An automatic tuner with short experiment and probabilistic plant parameterization

Soltesz, Kristian; Mercader, Pedro; Baños, Alfonso

Published in:
International Journal of Robust and Nonlinear Control

DOI:
10.1002/rnc.3640

2017

Document Version:
Peer reviewed version (aka post-print)

An automatic tuner with short experiment and probabilistic plant parameterization

Kristian Soltesz[1], Pedro Mercader[2] and Alfonso Baños[2]

[1]*Department of Automatic Control, Lund University, Lund, Sweden*
[2]*Department of Computer and Systems Engineering, University of Murcia, Murcia, Spain*

SUMMARY

A novel automatic tuning strategy is proposed. It is based on an experiment of very short duration, followed by simultaneous identification of LTI model parameters and an estimate of their error covariance. The parametric uncertainty model is subsequently exploited to design linear controllers with magnitude bounds on some closed-loop transfer function of interest, such as the sensitivity function. The method is demonstrated through industrially relevant examples. Robustness is enforced through probabilistic constraints on the $\mathcal{H}_\infty$ norms of the sensitivity function, while minimizing load disturbance integral error (IE) to ensure performance. To demonstrate the strength of the proposed method, identification for the mentioned examples is carried out under a high level of measurement noise.

KEY WORDS: Automatic tuning, robust identification, parametric uncertainty, uncertainty propagation

1. INTRODUCTION

1.1. An Industrial Perspective

A vast majority of closed-loop controllers in process industry are proportional and integrating (PI), and of the remainder most are proportional, integrating and derivative (PID). It is well-known that many controllers are poorly tuned, or running at their factory default parameters [1]. Both [2, 3] suggest that the percentage of adequately tuned controllers deceeds 50 %. In the end, this leads to significant revenue losses. To the authors' knowledge, there exist no recent reports, suggesting any improvement of the situation. Two contributing reasons to the current situation are that the resulting revenue losses are hard to estimate, and that it is expensive (in terms of man hours) to manually tune individual control loops. In [4], the average tuning price per loop is estimated to lie between USD 250 and USD 1000 in labor, while a representative process industrial factory typically has hundreds or thousands of such loops.

*Correspondence to: Kristian Soltesz, Department of Automatic Control, Lund University, P.O. Box 118, SE-221 00 Lund, Sweden. E-mail: kristian@control.lth.se, Phone: +46 46 222 87 77 , Fax: +46 46 13 81 18. Member of LCCC and ELLIIT research centra.

Contract/grant sponsor: Spanish government; contract/grant number: DPI2013-47100-C2-1-P, FPU12/01026

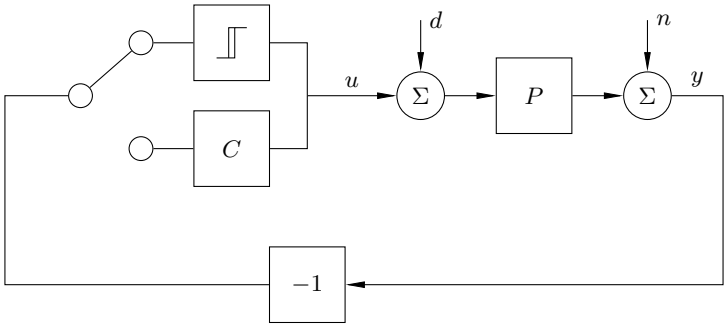
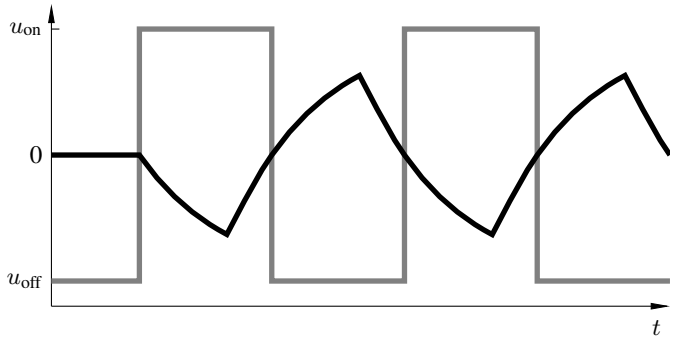
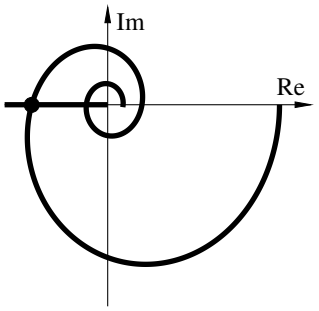


Figure 1. Block diagram of relay automatic tuner, with plant P, controller C, control signal u and plant output y. Also shown are the load disturbance d and measurement noise n.



(a) Control u (gray) and plant output y (black) from relay experiment.



(b) FOTD Nyquist curve, with obtained model (dot).

Figure 2. Signals of the classic relay experiment, and resulting model.

1.2. Automatic Tuners

The above section motivates the existence of automatic controller tuning procedures, which combine an experiment with subsequent system identification and controller tuning. The by far most known such procedure is the one introduced in [5]. The basic idea behind it, is to close a negative feedback loop over the plant to be controlled, in series with a relay nonlinearity (which can be well approximated by a P controller with high gain and control signal saturations), as shown in Figure 1. The inverse describing function of the relay intersects the plant Nyquist curve along the negative real axis, i.e., at a phase shift of $-180°$, as shown in Figure 2b. For classes of plants typically encountered in process industry, this results in stable limit cycle oscillation at a frequency critical for controller design, as illustrated in Figure 2a. In the original paper, and most industrial implementations, the experiment is terminated once a stable limit cycle has been reached. Subsequently, the fundamental oscillation frequency and an estimate of the gain at the same frequency are identified from control and measurement signal peak values and the limit cycle period. This yields a model consisting of the system response at the plant phase crossover frequency, marked with a dot in Figure 2b. The fact that only one frequency response point is identified is a major caveat, which has resulted in the development of several variants of the method. In [6] an integrator was connected in series with a second relay, to change the phase shift of the plant at which the limit cycle occurs. The method proposed in [7] utilizes the original experiment, but makes use of the identified frequency response

differently to obtain the PID parameters. Other modifications include extension to MIMO systems as covered by the survey [8].

The main strength common to all mentioned relay-based methods is that they produce an experiment with excitation focused to the frequency range most relevant for controller synthesis, being the third quadrant of the Nyquist plane. One significant weaknesses is their requirement for convergence to a stable limit cycle oscillation. Although this is achieved for most processes of industrial significance (see [9] for a discussion) the experiment requires a continuous stretch of time, during which the process must not be subject to disturbances. This makes application of the method impractical in several industrial settings.

The caveat of only identifying a single frequency response point was partially resolved in [10], by taking the shape of y into account. Another approach to address the same problem was taken in [11, 12], through the use of an asymmetric relay. Unlike its symmetric counterpart, the asymmetric relay experiment, with relay output levels $-u_{\text{off}} \neq u_{\text{on}}$, excites the process steady state. Once a stable limit cycle has been reached, this yields the possibility to identify parameters of a first order time delay (FOTD) model by only looking at switch durations and amplitudes. Formulae, based on describing function analysis, were presented in [12] and their practical application was demonstrated in [11]. However, analogous to the original formulation, as well as [10], only peak values and corresponding times are used in the identification. This makes all procedures presented thus far highly sensitive to measurement noise, even when the relay is equipped with a hysteresis to avoid chattering. (An adverse effect of introducing such a hysteresis is that it lowers the inverse describing function to a horizontal line below the negative real axis, see Figure 2b, which voids the assumption that the oscillation occurs exactly at the plant phase crossover.)

An attempt to resolve both the noise sensitivity issue and caveat of only identifying one frequency response point was proposed in [13], where the entire data set from the relay experiment was used for the identification of parametric transfer function models. However, the proposed optimization-based algorithm was numerically sensitive and the excitation obtained from the experiment limited the achievable model quality.

