Industrial test setup for autotuning of PID controllers in large-scale processes: Applied to Tennessee Eastman process ⋆

Esmaeil Jahanshahi*, Selvanathan Sivalingam, John B. Schofield

Research & Development - Subsea, Siemens AS, Bratsbergvegen 5, 7037 Trondheim, Norway (E-mail: esmaeil.jahanshahi@siemens.com).

Abstract:
Although many PID tuning approaches are available, it is not easy to find a method that does not require any engineer/operator interference. In this work, we present a fully automated approach for PID tuning based on relay feedback. This method involves sending the relay feedback test data from PLCs (Programmable Logic Controller) into a historian, analyzing the test data using a tuning application to generate a tuning report that contains PID parameters and sending the report back to the operator station to retune the controllers in PLCs.
This paper is focused on the following three keys steps: 1) A method to identify persistent steady-state conditions in a control loop using routine operating data because any tuning test is performed when the process is operating at steady state, 2) A novel procedure to implement relay based tuning test, 3) A new model identification method which is a combination of frequency-domain and time-domain analysis. Subsequently, the identified plant model is used to obtain PID tuning parameters based on IMC design.
The approach has been tested on an industrial test setup in which all the control loops of the Tennessee Eastman process are controlled by a Siemens PLC. The necessary relay parameters, the hysteresis and relay amplitude, for the test are estimated automatically where interference by an engineer or an operator is not required. The new method for model identification is robust against measurement noises. The proposed method is able to tune the important control loops in the Tennessee Eastman process successfully.

Keywords: Auto-tuning, process performance, industrial process control

1. INTRODUCTION

A typical industrial plant has hundreds of control loops where 90% of the loops are controlled by PID controllers (Desborough and Miller, 2002). These controllers have to be tuned individually to match process dynamics in order to provide good control performance. Although the heuristic approaches by control engineers have been proven adequate for a large number of control loops, the manual tuning methods are very cumbersome and time consuming in particular, for those plants with slow responses. Also, the improvement in control loop performance mainly depends on the experience and the process knowledge of personnel. It is a well-known fact that many industrial control loops are poorly tuned by trial and error where performance of the control loops is not taken into consideration. Hence, the tuning methods without human interference draw more and more attention of the researchers and practising engineers.

Industrial experience has clearly indicated that it is highly desirable to have an push-button option on the Human Machine Interface (HMI) to put a control loop in tune mode to obtain PID tuning values. Earlier authors proposed different autotuning methods which have great practical values. However, they all suffer from some major limitations that are explained well by Hang et al. (2002). For example, the Cohen-Coon method (Cohen and Coon, 1953) requires an open-loop test on the process and is thus inconvenient to apply. The disadvantage of the closed-loop step method by Yuwana and Seborg (1982) and the Bristol method is the need of large setpoint change to trigger the tuning which may drive the process away from the operating point. To overcome these disadvantages, Åström and Hägglund (1984) proposed an automatic tuning controller that was based on the relay feedback technique. This method soon became a superior alternative to the conventional tuning.

A huge progress has been made in the last three decades in the area of auto tuning of PID controllers. Majority of the progress is in line with or variations of the method suggested by Åström and Hägglund (1984). Luyben (2002) summarized the applications and extensions of auto-tuning method and proposed the Auto-Tuning Variation (ATV) method. In the ATV method, the ultimate frequency and the ultimate gain which represent the most important process information can be directly extracted

⋆ Funding for this research was provided by SIEMENS AS
* Corresponding author

using a describing function. The recent studies on auto-tuning of PID controllers are reported by Leva (2007).

The auto-tuning method based on relay feedback is used widely in process industries due to its ease of implementation (Blevins and Nixon, 2011). This technique has several advantages over other methods, for instance it is time-saving and easy to use. The method is carried out under closed-loop control and, with an appropriate choice of the relay parameters, the process output can be kept close to the set point. This maintains the operating point in the linear region where the frequency response is useful, hence the method works well on highly nonlinear processes (Åström and Hägglund (1988)). The method is also extended to be applicable in presence of disturbances in the process (Hang et al. (1993)).

The relay-based tuning method provides two pieces of information, namely ultimate gain and ultimate period which are used by for Ziegler-Nichols PI and PID tuning rules . However, these tuning rules do not provide a good trade-off between robustness and performance of control loops. On the other hand, PI and PID tuning rules based on IMC (Internal Model Control) design are preferred because of their Pareto-optimality between performance and robustness. The tuning test based on relay feedback can easily be automated which we use for the model identification.

Although relay based auto-tuning has so many advantages, estimating the relay parameters to initiate the tuning procedure in plants is not straightforward. Moreover, tuning procedure based on relay method needs to be initiated in a control loop at steady state conditions to obtain ultimate frequency and ultimate gain accurately. Generally, in plants, visual inspection of trend plots of controller output and process output is a way of judging whether the control loop is in steady state or not. This is not easy when the process variables are affected by too much measurement noise or other sources. These two issues are addressed in this article. We have developed a method to find the presence of consistent steady state conditions in control loops. In addition, we have fully automated relay based tuning procedure by developing methods to estimate the necessary parameters.

We have implemented our Relay Tuning Function Block (RTFB) in the Simatic PCS7 environment. RTFB detects consistent steady-state, estimates the required relay amplitude and hysteresis and performs the relay-feedback test by a command from HMI. Simba Profibus is used for communication between the PLC and the Tennessee Eastman process model (Ricker (2002)) running in MATLAB. Aspen InfoPlus21 is used as the historian in the PIMAQ® (Plant Information Management and Data Acquistion) framework, and the PIMAQ tuning application analyses the test data to obtain the tuning values. The PIMAQ framework is one of the in-house products of Siemens Oil and Gas Solutions.

