Difficulties When Applying SISO Relay Design Methods to a MIMO-System

Johansson, Karl Henrik

1993

Document Version:
Publisher's PDF, also known as Version of record

Link to publication

Citation for published version (APA):
Johansson, K. H. (1993). *Difficulties When Applying SISO Relay Design Methods to a MIMO-System*. (Technical Reports TFRT-7506). Department of Automatic Control, Lund Institute of Technology (LTH).

Total number of authors:
1

LUND UNIVERSITY

PO Box 117
221 00 Lund
+46 46-222 00 00

ISSN 0280–5316
ISRN LUTFD2/TFRT--7506--SE

Difficulties When Applying SISO Relay Design Methods to a MIMO-system

Karl Henrik Johansson

Department of Automatic Control
Lund Institute of Technology
May 1993

Department of Automatic Control
Lund Institute of Technology
P.O. Box 118
S-221 00 Lund Sweden

Document name
INTERNAL REPORT

Date of issue
May 1993

Document Number
ISRN LUTFD2/TFRT--7506--SE

Author(s)
Karl Henrik Johansson

Supervisor
Karl Johan Åström

Sponsoring organisation

Title and subtitle
Difficulties When Applying SISO Relay Design Methods to a MIMO-system

Abstract

This report discusses problems with cross-couplings in a MIMO-system when applying a SISO controller design method. The method used is the Ziegler-Nichols closed-loop method, where the ultimate gain K_u and the ultimate frequency ω_u are determined via relay experiments. The difficulties are exemplified on a 2×2-process. Further, there is some analysis done which state when a limit cycle for a relay system will be locally stable or not and explain why the difficulties appear. An approximate design method based on the relative gain array (RGA) is also included.

Key words
relay feedback, multivariable control, PI control

Classification system and/or index terms (if any)

Supplementary bibliographical information

ISSN and key title
0280–5316

ISBN

Language
English

Number of pages
16

Recipient's notes

Security classification

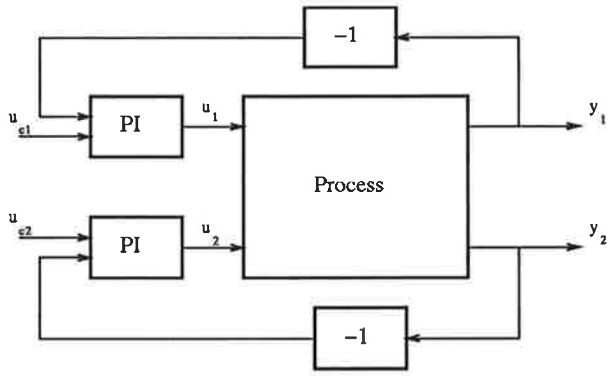

Figure 1. The 2×2-process, defined by Equation (1), is controlled by two PI-controllers.

1. Introduction

Controller design methods for SISO-systems based on relay technics are treated in e.g. [Åström and Hägglund, 1984]. These are simple methods for tuning PID-regulators, and has been successfully used in practice. This report treats the problem of using relay experiments for MIMO-systems. There has been some heuristic tries to do this extension, see e.g. [Vasnani and Loh, 1993],[Zhuang, 1992], and [Zhuang and Atherton, 1992], but the understanding in the area is far from complete. In this report we will show what difficulties that can arise and try to explain them. We will also use a design method based on the relative gain array (RGA).

Notice that our primary intention is not to design controllers that will give as good step responses as possible. We choose our controller parameters based on Ziegler-Nichols rule. Thus we will get a poorly damped closed-loop system, which will lead to the problem this report discusses.

In Section 2 we describe the artificial process and the design problem we are considering throughout the report. The relay design method is described in Section 3 together with the results from our design. Section 4 includes some analysis which state when a limit cycle for a relay system will be locally stable or not. In Section 5 we will use an approximate controller design method based on the RGA which is due to McAvoy [McAvoy, 1983]. Some conclusions are given in Section 6.

2. The Design Problem

Our goal is to design the two PI-controllers in the system in Figure 1. The PI-controllers are defined by the transfer function

$$G_{PI}(s) = K_c(1 + \frac{1}{T_i s})$$

We will consider a process slightly different from Rosenbrock's system, Example 6.8 in [Åström, 1983],

$$G(s) = \begin{pmatrix} g_{11}(s) & g_{12}(s) \\ g_{21}(s) & g_{22}(s) \end{pmatrix} = \begin{pmatrix} \frac{100}{(s+1)(s+10)^2} & \frac{200}{(s+3)(s+10)^2} \\ \frac{100}{(s+1)(s+10)^2} & \frac{100}{(s+1)(s+10)^2} \end{pmatrix} \tag{1}$$

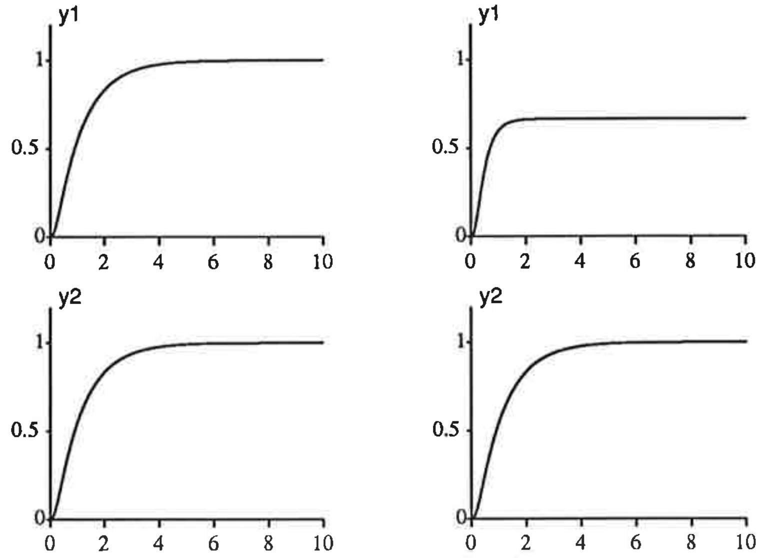

Figure 2. Step responses for the uncontrolled process. To the left is the response from u_1 to y_1 and y_2, respectively, and to the right is ditto from u_2.

