variance based index if the process has short or no dead-time (Section 3).
3. Calculate the re-sampled process model for the new sampling rate (Section 4.1 or 4.2).
4. Evaluate the Harris and modified index for the re-sampled system.
5. Repeat steps 3 and 4 for other sampling rates.

Note that Step 2 is not necessary. However, as shown in Chapter 3, for the cases mentioned, it yields a more reliable performance assessment.

4.5 Evaluation on industrial data

Frontend-temperature of a lime kiln

The data analysed in this section were collected at a Swedish pulp mill. It describes the frontend temperature in a lime kiln. The current controller, which is a model-predictive controller, uses a sampling interval of 2 minutes.

The process has a dominant time-constant of about 30 minutes and no physical dead-time. Since the calculations of the model-predictive controller are somewhat time-consuming, an extra sample delay is introduced by the controller, resulting in a total delay of two samples.

The speed of the MPC program has been considerably improved compared to when it was first installed in 1996. Today the control could easily be run at a sampling rate five times the current one, and the total time-delay would still only be two samples. This sampling-independent behaviour of the time-delay for systems with no or very short dead-time is important when applying the described re-sampling. In this case, the proposed modification in Section 3 will become significant.

The measured control error of the frontend-temperature of the lime kiln is shown in Figure 4.2. The time-series model of the control error was re-

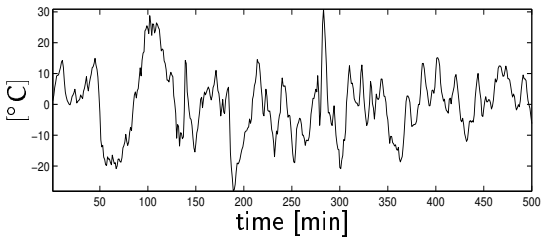


Figure 4.2: Control error of the frontend-temperature control in a lime kiln.

sampled for different sampling intervals $h_2 = [8,\ 4,\ 2,\ 1,\ 2/3,\ 1/2,\ 2/5]$. Since the current sampling is 2 min, this gives the re-sampling factors $\kappa = h_1/h_2$ of $\kappa = [1/4,\ 1/2,\ 1,\ 2,\ 3,\ 4,\ 5]$.

Figure 4.3 shows the original impulse response of the estimated ARMA model and the interpolated one at twice the sampling rate. Different performance indices were computed for all the re-sampled models.

In addition to the original and modified minimum-variance based index, the minimum-variance based index was also calculated assuming a fictitious physical dead-time of 1 minute. The modified index was calculated using λ computed according to (3.21) with a dominant time-constant $\tau = 30$ minutes and a speed-up factor $k_c = 2$. The different performance indices are plotted versus the inverse sampling rate in Figure 4.4. Not surprisingly, all indices increase if the sampling rate is increased. Since the number of delays is fixed

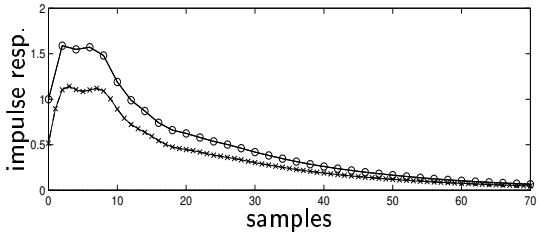


Figure 4.3: Impulse responses for original sampling rate (o) and interpolated model at twice the sampling rate (x).

at two, the original minimum-variance based index yields too large values to really be useful for assessing whether faster sampling would be beneficial. Both the modified index and the index with artificial dead-time tend to level out somewhat when sampling is made faster.

For example, the modified index for 5 times faster sampling has the value 3.6 compared to 2.6 at $h = 1$ minute. The likely conclusion is that increased sampling would not lead to any significant improvements in performance for this process.

Notice that indices are also computed for two cases of slower sampling. Again based on the modified index, it can probably be concluded that even the current controller sampling interval of 2 minutes is sufficient for this relatively slow process. The modified performance index does not change considerably for sampling rates of 8, 4 or 2 minutes.

Motor load control of a TMP refiner

Here data collected from a thermo-mechanical pulp refiner at a Swedish paper mill will be analysed. The data set contains data from a trial with automatic motor load control. Contrary to many other TMP refiners, the motor load is controlled by changing the dilution water flow rate rather than the chip feed rate. Since that dilution water flow rate has an almost immediate effect on the motor load, this is an appealing choice as a manipulated variable.

The dilution flow is locally controlled and the DCS is constructed such that both controllers in the cascaded loop introduce one sample of delay each. Hence, the overall time-delay is determined to be three samples, independent

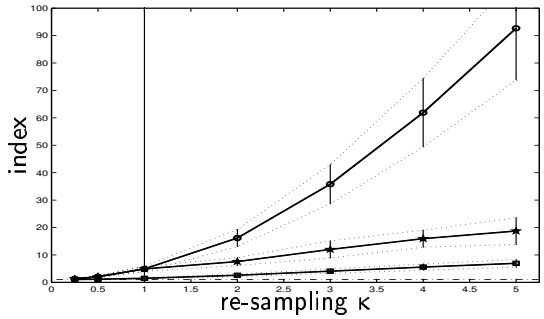


Figure 4.4: Performance indices for the lime kiln evaluated for different re-sampling rates κ. Original minimum-variance based index (•), modified minimum-variance based index (■) and original minimum-variance based index assuming a physical dead-time of one minute (★).The standard deviation is given by the dotted lines. The vertical solid line denotes the current situation, i.e. $h_1 = h_2$.

of the sampling rate. The current sampling interval is 1 second and the data are shown in Figure 4.5. Also for this case, the time-series model of the

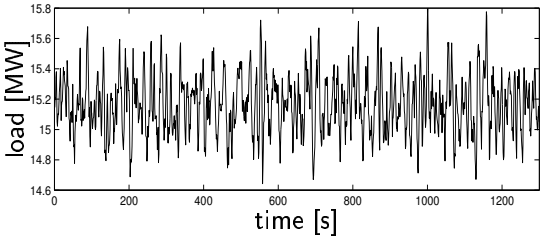


Figure 4.5: Motor load in a TMP refiner.

control error was re-sampled for different sampling intervals. Figure 4.6 shows the original impulse response and the interpolated one at twice the sampling rate. The same three indices were computed as for the lime kiln case, but the third one here introducing an artificial physical dead-time of one second. The modified index was computed using a dominant time-constant of 5 seconds, and again a maximum speed-up factor of 2. The performance

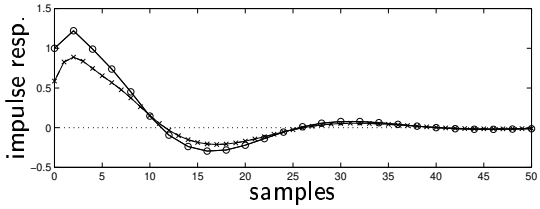


Figure 4.6: Impulse responses for original sampling rate (o) and interpolated model at twice the sampling rate (x).

indices are shown in Figure 4.7. The modified index here shows a more

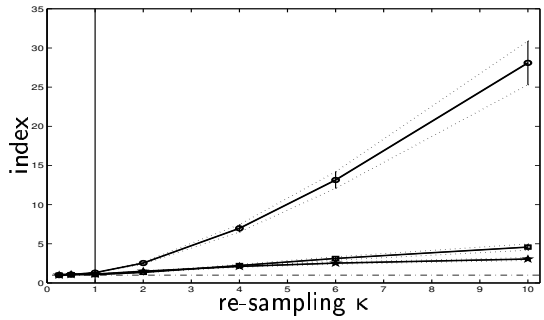


Figure 4.7: Performance indices for the pulp refiner evaluated for different re-sampling rates κ. Original minimum-variance based index (•), modified minimum-variance based index (■) and original minimum-variance based index assuming a physical dead-time of one second (★). The standard deviation is given by the dotted lines. The vertical solid line denotes the current situation, i.e. $h_1 = h_2$.

significant increase than for the lime-kiln case. At ten times the sampling rate the modified index becomes 4.6, which should be compared to 1.2 at the current sampling. Based on this it could probably be concluded that if good control of the motor load is important, then increased sampling speed should be considered.

A remark on spectral properties

Whenever dealing with sampling, one has to be aware of aliasing. In order to get reliable results, one has to assume that an anti-aliasing filter was used when collecting the measurement data. Clearly, then one can only assess the information which is left in the data, i.e. the low-frequency information. For the case of interpolation there will be no problem if the original data set contains all frequencies where the main information is located. That this requirement is easily satisfied for the data sets considered here is shown in Figures 4.9 and 4.8. It can be seen that the main information is at

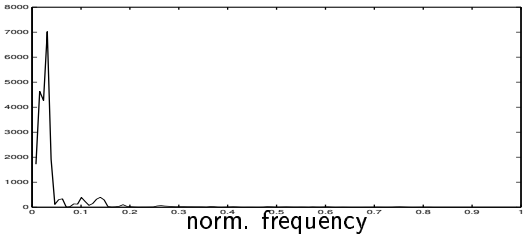


Figure 4.8: Power spectrum for the lime-kiln data set. The frequency axis is scaled such that the Nyquist frequency is unity.

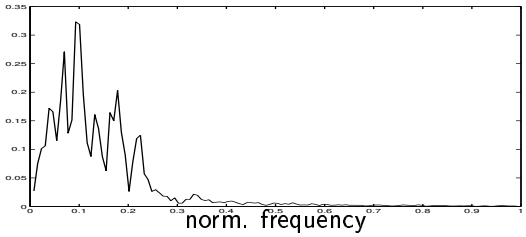


Figure 4.9: Power spectrum for the motor load data set. The frequency axis is scaled such that the Nyquist frequency is unity.

frequencies which are much lower than the Nyquist frequency.

When this data is then used for decimation, the Nyquist frequency will decrease, i.e. get closer to the important frequencies. It is clear that reliable

information can only be obtained as long as the Nyquist frequency is above the frequencies where the main information is located. As can be seen, this is also satisfied up to a decimation by a factor of three to four.

Summary

In process industry, too fast sampling in control loops seldomly occurs whereas too slow sampling may increase the output variability considerably. This will have a negative effect on product quality.

A method has been presented which allows interpolation and thereby the evaluation of performance indices at other sampling rates than the one used for data collection. Of course the index will increase for faster sampling, and it is then how significant this increase is which will determine if the faster sampling should be pursued or not.

The proposed exact method yields correct results as long as the discrete-time system has a continuous-time counterpart. However, the calculation of the latter is not necessary. The method cannot handle systems with poles on the unit circle. Furthermore, an approximate method has been presented which yields very good results as long as the same assumptions as for the exact method are made. The approximate method is computationally much simpler and seems to be numerically more stable.

The proposed approach was illustrated using two data sets from pulp and paper industry. Common to the two selected processes is that they have no apparent physical dead-time. For that case the original minimum-variance based index produces too large values to be used for the assessment. Therefore the modified index as presented in Chapter 2 was applied, which takes into account that in practice it is typically not advisable to speed up the closed-loop system more than 2-4 times compared to the open-loop one.

For processes with long physical dead-time the number of samples delay will be proportional to the increase in sampling rate. As a consequence, the benefit of increased sampling will typically be insignificant unless the current sampling is extremely slow.

Also notice, that even if fast sampling is usually not a problem, the proposed method can be used to check whether the system may actually be sampled slower.

Chapter 5

Performance index for oscillating loops

It has been pointed out that a large amount of control loops in the process industry are oscillating, see (Bialkowski, 1992) or (Ender, 1993). Therefore, when applying the Harris index to a large number of control loops as a first-level performance assessment, it will end up being evaluated for some oscillating signals. It is therefore interesting to investigate what kind of results the minimum-variance based index will give in these cases.

As a motivating example consider the process variable in a flow control loop in a pulp mill, see Figure 5.1. The signal is oscillating for the first 350

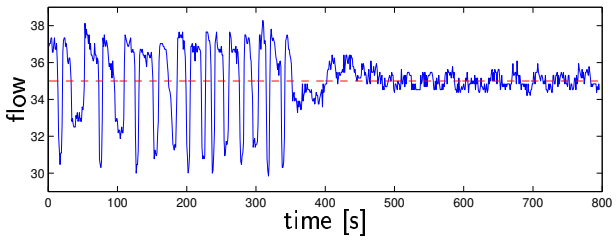


Figure 5.1: Measured data from an oscillating flow control loop.

samples, after that the oscillation vanishes. The reason for the oscillations is likely to be a nonlinear phenomenon and a change in the noise structure is

not expected within the short time period considered[1]. One would therefore expect that the performance index should be able to pick up the change in the performance.

From identification experiments is is known that the process dead-time is two samples. Table 5.1 shows the estimated indices and variances for both parts of the data. The time-series model used is an AR model of order 25.

data	σ_y^2	σ_{MV}^2	σ_e^2	I_p
$[1:350]$	5.87	1.92	0.75	3.06
$[500:800]$	0.14	0.087	0.081	1.71

Table 5.1: Performance index and estimated variances for the data shown in Figure 5.1.

Notice that the variance in the process output decreased by a factor of about 40. The performance index however, decreased by a factor less than two. It is also noteworthy that both the estimated minimum-variance terms and the noise variances differ considerably for both cases. On the other side, the output variances were obtained using the estimated time-series models and are in good agreement with the variances obtained directly from the data. The question then arises:

> *Why is the obvious change of performance not properly reflected in the index?*

In the rest of this chapter we will illuminate this question mainly by using some simulation examples. The finding will be that the performance index may be misleading in some cases when the signals used are oscillating. Furthermore, it will be argued that if one wants to calculate the Harris index for oscillating signals, AR- rather than ARMA-modelling should be used.

5.1 Simulation example

In this section four simulation examples will be shown, each of them dealing with an oscillating control loop. The impact of the oscillation cause on the resulting performance index will then be inpspected.

[1] Irregular changes between oscillating and non-oscillating intervals could be found for this loop over a period of several months.

Consider the open-loop system

$$y(t) = \frac{B(q^{-1})}{F(q^{-1})}u(t) + \frac{C(q^{-1})}{D(q^{-1})}e(t)$$

with $F(q^{-1}) = D(q^{-1}) = 1 - 1.7\,q^{-1} + 0.7\,q^{-2}$, $B(q^{-1}) = q^{-5}(1 + 0.5\,q^{-1})$ and $C(q^{-1}) = 1 - 0.9\,q^{-1}$. The noise $e(t)$ is zero-mean Gaussian with variance $\sigma_e^2 = 0.1$. Let the controller be of proportional type (gain K_c) and consider the following four different scenarios which all lead to oscillations:

1. linear system on the stability boundary ($K_c = 0.045$),
2. stable system with backlash[2] ($K_c = 0.035$, dead-zone$= 0.1$),
3. unstable system with saturation ($K_c = 1$, limit ± 0.5),
4. stable system with oscillating output disturbance ($K_c = 0.015$).

The disturbance in Case IV was realised by changing the noise model denominator to

$$D(q^{-1}) = 1 - 2\cos[2\pi \cdot 0.05]\,q^{-1} + q^{-2}.$$

This means that the noise model has two poles on the unit circle. As in the other three cases, white noise is fed into the noise model.

The minimum achievable variance σ_{MV}^2 can be computed easily from the series expansion of the noise transfer function $C(q^{-1})/D(q^{-1})$. For Case I-III this yields

$$\frac{C(q^{-1})}{A(q^{-1})} = 1 + 0.8\,q^{-1} + 0.66\,q^{-2} + 0.562\,q^{-3} + 0.4934\,q^{-4} + \ldots .$$

and therefore

$$\sigma_{MV}^2 \approx \left[1 + 0.8^2 + 0.66^2 + 0.562^2 + 0.4934^2\right] 0.1 = 0.263.$$

For Case IV the minimum variance term becomes

$$\sigma_{MV}^2 \approx \left[1 + 1.002^2 + 0.906^2 + 0.722^2 + 0.466^2\right] 0.1 = 0.356.$$

Simulation results for all cases can be seen in Figure 5.2. Then, when com-

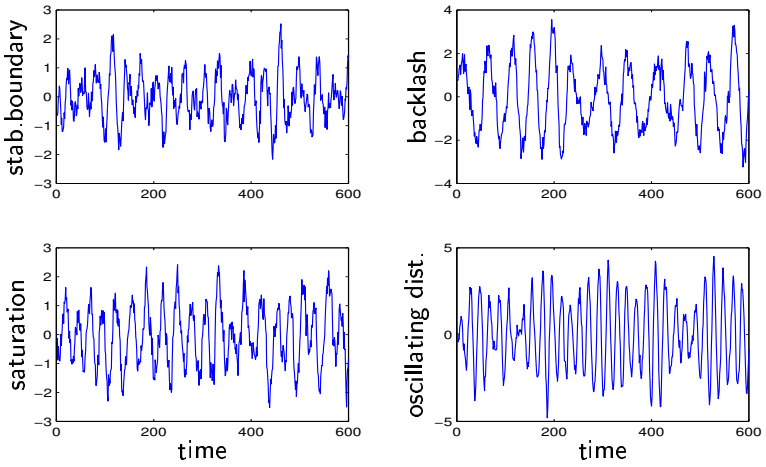


Figure 5.2: Simulation results for the four different cases.

puting the Harris index (using an AR(25) model), the results shown in Table 5.2 have been found. Several interesting remarks can be made. The actual output variances $\sigma^2_{y,m}$ – as obtained via the time-series model – are approximately equal to the ones obtained directly from data ($\sigma^2_{y,d}$) in all cases. Also the residual variances $\hat{\sigma}^2_e$, which are an estimate of the noise variance are close to the correct value ($\sigma^2_e = 0.1$) in all cases. The minimum-variance term $\hat{\sigma}^2_{MV}$ is only correctly estimated for Case I. For all other cases it was increased thus yielding a performance index which is lower than the theoretically correct one.

For further illustration consider also the first parts of the series expansion (i.e. the first Markov parameters) which are shown for all cases in Figure 5.3. Note that Case IV used a different noise model and is shown separately. The following remarks can be made from this example

- The actual output variance σ^2_y can usually be correctly estimated using the time-series model.
- The noise variance σ^2_e was correctly estimated in all cases. From experience with real data however, it is recommended to avoid the use

[2]The nonlinearities will be defined more detailed in Chapter 9.

Case	$\sigma^2_{y,m}$	$\sigma^2_{y,d}$	$\hat{\sigma}^2_{MV}$	σ^2_{MV}	$\hat{\sigma}^2_e$	σ^2_e	$\hat{I}_p$	I^0_p
I	0.64	0.61	0.24	0.263	0.095	0.1	2.65	2.43
II	2.28	2.20	0.41	0.263	0.115	0.1	5.56	8.67
III	1.01	1.06	0.39	0.263	0.112	0.1	2.59	3.84
IV	4.03	3.93	0.43	0.356	0.099	0.1	9.37	11.04

Table 5.2: Performance index ($\hat{I}_p$) and estimated variances for the data shown in Figure 5.2. The variable I_p denotes the theoretical performance index, i.e. the actual variance σ^2_y divided by the theoretical minimum-variance term σ^2_{MV}. The actual variance obtained from model and data are denoted $\sigma^2_{y,m}$ and $\sigma^2_{y,d}$ respectively.

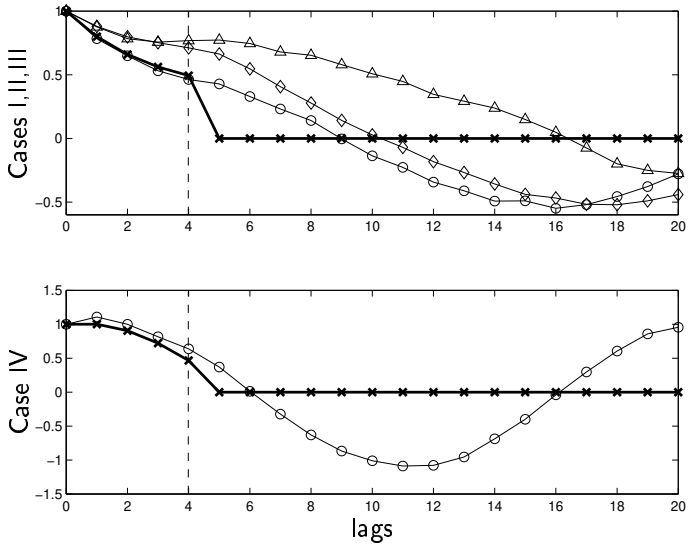


Figure 5.3: Series expansions based on the estimated AR(25)-model. Top: Theoretical expansion (×), Cases I(○), II(△) and III(◇). Bottom: Theoretical expansion (×), Case IV(○). The number of samples of dead-time is denoted by the dashed line.

of $\hat{\sigma}_e^2$ for the index calculation as already advocated in Chapter 2.

- The estimated minimum-variance term $\hat{\sigma}_{MV}^2$ may be wrong if nonlinearities are involved resulting in a wrong value of the performance index.

Recall again the large difference in the estimation of σ_{MV}^2 and σ_e^2 in Table 5.1. It is less surprising that the minimum-variance based performance index can run into problems when significant[3] nonlinearities occur since it relies on a *linear* concept. In the case of an external oscillating disturbance, the assumption (for minimum variance control) that the disturbance is stationary, is violated. However, as can be seen from the example, the performance index is able to indicate the bad performance in this special case.

The most puzzling case is Case I where the controller is tuned such that the loop is on the stability boundary. The oscillation amplitude is not very large in this case, thus yielding a modest value of the index. The estimated time-series model is dominated by the oscillating behaviour (i.e. it has two dominating complex poles near the unit circle). Then, even though the true noise model does not have such a behaviour the first Markov parameters are correctly estimated.

Notice again that experiences from real data show that the index may vary far more than the simulations shown here indicate. Also the results of the simulations exhibit considerable variations for different noise realisation.

The conclusion to be made is the following. Since it is not known a priori whether an oscillation is a linear or nonlinear phenomenon, it would be best to screen out oscillating loops before evaluating the performance index. However, if it cannot be avoided to compute an index for oscillating signals, one should choose a AR model rather than ARMA as will be motivated in the next section.

5.2 Modelling oscillating signals

As already mentioned, in order to describe an oscillating signal the time-series model needs at least two complex poles on the unit circle. Consider first the case when the chosen time-series model is ARMA.

[3]Significant here means that the nonlinearity actually dominates the loops behaviour, i.e. it causes the oscillation.

ARMA model

For simplicity, assume that the oscillating signal in question can be described as a sum of sinusoids with additive noise

$$\begin{aligned} y(t) &= x(t) + e(t), \\ x(t) &= \sum_{k=1}^{n} \alpha_k \sin(\omega_k t + \phi_k), \end{aligned} \tag{5.1}$$

where α_k, ω_k, ϕ_k and $e(t)$ are the amplitudes, the frequencies, initial phases for each component of the noise-free sinusoidal signal and the additive white observation noise respectively. Note then that

$$\left(1 - e^{-i\omega_k} q^{-1}\right)\left(1 - e^{i\omega_k} q^{-1}\right) x_k(t) = 0, \tag{5.2}$$

where $x_k(t)$ is the k-th component in $x(t)$. If (5.2) is used for each component in (5.1), one obtains

$$\begin{aligned} A(q)x(t) &= 0, \\ A(q) &= \prod_{k=1}^{n} \left(1 - 2\cos(\omega_k) q^{-1} + q^{-2}\right). \end{aligned}$$

The full ARMA-model for the noisy data $y(t) = x(t) + e(t)$ then becomes

$$A(q)y(t) = A(q)e(t). \tag{5.3}$$

A model such as (5.3) is called *degenerate* ARMA and all its poles and zeros are exactly located *on* the unit circle. Furthermore, the AR and the MA parts of the model are identical[4].

The consequence for the evaluation of the performance index is that one has to estimate the model parameters such that poles and zeros are located on the unit circle. Additionally, poles and zeros have to cancel each other. Since it was recommended in Chapter 2 to use the estimated model for calculation of σ_y^2, an important requirement is stability. The estimation of an ARMA model may hence be very difficult, especially since stability can in general not be guaranteed.

[4]Note that the polynomial $A(q)$ cannot be cancelled since $y(t) = e(t)$ is not true!

AR model

In Chapter 2 it was already mentioned that AR models are often sufficient when evaluating the performance index. Furthermore, using AR models stability can be guaranteed.

In order to describe an oscillation properly one usually needs high-order AR models. For such cases, the poles are spread out around the unit circle close to the stability boundary, see Figure 5.4 for a plot of the poles of an AR(25) model for an oscillating signal. The data used was taken from the same (oscillating) loop as the data in Figure 5.1. This interesting phe-

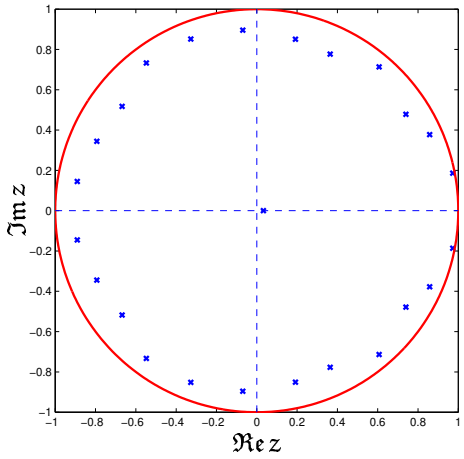


Figure 5.4: Poles of the AR(25) time-series model for an oscillating signal.

nomenon has been discussed and explained by Marí *et al.* (2000). They showed that the poles approach a circle around the origin with radius 1. Figure 5.5 shows the series expansion of the AR(25) model. It can be seen that the response is decaying since the AR model is stable. As a matter of fact it has to decay in order to yield the correct value of the output variance which otherwise would be infinite since

$$\sigma_y^2 = \left(\sum_{i=0}^{\infty} h_i^2 \right) \sigma_e^2. \tag{5.4}$$

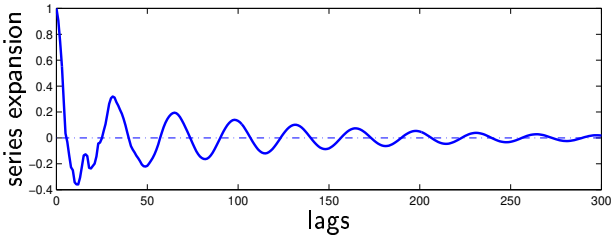


Figure 5.5: Series expansion of the estimated AR-model.

Laguerre-model

Note that the Laguerre filters have a real pole only. In order to fit oscillating data it would be better to consider the use of Kautz filters instead which contain a complex pole pair, see for example (Wahlberg, 1994). However, the use of Kautz filters would require knowledge whether or not the loop is oscillating, and of the oscillation frequency.

Conclusions

The question asked in the introduction of this chapter contained can now be answered:

> *Oscillations are a potential problem when evaluating the Harris index since the index may become too low.*

The best way is of course to detect oscillations prior to computing the performance index. If this is not possible, one should use AR modelling rather than ARMA modelling.

Part II

Event-triggered estimation

Chapter 6

Closed-loop identification

The evaluation of the control performance index introduced in Chapter 2 requires knowledge of the process dead-time. For fast and simple commissioning of a monitoring tool, it would be desirable to enable automatic dead-time estimation. Furthermore, as will be seen later on, simple process models may be of great help for oscillation diagnosis.

A natural question is therefore if one can obtain information about the process in question, e.g. the dead-time in a non-invasive manner. This chapter deals with the question of how to obtain suitable identification data from a control loop (in closed-loop) *without* performing experiments. Such a procedure will be called *event-triggered identification*. The theory about closed-loop identification is now well-explored, see for example (Forssell and Ljung, 1999).

In real control systems, there is always excitation present. This excitation can caused by setpoint changes or external disturbances. Limit cycles do also excite a control loop. In this chapter slight re-formulation of a result in (Ljung, 1999) is given. It establishes conditions for closed-loop data to be informative, both for single (Section 6.1) and cascaded (Section 6.2) control loops.

6.1 Informative closed-loop data

Let the system be described as

$$y(t) = G(q^{-1})\,u(t) + H(q^{-1})\,e(t),$$

where $G(q^{-1})$, $H(q^{-1})$ are process and noise transfer function respectively. Start with the following definitions (Ljung, 1999):

Definition 6.1. *Introduce the operation* $\bar{E}$ *defined by*

$$\bar{E}\, f(t) = \lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} E\, f(t)$$

where it is assumed that the limit exists whenever the symbol is used. □

Definition 6.2. *A quasi-stationary data set* Z^∞ *is informative enough with respect to the model set* $\mathcal{M}^*$ *if, for any two models* $W_1(q)$ *and* $W_2(q)$ *in the set,*

$$\bar{E}[(W_1(q) - W_2(q))\, z(t)]^2 = 0 \tag{6.1}$$

implies that $W_1(e^{i\omega}) = W_2(e^{i\omega})$ *for almost all* ω. □

Now the following theorem can be stated which is a more general formulation of a result, (Theorem 13.2), in (Ljung, 1999).

Theorem 6.1. *Consider the block-diagram in Figure 6.1, where* $F(q)$ *and* $G(q)$ *are linear time-invariant dynamical systems. Let* $u(t)$ *and* $y(t)$ *be the control signal as calculated by the controller and the measured process output respectively. Assume that* $r(t)$ *and* $v(t)$ *are uncorrelated.*

Then the closed-loop data set $Z = \{y(t), u(t)\}_{t=1,\dots,N}$ *is informative if and only if a persistently exciting signal (of sufficient order) enters the loop outside the shaded area in Figure 6.1, i.e. between* y—u. □

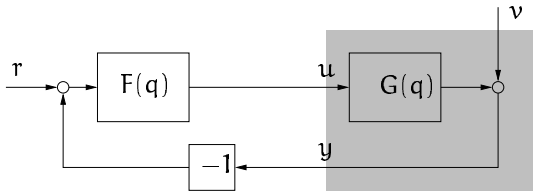


Figure 6.1: A general control loop.

Proof: Let $W(q) = [W_y(q)\ W_u(q)] = [H^{-1}(q)G(q)\ (1 - H^{-1}(q))]$ and $z(t) = [u(t)\ y(t)]^T$. Consider then Definition 6.1. Using

$$W_1(q) - W_2(q) = \begin{bmatrix} \Delta W_y(q) & \Delta W_u(q) \end{bmatrix},$$

(6.1) can be re-written as

$$\bar{E}[\Delta W_y(q)y(t) + \Delta W_u(q)u(t)]^2 = 0 \tag{6.2}$$

Here, $\Delta W_u(q)$ and $\Delta W_y(q)$ are filters that are not both zero and that are of about the same complexity as the models in the model set. For the regulator $u(t) = F[r(t) - y(t)]$ this would mean that

$$0 = \bar{E}\left|\begin{bmatrix}\Delta W_y & \Delta W_u\end{bmatrix}\begin{bmatrix}y\\u\end{bmatrix}\right|^2 \tag{6.3}$$

$$= \bar{E}\left|\begin{bmatrix}\Delta W_y & \Delta W_u\end{bmatrix}\begin{bmatrix}G_0 & S_0\\1 & -FS_0\end{bmatrix}\begin{bmatrix}FS_0 r\\v\end{bmatrix}\right|^2, \tag{6.4}$$

where $y(t) = G_0S_0r(t) + S_0v(t)$ and $u(t) = S_0r(t) - FS_0v(t)$ were used. Let

$$\tilde{W} = \begin{bmatrix}\tilde{W}_y & \tilde{W}_u\end{bmatrix} = \begin{bmatrix}\Delta W_y & \Delta W_u\end{bmatrix}\begin{bmatrix}G_0 & S_0\\1 & -FS_0\end{bmatrix}. \tag{6.5}$$

The determinant of the last matrix is $-G_0FS_0 - S_0 = -1$, so it is always invertible, which means that $\tilde{W} = 0$ will imply that both ΔW_y and ΔW_u are zero. Recall that r and v are uncorrelated by assumption, so

$$0 = \bar{E}\left|\tilde{W}\begin{bmatrix}FS_0 r\\v\end{bmatrix}\right|^2 = \bar{E}\left|\tilde{W}_y FS_0 r\right|^2 + \bar{E}\left|\tilde{W}_u v\right|^2. \tag{6.6}$$

The filtered noise v is persistently exciting, and if S_0r is that too, then the conclusion is that $\tilde{W} = 0$. Under that assumption it is thus shown that Equation (6.2) implies $\Delta W_y = 0$ and $\Delta W_u = 0$. Note also that FS_0r will be persistently exciting if r is, since the analytical function FS_0 can be zero at at most finitely many points.

