Automatic Re-tuning of Poor-Performing PI-based Control Systems

Stopakevych A. O.[1], Stopakevych O. A.[2], Tigarev A.M.[1], Vorobiova O.M.[1]
[1] State University of Intelligent Technologies and Telecommunications
[2] National Odesa Polytechnic University
Odesa, Ukraine

Abstract. The goal of the work is to create a new method for automatic re-tuning of PI controllers. It is achieved by solving the following problems. The first problem is to develop a method to identify a FOPDT estimation model by analyzing the dynamics of a control loop with a PI controller. The second problem is to demonstrate that the use of the estimated model allows to obtain better processes in control systems with PI controllers by comparing the resulting FOPDT model with a number of reduced models. The third problem is to develop a five-stage algorithm for automatic re-tuning of PI controllers. The fourth problem is to verify the software implementation of the developed algorithm. The most significant result was that the basis for estimating the FOPDT model in a loop with a Ziegler-Nichols tuned PI controller was the estimation the ratio of the delay to the time constant of the FOPDT model based on an assessment of the process overshoot. When the real control plant dynamics did not match with the dynamics of the FOPDT model this way was found effective for finding the optimal tuning of the PI controller. The significance of the results obtained was a structural robustness in case of estimated FOPDT model usage. The effectiveness of the procedure has been demonstrated on both linear and nonlinear models. The developed procedure is proposed to be used to automate the procedure of re-tuning SISO control systems when the control loop performance degradation has been detected.
Keywords: control system, automatic re-tuning of the PI controller, FOPDT evaluation model, delay, optimization, structural robustness, control performance, operation.

DOI: https://doi.org/10.52254/1857-0070.2024.2-62.14
UDC: 681.51

Reajustare automată a controlerelor PI în bucle cu control de calitate nesatisfăcător

Stopakevici A.O.[1], Stopakevici O.A.[2], Tigarev A.M.[1], Vorobiova O.M.[1]
[1] Universitatea de Stat de Tehnologii și Comunicații Inteligente, Odesa, Ucraina
[2] Universitatea Națională Politehnică din Odesa, Odesa, Ucraina
Rezumat. Scopul lucrării este de a crea o nouă metodă de reglare automată a controlerelor PI. Se realizează prin rezolvarea următoarelor probleme. Prima problemă este elaborarea unei metode de identificare a unui model de estimare FOPDT prin analiza dinamicii unui contur de control cu un controler PI. A doua problemă este de a demonstra că utilizarea modelului estimat permite obținerea unor procese mai bune în sistemele de control cu controlere PI prin compararea modelului FOPDT rezultat cu un număr de modele reduse. A treia problemă este elaborarea unui algoritm în cinci etape pentru reglarea automată a controlerelor PI. A patra problemă este verificarea implementării software a algoritmului elaborat. Cel mai semnificativ rezultat a fost că baza pentru estimarea modelului FOPDT într-un contur cu un controler PI reglat Ziegler-Nichols a fost estimarea raportului dintre întârzierea și constanta de timp a modelului FOPDT pe baza unei evaluări a depășirii procesului. Atunci când dinamica reală a centralei de control nu se potrivea cu dinamica modelului FOPDT, acest mod a fost găsit eficient pentru găsirea reglajului optim al controlerului PI. Semnificația rezultatelor obținute a fost o robustețe structurală în cazul utilizării estimate a modelului FOPDT. Eficacitatea procedurii a fost demonstrată atât pe modele liniare, cât și pe cele neliniare. Procedura elaborată este propusă pentru a fi utilizată pentru a automatiza procedura de reajustare a sistemelor de control SISO atunci când a fost detectată degradarea performanței conturului de control.
Cuvinte cheie: sistem de control SISO, controlere PI, reajustare automată, evaluare model FOPDT, întârziere, optimizare, robustețe structurală, calitate control, funcționare.
.

Автоматическая перенастройка ПИ-регуляторов в контурах с неудовлетворительным качеством управления

Стопакевич А. Ал.[1], Стопакевич Ал. А.[2], Тигарев А.М. [1], Воробьева Е.М. [1]
[1] Государственный университет интеллектуальных технологий и связи
[2] Национальный университет «Одесская политехника», Одесса, Украина

Аннотация. Целью работы является создание нового метода автоматической перенастройки ПИ-регуляторов. Поставленная цель достигается за счет решения следующих задач. Первая задача состоит в разработке метода идентификации оценочной модели в виде звена первого порядка с запаздыванием (FOPDT), анализируя динамику контура управления с ПИ-регулятором. Вторая задача состоит в сравнении полученной FOPDT модели с рядом редуцированных моделей, с целью демонстрации того, что использование оцененной модели дает возможность получения лучших процессов в системах управления с ПИ-регуляторами. Третья задача состоит в разработке пятиэтапного алгоритма автоматической перенастройки ПИ-регуляторов. Четвертая задача состоит в проверке функционирования программной реализации разработанного алгоритма. Наиболее существенными результатом является то, что. основой для оценки FOPDT модели в контуре с ПИ-регулятором, настроенным по Циглеру-Николсу, является оценка величины отношения времени запаздывания к постоянной времени FOPDT модели на основе оценки перерегулирования процесса. При несовпадении реальной динамики объекта управления с динамикой FOPDT модели именно полученное таким образом отношение является эффективным для поиска оптимальных настроек ПИ-регулятора. Значимость полученных результатов состоит в том, что сравнение динамики системы управления при использовании FOPDT модели, полученной из замкнутой системы управления с ПИ-регулятором, и динамики систем управления, в которых ПИ-регулятор управляет моделями высокого порядка, показывает выраженную структурную робастность, по сравнению с использованием для настройки моделей, полученной методами редукции. Эффективность процедуры показана как на линейной, так и на нелинейной моделях. Предложенную процедуру предлагается использовать для автоматизации процедуры перенастройки одноконтурных систем управления при выявлении факта деградации качества управления в процессе ее функционирования на технологическом объекте.

Ключевые слова: система управления, автоматическая перенастройка ПИ-регулятора, оценочная FOPDT модель, запаздывание, оптимизация, структурная робастность, качество управления, эксплуатация.

INTRODUCTION.

The problem of achieving high performance of technological processes control systems operation is still relevant. This is due to the fact that the parameters of the models, e.g. of power plants, which are used to design control systems, are changing, and in most cases there are not enough resources to perform a complete systematic re-tuning of hundreds of control loops. This is confirmed by many industrial studies (for example, the ABB study [3]), despite the fact that hundreds of tuning methods are available and numerous studies are being conducted [1, 2]. The article proposes a solution based on the development of a procedure for automatic re-tuning of controllers in existing SISO control loops of industrial technological plants.

When designing control loops for most plants of the considered class, it is accepted that PI controllers are used and their initial tuning is obtained according to the parameters of estimated FOPDT models $k \cdot e^{-\tau \times s} / (T \cdot s + 1)$ [1, 2]. However, the resulting tuning is most often achieved manually at the process plant to obtain the desired performance indices. Sometimes the tuning is done by taking into account the values that were obtained, for example, on a similar plant or that approximately correspond to the gain coefficient and response time of the plant. However, with these methods of controller tuning it is almost impossible to achieve robustness of the system, so degradation of the loops occurs [3], when direct performance indices become unacceptable after some time of operation.

There are not many methods that allow you to tune the PI controller by experiment without obtaining an explicit mathematical model of the process (model-free approach). The first of these is the Ziegler-Nichols (ZN) method [4]. However, its use leads to the same problems as empirical tuning, namely low robustness and, on average, not the best performance indices.

Adaptive controllers have problems with the inability to separate noise, control channel dynamics, and disturbance dynamics, and there are difficulties in accounting for delay [5].

Thus, the problem of automatic tuning of PI controllers in single-loop systems remains relevant. IFAC recently organized a Manufacturing Liaison Committee among its members who have industrial experience [6] and surveyed them on topics that are important to industrial objectives. The first position in the ranking of importance (100% for, 0% against) was the specified control problem.

MODELING PROBLEM

It is necessary to separate the problem of constructing models of dynamic systems for reproducing some significant properties of a plant and for designing control systems. It is often assumed that a dynamic model, both theoretical and identified, is sufficiently accurate if the transient processes of the model coincide with the processes of the original plant when certain standard input signals (a step or a series of repetitive signals of a given shape) are acting. For control system design, however, neither an accurate model nor a rough model is necessarily suitable. We illustrate this with two examples.

Example 1. Let us consider the oscillatory model $P = 0.05173/\left(0.003448 \cdot s^2 + 0.03448 \cdot s + 1\right)$ with complex conjugate poles. Such model is typical for plants with recirculation flows. The authors of [17] propose to use the FOPDT approximation P_1 with $k = 0.006, T = 0.1,\ \tau = 0.01$ for control design. We note that model P_1 transfer coefficient is 10 times less than that this one of the model P. On the other hand, FOPDT approximation P_2 with $k = 0.051781,\ T = 0.022327,\ \tau = 0.04108$ is more adequate according to the MSE index. It is worth noting that $P(s)$ has an infinite phase and amplitude margin, which FOPDT approximations cannot reflect. MATLAB pidtune designs I controller $C = 89/s$ for model P, PI controller $M_s = 1.4$ $C_1 = 174 + 5210/s$ and I controller $C_{11} = 163/s$ for model P_1; PI controller $C_2 = 5.73 + 249/s$ and I controller $C_{22} = 1100/s$ for model P_2. The loop $C_1 - P$ has a small margin of stability, but process with damped oscillations. The loops $C_2 - P$ and $C_{22} = P$ are unstable. The loop $C_{11} - P$ is stable and has 32% overshoot.