1.3. Objective

The objective of this paper is to present a relay method which preserves the strength of the aforementioned ones, while eliminating their weaknesses. In particular, the experiment duration is very short compared to other identification methods, as the need for limit cycle convergence is eliminated. The proposed method makes use of all collected data, not only peak values, to obtain a parametric plant model. Relay hysteresis is explicitly taken into account during identification.

The identification provides a statistical characterization of parameter uncertainties for the obtained model. Subsequently, a method for robust PI or PID controller synthesis is applied. The main strength of the proposed synthesis method is that it explicitly takes the parametric plant uncertainty into account. Finally, the combination of the proposed identification and synthesis methods is demonstrated in industrially relevant scenarios.

The main advantages of this novel approach to automatic tuning are:

1. A very short experiment is used, making the procedure less sensitive to occasional disturbances, while voiding the requirement for limit cycle convergence.

2. Identification is data-driven, utilizing all experiment data points, as opposed to only peak vales. This enables the drastic reduction of experiment time, while significantly improving noise sensitivity.

3. Identification of a parametric uncertainty model is performed. This allows for robust design, with probabilistic magnitude constraints on transfer functions related to robustness of the closed-loop system.

Experiment and identification are the topics of Section 2. The robust PID tuning procedure is covered in Section 3, and the joint method is demonstrated in Section 4. Results are discussed in Section 5.

2. EXPERIMENT AND IDENTIFICATION

2.1. Experiment

The experiment we are proposing is similar to the one used in [11], with an asymmetric relay. The relay output u takes on the value u_{on}, when its input $-y$ is positive, and u_{off}, when $-y$ is negative. The relation between the output levels is $u_{\text{on}} = -\gamma u_{\text{off}}$. The asymmetry level $\gamma = 1$ corresponds to the (classic) symmetric relay. As γ increases, the power spectrum of u is altered, shifting energy from the -180° phase of the process toward the steady state. This is useful, as PID synthesis demands high model fidelity in the third Nyquist quadrant, rather than the critical frequency. In existing autotuners such phase advance is often introduced by means of a hysteresis, which decreases the imaginary part of the negative inverse describing function. An advantage of the asymmetric relay approach lies in that the hysteresis level corresponding to a given phase advance depends on the process dynamics, and is hence a priorly unknown. Presence of measurement noise dictates a minimum value of $|u_{\text{off}}|$, and γ must consequently be limited to keep y within an acceptable range. A value of $\gamma = 1.5$, motivated by extensive evaluation, was recommend in [11], and will be used throughout this paper.

Instead of the $6 - 8$ relay switches typically needed for convergence to a stable limit cycle, the experiment is terminated after only 3 relay switches, as shown in Figure 3a.

An experiment with only two switches did not provide sufficient excitation. Although the output fits of identified models were good, both the parameter error and its estimated covariance $R_{\mathbf{p}}$ were significantly bigger than for the experiment with three switches. A similar decrease was not experienced when going from three to four switches, which is why experiments with three switches have been used throughout this paper.

While eliminating the need for limit cycle convergence, the proposed identification method shares vulnerabilities with other relay methods, in that it assumes that identification begins with the process in equilibrium, and the absence of load disturbances during the experiment. The short experiment enables use of the procedure in applications where it is expensive, or otherwise impractical, to isolate the process from disturbances during an extended time. If the process cannot be kept disturbance-free even during the short required experiment time, the proposed identification method can readily be extended to identify a constant additive load (input) disturbance together with values of the initial state of the process, corresponding to the $n + k$ first states of (6). The scenario where parameters of a

load disturbance model are identified, together with the initial state of the model, will be thoroughly investigated in future work. Of particular interest is the trade-off between disturbance magnitude and required experiment duration.

The method, as formulated herein, is limited to stable or marginally stable (integrating) systems with positive gain. The experiment with subsequent identification has been evaluated to work for all 134 industrially relevant process dynamics enlisted in [14]. Some of these processes exhibit "difficult" dynamics, such as non-minimum phase responses (right half-plane) zeros. In these cases the FOTD and SOTD models structures present a fundamental limitation to achievable model quality. However, the model structure (6) allows for explicit modeling of zero dynamics, by extending the $\tilde{b}$ vector. While outside the focus of this paper, this mechanism could enable more accurate models of processes with slow zeros or non-minimum phase dynamics.

Noise is assumed to be white, with zero mean and variance σ_n^2, and added to the process output y. An estimate $\hat{\sigma}_n^2$ of the noise variance is computed from open-loop data prior to the experiment, and the relay hysteresis level is set to $\mu = 2\hat{\sigma}_n$. This heuristic has worked well in simulation, and the method is not particularly sensitive to changes away from it. In order to limit the activity in u, it is, however, desirable to keep μ small, while an arbitrarily small value is not practical as it would result in chattering triggered by n.

The relay amplitude, $(1+\gamma)u_{\text{on}}$, is set to 10 % of the admissible control signal range, which is a reasonable value for well-designed processes, for which the end points of the range of the control signal approximately correspond to those of the measurement signal in stationarity.

It is worth mentioning how the proposed experiment relates to (optimal) input design methods proposed in classic *identification for control* literature, such as [15]. While providing a powerful tool, the application of identification for control methods require at least a rough estimate of the dynamics time scale (be it μs, minutes, or even hours), or relatedly the bandwidth, of the process dynamics to be identified. The traditional way to obtain such an estimate for a completely unknown process is through a step response experiment, which is typically of comparable duration to the entire experiment proposed above. By using relay feedback, our proposed experiment is automatically matched to the process time scale, enabling us to fully automatically obtain a useful model, occupying the process during less time than it would take to obtain the prerequisites for an identification for control approach.

2.2. Parameter identification

The plant input u and output y are sampled at period h. We have found it sufficient to keep 200 samples per experiment. The sampled signals are used to obtain parameter estimates $\bar{\mathbf{p}} = [\bar{\mathbf{b}}\ \bar{\mathbf{a}}\ \bar{L}]^\top$, corresponding to the assumed FOTD model structure

$$\hat{P}(s) = \frac{b}{s+a}e^{-sL}. \tag{1}$$

This is done by a version of the output error method from [13], presented below for the more general model structure

$$\hat{P}(s) = \frac{1}{s^k}\frac{b_1 s^{m-1} + b_2 s^{m-2} + \cdots + b_m}{s^n + a_1 s^{n-1} + \cdots + a_n}e^{-sL}, \tag{2}$$

parameterized by $\mathbf{p} = [\mathbf{b}\ \mathbf{a}\ L]^\top$, where $\mathbf{b} = [b_1\ \ldots\ b_m]$, and $\mathbf{a} = [a_1\ \ldots\ a_n]$. Continuous time models are used to limit the number of elements of $\mathbf{p}$, in presence of the delay L. The objective is to minimize (half the squared) $\mathcal{L}_2$-norm of the time domain output error $e = y - \hat{y}$:

$$J(\mathbf{p}) = \frac{1}{2}\int e^2 dt, \tag{3}$$

where $\hat{y}$ is the resulting output when $\hat{P}$ (parameterized by $\mathbf{p}$) is driven by u. The optimization is handled by an active-set solver (invoked from the `fmincon` command in Matlab). To improve convergence, the exact parameter sensitivity gradient $\nabla J = [\partial/\partial b\ \partial/\partial a\ \partial/\partial L]J$ and an approximation of the corresponding Hessian $\Delta J = \nabla^2 J$ are provided in each iteration. The gradient w.r.t. $\mathbf{p}$ is given by

$$\nabla J = \int e \nabla \hat{y} dt, \tag{4}$$

and the Hessian is

$$\Delta J = \int \left(\nabla \hat{y} \nabla \hat{y}^\top + e\Delta\hat{y}\right) dt. \tag{5}$$