Up to this point the proof followed closely the derivation in (Ljung, 1999) where a different block diagram for the closed-loop system was used.

Signal entering outside the shaded area. Consider now the case when an exciting signal enters outside the shaded area in Figure 6.1. Wherever the signal enters, it can always be expressed as a filtered setpoint signal. Therefore, the first part of the proof can be used in exactly the same way when $r(t)$ is replaced by $H(q^{-1})r(t)$ instead. For instance when the exciting signal is inserted after the controller, then the filter H is F^{-1}.

Signal entering inside the shaded area. In this case, all signals entering between u—y can be expressed as a filtered signal which enters the loop at the process output (where $v(t)$ already enters the loop). Let the new exciting signal be denoted by $d(t)$ and its filtered version as $H(q^{-1})d(t)$. Now, both exciting signals, $v(t)$ and $H(q^{-1})d(t)$ ($r = 0$ in this case) enter the loop at the same place. Then Equation (6.4) trivially becomes

$$0 = \bar{E} \left| \begin{bmatrix} \Delta W_y & \Delta W_u \end{bmatrix} \begin{bmatrix} S_0 & S_0 \\ -FS_0 & -FS_0 \end{bmatrix} \begin{bmatrix} Hd \\ v \end{bmatrix} \right|^2 .$$

The determinant of the matrix in the middle is zero. This means that the new signal will not yield an informative data set. □

Remark 6.1. *From the condition monitoring point of view the only realistic event in order to obtain informative closed-loop data is really to wait for setpoint changes.*

Remark 6.2. *Another reasoning can enlighten why some process disturbances do not give sufficiently informative data to identify the process. Consider the case of a persistently exciting incoming disturbance at the process output. Assume that the reference signal is constant and no extra signal is added to the control signal. Then the relation between control signal* $u(t)$ *and measured process output* $y(t)$ *is always determined by the controller, i.e.*

$$u(t) = -F(q)y(t).$$

This means that no matter how much u *and* y *vary, as long as the exciting signal enters the loop in the shaded box in Figure 6.1, only the controller can be identified. The same is true if the loop is oscillating. See also Section 6.3 for further discussion.*

6.2 Identification in cascaded loops

Consider now a cascaded control structure, see Figure 6.2. For the identification of the dynamics of the outer loop (G_o), Theorem 6.1 applies directly. For a better understanding consider the re-written block diagram as shown in Figure 6.3. There, S_i and T_i are the sensitivity and the complementary

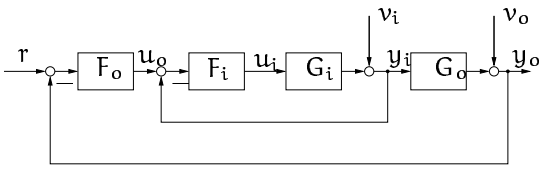


Figure 6.2: Example of a cascaded control loop.

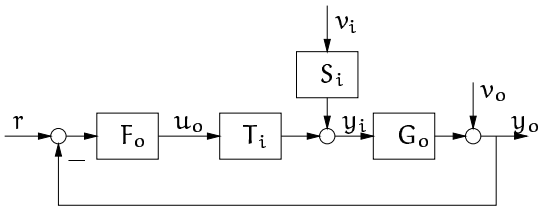


Figure 6.3: Example of a cascaded control loop.

sensitivity function of the inner loop respectively. It can now be seen that the disturbance v_i enters the loop *before* y_i is measured. That means – in terms of Theorem 6.1 – that it enters the loop outside the shaded area. A disturbance in v_i will hence yield informative data $[y_i\ y_o]$ for identification of G_o.

The identification of the dynamics of the inner loop (G_i) raises some interesting questions. In this case, any exciting signal within the cascaded control loop will result in an exciting setpoint of the inner loop (u_o). One could hence conjecture that it is possible to identify the inner process G_i without requiring that r is exciting. A potential problem is that u_0 will always be correlated with the exciting signal. The results from Theorem 6.1 will now be applied in order to confirm the conjecture.

First we need to express both u_i and y_i in terms of v_i and v_o:

$$\begin{aligned} u_i &= -\frac{F_i(1+F_oG_o)}{1+F_iG_i(1+F_oG_o)}v_i - \frac{F_iF_o}{1+F_iG_i(1+F_oG_o)}v_o, \\ y_i &= \frac{1}{1+F_iG_i(1+F_oG_o)}v_i - \frac{G_iF_iF_o}{1+F_iG_i(1+F_oG_o)}v_o. \end{aligned} \tag{6.7}$$

These expressions are then used to replace u and y respectively in (6.3)

yielding

$$0 = \bar{E}\left|\begin{bmatrix}\Delta W_y & \Delta W_u\end{bmatrix}\underbrace{\frac{1}{\Gamma}\begin{bmatrix}1 & -G_i F_i F_o\\ -F_i(1+F_o G_o) & -F_i F_o\end{bmatrix}}_{\Sigma}\begin{bmatrix}\nu_i\\ \nu_o\end{bmatrix}\right|^2, \qquad (6.8)$$

where $\Gamma = 1 + F_i G_i(1 + F_o G_o)$. It can then be seen that

$$\det \Sigma = \frac{-F_i F_o}{1 + F_i G_i\,(1 + F_o G_o)}.$$

It is hence concluded that – theoretically – for cascaded loops the setpoint r does not need to be exciting in order to be able to obtain informative identification data for both the inner (G_o) and the outer loop (G_o). Interestingly, if identification of the inner loop is considered, ν_i represents the noise whereas ν_o plays the rôle of the exciting signal. If one aims to identify the outer loop, the disturbances ν_i and ν_o change their meanings.

However, from a practical point of view there is a problem since the noise cannot – at the same time – be disturbance and excitation signal. Hence, realistically one can only identify either the inner or the outer loop, depending on how the ratio of ν_i and ν_o is. For identification of G_i a significant event in ν_o is needed whereas the noise level of ν_i has to be modest (compared to ν_o). Then, however, G_o cannot be identified, even if the data is – theoretically – informative. The same reasoning can be applied vice versa for the identification of G_o is there is an event in ν_i. An example for identification in a cascaded loop will be discussed in Chapter 8.

6.3 Controller identification

Knowledge of the controller parameters can be important for a supervision system. If the control structure and/or the controller parameters are unknown one has to identify the controller. This section deals with the question what the necessary conditions for the event-triggered identification of the controller are.

For that purpose, the same reasoning as in Section 6.1 will be used. Consider the block diagram of a normal control loop in Figure 6.4a. A

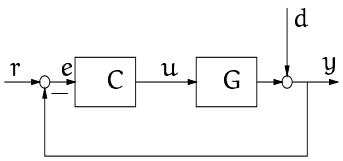


Figure 6.4a: Original

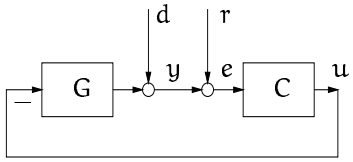


Figure 6.4b: Rearranged

Figure 6.4: Original SISO control loop and its re-arranged version.

simple rearrangement of the blocks yields the block diagram in Figure 6.4b.

Two things can be seen from the figure: Firstly, the setpoint signal $r(t)$ enters the loop in the same way as the disturbance $d(t)$. Secondly, applying the same theory as was used in Section 6.1, one can see that one obtains informative data when one of the inputs (d or r) is exciting. The transfer function that can be estimated is

$$u(t) = C(q^{-1})e(t) = C(q^{-1})\,[r(t) - y(t)].$$

It can hence be concluded that it is possible to estimate the controller transfer function whenever there is some excitation in the loop, no matter what the source is. Furthermore, contrary to the identification of the plant, controller identification is noise-free! Hence it should always be possible to obtain accurate models.

An example. Consider data from a level control loop shown in Figure 6.5. The setpoint is constant and the loop is excited by disturbances only. When fitting an ARX(1,2,0)-model to the data, the model obtained is

$$\hat{C}(q^{-1}) = \frac{3.16 - 3.14q^{-1}}{1 - 0.99q^{-1}}.$$

Comparing this to the discrete-time version of a PI-controller yields the controller parameters $\hat{K}_c = 3.16$ and $\hat{T}_i = 129$. The real values are $K_c = 3.44$ and $T_i = 200$. Even though the value of T_i is not correct, it is high compared to the controller gain. Both controllers are almost purely proportional. The validation of the estimated controller model can be seen in Figure 6.6.

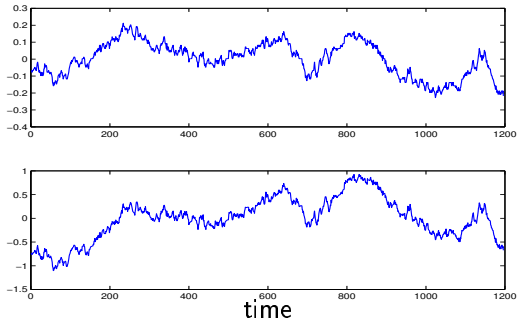


Figure 6.5: Identification data. Top: controller output, bottom: controller input (control error).

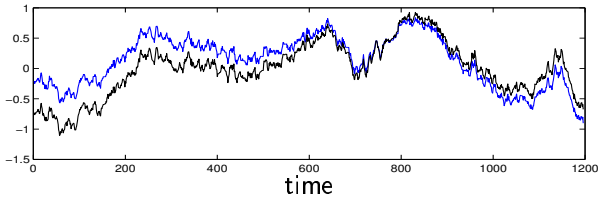


Figure 6.6: Validation of the estimated controller. Measured and simulated controller output.

Summary

Assuming that a monitoring tool is restricted to normal operating data, the only realistic event to hope for is setpoint changes. Disturbances that upset the system cannot be used since they typically enter in the shaded region of Figure 6.1. Luckily, many processes change operating points regularly in order to produce for example different qualities. A second reason for setpoint changes is if the loop is located in a cascade structure. Note that this is very likely in processes where the setpoints of low-level loops are controlled by high-level controllers like model-predictive controllers for example.

The identification of the controller may be necessary if the controller parameters are not available in a supervision system. Fortunately, the identification of the controller is always possible since the excitation originating from disturbances yields informative identification data.

Chapter 7

Segmentation of data for dead-time estimation

This chapter deals with the automatic selection of data for estimation of the process dead-time from a batch of operating data. As described in Chapter 2, the dead-time is a prerequisite for evaluating the minimum-variance based control performance index. For an automatic supervision tool, it is desirable that the dead-time is estimated automatically on a regular basis rather than specifying a value when commissioning the tool. The two main reasons for this are that the dead-time may be time-varying and that it is often unknown.

The kind of data which is suitable for event-triggered dead-time estimation has been discussed in Chapter 6, see also (Isaksson *et al.*, 2001). The main finding was that, in order to estimate the dead-time from normal operating data, the only excitation which one can expect is a change in the loop setpoint.

7.1 Description of the algorithm

The selection of informative identification data is done by selecting the transient parts of the data which are caused by setpoint changes. The selection criterion is as follows:

> *Start selecting data after each significant setpoint change and continue until the data has reached steady-state again.*

In order to do that, two different tasks have to be solved. Firstly, one has to detect significant setpoint changes, i.e. changes which are significantly larger than the noise level of the process output. This guarantees that only setpoint changes which are noticeable in the process output are chosen. Once an estimate of the noise variance of the process output is available, the detection of setpoint changes is a simple task.

The second problem to solve is the assessment whether or not the data has reached steady-state. For this, a method proposed by Cao and Rhinehart (1995) is used. It detects intervals of steady-state in data and estimates the noise variance as a by-product.

The main idea by Cao and Rhinehart (1995) is to consider the ratio of the noise variance, calculated in two different ways. First, one calculates the variance as the mean-square deviation from average for windowed data. The second way is to calculate the mean of squared differences of successive data. The ratio of these variance estimates will be close to one if the data is approximately steady-state. It will be larger than one otherwise. Let the control error be denoted as $e(t)$, then the method can be written down recursively:

$$\begin{aligned} e_f(t) &= \lambda_1 e(t) + [1-\lambda_1] e_f(t-1) \\ v_f^2(t) &= \lambda_2 [e(t) - e_f(t-1)]^2 + [1-\lambda_2] v_f^2(t-1) \\ \delta_f^2(i) &= \lambda_3 [e(t) - e(t-1)]^2 + [1-\lambda_3] \delta_f^2(t-1). \end{aligned}$$

The two mentioned estimates of the noise variance are

$$s_1^2(t) = \frac{2-\lambda_1}{2} v_f^2(t), \quad s_2^2(t) = \frac{\delta_f^2(t)}{2}.$$

The ratio of the two variance estimates gives the decision function:

$$R_i(t) = \frac{s_1^2(t)}{s_2^2(t)} = \frac{(2-\lambda_1) v_f^2(t)}{\delta_f^2(t)}. \tag{7.1}$$

This function is computed for each sample as the filtering is recursive and updated for each sample. The signal under investigation is decided to be in steady-state as long as $R_i < \bar{R}$. Cao and Rhinehart (1995) $\bar{R}$ suggested $\bar{R} = 2$ and the for filter constants $\lambda_1 = \lambda_2 = \lambda_3 = \lambda = 0.1$. The authors also show that this detection algorithm works well over a large range of cases. However, in this work we have chosen $\lambda = 0.05$ in order to guarantee that a

sufficient amount of data is selected. A smaller λ is chosen, the more data will be selected using the proposed algorithm. This is because a low value of λ will be more "critical" to accept data as stead-state.

Once stationary parts of the data are identified, the determination of the estimation data is straight forward as already described above. A significant setpoint change is detected when a setpoint change which is larger than for example three times the estimated noise standard deviation. From each detected setpoint change, data will be collected until the control error reaches steady-state again.

If a sequence of steps occurs, scattered parts of data have to be used for identification. How to do this properly is described in Chapter 14 of (Ljung, 1999). Combining the estimates for each interval may be difficult since the excitation in single intervals may be poor. A better way of estimating a model is to fit parameters which minimise the criterion

$$V(\theta) = \sum_{t \in T_1} [y(t) - \hat{y}(t|\theta)]^2 + \cdots + \sum_{t \in T_n} [y(t) - \hat{y}(t|\theta)]^2 , \qquad (7.2)$$

where T_i is the index set of the i-th segment and $y(t)$ and $\hat{y}(t|\theta)$ are the measured and the predicted process output respectively . One way of doing this is to build up the regressor matrix for the least-squares problem and then remove all rows which correspond to discarded data. This – theoretically sound – way may be difficult if the data batch in question is very large. In that case one may rather estimate models for each segment and merge these afterwards. This is properly done by weighting them according to their inverse covariance matrices, see (Ljung, 1999). Let the parameter estimate for segment i be denoted $\hat{\theta}_i$, and let the estimated covariance matrix be P_i. The resulting estimate is given by

$$\hat{\theta} = P \sum_{i=1}^{n} P_i^{-1} \hat{\theta}_i, \qquad P = \left(\sum_{i=1}^{n} P_i^{-1} \right)^{-1} . \qquad (7.3)$$

7.2 Examples

The proposed algorithm will now be illustrated for some examples.

Simulated examples

Data with different amount of excitation will be used here. The data are generated from simulation of a control loop with plant transfer function

$$G(s) = \frac{1}{10s+1} e^{-15s}.$$

The controller is of PI-type with $K_c = 0.44$ and $T_i = 13.75$. Noise is added to the process output as filtered (first-order low-pass filter with discrete-time pole in 0.8) white noise. In the first example the setpoint is constant and in the following examples, setpoint steps with increasing length are used. The results are shown in Figures 7.1 to 7.3. The upper plot in each figure is the original data. The light-grey bars denote the detected steady-state intervals. The dark-grey bars denote the selected data. The estimated noise standard deviation is compared with the real one in Table 7.1. In Figure 7.1 no

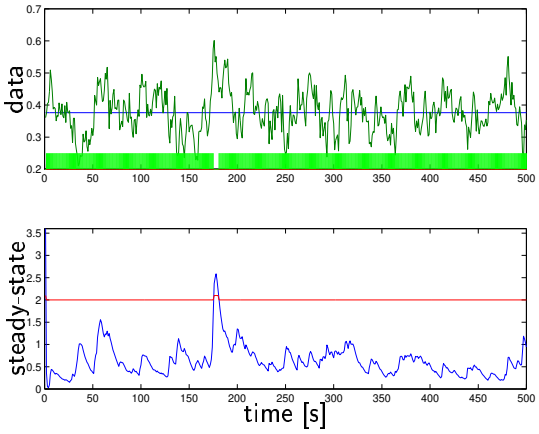


Figure 7.1: Results of the data selection for the data with *constant setpoint*.

data will be selected since the setpoint is always constant. The selection algorithm will never be triggered to start. Figure 7.2 contains a sequence of three steps in the setpoint. As a consequence the selection is started at each of these steps[1] and continues until the data has reached steady-state

[1] As a matter of fact, the selection does also include a fixed number of samples (here 10) *before* the step.

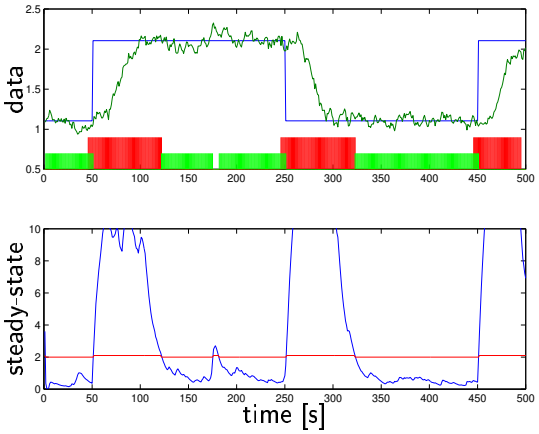


Figure 7.2: Results of the data selection for the data with *several steps (I)*.

example	noise std.dev.	real
no setpoint change	0.09	0.06
several steps I	0.10	0.06
several steps II	0.10	0.06

Table 7.1: Real and estimated noise standard deviations for the simulated examples.

again. The steps are so sparse that there are areas of steady-state between them. The last example, Figure 7.3 is very similar but contains more frequent setpoint changes such that the process output does not reach steady-state in between. The selected data set in this case is therefore not scattered.

Industrial examples

Here several examples using industrial data are shown. All loops are flow control loops in the stock preparation section of a paper mill. The results are shown in Figures 7.4 to 7.7. The estimated noise standard deviations can be seen in Table 7.2. The examples in Figures 7.4, 7.5 and 7.6 are very similar to the simulated ones, i.e. single or a series of steps in the setpoint,

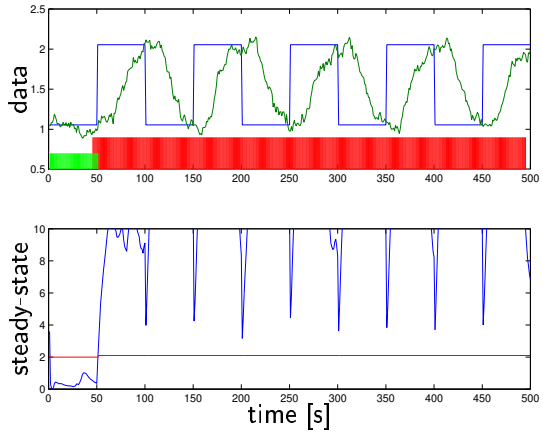


Figure 7.3: Results of the data selection for the data with *several steps (II)*.

data set	noise std.dev.
Ex I	1.69
Ex II	1.04
Ex III	0.94
Ex IV	5.53

Table 7.2: Estimated noise standard deviations for the industrial examples.

thus yielding scattered or coherent selected data sequences. Figure 7.7 shows the more special case of a cascaded loop, i.e. the setpoint of the loop in question is the controller output of an outer loop. As a consequence, the setpoint changes continuously. Notice however that the data selection is not started until a change larger than the noise dependent limit has been occurred (happens at $t = 550$ seconds).

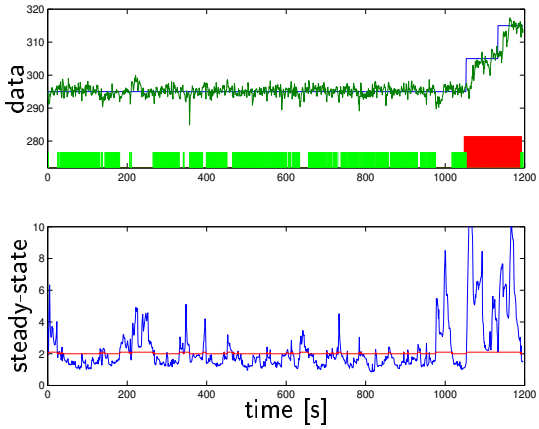


Figure 7.4: Results of the data selection for *Example I*.

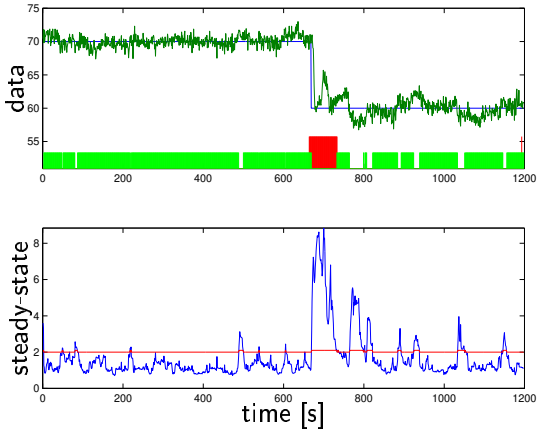


Figure 7.5: Results of the data selection for *Example II*.

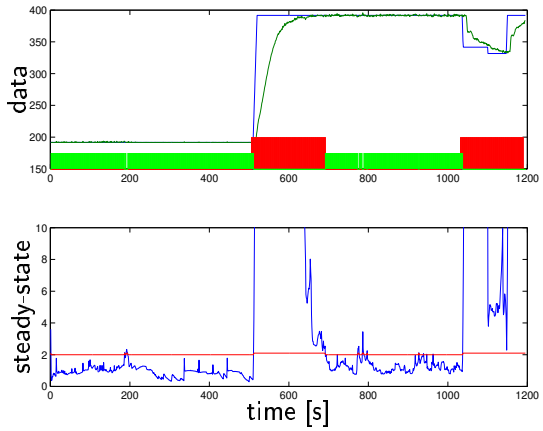


Figure 7.6: Results of the data selection for *Example III*.

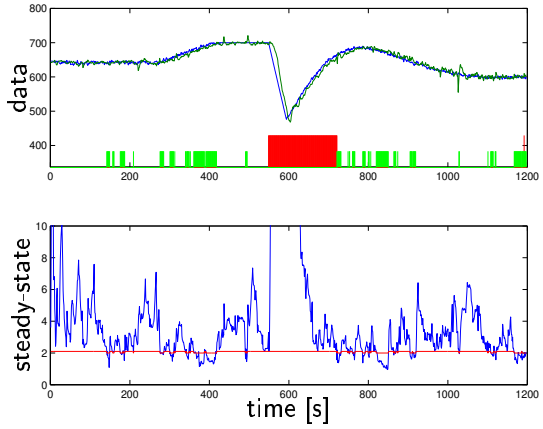


Figure 7.7: Results of the data selection for *Example IV*.

Chapter 8

Dead-time estimation

Assume that suitable identification data have been selected using for example the method proposed in Chapter 7. The problem dealt with in this chapter is the estimation of the process dead-time given an informative measurement data set. This question has also been addressed in (Isaksson *et al.*, 2000). There, several dead-time estimation algorithms have been compared for the use in a monitoring tool using Monte-Carlo simulations. The problem treated was the estimation of the dead-time k in a linear discrete-time model

$$y(t) = G(q^{-1})u(t-k) + H(q^{-1})v(t)$$

where q^{-1} denotes the backward shift operator, and $v(t)$ is zero-mean white noise. The methods tested there are briefly listed here:

1. **ARX-model:** Fit fixed-order ARX-models for different time-delays to the data. Choose the time-delay which gives minimal prediction error.
2. **OE-model:** Same as above but using an output error (OE) model structure, see also (Swanda, 1999).
3. **Laguerre-model:** Fit a Laguerre-network to the data. The time-delay has to be described by unstable zeros. This method will be described in more detail later on,see also (Isaksson, 1997).
4. **Variable regression:** At the time-delay the correlation between control signal and change of the process output signal is maximal (Elnaggar *et al.*, 1991) .

5. **Extended numerator:** Fit an ARX-model with a large number of numerator parameters. Small values of these parameters correspond to time-delay steps,see (Kurz and Goedecke, 1981), (Kurz, 1979).

Note that one can either use the setpoint r or the control signal u as input data because the dead-time from r to y is the same as from u to y. Except that in some cases the controller may introduce an extra delay of one sample. Hence, using the five mentioned estimation methods and two different data sets, one has ten approaches to choose from. In (Isaksson *et al.*, 2000), it was concluded that the most reliable method is the one based on Laguerre models when estimating a model using control signal u and process output y. This method will be used in the following.

The chapter is organised as follows: Section 8.1 gives a brief description of the Method 3. Section 8.2 presents a modification of that method. Examples and comparisons are shown in Section 8.3.

8.1 Laguerre model and Padé approximation

Assume that a Laguerre model has been fitted to the identification data. Then, one has obtained a parameterisation of the plant in so-called Laguerre filters, see e.g. (Wahlberg, 1991):

$$y(t) = \sum_{i=1}^{n_l} c_i L_i(q^{-1})u(t) + v(t) \tag{8.1}$$

where

$$L_i(q^{-1}) = \frac{\sqrt{1-\alpha^2}q^{-1}}{1-\alpha q^{-1}} \left(\frac{-\alpha + q^{-1}}{1-\alpha q^{-1}}\right)^{i-1}. \tag{8.2}$$

The model is linear in the coefficients c_i which can be estimated using the least-squares method on data filtered through the Laguerre filters L_i. It can be viewed as a generalised finite impulse response (FIR) model, where q^{-i} is replaced by the Laguerre filters instead. This parameterisation of the plant has the advantage over, for example, Output Error (OE) and ARX that it can well describe time-delays as part of the model, without choosing an explicit delay variable k. In the following examples $n_l = 10$ and $\alpha = 0.8$ are used. It was found that the resulting dead-time estimate is relatively insensitive to the choice of these design parameters.

The original approach described in (Isaksson, 1997) is to first estimate the Laguerre model (8.1), followed by a calculation of the zeros z_i. The discrete-time zeros are then converted to approximate continuous-time zeros as

$$s_i \approx \frac{1}{T_s} \log(z_i), \tag{8.3}$$

where T_s is the sampling interval. Finally, a comparison is made to a conventional Padé approximation, (Padé, 1892), of the continuous-time dead-time T_d.

$$e^{-sT_d} \approx \frac{1 - \frac{sT_d}{2} + \cdots}{1 + \frac{sT_d}{2} + \cdots} \tag{8.4}$$

If the rest of the plant has no non-minimum phase zeros, then all right-half plane zeros of the estimated process model originate from the Padé approximation. Hence

$$\frac{T_d}{2} = \sum_{RHP} \frac{1}{s_i}. \tag{8.5}$$

The estimate of the discrete-time dead-time k is then obtained as

$$\hat{k} = 1 + \frac{\hat{T}_d}{T_s} = 1 + \frac{1}{T_s} \sum_{RHP} \frac{2}{s_i}. \tag{8.6}$$

A weakness of the above method is that it makes use of two different approximations. Firstly, the conversion of discrete-time zeros to continuous-time introduces an error if the sampling interval is large enough. Secondly, by comparing the continuous-time zeros with a Padé-approximation which itself is an approximation of the true dead-time element. Another problem which may occur is the presence of unstable discrete-time zeros on the negative real axis. These will lead to a complex dead-time estimate if no remedies are taken.

Therefore, in the next section, an alternative method is presented which avoids such problems by *directly* estimating the dead-time from the discrete-time zeros.

8.2 Laguerre model and rational allpass filter

An alternative way to estimate the dead-time directly from the discrete-time zeros of the plant model *without* converting them to continuous-time zeros will be presented now.

Consider the factorisation of the estimated Laguerre model into two parts: a minimum phase (*mp*) part and an allpass (*ap*) part

$$G(z) = G_{mp}(z)\, G_{ap}(z).$$

Such a factorisation can be done for any linear, time-invariant dynamic system, see Figure 8.1 for an example. The allpass part has unit gain for all

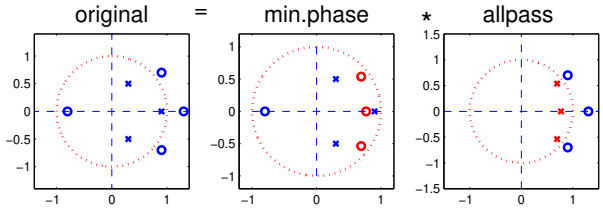


Figure 8.1: Pole(×)-zero(○) distribution when factorising a linear system into a minimum phase and an allpass part.

frequencies and contains all unstable zeros of the original model. The poles of the allpass filter are the unstable zeros mirrored into the unit circle.

Hence, it is reasonable to compare $G_{ap}(z)$ with a good approximation for the dead-time. Note that for discrete-time models a dead-time approximation is usually not necessary since – at least when the dead-time is a multiple of the sampling interval – there is an exact representation in z^{-k}. By the nature of Laguerre filters $G_{ap}(z)$ is, however, a rational transfer function with equal numerator and denominator degrees.

Consider then the following straightforward result.

Theorem 8.1. *Assume that* $G_{ap}(z)$ *is a discrete-time Padé approximation of the dead-time* T_d*, and*

$$\varphi(\omega) = \arg\{G_{ap}(e^{i\omega T_s})\}.$$

Then

$$\hat{T}_d = \lim_{\omega \to 0} \left(-\frac{\varphi(\omega)}{\omega} \right). \tag{8.7}$$

□

Proof: See Appendix C.

Remark 8.1. *The expression (8.7) can of course not be evaluated for $\omega = 0$. For implementation on a computer, a value of e.g. $\omega \approx 10^{-4}$ rad/s has been found to be sufficiently small when using typical industrial data.*

Remark 8.2. *In order to obtain the number of samples of dead-time one has to add one sample to the estimate. Then the number of samples of dead-time is calculated as*

$$\hat{k} = 1 + \frac{\hat{T}_d}{T_s} = 1 + \left. \frac{-\varphi(\omega)}{\omega T_s} \right|_{\omega \ll 1}. \tag{8.8}$$

Remark 8.3. *As noticed already above the estimation method advocated here assumes the system to be minimum phase, save for the dead-time. In a situation when it is important to distinguish between non-minimum phase zeros and dead-time this may be a concern. Since the intended use here is performance monitoring, it is not all that critical if one performance limiting factor is mistaken for another. The calculated achievable performance will still be accurate enough. Swanda and Seborg (1997) designate the described phenomenon as "apparent dead-time".*

The main potential disadvantage with this estimation method is that, due to the inherent approximations, it will never give completely unbiased estimates even in the absence of noise. Moreover, it yields non-integer dead-time estimates. The latter may, however, also be viewed as an advantage, since contrary to most other methods it in fact makes an effort to estimate the true continuous-time dead-time.

8.3 Examples

The methods described in Sections 8.1 and 8.2 will be denoted Method I and II respectively.

Simulated examples

Some simulated examples where the exact dead-time was known will be presented now. For these examples, Monte-Carlo simulations were made.