We notice that separately, the four processes defined by the elements in $G(s)$ are easy to control. But because of a transmission zero at $+1$, the 2×2-system will not be minimum-phase and is therefore hard to control.

If we let each of the inputs to the uncontrolled process be a step, we get the responses in Figure 2. To the left are the step responses from u_1 to y_1 and y_2 respectively, and to the right are the responses from u_2. We notice the large cross-coupling in the process; u_1 affects y_2 and u_2 affects y_1.

3. PI-design Based on Relay Experiments

In this section we will try to design the two PI-controllers via open-loop and closed-loop relay experiments. The system is *open-loop* if one loop is closed by a relay, while the other is *open*. In the same way, the system is *closed-loop* when one loop is closed by a relay and the other is *closed* by a PI-controller. Out of these relay experiments, we will derive the parameters K_c and T_i in the PI-controllers based on Ziegler-Nichols rules.

Relay Feedback

We will now shortly discuss relay feedback, see e.g. [Holmberg, 1991].

The characteristics for a relay with input m and output u is defined by Figure 3, where d is the gain and ε is the hysteresis. The arrows mark the allowed directions.

In Figure 4 we exemplify an arrangement of an open-loop relay experiment. We are doing an experiment for the u_1–y_1-loop. Let this loop by assumption have a limit cycle and let the closed loop begin to oscillate with a frequency close to the ultimate frequency. From describing functions theory, e.g. [Åström, 1971], we know that the frequency and the amplitude of the limit cycles are given by the points where the describing function intersects with the Nyquist curve. If a denotes the amplitude of the first harmonic of the relay

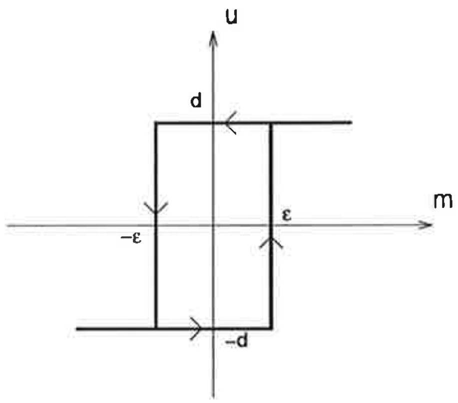

Figure 3. The relay with input m and output u is defined by its gain d and hysteresis ε. The arrows mark the allowed directions.

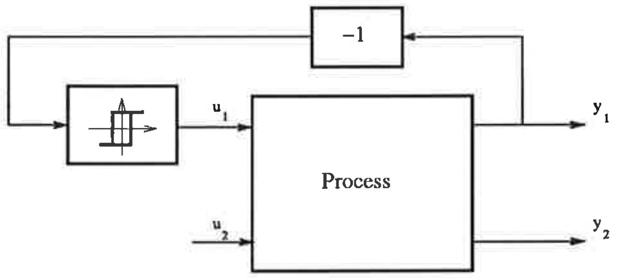

Figure 4. A block diagram for an *open*-loop relay experiment for the u_1–y_1-loop.

input, we will in our case have a describing function $N(a)$ given by

$$-\frac{1}{N(a)} = -\frac{\pi}{4d}\sqrt{a^2 - \varepsilon^2} - i\frac{\pi\varepsilon}{4d}$$

We notice that for a given relay gain and hysteresis, the reciprocal of the negative describing function will be a line parallel to the real axis, see Figure 5. For the special case of no hysteresis ($\varepsilon = 0$) this line will be the negative real axis.

The ultimate gain K_u and the ultimate frequency ω_u are defined by the equations

$$K_u = \frac{1}{|G(i\omega_u)|} \qquad \arg G(i\omega_u) = -\pi$$

K_u and ω_u will be given by an experiment with a relay without hysteresis; this will give the intersection marked '1' in Figure 5. If we choose the hysteresis $\varepsilon > 0$, e.g. because there is noise at the relay input, we will have an intersection that is below the negative real axis. This case is marked '2' in the figure. If ε is small, we can use hysteresis in the relay and let the given gain and frequency be a good approximation for K_u and ω_u.