The first term of the integrand (5) is positively semidefinite, $\nabla\hat{y}\nabla\hat{y}^\top \succeq 0$, while the integral of the second term is small, $e\Delta\hat{y} \approx 0$, under the realistic assumption that the output error is uncorrelated with its second derivative, $\mathbb{E}[e\Delta\hat{y}] = 0$. It is therefore fair to approximate the Hessian by the integral of the first term (although it is straightforward to extend the method outlined below, to include also the second term). In order to account for the k explicit integrators in (2), k zeros are appended to a, forming $\tilde{a} = [a\ 0_{1\times k}]$, while b is padded by leading zeros, $\tilde{b} = [0_{1\times n-m+k}\ b]$, to match the length of $\tilde{a}$. Using the results from [16], it is then possible to construct the continuous time LTI state space system

$$\begin{aligned}
\dot{x} &= \underbrace{\begin{bmatrix} -\tilde{a} & 0_{n+k\times n+k} \\ I_{n+k-1\times n+k} & \\ \tilde{b} & -\tilde{a} \\ 0_{n+k-1\times n+k} & I_{n+k-1\times n+k} \end{bmatrix}}_{A} x + \underbrace{\begin{bmatrix} 1 \\ 0_{2(n+k)-1\times 1} \end{bmatrix}}_{B} u \\
z &= \underbrace{\begin{bmatrix} \tilde{b} & 0_{m\times n+k} \\ 0_{m\times n+k-m}\ I_{m\times m} & \\ 0_{n+k\times n+k} & -I_{n+k\times n+k} \\ \tilde{r} & 0_{1\times n} \end{bmatrix}}_{C} x + \underbrace{\begin{bmatrix} 0_{m\times n+k+1} \\ q \end{bmatrix}}_{D} u,
\end{aligned} \tag{6}$$

where q and r are the deconvolution of $[-\tilde{b}\ 0]$ and $[1\ \tilde{a}]$, and $\tilde{r}$ is r with its first element removed. This essentially corresponds to a polynomial division of the numerator and denominator polynomials of (2). The first element of the output z of (6), is $\hat{y}$, followed by the vectors $\partial\hat{y}/\partial b$ and $\partial\hat{y}/\partial a$. The last element of z is $\partial\hat{y}/\partial L$. Consequently, both $\hat{y}$ and $\nabla\hat{y}$ are directly available through z, which enables straightforward computation of J, ∇J and (the mentioned approximation of) ΔJ, by

numeric integration of the solution of (6):

$$z = \mathcal{L}^{-1}((C(sI - A)^{-1}B + D)\mathcal{L}(u)). \tag{7}$$

Obtaining the minimum $\bar{J}$ and corresponding (expected) parameter vector $\bar{\mathbf{p}}$, thus only involves computing the solution z of (7) in each iteration of the active-set algorithm. Since the sampling period is constant, this can be done by simulation of a discrete time system. Each evaluation step of the objective and its parameter sensitivities (from which ∇J and ΔJ are assembled) thus only involves the computations of matrix exponentials associated with the zero-order-hold discretization of (6), being a system of order $2n + k$, and cheap matrix operations (matrix-vector multiplication, vector addition) required for simulation of the resulting discrete time system. (The identification algorithm was implemented in non-optimized Matlab code on a standard desktop computer. Execution time did not exceed 1 s for any of the models reported in this paper.) A further discussion of the time complexity associated with evaluation of the objective and associated sensitivities is found in [16].

2.3. Parametric uncertainty

In addition to the expectation $\bar{\mathbf{p}}$, the optimization provides the asymptotic covariance matrix

$$R_{\mathbf{p}} = \mathbb{E}\left[(\mathbf{p} - \bar{\mathbf{p}})(\mathbf{p} - \bar{\mathbf{p}})^{\top}\right] = \frac{2}{N}\bar{J}(\mathbf{p})\left(\Delta\bar{J}(\mathbf{p})\right)^{-1}, \tag{8}$$

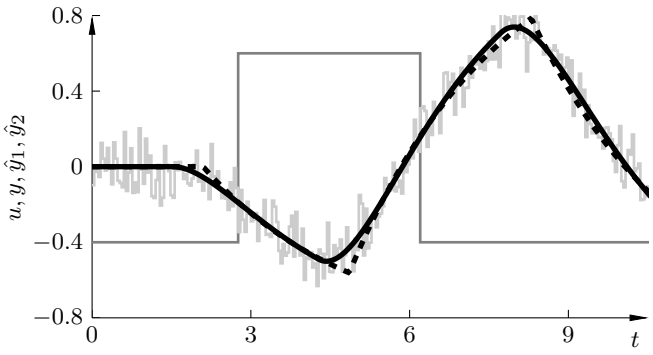
where N is the number of samples [16]. The standard deviations of the parameter estimates decrease $\propto 1/\sqrt{N}$, meaning that one cannot expect significantly improved estimation precision, by (small) increases in experiment duration.

It should be noted that the parameter covariances will be used solely to ensure robust performance of the synthesized controller, and not as a quality measure of the obtained model. For instance, Example 2.1 will show that individual parameter variances of an SOTD model can exceed those of the corresponding FOTD model, while latter constitutes a poorer (frequency domain) fit. This is explained by a decrease of optimization cost sensitivity with respect to individual parameters when the number of parameters is increased – a situation closely related to over-fitting.

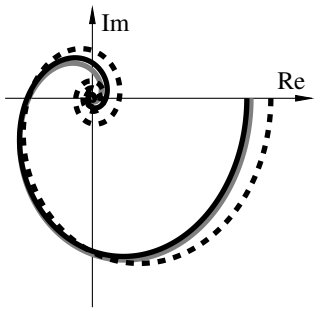
2.4. Notes on convergence

The lack of convexity of the output error minimization problem (due to the autoregressive a vector), results in a lack of formal convergence guarantees. Convergence instead relies on a sufficiently exciting input u.

By evaluation on 134 industrially relevant plant models enlisted in [14], it was validated that initialization of the active-set algorithm with the parameter vector $\mathbf{p} = 0_{3\times 1}$ was sufficient to produce the global optimum, with u generated by the experiment of Section 2. This held true for additive white measurement noise levels corresponding to what is shown in Figure 3a. It can also be noted that the global optima for second-order time-delay (SOTD) models (equation (2) with $k = 0, m = 1, n = 2$) were found for all 134 plant models, when $\mathbf{p}$ of each SOTD model was initialized based on $\bar{\mathbf{p}}$ of the corresponding FOTD model, as explained in Example 2.1. (Initializing the SOTD identification with the zero parameter vector yields the same result for most cases.



(a) Control signal u (gray, thin), measurement y (light gray), FOTD model output $\hat{y}_1$ (black, dashed), SOTD model output $\hat{y}_2$ (black, solid).



(b) Nyquist curves of process P (gray), FOTD model P_1 (black, dashed) and SOTD model P_2 (black, solid).

Figure 3. Outcome of experiment and subsequent identification carried out on P from (9).

However, with default numeric settings, the `fmincon` active-set solver identifies $\mathbf{p} = 0_{4\times1}$ as a possible local minimum for some.)

We have found that further increase of model order generally leads to significant deterioration of the asymptotic parameter covariance estimates, indicating the need of a more elaborate experiment design, should higher model orders be desired. Although not the focus of our work, a more thorough analysis of the number of parameters one can expect to estimate from a given experiment is available through the persistency of excitation of the input signal, defined as the number of singular values of the input covariance matrix exceeding a certain threshold [17, 18].

In this work, the obtained models are not the end result, but used for PID synthesis. In this context, model orders exceeding 2 do not contribute significantly to the end result, as the PID controller only has two zeros, by which the closed-loop poles are determined.