Example 1. Consider open-loop data from an over-damped process with dead-time,

$$G_1(s) = \frac{e^{-9s}}{(10s+1)(s+1)}.$$

In order to generate identification data, the transfer function was transformed to discrete-time using zero-order hold and a sampling time of $T_s = 1$. Zero-mean white noise with standard deviation $\sigma = 0.05$ was added to the process output. The input was a series of three steps, see Figure 8.2. The results of a

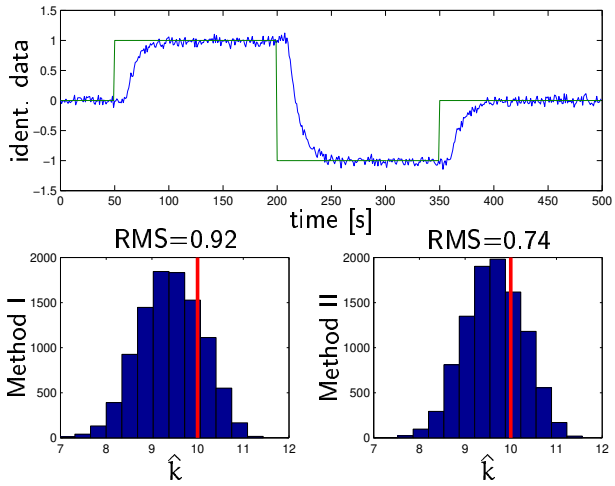


Figure 8.2: Top: Identification data for Example 1. Bottom: dead-time estimation results for both methods. The correct number of samples of dead-time is marked by a vertical line.

Monte-Carlo simulation with $N = 10000$ different noise realisations are also shown in Figure 8.2. It can be seen that Method II yields a lower RMS-value

which is defined as

$$\mathrm{RMS} = \sqrt{\frac{1}{N}\sum_{i-1}^{N}[\hat{k}_i - k]^2},$$

where $\hat{k}_i$ is the estimated number of samples of dead-time from the i-th data set and k is the correct number.

Example 2. Consider a similar case as in Example 1 but now with a dead-time of $T_d = 10$ seconds. Using a sampling rate of $T_s = 5$ seconds results in three samples of dead-time (including one sample due to the discretisation). The noise variance was increased to $\sigma = 0.15$. One realisation of the identification data and the results of the Monte-Carlo simulation are shown in Figure 8.3. Again, a better RMS value was found for Method II.

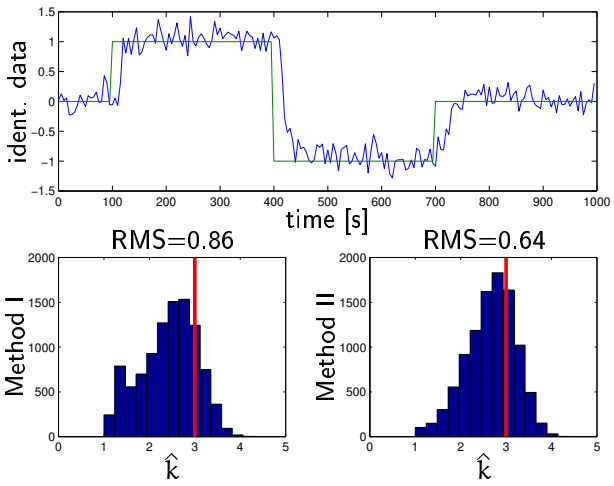


Figure 8.3: Top: Identification data for Example 2. Bottom: dead-time estimation results for both different methods. The correct number of samples of dead-time is denoted by a vertical line.

Industrial examples

Data from industry will be used now to illustrate and compare the two described methods.

Example 3. Consider data from a flow control loop in a paper mill, see Figure 8.4. When estimating the dead-time from this data values of 7.79 (Method I) and 8.27 (Method II) were obtained. From visual inspection of the data it can be stated that the "correct" dead-time should be 5-6 samples. Figure 8.4 also shows the dead-time estimate using Method II for different frequencies (i.e. Equation (8.8) as a function of frequency). Remember that the value for the lowest frequency is chosen since the true phase curve error decreases for decreasing frequency. This can clearly be seen in the bottom plot of Figure 8.4 where the allpass phase curve and the phase curve of a pure dead-time element (with the "correct" dead-time) are plotted as a function of frequency.

Example 4. Consider yet another flow control loop. The "correct" dead-time is known to be 5 samples. The identification data and the estimation results are shown in Figure 8.5. The calculated dead-times in this case where: 5.72 (Method I) and 5.79 (Method II). Again, the values are rather close.

8.4 Data segmentation and dead-time estimation

For the use in an autonomous monitoring tool, the data selection method described in Chapter 7 has to be combined with the dead-time estimation presented in this chapter. The quality of the estimate depends on a number of factors. Sampling frequency, loop bandwidth, controller tuning, process dynamics are such factors. The effects of some of these factors on the properties on the estimate have also been investigated (for different dead-time estimation methods) in (Isaksson *et al.*, 2000).

In the following, the effect of selecting data out of a long data set with a few setpoint changes on the estimate will be investigated via Monte-Carlo simulations.

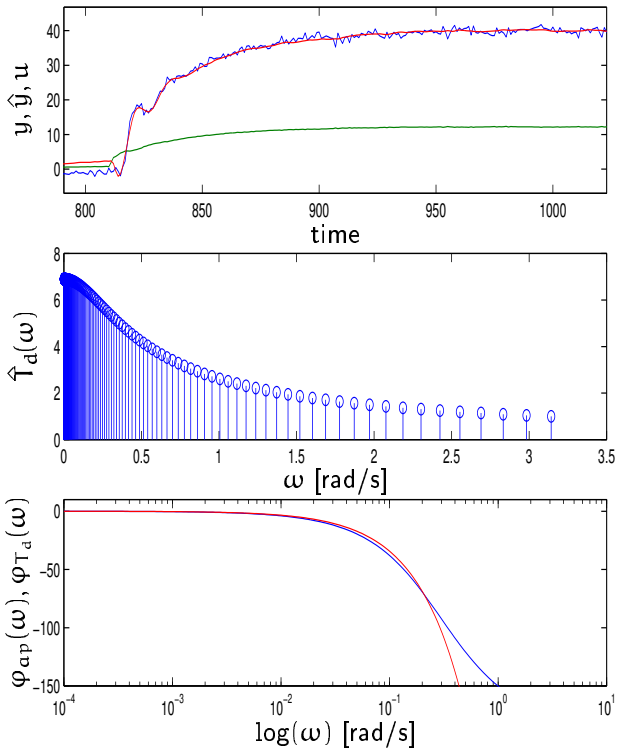


Figure 8.4: Example 3: Top: Identification data, process output y, control signal u and model prediction $\hat{y}$. Middle: Dead-time-estimate for different frequencies. Bottom: Phase curves for allpass part and "correct" dead-time element.

The models and controllers used for simulation are shown in Table 8.4. The sampling interval was $T_s = 1$ for all cases. The noise-free step responses are shown in Figure 8.6. Note the different time-scales! The models and controllers were chosen in order to illustrate the effect of the process time-constant (fast–slow), the controller tuning (tight–sluggish) and the type of process noise assumed (filtered by the process denominator – partly filtered by the process denominator – not filtered at all). For each of the twelve cases a rather long data set was generated (denoted below as the *original*

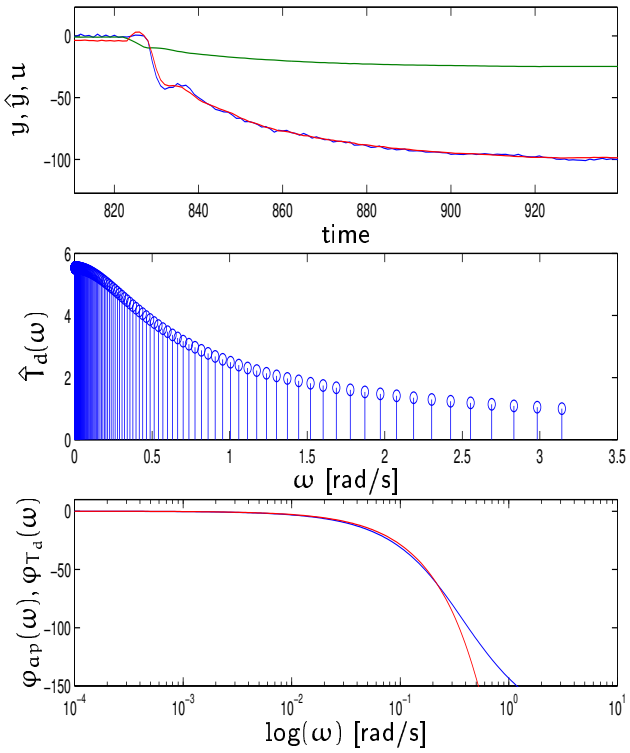


Figure 8.5: Example 4: Top: Identification data, process output y, control signal u and model prediction $\hat{y}$. Middle: Dead-time-estimate for different frequencies. Bottom: Phase curves for allpass part and "correct" dead-time element.

one. Then the segmentation method from Chapter 7 was used to select a shorter data set, denoted below as the *selected* data set.

Results

The complete results for all twelve cases are listed in Table 8.2. Typical data and the resulting histograms for ARX I, FILT III and OE IV are shown in Figures 8.7, 8.8 and 8.9 respectively. From Table 8.2 the following con-

model	gain	poles	T_d	noise model	σ_e^2	K_c	T_i
ARX I	5	[-1,-0.1]	9	$1/A(q^{-1})$	0.1	0.72	6
ARX II	0.05	[-10,-1]	9	$1/A(q^{-1})$	0.01	0.612	51
ARX III	0.05	[-10,-1]	9	$1/A(q^{-1})$	0.02	0.975	13
ARX IV	5	[-1,-0.1]	9	$1/A(q^{-1})$	0.1	0.3	6
Filt I	5	[-1,-0.1]	9	$0.98q/(q-0.2)$	0.1	0.72	6
Filt II	0.05	[-10,-1]	9	$0.71q/(q-0.7)$	0.05	0.612	51
Filt III	0.05	[-10,-1]	9	$0.71q/(q-0.7)$	0.05	0.975	13
Filt IV	5	[-1,-0.1]	9	$0.98q/(q-0.2)$	0.1	0.3	6
OE I	5	[-1,-0.1]	9	1	0.1	0.72	6
OE II	0.05	[-10,-1]	9	1	0.02	0.612	51
OE III	0.05	[-10,-1]	9	1	0.1	0.975	13
OE IV	5	[-1,-0.1]	9	1	0.1	0.3	6

Table 8.1: Models and controllers used in the Monte-Carlo simulations. $A(q^{-1})$ denotes the process denominator.

clusions can be drawn: The estimation based on the short data set usually exhibits a larger variance but the bias is much smaller. In fact so much smaller that the RMS error sometimes becomes significantly smaller for short (segmented) data than long data sets. This fact has been noted before and is discussed in, for example, (Carrette *et al.*, 1996) for the framework of ARX-modelling.

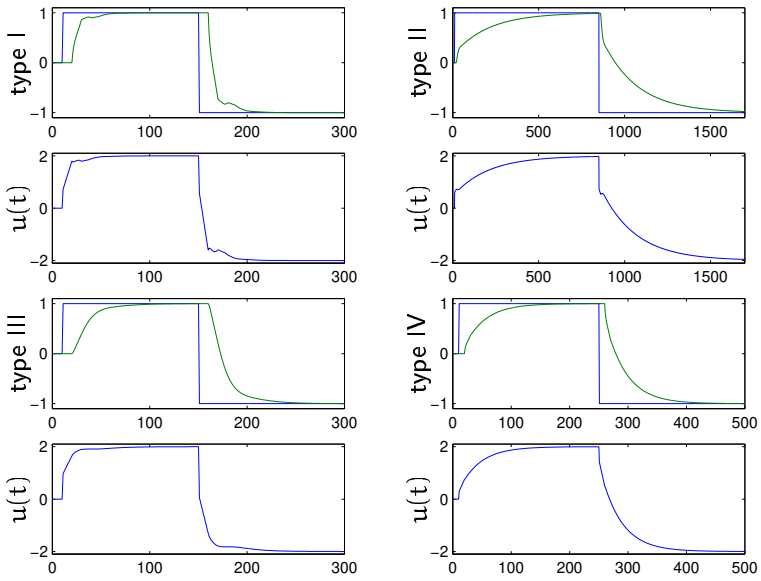


Figure 8.6: Step responses for the four model types I,II,III and IV.

	RMS		mean		std	
noise	sel.	orig.	sel.	orig.	sel.	orig.
ARX I	1.54	**1.01**	**9.45**	10.79	1.44	**0.62**
ARX II	**2.31**	4.61	**10.69**	14.21	2.20	**1.88**
ARX III	**2.45**	4.27	**10.85**	13.82	2.31	**1.90**
ARX IV	2.41	**2.14**	**9.85**	11.96	2.40	**0.86**
Filt I	1.48	**0.90**	9.16	**9.28**	1.22	**0.55**
Filt II	**3.53**	8.79	**12.08**	18.70	2.85	**1.24**
Filt III	**1.80**	3.26	**10.71**	13.13	1.66	**0.91**
Filt IV	2.63	**1.44**	8.67	**8.90**	2.27	**0.94**
OE I	**1.61**	2.29	**8.78**	7.79	1.04	**0.60**
OE II	0.87	**0.78**	**9.86**	9.59	0.86	**0.66**
OE III	1.72	**1.03**	10.73	**10.10**	1.56	**1.03**
OE IV	**3.04**	5.02	**7.63**	5.10	1.90	**1.09**

Table 8.2: Results of the Monte-Carlo simulations. The bold style denotes the better value, i.e. if the estimation was better for the original or the selected data set.

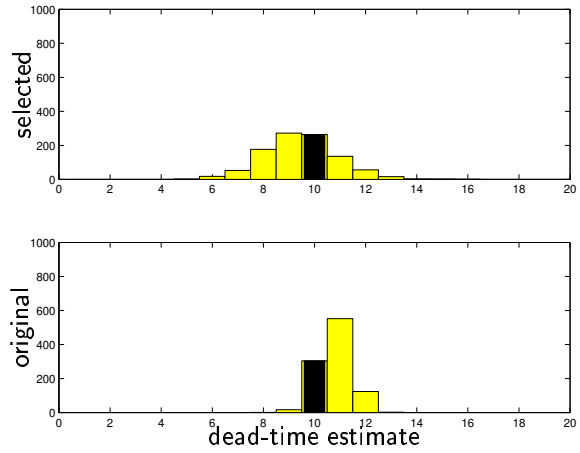


Figure 8.7: Model ARX I. Monte-Carlo simulation results. Distribution of estimated number of samples of time-delay. Top: Short data set, bottom: long data set.

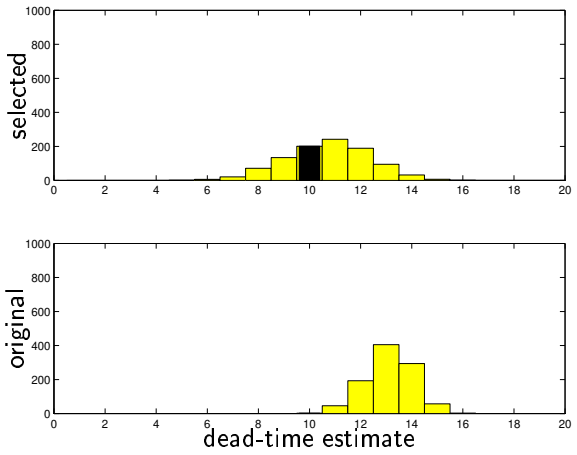


Figure 8.8: Model FILT III. Monte-Carlo simulation results. Distribution of estimated number of samples of time-delay. Top: Short data set, bottom: long data set.

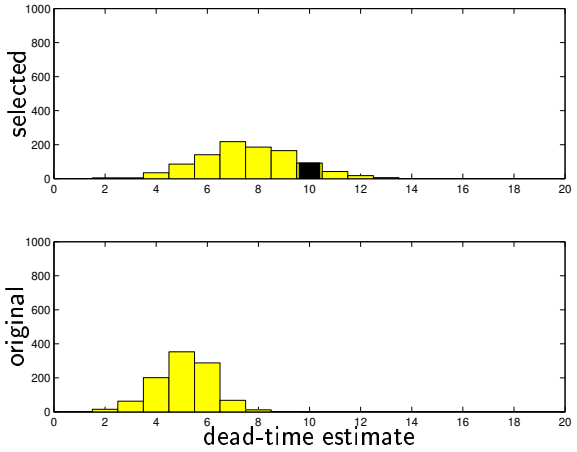


Figure 8.9: Model OE IV. Monte-Carlo simulation results. Distribution of estimated number of samples of time-delay. Top: Short data set, bottom: long data set.

Part III

Oscillation diagnosis

Chapter 9

Oscillations in process control loops

Oscillations in process control loops are a very common problem. In Canadian studies it was found that up to 30% of all loops may be oscillating in typical pulp and paper processes, see (Bialkowski, 1992) and (Ender, 1993). The fact that some loops oscillate is often known to the operators. However, since the cause is usually unknown, appropriate measures may not be obvious. Therefore, oscillating loops tend to be neglected or be put in manual mode. Unfortunately, none of these measures maintains the benefit of control.

It is therefore important that a monitoring tool can do more than pure oscillation detection. It should also be able to indicate likely causes. A third (and very desirable) step would be to indicate measures in order to remove the oscillation like *"de-tune the controller", "valve maintenance needed", "check neighbouring loop", etc.*

However, the automatic diagnosis of oscillations without performing any experiments is a difficult task and may often not even be possible for some causes. In this chapter the most important reasons for oscillation in process control loops are briefly discussed. However, first methods for automatic detection of oscillations are reviewed.

9.1 Automatic detection of oscillations

Before being able to diagnose different oscillations, an automatic monitoring tool has to detect oscillating signals. Even though a simple task for the human eye, it is not trivial for a computer to perform. One of the problems is the lack of a simple mathematical definition of an oscillation which can be checked automatically.

Textbooks often discuss spectral analysis in connection with oscillations. Again, for the human eye a peak in the spectrum of a signal is simple to detect. Making this process automatic is non-trivial however. The existence of several peaks or very broad maxima are difficult to handle automatically.

Hägglund (1995) has proposed a method in the time-domain which considers the integrated error (IE) between all zero-crossings of the signal. If the (IE) is *large enough*, a counter is increased. If this counter exceeds a certain number, an oscillation will be indicated. In order to quantify what *large enough* means, the ultimate frequency of the loop in question (as may be known from identification using relay-experiments) is used. Alternatively, one may use the integral time of the controller, assuming that the controller is reasonably tuned. This method is very appealing but has two disadvantages: Firstly, it is assumed that the loop oscillates at its ultimate frequency which may not be true, e.g. in the case of stiction. Secondly, the ultimate frequency is not always available and the integral time may be a bad indicator for the ultimate period. A strength of the method is that it quantifies the size of the oscillation.

A very similar idea was used in the detection method by Forsman and Stattin (1999). There, the time instants of all zero-crossings and all IEs are computed and compared pairwise, see Figure 9.1. Then, all IEs and zero-crossings which are pairwise approximately equal are counted. This number is divided by the total number of half-periods yielding an index between 0 (no oscillation) and 1 (perfect oscillation). A reasonable threshold was found to be 0.3. An advantage with this method is that an estimate of the frequency and oscillation amplitude are computed as a by-product. There are some tuning parameters which determine what *approximately equal* means but these do usually not have to be changed.

Yet another method was presented by Seborg and Miao (1999). Here the auto-correlation function of the signal in question is estimated. Then a measure similar to the damping ratio is computed. If this measure exceeds

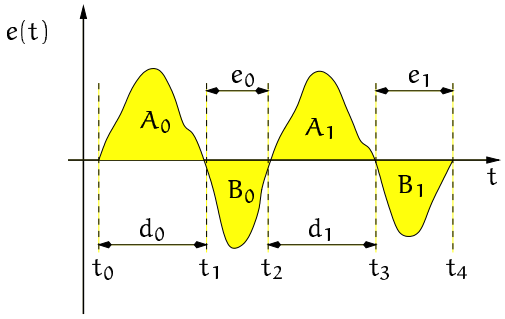


Figure 9.1: Definition of time-domain variables for oscillation detection.

a certain threshold, an oscillation is indicated.

Note that only the method by Hägglund (1995) makes an attempt to quantify the oscillation amplitude. The other two mentioned approaches may be fooled by very small oscillation amplitudes. The question whether such oscillations should be detected or not is a bit philosophical. There must be a reason for the oscillation, on the other hand, even if it does not do any harm.

9.2 Reasons for oscillations

Static friction

High static friction (stiction) in a control valve often causes oscillations. The mechanism is simple: Assume that a valve is stuck (stick-phase) in a certain position due to high static friction. The (integrating) controller then increases the setpoint to the valve until the static friction can be overcome. Then the valve breaks off and moves (note that the dynamic friction is smaller than the static friction) to a new position (slip-phase) where it sticks again. The new position is usually on the other side of the desired setpoint such that the same process starts again in the opposite direction. A very simple simulation model, based on the work by Olsson (1996) was presented in (Horch and Isaksson, 1998). Typical stick-slip behaviour results in a rectangular process output and a triangular control signal, see Figure 9.2 for an industrial

example. Oscillations due to stiction are mostly tuning-independent as long as there is integral action. Oscillations due to stiction can hence usually not be removed by re-tuning the controller. An ad hoc control strategy – which could be used until the next maintenance stop – could be the use of a dead-zone in the controller as suggested by Ender (1997). The tuning of PI-controllers in loops with stiction was also discussed by Piiponen (1998).

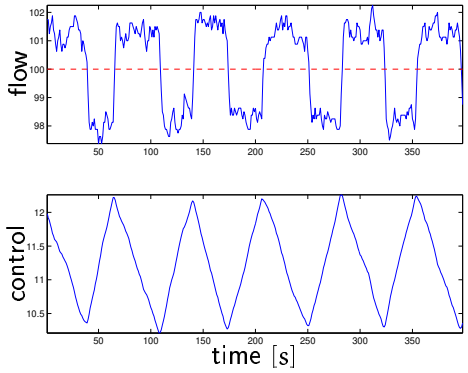


Figure 9.2: Typical behaviour of a control valve with stiction. Top: measured flow (solid) and flow setpoint (dashed). Bottom: control signal.

Dead-zone

Dead-zone is the combined nonlinearity that represents the amount the input signal needs to be changed before the actuator actually moves. This kind of nonlinearity can for example occur in the control system software. For the analysis done in this chapter, the concept of the describing function will be used. The describing function method is a standard analysis tool for control loops containing static nonlinearities, see for example Atherton (1981). The describing functions used in the following can be found in most textbooks about nonlinear control.

Figure 9.3 shows the time-domain behaviour of a dead-zone element. The describing function is given by

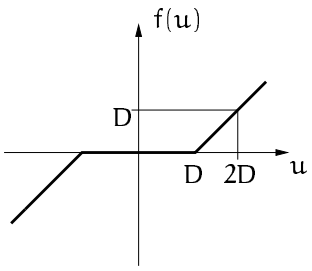


Figure 9.3: Time-domain characteristic of a dead-zone element.

$$N(C) = \begin{cases} 1 - \frac{2}{\pi}\left(\arcsin\left(\frac{D}{C}\right) + \frac{D}{C}\sqrt{1-\frac{D^2}{C^2}}\right) & C \geq D \\ 0 & C < D \end{cases}$$

See Figure 9.4 for a plot of $N(C)$. The negative inverse of this always

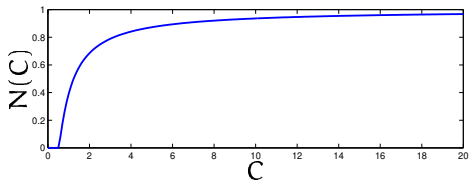


Figure 9.4: Describing function for a dead-zone nonlinearity with $D = 0.5$.

covers the negative real axis to the left of -1. A *stable* limit cycle can only be obtained if the closed-loop without the nonlinearity itself crosses the negative real axis left of the critical point -1 without being unstable at the same time. A possible scenario to satisfy such a condition can for example be a PID-controlled plant which contains a fist-order dynamics and an integrator (e.g. a level control loop).

An industrial example of a level control loop where an oscillation is caused by a dead-zone in the control system is shown in Figure 9.5.

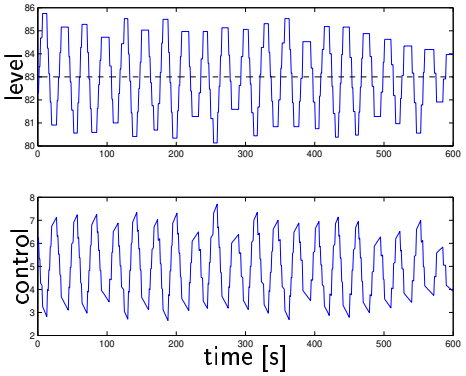


Figure 9.5: Example of a level control loop which oscillates due to deadband.

Backlash

Backlash is present in every mechanical system where the driving element is not directly connected with the driven element. Figure 9.6 shows the time-domain behaviour for a backlash element. Backlash is a dynamic nonlinearity,

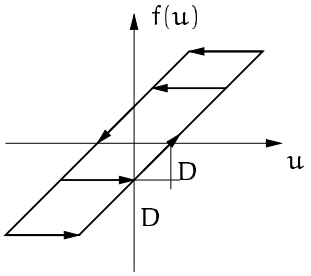


Figure 9.6: Time-domain characteristic of a backlash element.

i.e. its state depends on the input *and* past states. The describing function for the backlash is therefore a complex valued function. Assuming a unit slope in the input-output relation, the describing function for $C \geq D$ where

D is the width of the dead-band is given as

$$\begin{aligned}\Re\mathfrak{e}\{N(C)\} &= \frac{1}{\pi}\left(\frac{\pi}{2}+\arcsin\left(1-\frac{2D}{C}\right)+2\left(1-\frac{2D}{C}\right)\sqrt{\frac{D}{C}\left(1-\frac{D}{C}\right)}\right)\\ \Im\mathfrak{m}\{N(C)\} &= -\frac{4D}{\pi C}\left(1-\frac{D}{C}\right).\end{aligned}$$

It is not possible – for general process models – to state simple conditions for a limit cycle to occur. Clearly, the loop without the nonlinearity may be stable or unstable. For a limit cycle to occur it must be higher than first order, which is always true for a first-order plus dead-time process controlled by a PI-controller. An example of the negative inverse of the describing function is shown in Figure 9.7. As can be seen, de-tuning of the controller

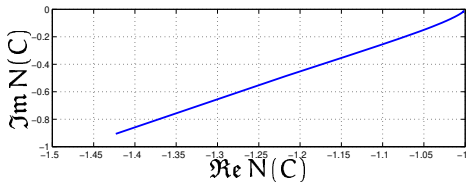


Figure 9.7: Plot of the negative inverse of the describing function for a backlash nonlinearity with unit slope.

will probably remove a backlash-induced oscillation in many cases.

Saturation

Real control signals do always have constraints. Therefore, alarm limits are set for each control signal. If the controller is tuned such that the loop is unstable, there will be an oscillation due to the saturation nonlinearity. Figure 9.8 shows the time-domain behaviour for a backlash element.

The negative inverse of the describing function for a saturation nonlinearity is located on the negative real axis left of the critical point. For an oscillation due to saturation, the phase difference between control signal and the process output will therefore ideally be 180 degrees. An industrial example is shown in Figure 9.9.

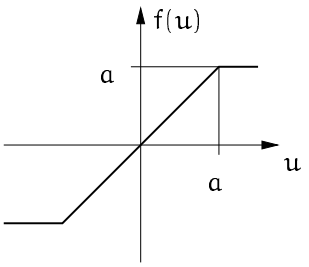


Figure 9.8: Time-domain characteristic of a saturation element.

Quantisation

Quantisation is always present in modern control systems since analog values have to be discretised and vice versa. Especially in older systems where the converters have a relatively low resolution this may become a problem. Since oscillation due to quantisation is less known, it will here be discussed in some more detail.

The input-output characteristic of a quantiser with a quantisation level of q is given by

$$Q(x) = q \text{ round } (x/q). \tag{9.1}$$

This nonlinearity is plotted in Figure 9.10. The describing function for this nonlinearity can be found by either extrapolating results for the three-point and five-point nonlinearities or by considering the addition of signum-with-dead-zone functions (Franklin *et al.*, 1998). The describing function is

$$N(C) = \begin{cases} 0 & 0 \le C < \frac{q}{2} \\ \frac{4q}{\pi C} \sum_{i=1}^{n} \sqrt{1 - \left(\frac{2i-1}{2C} q\right)^2} & \left(n - \frac{1}{2}\right) q \le C \\ & < \left(n + \frac{1}{2}\right) q \end{cases} \tag{9.2}$$

and is plotted in Figure 9.11. The variable n denotes the number of the step in the quantisation curve (Figure 9.10) which corresponds to the actual value of the amplitude C. It is noteworthy that the maximum of $N(C)$ does not change with the quantisation level q. It can be calculated as

$$\max_C N(C) = N(q/\sqrt{2}) = \frac{4}{\pi}.$$

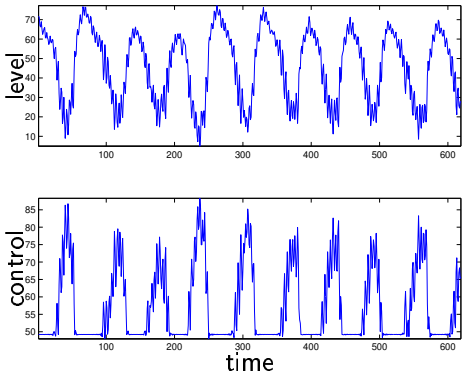


Figure 9.9: Data from a preheater level control loop. The oscillation is due to tight tuning in connection with saturation.

The fact that the maximum of $N(C)$ is constant even for $q \to 0$ reminds of the Gibbs phenomenon, since the describing function is based on a truncated Fourier series. This has an interesting implication on the analysis using the describing function method. Since $N(C)$ is purely real, the negative inverse of $N(C)$ will be on the negative real axis only. It starts in $-\infty$ when $0 \leq C < q/2$. The maximal value is then given by

$$\max_C \frac{-1}{N(C)} = -\frac{\pi}{4} \approx -0.785. \tag{9.3}$$

Note that the function $-1/N(C)$ alternates around the critical point -1 which can lead to ambiguities in the estimation of the oscillation amplitude.

If remains to discuss what happens for $q \to 0$, i.e. when the quantisation nonlinearity vanishes completely. As can be seen from Figure 9.11, if $q \to 0$, the function $N(C)$ will be squeezed together without changing its shape in the vertical direction. For the negative inverse, $-1/N(C)$, this means that the complete curve will not change with q but so does the amplitude dependence. Assume that there is an intersection between the Nyquist curve and the describing function for some finite q. Then there will still be an intersection no matter how much q is decreased. The amplitude of the oscillation will, however, tend to zero.

For applying describing function analysis to a loop with quantization,

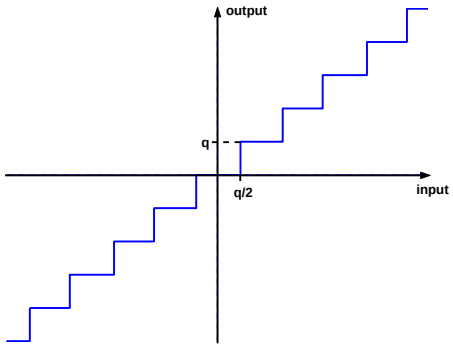


Figure 9.10: Time-domain characteristic for a quantiser.

assume that the controller $F(s)$ stabilises the linear part of the loop. Intersection of $-1/N(C)$ and the frequency function $L(i\omega) = F(i\omega)G(i\omega)$ can then occur in several ways. Firstly, if $\sphericalangle L(i\omega)$ is smaller than $-180°$ for small frequencies. In that case, the intersection lies in the interval $[-\infty, -1]$. Secondly, if $L(i\omega)$ crosses the negative real axis in the interval $[-1, -\pi/4]$.