PI-design

A design procedure and the results from applying it will now be discussed. Let us consider the system given by Equation (1) and Figure 1. The ultimate gain K_u and the ultimate frequency ω_u for the u_1–y_1-loop can be determined in the way illustrated in Figure 4. This is an open-loop experiment. Out of

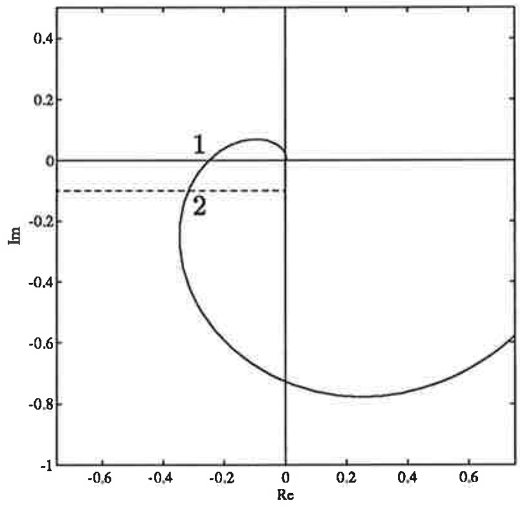

Figure 5. The Nyquist curve is interconnected with two describing functions, one is dashed and one is the negative real axis. The intersection is marked '1' for the case of no hysteresis and '2' for positive hysteresis.

K_u and ω_u, we can derive the parameters of the PI-controllers based upon Ziegler-Nichols choice, i.e.

$$K_c = 0.4K_u \qquad\qquad T_i = 0.8T_u = 0.8\frac{2\pi}{\omega_u}$$

There are four different loops to close by the relay. These are given by the pairs u_1–y_1, u_1–y_2, u_2–y_1, and u_2–y_2. Thus we can design four different PI-controllers. Further by using each of these PI-controllers, it is possible to do four closed-loop relay experiments. In Figure 6 we show an experiment, where we have used the PI-controller designed by an open-loop relay experiment in the u_2–y_2-loop. From experiments like the one shown in Figure 6, we can derive PI-controllers for the second loop. The connection in the figure will give a controller for the u_1–y_1-loop.

The algorithm for our PI-controller design based on relay experiments will be as follows.

1. Do the four possible open-loop relay experiments.
2. Derive the parameters for the PI-controllers given by the open-loop experiments.
3. Do the four possible closed-loop relay experiments, i.e. use a controller derived in 2 to close one loop while doing relay experiment in the other.
4. Derive new PI-parameters given by the closed-loop experiments.

It would be reasonable to iterate 3 and 4, if the algorithm converges.

Simulations

We will now try the algorithm given above on our system in Equation (1). In all our simulations we have used a discrete time relay with sampling period 0.01 s and hysteresis $\varepsilon = 0.01$. When we do an open-loop relay experiment for the u_1–y_1-loop, the typical relay oscillations appear as is shown in Figure 7.

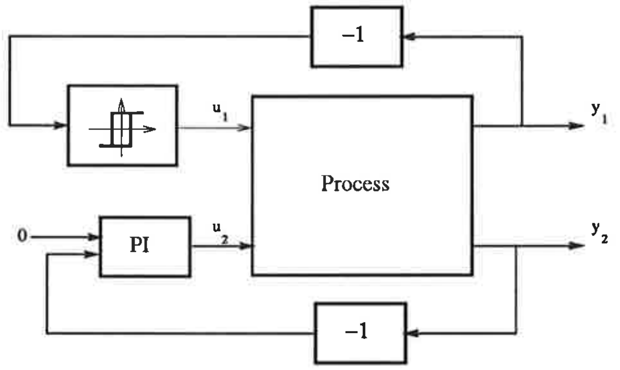

Figure 6. A block diagram for a *closed*-loop relay experiment for the u_1–y_1-loop. The u_2–y_2-loop is controlled by a PI-controller.

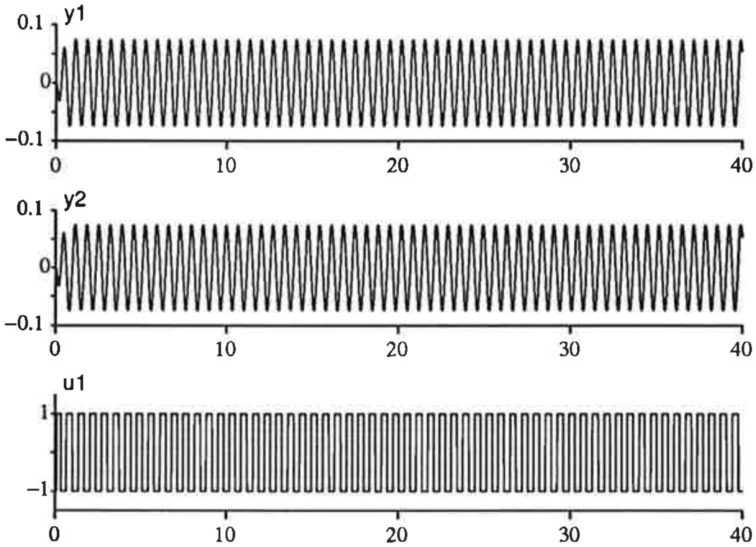

Figure 7. Results from the simulated relay experiment shown in Figure 4.