Example 2. Let us consider the equipole model with monotonic process $1/\left(s+1\right)^5$. We design a controller for this model using the pidtune program, which provides a phase margin of 60° (standard settings). Which FOPDT model will describe this plant accurately enough so that a closed loop system with a controller and this model has the same robust properties as with the original plant? To answer this question we use five ideologically different approaches.

Method 1 – Half rule (HR). The idea is to transfer non-dominant time constants to delay. Method 2 – Two areas. The idea is the ratio of two areas: under the curve and above the curve. Method 3 – Procest. The idea is optimization using the least squares method. The delay search setting was used. Method 4 – Optimization according to the ISE index with a starting point in the form of a rough FOPDT model. Method 5 – Reverse optimization. Given the settings $C(s)$, find the parameters of the FOPDT model $(T,\ \tau)$ from the integral of the absolute value of the difference between the processes of control systems with $P(s)$ and the FOPDT model whose parameters are being sought. From Table 1 we see that the methods give significantly different results. The HR method overestimates the delay, and the Procest method overestimates the time constant. Other methods estimate the τ/T ratio differently. Further we design controllers for each model obtaining in Table 1 using pidtune. In Table 2 we can see direct performance indices when connecting each controller to the original model.

Table 1.

The resulting FOPDT models from the model $P(s)=1/(s+1)^5$

No	Method	$k/T/\tau$	GM/PM	On the base
1	Half-rule [1]	1/1.5/3.5	1.36/49.5	$P(s)$
2	Two areas [7]	1/2.38/2.63	1.99/62.4	Step responces $P(s)$
3	MATLAB Procest	1.058/3.757/1.414	3.71/67.4	
4	Optimization using model dynamics	1/2.624/2.59	2.12/61.7	
5	Optimization using system dynamics	1/2.876/2.68	2.17/59.2	Processes in 2 control systems

Table 2.

Control systems performance indices using models obtained from methods presented in table 1

Index / Model No	1	2	3	4	5	Original
Settings k_p / k_i	0.323/ 0.166	0.383/ 0.185	1.24/ 0.296	0.434/ 0.188	0.955/ 0.223	0.887/ 0.225
Settling time (*ST*)	20.06	18.60	36.84	17.04	21.18	18.54
Overshoot (*OV*)	5.61	5.77	32.26	7.27	13.00	12.83
Rise time (*RT*)	6.61	5.77	2.84	5.54	3.56	3.69

We see that the most stable parameter preserved during such a transformation is the settling time. Optimizing for the dynamics of the control system gives the best fit. Note that, on average, the HR method can give better results when the plant time constants are quite different, but transferring "extra" time constants to the delay is a much more correct strategy than trying to achieve greater compliance along the step response. It may seem that the problem can be solved simply by choosing the appropriate method, but in a real control problem, we do not have an exact model. Even if we want to obtain the most accurate linear model, the answers to such questions as the real order of the model, the parametric (and possibly structural) model uncertainty under the influence of disturbances on the plant dynamics are unknown from a priori knowledge. It will also be problematic to assess the overall influence of servomechanisms and instrumentation on the dynamics of the system. In [8], an experiment was conducted with two methods, one of which was HL. It is shown that the approach where an accurate high-order model is first obtained by identifying transient processes in MATLAB when a chirp signal is applied, and then reduced by two popular methods leads to a significant error, much more significant than comparing any absolutely accurate model and a reduced one. It is also claimed that using a square-wave pulse as the signal gives similar results. The examples given clearly show that it is advisable to isolate a separate class of dynamic models for controller design. If it were possible to obtain an accurate high-order model, then the use of frequency optimization methods similar to those implemented in the pidtune program would greatly simplify the design of control systems. However, this program also has limitations that go beyond the limitations of frequency analysis. The pidtune optimization algorithm tries to achieve the maximum norm of the main Ms and additional Mt sensitivity functions, as well as the desired phase margin of the control system (usually PM = 60°).

PURPOSE AND OBJECTIVES OF THE STUDY

The objective of the study is to develop a program for the automatic re-tuning of PI controllers in a functional control loop. For this purpose, a tuning algorithm has been developed whose main elements are obtaining an estimation model and optimizing the controller parameters based on it. The key idea of tuning is to obtain an estimated FOPDT model from a working control loop and to retune the controller based on its parameters. To retune the controller according to the evaluation model, a special optimization criterion is used while ensuring the robustness of the control system (the phase margin may be about 60°). The rationale for the suitability of the estimation model and the tuning method is given.

PRELIMINARY CONDITIONS FOR THE EXECUTION OF THE ALGORITHM

In order to automate the process of diagnostics and re-tune the control loops in case of a significant number of them, it is proposed to use a specialized centralized program that will interact with control devices through a specially implemented interface. It is assumed that the ISA standard form of the PI controller will be used [2,1]. In addition, the controller must be switched to a two-position mode (preferably with the ability to provide a parallel gain to the control signal). For safety, it is necessary that the controller has control limits. It is assumed that the loop is already working and has been tested for the basic operability. Therefore, the constraints on control, settling time, process type, degree of delay dominance, etc. are approximately known. Requirements for using the program: 1) the controller should not be in a loop with significant cross-links in the channels; 2) the plant must be stable and static (with self-leveling), with a step response close to monotonous, but slight swelling is acceptable; 3) the plant must have dynamic properties that allow it to be put into self-oscillation mode; theoretically, minimal phase plants are not allowed; 4) the delay should not be dominant and should not be too small. The initial data are: 1) T_{95} is approximate to 95% of the plant step response settling time (you can take 3-4 T_i of the current PI controller); 2) maximal deviation of the reference (SP); 3) permissible limitations on controls u_{min}, u_{max}; 4) controller's discrete step is T_s; 5) minimum oscillation amplitude is $h_{min} \geq 0.1$; 6) minimum interval between oscillations is $w_{min} \approx T_{95} / 6$; 7) the magnitude of the difference in derivatives between the amplitudes (for example, for $T_s = 0.1,\ d_m = 0.02$). The latter parameters are needed because under the influence of noise and disturbances the symmetry of the oscillations is broken and it is difficult to distinguish where the peak is and where the noise is. Further, for simplicity, operations for taking into account plants with negative gains and abnormal behavior that requires stopping the program are not described. The autotuning program algorithm is performed in 5 stages and can be implemented in any high-level language (the authors used MATLAB).

STAGE 1. ESTIMATION OF CRITICAL LOOP GAIN

A relay controller is included in the loop instead of the PI controller. It generates signals u_{min}, u_{max} using a control error e. To increase the accuracy and smooth the control signal, a gain with the coefficient $T_s/10$ is connected in parallel with the relay controller. The permissible value of the deviation of the SP reference is set. The algorithm must process $\approx(T_{95}{\cdot}2)/T_s$ of the last points. The peaks P_k and their indices L are determined on the base of a sufficient number of non-zero (in deviations) signal points S taking into account $h_{min}, w_{min}/T_s$. If two or more peaks are found, then $d = diff(P_k, S)$ is calculated, and if $|d| \le dm$ then $d = diff(P_k / S)$ and we start counting the time until the end of the experiment, calculated from the difference of the last L points. If during time $3L$ the condition $|d| \le dm$ is satisfied, then we complete the stage. As a result, we determine $a = max(P) - SP$ and $P_u = (L(end) - L(end-1))/T_s$. Then we can calculate the approximate critical coefficient of the controller $K_{ue} = (4{\cdot}u_{max})/(\pi{\cdot}a) - 0.1{\cdot}T_s$. If the calculation is successful, then we need to remove the reference deviation and switch to manual mode, set the nominal control value and wait for the end of the transient processes.