Simulated example. Consider an example where

$$\begin{aligned} G(s) &= \frac{10}{(0.1s+1)(10s+1)} e^{-10s}, \\ F(s) &= 0.05\left(1 + \frac{1}{10s}\right), \end{aligned}$$

and the quantisation level is $q = 0.05$. The resulting simulation is shown in the upper part of Figure 9.12. The intersection of $L(i\omega)$ with the negative real axis is at $s = -0.33$ at a frequency of $0.156\ \mathrm{rad/s}$. As can be seen, there is no limit cycle. When the controller gain is changed from 0.05 to 0.13, it can be seen that a limit cycle is present. In this case, the intersection of $L(i\omega)$ with the negative real axis is at $s = -0.84$ at a frequency of $0.156\ \mathrm{rad/s}$ yielding an oscillation period of about 40 seconds.

Industrial example. Consider a flow control loop which is oscillating, see left part of Figure 9.13. The reason for this oscillation was found to be an 8-bit resolution in the D/A-converter, yielding a quantisation level $q = 0.5\%$of

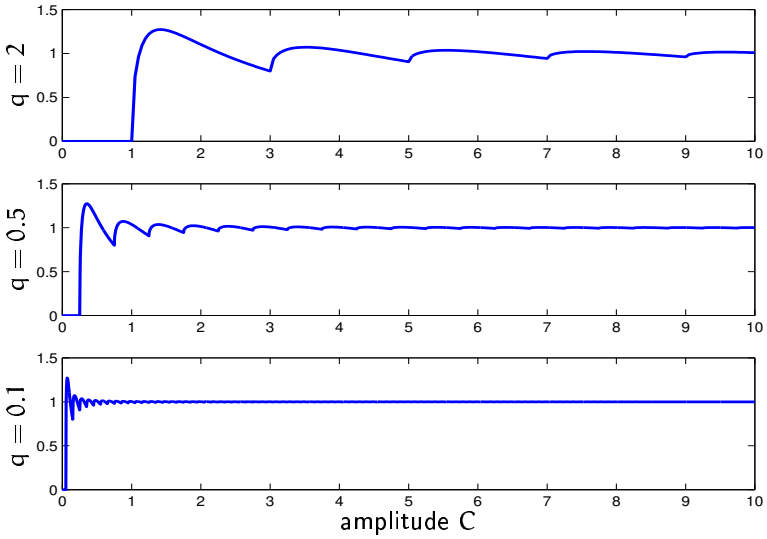


Figure 9.11: Describing function of a quantiser for different levels of quantisation.

the control signal. After a 12-bit converter was installed, the oscillation has decreased drastically, see right part of Figure 9.13. The explanation is that the quantisation level was now decreased to $q = 0.06\%$. The theory then predicts that the amplitude should decrease to $0.06/0.5 = 0.12$ times the original amplitude.

External oscillating disturbances

Despite the use of single-loop controllers in the process industry, many basic control loops are coupled. Therefore, if one loop is oscillating, it will likely influence other loops too. In many cases the oscillations are in a frequency range such that the controllers cannot remove them. Then an oscillation is present even though the controller is well tuned (it might have been tuned for some other control task). Industrial data for this kind of root cause will be shown later, see Figure 10.9.

External disturbances are a challenge for an automatic monitoring system. When having detected an oscillation, it is important to distinguish between

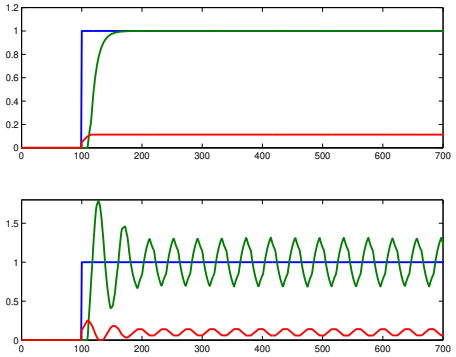


Figure 9.12: Simulation of a loop with a quantiser. Top: no intersection with describing function. Bottom: intersection.

internally and externally caused oscillation.

Other reasons

There are of course other reasons for oscillations. It is for example possible that two well-tuned single loops start oscillating when they are coupled (e.g. due to nonlinearity or multivariable effects). However, using normal operating data and without any process knowledge it seems to be almost impossible to diagnose such a cause.

Another cause which has not been mentioned so far is an oscillating loop setpoint. This problem has been analysed by Ettaleb *et al.* (1996) and will typically only appear in cascaded control loops.

Of course, often there may be several causes at the same time responsible for a certain oscillation, as Figure 9.14 tries to illustrate. This makes the diagnosis even more difficult.

9.3 Signal-based diagnosis

The purpose of this section is to try to motivate – by means of a simulated example – that some causes of oscillations may be impossible to distinguish

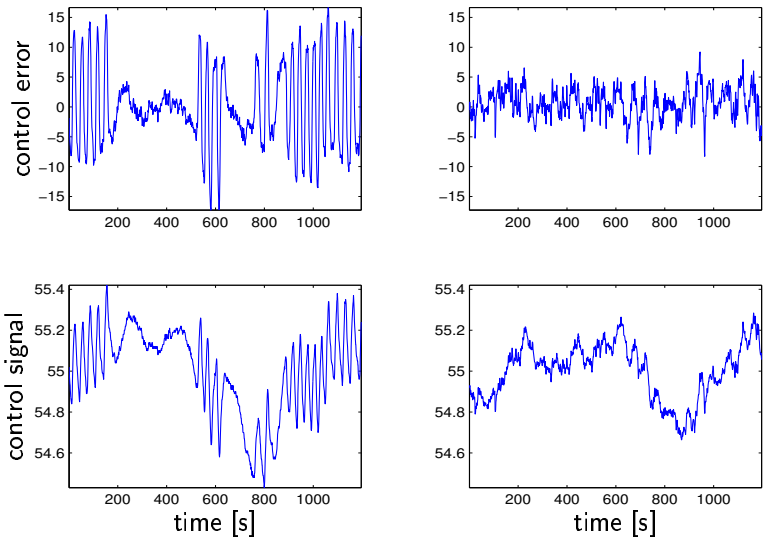


Figure 9.13: Data from a flow control loop. Left: Oscillation due to quantisation in the D/A-converter. Right: Same loop with higher resolution in D/A-converter. Top: control error, bottom: control signal.

between when no process knowledge is available. It is important to recall the assumptions used in this chapter. We assume that

1. the available signals are setpoint, process output and control signal,
2. no specific process knowledge is available,
3. the setpoint is constant,
4. no experiments are made.

In the following, a simulated example will be presented using three different causes for oscillations, namely, saturation, backlash and external disturbances. It will be shown that they lead to signals which are almost identical. It will therefore be very difficult (if not impossible) to distinguish between these cases simply by using normal operating data without any process knowledge.

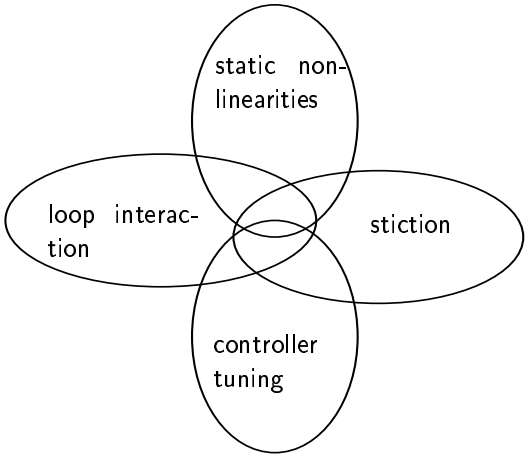


Figure 9.14: Typical reasons for oscillations in process control systems.

A simulated example

Consider a single-input single-output (SISO) loop as shown in Figure 9.15.

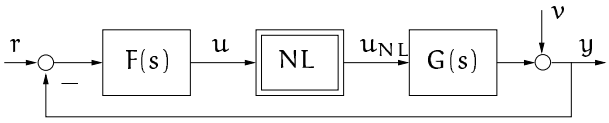


Figure 9.15: A simple SISO-control loop with actuator nonlinearity

Let then the process be described by

$$G(s) = \frac{2}{(3s+1)(10s+1)} e^{-sT_d}$$

and the PI-controller as

$$F(s) = 2\left(1 + \frac{1}{15s}\right).$$

Consider first the case when the time-delay is $T_d = 3$ and the actuator nonlinearity (NL) is pure backlash (dead-band = 1). A simulation of the control loop in Figure 9.15 is shown in Figure 9.16. Then consider the case

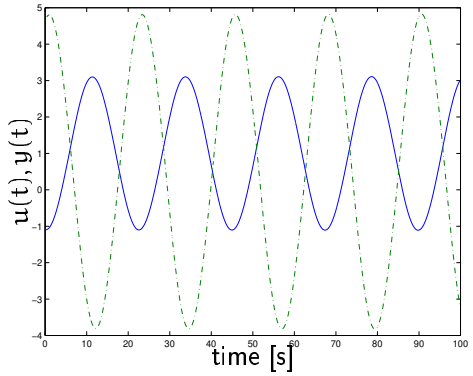


Figure 9.16: Simulation with backlash. Process output (solid) and control signal (dash-dotted).

where the actuator nonlinearity is a saturation (limit ± 1). The controller gain was increased to $K_c = 3$, making the loop unstable. The resulting simulation is shown in Figure 9.17. Finally, consider the case where the actuator is linear

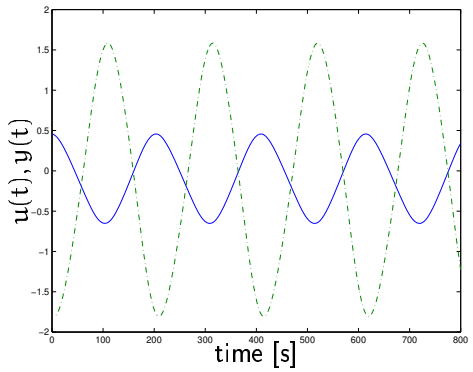


Figure 9.17: Simulation with saturation. Process output (solid) and control signal (dash-dotted).

($\equiv 1$) and there is an oscillating load disturbance $v(t) = \sin(0.25t)$. The time-delay is again $T_d = 3$. The resulting simulation is shown in Figure 9.18.

It can be seen from the three simulations that the different scenarios result

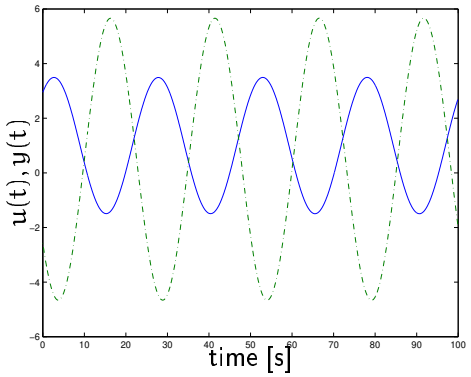


Figure 9.18: Simulation with oscillating load disturbance. Process output (solid) and control signal (dash-dotted).

in almost equal signals. All of them are almost purely sinusoidal and both the phase shift and the amplitude ratio between $u(t)$ and $y(t)$ do not differ significantly.

Conclusions

Using a simulated example, it was argued that it may in fact be impossible to distinguish between certain causes of oscillations based on normal operating data unless some process knowledge is available.

For diagnosis, it is necessary to be able to extract some feature which is unique for a certain cause of oscillation. For the case of static friction (stiction) in control valves, there is such a feature, see Chapters 10 and 11.

In practice it may not be very important to distinguish between for example backlash and dead-band. A more relevant distinction would be if an oscillation is generated internally or externally. This gives an indication in *which* loop to look for a problem. As a matter of fact, such a distinction can be made if one assumes that the process dead-time or a simple process model (see Chapter 12) are available. If such information is not available, it may be obtained by event-triggered estimation

Chapter 10

Detection of static friction for self-regulating plants

Oscillations usually indicate a severe deterioration of performance in process control loops. When an oscillation has been detected, one would like a monitoring system to indicate the possible cause of the problem in order to enable the operators to alleviate the problem. Only few methods to solve this task have been proposed, see e.g. (Deibert, 1994), (Ettaleb *et al.*, 1996), (Hägglund, 1995), (Horch and Isaksson, 1998), (Taha *et al.*, 1996) or (Wallén, 1997). Unfortunately all the approaches mentioned require either detailed process knowledge, user-interaction or rather special process structures.

It has been motivated in Chapter 9 that it might be impossible to distinguish between certain oscillation causes based on signal information only. However, in this chapter it will be shown that the measurements in the case of static friction (stiction), see also Chapter 9.2, show a rather unique behaviour. This fact will enable a simple diagnosis procedure.

Ideally, the stiction phenomenon results in signals as shown in the left half of Figure 10.1. In basically all other cases, the signals are more or less sinusoidal, as indicated in the right half of Figure 10.1. In practice, of course, the signal shapes do vary quite a lot. This is a dilemma for all methods which use the signal shape itself for diagnosis. A very robust property of the signals in the stiction case is the following: If the left signals in Figure 10.1 are multiplied for all samples and added up, the sum will be

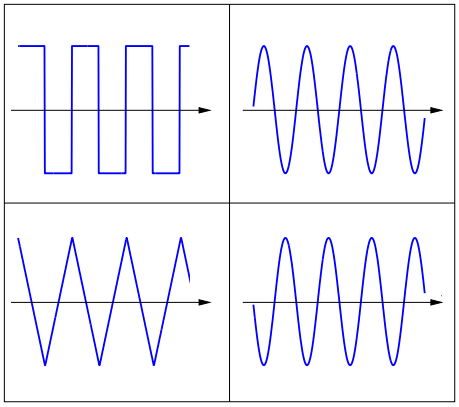


Figure 10.1: Ideal signals. Stiction (left), no stiction (right), process output (top) and control signal (bottom).

exactly zero. If then the same thing is done for the signals on the right hand side of Figure 10.1, the sum will be different from zero (in this case negative). Then, consider the case where the signals are shifted against each other prior to multiplication and summation. This procedure will yield the cross-correlation function between the two signals. Doing this for both cases in Figure 10.1, the (ideal) cross-correlation functions as shown in Figure 10.2 are obtained. Now, one can define a simple strategy to diagnose stiction:

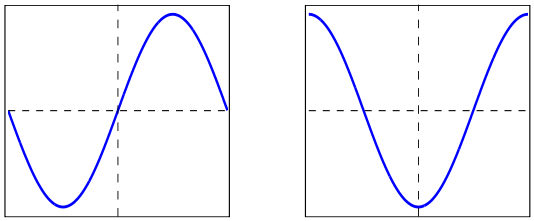


Figure 10.2: Cross-correlation between process output and control signal. Stiction (left), no stiction (right).

If the cross-correlation function (CCF) between controller output and process output is an odd function (i.e. asymmetric w.r.t .the vertical axis), the likely cause of the oscillation is stiction. If the CCF is even (i.e. symmetric w.r.t .the vertical axis), then stiction is not likely to having caused the oscillation.

Note that the proposed method will work under the following assumptions which will be discussed later on.

1. An oscillation has already been detected
2. The plant itself is not integrating
3. The controller has (significant) integral action
4. The loop does not handle compressible media

Note that the method may give questionable results if any of these assumptions are violated.

The remainder of this chapter is organised as follows: First, the algorithm based on the CCF between process output and control signal is presented. The method has been extensively tested on industrial data. Examples where the cause of the oscillation was known are presented in Section 10.2. After a theoretical motivation in Section 10.3, some conclusions are given.

10.1 The cross-correlation function

In this section, a simple algorithm is described that allows diagnosis of stiction as motivated in the last section. As mentioned there, this will be done by calculating the CCF between the process output $y(t)$ and the control signal $u(t)$. The CCF between the discrete-time stationary (ergodic) signals $u(k)$ and $y(k)$ is defined as

$$r_{uy}(\tau) = E\{u(k+\tau)y(k)\}. \tag{10.1}$$

Assume furthermore that the data is pre-treated so that it is zero mean. Since the available data sets are of a finite length N, the correlation function

cannot be calculated exactly and the computation has to be done from finite data. The estimate of $r_{uy}(\tau)$ can be obtained by

$$r_{uy}(\tau) \approx \begin{cases} \sum_{k=0}^{N-\tau} u(k+\tau)\, y(k) & \text{for } \tau \geq 0 \\ r_{yu}(\tau) & \text{for } \tau < 0. \end{cases} \tag{10.2}$$

Note that for the purpose of this paper normalisation is not important and can be chosen arbitrarily.

For automatic distinction between odd and even correlation functions, it would be sufficient to consider the CCF at lag zero. However, for a practically working algorithm it was found to be better to make use of the CCF up to the first zero-crossing in each direction.

For a human it is simple to tell whether or not a function is odd or even. For automatic diagnosis, one can define different measures to distinguish between odd and even functions. Note that it is important that a procedure for automatic distinction between odd and even functions needs to have a dead-zone. That means that the method must not give any indication if the CCF is neither odd nor really even. This may for example be the case when there is a compressible medium in the loop.

For (one possible) automatic distinction of odd and even CCFs, define the following measures:

$$\begin{aligned} \tau_r &= \text{zero crossing for positive lags} \\ -\tau_l &= \text{zero crossing for negative lags} \\ r_0 &= \text{CCF at lag 0} \\ r_{opt} &= \text{sign}(r_0) \cdot \max_{\tau \in [-\tau_l, \tau_r]} |r_{uy}(\tau)| \\ \Delta\tau &= \frac{|\tau_l - \tau_r|}{\tau_l + \tau_r} \\ \Delta\rho &= \frac{|r_0 - r_{opt}|}{|r_0 + r_{opt}|}, \end{aligned}$$

see also Figure 10.3. Theoretically, it would be sufficient to consider either $\Delta\tau$ or $\Delta\rho$. However, the use of both variables will result in a more reliable procedure. For automatic distinction between the two mentioned oscillation cases, one has to set up limits for $\Delta\tau$ and $\Delta\rho$. Note that both test variables are bounded in the interval $[0 \ldots 1]$ where 0 corresponds to an even CCF and 1 to an odd one.

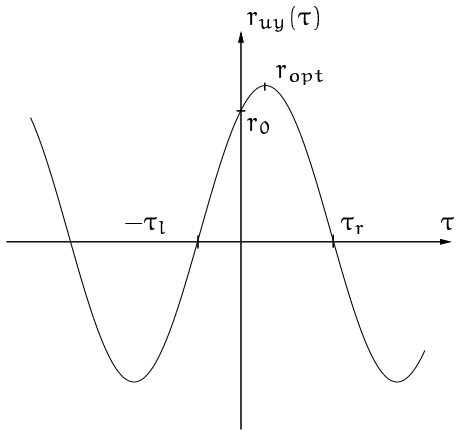


Figure 10.3: Definition of key variables for the correlation function.

Let now $\Delta\phi$ denote the deviation of the CCF from the "ideal" position as was shown in Figure 10.2. Of course, one has to allow a certain deviation range for both cases. In between, however, it is important to have an interval where no decision is taken. This interval corresponds to a CCF which is neither odd nor even. This will typically happen when the oscillation is strongly asymmetrical. It may be an indication of a more unusual problem like for example sensor or other equipment faults.

If all three intervals are defined equally large, we allow a deviation of 1/12th of the period (T_p) of the CCF for each case. See Figure 10.4. Using

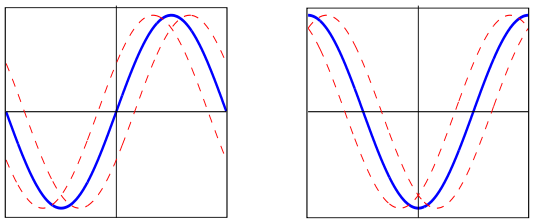


Figure 10.4: Allowed deviation ($\Delta\phi = \pm T_p/12$) of the CCF from the ideal positions. Stiction (left) and non-stiction case (right).

these allowed deviation intervals, the values for $\Delta\tau$ and $\Delta\rho$ are related to the diagnosis as follows:

$$\left.\begin{array}{c} 0.0 < \Delta\rho \leq \frac{2-\sqrt{3}}{2+\sqrt{3}} \\ 0.0 < \Delta\tau \leq \frac{1}{3} \end{array}\right\} \Rightarrow \text{no stiction}$$

$$\left.\begin{array}{c} \frac{2-\sqrt{3}}{2+\sqrt{3}} < \Delta\rho \leq \frac{1}{3} \\ \frac{1}{3} < \Delta\tau \leq \frac{2}{3} \end{array}\right\} \Rightarrow \text{no decision} \qquad (10.4)$$

$$\left.\begin{array}{c} \frac{1}{3} < \Delta\rho < 1.0 \\ \frac{2}{3} < \Delta\tau < 1.0 \end{array}\right\} \Rightarrow \text{stiction}$$

Note that these results are derived assuming a pure sinusoidal CCF. The rules are also illustrated graphically in Figure 10.5.

It may be interesting to know what the limits of $\Delta\tau$ and $\Delta\rho$ are if *no* dead-zone is used. In that case, the deviation from each ideal CCF is allowed to be 1/8th of a period. The limits for $\Delta\tau$ and $\Delta\rho$ are then as depicted in Figure 10.5.

10.2 Industrial examples

In this section, the new method will be evaluated on real-world data sets, collected from different pulp and paper mills. In the following, for each data set, the control signal (dashed) - process output (solid) are plotted (top plot) together with their CCF estimate (bottom plot).

Loop interaction I Consider two coupled oscillating loops from a paper mill. Figure 10.6 shows a process schematic. Pulp is diluted with water in order to obtain a desired consistency. The water flow is controlled by one control valve and the pulp flow by another. The water valve is known to have static friction that is too high, whereas the pulp valve performs satisfactorily. The measured data and the CCF for the consistency control loop are shown in Figure 10.7. The CCF here is clearly odd. Note that the data does not have the "ideal" stiction behaviour with the typical rectangular and triangular shapes.

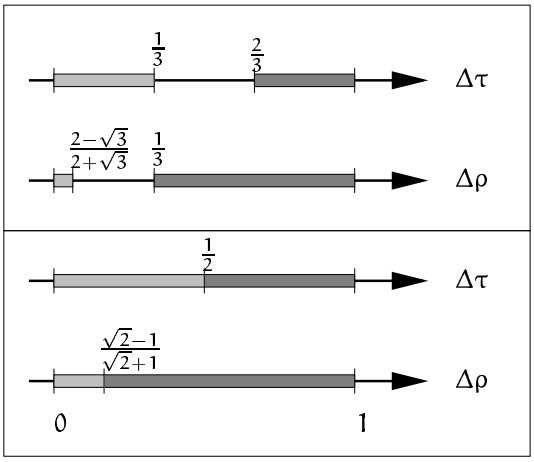


Figure 10.5: The decision rules *with* (top) and *without* dead-zone (bottom). The limits for non-stiction are marked in light-grey, the ones for stiction are marked in dark-grey. The dead-zone is the interval between the light- and the dark-grey bars.

Now, consider the flow control loop in Figure 10.6. Since the consistency loop oscillates, the flow control loop does so too. Thus there is a situation of an oscillating load disturbance. Data and CCF patterns are shown in Figure 10.8. As can be seen, the CCF is approximately even and the result is in agreement with the knowledge about the loop.

Loop interaction II Consider a two-by-two system similar to the one in Figure 10.6. It is known from experiments that both loops oscillate because the PI-controller in the consistency loop was too tightly tuned. The oscillations have been eliminated by re-tuning the controller. The oscillation in the flow control loop was also stopped by this. Hence, the correlation functions for both loops are expected to be even functions. Figures 10.9 and 10.10 show that this is actually the case. Note that the CCFs at lag zero are positive and negative respectively. This is because the consistency loop has a negative gain.

In this example we obviously have a situation where the root cause of

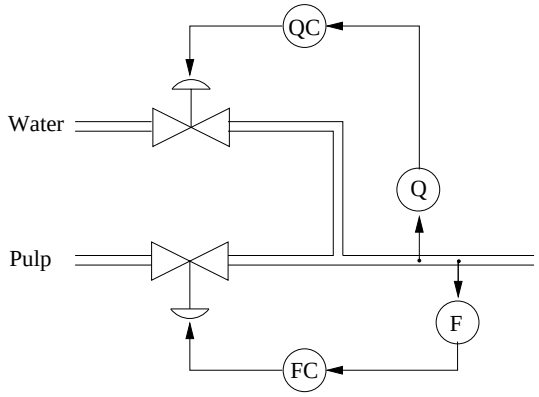


Figure 10.6: Two coupled loops from a consistency control loop.

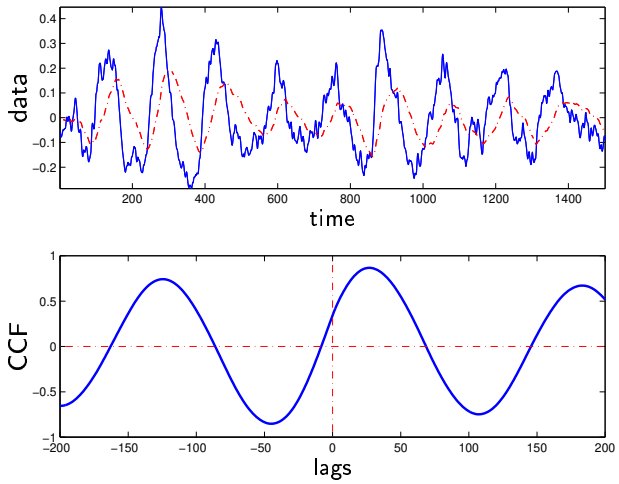


Figure 10.7: Consistency loop (Loop interaction I).

the problem cannot be diagnosed using the CCF. Another way of attacking this problem – based on some process knowledge – will be described in Chapter 12.

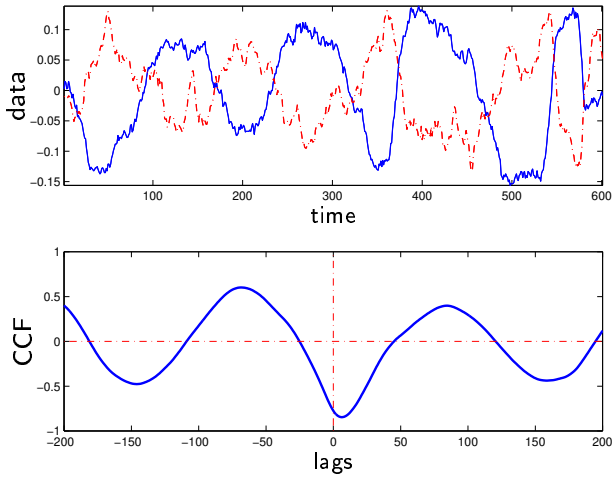


Figure 10.8: Flow loop (Loop interaction I).

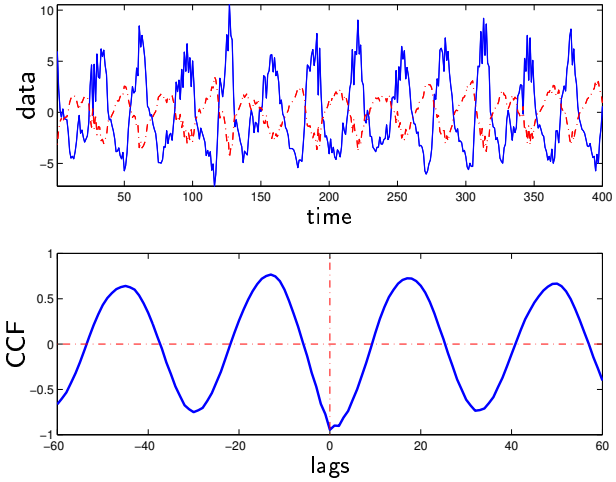


Figure 10.9: Flow loop (Loop interaction II).

Flow control loop I Another example for an oscillating flow control loop is considered here. From experiments it is known that interaction with other control loops is not very likely. Also the controller re-tuning did not remove

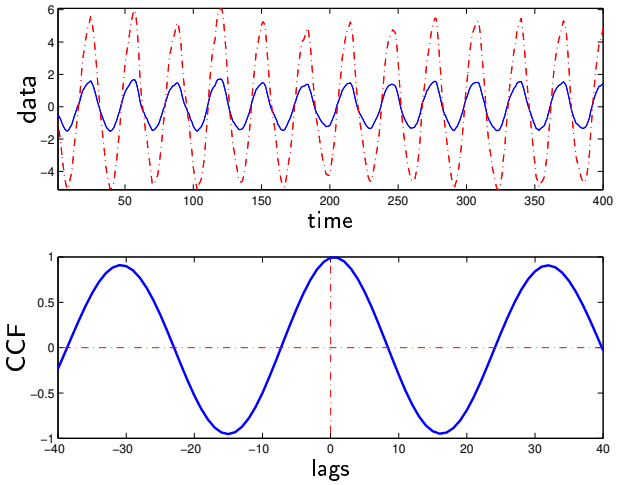


Figure 10.10: Consistency loop (Loop interaction II).

the oscillation. One likely explanation is a sticking control valve. There are data sets available for this loop that were collected daily over a period of several months. The oscillation is mostly present but it changes shape dramatically on different days. The correlation result for a case that is not easily diagnosed by visual inspection is shown in Figure 10.11.

This can be compared with another data set from the same control loop shown in Figure 10.12. It can be seen that the CCFs are odd functions in both cases. Even though the signal shapes are different, there is almost no difference in the CCF pattern. Hence, one may (correctly) conclude that the loop exhibits stiction behaviour.

Flow control loop II For this example it is known that the loop in question has problems with valve stiction. Apart from that, the flow measurement is very noisy. If one assumes that the oscillation has been detected, the CCF-method will easily diagnose stiction. The results from correlation analysis are shown in Figure 10.13a. Note that the measured process output does not have a pronounced rectangular shape as stiction often has. However, it does if one filters the data, but this is not necessary when using the CCF. It will yield the same results in either case. Note also that the length of

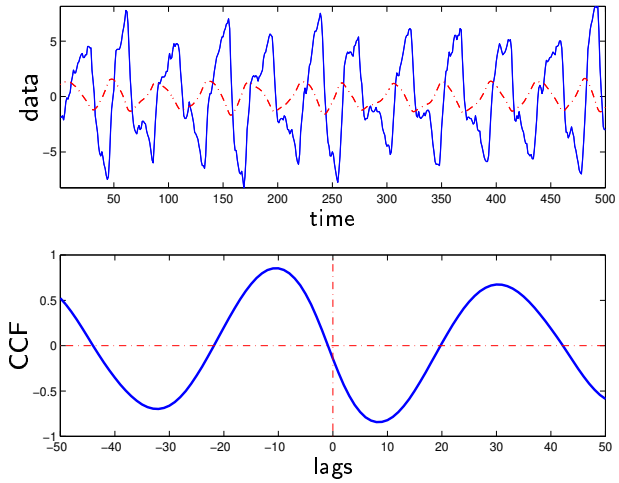


Figure 10.11: Flow loop I (a)

.

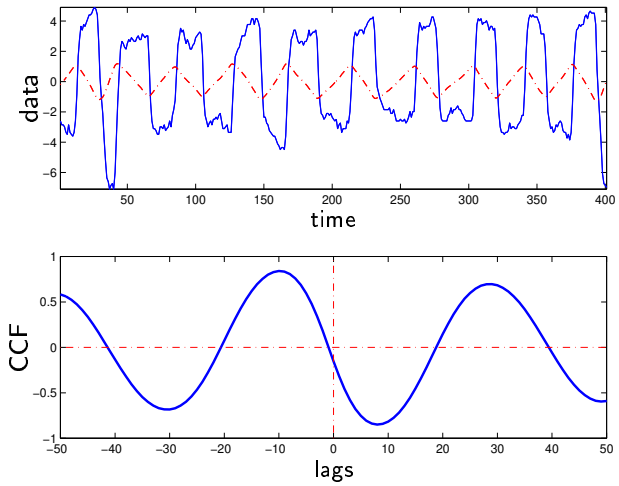


Figure 10.12: Flow loop I (b).

the data batch can be very short. As an illustration consider Figure 10.13b where very little data from the same loop was used. The interesting result is

that the CCFs are very similar (at least concerning the asymmetry w.r.t the vertical axis), no matter how much data one uses.