This paper is organized as the following. The technical description of RTFB and PIMAQ framework are described in Section 2. The identification method and the tuning rules are introduced in Section 3. The Tennessee Eastman process is briefly described in Section 4. Results are presented and discussed in Section 5, and concluding remarks are given in Section 6.

2. INDUSTRIAL AUTO-TUNING SETUP

In order to develop a fully automatic tuning procedure based on relay method, we need the following as has been explained earlier.

(1) a method to detect an existence of a consistent steady state behaviour in a loop
(2) a procedure to estimate necessary relay parameters
(3) an automatic way of selecting the controller (PID or PI) and the tuning method (*e.g.* Ziegler-Nichols, IMC based or SIMC method)

All the three issues are addressed below.

2.1 Implementation of RTFB

RTFB (Relay Tuning Function Block) is implemented in the PCS7 Simatic environment (using SCL programming language). RTFB is connected in series with the existing PID controller (CA Function Block). All function blocks are compiled, then binaries are executed in PLCs. The main algorithms implemented in the RTFB are as follows.

Detection of a consistent steady state behavior: In general, the visual inspection of process variables in trend plots is a way to identify the presence of steady state conditions. However, the steady state detection in control loops of process plants can be automated by comparing process output with set-point based on the student's t-test (Narasimhan and Jordache (1999)).

A simple algorithm is proposed in this paper to detect the presence of a consistent steady state without using the set-point information. The proposed algorithm is based on the basic definition of the steady state: Steady state is defined as the state of a system when it becomes settled (*If derivative of a quantity with respect to time is zero, the quantity is said to be at steady state*). However, in the reality it is not possible to have a constant process variable in a control loop due to noise.

The steady state detection algorithm in this paper is based on three indices, I_1, I_2, and I_3 which are estimated using process output when the control loop is in the routine operation. These indices must satisfy specific conditions in order to conclude that the control loop is in consistent steady state. The consistent steady state behaviour here refers to the situation in which the process remains in the steady state for a considerable time duration (*e.g* at least 30 sec in chemical process plants).

The process output is denoted by Y here. The main steps involved in the method are as follows.

(1) Estimate recursive mean of Y (low pass filer), Y_μ
(2) Estimate windowed mean of Y, Y_{win}
(3) Calculate VY $= (Y - Y_{win})^2$
(4) Estimate recursive mean of VY, σ_Y^2
(5) Calculate standard deviation of Y, σ_Y
(6) Calculate upper and lower limits of Y using
$Y_{UL} = Y_\mu + 0.5\sigma_Y$
$Y_{LL} = Y_\mu - 0.5\sigma_Y$

(7) If Y_{win} is between Y_{UL} and Y_{LL}, generate an index, $I_1 = 1$, else $I_1 = 0$ (This means that the process has not reached steady state) .
(8) Generate second index I_2 by calculating the absolute value of the time derivative of I_1,
$$I_2 = |\Delta I_1/\Delta t| = |(I_{1,k} - I_{1,k-1})/\Delta t|$$
where t is time.
(9) Generate third index I_3 by calculating the windowed integral of I_2 (window length N_w)
$$I_{3,k} = \sum_{j=k-N_w}^{k} I_{2,j}$$
(10) Consistent steady state exists if $I_3 = 0$ and $I_1 = 1$.

The advantages of the method are as follows: (1) it does not require set point information, (2) the method works well even in the presence of offset where the process output never reaches the set-point, (3) the method detects consistent steady state behaviour, (4) since the method does not use setpoint, it can be used for open-loop systems as well and (5) it is mathematically simple.

Relay hysteresis (PHYS): The relay test is sensitive to the measurement noise, and may relay chattering happen because of the noise. The easiest way to reduce the influence of the noise is to use relay with a hysteresis. Although, a small value for the hysteresis is chosen so that it does not affect the process response, the hysteresis value should be related to amount of the noise.

In the steady-state detection algorithm in above, we estimate standard deviation of the process output σ_Y. When the process is at the steady-sate, σ_Y is the noise standard deviation. We set the relay hysteresis as
$$PHYS = 5\sigma_Y \tag{1}$$

Relay amplitude (PRAMP): Tiny oscillations in the process output are expected when performing the relay test. More precisely, the process output y should not vary more than 10% of y_{ss} where y_{ss} is the steady-state value just before staring the relay test. In addition, the relay output cannot go beyond the controller saturation values (u_{max} and u_{min}). Therefore, the relay amplitude should not be too large. On the other hand, the relay amplitude must be specified large enough such that the process output not be trapped inside the hysteresis, especially for the processes with small gain or long time constant. For this, the inverse gain of the process (G^{-1}) is estimated by RTFB and used to calculate a relay amplitude that produces not larger than 10% variation in process output. However, the variation is needed to be larger than the noise level of the process output.

PRAMP is estimated using the following expression:
$$PRAMP = \min(\Delta r_1, \Delta r_2) \tag{2}$$
where

$$\Delta r_1 = 0.9 \times \min(u_{ss} - u_{min}, u_{max} - u_{ss}) \tag{3}$$
$$\Delta r_2 = 0.1 \times y_{ss} G^{-1} \tag{4}$$

Here, u_{ss} and y_{ss} are the controller and the process outputs at steady state respectively. u_{max} and u_{min} are maximum and minimum controller outputs set in PID algorithm (also known as input constraints).