If the intended controller structure is not known at the time of synthesis, the Vinnicombe ν-gap metric is a useful tool to assess model quality for control, as discussed in [18].

Before moving on to controller synthesis, the proposed identification scheme is demonstrated using a realistic example.

Example 2.1. This example demonstrates the experiment and subsequent identification on the plant

$$P(s) = \frac{2}{(s+1)^3} e^{-s}. \tag{9}$$

The sampling period is $h = 0.04$, which corresponds to 100 samples per average residence time, defined as $T_{ar} = P'(0)/P(0)$ in [19]. This conservatively meets the suggested minimum of 200 samples per experiment. (In an experiment conducted on a real process, one could initially sample as often as admissible, and subsequently downsample so that h exactly corresponds to 200 samples for the entire experiment.) The noise level in this experiment is determined by the standard deviation $\sigma_n = 0.1$. Figure 3a shows the control signal u (gray, thin), measurement y (light gray), and FOTD model output $\hat{y}$ (black, dashed). The obtained FOTD model consists of the expected parameter vector $\bar{\mathbf{p}}_1 = [\bar{b}\ \bar{a}\ \bar{L}]^\top = [0.80\ 0.35\ 2.10]^\top$, corresponding to

$$\hat{P}_1(s) = \frac{2.29}{2.82s + 1} e^{-2.10s}, \tag{10}$$

and the asymptotic covariance matrix

$$R_{\mathbf{p}_1} = 10^{-3} \begin{bmatrix} 0.3994 & 0.4128 & 0.4219 \\ 0.4128 & 0.7260 & 0.7379 \\ 0.4219 & 0.7379 & 1.0729 \end{bmatrix}. \tag{11}$$

By taking the square roots of its diagonal elements, and dividing these values by the expected parameter values, we obtain the relative standard deviations $\tilde{\sigma}_{\bar{\mathbf{p}}_1} = [\tilde{\sigma}_b \ \tilde{\sigma}_a \ \tilde{\sigma}_L]^\top = [2.5 \ 7.6 \ 1.6]^\top$ %.

The FOTD model enables subsequent identification of an SOTD one, for which the initial parameter vector is obtained by assuming two equal real poles and matching static gains, average residence times and normalized time delays [19] with the identified FOTD model $\bar{\mathbf{p}}_2$. The resulting SOTD model has parameter vector $\bar{\mathbf{p}}_2 = [\bar{b} \ \bar{a}_1 \ \bar{a}_2 \ \bar{L}]^\top = [1.01 \ 1.44 \ 0.56 \ 1.52]^\top$, corresponding to

$$\hat{P}_2(s) = \frac{1.10}{s^2 + 1.44s + 0.56} e^{-1.52s}, \tag{12}$$

and asymptotic covariance matrix

$$R_{\mathbf{p}_2} = 10^{-2} \begin{bmatrix} 2.9521 & 4.2102 & 0.9837 & 1.0884 \\ 4.2102 & 6.0872 & 1.3583 & 1.5098 \\ 0.9837 & 1.3583 & 0.4578 & 0.4899 \\ 1.0884 & 1.5098 & 0.4899 & 0.5608 \end{bmatrix}. \tag{13}$$

The covariance of (13) corresponds to relative parameter standard deviations $\tilde{\sigma}_{\mathbf{p}_2} = [\tilde{\sigma}_b \ \tilde{\sigma}_{a_1} \ \tilde{\sigma}_{a_2} \ \tilde{\sigma}_L]^\top = [16 \ 17 \ 12 \ 5.0]^\top$ %. Its Nyquist curve, as well as its output $\hat{y}_y$, when driven by u, are shown in Figure 3.

Note that the structure of the process dynamics (9), being a third order time-delayed system, does not match that of the identified FOTD model. (Examples where process and model structures match will be given in Section 4.1.) One consequence of the structural mismatch is that the delay estimate exceeds the actual delay, more so for the FOTD than SOTD model, in order to account for the negative phase shift introduced by unmodeled poles.

While not available in a real identification scenario, comparison between the actual process Nyquist curve, and those of the identified (mean) models, serves as a good indicator of model quality. For the current example, this comparison is shown in Figure 3b, where the actual process P is shown (light grey), together with the identified FOTD model P_1 (black), and SOTD model P_2 (dark gray). As expected, model fit is best in the third quadrant, to which the power spectrum of u is concentrated for asymmetric relay experiments as discussed in [18]. For the FOTD model P_1, this model fit comes at the cost of a comparatively poor steady state gain estimation, as seen in Figure 3b. However, the steady state gain has very little impact on robust PID synthesis.

Ultimately, the model quality is reflected in the achieved closed-loop performance. To this end, controllers based on the models from this example will be studied in Example 3.1.

3. SYNTHESIS

The setup of Figure 1 is considered, where C is a PID controller parametrized as:

$$C(s) = k_p + \frac{k_i}{s} + k_d s. \tag{14}$$

The identification procedure described in Section 2 is used to obtain a parametric model of the process P, together with an estimate of the parameter covariance. I.e., it provides a probabilistic characterization of the uncertainty. While robust control design techniques typically deal with uncertain-but-bounded perturbations ($\mathcal{H}_\infty$ control, QFT, a version of uncertain IMC for PID [20], etc.), most parameter identification methods, such as the output-error method introduced above, yield uncertainty descriptions characterized by an expectation and covariance. These are often assumed to stem from an underlying multivariate Gaussian, which is of an unbounded nature.

The purpose of this section is to introduce a general method by which synthesis methods relying on constrained optimization can be readily extended to handle unbounded model parameter uncertainty. While this paper considers PID design, the proposed synthesis method is not limited to the controller structure (14). In fact, the same methodology is applicable for synthesis of any linear controller.

3.1. Control design problem

The synthesis problem formulation is based on propagating the model uncertainty (assuming that model parameters obey a multivariate Gaussian distribution) through to a performance index J, which is minimized, and robustness indices φ_i, which are constrained. (Propagation of the model uncertainty through some closed-loop transfer functions of interest was previously presented in [21].) This allows the formulation of the synthesis problem as a stochastic optimization problem:

$$\begin{aligned} &\text{minimize} \quad \overline{J}, \\ &\underset{\forall i}{\text{subject to}} \quad \overline{\varphi}_i + \alpha_i \sigma_{\varphi_i} \leq 0. \end{aligned} \tag{15}$$

The design parameters α_i enable the user to specify the confidence with which the individual robustness constraints should be met.

3.2. Objective and Constraint Choice

Most of the control loops present in industry work in regulatory mode, i.e. the main concern is to attenuate disturbances (we assume disturbances entering at the process input). For situations where reference tracking is of priority, standard practice, as proposed in e.g. [14], is to use a two degrees of freedom (2 DOF) controller, where first a feedback controller is synthesized to handle disturbances and provide robustness, whereupon a feed forward controller is synthesized to handle reference tracking.

A common performance measure, to evaluate disturbance attenuation, is the integrated absolute error (IAE)

$$\text{IAE} = \int_0^\infty |e(t)| dt, \tag{16}$$

where $e(t)$ is the error due to a unit step load disturbance. An analytical solution is difficult (if not impossible) to obtain with IAE as performance index. As a tractable alternative it is common to use the integrated error (IE)

$$\text{IE} = \int_0^\infty e(t)dt. \tag{17}$$

In the context of our synthesis scenario, minimization of (17) is equivalent to maximization of the integral gain, k_i in (14). Note that this corresponds to a non-stochastic performance index, as k_i is not explicitly dependent of the process parameters.

For non-oscillatory responses, the two indices (IAE and IE) are equal, and for well-damped systems, (17) constitutes a good approximation of (16). The approach taken herein is therefore to minimize $-k_i$ (i.e., maximize k_i), and ensure well-damped behavior through the robustness constraints.