STAGE 2. DETERMINATION OF THE CRITICAL GAIN AND PERIOD OF SELF-OSCILLATIONS

Real $K_u \in (K_{umin} = 0.5K_{ue}, K_{umax} = 2K_{ue})$. To determine it, we will include a P controller in the loop with the tuning $K_p = 0.5\,K_{ue}$ (following the classic Ziegler-Nichols ZN-1 method). Let us fix the time to achieve 95% of the task multiplied by 1.2 and denote it as T_r. We set the initial vector of possible settings of the P regulator from 4 initial values $KK = [K_{min}, K_{min} + (K_{max} - K_{min})/3, K_{min} + 2\cdot(K_{max} - K_{min})/3, K_{max}]$. Let us select in the program associative arrays of attributes that will be filled in as we study processes in a closed-loop control system with a P controller, namely, an array of attributes KD and an array of PP that includes time intervals between peak. The array keys will be the KK values. Next, the algorithm is to find $K_p = K_u$. Going cyclically through KK, we accept $K_p = KK(i)$, if there is no sign in KD. We submit the control reference and wait for the end of the closed loop system step response. The process is completed if: 1) the difference between the peaks is insignificant; 2) the average deviation value of the exit beyond $T_r/5$ minus the starting point is close to zero; 3) control has reached the limit; 4) time Tr has expired. After the process is completed, 1 is entered in KD if the difference between the peaks amplitude increases, –1 if the difference decreases. If process has quickly reached control limits, then we enter 1 in $KD(i)$ and delete all subsequent elements in KK, starting with $KK(i+1)$. If the difference between the peaks is less than the threshold, then the cycle is completed, $K_u = K_p$ is accepted and the interval between the peaks of the current experiment is accepted as P_u and it is entered in PP. After each i experiment, we reset the value of K_p to zero and wait for the process to return to the initial state (taken as 0) during time T_{95}, or if the average value of the last values of the process during time $T_{95}/5$ becomes close to zero. If all elements of the K_u cycle have been passed and K_p has not been found, then we find the points $KD(i)$ and $KD(i+1)$ of changes in the sign of the signs. If the difference between $KK(i)$ and KK(i+1) is less than 1%, then take K_u as the average between $KK(i)$ and $KK(i+1)$ and P_u as the average between $PP(i)$ and $PP(i+1)$. If the difference is greater, we reform the KK array, taking $KK(i)$ as the first element and then insert 6 new values in the interval [$KK(i)$), $KK(i+1)$], observing the ascending order of the values, and move to the beginning of the program cycle.

STAGE 3. PRIMARY RE-TUNING OF THE PI CONTROLLER

Let's calculate the settings of the PI controller

$$K_p = K_u / 2.2, \quad T_i = P_u / 1.2, \tag{1}$$

where K_p is a setting, which has transferred the control system with the P controller to a self-oscillating mode, P_u is the interval between the peaks of this process. Having installed tuning in the PI controller, we will start step response by increasing the control reference and them waiting for transients completion. We determine the estimated FOPDT model gain k_e as the ratio of the averages between 10 exit and entry points of the plant after completion of the response. We reset the reference and wait for the transients to complete without turning off the controller. If the overshoot is in the range of 10-50%, then we move to the stage 4. If the overshoot is less than required, we begin the search cycle. If overshoot value is small then we increase K_p by $0.1(K_u - K_p)$, else we reduce K_p. If we approach K_u or 0, then we complete the operation with an error. Otherwise, we increase the control reference and wait for the stable behavior of the response process, after which we return the reference to its original value. We estimate the overshoot based on the magnitude of the peak achieved. If we have reached the required overshoot value, then we fix the new K_p as the setting and the achieved overshoot value and proceed to the stage 4.

STAGE 4. ESTIMATED MODEL PARAMETERS CALCULATION

Now having k_e of the estimated FOPDT model, it is necessary to estimate the two remaining parameters of this model. The delay dominance indicator is calculated using an empirical formula that is related to the amount of overshoot σ (%)

$$d_e = \frac{\tau_e}{T_e} = \frac{44.67 - 0.6015 \cdot \sigma}{45.35 + \sigma} \quad (2)$$

The delay of the estimation model τ_e is directly dependent on T_i, since the ZN-1 method implicitly includes the criterion that the rise time should be approximately equal to the delay time. It follows from that T_i must be proportional to τ_e. The P_u value is related to the cutoff frequency w_{cg} and corresponds to the formula $2 \cdot \pi / w_{cg}$. It is known that the delay does not affect the amplitude response and is expressed in a proportional phase shift. It is also known that the cutoff frequency is the limit up to which the system can track input signals without significant attenuation. This is related to the delay because the greater the delay, the slower the system can respond to changes in the input signal, and therefore the lower the frequency at which the system begins to decay d_e. Therefore, the formula $\tau_e = 1.745 / w_{cg}$ is valid. Its relative error is 6..12% depending on d_e and this error can be related to absolute accuracy by the formula $\varepsilon = 12.98 \cdot e^{0.3634 \cdot d_e} - 24.03 \cdot e^{1.643 \cdot d_e}$. However, since in real conditions it is impossible to establish an absolutely accurate value of overshoot due to noise and its assessment is more likely to be overestimated than underestimated, there is not a practical point in complicating the formula. Therefore, we will focus on the equivalent to $\tau_e = 1.745 / w_{cg}$.

$$\tau_e = T_i / 3 \quad (3)$$

We emphasize that since the delay is derived from the frequency criterion, we are not talking about a pure delay, but a delay adequate for the estimated model. Thus, for the plant $1/(s+1)^5$ (see Table 1), formula (3) gives the value $\tau_e = 2.4$, which is in the zone of adequate values between the extremes of finding a pure delay in Procest (value 1.4) and HR (value 3.5) at a point close to the optimum for the estimation model.

Having calculated (2) and (3) it is easy to obtain the time constant

$$T_e = \tau_e / d_e \quad (4)$$

In a transient process without noise, the following formula is valid:

$$k_e \approx 0.7865 / (K_p \cdot d_e) \quad (5)$$

The noise of sensors and actuators mainly affects the value of k_e. If the difference between k_e. obtained in the loop at the third stage and according to formula (5) is large (>20%), this means that the noise is significant. This may also affect the accuracy of the overshoot estimation. In this case, it is worth adding or improving filtering of the measurement signal in the loop at the cost of increasing the inertia of the plant. We will substantiate the adequacy of the proposed formulas (2)-(5) using the method of mathematical experiment. To do this, we will generate many models that can be reduced to FOPDT and do not have excessive fluctuation. To conduct the experiment, 50 mathematical models of dynamics of different orders were randomly generated, which correspond to the requirements of the ZN method (Table 3).

For simplicity, models with integer coefficients were generated. The tuning for these models were obtained using the ZN-1 method.

We will consider the FOPDT approximation to be sufficiently accurate if the processes in the

Table 3.

Models for study

$1:\dfrac{3\cdot e^{-3\cdot s}}{7\cdot s+1}$	$11:\dfrac{(2\cdot s+2)\cdot e^{-1\cdot s}}{4\cdot s+3\cdot s^2+1}$	$21:\dfrac{(5\cdot s+2\cdot s^2+2)\cdot e^{-1\cdot s}}{9\cdot s+14\cdot s^2+6\cdot s^3+1}$	$31:\dfrac{(15\cdot s+8\cdot s^2+s^3+3)\cdot e^{-7\cdot s}}{20\cdot s+31\cdot s^2+11\cdot s^3+s^4+1}$
$2:\dfrac{3\cdot e^{-3\cdot s}}{9\cdot s+1}$	$12:\dfrac{(4\cdot s+3)\cdot e^{-1\cdot s}}{5\cdot s+5\cdot s^2+1}$	$22:\dfrac{(14\cdot s+5\cdot s^2+8)\cdot e^{-1\cdot s}}{8\cdot s+11\cdot s^2+4\cdot s^3+1}$	$32:\dfrac{(12\cdot s+11\cdot s^2-4\cdot s^3+2)\cdot e^{-s}}{10\cdot s+30\cdot s^2+31\cdot s^3+10\cdot s^4+1}$
$3:\dfrac{2\cdot e^{-4\cdot s}}{5\cdot s+1}$	$13:\dfrac{(5\cdot s+8)\cdot e^{-1\cdot s}}{6\cdot s+4\cdot s^2+1}$	$23:\dfrac{(11\cdot s+9\cdot s^2-s^3+2)\cdot e^{-2\cdot s}}{12\cdot s+33\cdot s^2+23\cdot s^3+1}$	$33:\dfrac{(6\cdot s-5\cdot s^3-1\cdot s^4+2)\cdot e^{-3\cdot s}}{12\cdot s+39\cdot s^2+43\cdot s^3+15\cdot s^4+1}$
$4:\dfrac{9\cdot e^{-1\cdot s}}{7\cdot s+1}$	$14:\dfrac{(3\cdot s+2)\cdot e^{-1\cdot s}}{4\cdot s+3\cdot s^2+1}$	$24:\dfrac{(21\cdot s+4\cdot s^2+5)\cdot e^{-1\cdot s}}{9\cdot s+22\cdot s^2+10\cdot s^3+1}$	$34:\dfrac{(37\cdot s+38\cdot s^2-s^3-8\cdot s^4+9)\cdot e^{-s}}{18\cdot s+66\cdot s^2+72\cdot s^3+22\cdot s^4+1}$
$5:\dfrac{2\cdot e^{-4\cdot s}}{5\cdot s+1}$	$15:\dfrac{(s+2)\cdot e^{-8\cdot s}}{11\cdot s+5\cdot s^2+1}$	$25:\dfrac{(9\cdot s+s^2+11)\cdot e^{-2\cdot s}}{18\cdot s+21\cdot s^2+6\cdot s^3+1}$	$35:\dfrac{(3\cdot s+2\cdot s^2+1)\cdot e^{-2\cdot s}}{8\cdot s+16\cdot s^2+12\cdot s^3+3\cdot s^4+1}$
$6:\dfrac{2\cdot e^{-2\cdot s}}{4\cdot s+1}$	$16:\dfrac{(2\cdot s+3)\cdot e^{-2\cdot s}}{4\cdot s+2\cdot s^2+1}$	$26:\dfrac{(13\cdot s+13\cdot s^2+3)\cdot e^{-1\cdot s}}{8\cdot s+19\cdot s^2+14\cdot s^3+1}$	$36:\dfrac{(13\cdot s+8\cdot s^2+5)\cdot e^{-2\cdot s}}{13\cdot s+34\cdot s^2+31\cdot s^3+9\cdot s^4+1}$
$7:\dfrac{(5-2\cdot s)\cdot e^{-1\cdot s}}{5\cdot s+1}$	$17:\dfrac{(3\cdot s+s^2+2)\cdot e^{-s}}{5\cdot s+4\cdot s^2+1}$	$27:\dfrac{(7\cdot s+4\cdot s^2+3)\cdot e^{-5\cdot s}}{10\cdot s+15\cdot s^2+6\cdot s^3+1}$	$37:\dfrac{(72\cdot s+99\cdot s^2+24\cdot s^3+5)\cdot e^{-4\cdot s}}{29\cdot s+223\cdot s^2+351\cdot s^3+144\cdot s^4+1}$
$8:\dfrac{(4-1\cdot s)\cdot e^{-1\cdot s}}{3\cdot s+1}$	$18:\dfrac{(2\cdot s+2)\cdot e^{-1\cdot s}}{3\cdot s+2\cdot s^2+1}$	$28:\dfrac{(13\cdot s+10\cdot s^2+3)\cdot e^{-4\cdot s}}{19\cdot s+37\cdot s^2+19\cdot s^3+1}$	$38:\dfrac{(13\cdot s+10\cdot s^2+s^3+4)\cdot e^{-1\cdot s}}{7\cdot s+15\cdot s^2+10\cdot s^3+2\cdot s^4+1}$
$9:\dfrac{2\cdot e^{-3\cdot s}}{5\cdot s+1}$	$19:\dfrac{(2\cdot s+3)\cdot e^{-3\cdot s}}{6\cdot s+4\cdot s^2+1}$	$29:\dfrac{(7\cdot s+2\cdot s^2+2)\cdot e^{-1\cdot s}}{5\cdot s+6\cdot s^2+2\cdot s^3+1}$	$39:\dfrac{(8\cdot s+8\cdot s^2+s^3+2)\cdot e^{-3\cdot s}}{8\cdot s+20\cdot s^2+17\cdot s^3+2\cdot s^4+1}$
$10:\dfrac{2\cdot e^{-2\cdot s}}{3\cdot s+1}$	$20:\dfrac{(2\cdot s+3)\cdot e^{-4\cdot s}}{7\cdot s+6\cdot s^2+1}$	$30:\dfrac{(5\cdot s+3\cdot s^2+2)\cdot e^{-2\cdot s}}{7\cdot s+9\cdot s^2+3\cdot s^3+1}$	$40:\dfrac{(28\cdot s+29\cdot s^2+7\cdot s^3+6)\cdot e^{-3\cdot s}}{23\cdot s+54\cdot s^2+41\cdot s^3+9\cdot s^4+1}$