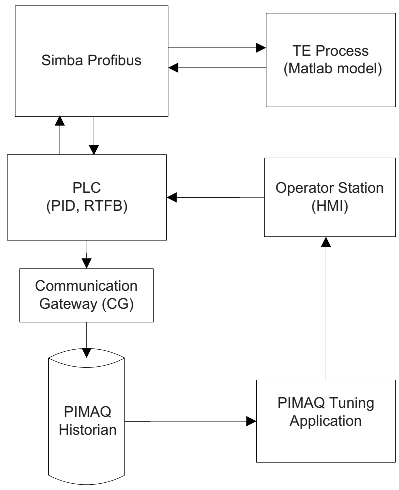

Fig. 1. Information flow in the test setup

Disturbance rejection during the relay test: Although relay feedback test is less sensitive to an external disturbance, the errors in the estimates of ultimate gain and ultimate frequency grow as the magnitude of the load change (disturbance) increases (Yu (2006)). In order to reject load disturbances during the relay-test, we add an integral action to the relay output. The integral time constant for this integral is specified in number of the relay cycles. This parameter (PTINT) specified based on the user experience; we have used PTINT = 2000.

2.2 PIMAQ Framework

PIMAQ® (Plant Information Management and Data Acquisition) developed by Siemens, Norway is a part of the industrial IT solutions delivered for oil & gas production. Since it logs all data from the PLC's and HMI screens, it can be used for analysis, production reporting and condition monitoring events. It is also a central component of Integrated Operations solutions.

We have developed our auto-tuning setup based on this framework; the main components of the setup and the information flow are illustrated in Fig. 1. The main steps are as the following:

(1) The necessary relay parameters are either calculated by the logic or set by operator.
(2) The chosen control loop is put into tune mode after necessary conditions (*e.g.* steady-state) are verified by the function block (RTFB). The procedure will be stopped automatically if (a) controller constraints are violated (b) experiment time exceeds the maximum time set for tuning and (c) the operator sends a command to stop the tuning.
(3) The communication gateway (CG) finds the signals that are needed to be logged and sends them to the PIMAQ historian.
(4) The tuning application developed in PIMAQ framework retrieves the data form the historian and evaluates tuning parameters based on the specified controller structure and the tuning method. The default controller structure varies based on type of the control loop (*i.e.* temperature, level, flow or pressure control). If it is a temperature control loop the default

controller structure is set as PID. For other types of control loops, the default controller structure is PI.

(5) A tuning report that contains calculated tuning values is sent back to the historian from where it is sent to the HMI. The tuning values can now be tested in function block that contains PID algorithm in test mode for some time and can then be accepted by the operator if satisfied.

In the test setup presented in this paper, the PLC (where PID controller and RTFB are running) communicates with the Tennessee Eastman process I/Os using Simba profibus. Simba Profibus is a hardware interface that simulates inputs and outputs for the PLC and supports up to 125 profibus slaves in real-time. In the automation industry, various protocols such as HART, Foundation Fieldbus or PROFINET may be used for the communication between field devices and PLCs.

3. IDENTIFICATION AND TUNING

3.1 Identification

Mainly, two pieces of information, ultimate period and ultimate gain, are directly estimated from relay feedback test data. Conventionally, the ultimate period P_u is found by finding crossings of the process output and the setpoint in the time domain. Then, the ultimate or critical frequency ω_c is found as

$$\omega_c = \frac{2\pi}{P_u}. \tag{5}$$

Instead, we use a frequency domain analysis which is robust against measurement noises. We exploit the fact that ω_c is the fundamental frequency of the signal which contains the peak power. We use Welch's averaged, modified periodogram method (Welch, 1967) to estimate the Power Spectral Density (PSD) of the relay test data. Then, we take the frequency with the peak power as ω_c. The Power Spectrum method requires enough number of relay oscillations (approximately 10 to 15 cycles) to produce an accurate estimate of the ultimate frequency.

The ultimate gain is calculated by

$$K_u = \frac{4h}{\pi a}, \tag{6}$$

where h is the relay amplitude and a is the response amplitude. Instead of h and a, we use square root of average powers of the input and output.

The ultimate values (K_u and ω_u) are enough for tuning the PID controllers using Ziegler-Nichols or modified Ziegler-Nichols equations. In addition, the relay test data can be used for closed-loop process model identification. The model identification from the relay feedback has attracted significantly increasing attentions in the process control community. The model identification procedure used in this work is described in the following.

Model Structures Most of the processes in chemical plants are approximated by first order plus time delay (FOPTD), second order plus time delay (SOPTD) and integrating plus time delay (IPTD) models. The obvious choice for the PI tuning purpose for stable processes is a first order plus time delay model

$$FOPTD : G_1(s) = \frac{K_p e^{-\theta s}}{\tau_1 s + 1}. \tag{7}$$

For the PID tuning it is beneficial to use a second order plus time delay model,

$$SOPTD : G_2(s) = \frac{K_p e^{-\theta s}}{\tau^2 s^2 + 2\zeta\tau s + 1}. \tag{8}$$

Many processes in the chemical industry and the oil production such as liquid level in a tank are ,

$$IPTD : G_3(s) = \frac{K_p e^{-\theta s}}{s}. \tag{9}$$

where K_p is static gain of the process, τ is the time constant, θ is the time delay and ζ is the damping ratio of the process response. Typically, response of a SOPTD process is classified into critical damping, over-damping or under-damping based on the ζ value.