Robustness is enforced through stochastic $\mathcal{H}_\infty$ constraints on the sensitivity, $S = (1 + PC)^{-1}$, and complementary sensitivity, $T = 1 - S$. Consequently, the optimization problem (15) takes on the following form

$$\begin{aligned} \underset{k_p,k_i,k_d}{\text{minimize}} \quad & -k_i, \\ \text{subject to} \quad & \overline{\|S\|}_\infty + \alpha_s \sigma_{\|S\|_\infty} \leq M_s, \\ & \overline{\|T\|}_\infty + \alpha_t \sigma_{\|T\|_\infty} \leq M_t. \end{aligned} \tag{18}$$

Note that this control design problem is a stochastic version of the well-known M-constraint Integral Gain Optimization (MIGO) design, introduced in [22]. The same approach may be used to deal with any other closed-loop transfer function of interest, simply by adding new constraints.

3.3. Uncertainty propagation

Most optimization algorithms rely on performing an extensive number of cost and constraint evaluations. Here, the formulation (18) poses a problem, due to the complication associated with constraint evaluation. In order to illustrate this, and review a few different options, we introduce a generic function g, which will take on the roles of $\|S\|_\infty$ or $\|T\|_\infty$, and denote by f the joint probability density function of $\mathbf{p}$.

Evaluating the functions of each constraint corresponds to evaluating the following multidimensional integrals

$$\overline{g}(\mathbf{p}) = \mathbb{E}[g(\mathbf{p})] = \int_{\mathbb{R}^n} g(\mathbf{p}) f(\mathbf{p}) d\mathbf{p}, \tag{19}$$

$$\sigma^2_{g(\mathbf{p})} = \int_{\mathbb{R}^n} (g(\mathbf{p}) - \overline{g}(\mathbf{p}))(g(\mathbf{p}) - \overline{g}(\mathbf{p})) f(\mathbf{p}) d\mathbf{p}, \tag{20}$$

where n is the dimensionality of $\mathbf{p}$. As analytic evaluation of (19) and (20) is generally not possible, numeric approximations need to be made. There exist a few options:

One approach lies in approximating g by the first terms of its Taylor series expansion around $\overline{\mathbf{p}}$. In nonlinear estimation, this approach is referred to as the Extended Kalman Filter (EKF) [23]. Adopting it to the considered context of robust synthesis was proposed in [21]. Note that for a second-order approximation it is necessary to compute both the Jacobian and Hessian of g.

Another approach is the use of Monte Carlo (MC) quadrature. This approach relies on drawing many samples from the distribution generated by f, applying g to each sample, and computing the ensemble mean or variance. This method is conceptually very easy to apply, but typically results in an excessive computational burden.

A third approach, also adopted from nonlinear estimation, is the use of test-point methods, as introduced in [24], and further explained in Section 3.4. Like MC, test-point methods rely on evaluating g at a number of points (the test-points). However, these points are chosen in a clever and deterministic way, keeping their numbers much smaller than what is needed for MC quadrature.

3.4. Test-point methods

The idea behind test-point methods is to approximate the expectation integral (19) as a weighted sum

$$\overline{g}(\mathbf{p}) \approx \sum_{i=0}^{N-1} w_i g(\mathbf{p}^{(i)}), \tag{21}$$

of the integrand values at specific points

$$\mathbf{p}^{(i)} = \left[p_1^{(i)}, p_2^{(i)}, \ldots, p_n^{(i)}\right]^\top. \tag{22}$$

These points, $\mathbf{p}^{(i)}$, are referred to as the test-points, and chosen such that the resulting approximation matches that obtained from a Taylor expansion of order d, without the need to compute the Jacobian, Hessian, and possibly higher order sensitivities. In other words, test-point methods guarantee *exact* integration when g is a multivariate polynomial of order not exceeding d. The way this guarantee is achieved, is outlined below.

Consistent with Section 2, it is assumed that $\mathbf{p}$ is a multivariate Gaussian. Without loss of generality, as shown later, it will suffice to consider the case where $\mathbf{p}$ is of zero mean and unity variance. Using the Taylor series expansion of $g(\mathbf{p})$ about the expected value $\overline{\mathbf{p}} = \mathbf{0}$, (19) can be rewritten

$$\overline{g}(\mathbf{p}) = \sum_{N_1=0}^{\infty} \cdots \sum_{N_n=0}^{\infty} \frac{\mathbb{E}\left[p_1^{N_1} \cdots p_n^{N_n}\right]}{N_1! \cdots N_n!} \frac{\partial^{N_1+\cdots+N_n} g}{\partial p_1^{N_1} \cdots \partial p_n^{N_n}}(\mathbf{0}). \tag{23}$$

Combining (23) with (21) yields

$$\overline{g}(\mathbf{p}) \approx \sum_{N_1=0}^{\infty} \cdots \sum_{N_n=0}^{\infty} \frac{\displaystyle\sum_{i=0}^{N-1} w_i \left((p_1^{(i)})^{N_1} \cdots (p_n^{(i)})^{N_n}\right)}{N_1! \cdots N_n!} \frac{\partial^{N_1+\cdots+N_n} g}{\partial p_1^{N_1} \cdots \partial p_n^{N_n}}(\mathbf{0}). \tag{24}$$

Equating (23) and (24) leads to a set of equations

$$\sum_{i=1}^{N-1} w_i \left((p_1^{(i)})^{N_1} \cdots (p_n^{(i)})^{N_n}\right) = \mathbb{E}\left[p_1^{N_1} \cdots p_n^{N_n}\right], \tag{25}$$

referred to as the moment constraint equations (MCE). The idea behind test-point methods is to choose test points and corresponding weights, to fulfill all MCEs for which $N_1 + \cdots + N_n \leq d$, where d is referred to as the order of the MCE.

The most widely-known test-point method is the Unscented Transform (UT) [25, 26], and its extensions, such as the Conjugate Unscented Transform (CUT) [27, 28]. These will be briefly introduced below.

The UT makes use of $N = 2n + 1$ test points (n being the dimensionality of $\mathbf{p}$), and corresponding weights, to establish a match up to order $d = 3$. The test points and weights are:

$$\begin{array}{ll} \mathbf{p}^{(0)} = \mathbf{0}, & w_0 = \frac{\kappa}{n+\kappa}, \\ \mathbf{p}^{(i)} = \sqrt{n+\kappa}\mathbf{e}_i, & w_i = \frac{1}{2(n+\kappa)}, \\ \mathbf{p}^{(i+n)} = -\sqrt{n+\kappa}\mathbf{e}_i, & w_{i+n} = \frac{1}{2(n+\kappa)}, \end{array} \tag{26}$$

where $\mathbf{e}_i, i = 1 \ldots, n$ is the unit vector along the i^{th} principal axis, and $\kappa = 3 - n$ (which minimizes error in moment constraint equations of order 4). See [25] for a derivation.