$41:\dfrac{8\cdot s+7\cdot s^2-1\cdot s^4+2}{11\cdot s+35\cdot s^2+44\cdot s^3+23\cdot s^4+4\cdot s^5+1}\cdot e^{-5\cdot s}$	$42:\dfrac{10\cdot s+2\cdot s^2-7\cdot s^3-2\cdot s^4+4}{11\cdot s+31\cdot s^2+32\cdot s^3+11\cdot s^4+s^5+1}\cdot e^{-2\cdot s}$
$43:\dfrac{26\cdot s+70\cdot s^2+64\cdot s^3+18\cdot s^4+s^5+2}{20\cdot s+91\cdot s^2+148\cdot s^3+95\cdot s^4+20\cdot s^5+1}\cdot e^{-1\cdot s}$	$44:\dfrac{35\cdot s+91\cdot s^2+83\cdot s^3+22\cdot s^4+s^5+4}{13\cdot s+56\cdot s^2+95\cdot s^3+59\cdot s^4+6.0\cdot s^5+1}\cdot e^{-1\cdot s}$
$45:\dfrac{8\cdot s+8\cdot s^2+2\cdot s^3+2}{8\cdot s+20\cdot s^2+20\cdot s^3+8\cdot s^4+s^5+1}\cdot e^{-1\cdot s}$	$46:\dfrac{36\cdot s+18\cdot s^2-58\cdot s^3-16\cdot s^4+4}{23\cdot s+158\cdot s^2+361\cdot s^3+270\cdot s^4+61\cdot s^5+1}\cdot e^{-1\cdot s}$
$47:\dfrac{18\cdot s+16\cdot s^2-33\cdot s^3-48\cdot s^4-14\cdot s^5+3}{14\cdot s+66\cdot s^2+132\cdot s^3+109\cdot s^4+27\cdot s^5+1}\cdot e^{-1\cdot s}$	$48:\dfrac{11\cdot s+18\cdot s^2+9\cdot s^3+s^4+2}{9\cdot s+27\cdot s^2+34\cdot s^3+18\cdot s^4+3\cdot s^5+1}\cdot e^{-2\cdot s}$
$49:\dfrac{46\cdot s+90\cdot s^2+62\cdot s^3+13\cdot s^4+7}{13\cdot s+53\cdot s^2+84\cdot s^3+49\cdot s^4+9\cdot s^5+1}\cdot e^{-1\cdot s}$	$50:\dfrac{29\cdot s+70\cdot s^2+64\cdot s^3+17\cdot s^4+4}{24\cdot s+107\cdot s^2+180\cdot s^3+127\cdot s^4+31\cdot s^5+1}\cdot e^{-7\cdot s}$

control system with a PI controller tuned according to the ZN-1 method and the exact model do not differ significantly both in the time domain (transient processes) and in the frequency domain, when both the exact model and its FOPDT equivalent are connected. In the time domain, we will choose an integral criterion by which the ZN-1 method is close to the optimum (IAE for the output variable), as well as such direct performance indicators as overshoot, settling time, and damping. We choose standard frequency criteria, namely phase margin and amplitude margin. For comparison with the estimated model obtained by our method proposed in step 4, the Half-Rule (HR) reduction methods [1] and reduce (MATLAB function) were used. These methods are ideologically different.

The HR method is used only for models without complex conjugate poles. The method extracts the most significant pole, the second most significant pole, the remaining poles and the remaining zeros from the model written in zpk form. In simple terms, the model retains the significant pole, half the second significant pole, and the remaining poles and zeros form the delay. Since this can result in a negative delay, its full implementation involves the use of additional reduction rules and the solution of the optimization problem. The full software implementation of SIMC 2010 from Mathworks [9] was used.

The "reduce" function uses the balancing and modal truncation method (balancmr). This method is based on the concept of controllability and observability of linear systems. Those states of the model that are more observable and controllable are considered significant. Gram matrices are used for evaluation, providing a quantitative estimate of the energy required to control and measure each state. If the original model has a delay, the model without delay is reduced and the delay of the original model is added to the reduced model.

Figure 1 shows the deviations of the parameters of the FOPDT models obtained by two methods from the FOPDT model obtained by our method.

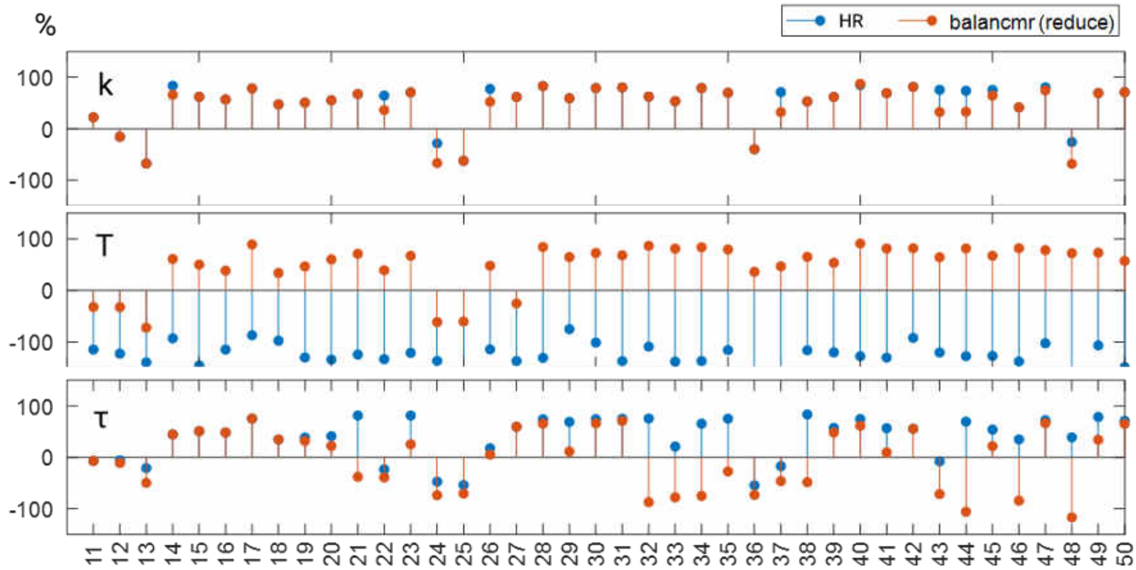

Fig. 1. The relative deviation of the parameters of the FOPDT models obtained by both reduction methods from the parameters of the model estimated by our method. The Y-scale is logarithmic, the X-scale shows the model numbers from Table 3.