Out of these oscillations we can determine the ultimate gain K_u, the ultimate cross gain $K_u^{'}$ (the ultimate gain calculated for the opposite output), and the ultimate frequency ω_u for the u_1–y_1-loop. The results from the four possible open-loop relay experiments are given by the following table.

loop	K_u	$K_u^{'}$	ω_u
u_1–y_1	16.3	16.3	9.0
u_2–y_1	12.3	25.5	10.9
u_1–y_2	16.3	16.3	9.0
u_2–y_2	16.3	8.9	9.0

$K_u^{'}$ is not used in our design, but could be considered as a measurement of the cross-coupling. The values above will give the following PI-designs based on Ziegler-Nichols rules.

loop	K_c	T_i
u_1–y_1	6.5	0.56
u_2–y_1	4.9	0.46
u_1–y_2	6.5	0.56
u_2–y_2	6.5	0.56

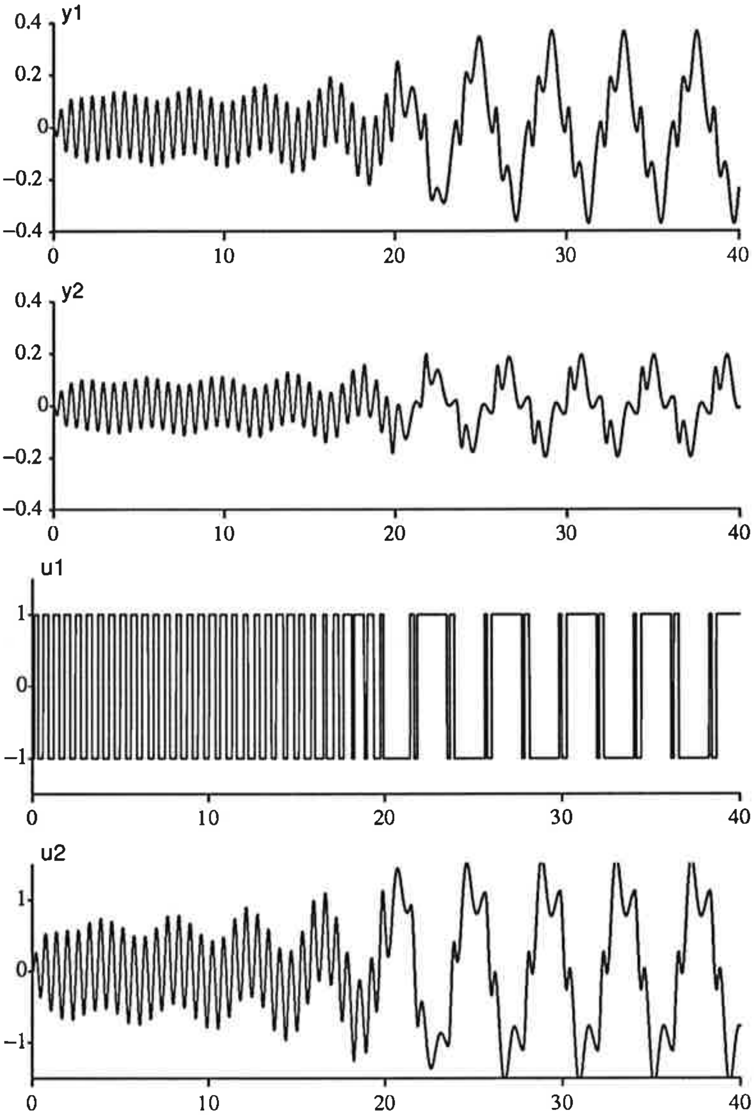

Figure 8. Results from the simulated relay experiment shown in Figure 6. We see that the signals are changing behaviour after about 15 s.

If we try to use two PI-controllers, e.g. connected as in Figure 1, and designed the way described above, we will get an unstable system. The design method has not taken care of the cross-couplings in the process.

We now use one of the PI-controllers above to control one process loop, while doing a closed-loop relay experiment in the other. Figure 6 illustrates such a relay experiment. We are interested in examining if it is possible to use this experiment to design controllers like in the open-loop case described above.

In Figure 8 the results from a simulation of a relay experiment in the u_1–y_1-loop while controlling the u_2–y_2 is shown. The output y_1 shows that we initially are having an almost stable sinusoidal oscillation, but after about 20 s it is changing its appearance drastically. The reason for this behaviour is that the transfer function u_1–y_1 has two complex conjugate poles with low damping. If we assume that the u_2–y_2-loop is perfectly controlled, the damping (ζ) of the two poles will be 0.18. Poorly damped poles give an oscillating step response. Since the relay output can be viewed as consecutive positive and