The CUT extends the test-point sets with points along the conjugate coordinate axis $\mathbf{c}_m^{(i)}$, generated by

$$\left\{ \mathbf{c}_m^{(i)}, 1 \leq i \leq 2n\binom{n}{m} \right\} = \text{FS}\big[[\underbrace{1, \ldots 1}_{m}, \underbrace{0, \ldots, 0}_{m-n}]^\top\big], \tag{27}$$

where the FS $[\cdot]$ operator generates a fully symmetric set, closed under all sign and coordinate permutations. For instance, the unit vectors $\mathbf{e}_i$ along the principal axes together with their negated counterparts $-\mathbf{e}_i$, used in the UT, are generated by

$$\left\{ \mathbf{s}^{(i)}, 1 \leq i \leq 2n \right\} = \text{FS}\left[[1, 0, \ldots, 0]^\top\right]. \tag{28}$$

One of the CUT methods, CUT4, is exact up to order $d = 5$ and relies on $N = 1 + 2n + 2^n$ test points defined through

$$\begin{array}{lll} \mathbf{p}^{(0)} = \mathbf{0}, & w_0 = W_0, & \\ \mathbf{p}^{(i)} = r_1\mathbf{s}^{(i)}, & w_i = W_1, & 1 \leq i \leq 2n, \\ \mathbf{p}^{(i+2n)} = r_2\mathbf{c}_n^{(i)}, & w_{i+2n} = W_2, & 1 \leq i \leq 2^n, \end{array} \tag{29}$$

with scaling factors r_1, r_2, and weights W_0, W_1, W_2 for $n > 2$ defined as

$$\begin{array}{lll} r_1 = \sqrt{\frac{n+2}{2}}, & r_2 = \sqrt{\frac{n+2}{n-2}}, & \\ W_1 = \frac{4}{(n+2)^2}, & W_2 = \frac{(n-2)^2}{2^n(n+2)^2}, & W_0 = 0. \end{array} \tag{30}$$

For $n \leq 2$ numerical values are presented in [27]. The CUT4 method can be extended to match higher moments, by introducing test points along additional directions defined through (28). This paper makes exclusive use of the CUT4 method, which will simply be referred to as CUT.

The test-points and weights presented in (26), (29) and (30) assume that $\mathbf{p}$ is of zero mean and unit covariance. In order to consider Gaussians with mean $\mathbf{z}$ and covariance P, it is sufficient to transform the test point set through

$$\mathbf{z}^{(i)} = \bar{\mathbf{z}} + S\mathbf{p}^{(i)}, \tag{31}$$

where $P = SS^\top$, which may be found using Cholesky decomposition.

3.5. Optimization

Once the constraints of (18) can be efficiently evaluated, the resulting (approximated) optimization problem can be solved by any standard optimization solver, applicable for the deterministic counterpart. In this work, the active-set solver (invoked from the `fmincon` Matlab command) has been used, as in the identification procedure of Section 2.2. Execution time for individual synthesis examples of this paper were in the range 5–10 s (non-optimized Matlab code on standard desktop computer).

Example 3.1. This example illustrates the synthesis method outlined above. We will investigate synthesis for the dynamics (9) of Example 2.1. In addition to the objective of maximizing the integral gain, we impose the stochastic constraint $\overline{\|S\|}_\infty + \sigma_{\|S\|_\infty} \leq 1.4$. I.e., the expected maximal sensitivity, increased by one standard deviation, should not exceed 1.4.

For this example, the CUT relies on 14 test points in the case of FOTD model (actually 15 test point, but with weight $W_0 = 0$ of $\mathbf{p}^{(0)}$), and 24 test points in the case of SOTD model. Using it to propagate the mentioned parameter uncertainty through the constraint function, yields the controller parameters shown in the first two rows of Table I (one row for the FOTD model, one for the SOTD). In order to analyze validity of the approximation introduced in evaluation of the expectation integrals, a Monte Carlo simulation with 10^5 samples from $\mathcal{N}(\bar{\mathbf{p}}, R_{\mathbf{p}})$ was conducted. The resulting $\overline{\|S\|}_\infty + \sigma_{\|S\|_\infty}$ values are shown in column four of Table I (except for the last two rows, showing $\|S\|_\infty$ values). As seen from the first two rows, the approximation of the expectation integrals introduced through the CUT is fair (as both values lie close to the specified 1.4). Next, we obtain optimal controllers assuming $R_{\mathbf{p}} = 0$. I.e., neglecting the uncertainty information and using the mean $\bar{\mathbf{p}}$ alone. The result is shown for the FOTD and SOTD models on rows 3 and 4 of Table I. As expected, the performance (measured as the k_i-value) is somewhat larger than for the uncertain case. However, MC simulations show that the stochastic robustness constraints are not met.

In order to compare our proposed method with what is standard practice, the following controllers are also presented: AMIGO PID [14], SIMC PI [29], and the IAE-optimal controller honoring $\|S\|_\infty \leq 1.40$) [30]. It can be noted that the SIMC PI tuning in our example coincides with that of the lambda [31] method (with the λ parameter set equal to the process time delay). These tunings were based on the mean FOTD model (10), obtained in Example 2.1. (The methods require models with real poles, while the mean SOTD model (12) obtained in Example 2.1 has complex, but well damped, poles.)

The AMIGO tuning rule was designed to produce controllers with $M_s \approx 1.4$ while SIMC was designed for $M_s \approx 1.5$. However, the outcome of the MC simulations show that one cannot expect to meet these constraint levels. Note that the M_s values given in Table I are mean plus one standard deviation with respect to the identified uncertain stochastic model. They were obtained through MC simulation, and are intended to serve as a fair robustness comparison between the evaluated tuning strategies.

Table I illustrates the benefit of utilizing the uncertainty information. As mentioned in Section 3.2, the integral gain k_i serves as a measure of load disturbance attenuation, and hence performance, while M_s is chosen as our measure of robustness – lower values correspond to higher robustness. As seen in Table I, our method lies close in performance and robustness to what one would get if the actual process dynamics were known, with the SOTD version performing better than the FOTD

	k_p	k_i	k_d	M_s	I AE
Optimal FOTD (uncertain)	0.323	0.147	0.342	1.40	8.76
Optimal SOTD (uncertain)	0.343	0.190	0.457	1.40	6.93
Optimal FOTD (mean)	0.332	0.159	0.361	1.44	8.55
Optimal SOTD (mean)	0.356	0.214	0.507	1.46	6.83
AMIGO PID FOTD (mean)	0.351	0.129	0.409	1.51	8.73
SIMC PI FOTD (mean)	0.293	0.104	0	1.61	9.71
IAE optimal (mean)	0.354	0.150	0.307	1.43	8.18
Actual model IE optimal	0.341	0.219	0.531	1.40	7.19
Actual model IAE optimal	0.358	0.217	0.504	1.40	6.97

Table I. PID controller parameters for models obtained in Example 2.1. Rows 1–2 show controller optimized while accounting for uncertainty. Rows 3–4 show optimized controllers, neglecting the uncertainty. Rows 5–7 show controllers synthesized using standard synthesis methods (disregarding uncertainty). The last two rows show controllers optimized directly for (9). For rows 1–7, Column 4 shows mean plus one standard deviation of the optimization constraint, obtained through MC simulation, and column 5 shows the corresponding mean IAE. For the last two rows, columns 4–5 show $\|S\|_\infty$ and IAE values for the (9).

counterpart. The proposed method is superior to AMIGO and SIMC in terms of both performance and robustness.

The last column of Table I shows the resulting mean IAE (16) of the controllers – computed in the same fashion as the values of column 4. The mean IAE of the controller optimized for the uncertain SOTD model (row 2) lies within 5 % of the achievable IAE for the actual model (9) (last row). (The fact that the mean IAE over the uncertain model is slightly smaller than the achievable IAE for the actual model is explained by the approximate nature of the uncertain model.)