Estimation of plant gain (K_p): K_p is estimated by using cross spectrum and power spectrum estimates in frequency domain.

$$K_p = \frac{P_{u,y}(\omega_0)}{P_{u,u}(\omega_0)} \tag{10}$$

where $P_{u,y}(\omega)$ is cross spectrum between controller output (u) and process output (y), $P_{u,u}(\omega_0)$ is power spectrum of u and ω_0 is zero. The estimation of K_p is independent of model order and structure. The method has been tested for different processes with known gains where good estimates of the process gains were achieved even in presence of measurement noise. K_p can be estimated for integrating systems easily by using the expression in Equation (11).

$$K_p = \frac{2\pi}{K_u P_u} \tag{11}$$

Estimation of time constant (τ): The time constant of the process is calculated similar to the approach by Wang et al. (2007) with a slight modification.

The time constant for FOPTD: The time constant for FOPTD when controlled by relay controller is derived as

$$\tau = \frac{\sqrt{K_p^2 k_1^2 - 1}}{\omega_c} \tag{12}$$

which is the equation originally used to estimate the time constant (Yu, 2006) with K_u replaced by k_1. The constant k_1 is calculated by Wang et al. (2007) as follows.

$$k_{2l-1} = \left|\frac{1}{G[j(2l-1)\omega_c]}\right| \tag{13}$$

$$k_{2l-1} = \frac{4h}{\omega_c(2l-1)\left|\int_T y(t)\exp(-j(2l-1)\omega_c t)dt\right|} \tag{14}$$

Computation of integral over one period in equation (14) is not accurate due to measurement noise. Hence, it is proposed to estimate the integral over the whole data length and then divide the integral by number of the periods to minimize the effect of noise in the data.

The time constants (τ_1 and τ_2) for SOPTD are estimated by calculating k_1 and k_3 from equation (14) and then solving the following system of equations.

$$K_p^2 k_1^2 - (1+\tau_1)(1+\tau_2) = 0 \tag{15}$$

$$K_p^2 k_3^2 - (1+9\tau_1)(1+9\tau_2) = 0 \tag{16}$$

Table 1. SIMC-PI tuning rules

Structure	G(s)	K_c	T_i
FOPTD	$\frac{K_p e^{-\theta s}}{\tau_1 s+1}$	$\frac{1}{K_p}\frac{\tau_1}{\tau_c+\theta}$	$\min\{\tau_1, 4(\tau_c+\theta)\}$
SOPTD	$\frac{K_p e^{-\theta s}}{(\tau_1 s+1)(\tau_2 s+1)}$	$\frac{1}{K_p}\frac{\tau_1}{\tau_c+\theta}$	$\min\{\tau_1, 4(\tau_c+\theta)\}$
IPTD	$\frac{K_p e^{-\theta s}}{s}$	$\frac{1}{K_p}\frac{1}{\tau_c+\theta}$	$\min\{\tau_1, 4(\tau_c+\theta)\}$

Two solutions for τ_1 and two solutions for τ_2 are found by solving this system of equations. For τ_1 the larger solution, and for τ_2 the smaller solution are the correct solutions.

Estimation of time-delay (θ): The time delay between u and y is estimated by using the famous cross correlation method with a slight modification. We use $\dot{u}$, the derivative of u, which gives series of spikes for positive and negative direction of relay ($-h$ and $+h$). The lag at which cross correlation between derivative $\dot{u}$ and y reaches maximum is the time delay between u and y. Using relay with hysteresis reduces relay chattering significantly during the test. In addition, we apply an algorithm to remove any possible chattering from the data. Estimation of time delay does not differ significantly between FOPTD and SOPTD systems for the identification based on relay feedback test. The modified cross-correlation method has been tested for different processes with known time delays and acceptable estimates of the time delays were obtained even with highly noisy measurements. Obviously, the time delay for integrating systems is simply $P_u/4$.

The selection of model structure is done based on shape factor analysis developed by Luyben (2001).

3.2 Tuning Rules

We consider the following structure for the PID controller.

$$K_{PID}(s) = K_c\left(1 + \frac{1}{sT_i} + \frac{T_d s}{T_f s + 1}\right), \tag{17}$$

where T_f is the time constant of the low-pass filter on the derivative action. It is usually chosen to be 10% of the derivative time T_d. Here, the low-pass filter is used only for the purpose of reducing the measurements noise effect. And the PI controller is as the following:

$$K_{PI}(s) = K_c\left(1 + \frac{1}{sT_i}\right) \tag{18}$$

Three types of tuning rules used in this work are SIMC-PI, IMC-PID and Modified Zeigler-Nichols PI. The Modified Zeigler-Nichols tuning rules (Yu, 2006) in equation (19) are the same regardless of the model structure.

$$K_c = K_u/3, \quad T_i = 2\times P_u, \tag{19}$$

where K_u and P_u are the ultimate gain and the ultimate period respectively. The Skogestad PI (SIMC) tuning rules (Skogestad and Grimholt, 2012) are summarized in Table 1.

In order to use the SIM-PI tuning rules for the SOPTD model, we need to use the half-rule to obtain a FOPTD model (Skogestad and Grimholt, 2012).

$$\tau_1 = \tau_{10} + \frac{\tau_{20}}{2}, \quad \theta = \theta_0 + \frac{\tau_{20}}{2}, \tag{20}$$

where τ_{10}, τ_{20} and θ_0 are the parameters of the SOPTD model. The IMC-PID tuning rules are given in Table 2 (Lee et al., 2006).