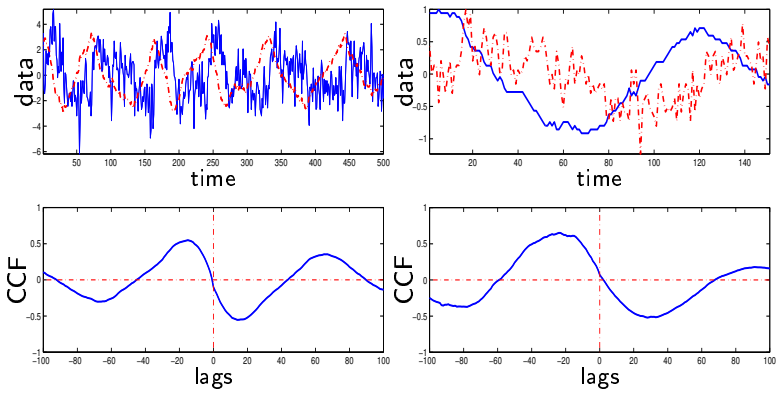


Figure 10.13a: Long data set. Figure 10.13b: Short data set.

Figure 10.13: Flow loop II. Cross-correlation for long and short data set.

Level control Consider now an example of a level control loop which is known to oscillate due to stiction. As can be seen in Figure 10.14, the CCF is even which would result in a deceiving diagnosis. The problem of diagnosing stiction in control loops with integrating processes will be discussed in Chapter 11. There, a new method – based on the detection of abrupt changes in the process variable – will be proposed in order to cover the case of integrating plants.

10.3 Theoretical explanation

The aim of this section is to motivate the diagnosis procedure described earlier in this chapter. Consider a control loop which oscillates with constant amplitude and frequency, i.e. all transients have vanished. Assume that the reference value is constant, see Figure 10.15. Then, the relationship between controller output $u(t)$ and process output $y(t)$ is completely described by

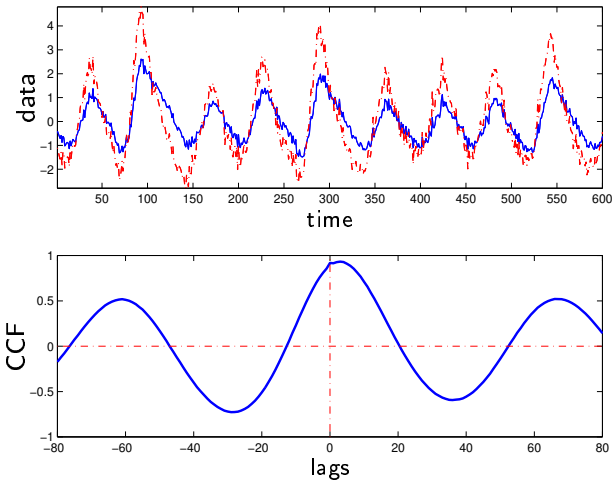


Figure 10.14: Level control loop.

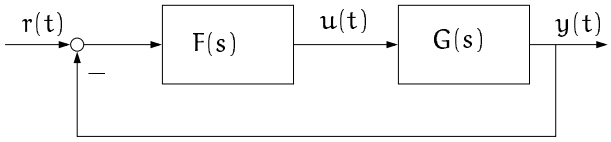


Figure 10.15: A general control loop.

the controller $F(s)$,

$$U(s) = -F(s)Y(s).$$

where $U(s)$ and $Y(s)$ are the Laplace transform of process output $y(t)$ and control signal $u(t)$ respectively. This relation will be used frequently in the following where the theoretical CCF for different possible root causes of oscillations are discussed.

Correlation for oscillating external disturbances

Assume that the process in question is controlled by a PI-controller, i.e.

$$F(s) = K_c \left(1 + \frac{1}{T_i s}\right).$$

Since one is interested in the phase-shift between $u(t)$ and $y(t)$, consider the Bode plot of $-F(s)$. The phase curve of $-F(s)$ is obtained as

$$\Delta\phi = \arg\{-F(i\omega)\} = -\frac{3}{2}\pi + \arctan(T_i\omega). \tag{10.5}$$

The Bode plot for $-F(s)$ is shown in Figure 10.16 where the break frequency is $\omega_c = 1/T_i$. In general low-frequency disturbances will be eliminated

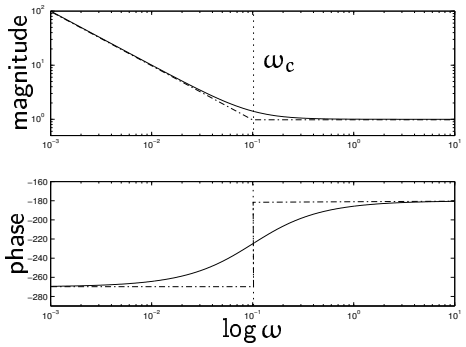


Figure 10.16: Bode diagram for $-F(s)$. Exact (solid) and asymptotic (dash-dotted).

efficiently by the PI-controller since a controller with integral action yields high loop gain at low frequencies.

On the other hand, if a medium or high frequency oscillating disturbance enters the loop, it may not be attenuated sufficiently so that an oscillation will be detected. However, in that case the phase-shift will typically be approximately π and thus be classified correctly by the algorithm[1]. This is because the second term of (10.5) approaches $\pi/2$ for high frequencies.

[1] If the phase shift if *exactly* $-\pi$, the deviation of the CCF from the ideal position, see Figure 10.2, will be zero.

It is therefore important that only signals which are clearly oscillating are used. Therefore, one of the detection methods described earlier must have detected an oscillation.

Tight tuning

Oscillation may also be caused by tight controller tuning in connection with some nonlinearity (except for stiction), typically backlash or dead-zone. In such a case, the CCF will typically be even.

The following reasoning can be used: When a loop oscillates due to any of the reasons mentioned above, it will do it at (or close to) its critical frequency ω_u, i.e. when the phase-shift between control error and process output is (approximately) $-\pi$. Hence the CCF will be approximately even. Therefore, as mentioned before, the CCF-method cannot be used to distinguish between external load disturbances and oscillation caused by tight control in connection with some nonlinearity.

Correlation in the presence of stiction

For ideal stiction one can calculate the CCF analytically. Stiction implies that the valve piston periodically pops from one position to another. This, together with the assumption of a PI-controller results in the typical stiction pattern (see Figure 10.12), i.e. a square wave (process output) and a triangular wave (control signal).

Expressions for the CCF in the stiction case will now be derived. Then, let each of the signals $y(t)$ and $u(t)$ be expanded in a Fourier series

$$f(t) = \frac{a_0}{2} + \sum_{k=1}^{\infty} \left(a_k \cos[kt] + b_k \sin[kt] \right). \tag{10.6}$$

Assume that the signals considered here are described such that they are asymmetric with respect to the origin[2], see Figure 10.17. This implied that $a_i = 0, \forall\, i$ in (10.6). The key point of the new method is to look whether the CCF is odd or even. This is the same as determining whether the phase shift between both signals is $-\pi$ or $-\pi/2$ respectively. Let the signals hence

[2]Here it is assumed – without loss of generality – that the signals are periodic with period 2π.

be described by their Fourier series expansion as

$$u(t+\tau) = \sum_{k=1}^{\infty} b_k \sin[k(t+\tau)] \tag{10.7}$$

$$y(t+\Delta\phi) = \sum_{l=1}^{\infty} c_k \sin[l(t+\Delta\phi)], \tag{10.8}$$

where τ is the lag of the CCF. Inserting these in the continuous-time definition of the cross-correlation

$$r_{uy}(\tau) = \lim_{T\to\infty} \frac{1}{2T} \int_{-T}^{T} u(t+\tau)y(t)\,dt \tag{10.9}$$

gives

$$r_{uy}(\tau,\Delta\phi) = \lim_{T\to\infty} \frac{1}{2T} \sum_{k=1}^{\infty}\sum_{l=1}^{\infty} b_k c_l \int_{-T}^{T} \sin[k(t+\tau)]\sin[l(t+\Delta\phi)]\,dt.$$

After some tedious but straight forward calculations this is evaluated to be

$$r_{uy}(\tau,\Delta\phi) = \frac{1}{2}\sum_{k=1}^{\infty} b_k c_k \cos[k(\tau-\Delta\phi)] \tag{10.10}$$

where

$$\begin{aligned} b_k &= \frac{1}{\pi}\int_{-\pi}^{\pi} u(t)\sin[kt]\,dt \\ c_k &= \frac{1}{\pi}\int_{-\pi}^{\pi} y(t)\sin[kt]\,dt \end{aligned} \tag{10.11}$$

There $u(t)$ and $y(t)$ are as shown in Figure 10.17. For the case of a rectangular signal $y(t)$,

$$b_k = \begin{cases} \frac{4C}{k\pi} & \text{if } k = 1,3,5,\ldots \\ 0 & \text{if } k = 2,4,6,\ldots \end{cases}. \tag{10.12}$$

A triangular signal $u(t)$ results in

$$c_k = \begin{cases} \frac{4m}{k^2\pi}(-1)^{\frac{(k+3)}{2}} & \text{if } k = 1,3,5,\ldots \\ 0 & \text{if } k = 2,4,6,\ldots \end{cases}. \tag{10.13}$$

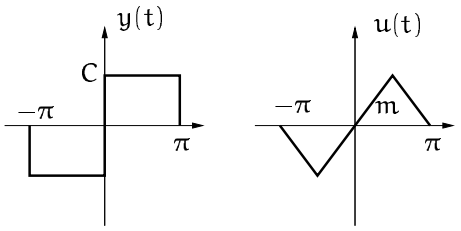


Figure 10.17: Ideal control signal $u(t)$ (right) and process output $y(t)$ (left) signal in the presence of stiction.

Inserting (10.12) and (10.13) into (10.10) yields

$$r_{uy}(\tau, \Delta\phi) = \frac{8}{\pi^2} \sum_{i=1}^{\infty} \frac{\cos[(2i-1)(\tau - \Delta\phi)]}{(2i-1)^3} (-1)^{i+1}. \qquad (10.14)$$

Figure 10.18 shows[3] $r_{uy}(\tau, -\frac{\pi}{2})$ in (10.14) using only the first element and after having used the first 50 elements of the sum in (10.14). It can be

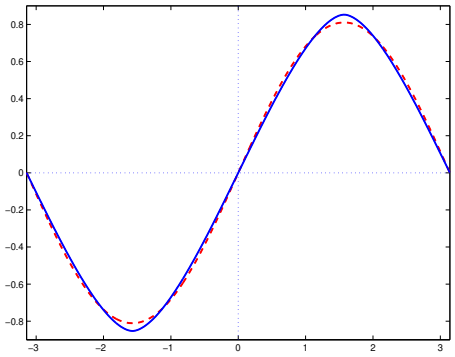


Figure 10.18: Approximation of $r_{uy}(\tau, -\frac{\pi}{2})$. Truncation after first term (dashed) and after 50 terms (solid).

seen that it is sufficient to truncate the infinite sum after the first element.

[3]Note that $\Delta\phi = -\pi/2$ corresponds to the ideal stiction signals as shown in the left part of Figure 10.1.

Truncating this series after the first element and with $\Delta\phi = -\pi/2$ gives

$$r_{uy}\left(\tau, -\frac{\pi}{2}\right) \approx \frac{8}{\pi^2}\sin(\tau). \tag{10.15}$$

Now it can easily be seen that the CCF for $u(t)$ and $y(t)$ for the case of stiction is in good agreement with the empirical results described above.

Chapter 11

Detection of static friction for integrating plants

As already pointed out, a weakness of the CCF-method described in Chapter 10 is that it cannot be used for integrating processes. The reason being the fact that the integrator in the process changes the correlation structure differently for the stiction and the non-stiction case. An ad hoc remedy would be to try re-calculate the valve position signal, i.e. to differentiate the process output. Unfortunately, this does not work as is motivated in Appendix D.

A new method will therefore be introduced in Section 11.1. The main idea is to check whether the process output exhibits regular abrupt changes or not. These abrupt changes will be detected by considering the second (filtered) derivative of the process output $y(t)$. If $y(t)$ is triangular (perfectly and without noise), the second derivative will be a pulse train only. Then, neglecting the pulse train and assuming additional Gaussian noise the second derivative will have a probability distribution which is normal. On the other hand, in the non-stiction case, the (oscillating) process output and its second derivative are usually relatively sinusoidal. Hence the probability distribution has two separate maxima. The proposed detection algorithm tests which of the two distributions is more likely to fit the data. Examples using industrial data are presented in Section 11.2. Since one has to be careful when differentiating noisy signals, the signals will be filtered before. The filter constant can be chosen dependent on the oscillation frequency.

As a matter of fact, the idea of considering the distribution of the second

derivative of the process output can also be used for detection of stiction in self-regulating processes (for which the CCF-method usually works well). The only difference is that one has to consider the first (instead of the second) derivative of the process output since there is no integration in the process.

11.1 A new detection method

In this section, a simple method is described which detects abrupt changes in the process output. The main idea is to fit two different distribution functions (one corresponding to the stiction case and one to the non-stiction case) to the sample histogram of the second derivative of the measured process output. For ideal signals and pure differentiation (without filtering) the second derivatives (of the process output), and their sample histograms, are shown in Figure 11.1. The decision procedure can now be summarised as follows.

> *Consider the second (filtered) derivative of the process output. Check whether the probability density function fits better to the top or the bottom distribution in Figure 11.1. A better fit to the first distribution will indicate stiction, a better fit for the second one non-stiction.*

As a motivation consider Figure 11.2 where ideal signals for the stiction and the non-stiction case are shown. It can be seen that if the process output is a sinusoid, the second derivative is a (negative) sinusoid as well. For the ideal stiction case, the second derivative is a pulse train with alternating sign.

In the following, first the filtering and the differentiation of the process output will be discussed. Then, it will be described how to obtain the sample histogram of the second derivative. Finally, in order to be able to fit distribution functions to the sample histograms, the theoretical, expected distributions have to be derived.

Differentiation and Filtering

It is well known that differentiation of noisy signals is a trade-off between accuracy and noise amplification. The choice of the filter bandwidth can

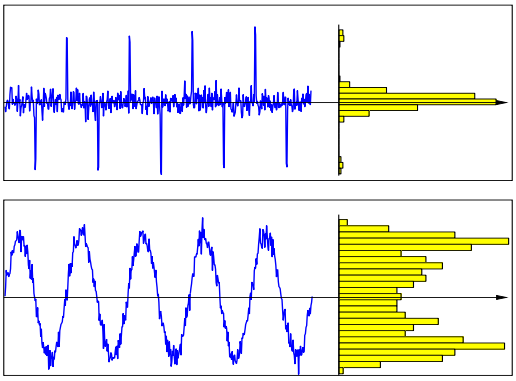


Figure 11.1: Ideal second derivatives of the process output and their sample histograms for the stiction case (top) and the non-stiction case (bottom).

be done using information of the oscillation frequency which can easily be obtained, for example as a by-product from the oscillation detection. The filter constant has to be chosen such that the noise is filtered out without affecting the shape of the oscillation too much. In this work, a first-order discrete-time low-pass filter,

$$H_1(q^{-1}) = \frac{1-\alpha}{1-\alpha q^{-1}}, \tag{11.1}$$

was used before each differentiation. The cut-off frequency of the filter is chosen as $\omega_c = 3 * \omega_{osc}$, where ω_{osc} is the oscillation frequency. Note that the choice of the filter cut-off frequency is very important. If it is too small, the form of the oscillating signal will become too smooth (yielding almost surely a Gaussian distribution). If it is too large the noise will be amplified too much by the differentiation. The filter pole in (11.1) is determined as

$$\alpha = e^{-\omega_c T_s}, \tag{11.2}$$

where T_s is the sampling interval. The whole filtering and differentiation procedure can be summarised in the filter

$$H_2(q^{-1}) = \left(\frac{(1-\alpha)(1-q^{-1})}{(1-\alpha q^{-1})} \right)^2, \tag{11.3}$$

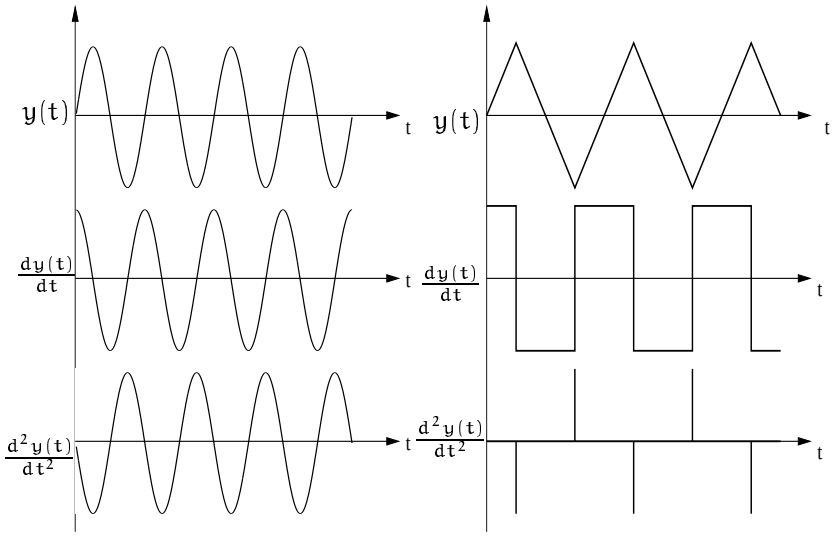


Figure 11.2: Motivation of the proposed detection mechanism. The process output and its first and second derivatives are shown for the stiction case (right) and the non-stiction case (left).

yielding the second (filtered) derivative of the process output as

$$y_{df}(t) = H_2(q^{-1})\, y(t). \tag{11.4}$$

The sample histogram

The sample histogram of the second derivative of the process output can easily be obtained. The only choice one has to make is the number of classes. In the statistical literature guidelines for this choice are given when one wants to perform a goodness-of-fit test. A suitable number of classes K for data batches up to $N = 2000$ samples has been proposed by Williams (1950). For batches with more samples, Bendat and Piersol (1967) propose to use $K = 1.87\,(N-1)^{0.4}$. The combination of both suggestions is shown in Figure 11.3[1]. The values in Figure 11.3 have been found to fit well for

[1] These rules were developed for a χ^2 Goodness-of-fit test with a 5% level of significance.

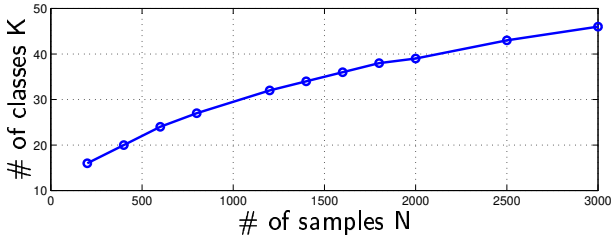


Figure 11.3: Recommended number of histogram classes K for use in goodness-of-fit test.

the purpose of this chapter.

Of course, the raw histogram has to be normalised for comparison with theoretical probability density functions. This is achieved as follows. Let N_i, N and Δx denote the number of elements in class i, the number of samples of data and the width of each class respectively. Then the probability $p_i(x)$ that a data sample is an element of class i is obtained as

$$p_i(x) = \lim_{\substack{\Delta x \to 0 \\ N \to \infty}} \frac{N_i}{\Delta x N}. \tag{11.5}$$

In practice one cannot reach the limit and once N is determined, one has to divide N_i by the number of samples N and the class width Δx in order to obtain an estimate of $p_i(x)$.

A note has to be made here about the length of the data batch. Assuming a reasonable sampling frequency (i.e. fast enough to cover the oscillation well), one has to make sure that the data contains only a small amount of oscillation periods. This is especially important if the signal shape changes a lot over time. It has been found that a value of 5-10 oscillation periods is a reasonable value.

Distribution for the stiction case

As can be seen from Figure 11.1, for ideal stiction the sample histogram of the second derivative of the process output will be purely white noise with a pulse train (assuming Gaussian measurement noise). The distribution will thus mainly be Gaussian if we assume that the number of oscillation

periods is small compared to the number of samples[2]. The ideal probability distribution for the stiction case is hence normal, i.e.

$$f_G(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp \frac{-(x-\mu)^2}{2\sigma^2}. \tag{11.6}$$

This distribution can easily be fitted to the sample histogram since the data mean μ and standard deviation σ can be computed directly from the data.

However, for real data, the peaks in the second derivative will not be infinitely narrow. The sample histogram will therefore have significant tails on both ends which will typically increase the variance. The same effect has the – already mentioned – filtering which is necessary prior to differentiation. The choice of the filter bandwidth will of course influence the resulting distribution.

Therefore, the distribution (11.6) will now be modified. One alternative being the computation of the theoretical probability distribution when filtering and differentiating a triangular signal with additional white noise. This may be possible but will lead to very complicated expressions. Therefore, another approach is chosen here. Consider a noise-free triangular signal and its second (filtered) derivative. Let the filter pole be chosen from (11.2). Figure 11.4 shows the (noise-free) data and the resulting sample histogram. It can clearly be seen that the peaks in the second derivative have now a significant contribution to the sample histogram (compare with Figure 11.1!). A simple way of approximating the histogram in Figure 11.4 would be to combine a Gaussian distribution with a uniform distribution, i.e.

$$f_E(x) = (1-\varepsilon) f_G(x) + \varepsilon f_u(x), \tag{11.7}$$

where

$$f_u(x) = \begin{cases} \frac{1}{2A} & |x| \le A \\ 0 & |x| > A \end{cases},$$

where A is the amplitude of $d^2y(t)/dt^2$. Note that the integral of (11.7) is

$$\int_{-\infty}^{\infty} f_E(x)\, dx = 1$$

as required. It was found that this modification is sufficient for real data. Finally, see Figure 11.5 for a plot of $f_E(x)$ as a function of x. Of course,

[2]This is an implicit assumption that the data has to be sampled sufficiently fast such that the oscillation has an reasonable (e.g. > 30) amount of samples per period.

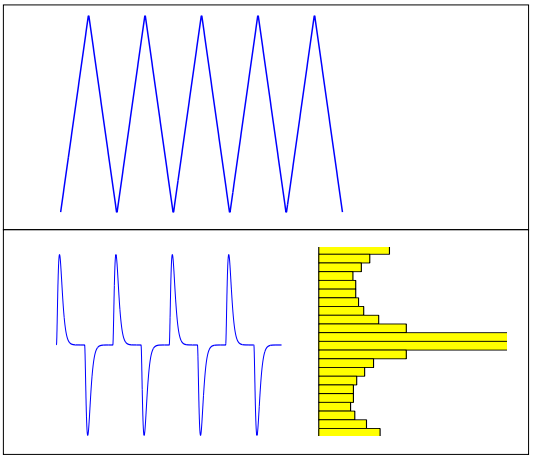


Figure 11.4: Noise-free (ideal) signals in the stiction case. Process output (top) and second (filtered) derivative with its sample histogram (bottom).

one has to choose the parameter ε. It could be estimated together with the variance of the distribution. However, as will shown later, for the non-stiction case, there is only one parameter to be estimated. We therefore chose to fix the parameter ε to a pre-defined value of 0.3. This choice has been found by extensive tests on both real and simulated data. Note that $\varepsilon = 0$ corresponds to a pure Gaussian distribution and $\varepsilon = 1$ to a pure uniform distribution. In no case it was found that the diagnose was changed for reasonable variation of ε. Even a choice of $\varepsilon = 0$ has been tested on data without problems. However, using the proposed choice will make the decision more unique. Although modified, the distribution (11.7) will be referred to as the *Gaussian distribution* in the following.

The variance σ^2 in (11.6) will hence be estimated using a nonlinear least-squares algorithm. If it is assumed that de-trended data is used such that we have $\mu = 0$. The estimation of the standard deviation may improve the fit compared to using the variance as calculated directly from data. In order to avoid local minima, the standard deviation obtained directly from data is used as an initial guess. The estimation will be done in the same way as for the non-stiction case described below.

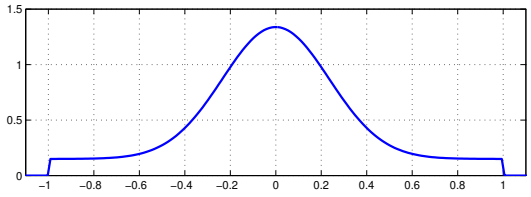


Figure 11.5: Combined Gaussian and uniform distribution $f_E(x)$ from Equation (11.7).

Distribution for the non-stiction case

Consider the bottom plot of Figure 11.1. As motivated earlier, in the non-stiction case, the signal in question is assumed to be a sinusoid with additional white noise. It is assumed that the noise is independent of the sinusoid. In order to justify to describe a sinusoid in probabilistic terms, the initial phase can be thought of as being a random variable. The problem is then to find the probability function for the signal $z(t) = x(t)+e(t) = A\sin[\omega t+y(t)]+e(t)$ where $e(t)$ is zero mean white noise and $y(t)$ is uniformly distributed.

The resulting probability density function when considering the sum of two stochastic variables $x(t)$ and $e(t)$ is given by the convolution of their individual density functions, i.e.

$$f_Z(z) = \int_{-\infty}^{\infty} f_X(x) f_G(z-x)\,dx. \tag{11.8}$$

The density function for a sinusoid with random initial phase can be found using the following result from the statistical literature, see for example (Hines and Montgomery, 1990).

Theorem 11.1. *If Y is a continuous random variable with probability density function $f_Y(y)$ that satisfies $f_Y(y) > 0$ for $a < y < b$, and $x = H(y)$ is a continuous strictly increasing or strictly decreasing function of y, then the random variable $X = H(Y)$ has the density function*

$$f_X(x) = f_Y(H^{-1}(x)) \left| \frac{d}{dx} H^{-1}(x) \right|$$

□

For the case of a sinusoid, $x(t) = A\sin[\omega t + y(t)]$, we have $H^{-1}(x) = -\omega t + \arcsin\frac{x(t)}{A}$ and

$$f_Y(H^{-1}(x)) = \begin{cases} 1/\pi & \text{for } H(-\frac{\pi}{2}) < x < H(\frac{\pi}{2}) \\ 0 & \text{else} \end{cases},$$

and for $|x| < A$,

$$\frac{d}{dx}H^{-1}(x) = \frac{1}{\sqrt{A^2 - x^2}}.$$

Then, it follows that

$$f_X(x) = \begin{cases} \left(\pi\sqrt{A^2 - x^2}\right)^{-1} & \text{for } |x| < A \\ 0 & \text{else} \end{cases}, \tag{11.9}$$

where $\int_{-\infty}^{\infty} f_X(x)\,dx = 1$ is satisfied. Figure 11.6 shows a plot of (11.9) as a function of x. Then, with $f_G(x)$ from (11.6) the convolution integral (11.8)

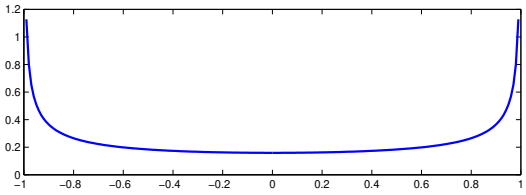


Figure 11.6: Probability density function $f_X(x)$ of a pure sinusoid with stochastic initial phase ($A = 1$).

becomes

$$f_Z(z) = \frac{1}{\sigma\sqrt{2\pi^3}}\int_{-A}^{A}\frac{\exp\left(-\frac{(z-x-\mu)^2}{2\sigma^2}\right)}{\sqrt{A^2 - x^2}}\,dx. \tag{11.10}$$

Figure 11.7 shows a plot of the distribution $f_Z(z)$. The distribution (11.10) exhibits a certain amount of *camelicity*[3]. It will therefore be referred to as

[3]This term was coined by Dr. Krister Forsman from ABB Automation Systems AB, Sweden.

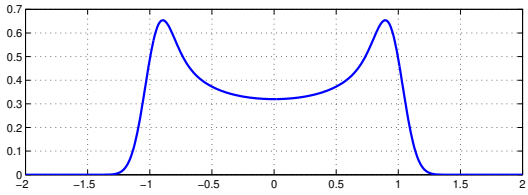


Figure 11.7: Probability density function of a sinusoid ($A = 1$) with additional zero-mean white noise ($\sigma = 0.1$).

the *"Camel distribution"* in the sequel. Unfortunately, there is no analytic solution to (11.10) and it has therefore to be evaluated numerically. There are three unknowns in (11.10), two of which can easily be determined. Since the data used is assumed to be de-trended we have $\mu = 0$. The amplitude A can easily be estimated from data. Then the parameter σ is used to fit the distribution (11.10) to the sample histogram:

$$\hat{\sigma} = \arg\min_{\sigma} \sum [f_Z(z, \sigma) - Y]^2 \tag{11.11}$$

where Y is the data from the sample histogram. The optimisation can be done using a standard non-linear least-squares fit algorithm (i.e. `lsqcurvefit` in `MATLAB`). A disadvantage is that the integral (11.10) has to be evaluated numerically during the fitting procedure. This will, however, be very fast since the limits of the integral are finite.

Note also that one has to make sure that the fitting algorithm does not find a solution which is very similar to a normal distribution (i.e. that the two maxima in Figure 11.7 are merged into one). This may happen when the estimated value of σ is "too large" compared to A. The disappearance of the two maxima can be avoided by for example constraining the admissible values for σ to be significantly smaller than A when running the nonlinear least-squares algorithm (here we chose $\sigma < 0.4\,A$).

To summarise: With some additional white noise, one may have a situation as shown in Figures 11.8 and 11.9. There, the measured process outputs are shown in the top left plot, the (filtered) second derivative in the top right plot and the sample histogram with the best fit of Gaussian and Camel distributions in the bottom left and right plot respectively. The results from all examples will be shown in plots with the same layout.

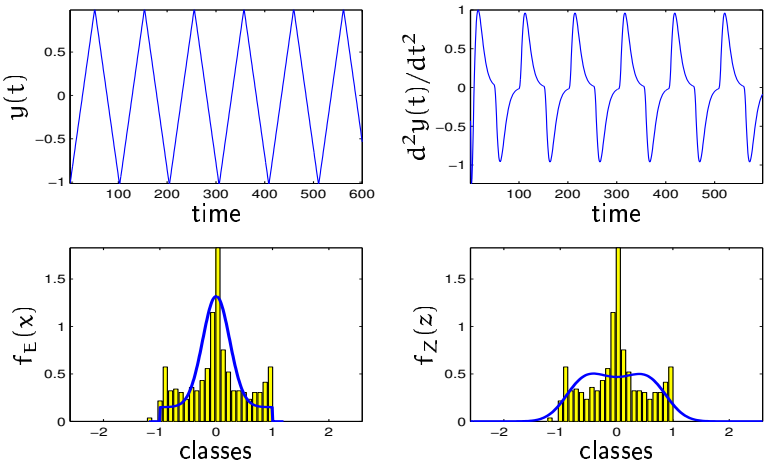


Figure 11.8: The stiction case, ideal data.

The algorithm for diagnosing stiction in integrating control loops is summarised in Figure 11.10 It is worth mentioning that for each distribution, one single parameter is estimated (standard deviation of the noise). It is therefore fair to compare the resulting MSEs directly. The MSE is calculated from

$$\text{MSE} = \sum \left[f(\hat{\sigma}) - Y \right]^2 . \tag{11.12}$$

where Y is the data from the sample histogram and $f(\hat{\sigma})$ is the distribution function for the case in question using the standard deviation which gives the best fit to the data. The MSE can be obtained as the value of the loss function from the numerical optimisation (11.11).

11.2 Examples

The method is now demonstrated on two examples.