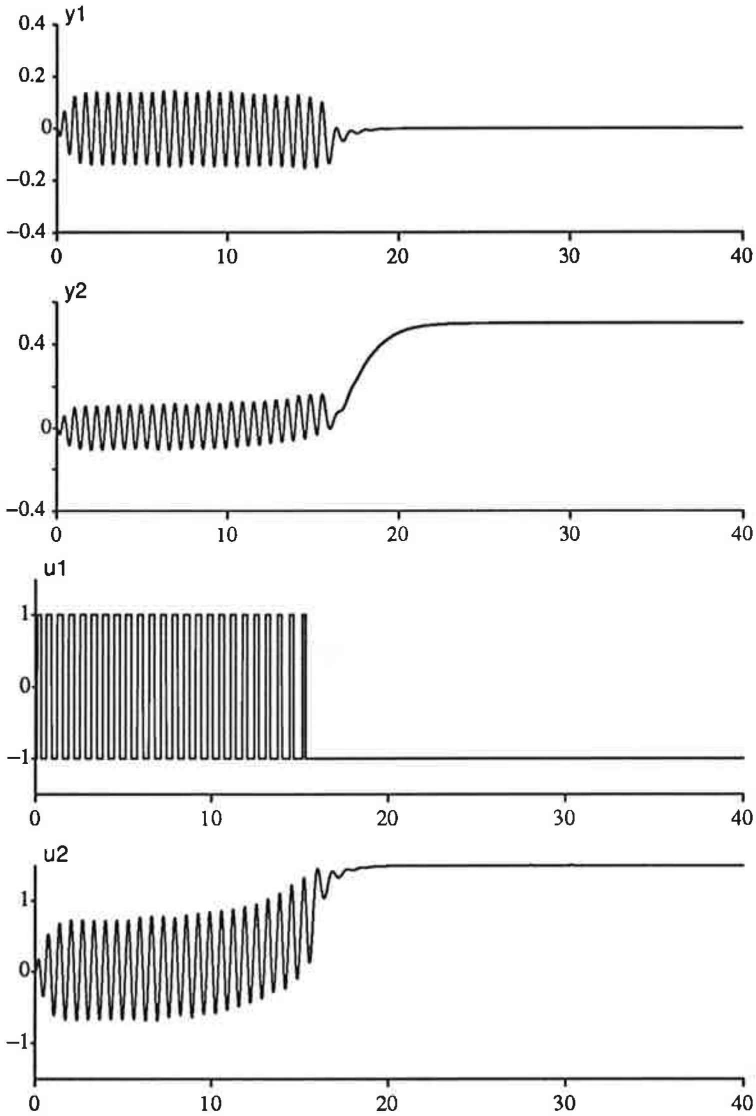

Figure 9. Results from a simulated relay experiment. We have closed the u_1–y_2-loop with a relay, while a PI-controller closes the u_2–y_1-loop. As in Figure 8 we can see that the signals are changing behaviour after about 20 s. This time the inputs and the outputs of the process reach constant steady-state values.

negative steps, the process output y_1 will for each step have an increasing oscillation. This will give the signals in Figure 8. The poorly damped poles are due to that the Ziegler-Nichols rules were used. A behaviour similar to Figure 8 is shown in [Vasnani and Loh, 1993] for a 4×4-system.

Another interesting behaviour appears when we do relay experiments in the u_1–y_2-loop while controlling the u_2–y_1-loop. In Figure 9 the results of a simulation are shown. The slowly drift of the process outputs, will drive the system in about 15 s into a state with constant outputs. Easy calculations for the steady-state system, i.e. considering $G(0)$, gives the same final values as in Figure 9.

The results given above from the closed-loop relay experiments are not easy to use for design based on the Ziegler-Nichols rules; we do not have a stable limit cycle. In the next section, we will give some explanations on why the relay oscillations above go unstable.

A reasonable way to avoid the strange behaviour of the closed-loop experiments is to use a more conservative tuning rule than Ziegler-Nichols. Then the controlled loop will affect the loop closed by the relay less, and we will have stable oscillations. A drawback with this approach is that the final system will be slow.

Summary

We cannot design a PI-controller for the process defined by Equation (1) using Ziegler-Nichols closed-loop method in the way described by the algorithm above. The cross coupling in the process $G(s)$ is too strong, so when we are doing closed-loop relay experiments, the controlled loop will influence the other loop too much. Hence, we cannot allow us to use relay design methods for SISO-systems in a straight forward way on a MIMO-system. There are however possibilities to use more conservative tuning rules, but these give a slow system.

4. Stability of Relay Oscillations

In this section we will use our process knowledge to show that the oscillations for the closed-loop experiment above will be unstable. The theory we will use are due to [Åström, 1993], but the basic theory is also included in [Åström and Wittenmark, 1989].

An Algorithm for Analyzing Stability of Relay Oscillations

Consider the system

$$\begin{aligned} \frac{dx}{dt} &= Ax + Bu \\ y &= Cx \end{aligned} \tag{2}$$

Let the system output be fed back via a relay with amplitude d and hysteresis ε. The control signal u will be given by

$$u(t) = \begin{cases} d & \text{if } \; e > \varepsilon \; \text{ or } \; e > -\varepsilon \; \text{ and } \; u(t-) = d \\ -d & \text{if } \; e < -\varepsilon \; \text{ or } \; e < \varepsilon \; \text{ and } \; u(t-) = -d \end{cases} \tag{3}$$

where $e = r - y = -y$, if the reference signal $r = 0$.

THEOREM 1
Consider the system (2) with the feedback law (3). Assume that there exists a periodic solution with period $2h$. Then the following conditions hold.

$$C(I + e^{Ah})^{-1} \int_0^h e^{As} ds B = \frac{\epsilon}{d} \tag{4}$$

$$y(t) = C(e^{At}x(0) - \int_0^t e^{As} ds B d) > -\epsilon \quad \text{for} \quad 0 \leq t < h$$

The periodic solution is the solution with the initial condition

$$a \triangleq x(0) = (I + e^{Ah})^{-1} \int_0^h e^{As} ds B d$$

□

A proof is given in [Åström, 1993].

Now define v as the velocity of the state vector at time $t = 0-$, i.e.

$$v = \frac{dx}{dt}(0) = Aa + Bd \tag{5}$$

Then we have the following theorem.

THEOREM 2

Consider the system (2) with the feedback (3). Assume that there is a symmetric periodic solution. Let a be the initial state that generates the periodic motion, and

$$W = (I - \frac{vC}{Cv})\Phi$$

where $\Phi = e^{Ah}$ and v is given by (5). The limit cycle is locally stable if and only if W has all its eigenvalues inside the unit disk. □

A proof is given in [Åström, 1993].

We define a function $f = f(h)$ as the left part of (4), thus

$$f(h) = C(I + e^{Ah})^{-1} \int_0^h e^{As} ds B \tag{6}$$

Then Theorem 1 and 2 give the following short algorithm for analyzing a system.