4. TENNESSEE EASTMAN PROCESS PLANT

The Tennessee Eastman (TE) process model which is a standard benchmark in the process control community is used in the test setup. The TE problem was introduced by Downs and Vogel (1993) in 1990. The process has eight components, including four reactants (A, C, D, and E), two products (G and H), an inert (B), and a byproduct (F).

The reactions are

A(g) + C(g) + D(g) $\longrightarrow$ G(I) (product)

A(g) + C(g) + E(g) $\longrightarrow$ H(I) (product)

A(g) + E(g) $\longrightarrow$ F(I) (byproduct)

3 D(g) $\longrightarrow$ 2F(I) (byproduct)

All the reactions are irreversible and exothermic. The reaction rates are a function of temperature through an Arrhenius expression. The reactions are approximately first-order with respect to the reactant concentrations.

The process has five major units: a reactor, a product condenser, a vapor-liquid separator, a recycle compressor, and a product stripper (see Fig. 2). There are 41 measurements and 12 manipulated variables. The process is explained in details by Downs and Vogel (1993).

The control objectives for this process are typical for a chemical process:

- Maintain process variables at desired values.
- Keep process operating conditions within equipment constraints
- Minimize variability of product rate and product quality during disturbances (stream 11).
- Minimize movement of valves which affect other processes
- Recover quickly and smoothly from disturbances, production rate changes or product mix changes.

From the goals listed by Ricker (1996), the following process variables need to be controlled:

(1) Production rate
(2) Mole % G in product
(3) Reactor pressure
(4) Reactor liquid level
(5) Separator liquid level
(6) Stripper liquid level

We use the control structure for Self-Optimizing Control by Larsson et al. (2001). This control structure contains 17 feedback control loops. Most of the control loops have been tuned by the automatic tuning methodology proposed in this article. The exceptions were composition control of Mole % of C in purge gas (Xmeas31) and the reactor pressure (Xmeas7). Due to very slow dynamics, the tuning takes more than 24 hours for these two loops, and the operating point changes in this time. The results for eight important control loops are discussed in the next section.

Table 2. IMC-PID tuning rules

Structure	G(s)	K_c	T_i	T_d
FOPTD	$\frac{K_p e^{-\theta s}}{\tau_1 s+1}$	$\frac{1}{K_p}\frac{T_i}{\tau_c+\theta}$	$\tau_1 + \frac{\theta^2}{2(\tau_c+\theta)}$	$\frac{\theta^2}{6(\tau_c+\theta)}\left[3-\frac{\theta}{T_i}\right]$
SOPTD	$\frac{K_p e^{-\theta s}}{(\tau^2 s^2+2\tau\zeta s+1)}$	$\frac{1}{K_p}\frac{T_i}{\tau_c+\theta}$	$2\zeta\tau_1 + \frac{\theta^2}{2(\tau_c+\theta)}$	$\frac{\tau^2-\frac{\theta^3}{6(\tau_c+\theta)}}{T_i} + \frac{\theta^2}{2(\tau_c+\theta)}$
IPTD	$\frac{K_p e^{-\theta s}}{s}$	$\frac{1}{K_p}\frac{1}{\tau_c+\theta}$	–	$\frac{\theta^2}{2(\tau_c+\theta)}$