The purpose of forming the FOPDT model in our study is the most adequate tuning of the PI controller and the coincidence of the dynamic properties of the model in the control system with the conditionally "accurate" and reduced models. This criterion is empirical, so we cannot formulate and compare the "best" FOPDT model as a reference point. The graphs show the following trends. HR and the "reduce" function generally tend to use higher gain. HR always gives a larger time constant and reduce a smaller one. Regarding the delay, it is impossible to visually identify a general pattern, except that the delays of the second-order models are close on average.

The results of the experiment comparing the performance indicators of the studied control systems with the full model and our FOPDT model are shown in Fig. 2.

An experiment with closing the PI controller with tuning obtained from the original model using ZN-1 and with the reduced models showed that the HR method gives an unsatisfactory approximation in three cases: the 21th model is close to the point of loss of stability (phase margin 2°); the 23rd model does not meet all the criteria by several times; the 38th model is unstable.

The values of the indicated models obtained by the HR method are not shown in the graph.

The models obtained with the "reduce" function and the evaluated models were always stable in the closed-loop control system.

The first 10 models, since they are already FOPDT and therefore cannot be reduced, are compared only with the estimated models.

The analysis of the data in Fig. 2 allows us to conclude that

1) Although in almost all cases the use of a reduced model leads to a deterioration in performance according to the IAE criterion, the use of an estimated model does not deteriorate the IAE by more than 10%.

2) Ideally, the correspondence between the PM and GM ratios should be close to one. Using the indicator. For higher order models, the models generated by the reduction function may lead to inadequate estimates, especially of the amplitude range.

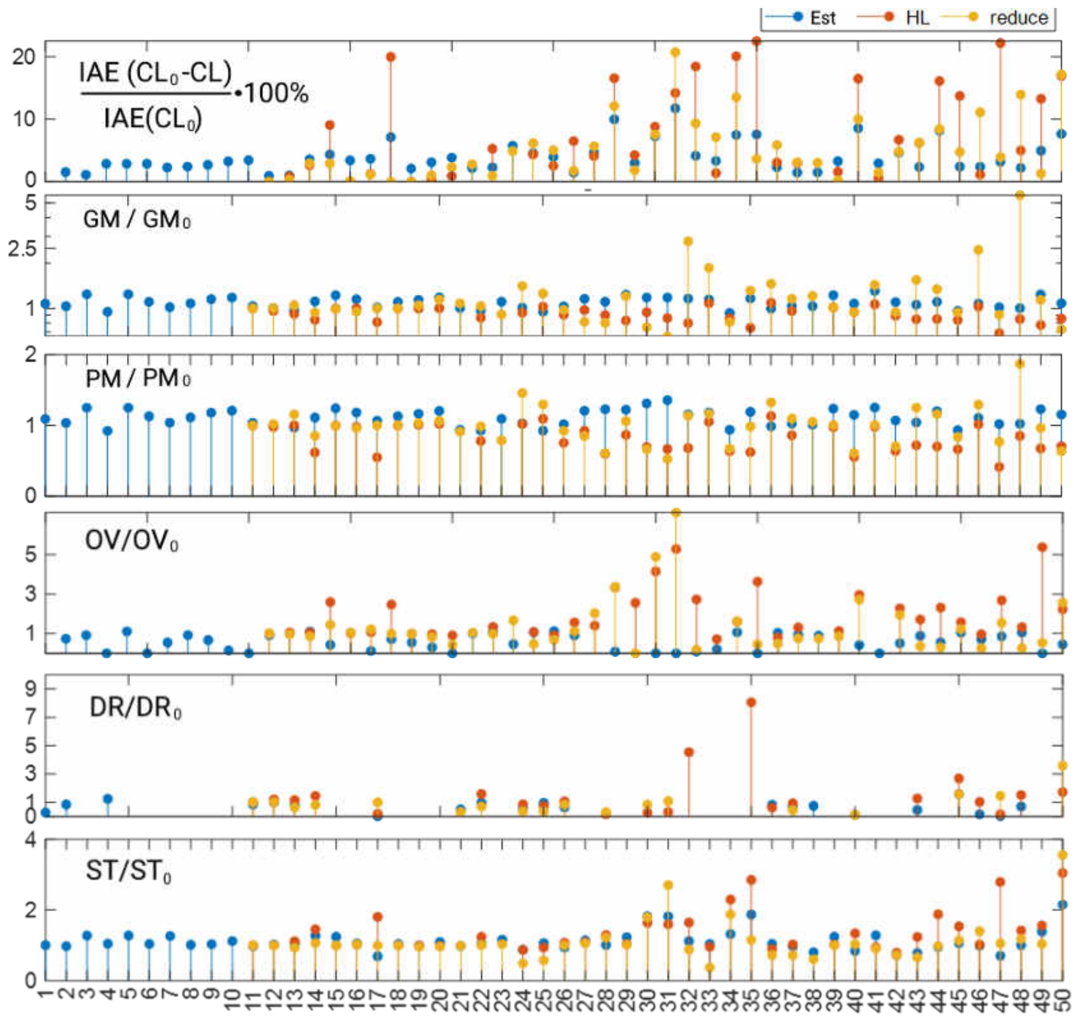

Fig. 2. Comparison of control system performance indicators (with full-order model and various FOPDT models). Notations: CL-model of closed-loop control system, GM-gain margin, PM-phase margin, OV-overshoot, DR-decay ratio (damping), ST-settling time, index "0" means that the indicator is related to the original full-order model.

3) A relatively reliable indicator, the settling time, is on average equal to the settling time of the reference control system. On the average, the ratio of the time to establish control systems using the evaluated model to the time to establish the reference control system is 1.12, using the reduced model 1.10, using the HR model 1.36.

4) The use of any reduced model in the control system does not allow to accurately reproduce the overshoot. The use of the estimated model underestimates, on average, the amount of overshoot, while the use of other models leads to a multiple increase in the amount of overshoot.

In general, the use of the estimated model reproduces the stability margins and the value of the IAE criterion of the reference control system quite accurately. With some caution, the estimated model can be used to estimate the settling time.

However, it is not recommended that the estimated model be used to evaluate overshoot and damping.

STAGE 5. PI CONTROLLER TUNING SEARCH THROUGH THE OPTIMIZATION PROCEDURE USING THE ESTIMATED MODEL PARAMETER.

The known drawbacks of the ZN method for PI controllers are low robustness, excessive oscillation, and excessive settling time. At the same time, the ZN method provides good initial settings for optimization. One of the most important works on the optimization of PI controllers is that of [10]. Having chosen the ZN method as a standard, the authors consider the use of the maximum norm of the main sensitivity function M_s of the control system as an optimization criterion. The M_s criterion is essentially the inverse of the minimum distance of the curve from the critical point on the Nyquist plot for the control system. The minimum value $M_s = 1$ means maximum robustness of the control system, and the maximum value $M_s = 2$ means low robustness but high speed of processes in the control system. Assuming that the model is monotonic, the optimization problem with respect to M_s is reduced to the solution of an algebraic equation. However, for systems with delay, the calculation is performed with an error, since the delay must be approximated. It is known that for a control system with a PI controller and an FOPDT plant with a non-dominant delay, the M_s criterion is associated only with K_p, but not with K_i, and minimization leads to a controller close to the integral one. Achieving a certain specified value of M_s is not associated with such direct performance indicators as overshoot and settling time. Therefore, the optimization problem must be multicriteria. The problem of minimizing the convolution of all sufficiently independent criteria was formulated in [11]

$$J(K) = w_1 M_p + w_2 t_r + w_3 t_s + w_4 E_{ss} + \int_0^{t_{ss}} (w_5 |e(t)| + w_6 u^2(t))dt + \frac{w_7}{PM} + \frac{w_8}{GM} \tag{6}$$

where M_p is overshoot, t_r is a rise time, t_s is a settling time, E_{ss} is a steady state error.

However, such an optimization problem does not lend itself well to standard algorithms and requires either the use of heuristic algorithms or complex partitioning, and the weighting coefficients must be different in each specific case for the problem to have a solution. We suggest using the following criterion

$$J(K_p, T_i) = \max_{\Delta} \begin{bmatrix} w_1 \cdot \text{ITAE} \\ + w_2 \cdot (y_{max} - r) \\ + w_3 \cdot M_s \end{bmatrix} \tag{7}$$

$w_1 = 10, w_2 = 1, w_3 = 10$.

By $\max_{\Delta}$ we mean the maximum value when the parameters k, T, τ deviate by ±20%, which is the maximum expected error in the FOPDT model approximation. If M_s is associated mainly with K_p, then ITAE establishes a balance between K_p and T_i, providing the desired speed. Like ISE and IAE, the ITAE index provides a fairly smooth convergence when using the simplex optimization method, but leads to minimal robustness. By minimizing over M_s and ITAE simultaneously with the same weighting coefficients, we strive to achieve some compromise between speed and robustness. The second parameter is insurance against the fact that the ITAE criterion introduces some oscillation and tends to greedy control. As a reference step response, we will consider a process with no more than 20% overshoot and a phase margin close to 60°.