1. Find h such that $f(h) = \varepsilon/d$.
2. Compute a, v, and W and check the conditions

$$\frac{dy}{dt}(0) = Cv > 0$$
$$|\lambda(W)| < 1$$
$$y(t) > -\varepsilon \text{ for } 0 \leq t < h$$

Stability Analysis for the Considered Process

Now let us use this algorithm for our process. We will especially look at two different relay loops.

First we consider the open-loop relay experiment, when there is a relay in the u_1–y_1-loop. Since u_2 and y_2 do not affect the behaviour of the system, we can treat it as a SISO-system. We determine a state space realization (2) for this SISO-system and calculate f as in (6). The equation

$$f(h) = \varepsilon/d$$

where $\varepsilon = 0.01$ and $d = 1$, will have one solution: $h = 0.33$. This solution corresponds to an oscillation with period $T = 2h = 0.66$ s, which equals the period in the simulation in the previous section. Since

$$\frac{dy}{dt}(0) = Cv = 0.70 > 0$$
$$\lambda(W) = \{-0.24, 0, 0.00, 0.04, 0.04, 0.37, 0.72\}$$

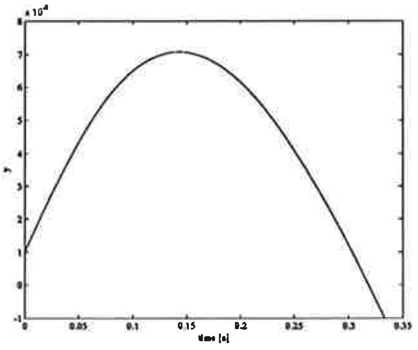

Figure 10. The condition $y(t) > -\varepsilon = 0.01$ for $0 \leq t < h = 0.33$ for stable oscillation is fulfilled.

and $y(t) > -\varepsilon = -0.01$ for $0 \leq t < h = 0.33$ (see Figure 10), the three conditions in Step 2 in the algorithm are fulfilled. Thus the oscillation is stable.

We now consider the closed-loop relay experiment when there is a relay in the u_1–y_1-loop, and the u_2–y_2-loop is controlled by a PI-controller designed as in the previous section. From the input u_1 to the output y_1, we can treat the system as a SISO-system and determine a state space realization as above. This time $f(h) = \varepsilon/d$ will have three solutions: $h_1 = 0.31$, $h_2 = 0.66$, and $h_3 = 2.02$.

$T_1 = 2h_1 = 0.62$ s is the period of the unstable oscillation in Figure 8. The eigenvalues of W are in this case

$$\{-0.53, 0, 0.00 \pm 0.03i, 0.03, 0.57, 0.95 \pm 0.48i\}$$

Thus there are two eigenvalues outside the unit disc, and the oscillation should be unstable.

For the solution $h_2 = 0.66$ the first condition in Step 2 in the algorithm is contradicted:

$$\frac{dy}{dt}(0) = -2.55 < 0$$

For the third solution all the three conditions in the algorithm are fulfilled. Hence there could be a stable oscillation with period $T_3 = 2h_3 = 4.04$ s. As is shown in Figure 8, the stable oscillation that appear after about 20 s has this period.

5. Modified SISO Approach

We will now turn to another approach to design the two PI-controllers in Figure 1. We will use a method that is described in [McAvoy, 1983]. We will in this section for simplicity, exemplify the idea of the method with a discussion around the design of a PI-controller for the u_1–y_1-loop.

The idea of McAvoy is to determine the ultimate gain K_u and the ultimate frequency ω_u for the open u_1–y_1-loop as we did in the previous section. Out of these values we calculate the ultimate gain and frequency for the interacting system K_{uI} and ω_{uI}, respectively. Further, K_{uI} and ω_{uI} are used to determine the parameters of the PI-controller in the u_1–y_1-loop via Ziegler-Nichols proposals, as in the previous sections. By this way, we will get a PI-controller that is better suited for the multivariable process than if we had used the original PI-controller determined by K_u and ω_u.

In the first subsection we will introduce the dynamic relative gain array, and in the second we will design PI-controllers and show the performance of the closed-loop system.

Dynamic Relative Gain Array

The calculation of K_{uI} and ω_{uI} are based upon a dynamic generalization of the relative gain array. It is common to define the relative gain array Λ as

$$\Lambda = G(0) \times (G(0)^{-1})^T$$

where $\times$ denotes the element-by-element product and G(0) the stationary gain of the process G, see e.g. [Åström *et al.*, 1990]. We define the dynamic relative gain array by

$$\Lambda_D = G(s) \times (G(s)^{-1})^T \tag{7}$$

Thus, for the process given by Equation (1), Λ_D can be written as

$$\Lambda_D = \begin{pmatrix} \lambda_{11}(s) & \lambda_{12}(s) \\ \lambda_{21}(s) & \lambda_{22}(s) \end{pmatrix} \tag{8}$$

or

$$\Lambda_D = \frac{1}{-s+1} \begin{pmatrix} s+3 & -2(s+1) \\ -2(s+1) & s+3 \end{pmatrix}$$

Hence

$$\Lambda_D \to \Lambda = \begin{pmatrix} 3 & -2 \\ -2 & 3 \end{pmatrix} \qquad \omega \to 0$$

and

$$\Lambda_D \to \begin{pmatrix} -1 & 2 \\ 2 & -1 \end{pmatrix} \qquad \omega \to \infty$$

The dynamic relative gain array matrix gives a hint of what control loops that should be chosen. A diagonal Λ_D is preferable. The different limit signs of the elements above indicate that the system might be hard to control; the behaviour of the system changes with the frequency.