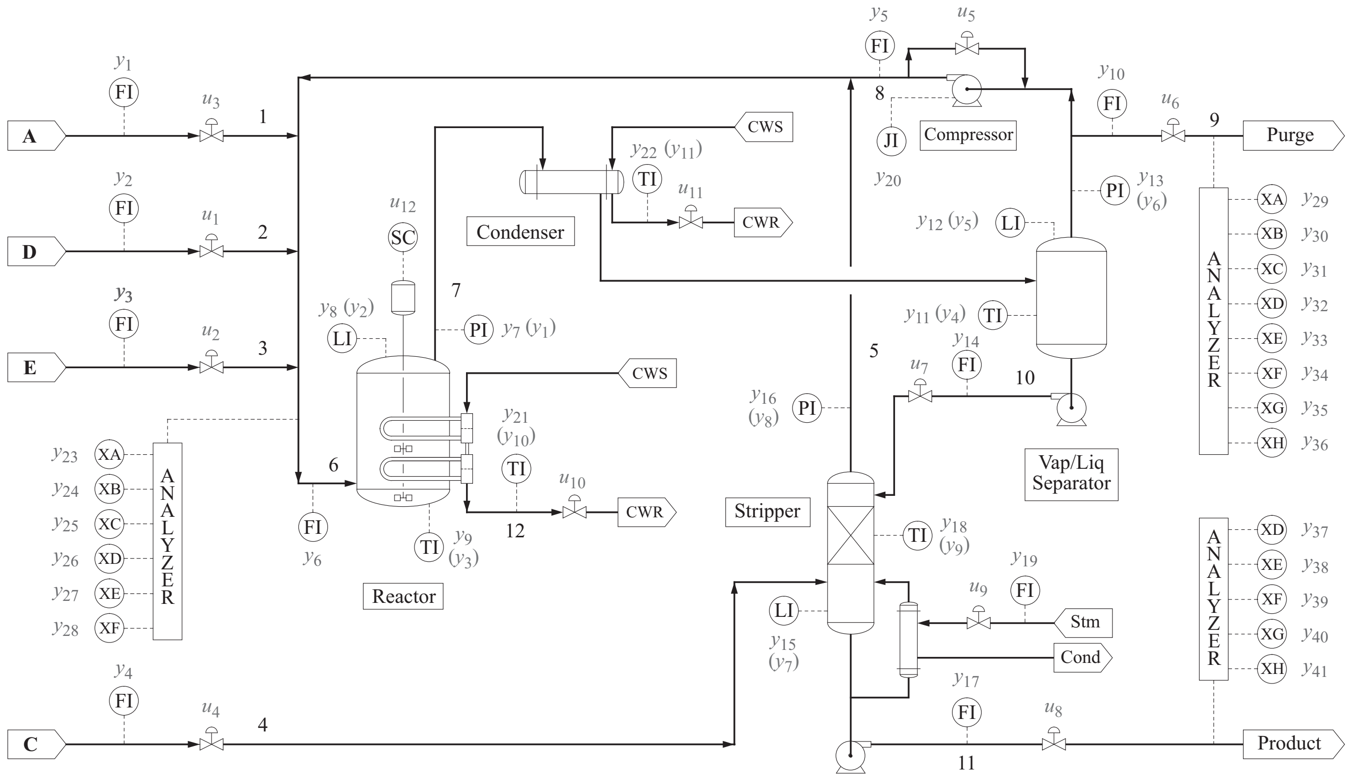

Fig. 2. Tennessee Eastman process

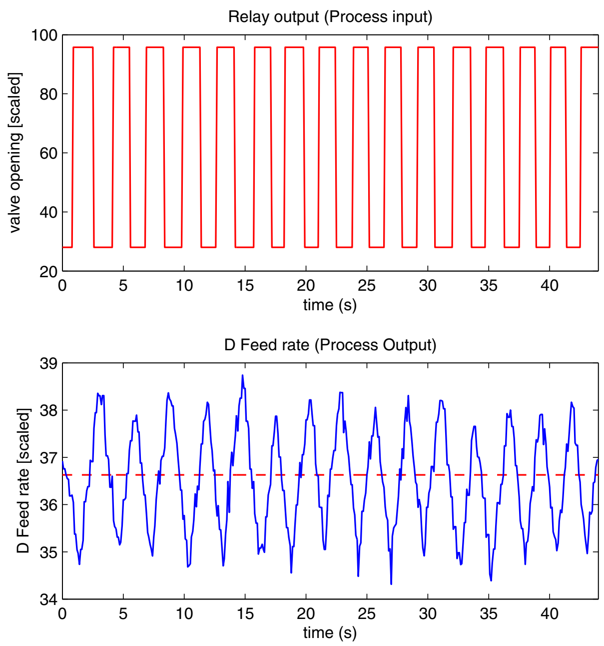

Fig. 3. Tuning data for D feed rate

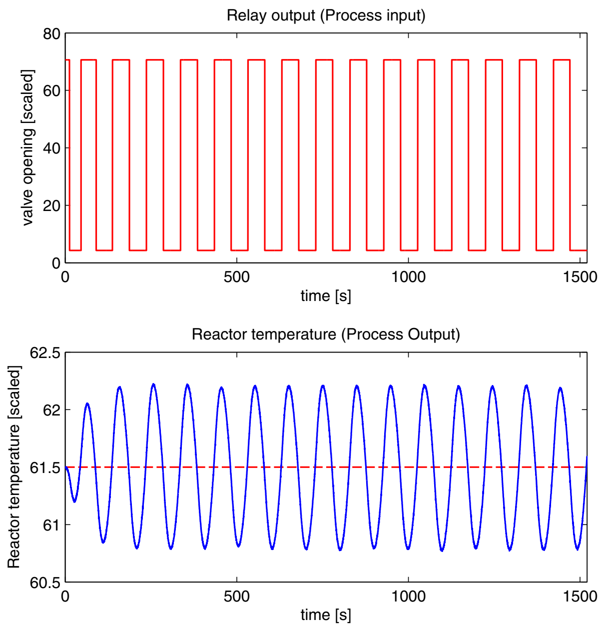

Fig. 4. Tuning data for Reactor temperature

Table 3. Control loops chosen for tuning

CVs	Scaling factor	MVs
A feed (xmeas1)	100	xmv3
D feed (xmeas2)	0.01	xmv1
E feed (xmeas3)	0.01	xmv2
C feed (xmeas4)	1	xmv4
Stripper underflow (xmeas17)	1	xmv7
Production rate (xmeas17)	1	Fp
Reactor temperature (xmeas9)	0.5	xmv10
Reactor level (xmeas8)	1	SP17

5. RESULTS AND DISCUSSION

We choose the important control loops in the Tennessee Eastman process for the tuning test. These controlled variables and the related manipulated variables are listed in Table 3. We have scaled some of the controlled variables such that their values become between 0 and 100. This scaling is necessary for the analogue input drivers in the control system. Also, by this scaling, very small or very large numbers for the tuning values are avoided.

Fig. 3 shows the tuning data from the flow control loop for the reactant D (xmeas2). The model identification algorithm uses shape of the relay response to identify the model structure as an integrating process,

$$G_3(s) = \frac{0.059e^{-0.67s}}{s}. \tag{21}$$

The identified models for the selected control loops and the tuning parameters using different tunings rules are given in Table 4. We have used $\tau_c = 2 \times \theta$ for SIMC-PI and IMC-PID tunings, whereas the Modified Ziegler-Nichols tuning rules do not require such a tuning parameter.

Fig. 4 shows the relay test data for the reactor temperature control loop (xmeas9). The relay response is very similar to a sine wave which suggests a SOPTD model. The model and the related tunings are provided in Table 4. The responses of the reactor temperature for a step change in the setpoint using different tunings are shown in Fig. 5. As shown in this figure the Ziegler-Nichols tuning gives the largest over-shoot. This is in agreement with the theoretical robustness measures such as sensitivity peaks and phase-margin which are given in the results table.