Note that the criterion is almost linear depending on the ratio of K_p and T_i if we remove the requirement of parametric robustness, but this requirement clearly improves the quality of control when regulating nonlinear plants. As mentioned earlier, not only oscillation and damping, but also the stability margin depends on the value of τ / T . The ZN method has an approximate optimum at τ/T =0.5. Consider three FOPDT plants with different values of τ / T

$$\begin{aligned} P_1(s) &= 2 \cdot e^{-1s} / (5s+1), \\ P_2(s) &= 2 \cdot e^{-2.5s} / (5s+1), \\ P_3 &= 2 \cdot e^{-4.5s} / (5s+1) \end{aligned} \tag{1}$$

The main quality indicators obtained when using the tabular version of the ZN method for FOPDT plants (ZN-2) and optimized settings are given in Table 4.

In general, we see that the proposed optimization criterion for the PI controller is suitable for the problem under consideration. Based on the parameters of the estimated FOPDT model and using the starting tunes obtained using the ZN method, it is possible to obtain a robust and fairly fast PI controller with non-critical overshoot. The optimization criterion is quite undemanding in terms of the complexity of the optimization algorithm.

Table 4.

Results of optimization by criterion (7) for models (8).

Model P	Method	ST	OV	GM	PM
1	ZN-2	12.02	58.99	1.62	28.91
	opt	5.19	2.14	3.23	63.05
2	ZN-2	26.07	21.31	1.89	51.58
	opt	12.38	2.32	3.07	63.08
3	ZN-2	66.22	0	2.25	85.97
	opt	31.55	0.9	2.71	65.49

STUDY OF SOFTWARE IMPLEMENTATION OF THE METHOD ON SISO LINEAR MODELS.

Let's conduct an experiment on choosing settings using the developed program for a plant specified by the models from Table 3

We selected two dynamically different models: #14 (u_1-y_1) and #41 (u_2-y_2) with u_{min}=-0.5, $u_{max} = 0.5$, $Ts = 0.1$ and measurement noise intensity $0.01 / T_s$.

Transient processes when performing the second and fifth stages of the auto-tuning procedure are shown in Figs. 3. and 4.

The results are summarized in Table 5.

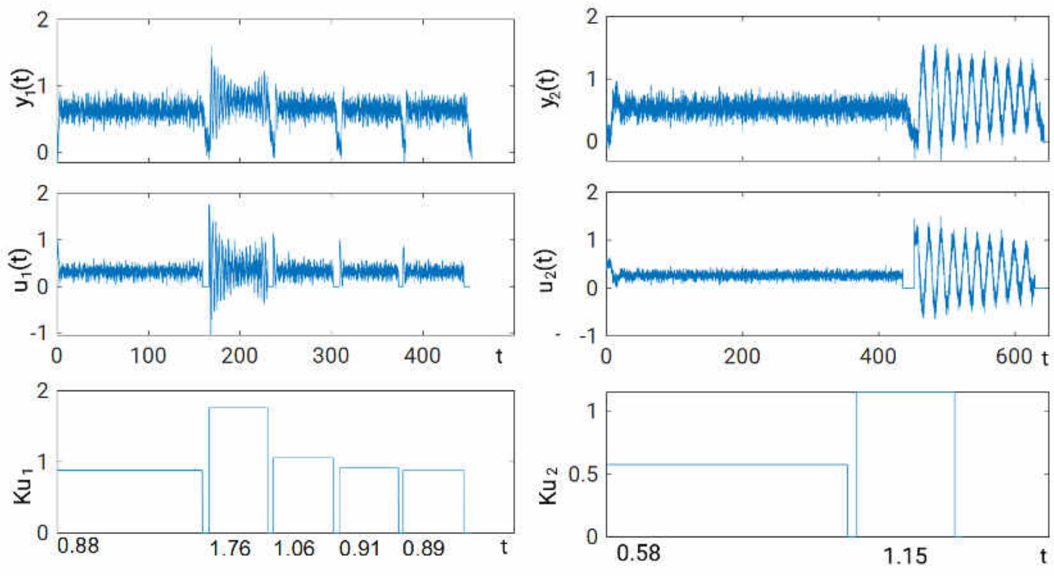

Fig. 3. Determination of K_u for models No. 14 and No. 41 (Table 3).

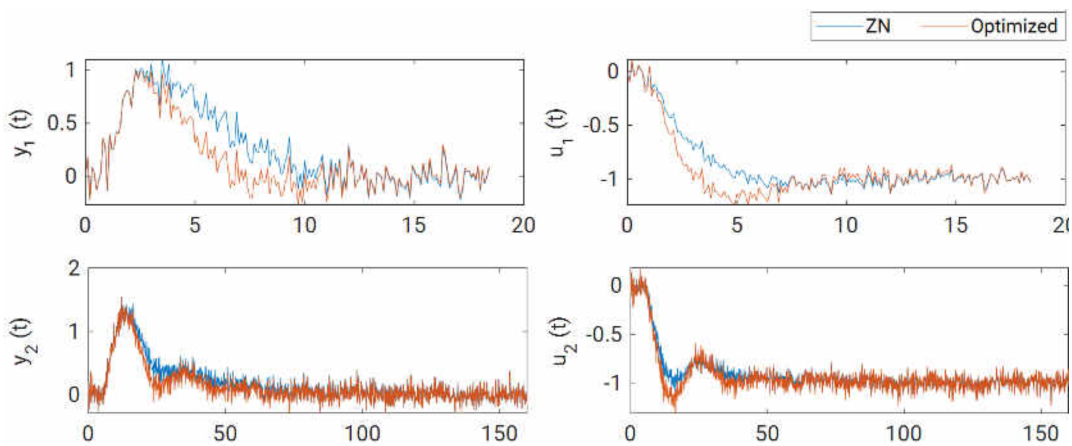

Fig 4. Transitions of the control systems with models No. 14 and No. 41 (Table 3) after load step disturbance. "ZN" means tuning using the ZN-1 method based on K_u value, "Optimized" means tuning using the proposed auto-tuning method.

Table 5.

Obtained tuning of controllers

Model P	Estimated model	K_u	P_u	C_{ZN}	C_{opt}
№14	$\frac{2.036 \cdot e^{-0.693 \cdot s}}{1.37\ s+1}$	0.8831	2.3	$K_p = 0.4014, T_i = 1.9167$	$K_p = 0.511, T_i = 1.44$
№41	$\frac{2.093 \cdot e^{-4.94 \cdot s}}{12.82\ s+1}$	1.1521	17.8	$K_p = 0.5237, T_i = 14.83(3)$	$K_p = 0.664, T_i = 13.9$

From Figs. 3, 4 it is clear that the controllers obtained from the evaluation model, included in the control system with original models No. 14 and No. 41 and noise, give satisfactory transient processes, while the controller configured according to the proposed method gives the best quality compared to a controller tuned using the ZN method.

USING THE METHOD FOR A NONLINEAR PLANT IN MANUAL MODE

The autotuning method can also be used in manual mode. For this purpose, it is recommended to select the tuning by the classical ZN-1 method, to evaluate the model by this method and to retune the controller by solving an optimization problem or by using rules. Let's consider the problem of controlling a chemical reactor [12-14], in which acetic acid (A) reacts with ethanol (B). The result is a solution of ethyl acetate (C) and water. The purpose of the control is to stabilize the concentration of C. The reactor (Fig. 5) is represented by a nonlinear model of the form

$$m = \left[\frac{\rho_A}{S \cdot \rho_W} \cdot F_{Ai} + \frac{\rho_B}{S \cdot \rho_w} \cdot F_{Bi} \right], n = k_o e^{\frac{-E}{RT}} C_A \cdot C_B$$

$$\dot{h} = m - \frac{r}{S}\sqrt{h}$$

$$\dot{C}_A = \frac{C_{Ai} F_{Ai}}{S} \frac{1}{h} - m \frac{C_A}{h} - n$$

$$\dot{C}_B = \frac{C_{Bi} F_{Bi}}{S} \frac{1}{h} - m \frac{C_B}{h} - n \qquad (9)$$

$$\dot{C}_C = -m \frac{C_C}{h} + n$$

$$\dot{T} = \frac{\rho_A \cdot cp_A \cdot F_{Ai}}{\rho_W \cdot cp_W \cdot S} \cdot \frac{(T_{Ai} - T_{ref})}{h} + \frac{\rho_B \cdot cp_B \cdot F_{Bi}}{\rho_W \cdot cp_W \cdot S} \cdot \frac{(T_{Bi} - T_{ref})}{h} - m \frac{(T - T_{ref})}{h} + \frac{(-\Delta h_r)}{\rho_W\, cp_W} n$$

The designations of the variables in the system of equations (9) are given in Table 6. A feature of the reactor is its sensitivity to changes in the concentration of the input streams, so the concentrations should not change and the components must be supplied in a strict ratio. A change in the temperature of the input streams is considered a disturbance that the control system must cope with.