4. RESULTS

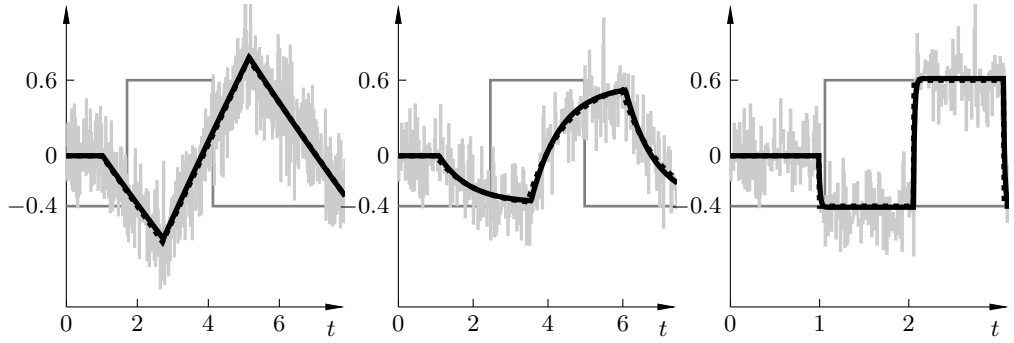
In this section the methods of Section 2 and Section 3 are combined, to form an automatic tuning procedure. Its use is demonstrated on one lag dominant (32), one balanced (33), and one delay dominant (34) plant,

$$P_1(s) = \frac{1}{s}e^{-s}, \tag{32}$$

$$P_2(s) = \frac{1}{s+1}e^{-s}, \tag{33}$$

$$P_3(s) = e^{-s}. \tag{34}$$

Although simple, these plant models are good approximations of a vast majority of process industrial plants. (An example with higher-order process dynamics, where the process and model structures are not matching, was provided in Example 3.1.)



(a) Lag dominant process P_1. (b) Balanced process P_2. (c) Delay dominant process P_3.

Figure 4. Experiment and identification results for (32)–(34), with $\sigma_n = 0.2$. Line styles and colors correspond to those of Figure 3.

	$\mathbf{p}_1^*$	$\bar{\mathbf{p}}_1$	$\sigma_{\mathbf{p}_1}$	$\mathbf{p}_2^*$	$\bar{\mathbf{p}}_2$	$\sigma_{\mathbf{p}_2}$	$\mathbf{p}_3^*$	$\bar{\mathbf{p}}_3$	$\sigma_{\mathbf{p}_3}$
b	1	0.9968	0.0193	1	1.1650	0.0934	$1/\epsilon_0$	71.5911	16.8403
a	0	0.0399	0.0225	1	1.2444	0.1520	$1/\epsilon_0$	70.1077	16.7293
L	1	1.0209	0.0233	1	1.0880	0.0444	1	0.9900	0.0017

Table II. True parameters $\mathbf{p}^*$, estimates $\bar{\mathbf{p}}$ and corresponding standard deviations $\sigma_{\mathbf{p}}$, for (32)–(34), with $\sigma_n = 0.2$. The true a and b parameters of $\mathbf{p}_1$ lie in the limit $\epsilon_0 \to 0$.

	$\mathbf{p}_1'$	$\mathbf{p}_2'$	$\mathbf{p}_3'$
K	24.983	1.068	0.979
T	25.063	0.804	0.014
L	1.021	1.088	0.990

Table III. Parameter estimates $\hat{\mathbf{p}}'$ for (32)–(34).

4.1. Experiment and identification

The proposed experiment of Section 2.1 was conducted on (32)–(34), with additive white measurement noise of zero mean and variance $\sigma_n^2 = 0.04$. The outcomes are shown in Figure 4, while Table II shows the identified parameters $\bar{\mathbf{p}}$, together with true values $\mathbf{p}^*$. It also shows the estimated standard deviations $\sigma_{\mathbf{p}} = \operatorname{diag}\left(\sqrt{R_{\mathbf{p}}}\right)$, where the square root is element-wise. Reparameterization from $\mathbf{p} = [b\ a\ L]^\top$ to $\mathbf{p}' = [K\ T\ L]^\top = [b/a\ 1/a\ T]^\top$, corresponding to the structure

$$P(s) = \frac{K}{sT+1} e^{-sL}, \tag{35}$$

gives the values shown in Table III. Although errors in b and a of $\mathbf{p}_1$ in Table II may seem large, they correspond to a time constant $T = 0.014$, as shown in Table III. This is less than the system sampling period of $h = 0.040$. Changing the plant time constant in this range would not affect behavior of the closed-loop system noticeably – as both robustness and performance are limited by the plant delay. In a similar fashion, the design of the experiment limits the accuracy by which a (or equivalently

	k_p	k_p^*	k_i	k_i^*
P_1	0.299	0.298	0.041	0.041
P_2	0.309	0.361	0.334	0.373
P_3	0.146	0.158	0.433	0.472

Table IV. PI controller parameters for (32)–(34), together with the optimal ones (marked $*$).

T) is estimated, when the plant is lag dominant, such as (32). For the noise free case (dashed black line in Figure 4a), the longest time during which y is monotonically growing, can be expressed analytically:

$$t = L\left(1 + \frac{u_{\text{on}}}{u_{\text{off}}}\right) + 2\frac{\mu}{bu_{\text{off}}} = 2.5L + 10\hat{\sigma}_n^2 = 2.9. \tag{36}$$

From (36) it is clear that it is not possible to distinguish between $a = 0$ and $a \ll 1/L \Leftrightarrow T \gg L$, based on the experiment. To see why this is not a problem, we note that the design and performance of a robust controller is mainly influenced by the plant dynamics around a phase shift of $-\pi$ (radians). The phase of the delay $Ke^{-i\omega L}$ equals $-\pi/2$ for $\omega = \pi/(2L)$. The phase of the pure lag at this frequency is $-\arctan(\frac{\pi}{2}\frac{T}{L})$, which is close to $-\pi/2$ for $T \gg L$ (resulting in a phase shift of $-\pi$ of the combined delay and lag). From a controller synthesis perspective it is therefore reasonable to approximate the pure lag with an integrator whenever $T \gg L$, and vice versa (as in our case).

To summarize, we note two things:

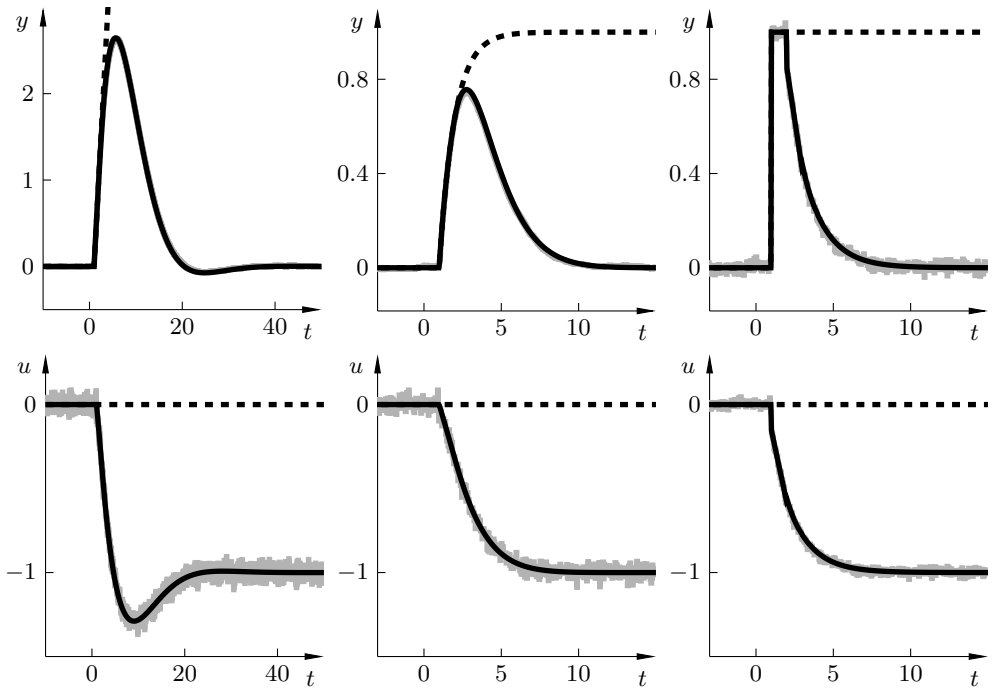
1. The accuracy by which a is estimated is limited by h for delay dominated plants, and by L for lag dominated ones.
2. Accurate estimates of a are secondary to accurate estimates of L for both delay- and lag dominated plants.