PI-design

We will now develop a design method for the PI-controllers via the dynamic relative gain array. Figure 11 defines the considered system and illustrates the cross-couplings in the process.

Let us consider the u_1–y_1-loop when the u_2–y_2-loop is perfectly controlled, i.e.

$$Y_2(s) = g_{21}(s)U_1(s) + g_{22}(s)U_2(s) = 0$$

The characteristic equation for the u_1–y_1-loop will then be (if we exclude the transform argument s)

$$1 + g_{c1}g_{11} + g_{c2}g_{22} + g_{c1}g_{c2}(g_{11}g_{22} - g_{12}g_{21}) = 0$$

which can be written as

$$1 + g_{c1}g_{11} + g_{c2}g_{22} + \frac{g_{c1}g_{11}g_{c2}g_{22}}{\lambda_{11}} = 0 \tag{9}$$

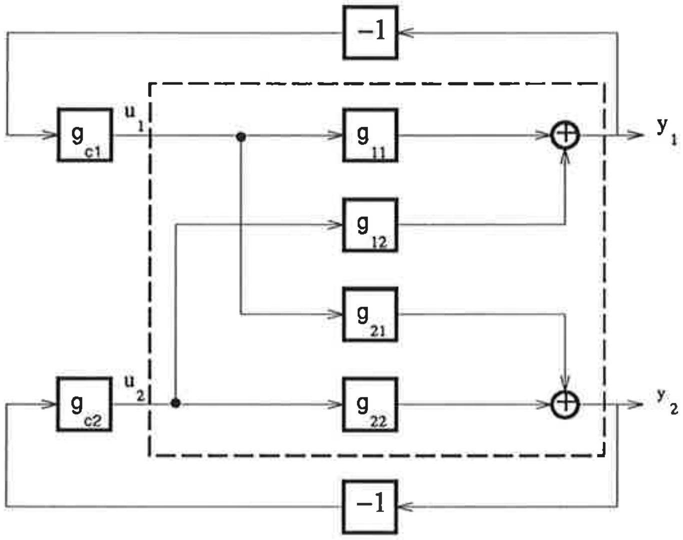

Figure 11. The four transfer functions g_{11}, g_{12}, g_{21}, and g_{22} characterize the coupling between the two inputs and the two outputs. The controllers are g_{c1} and g_{c2}.

We recall from Equation (1) that for our process $g_{11}(s) = g_{22}(s)$. To get a simple and approximate analysis we do the rough assumption that $g_{c1}(s) = g_{c2}(s)$, due to the almost symmetric process. Then we can rewrite Equation (9) as

$$(1+\frac{g_{c1}(s)g_{11}(s)}{\lambda_{11}(s)+\sqrt{\lambda_{11}^2(s)-\lambda_{11}(s)}})(1+\frac{g_{c1}(s)g_{11}(s)}{\lambda_{11}(s)-\sqrt{\lambda_{11}^2(s)-\lambda_{11}(s)}})=0$$

In the frequency band around the ultimate frequency for the interacting system ω_{uI}, it is reasonable to think of $\lambda_{11}(s)$ as a complex constant, i.e. $\lambda_{11}(s) = \lambda_{11}(i\omega_{uI})$. Furthermore, let us do the assumption that $\lambda_{11}(i\omega_{uI}) \approx \lambda_{11}(i\omega_u)$. Then theoretically the ultimate gain for the interacting system for the u_1–y_1-loop becomes

$$K_{uI} = \left|\frac{\lambda_{11}(i\omega_{uI})+\sqrt{\lambda_{11}^2(i\omega_{uI})-\lambda_{11}(i\omega_{uI})}}{g_{11}(i\omega_{uI})}\right| \approx \frac{0.42}{|g_{11}(i\omega_{uI})|}$$

where the ultimate frequency for the interacting system is given by

$$\arg\left\{\frac{g_{11}(i\omega_{uI})}{\lambda_{11}(i\omega_{uI})+\sqrt{\lambda_{11}^2(i\omega_{uI})-\lambda_{11}(i\omega_{uI})}}\right\} = -\pi$$

Since for our process

$$\arg\{\lambda_{11}(i\omega_{uI})+\sqrt{\lambda_{11}^2(i\omega_{uI})-\lambda_{11}(i\omega_{uI})}\} \approx 0.052 \approx 0$$

we have approximately

$$\arg\{g_{11}(\omega_{uI})\} \approx -\pi \qquad \Rightarrow \qquad \omega_{uI} \approx \omega_u$$