Table 6.

The designations of the variables in the system of equations (9)

Symbols.	Descriptions	Units
C_{Ai}, C_{Bi}	Input concentrations A, B	$\frac{mol}{m^3}$
C_A, C_B, C_C	Output concentrations A, B, C	$\frac{mol}{m^3}$
cp_A, cp_B, cp_W	Specific heat capacities of A, B and water	$\frac{J}{kg \cdot K}$
ρ_A, ρ_B, ρ_W	Densities of A, B and water	$\frac{kg}{m^3}$
T_{Ai}, T_{Bi}	Input A, B temperatures	K
T	Output temperature	K
F_{Ai}, F_{Bi}	Input flows A, B	$\frac{m^3}{s}$
F	Output flow	$\frac{m^3}{s}$
h_A, h_B, h_C, h_W	Specific enthalpies of A, B, C and water	$\frac{kJ}{mol}$
Δh_r	$(h_W + h_C - h_B - h_A) \cdot 1000$	$\frac{J}{mol}$
S	Reactor area	m^2
h	Reactor level	m
k_0	Factor	$\frac{m^3}{mol \cdot s}$
E	Activation energy	$\frac{J}{mol}$
R	Gas constant	$\frac{J}{mol \cdot s}$
T_{ref}	Temperature at which h_i=0	K
r	Output valve resistance	$\frac{m^{2.5}}{s}$

The following model parameters were used for modeling:

$c_{Ai} = c_{Bi} = 1000; c_{pA} = 2069.9417; c_{pB} = 2419.3618; A = 2;$

$\Delta_{hr} = -690; c_{pW} = 4178.1909; \rho_A = 1050; \rho B = 789; \rho_W = 1000;$

$ko = 1000, E/R = 5458.2632, r = 0.894, T_{Ai} = T_{Bi} = 400.$

Initial values:

$y_1 = C_A = 682.9949686543955067,$

$y_2 = C_B = 243.6276575823393102$,
$r = y_3 = C_C = 49.2838831323648137$,
$y_4 = h = 0.5833268771676953$,
$y_5 = T = 350.6083211506618795$.

All changes in thermodynamic parameters can be equated to changes in temperatures. Nominal inputs: $F_A = 0.5,\ F_B = 0.2$. The control action of the controller is converted into percentages with a nominal value of 50%. Then $F_A = u{\cdot}0.01/(3s+1),\ F_B = u{\cdot}0.004/(3s+1)$. The concentration sensor Cc is modeled with a delay of $e^{-10\cdot s}$. For researching in Simulink, model (9) is implemented as a block with a Level-2 function.

The model of the control system with Level 2 function block and P controller in self-oscillating mode is shown in Fig. 6.

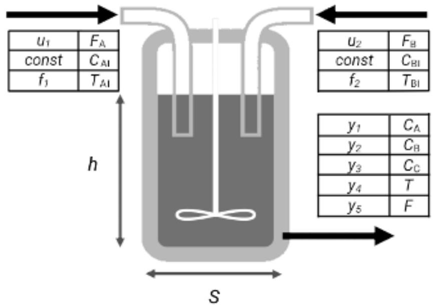

Fig. 5. The reactor scheme.

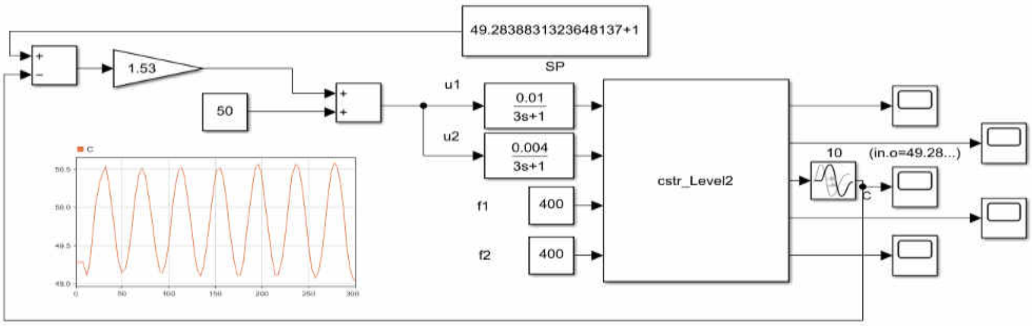

Fig 6. The model of the control system with Level 2 function block and P controller in self-oscillating mode.

Thus, we see that the critical coefficient K_u = 1.53, the interval between peaks P_u = 42.31. Using formula (1) we obtain $K_p = 0.7,\ T_i = 35.26$. But, we get a process with oscillation without overshoot. We increase K_p by 1.5 times and get 10.1% overshoot. Using formulas (2)-(5) we obtain the estimated model $P_e = 1.08{\cdot}e^{-11.73s}/(16.39{\cdot}s+1)$. Having carried out optimization according to criterion (6), we obtain the tunes $K_p = 0.75,\ T_i = 19.17$. The pidtune program gives $K_p = 0.8888,\ T_i = 20.1$. Control system step responses according to the reference and disturbance are shown in Fig. 7. As can be seen, in general, the use of the proposed optimization criterion makes it possible to achieve a higher quality of transient processes. The use of tunings from pidtune also significantly improves the performance in comparison with the ZN method, however, in this case it loses somewhat to the PI controller with tunings optimized according to the proposed criterion.

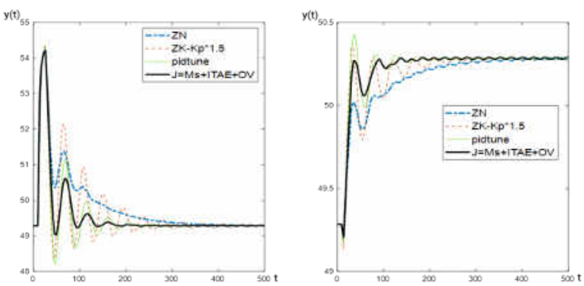

Fig. 7. The disturbance (left) and reference (right) control system step responses.

DISCUSSION

The impossibility of obtaining an accurate model for controller design in the power industry and many other industrial automation applications is well known. However, in general, modern control theory tries to take into account the inaccuracy of the model from the point of view of robustness [15, 16]. If we accept that any of the reduced FOPDT models obtained from the models in Table 3 is accurate, then when designing control systems, one should require high robustness to multiple deviations of model parameters. For example, in [17] it is shown how to find the zone of permissible tunings of the PI controller with an explicitly specified uncertainty of the FOPDT model parameters using maximum sensitivity. And in [18] it is shown that the ratio τ / T and M_s determine the parametric robustness of a control system with a PI controller. A decrease in M_s leads to a significant increase in the margin for k and τ, but not for T (for a plant with almost no delay, the allowable uncertainty is 61.2% at $M_s = 2$ and 79.4% at $M_s = 1.4$). At M_s =1.4, the allowable uncertainty in k reaches on average more than 300% and does not depend much on τ / T. However, if the model from [20] is considered from the point of view of stability to synchronous parameter changes, the stability limits will be smaller. For the model $c{\cdot}e^{-c{\cdot}s} / \left(10{\cdot}s / c + 1\right)$ with a nominal c =1 and a PI controller at M_s =1.2, the maximum permissible value is c=2.3. If we extend the context of uncertainty not only to parametric, but also to structural, then the problem of reserves can become more acute, as shown at the beginning of the article using the example of the FOPDT approximation of the oscillatory link $15 / \left(s^2 + 10s + 290\right)$. In general, if we ask the question in the context of a controller for significant model uncertainty, then the PI controller is not the best choice due to the fact that the integral term imposes significant constraints on robustness. In this approach, it is more appropriate to use a modified PI-like controller with the introduction of a delay (delay or aperiodic link with a tuning parameter) in the integral term [18].