Fig. 6 shows response of the flow rate of D to a step in the setpoint. The IMC-PID tuning does not give an integral action for the integrating processes as we see in Table 4. This leads to an sustained offset in the step response as shown in Fig. 6.

We recommend to use PID settings for the slow loops (with long time delay or long time constant) such as temperature control loops. Having a SOPTD model is beneficial to obtain proper PID settings as we identified for the reactor temperature. However, for the integrating processes using the SIMC-PI tuning is recommended, because IMC-PID settings do not give an integral action. Any load disturbance may cause the process deviate from the desired operating point.

6. CONCLUSIONS

An industrial test setup has been developed as part of the work in the article which can be used for different

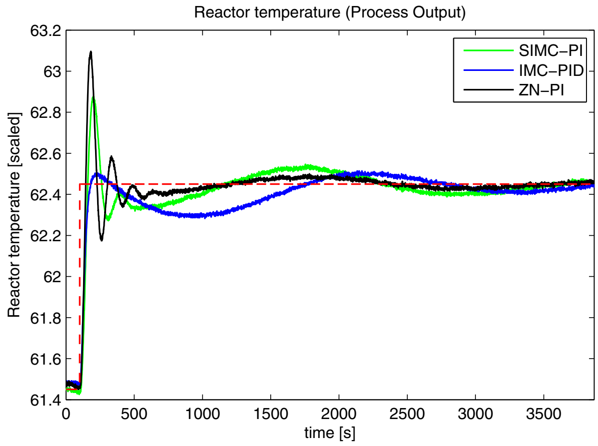

Fig. 5. Response of reactor temperature to a step in setpoint with different controllers given in Table 4

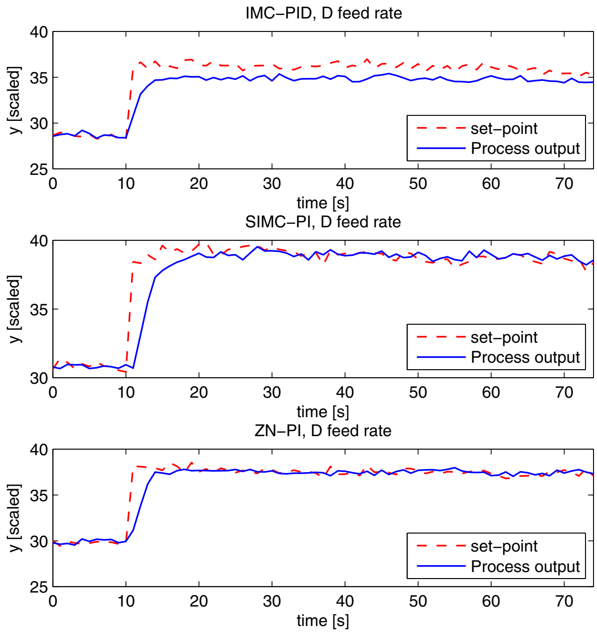

Fig. 6. Response of D feed rate to a step in setpoint with different controllers given in Table 4

purposes like fault diagnosis and controller performance assessment. A novel way of detecting a consistent steady state behaviour in control loops both for open- and closed loop scenario has been proposed. A procedure for estimating necessary relay parameters to make relay based autotuning method fully automatic without any manual intervention is presented. The procedure for estimating relay parameters and the method for detecting a steady state behaviour in control loops have been successfully tested using Tennessee Eastman standard problem. Methods for identifying model parameters from relay test data have been applied to Tennessee Eastman process and PID controllers are tuned based on the estimated models.

REFERENCES

Åström, K. and Hägglund, T. (1984). Automatic tuning of simple regulators with specifications on phase and amplitude margins. *Automatica*.

Table 4. Identified models and PI(D) tuning values for selected control loops in TE process