Apart from the inherent, and relatively harmless, errors discussed above, the parameter standard deviations of Table II, as well as the output fits shown in Figure 4, indicate good estimation performance, despite short experiment duration and high noise level.

4.2. Synthesis

PI controllers were designed for each of (32)–(34), based on the estimated parameter means presented in Table II and the corresponding covariance matrices. Sensitivities were constrained by $\overline{\|S\|}_\infty + 2\sigma_{\|S\|_\infty} \leq 1.4$ and $\overline{\|T\|}_\infty + 2\sigma_{\|T\|_\infty} \leq 1.4$, while maximizing the integral gain, k_i. The resulting controller parameters are shown in Table IV, together with those of the true IE-minimizing controllers, honoring $\|S\|_\infty \leq 1.4$ and $\|T\|_\infty \leq 1.4$. (The latter would have been obtained for the certain plant model $\bar{\mathbf{p}} = \mathbf{p}^*$ and $R_{\mathbf{p}} = 0$.)

Figure 5 shows time domain simulations of the resulting closed-loop systems, with noise intensity as during identification. The top row shows the plant outputs, the bottom row shows the corresponding control signals. One can note how the inherent noise filtering of the plants increase as they get more lag dominated.



(a) Lag dominant process P_1. (b) Balanced process P_2. (c) Delay dominated process P_3.

Figure 5. Load disturbance unit step responses of the resulting closed-loop systems (gray), together with the responses of the (unavailable) optimal controllers for the noise free case (black). Dashed lines show the open-loop responses.

The sensitivity peaks for the designs are $M_{s,1} = 1.40$, $M_{s,2} = 1.35$, and $M_{s,3} = 1.36$; while the complementary sensitivity peaks are $M_{t,1} = 1.40$, $M_{t,2} = 1.00$, and $M_{t,3} = 1.00$. These designs have been carried out with parameters $\alpha_s = \alpha_t = 2$. Using Chebyshev's inequality, two standard deviations gives a confidence of at least 75 % independently of the distribution [32] (and for the case of normal distribution, a confidence of 95 %).

To get a sense of how much performance is degraded, the responses obtained with the IE-optimal controllers for $M_s \leq 1.4$, in the absence of noise, have been added to the plots of Figure 5.

5. SUMMARY

The main contributions of this work are the demonstrated sufficiency of short experiments, even in the presence of heavy noise, and the possibility to explicitly model, and account for, model parameter uncertainty in the design. This is enabled by simultaneous identification of a nominal model, and a joint Gaussian uncertainty description of its parameters, followed by a propagation of model parameter uncertainty through stochastically formulated robustness constraints. The latter is performed using an extension of the unscented transform, making it computationally tractable, as compared with e.g. MC quadrature.

The proposed automatic tuning procedure has been demonstrated to honor constraints, while being close to optimal for both lag dominated, balanced and delay dominated dynamics, despite very short experiment durations and high noise levels.

REFERENCES

1. Vilanova R, Visioli A. *PID control in the third millennium*. Springer, 2012.
2. Ender DB. Process control performance: Not as good as you think. *Control Engineering* 1993; **40**(2):180–190.
3. Bialkowski WL. Dream versus reality: a view from both sides of the gap. *Pulp & Paper Canada* 1993; **94**(11):19–27.
4. Desborough L, Miller R. Increasing customer value of industrial control performance monitoring – Honeywell's experience. *AiChe Symposium Series* 2002; **326**:169–189.
5. Åström KJ, Hägglund T. Automatic tuning of simple regulators with specifications on phase and amplitude margins. *Automatica* 1984; **20**(5):645–651.
6. Friman M, Waller KV. A two-channel relay for autotuning. *Industrial and Engineering Chemistry Research* 1997; **36**(7):2662–2671.
7. Ionescu C. The next generation of relay-based PID autotuners (part 1): Some insights on the performance of simple relay-based PID autotuners. *IFAC Conference on Advances in PID Control*, Brescia, Italy, 2012.
8. Menani S, Koivo H. A comparative study of recent relay autotuning methods for multivariable systems. *International Journal of System Science* 2001; **32**(4):443–466.
9. Johansson KH, Barabanov AE, Åström KJ. Limit cycle with chattering in relay feedback system. *IEEE Transactions on Automatic Control* 2002; **47**(9):1414–1423.
10. Panda RC, Yu CC. Shape factor of relay response curves and its us in autotuning. *Journal of Process Control* 2005; **15**(8):893–906.
11. Berner J, Åström KJ, Hägglund T. Towards a new generation of relay autotuners. *19th IFAC World Congress*, Cape Town, South Africa, 2014.
12. Kaya I, Atherton DP. Parameter estimation from relay autotuning with asymmetric limit cycle data. *Journal of Process Control* 2001; **11**(4):429–439.
13. Soltesz K, Hägglund T, Åström KJ. Transfer function parameter identification by modified relay feedback. *American Control Conference*, Baltimore, Maryland,USA, 2010.
14. Åström KJ, Hägglund T. Revisiting the Ziegler-Nichols step response method for PID control. *Journal of Process Control* 2004; **14**(6):635–650.
15. Gevers M, Ljung L. Optimal experiment designs with respect to the intended model application. *Automatica* ; **22**(5):543–554.
16. Åström KJ. Maximum likelihood and prediction error methods. *Automatica* 1980; **16**(5):551–574.
17. Ljung L. *System Identification - Theory For the User*. Prentice Hall, 1999.
18. Berner J. Automatic tuning of PID controllers based on asymmetric relay feedback. *Licentiate Thesis ISRN LUTFD2/TFRT--3267--SE*, Department of Automatic Control, Lund University, Sweden 2015.
19. Åström KJ, Hägglund T. *Advanced PID Control*. ISA - The Instrumentation, Systems, and Automation Society: Research Triangle Park, NC 27709, 2006.
20. Leva A, Colombo AM. Estimating model mismatch overbounds for the robust autotuning of industrial regulators. *Automatica* 2000; **36**(12):1855–1861.
21. Boje E. Quantitative feedback design for systems with probabilistic parameterizations. *International journal of robust and nonlinear control* 2007; **17**(2–3):173–179.
22. Åström KJ, Panagopoulos H, Hägglund T. Design of PI controllers based on non-convex optimization. *Automatica* 1998; **34**(5):585–601.
23. Daum F. Nonlinear filters: beyond the Kalman filter. *Aerospace and Electronic Systems Magazine* 2005; **20**(8):57–69.
24. Mercader P, Soltesz K, Baños A. PID synthesis under probabilistic parametric uncertainty. *American Control Conference (Accepted)*.
25. Julier SJ, Uhlmann JK. A general method for approximating nonlinear transformations of probability distributions. *Technical Report*, Department of Engineering Science, University of Oxford 1996.
26. Julier SJ, Uhlmann JK. New extension of the Kalman filter to nonlinear systems. *AeroSense'97*, International Society for Optics and Photonics, 1997; 182–193.
27. Adurthi N, Singla P, Singh T. The conjugate unscented transforman approach to evaluate multi-dimensional expectation integrals. *American Control Conference (ACC), 2012*, IEEE, 2012; 5556–5561.
28. Adurthi N, Singla P. Conjugate unscented transformation-based approach for accurate conjunction analysis. *Journal of Guidance, Control, and Dynamics* 2015; :1–17.
29. Skogestad S. Simple analytic rules for model reduction and PID controller tuning. *Journal of Process Control* 2003; **13**(4):291–309.

30. Garpinger O, Hägglund T. Software-based optimal PID design with robustness and noise sensitivity constratins. *Journal of Process Control* 2015; **33**(9):90–101.
31. Dahlin EB. Designing and tuning digital controllers. *Instruments and Control Systems* 1968; **41**(6):77–83.
32. Papoulis A, Pillai SU. *Probability, Random Variables and Stochastic Processes*. 4 edn., McGraw-Hill, 2002.