Assuming that an accurate high-order model is available, the problem of model reduction must be solved in order to effectively tune the PI controller. As shown in [19], a reasonable reduction criterion can be the L_2 (Euclidean) norm of the model error when estimating the transient process from a pulsed disturbance. However, this approach requires a rather complex optimization problem. The L_2 norm has been widely used in machine learning techniques to control or reduce model complexity by limiting the size of the coefficients, thereby reducing the risk of overfitting. In [8], a good approximation for a transient process with a given disturbing signal is obtained by using a genetic algorithm. Against the background of computationally complex methods, the proposed method for obtaining an estimation model itself looks like a much simpler way to obtain a FOPDT model, and the tuning of the controller can be reduced to a simple optimization problem. The problem for pure mathematization without simulation is to estimate and achieve the required overshoot in a control system with a PI controller and arbitrary high-order plants. The problem of finding an optimal model can also be formulated as an adaptive control problem [21]. Starting with a working controller, the search procedure becomes dual, i.e., the model is refined, then the controller is retuned, and so on until the error is minimized. The problem of minimizing the error is solved by introducing a filter that contains the characteristic polynomial of the closed system. We can consider two minimization problems: by the error of the model equations (in this case the problem is reduced to minimizing the L_2 error norm) and by the error of the performance criterion. Obviously, convergence can be guaranteed if the model is structurally well chosen. In [20] the problem of autotuning the PID controller is formulated algorithmically. It is based on an expert system for a UNIX-like operating system. The purpose of the software solution is to automatically design the control loop and subsequently diagnose the loop. The general idea is that it is necessary to estimate the amount of noise, select, based on this, the relay test, form a transient process based on the disturbance and, based on an assessment of the plant gain, decide whether to use the tuning method (considered which is not worth it if $K_p - K > 1.2$) and use the improved ZN rule for the PID controller with task weighting and factor integral coefficient (correcting the tuning results for plants with a dominant delay). In general, the work is devoted to the problems of software and hardware implementation that are not currently relevant. At the same time, issues such as the choice of relay test, noise assessment and control loop diagnostics are not disclosed. The key idea of the method is to replace the original ZN formulas with slightly different ones, which are positioned as the best. The formulas of the ZN method have been revised several times to reduce overshoot. The current list of modifications is given in [21]. However, it is difficult to derive the best modification. First, the IAE criterion under consideration is the balance between overshoot and control time. For example, the Tyreus-

Luyben rule [2] achieves its objective at the cost of a significant increase in settling time. And secondly, for systems with more complex dynamics than FOPDT, it is characteristic that when ZN is used, there is no overshoot at all. Let us note the work [22], in which the problem of choosing between the original ZN and its modifications is studied using machine learning technologies, due to the complexity of forming an empirical criterion for the best rule for a specific plant. A similar problem has been considered in [23], but the solution is slightly different. The focus was mainly on obtaining a PID controller whose output is obtained by the direct synthesis method. The author assumes that the error of the relay test can reach 20%, so he uses the Fourier series expansion of the transient process in the self-oscillation mode. The main focus of the work is the estimation of the magnitude of the delay, for which an iterative procedure is used to calculate the estimated delay using a constructed neural network. In the modeling, measurement noise, which naturally degrades the quality of the algorithm for estimating all parameters, was not taken into account and an analog controller was used.

CONCLUSIONS

The paper proposes an algorithm that can be the basis for complex solutions in the field of software for automatic configuration of simple SISO control systems in industry. The algorithm is based on the formulas proposed by the authors for the evaluation of FOPDT models in a closed loop and on the optimization criterion proposed for the search of PI controller tunings for FOPDT models. Together with the implementation in software of automatic detection of control loop degradation, this direction of research can lead to a high degree of automation of the task of maintaining a large set of single-loop control systems based on PID family controllers, while obtaining acceptable control performance. One way to improve the algorithm could be the adaptive selection of a relay test depending on the nature of the disturbing noise. A detailed description of 8 ways to implement such a test on systems without significant delay is given in [24]. Another possibility is to extend the class of plants for which controller tunings are determined.

References

[1] Vilanova |R., Visioli A. (eds.) PID control in the Third Millennium. Advances in Industrial Control. London, Springer-Verlag, 2012.

[2] O'Dwyer A. Handbook of PI and PID Controller Tuning Rules (3rd Edition). London, Imperial College Press, 2009.

[3] Koelsch J. R. Tuning Tools Maintain Harmony in PID Loops. *AutomationWorld*, 2014. https://www.automationworld.com/products/software/article/13311005/tuning-tools-maintain-harmony-in-pid-loops

[4] Hang C., Aström K., Ho W. Refinements of the Ziegler-Nichols tuning formula. *IEE Proceedings-Control Theory and Applications*,1991, 138 (2): 111-118.

[5] Anderson B.D.O., Gevers M. Fundamental Problems in Adaptive Control. *Perspectives in Control*. Springer, London. doi:10.1007/978-1-4471-1276-1_2

[6] Samad T. A Survey on Industry Impact and Challenges Thereof [Technical Activities]. *IEEE Control Systems Magazine*, 2017, vol. 37, no. 1, pp. 17-18 doi:10.1109/MCS.2016.2621438

[7] Nishikawa H. A first-order system approach for diffusion equation. i: Second-order residual-distribution schemes. *Journal of Computational Physics*, 2007, vol. 227, no. 1, pp. 315–352.

[8] Yang, Z., Seested, G. T. Time-Delay System Identification Using Genetic Algorithm: Part Two: FOPDT/SOPDT Model Approximation. *Proceedings of the 3rd IFAC International Conference on Intelligent Control and Automation Science*, 2013, vol. 3, pp. 568-573. doi: 10.3182/20130902-3-CN-3020.00117

[9] Eryilmaz B. SIMC model reduce function. https://folk.ntnu.no/skoge/publications/2003/tuningPID/mathworks/

[10] Astrom J., Panagopoulos H., Hagglund T. Design of PI Controllers based on Non-Convex Optimization. *Automatica*, 1998, vol. 34, no.5, pp. 585-601.

[11] Zamani M., Karimi-Ghartemani M., Sadati N., Parniani M.. Design of a fractional order PID controller for an AVR using particle swarm optimization. *Control Eng Pract*, 2009, vol.17, no. 12, pp. 1380-1387.

[12] Ching R. Dynamic Simulation of a Non-isochoric & Non-isothermal CSTR, 2021. URL: https://www.mathworks.com/matlabcentral/fileexchange/89857-dynamic-simulation-of-a-non-isochoric-non-isothermal-cstr

[13] Atkins P. W., Paula J. D. Atkins physical chemistry: Molecular thermodynamics and kinetics (10th ed.). Oxford: Oxford University Press, 2014.

[14] Borovinskaya E, Khaydarov V, Strehle N, Musaev A, Reschetilowski W. Experimental Studies of Ethyl Acetate Saponification Using

Different Reactor Systems: The Effect of Volume Flow Rate on Reactor Performance and Pressure Drop. *Applied Sciences*. 2019; vol. 9, no.3, 532. doi: 10.3390/app9030532

[15] Bhat V.S., Thirunavukkarasu I., Shanmuga Priya S. Design of centralized robust PI controller for a multivariable process. *Journal of Engineering Science and Technology*, 2018, vol. 13, no. 5, pp.1253–1273

[16] Bhat V.S., Bhat S., Gijo E.V. Robust design of proportional integral controllers: A Taguchi-grey approach. *International Journal of Modelling Identification and Control*, 2020, vol. 35, pp. 363-375. doi:10.1504/IJMIC.2020.114791.

[17] Chu M., Xu C., Chu J. Graphic IMC-PID tuning based on maximum sensitivity for uncertain systems. Transactions of the Institute of *Measurement and Control,* 2019, vol. 41, no.8, pp. 2196-2204. doi:10.1177/0142331218775491

[18] Shamsuzzoha M. A unified approach for proportional-integral-derivative controller design for time delay processes. *Korean Journal of Chemical Engineering*. 2014, vol. 32, pp. 583-596. doi: 10.1007/s11814-014-0237-6

[19] Casagrande D., Krajewski W. Viaro U. L2–optimal FOPDT models of high–order transfer functions. *Asian Journal of Control*, 2017, vol. 19, no. 2, pp. 1–10 doi: 10.1002/asjc.1354

[20] P. K. Vue, T. H. Lee, C. C. Hang and W. K. Ho. Realization of an expert system based PID controller using industry standard software and hardware environments. *IFAC Intelligent Tuning and Adaptive Control (*Singapore), 1991, pp .327-333.

[21] Bucz Š., Kozáková A. Advanced Methods of PID Controller Tuning for Specified Performance. *PID Control for Industrial Processes,* 2018. InTech. doi: 10.5772/intechopen.76069

[22] Calvo-Rolle J. L., Corchado E., Quintian-Pardo H., García R. F., Román J. Á., Hernández P. A. A Novel Hybrid Intelligent Classifier to Obtain the Controller Tuning Parameters for Temperature Control. In Lecture Notes in Computer Science, 2012, Springer Berlin Heidelberg, pp. 677–689. doi:10.1007/978-3-642-28942-2_61

[23] Huang H.P., Jeng J.C., Lin F.Y. Model-Based Autotuning System Using ANN and Relay Feedback Test. *IFAC Proceedings Volumes*, 2004, vol.37, pp. 305-310. doi: 10.1016/S1474-6670(17)31828-1

[24] Hornsey S. A Review of Relay Auto-tuning Methods for the Tuning of PID-type Controllers *Reinvention: an International Journal of Undergraduate Research*, 2012, vol. 5, no. 2

About authors.

Stopakevych Andrii
PhD, Associate Professor. SUITT. Area of scientific interests: complex systems for automatic control of technological processes, software technologies in the field of industrial automation.
E-mail: stopakevich@gmail.com

Stopakevych Oleksii
PhD, Associate Professor, Senior Scientis. NOPU. Area of scientific interests: system analysis and theory of complex control systems, multivariable systems, cybersecurity problems, software development.
E-mail: stopakevich@op.edu.ua

Tigarev Anatolii
PhD, Associate Professor. SUITT. Area of scientific interests: automation of technological processes.
E-mail
a.m_tiharev@suitt.edu.ua

Vorobiova Olena
PhD, Associate Professor, SUITT. Area of scientific interests: control systems, electronics, software development.
E-mail:
e.m_vorobiova@suitt.edu.ua