A level control loop with stiction. Consider a level control loop which is known (from experiments) to oscillate due to stiction. When applying the decision algorithm to the process output, the results shown in Figure 11.11

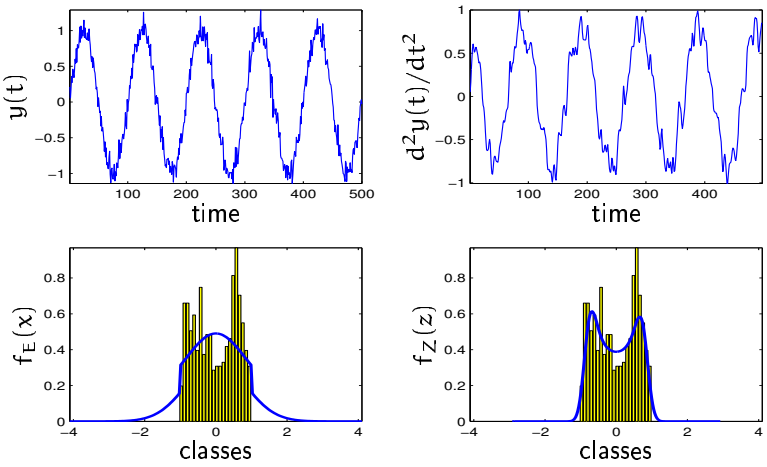


Figure 11.9: The non-stiction case, ideal data.

are obtained. The mean square error (MSE) for the Gaussian distribution is 0.97 compared to 2.01 for the Camel distribution. It can hence be (correctly) concluded that the oscillation is due to stiction.

A level control loop without stiction. Consider a level control loop which is known to oscillate for a reason other than stiction. The oscillation could in this case be removed by re-tuning the controller. The results when applying the diagnosis algorithm are shown in Figure 11.12. The fit for the Gaussian distribution is 1.17 compared to 0.46 for the Camel distribution. It can hence be (again correctly) concluded that the oscillation is *not* due to stiction.

A level control loop with dead-band. Consider another level control loop which is known to oscillate due to dead-band in the controller software. The oscillation could in this case be removed by re-tuning the controller. The results when applying the diagnosis algorithm are shown in Figure 11.13. The fit for the Gaussian distribution is 1.52 compared to 0.70 for the Camel distribution, therefore yielding a correct diagnosis, namely *non-stiction*.

1. Check if the loop is oscillating and determine the frequency of oscillation (Forsman and Stattin, 1999).
2. Compute the filtered second derivative $y_f(t)$ using (11.3).
3. Compute the sample histogram (11.5) of the second derivative.
4. Estimate the standard deviation in (11.7) in order to fit the sample histogram as well as possible.
5. Determine the (approximate) amplitude of the second filtered derivative $y_f(t)$.
6. Estimate the standard deviation in (11.10) in order to fit the sample histogram as well as possible.
7. If the fit for the Gaussian distribution is significantly better than the fit of the Camel distribution, conclude *stiction*. In the opposite case, conclude *no stiction*. If both fits are approximately equal (e.g. difference smaller than 10%), no decision is made.

Figure 11.10: Diagnosis algorithm.

11.3 Self-regulating processes

As mentioned in the introduction, oscillation in self-regulating processes due to stiction can very conveniently be detected using the CCF-method. However, the same idea as presented in the previous section can also be used on self-regulating processes. The only difference is that one makes use of the first derivative of the process output instead of the second. The method is now demonstrated on two examples.

A flow control loop with stiction. Consider a flow control loop which is known to oscillate due to stiction. The results of the diagnosis algorithm are shown in Figure 11.14. The mean square error for the Gaussian distribution is

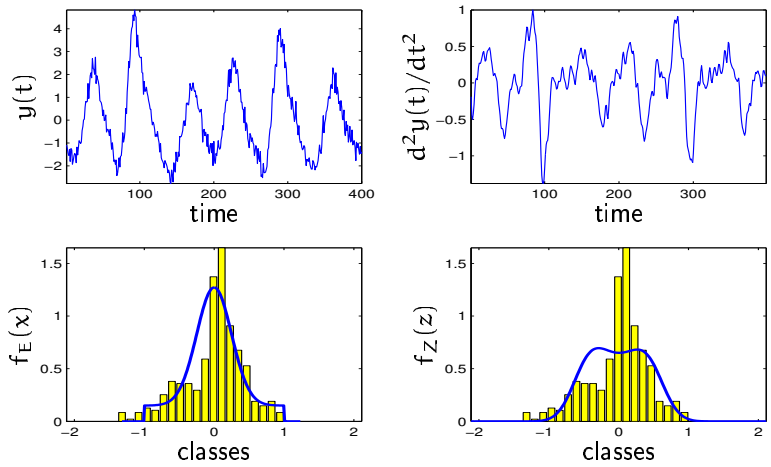


Figure 11.11: Level control loop *with* stiction.

0.80 compared to 1.36 for the Camel distribution. It can hence be concluded that the oscillation is due to stiction.

A flow control loop without stiction. Consider a flow control loop which is known to oscillate due to interaction with another loop. The results when applying the diagnosis algorithm are shown in Figure 11.15. The fit for the Gaussian distribution is 2.44 compared to 1.47 for the Camel distribution. It can hence be concluded that the oscillation is *not* due to stiction.

Loops with dominant P-control. There may be cases where the CCF method gives no (or possibly a wrong) indication. One of them is when the controller is basically proportional. Then the control signal has a very steep initial phase after each peak, see Figure 11.16. These potential problems were also discussed in (Forsman, 2000). In such a case the CCF is neither even nor odd. See Figure 11.17 for an illustration. See Figure 11.3 where the method described in this chapter is used on the above data. The MSEs are 0.77 and 1.87 for the Gaussian and the Camel distribution respectively. It can hence (correctly) be concluded that the loop oscillates due to stiction.

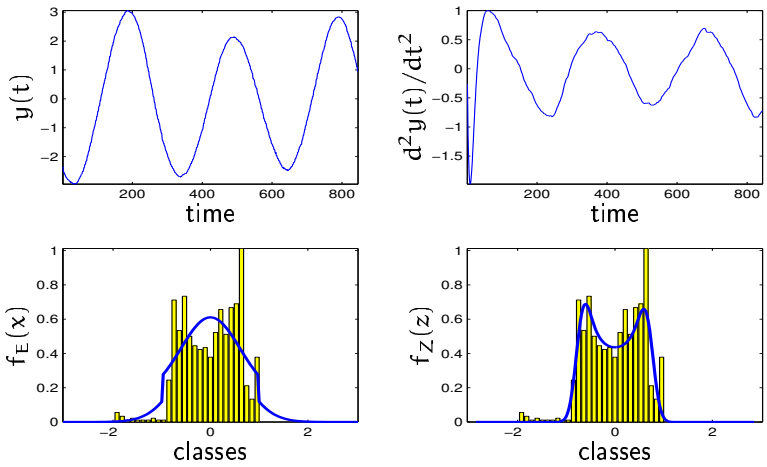


Figure 11.12: Level control loop *without* stiction.

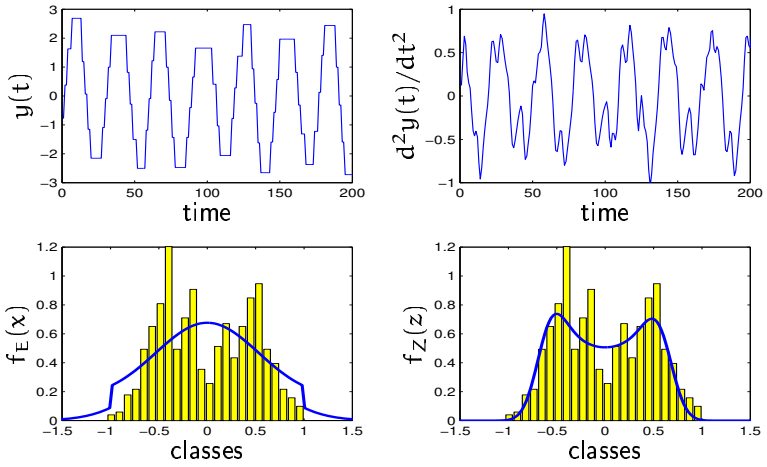


Figure 11.13: Level control loop with dead-band.

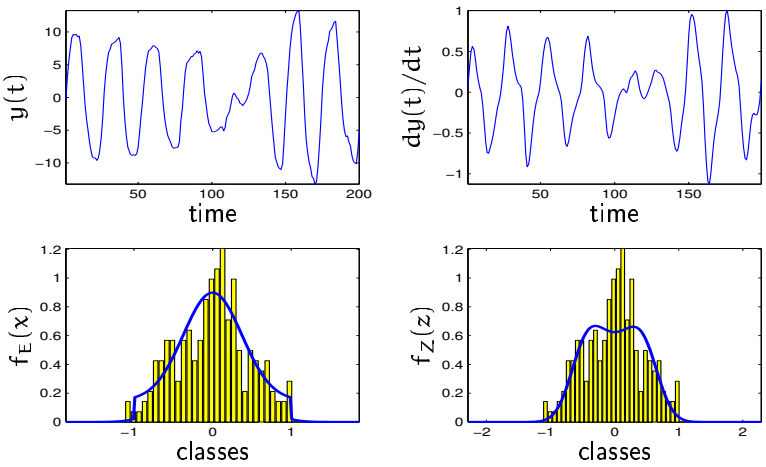


Figure 11.14: Flow control loop *with* stiction.

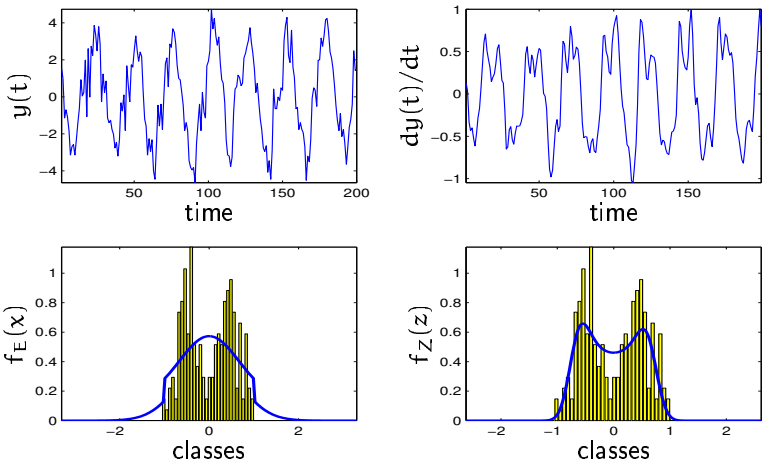


Figure 11.15: Flow control loop *without* stiction.

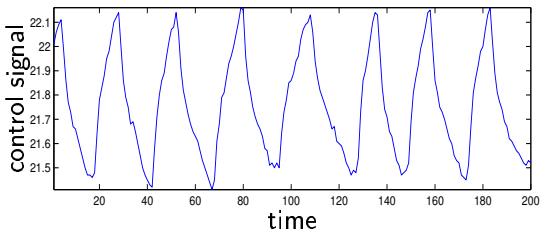


Figure 11.16: Typical control signal from a loop exhibiting stiction when the proportional part dominates the controller.

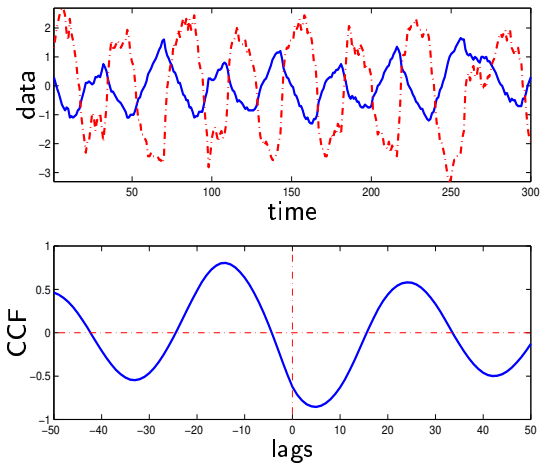


Figure 11.17: Example of a flow control loop with stiction. The CCF-method is used for detection. Top: control signal (solid) and process output (dash-dotted).

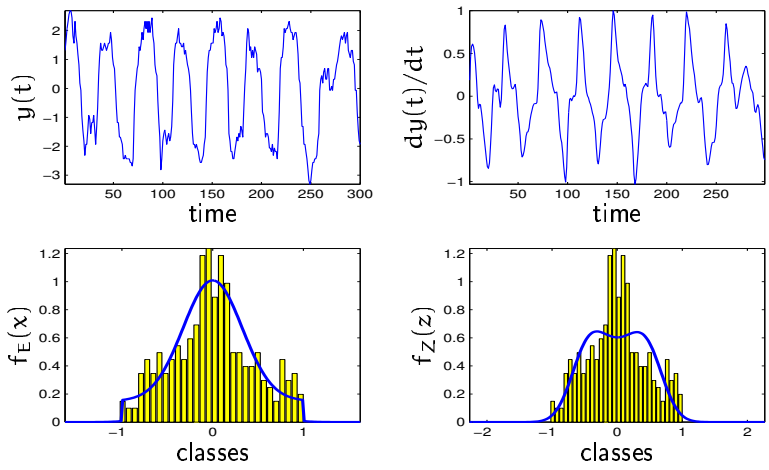


Figure 11.18: Example of a flow control loop with stiction and dominant P-control.

Chapter 12

Diagnosis using process knowledge

As a motivation re-consider a typical scenario in the process industry, see Figure 12.1. If both process outputs, y_1 and y_2, are oscillating (and none

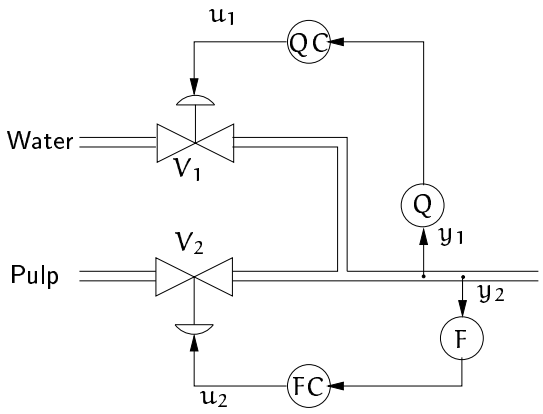


Figure 12.1: System of two coupled control loops. Which loop is likely to cause the oscillation?

due to stiction), it is of interest to have an indication which of the loops needs maintenance. In Chapter 9 we tried to argue that this may not be

possible based on the measured signals only. In this chapter two methods are proposed in order to solve the stated problem, assuming some process knowledge.

In Section 12.1 diagnosis is discussed assuming that the process dead-time is known. The idea is to make use of the Harris index after having the removed the oscillation in the time domain. Then in Section 12.2 the available process knowledge is assumed to be a simple process model. Using that model and an estimate of the controller, the ultimate frequency is calculated and compared to the actual oscillation frequency.

12.1 Process dead-time

This section deals with the diagnosis of oscillations when the process dead-time is assumed to be known. It can be known a priori or be estimated using the method proposed in Chapter 8. As a matter of fact, the Harris index – which requires knowledge of the dead-time – will be used in this section for the purpose of oscillation diagnosis.

Lin *et al.* (1998) and Owen (1997) proposed to remove the oscillation contribution in the variance expressions when computing the Harris performance index for oscillating signals. The motivation for such an approach appears to be the following: If a loop is badly tuned such that it oscillates, it is likely that the control performance is bad also at other frequencies than the oscillation frequency. Hence after removing the contribution of the oscillation to the performance index, one should still obtain a "high" index. On the other side, for a loop which is well tuned but oscillates due to interaction, the performance index should be small when the oscillation contribution is removed. In such a way one could tell apart two interacting oscillating loops.

In Chapter 5 the evaluation of the Harris index using oscillating signals was discussed. The conclusion there was that one should avoid to evaluate the Harris index for oscillating signals. In order to circumvent potential problems, an alternative method is proposed here: Remove the oscillation contribution *before* the performance index is calculated. This can be done by applying a notch filter to the process output. Then the Harris index can be evaluated on the filtered signal as usual.

Note that the case of stiction in valves needs special attention. When a control valve exhibits stiction, the loop may oscillate, no matter if the

PI-controller is well tuned or not. Hence, the evaluation of the Harris index will usually not give any interesting information. Therefore, prior to using the method proposed in this chapter, one should test the (oscillating) data for stiction as described in Chapters 10 and 11.

A motivation in the frequency domain

In Chapter 2, the Harris performance index was derived in the time-domain. It is of course also possible to evaluate the index in the frequency domain.

Start therefore with some definitions: The auto-covariance function of the (zero-mean) process output $y(t)$ is

$$r_y(\tau) = E[y(t)y(t+\tau)]. \tag{12.1}$$

Then the power spectrum of $y(t)$ is determined by

$$\Phi_y(\omega) = \sum_{\tau=-\infty}^{\infty} r_y(\tau) e^{-i\omega\tau}. \tag{12.2}$$

Using the inverse Fourier transform,

$$r_y(\tau) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \Phi_y(\omega) e^{-i\omega\tau}\, d\omega. \tag{12.3}$$

Thus, the variance of $y(t)$ can be expressed as

$$\sigma_y^2 = r_y(0) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \Phi_y(\omega)\, d\omega \tag{12.4}$$

and consequently the performance index can be calculated as

$$I_p = \frac{\int_{-\pi}^{\pi} \Phi_y(\omega)\, d\omega}{\int_{-\pi}^{\pi} \Phi_{mv}(\omega)\, d\omega}. \tag{12.5}$$

Here $\Phi_y(\omega)$ and $\Phi_{mv}(\omega)$ are the process output and minimum-variance spectra respectively, and are given by

$$\Phi_y(\omega) = \left|\frac{C(e^{i\omega})}{A(e^{i\omega})}\right|^2 \sigma_e^2,$$

where C/A is the time-series model from noise to process output, cf. (2.2) and

$$\Phi_{mv}(\omega) = \left|G_{mv}(e^{i\omega})\right|^2 \sigma_e^2,$$

where $G_{mv}(q^{-1}) = \left[h_0 + h_1 q^{-1} + \cdots + h_{d-1} q^{-(d-1)}\right]$, cf. (2.4). Note that the performance index (12.5) contains information for all frequencies. As an alternative to the Harris index which concentrates all information into a single number, one could compare the control performance in different frequency regions. Such an approach is difficult to handle automatically but offers of course more information to the user. A similar idea was also discussed by Huang and Shah (1999). As mentioned above, Lin *et al.* (1998) proposed the exclusion of a small interval around the oscillation frequency ω_{osc}. If such an interval is denoted $[\omega_1, \omega_2]$ where $\omega_1 < \omega_{osc} < \omega_2$, a modified index could be defined as

$$I_{mod} = \frac{\int_{-\pi}^{\pi} \Phi_y(\omega)\,d\omega - 2\int_{\omega_1}^{\omega_2} \Phi_y(\omega)\,d\omega}{\int_{-\pi}^{\pi} \Phi_{mv}(\omega)\,d\omega - 2\int_{\omega_1}^{\omega_2} \Phi_{mv}(\omega)\,d\omega}. \tag{12.6}$$

However, this approach still involves oscillating signals when fitting a time-series model. Regarding the results from Chapter 5, it is therefore proposed to remove the oscillation contribution *before* a time-series model is estimated. That means that the oscillation will be removed in the time-domain rather than in the frequency-domain. For the filtering a band-stop (Butterworth) filter was chosen. For the industrial data used in this work, a filter-order of four was found to be sufficient, i.e.

$$H_4(q^{-1}) = \frac{b_0 + b_1 q^{-1} + \ldots + b_4 q^{-4}}{1 + a_1 q^{-1} + \ldots + a_4 q^{-4}}, \tag{12.7}$$

and the filtered process output will be $y_f(t) = H_4(q^{-1})y(t)$.

The diagnosis procedure can now be summarised:

1. Exclude valve stiction as a likely cause.
2. Determine the oscillation frequency ω_{osc} (e.g. by-product from the oscillation detection).
3. Design the band-stop filter (12.7) with stop-band around the oscillation frequency.
4. Filter the process output.
5. Evaluate the Harris index as usual using the filtered data.

Examples

Numerical example: Filtering.

Consider a discrete-time ARMA model $y(t) = G(q^{-1})e(t)$ with

$$G(q^{-1}) = \frac{0.068\,q^{-1} - 0.26\,q^{-2} + 0.26\,q^{-3}}{1 - 2.51\,q^{-1} + 2.24\,q^{-2} - 0.67\,q^{-3}}.$$

The noise variance is $\sigma_e^2 = 2$ and the sampling rate is $T_s = 0.2$. The model has two complex poles on the unit circle and a realisation is shown in Figure 12.2. The effect of filtering can be seen in Figure 12.3. The

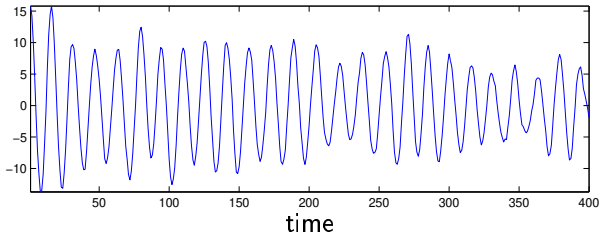


Figure 12.2: Realisation of the ARMA process.

Butterworth filter had a band-stop interval between 0.11 and 0.15 $\mathrm{rad/s}$ and the time-series model used was an auto-regressive model of order 25.

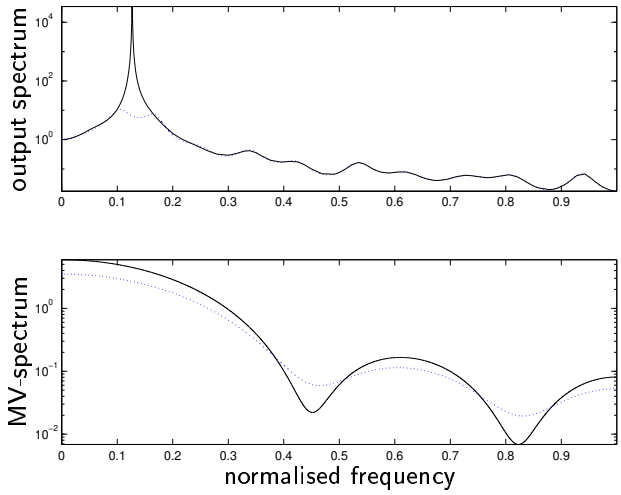


Figure 12.3: Output (top) and minimum-variance (bottom) spectrum before (solid) and after (dash-dotted) filtering.

Simulated example: coupled system

Consider the following multivariable system (cf. Figure 12.1)

$$G(s) = \begin{bmatrix} \frac{2e^{-4s}}{(3s+1)(10s+1)} & \frac{0.5e^{-3s}}{(5s+1)(10s+1)} \\ \frac{0.5e^{-4s}}{(5s+1)(10s+1)} & \frac{2e^{-3s}}{(3s+1)(10s+1)} \end{bmatrix}.$$

It is controlled using two PI-controllers with gains K_{c1}, K_{c2} and integral time $T_{i1} = T_{i2} = 15$. The system is simulated in discrete-time with a sampling time of $T_s = 0.5$.

Badly tuned Loop I. Let $K_{c1} = 2$, $K_{c2} = 0.6$ and add a dead-zone (width 0.5) in valve V_1 in Loop I. The resulting simulation (assuming white measurement noise in both loops) is shown in Figure 12.4. Application of the proposed algorithm, i.e. band-stop filtering and evaluation of the Harris index using the filtered data yields an index of 2.95 for output y_1 and 1.16 for output y_2. The conclusion would (correctly) be that Loop I causes the oscillation.

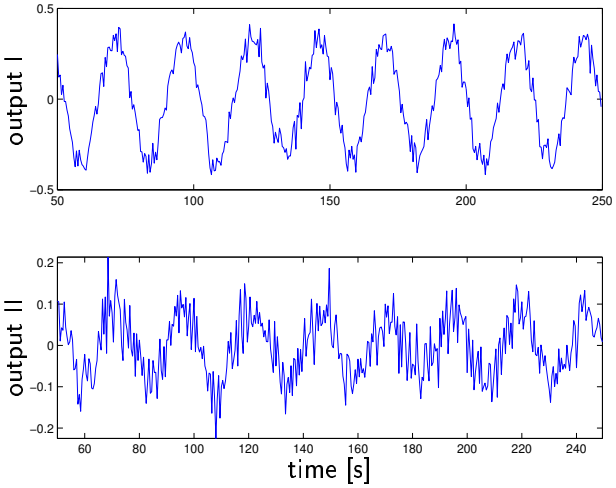


Figure 12.4: Coupled 2×2-system. Loop I (top) is causing the oscillation.

Badly tuned Loop II. Let $K_{c1} = 0.5$, $K_{c2} = 2$ and add a backlash element (dead-band 5) in valve V_2 in Loop II. The resulting simulation (assuming white measurement noise in both loops) is shown in Figure 12.5. Application of the proposed algorithm, yields an index of 1.67 for output y_1 and 5.87 for output y_2. The conclusion would (again correctly) be that Loop II causes the oscillation.

Industrial example: coupled system

Consider a similar case as in the previous subsection, i.e. a two-dimensional system with two decentralised PI-controllers. The process outputs are flow and consistency respectively and the same data was already used in Chapter 10.2 (Figures 10.9 and 10.10). Both process outputs are shown once again in Figure 12.6. The filtering approach described above is now used on that data yielding the filtered signals in Figure 12.7. The dead-times are 3 (flow) and 12 (consistency) seconds and the Harris index for the filtered signals are 1.67 (flow) and 3.50 (consistency). The (correct) conclusion is here that the consistency loop is likely to having caused both oscillations.

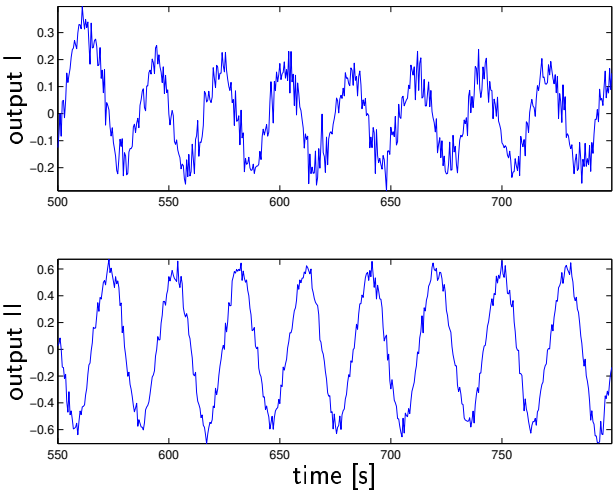


Figure 12.5: Coupled 2×2-system. Loop II (top) is causing the oscillation.

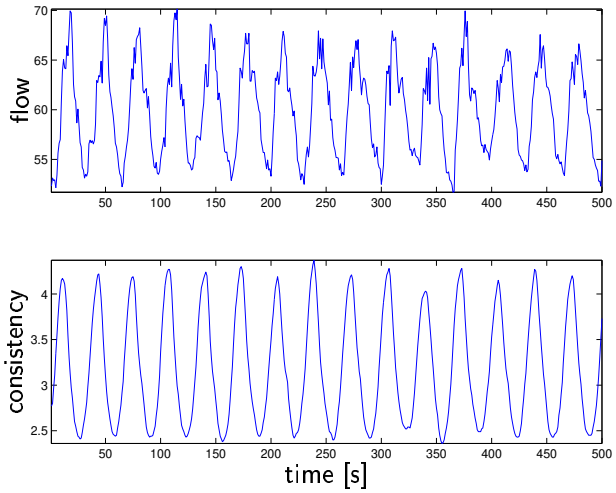


Figure 12.6: Measurements from an oscillating flow-consistency loop. The loop oscillates due to tight tuning in the consistency loop.

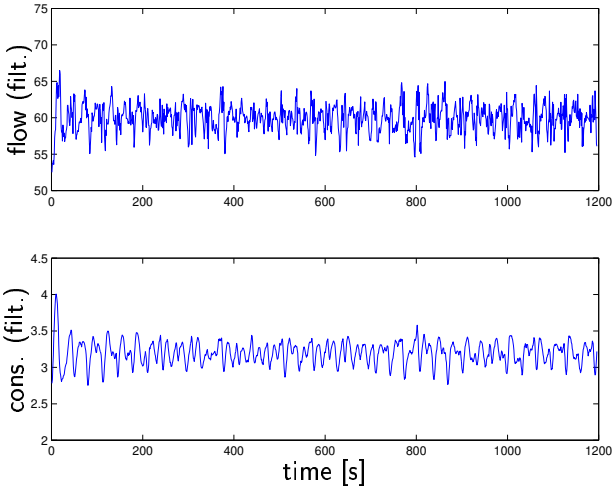


Figure 12.7: Filtered data from the oscillating flow-consistency loop.

12.2 Process model

In this section, oscillation diagnosis will be discussed based on the knowledge of a simple process model. Such a knowledge may be available from model-based controller tuning or by applying the automatic dead-time estimation procedure described in Chapter 8. The latter will provide the dead-time as well as the rest of the process dynamics.

The problem to solve is very similar to the one discussed in the previous section. Given oscillating data, it is of interest to know whether the oscillation is generated internally or externally. Recall that one can estimate the controller transfer function easily using any data set from the loop in question, see Chapter 6. Having process and controller transfer function, it is possible to estimate the ultimate frequency of the loop. By comparing the actual oscillation frequency to the ultimate frequency, one can separate external from internal oscillations. The actual oscillation frequency can be estimated as a by-product using one of the methods discussed in Chapter 9.1.

The diagnosis algorithm

The process model as estimated from a dead-time estimation procedure or available from some model-based control procedure may be given as

$$G_1(s) = \frac{K_p}{\tau s + 1} e^{-\theta s}, \quad G_2(s) = \frac{K_p}{s} e^{-\theta s} \tag{12.8}$$

for self-regulating and integrating processes respectively. Both transfer functions can well describe a large amount of basic control loops in the process industry.

As discussed in Chapter 6, one can always easily identify the controller transfer function $F(s)$, no matter if the loop is oscillating or not. Using both the controller and the process transfer function ($F(s)$ and G(s) respectively), one can calculate the ultimate frequency ω_u, i.e. the frequency ω for which

$$\arg G(i\omega)F(i\omega) = -180°. \tag{12.9}$$

When the oscillation is internally caused (and stiction is excluded!), the oscillation frequency will typically be near the ultimate frequency. In that case the phase shift between controller output and process output is approximately 180 degrees. Then, by comparing the estimated ultimate frequency ω_u to the actual frequency ω_{osc}, it is possible to make the diagnosis as follows:

1. Exclude valve stiction as a likely cause.
2. Compute the oscillation frequency ω_{osc}.
3. Compute the ultimate frequency ω_u (12.9).
4. Define a dead-zone, i.e. an interval where both frequencies are considered to be equal.
5. Decision:

$$\omega_u \approx \omega_{osc} \Longrightarrow \text{ internally caused}$$
$$\omega_u \gtrless \omega_{osc} \Longrightarrow \text{ externally caused.}$$

It is of course possible that some external disturbance happens to have a frequency close to the ultimate frequency of the loop in question. In such a case one might wrongly indicate internal problems. However, it seems more likely that the problem is internal in such a case rather than external.

Examples

Consider some examples where the true cause of oscillation is known.

Simulated examples

Let the process be modeled by

$$G(s) = \frac{2}{(3s+1)(10s+1)}e^{-3s}$$

which is controlled by a PI-controller with $K_c = 1$ and $T_i = 15$. The closed-loop is stable and an oscillating load disturbance enters at the process output. For simulation, the system was discretised using a sampling rate of $T_s = 0.5$ seconds. The disturbance has a frequency of 0.25 $\mathrm{rad/s}$ and is a square wave. Zero-mean white noise with unit variance is also added. Before entering the loop, the sum of both signals passes a filter

$$H(q) = \frac{0.2}{q - 0.8}.$$

Using the simple diagnosis algorithm described above leads to the results presented in Table 12.1 (Case I). Simulated closed-loop data is shown in Figure 12.8. Another experiment (Case II) is then made by increasing the controller gain to $K_c = 3$ and removing the load disturbance. The closed-loop is now unstable and in connection with a saturation element (saturation limit = 10), a stable oscillation is obtained, see Figure 12.9. The diagnosis for this case (II) is also shown in Table 12.1. As can be seen the diagnosis is correct in both cases. The comparison of the frequencies was allowing a 15% variation around the calculated ultimate frequency.