From the calculations above, we finally get the following rough proposals

$$K_{uI} = 0.42K_u \qquad\qquad \omega_{uI} = \omega_u$$

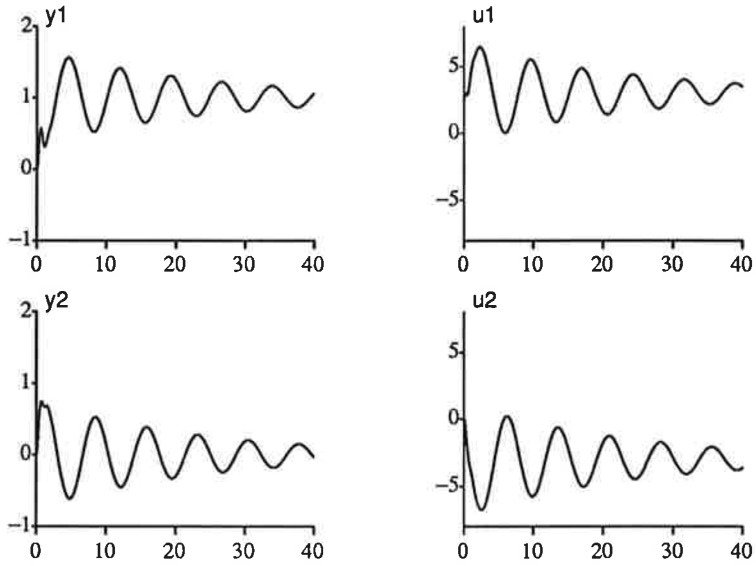

Figure 12. The step responses from u_{c1} to y_1 and y_2, together with the control signals u_1 and u_2, when we used the design method following McAvoy's idea. We notice that the step responses are oscillating a lot.

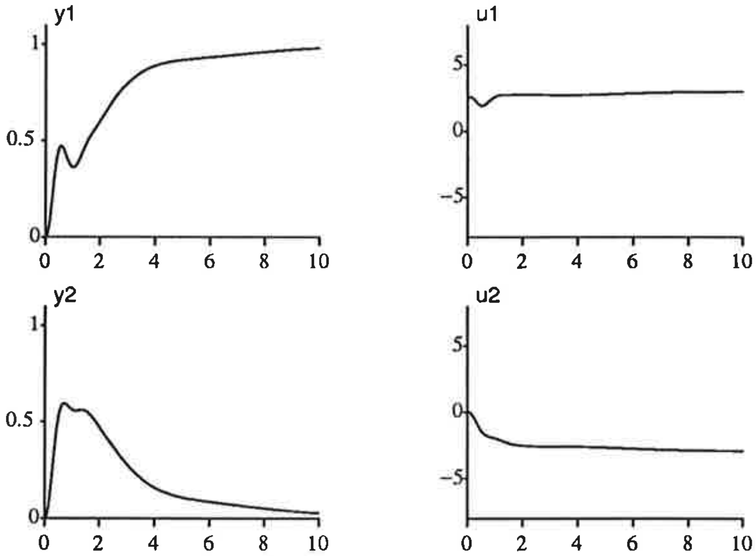

Figure 13. The step responses from u_{c1} to y_1 and y_2, together with the control signals u_1 and u_2. The circumstances are the same as in Figure 12, but the integrator times (T_i) are tripled in the PI-controllers. (Notice that the time axes are different in Figure 12.)

We now use these proposals to design a PI-controller for the u_1–y_1-loop and repeat the design procedure for the u_2–y_2-loop. This will lead to two PI-controllers, both with $K_c = 2.46$ and $T_i = 0.56$. The step responses from u_{c1} to y_1 and y_2 are shown in Figure 12 together with the control signals u_1 and u_2. We notice that the step responses are oscillating.

An interesting observation is illustrated in Figure 13, where we have tripled the integration time T_i in both PI-controllers. The step responses are then much better and about as fast as those of the open loop system, see Figure 2, but the steady state values of y_1 and y_2 are the desired ones in Figure 13.

Summary

The idea by McAvoy to calculate the two controller design parameters, the ultimate gain and the ultimate frequency for the interacting system, out of the open loop system feels intuitively correct. However, these PI-designs are not satisfactory for our process. This is due to the large cross-coupling in the process, which gives too large ultimate frequencies and thus to small integrator times T_i.

6. Conclusions

We have tried to design PI-controllers for a 2×2-process in two different ways, both based on Ziegler-Nichols closed-loop method. We have also done some analysis of the stability of relay oscillations.

First we considered the idea of doing closed-loop relay experiments, i.e. doing relay experiment in one loop while controlling the other. This gave outputs of the system that we could not use for a design based on Ziegler-Nichols closed-loop method. We did some analysis, which stated that the relay oscillations for our process would have the observed behaviour.

As another approach for the PI-design, we made an analysis due to McAvoy [McAvoy, 1983]. Probably because of the rough assumptions made, e.g. that the controllers are identical, this design was not acceptable. We made a small modification to this design, to show that it is possible to control the process.

7. References

Åström, K. J. (1971): *Olinjära system.* Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Åström, K. J. (1983): "Reglerteknik — en elementär introduktion, kapitel 6," (Control engineering—an elementary introduction, chapter 6). Report TFRT-3171, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Åström, K. J. (1993):. "Oscillations in relay systems.". To appear.

Åström, K. J. and T. Hägglund (1984): "Automatic tuning of simple regulators with specifications on phase and amplitude margins." *Automatica*, **20:5**, pp. 645–651.

Åström, K. J. and B. Wittenmark (1989): *Adaptive Control.* Addison-Wesley Publishing Company.

Åström, K. J., B. Wittenmark, and S. Bay Jørgensen (1990): *Process Control.* Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Holmberg, U. (1991): *Relay Feedback of Simple Systems.* PhD thesis, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

McAvoy, T. (1983): *Interaction Analysis: Priciples and Applications.* Instrument Society of America.

Vasnani, V. U. and A. P. Loh. (1993):. "Auto-tuning of multi-loop PI controllers for multivariable plants.". Submitted to IFAC 1993.

Zhuang, M. (1992): "PID controller design for a TITO system." Technical Report CE/S/61, School of Engineering, University of Sussex.

Zhuang, M. and D. Atherton (1992): "PID controller design for a TITO system." In *American Control Conference 1992.*