A feed	Tuning rule	K_c	T_i	T_d	$\|S\|$	$\|T\|$	GM	PM	DM	IAE
	SIMC-PI	4.33	6.96	–	1.39	1.22	4.54	56.69	1.67	30.39
$G(s) = \frac{0.13e^{-0.58s}}{s}$	IMC-PID	4.33	–	0.097	1.29	1.00	5.30	74.04	2.24	17.88
	ZN-PI	6.80	4.64	–	1.75	1.30	2.83	46.10	0.87	22.71
D feed	Tuning rule	K_c	T_i	T_d	$\|S\|$	$\|T\|$	GM	PM	DM	IAE
	SIMC-PI	8.84	8.05	–	1.39	1.22	4.53	56.65	1.93	40.82
$G(s) = \frac{0.059e^{-0.67s}}{s}$	IMC-PID	8.44	–	0.11	1.29	1.00	5.29	74.02	2.59	125.61
	ZN-PI	13.23	5.37	–	1.75	1.30	2.82	46.1	1.00	40.28
E feed	Tuning rule	K_c	T_i	T_d	$\|S\|$	$\|T\|$	GM	PM	DM	IAE
	SIMC-PI	4.19	11.35	–	1.39	1.21	4.53	56.67	2.72	49.23
$G(s) = \frac{0.084e^{-0.95s}}{s}$	IMC-PID	4.19	–	0.15	1.29	1.00	5.30	74.03	3.66	28.86
	ZN-PI	6.58	7.57	–	1.75	1.30	2.83	46.10	1.42	36.73
C feed	Tuning rule	K_c	T_i	T_d	$\|S\|$	$\|T\|$	GM	PM	DM	IAE
	SIMC-PI	24.5	12.25	–	1.39	1.22	4.53	56.66	2.94	53.12
$G(s) = \frac{0.013e^{-1.02s}}{s}$	IMC-PID	24.50	–	0.17	1.29	1.00	5.30	74.02	3.95	31.13
	ZN-PI	38.43	8.18	–	1.75	1.3	2.83	46.1	1.53	39.66
Stripper underflow	Tuning rule	K_c	T_i	T_d	$\|S\|$	$\|T\|$	GM	PM	DM	IAE
	SIMC-PI	-2.03	5.92	–	1.38	1.00	4.27	73.16	7.22	60.50
$G(s) = \frac{-0.4865e^{-2s}}{5.26s+1}$	IMC-PID	-1.92	5.59	0.29	1.29	1.00	5.27	73.94	7.72	60.51
	ZN-PI	-6.88	7.86	–	4.74	3.88	1.31	24.28	0.68	78.96
Fp (production index)	Tuning rule	K_c	T_i	T_d	$\|S\|$	$\|T\|$	GM	PM	DM	IAE
	SIMC-PI	5.09	72.25	–	1.39	1.22	4.54	56.69	17.36	309.70
$G(s) = \frac{0.01e^{-6.02s}}{s}$	IMC-PID	5.09	–	1.00	1.29	1.00	5.30	74.03	23.30	181.12
	ZN-PI	8.00	48.18	–	1.75	1.30	2.83	46.11	9.01	231.10
Reactor temperature	Tuning rule	K_c	T_i	T_d	$\|S\|$	$\|T\|$	GM	PM	DM	IAE
	SIMC-PI	-16.65	280.95	–	1.69	1.29	3.61	46.17	56.42	1323
$G(s) = \frac{-1.67e^{-20.20s}}{3716s^2+1947s+1}$	IMC-PID	-19.33	1950.28	22.41	1.42	1.00	3.83	72.95	74.80	606.5
	ZN-PI	-25.16	192.73	–	2.39	1.88	2.28	31.52	26.46	1172
Reactor level	Tuning rule	K_c	T_i	T_d	$\|S\|$	$\|T\|$	GM	PM	DM	IAE
	SIMC-PI	1.52	2365	–	1.37	1.00	4.47	71.33	1435	1.20×10^4
$G(s) = \frac{1.29e^{-403s}}{2231s+1}$	IMC-PID	1.47	2297.71	63.19	1.29	1.00	5.28	73.97	1556	1.21×10^4
	ZN-PI	2.48	3542	–	1.67	1.00	2.80	64.20	803.44	1.07×10^4

Åström, K. and Hägglund, T. (1988). Automatic tuning of pid controllers. *Instrument Society of America*.

Blevins, T. and Nixon, M. (2011). *Control Loop Foundation, Batch and Continuous Processes*. International Society of Automation, Durham, NC.

Cohen, G. and Coon, G. (1953). Theoretical consideration of retarded control. *Trans. ASME*, 75.

Desborough, L. and Miller, R. (2002). Increasing customer value of industrial control performance monitoring: Honeywell's experience. *Proc. AIChE Symp. Ser.*, 98, 153–186.

Downs, J. and Vogel, E. (1993). A plant-wide industrial process control problem. *Computers Chemical Engineering*.

Hang, C., Åström, K., and Ho, W. (1993). Relay autotuning in the presence of static load disturbance. *Automatica*.

Hang, C., Åström, K., and Wang, Q. (2002). Relay feedback auto-tuning of process controllers — a tutorial review. *Process Control*.

Larsson, T., Hestetun, K., Hovland, E., and Skogestad, S. (2001). Self-optimizing control of a large-scale plant: The tennessee eastman process. *Industrial & Engineering Chemistry Research*.

Lee, Y., Park, S., and Lee, M. (2006). Consider the generalized imc-pid method for pid controller tuning of time-delay processes. *Hydrocarbon Engineering*.

Leva, A. (2007). Comparative study of model-based pi(d) autotuning methods. In *American Control Conference*.

Luyben, W. (2001). Getting more information from relay feedback tests. *Industrial & Engineering Chemistry Research*.

Luyben, W. (2002). Derivation of transfer functions for highly nonlinear distillation columns. *Process Control*.

Narasimhan, S. and Jordache, C. (1999). *Data reconciliation and Gross Error Detection: An intelligent use of process data*. Data reconciliation and Gross Error Detection: An intelligent use of process data.

Ricker, N.L. (1996). Decentralized control of the tennessee eastman challenge process. *Process Control*.

Ricker, N.L. (2002). Tennessee eastman challenge archive. URL `http://depts.washington.edu/control/LARRY/TE/download.html`.

Skogestad, S. and Grimholt, C. (2012). The simc method for smooth pid controller tuning. In *PID Control in the Third Millennium*. Springer London.

Wang, P., Gu, D., and Zhang, W. (2007). Modified relay feedback identification based on describing function analysis. *Industrial & Engineering Chemistry Research*.

Welch, P.D. (1967). The use of fast fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms. *Audio and Electroacoustics, IEEE Transactions on*, 15(2), 70–73.

Yu, C.C. (2006). *Autotuning of PID Controllers - A Relay Feedback Approach*. Springer.

Yuwana, M. and Seborg, D.E. (1982). A new method for on-line controller tuning. *AIChE Journal*, 28(3), 434–440.