Industrial examples

Flow control loop. Consider a flow control loop. From experiments it is known that this loop oscillates due to external oscillating load disturbances,

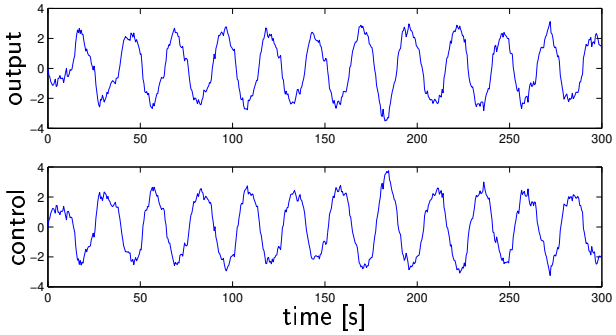


Figure 12.8: Process output and control signal when the loop is **externally** excited (Case I).

case	K_c	$\hat{K}_c$	T_i	$\hat{T}_i$	ω_u	ω_{osc}	diag.
I	1.0	1.0	15	13.4	0.31	0.24	**ext.**
II	3.0	3.0	15	11.4	0.30	0.29	**int.**

Table 12.1: Results from the diagnosis for the two simulated examples. The frequencies are expressed in $[\mathrm{rad/s}]$.

see Figure 12.10. A discrete-time process model has been identified from another data batch where the loop was not oscillating and a setpoint step was performed. These data are shown as the bottom plot of Figure 12.10. The (continuous-time) process model was estimated to be

$$\hat{G}_f(s) = \frac{3.0}{9.5s + 1} e^{-7s}.$$

The identified and the real (known) controller parameters, the estimate of the ultimate frequency and the resulting diagnosis are presented in Table 12.2.

Consistency control loop. Consider now an oscillating consistency control loop. This loop is known to be too tightly tuned and oscillates due to some static nonlinearity. For this loop some identification data is available from a series of open-loop PRBS experiments. Both data sets are shown in Figure 12.11. From the identification data, a simple process model was

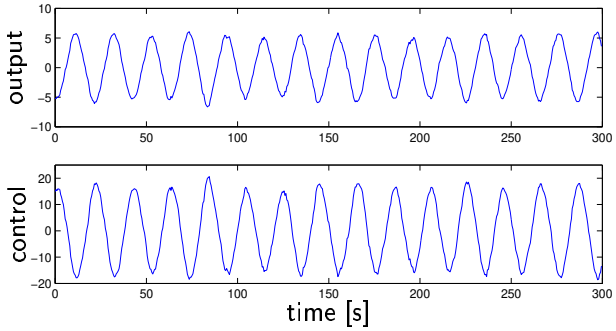


Figure 12.9: Process output and control signal when the loop is **internally** excited (Case II).

estimated and the continuous-time transfer function is given by

$$\hat{G}_c(s) = \frac{-0.05}{28s+1} e^{-7s}.$$

Then, oscillating data was used to estimate the PI-controller parameters and hence the ultimate frequency ω_u. Using the diagnosis algorithm, the results presented in Table 12.2 are obtained. For the diagnosis, again a deviation of $\pm 15\%$ from the ultimate frequency was allowed to conclude that the two frequencies are equal. As can be seen, the algorithm gives the correct diagnosis also in this case.

case	K_c	$\hat{K}_c$	T_i	$\hat{T}_i$	ω_u	ω_{osc}	diag.
flow	0.06	0.06	6.0	5.95	0.203	0.354	**ext.**
cons.	-7	-6.75	30	22.4	0.233	0.208	**int.**

Table 12.2: Results from the diagnosis for the two industrial examples. The frequencies are expressed in $[\mathrm{rad/s}]$.

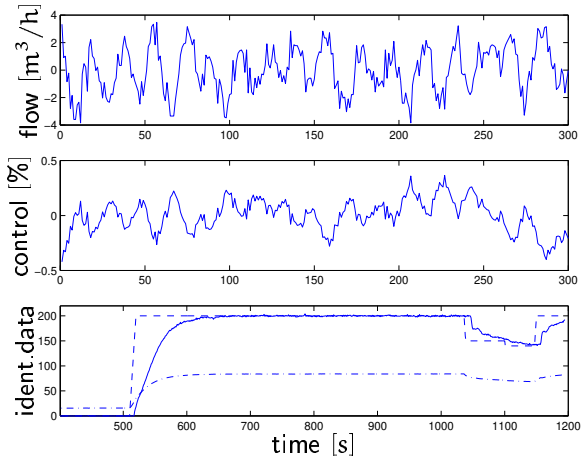


Figure 12.10: Flow loop data. Top: Oscillating process output, middle: oscillating control signal. Bottom: identification data. Setpoint (dashed), process output (solid) and control signal (dash-dotted).

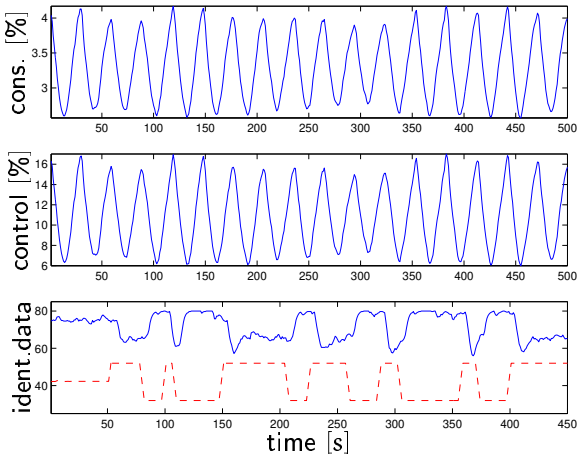


Figure 12.11: Consistency loop data. Top: Oscillating process output, middle: oscillating control signal. Bottom: identification data. Setpoint (dashed), process output (solid) and control signal (dash-dotted).

Chapter 13

Summary and conclusions

Part I

The concept of the minimum-variance based performance index is very popular due to its conceptual and computational simplicity. However, in some cases one has to be aware of the limitations of this approach. Therefore, a modification has been proposed in order to achieve a more realistic performance assessment in these cases.

It has also been discussed how the same approach can be used when additional process knowledge is available. If process and controller models are available, one can assess the performance for other than the stochastic control task or restrict the assessment to certain controller structures.

Care has to be taken when dealing with oscillating loops. The Harris index should not be used for these loops directly. A better alternative is to detect oscillations prior to evaluating the Harris index.

It has also been shown that the Harris performance index can be computed even for other than the actual sampling rate. In that way it is possible to quantify the potential benefit of faster sampling.

Part II

Dynamic models of the controlled processes are usually unavailable for control performance assessment. However, described in this thesis, the availability of models is of great advantage.

Still, a process or controller model should not be a prerequisite for the use of a monitoring tool. Therefore, the concept of event-triggered estimation was introduced: whenever there is a possibility to estimate either the process or the controller, a monitoring tool should do so.

The estimation of the controller transfer function is always possible since excitation conditions are satisfied for any disturbance. However when one wants to identify the process model or estimate the process dead-time, the only possibility is to wait for setpoint changes. The presence of disturbances is not sufficient in that case.

When using a monitoring tool which continuously collects data, some screening and data selection has to be done prior to estimating the process model. A method which selects informative identification data from normal operating data has been presented. The method is of low complexity and can be implemented on-line. It makes use of a steady-state detection algorithm in connection with the detection of significant setpoint changes.

Dead-time estimation can be carried out by fitting Laguerre models to the identification data. The dead-time can then obtained by comparison of the phase curve of the allpass part of the Laguerre model and that of a pure dead-time.

Part III

Oscillations are a very drastic form of performance degradation. The benefit by using feedback is lost and in many cases the oscillation is only possible due to the feedback mechanism.

The automatic detection of oscillating control loops is a relatively simple task whereas the diagnosis of the root cause is much more difficult. The success of such an undertaking is very much dependent on the available information about the loop in question.

It has been shown that oscillations caused by valve stiction can be easily diagnosed. The required information is whether or not the process in the loop is integrating. For self-regulating processes two different methods have been proposed, one of which also can be used for integrating processes.

If the process dead-time is also known (or being estimated), one may use the concept of the Harris index in order to distinguish between internally and externally caused oscillations. If a simple process model is also available, the same task can be solved by studying the ultimate frequency

The task of distinguishing causes like dead-band, backlash or quantisation has not been addressed. Firstly because is may be impossible, secondly because the mentioned distinction between internal and external causes seems much more relevant. The diagnosis of backlash and other nonlinearities can probably more easily be solved using intelligent, self-diagnosing actuators and sensors.

Final remarks

The starting point of this thesis was the use of indices for performance assessment. During the work reported here, it became evident, that one of the most important problems in process industry is the widespread presence of oscillations. It is therefore my belief that these problems have to be the primary target for a monitoring tool. The tuning of the controller may be of secondary interest if the control equipment itself does not work properly.

Of course, the methods presented here do not solve all diagnosis problems in connection with oscillations. It is difficult to assess to what extent diagnosis can be done given only normal operating data.

When the problem of oscillations is dealt with as well as possible, however, the next step should be the use of performance index-like methods which assess the tuning of single controllers. Many methods for this purpose are available some of which have been discussed and proposed in this thesis.

It is my belief that automatic performance monitoring tools can be of great help. However, there is no doubting the fact that at some point the presence of a human being with a considerable amount of process understanding is unalterable.

Appendix A

Proof of Theorem 3.1

Let the true process transfer function $G(q^{-1})$ and its model $\hat{G}(q^{-1})$ be given by (3.18) and (3.19). Assume that $\deg(B) < \deg(A)$ and that all poles and zeros are stable. The controller design will not be done using minimum-variance control since we are to place one of the poles arbitrarily. We will therefore make use of the Internal Model Control (IMC) concept, which is very close in spirit to minimum-variance control, see (Morari and Zafiriou, 1989). Factor therefore the process model as

$$\hat{G}(q^{-1}) = \underbrace{\frac{B(q^{-1})}{A(q^{-1})}}_{\hat{G}_-} \cdot \underbrace{q^{-\hat{d}}}_{\hat{G}_+}. \tag{A.1}$$

In order to place one pole in $q = \mu$, using IMC we need a filter

$$F(q^{-1}) = \frac{1-\mu}{1-\mu q^{-1}}.$$

The IMC controller written down for a conventional control loop reads as

$$C(q^{-1}) = \frac{F(q^{-1})\hat{G}_-^{-1}(q^{-1})}{1 - F(q^{-1})\hat{G}_+(q^{-1})} \tag{A.2}$$

and results in the following open-loop transfer function

$$L(q^{-1}) = C(q^{-1})G(q^{-1}) = \frac{(1-\mu)q^{-d}}{1-\mu q^{-1} - (1-\mu)q^{-\hat{d}}}. \tag{A.3}$$

The closed-loop transfer function from setpoint to output is then obtained as

$$\begin{aligned} G_{cl}(q^{-1}) &= \frac{L(q^{-1})}{1+L(q^{-1})} \\ &= \frac{(1-\mu)q^{-d}}{1-\mu q^{-1}-(1-\mu)q^{-\hat{d}}+(1-\mu)q^{-d}}. \end{aligned} \tag{A.4}$$

We now want to consider the robustness properties of this system. Let us therefore introduce a process gain margin K such that the new characteristic equation becomes

$$P(q^{-1}) = 1-\mu q^{-1}-(1-\mu)q^{-\hat{d}}+K(1-\mu)q^{-d} \tag{A.5}$$

Two cases have to distinguished depending on if we assume the real time-delay to be $d = \hat{d}+1$ (Case 1) or $d = \hat{d}-1$ (Case 2).

Case 1

If we rewrite (A.5) in terms of q, using the values for $K = 2(1-\varepsilon)$ and $\mu = 0.5$, we get

$$P_1(q) = q^{\hat{d}+1}-0.5q^{\hat{d}}-0.5q+(1-\varepsilon) = 0 \tag{A.6}$$

We will show that this equation has at least one root on the unit circle by using Jury's stability criterion which is for example described in Åström and Wittenmark (1997). The coefficients of (A.6) can be written as

$$a_i = [1\,,\,-0.5\,,\,0\,,\,0,\,\ldots\,,\,0\,,\,0\,,\,-0.5\,,\,(1-\varepsilon)].$$

If we set $\varepsilon = 0$, we get a symmetric coefficient scheme. The stability test then ascertains that the system does not have all roots strictly inside the unit circle. For the case of $\varepsilon > 0$ the test states that all roots lie strictly inside the unit circle. The implication of this is that for $\varepsilon = 0$ there is at least one root that lies on the stability boundary. As a matter of fact, all roots of Equation (A.6) lie exactly on the unit circle for $K = 2$ and $\mu = 0.5$, no matter what d is.

To summarise, for the values given above, the closed-loop system is exactly on the stability boundary if the controller gain is increased by a factor of two. Hence the gain margin is two.

Case 2

Using Equation (A.5) and $K = 2$, we can write down the characteristic equation for this case.

$$P_2(q) = q^{\hat{d}} - 0.5q^{\hat{d}-1} + q - 0.5 = 0.$$

It can immediately be seen that $q = 0.5$ is a root of $P(q)$. Applying polynomial division by $(q - 0.5)$ gives

$$P_2(q) = (q - 0.5)(q^{\hat{d}-1} + 1) = 0.$$

Since $(q^{\hat{d}-1} + 1)$ has all its roots on the unit circle, we proved that for $K = 2$ and $\mu = 0.5$, the system has reached the stability boundary.

Appendix B

The adjustment of the noise covariance

Consider the continuous-time system

$$\begin{aligned} \dot{x}(t) &= \mathbf{A}x(t) + Bv(t) \\ y(t) &= Hx(t) \\ \mathbf{R}_c &= E\{Bv^2(k)B^T\} = B\eta^2 B^T. \end{aligned} \tag{B.1}$$

Let this system be sampled with sampling rate h. Then the covariance matrix of the sampled noise becomes

$$\mathbf{R}_d = \int_0^h e^{\mathbf{A}t} B\eta^2 B^T e^{\mathbf{A}^T t}\, dt. \tag{B.2}$$

Even though this quantity can be calculated, see e.g. Söderström (1994), consider the approximation

$$\mathbf{R}_d \approx \frac{1}{h} G\eta^2 G^T, \tag{B.3}$$

where

$$G = \int_0^h e^{\mathbf{A}t} B\, dt \tag{B.4}$$

which can be calculated using the matrix exponential. Equation (B.3) can be motivated as follows. It holds that

$$\begin{aligned} G &= \int_0^h e^{\mathbf{A}t} B \, dt = \int_0^h (\mathbf{I} + \mathbf{A}t + \cdots) B \, dt \\ &= \left(hI + \frac{h^2}{2}\mathbf{A} \right) B + \mathcal{O}(h^3) \end{aligned} \tag{B.5}$$

and hence

$$G\eta^2 G^T \approx h \left(hB\eta^2 B^T + \frac{h^2}{2} \left(\mathbf{A}B\eta^2 B^T + B\eta^2 B^T \mathbf{A}^T \right) \right) + \mathcal{O}(h^4) \tag{B.6}$$

Furthermore, from (B.2) one gets for the exact covariance

$$\begin{aligned} \mathbf{R}_d &= \int_0^h e^{\mathbf{A}t} B\eta^2 B^T e^{\mathbf{A}^T t} \, dt \\ &= \int_0^h \left(B\eta^2 B^T + t(\mathbf{A}B\eta^2 B^T + B\eta^2 B^T \mathbf{A}^T) + \cdots \right) \, dt \\ &= hB\eta^2 B^T + \frac{h^2}{2} \left(\mathbf{A}B\eta^2 B^T + B\eta^2 B^T \mathbf{A}^T \right) + \mathcal{O}(h^3). \end{aligned} \tag{B.7}$$

From (B.6) and (B.7) it can be seen that (B.3) is exact up to the order of h^2.

When using deterministic re-sampling, converting from one discrete-time process with sampling interval h_1 to another one with sampling interval h_2, a compensation of the kind (B.3) has to be made twice. For the sake of the argument assume that the re-sampling is made by first converting to a continuous-time model.

The first step is then to transform the original ARMA process

$$\begin{aligned} x(kh_1 + h_1) &= \mathbf{F}_{d1} \, x(kh_1) + G_{d1} \, e_1(kh_1) \\ y(kh_1) &= H \, x(kh_1) \\ \mathbf{R}_{d1} &= E\{G_{d1} \, e_1^2(kh_1) \, G_{d1}^T\} = G_{d1} \, \sigma^2 \, G_{d1}^T. \end{aligned} \tag{B.8}$$

to a continuous-time counterpart. Assume that this is done using deterministic sampling, i.e. by inverting the relationship (B.4) to find B. The continuous-time model thus obtained has the noise covariance

$$\mathbf{R}_c = B \, \sigma^2 \, B^T. \tag{B.9}$$

As shown by (B.6) and (B.7), the *approximate* way of calculating $\mathbf{R}_c$ from the given $\mathbf{R}_{d1}$ is off by a factor h_1, i.e. were one to complete the loop by converting back to discrete-time using exact sampling the resulting noise covariance would be approximately $\mathbf{R}_{d1}/h_1$. Hence, the continuous-time covariance needs to be multiplied by h_1 (or equivalently B substituted by $\sqrt{h_1}B$) to more accurately represent the discrete-time covariance.

The second step would then be to apply deterministic sampling to the continuous-time model. This yields

$$\begin{aligned} x(kh_2 + h_2) &= \mathbf{F}_{d2}\, x(kh_2) + G_{d2}\, e_2(kh_2) \\ y(kh_2) &= H\, x(kh_2) \\ \mathbf{R}_{d2} &= E\{G_{d2}\, e_2^2(kh_2)\, G_{d2}^T\} = G_{d2}\, \sigma^2\, G_{d2}^T. \end{aligned}$$

Again, as shown above in (B.3), the *approximate* way of calculating $\mathbf{R}_{d2}$ from $\mathbf{R}_c$ needs correction by dividing by h_2, or equivalently G_{d1} substituted by $G_{d1}/\sqrt{h_2}$.

The presented method in fact calculates the new G_{d2} directly from G_{d1}. Combining the two steps described above means that the vector G_{d2} has to be substituted by $\sqrt{\frac{h_1}{h_2}}G_{d2}$. Note that the substitution does not yield a perfect compensation. It will be a good approximation for small sampling intervals and get worse when the sampling interval increases.

Appendix C

Proof of Theorem 8.1

In the Laplace domain we have the Padé approximation, which is often derived based on a series expansion of the exponential. There is however an alternative derivation based on a phase curve argument, see Cunningham (1954) or Giloi and Lauber (1963), which is briefly summarized below.

Consider a continuous-time n-th order rational allpass element which satisfies the condition $|G_{ap}(i\omega)| = 1 \, \forall \omega$,

$$G_{ap}(s) = \frac{1 - a_1 s T_d + a_2 (s T_d)^2 \mp \ldots (-1)^n a_n (s T_d)^n}{1 + a_1 s T_d + a_2 (s T_d)^2 + \ldots a_n (s T_d)^n}. \tag{C.1}$$

Let the phase curve of (C.1) be described by

$$\varphi(\omega) = \arg\{G_{ap}(e^{i\omega T_s})\}. \tag{C.2}$$

The phase curve of a pure dead-time element $G(s) = e^{-sT_d}$ is

$$\varphi_{T_d}(\omega) = -\omega T_d. \tag{C.3}$$

The parameters a_i shall be chosen such that (C.1) approximates e^{-sT_d} as well as possible. This can for example be done by choosing a_i such that the phase error $\Delta\varphi(\omega) = \varphi(\omega) - \varphi_{T_d}(\omega)$ goes to zero for low frequencies.

From (C.3) it is obvious that the phase is a linear function in the frequency, therefore

$$\frac{d\varphi}{d\omega} = -T_d. \tag{C.4}$$

Then, require (C.4) to hold at least at frequency zero, i.e.

$$\begin{aligned} \left.\frac{d\varphi}{d\omega}\right|_{\omega=0} &= -T_d, \\ \left.\frac{d^i\varphi}{d\omega^i}\right|_{\omega=0} &= 0, \quad i = 3, \ldots, 2n-1. \end{aligned} \tag{C.5}$$

Note that even though the allpass element (C.1) is of order n, it is possible to set the first $2n-1$ derivatives of (C.4) equal to zero (when evaluated at zero frequency). The reason for that is the fact that the phase curve of (C.1) is an odd function in the frequency and therefore, all even-order derivatives of (C.2) are zero for any frequency.

Using this alternative derivation, the limit

$$\hat{T}_d = \lim_{\omega\to 0}\left(-\frac{\varphi(\omega)}{\omega}\right) \tag{C.6}$$

follows immediately from a Taylor series expansion of $\varphi(\omega)$ around $\omega = 0$,

$$\begin{aligned} \varphi(\omega) &= \varphi(0) + \left.\frac{d\varphi(\omega)}{d\omega}\right|_{\omega=0}\omega + \mathcal{O}(\omega^2) \\ &= -\omega T_d + \mathcal{O}(\omega^2). \end{aligned} \tag{C.7}$$

Even though the derivation was done for a continuous-time allpass element, the same reasoning holds for a discrete-time rational allpass filter

$$H_{ap}(q^{-1}) = \frac{1 + a_1 q^{-1} + a_2 q^{-2} + \ldots + a_n q^{-n}}{a_n + a_{n-1} q^{-1} + \ldots + a_1 q^{-n+1} + q^{-n}}, \tag{C.8}$$

as obtained from the factorisation of the estimated (discrete-time) Laguerre model.

Appendix D

Differentiation of the process output

An ad hoc remedy to overcome the problems with the CCF-method for integrating processes would be to differentiate the process output in order to compensate for the integration in the process. In this section, it will be motivated why such an approach would not work.

The stiction case

In the case of stiction the process output is triangular due to the integration in the process. The controller output will – as a consequence – not be ideally triangular (as it is for self-regulating processes) but rather be a smoothened triangular signal. However, the valve position will still be a rectangular signal due to the stiction phenomenon, see Figure D.1. Note that the control signal is plotted as perfectly triangular rather than smoothed which will not change the reasoning. Obviously the CCF between process output $y(t)$ and control signal $u(t)$ is an even function. When the process output is differentiated[1], the (rectangular) derivative can be obtained. Hence, one could – after differentiation (and filtering) of the process output – use the CCF method in the same way as was described for self-regulating processes.

[1] Since the signal in question is noisy, some filtering would have to be used. The filter must not change the phase of the signal.

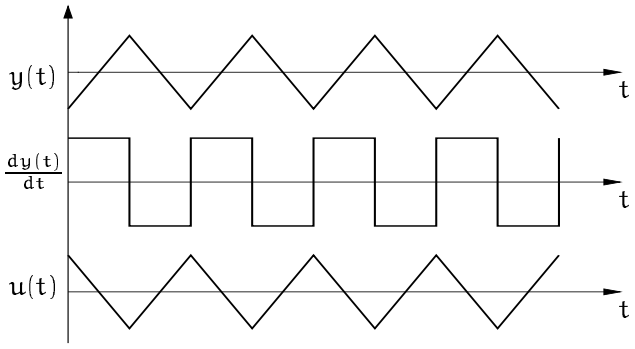


Figure D.1: Ideal signals in the stiction case. Process output (top), derivative of the process output (middle) and controller output (bottom).

The non-stiction case

Consider an oscillating control loop where stiction is not the main cause. In the non-stiction case, a control loop oscillates approximately at its ultimate frequency. This means that the process output and the control signal are shifted against each other by approximately $-\pi$.

Now, if the process output is differentiated in order to obtain compensate for the integration, its phase will be shifted by $-\pi/2$. Let the process output be described as $y(t) = \sin(\omega t)$. The derivative of the process output is then $\frac{d}{dt}y(t) = \cos(\omega t) = \sin(\pi/2 - \omega t)$. At the same time the controller output is basically[2] $u(t) = -\sin(\omega t)$, see Figure D.2.

Then, when using the CCF-method – for the derivative of the process output $y(t)$ and control signal $u(t)$ – the CCF is obviously an odd function. As this is the same result as for the stiction case, there is obviously a problem with this way of dealing with integrating processes.

[2]The controller does of course more than a simple negation. If the input to a PI-controller is triangular, the controller output will typically be a smoothed triangular signal (resembling a sinusoid). However, for the problem considered here, this assumption is sufficient.

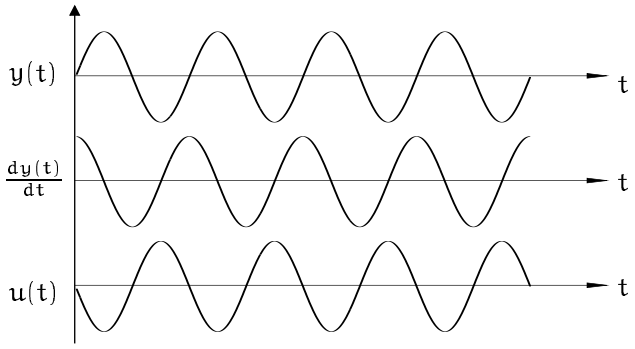


Figure D.2: Ideal signals in the non-stiction case. Process output (top), derivative of the process output (middle) and controller output (bottom).

Conclusions

Summarising the discussion above the following conclusions can be stated: When dealing with integrating processes,

- the CCF-function between process output $y(t)$ and control signal $u(t)$ is *even* no matter what the root cause of the oscillation is,
- the CCF-function between *the derivative of* process output $y(t)$ and control signal $u(t)$ is *odd* no matter what the root cause of the oscillation is.

Hence, the ad hoc idea of using differentiation does not solve the problem. This is a somewhat surprising result and can be explained as follows: The reason why the CCF-method does work is that the correlation structure between process output and control signal is also determined by the special signal shapes in the stiction case. (Even though it can allow rather large deviations from the ideal shapes!). If there is integration in the process, this advantage of the CCF-method is destroyed and cannot be retrieved.

Example

Consider finally two simple examples (stiction and non-stiction respectively) where both the process output itself and its derivative are used for the CCF-

method. In Figure D.3 it can be seen that in both the stiction and the non-stiction case the CCF-method yields identical results. Hence no diagnosis can be made. This result illustrates the conclusions given above.

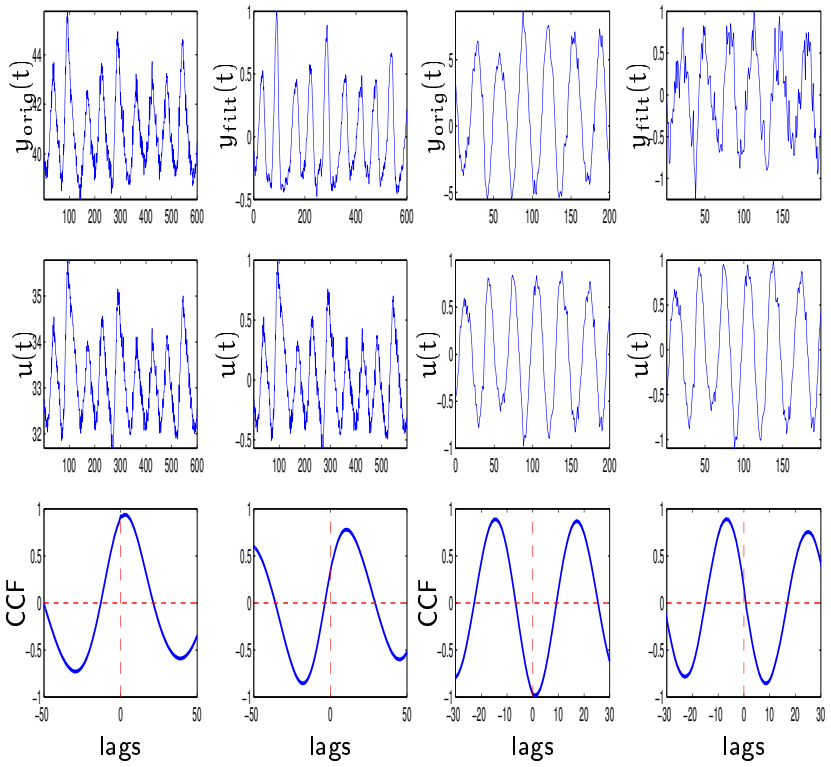


Figure D.3a: Stiction.

Figure D.3b: No stiction.

Figure D.3: The plots in (a) and (b) are: Top right: original process output. Top left: filtered and differentiated process output. Middle: control signal (unchanged). Bottom left: CCF-function for the original data. Bottom right: CCF-function for the filtered data. Note that the process gain in the stiction case is negative why the CCF is inverted.

Appendix E

A graphical software tool

Most of the work reported in this thesis and some published algorithms for performance monitoring have been implemented in MATLAB™ for testing on real data. Furthermore, a graphical software tool was developed which facilitates the handling of large amounts of data and the evaluation of different algorithms for performance monitoring. Figure E.1 shows the main functions implemented in the tool. The terms used in the figure refer to

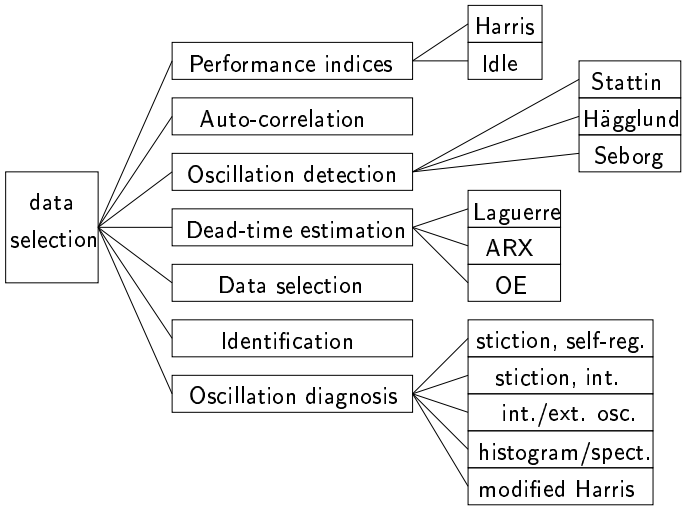


Figure E.1: Functionality of the graphical software tool for performance monitoring.

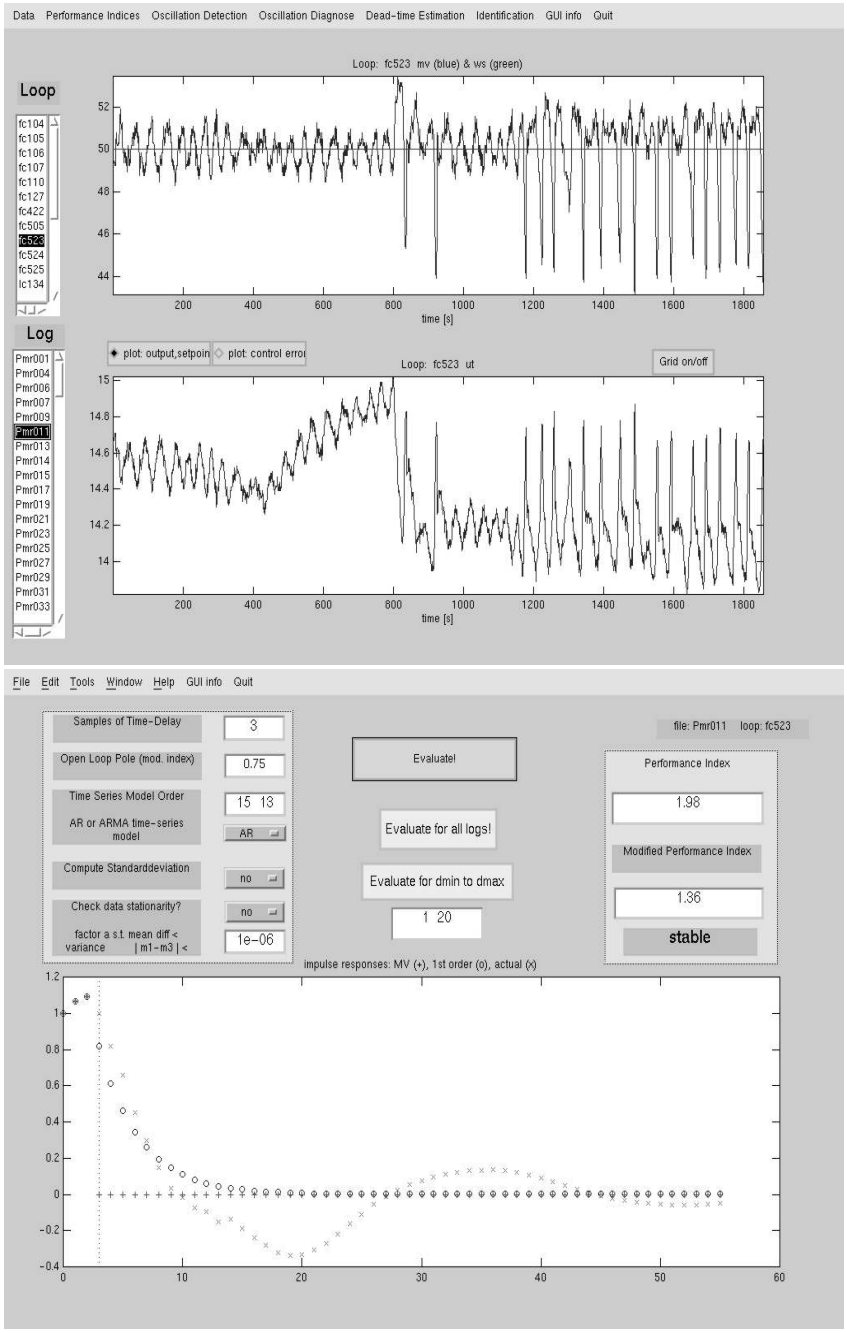
the algorithms as shown in Table E.1. For the algorithms which relate to this thesis, the examples of the user interface are shown in the Figures referred to

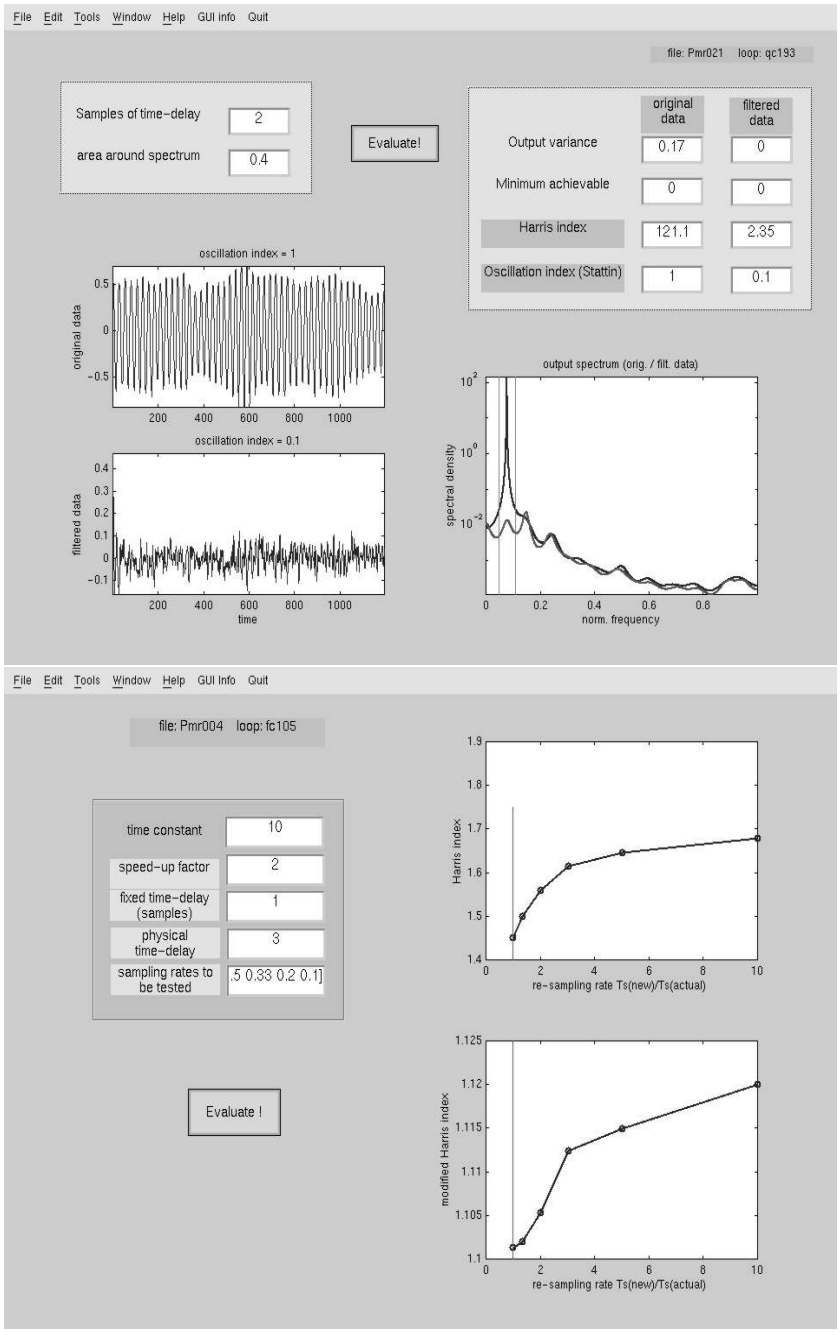
in Table E.1. Note that this tool was designed for research purposes mainly.

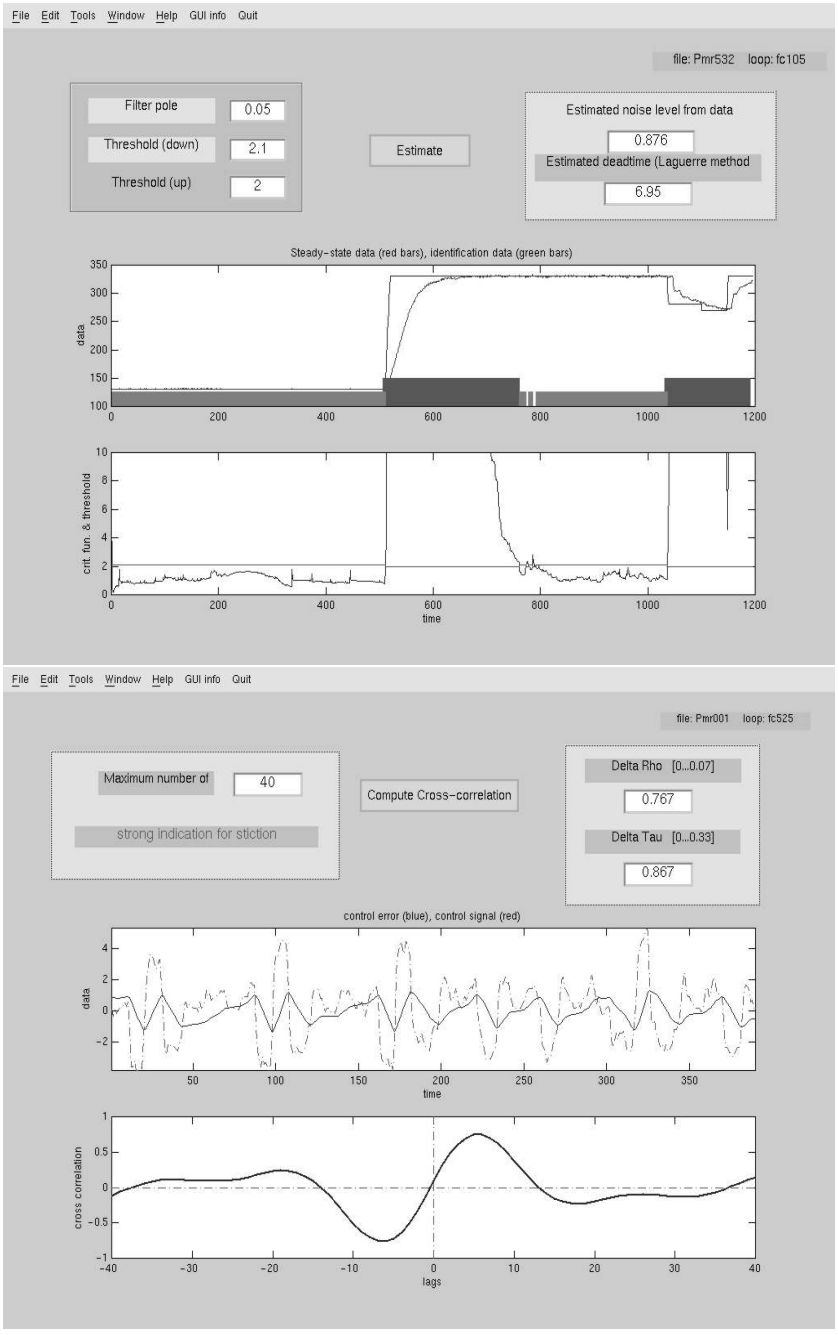
feature	reference	figure location
data selection	—	page 193, top
identification	Chapter 6,7	page 195, top
Harris	Chapters 3 and 4	pages 193 bottom, 194 bottom
idle	idle index (Hägglund, 1999)	–
Stattin	(Forsman and Stattin, 1999)	–
Hägglund	(Hägglund, 1995)	–
Seborg	(Seborg and Miao, 1999)	–
Laguerre	Chapter 8	page 195 top
ARX	(Isaksson *et al.*, 2000)	–
OE	(Isaksson *et al.*, 2000)	–
stiction, self-reg.	Chapter 10	page 195 bottom
stiction, int.	Chapter 11	page 196 top
int./ext. oscillation	Chapter 12	page 196 bottom
modified Harris	Chapter 12	page 194 top

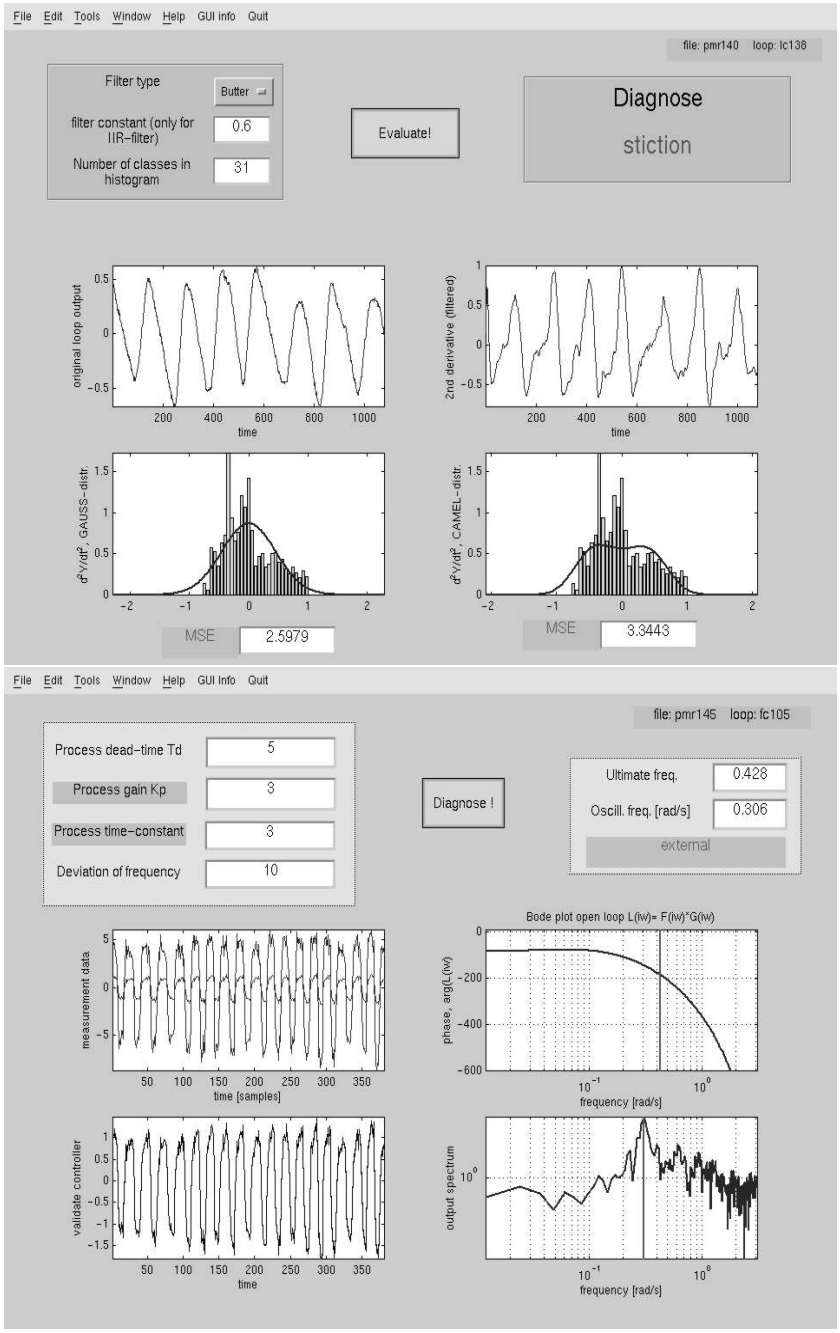
Table E.1: Reference and location of the GUI-examples for the implemented algorithms.

This implies that some design parameters which easily can have default values in a commercial tool still can be changed by the user. Furthermore, the tool is not supposed to work in an autonomous manner (with the exception that for example some of the indices may be calculated for all available data sets).


Data Performance Indices Oscillation Detection Oscillation Diagnose Dead-time Estimation Identification GUI info Quit
Loop
Log
Loop: fc523 mv (blue) & ws (green)
time [s]
plot: output,setpoin
plot: control erro
Loop: fc523 ut
Grid on/off
File Edit Tools Window Help GUI info Quit
Samples of Time-Delay
3
Open Loop Pole (mod. index)
0.75
Time Series Model Order
15 13
AR or ARMA time-series model
AR
Compute Standarddeviation
no
Check data stationarity?
no
factor a s.t. mean diff < variance | m1-m3 | <
1e-06
Evaluate!
Evaluate for all logs!
Evaluate for dmin to dmax
1 20
file: Pmr011 loop: fc523
Performance Index
1.98
Modified Performance Index
1.36
stable
impulse responses: MV (+), 1st order (o), actual (x)


File Edit Tools Window Help GUI info Quit
file: Pmr021 loop: qc193
Samples of time-delay
2
area around spectrum
0.4
Evaluate!
original data
filtered data
Output variance
0.17
0
Minimum achievable
0
0
Harris index
121.1
2.35
Oscillation index (Stattin)
1
0.1
oscillation index = 1
original data
oscillation index = 0.1
filtered data
time
output spectrum (orig. / filt. data)
spectral density
norm. frequency
File Edit Tools Window Help GUI Info Quit
file: Pmr004 loop: fc105
time constant
10
speed-up factor
2
fixed time-delay (samples)
1
physical time-delay
3
sampling rates to be tested
.5 0.33 0.2 0.1]
Evaluate !
Harris index
re-sampling rate Ts(new)/Ts(actual)
modified Harris index
re-sampling rate Ts(new)/Ts(actual)


File Edit Tools Window Help GUI info Quit
file: Pmr532 loop: fc105
Filter pole
0.05
Threshold (down)
2.1
Threshold (up)
2
Estimate
Estimated noise level from data
0.876
Estimated deadtime (Laguerre method
6.95
Steady-state data (red bars), identification data (green bars)
data
crit. fun. & threshold
time
File Edit Tools Window Help GUI info Quit
file: Pmr001 loop: fc525
Maximum number of
40
strong indication for stiction
Compute Cross-correlation
Delta Rho [0...0.07]
0.767
Delta Tau [0...0.33]
0.867
control error (blue), control signal (red)
data
time
cross correlation
lags


File Edit Tools Window Help GUI info Quit
file: pmr140 loop: lc138
Filter type
Butter
filter constant (only for IIR-filter)
0.6
Number of classes in histogram
31
Evaluate!
Diagnose
stiction
original loop output
time
2nd derivative (filtered)
time
d²Y/dt², GAUSS-distr.
d²Y/dt², CAMEL-distr.
MSE
2.5979
MSE
3.3443
File Edit Tools Window Help GUI Info Quit
file: pmr145 loop: fc105
Process dead-time Td
5
Process gain Kp
3
Process time-constant
3
Deviation of frequency
10
Diagnose !
Ultimate freq.
0.428
Oscill. freq. [rad/s]
0.306
external
measurement data
time [samples]
Bode plot open loop L(iw)= F(iw)*G(iw)
phase, arg(L(iw))
frequency [rad/s]
validate controller
time
output spectrum
frequency [rad/s]

Bibliography

Åström, K. (1970). *Introduction to Stochastic Control Theory*. Academic Press. New York.

Åström, K. and B. Wittenmark (1997). *Computer Controlled Systems, Theory and Design*. 3rd edn. Prentice Hall. Englewood Cliffs, New Jersey.

Atherton, D. (1981). *Stability of nonlinear systems*. Chistester: Research studies P.

Bendat, J. and A.G. Piersol (1967). *Measurement and Analysis of Random Data*. Wiley. New York.

Bezergianni, S. and C. Georgakis (2000). 'Controller performance assessment based on minimum and open-loop output variance'. *Cont. Eng. Pract.* **8**, 791–797.

Bialkowski, W. (1992). "Dreams vs. Reality: A View from Both Sides of the Gap". In 'Control Systems '92: Dream vs. Reality'. Whistler, Canada. pp. 283–298.

Cao, S. and R.R. Rhinehart (1995). 'An efficient method of on-line identification of steady state'. *J.Proc.Cont.* **5**(6), 363–374.

Carrette, P., G. Bastin, Y.Y. Genin and M. Gevers (1996). 'Discarding data may help in system identification'. *IEEE Trans. Signal Processing* **44**(9), 2300–2310.

Cunningham, W. (1954). 'Time-delay networks for an analog computer'. *IRE Transactions* pp. 16–18.

Deibert, R. (1994). Model based fault detection of valves in flow control loops. In 'IFAC Symp. on Fault Detection, Supervision and Safety for Techn. Processes. – SAFEPROCESS '94'. Vol. 2. Espoo, Finland. pp. 445–450.

Desborough, L. and T. Harris (1992). 'Performance assessment measures for univariate feedback control'. *The Canadian Journal of Chemical Engineering* **70**, 1186–1197.

Desborough, L. and T. Harris (1993). 'Performance assessment measures for univariate feedforward/feedback control'. *The Canadian Journal of Chemical Engineering* **71**, 605–616.

DeVries, W. and S.M. Wu (1978). 'Evaluation of process control effectiveness and diagnosis of variation in paper basis weight via multivariate time-series analysis'. *IEEE Trans. on Automatic Control* **AC-23**(4), 702–708.

Dumont, G., C.C. Zervos and G.L. Pageau (1990). 'Laguerre-based Adaptive Control of *pH* in an Industrial Bleach Plant Extraction Stage'. *Automatica* **26**(4), 781–787.

Elnaggar, A., G.A. Dumont and A.L. Elshafei (1991). Delay estimation using variable regression. In 'American Control Conference'. Boston, MA. pp. 2812–2817.

Ender, D. (1993). 'Process control performance: Not as good as you think'. *Control Eng.* **40**(10), 180–190.

Ender, D. (1997). Implementation of dead-band reset scheduling for the elimination of stick-slip cycling in control valves. In 'Process Control Electrical & Information Conf.'. pp. 83–88.

Eriksson, P. (1995). On Control Performance Assessment and Disturbance Classification. Licentiate's thesis. Royal Institute of Technology, Dept. of Signals, Sensors and Systems. Stockholm, Sweden.

Eriksson, P. and A.J. Isaksson (1994). Some aspects of control loop performance monitoring. In 'The 3rd IEEE Conference on Control Applications'. Glasgow, Scotland. pp. 1029–1034.

Ettaleb, L. (1999). Control Loop Performance Assessment and Oscillation Detection. PhD thesis. Dept.of Electrical and Computer Engineering, Faculty of Applied Science, The University of British Columbia. Vancouver, Canada.

Ettaleb, L., M.S. Davies, G.A. Dumont and E. Kwok (1996). Monitoring oscillations in a multiloop system. In 'Proceedings of the 1996 IEEE Int. Conf. on Control Applications'. Dearborn, MI. pp. 859–863.

Forsman, K. (2000). On detection and classification of valve stiction. In 'TAPPI Conf. on Process Control'. Williamsburg VA, USA.

Forsman, K. and A. Stattin (1999). A new criterion for detecting oscillations in control loops. In 'European Control Conference'. Karlsruhe, Germany. CP8-3.

Forssell, U. and L. Ljung (1999). 'Closed-loop identification revisited'. *Automatica* **35**, 1215–1241.

Franklin, G., J.D. Powell and M. Workman (1998). *Digital Control of Dynamic Systems*. 3rd edn. Addison-Wesley.

Fu, Y. and G.A. Dumont (1995). On-line evaluation of control loop performance. In '4th IEEE Conference on Control Applications'. Albany, New York. pp. 655–656.

Giloi, W. and R. Lauber (1963). *Analogrechnen. Programmierung, Arbeitsweise und Anwendung des elektronischen Analogrechners*. Springer-Verlag. In German.

Hägglund, T. (1995). 'A control-loop performance monitor'. *Cont.Eng.Pract.* **3**(11), 1543–1551.

Hägglund, T. (1999). 'Automatic detection of sluggish control loops'. *Cont.Eng.Pract.* **7**(12), 1505–1511.

Harris, T. (1989). 'Assessment of control loop performance'. *The Canadian Journal of Chemical Engineering* **67**, 856–861.

Harris, T., C.T. Seppala and L.D. Desborough (1999). 'A review of performance monitoring and assessment techniques for univariate and multivariate control systems'. *J.Proc.Cont.* **9**(1), 1–17.

Harris, T., C.T. Seppala, P.J. Jofriet and B.W. Surgenor (1996*a*). 'Plant-wide feedback control performance assessment using an expert-system framework'. *Cont.Eng.Pract.* **4**(9), 1297–1303.

Harris, T., F. Boudreau and J.F. MacGregor (1996*b*). 'Performance assessment of multivariable feedback controllers'. *Automatica* **32**(11), 1505–1518.

Hassibi, B. (1996). "Indefinite Metric Spaces in Estimation, Control and Adaptive Filtering". PhD thesis. Information Systems Laboratory, Dept. of Electrical Engineering, Stanford University.

Hines, M. and D.C. Montgomery (1990). *Probability and Statistics in Engineering and Management Science*. 3rd edn. Wiley.

Horch, A. and A.J. Isaksson (1998). A method for detection of stiction in control valves. In 'IFAC Workshop on On-Line-Fault Detection and Supervision in the Chemical Process Industry'. Lyon, France. Session 4B.

Huang, B. and S. Shah (1999). *Performance Assessment of Control Loops. Theory and Applications*. 1st edn. Springer.

Huang, B. and S.L. Shah (1997). Practical issues in multivariable feedback control performance assessment. In 'Preprints IFAC ADCHEM'. Banff, Canada. pp. 429–434.

Huang, B., S.L. Shah and E.K. Kwok (1997). 'Good, bad or optimal? performance assessment of multivariable processes'. *Automatica* **33**(6), 1175–1183.

Isaksson, A. (1996). PID controller performance assessment. In 'Control Systems '96'. Halifax, Nova Scotia, CA. pp. 163–169.

Isaksson, A., A. Horch and G.A. Dumont (2000). Event-triggered deadtime estimation – comparison of methods. In 'Control Systems 2000'. Victoria, B.C., Canada. pp. 209–215.

Isaksson, A., A. Horch and G.A. Dumont (2001). Event-triggered deadtime estimation from closed-loop data. In 'American Control Conference'. Arlington, USA. Submitted.

Isaksson, M. (1997). A comparison of some approaches to time-delay estimation. Master's thesis. Dept. of Automatic Control, Lund Inst. of Technology.

Isermann, R. and P. Ballé (1997). 'Trends in the application of mode-based fault detection and diagnosis of technical processes'. *Cont.Eng.Pract.* **5**(5), 709–719.

Jofriet, P., C. Seppala, M. Harvey, B. Surgenor and T. Harris (1995). An expert system for control loop performance analysis. In '81st Annual Meeting, Technical Section, CPPA'. Canadian Pulp and Paper Association. pp. B41–49.

Kendra, S. and A. Cinar (1997). 'Controller performance assessment by frequency domain techniques'. *J.Proc.Cont.* **7**(3), 181–194.

Ko, B. and T.F. Edgar (1998). Assessment of Achievable PI Control Performance for Linear Processes with Dead Time. In 'Proc. American Control Conference'. Philadelphia, USA. pp. 1548–1552.

Kozub, D. (1996). Controller performance monitoring and diagnosis: Experiences and challenges. In 'Chemical Process Control V'. Tahoe City, California. pp. 83–96.

Kozub, D. and C.E. Garcia (1993). Monitoring and diagnosis of automated controllers in the chemical process industries. In 'AIChE Meeting'. St. Louis, MO.

Kurz, H. (1979). Digital parameter-adaptive control of processes with unknown constant or time-varying dead time. In '5th IFAC Symposium on Identification and System Parameter Estimation'. Darmstadt. pp. 1187–1194.

Kurz, H. and W. Goedecke (1981). 'Digital parameter-adaptive control of processes with unknown dead time'. *Automatica* **17**, 245–252.

Lin, L., F. Paradis and A. Roche (1998). Several practical issues in control loop monitoring and diagnosis. In 'Control Systems 98'. Porvoo, Finland. pp. 245–250.

Ljung, L. (1999). *System Identification, Theory For The User*. 2nd edn. Prentice Hall. Englewood Cliffs.

Lynch, C. and G.A. Dumont (1996). 'Control Loop Performance Monitoring'. *IEEE Trans.Cont.Syst.Techn.* **4**(2), 185–192.

MacGregor, J. (1976). 'Optimal choice of the sampling interval for discrete process control'. *Technometrics* **18**(2), 151–160.

Marí, J., A. Dahlén and A. Lindquist (2000). 'A covariance extension approach to identification of time-series'. *Automatica*. To be published.

Morari, M. and E. Zafiriou (1989). *Robust Process Control*. Prentice-Hall. Englewood Cliffs.

Müller, U. (1986). *Zur kontinuierlichen Analyse der Stabilität von Regelkreisen (On the continuous stability analysis of control loops)*. Vol. VDI Reihe 8, Nr. 103. VDI Verlag. Düsseldorf. PhD Thesis, Dept. of Cont. Eng., RWTH Aachen, Germany (in GSerman).

Ogawa, S. (1998). A data analysis and graphical presentation system for control loop performance assessment. In 'Proc. TAPPI Process Control Conference 98'. Vancouver, Canada.

Olsson, H. (1996). Control Systems with Friction. PhD thesis. Department of Automatic Control, Lund Institute of Technology.

Owen, J. (1997). 'Automatic control loop monitoring and diagnostics'. Patent Application. Int. Publ. No.WO 97/41494, Int. Appl. No.PCT/CA97/00266.

Owen, J., D. Read, H. Blekkenhorst and A.A. Roche (1996). A mill prototype for automatic monitoring of control loop performance. In 'Preprints Control Systems '96'. Halifax, Canada. pp. 171–178.

Padé, H. (1892). 'Sur la représentation approchée d'une fonction par des fractions rationelles'. *Annales de l'Ecole Normale* **9**(3), 1–93, suppl.

Panagopoulos, H., T. Hägglund and K.J. Åström (1997). The Lambda method for tuning PI controllers. Technical report. Department of Automatic Control. Lund Institute of Technology, Lund, Sweden.

Perrier, M. and A.A. Roche (1992). Towards Mill-Wide Evaluation of Control Loop Performance. In 'Control Systems '92'. Whistler, Canada. pp. 205–209.

Piiponen, J. (1998). 'Nonideal valves in control: tuning of loops and selection of valves'. *Pulp & Paper Canada* **99**(10), 64–69.

Rhinehart, R. (1995). A watchdog for controller performance monitoring. In 'Proceedings of the American Control Conference'. Seattle, Washington. pp. 2239–2240.

Seborg, D. and T. Miao (1999). Automatic detection of excessively oscillatory feedback control loops. In 'IEEE Int. Conf. on Control Applications'. Hawaii, USA. pp. 359–364.

Söderström, T. (1991). 'Computing stochastic continuous-time models from ARMA models'. *Int. J. Control* **53**(6), 1311–1326.

Söderström, T. (1994). *Discrete-Time Stochastic Systems. Estimation & Control*. Prentice-Hall (International Series in Systems and Control Engineering).

Stanfelj, N., T.E. Marlin and J.F. MacGregor (1991). Monitoring and Diagnosing Process Control Performance: The Single-Loop Case. In 'American Control Conference'. Boston, MA. pp. 2886–2892.

Swanda, A. (1999). PID Controller Performance Assessment Based on Closed-Loop Response Data. PhD thesis. Dept. of Chemical Engineering, University of California Santa Barbara.

Swanda, A. and D.E. Seborg (1997). Evaluating the performance of PID-type feedback control loops using normalized settling time. In 'ADCHEM'97'. IFAC. Banff, Canada. pp. 283–288.

Taha, O., G.A. Dumont and M.S. Davies (1996). Detection and diagnosis of oscillations in control loops. In '35th IEEE Conference on Decision and Control'. Kobe, Japan. pp. 2432–2437.

Thornhill, N., M. Oettinger and P. Fedenczuk (1999). 'Refinery-wide control loop performance assessment'. *J.Proc.Cont.* **9**, 109–124.

Tyler, M. and M. Morari (1995). Performance assessment for unstable and nonminimum-phase systems. In 'IFAC Workshop on On-line Fault Detection and Supervision in the Chemical Process Industries'. Newcastle upon Tyne, England.

Tyler, M. and M. Morari (1996). 'Performance monitoring of control systems using likelihood methods'. *Automatica* **32(8)**, 1145–1162.

Venkataramanan, G., V. Shukla, R. Saini and R.R. Rhinehart (1997). An automated on-line monitor of control system performance. In 'American Control Conference'. Albuquerque, NM. pp. 1355–1359.

Vishnubhotla, A., S. L. Shah and B. Huang (1997). Feedback and Feedforward Performance Analysis of the Shell Industrial Closed-Loop Data Set. In 'Preprints IFAC ADCHEM'. Banff, Canada. pp. 295–300.

Wahlberg, B. (1990). 'ARMA Spectral Estimation of Narrow-Band Processes Via Model Reduction'. *IEEE Trans.Acoustics, Speech, and Signal Proc.* **38**(7), 1144–1154.

Wahlberg, B. (1991). 'System identification using laguerre models'. *IEEE Trans. on Automatic Control* **AC-36**(5), 551–562.

Wahlberg, B. (1994). 'System Identification using Kautz models'. *IEEE Trans.Autom.Control* **39**(6), 1276–1281.

Wahlberg, B. and E.J. Hannan (1993). 'Parametric Signal Modelling using Laguerre Filters'. *The Annals of Applied Probability* **3**(2), 467–496.

Wahlberg, B., L. Ljung and T. Söderström (1993). 'On sampling of continuous time stochastic processes'. *Control Theory and Advanced Technology* **9**(1), 99–112.

Wallén, A. (1997). Valve Diagnostics and Automatic Tuning. In 'Proceedings of the American Control Conference'. Albuquerque, New Mexico. pp. 2930–2934.

Wang, L. and W.R. Cluett (2000). *From Plant Data to Process Control*. Taylor & Francis.

Williams, C. (1950). 'On the Choice of the Number and Width of Classes for Chi-Square Test of Goodness of Fit'. *J. Am. Statistical. Assoc.* **45**, 77–86.