HUNGARIAN JOURNAL
OF INDUSTRIAL CHEMISTRY
VESZPRÉM
Vol. 33 (1-2). pp. 81-88. (2005)

CONSTRAINED PI(D) ALGORITHMS (C-PID)

F. SZEIFERT, L. NAGY, T. CHOVÁN* AND J. ABONYI

Department of Process Engeneering, University of Veszprém,
Veszprém, Egyetem u. 10, H-8200, HUNGARY,
www.fmt.vein.hu, chovan@fmt.vein.hu

Majority of control algorithms used in industrial processes is PID or PID modification and many of these is badly tuned. The reason for this is that the physical constraints of the manipulated variable are neglected. The PID algorithm, presented in the paper, is obtained by inverting the standard PID twice and it is able to handle the constraints. The first analytical inverting step results in a proper PID inverse. This is then transformed into a state-space model. The state-space model is then inverted again by using the same method which is applied in Globally Linearizing Control and taking into account the physical constraints of the manipulated variable. The constrained PID (C-PID) algorithm obtained this way is an anti reset wind-up algorithm which can be readily implemented. A possible design methodology is also proposed. At the same time, regarding processes with not higher than second order dynamics, the solution a rigourous model-based one.

Keywords: constrained control, PID algorithm, model-based, constrained inverse

Introduction

Based on different surveys, 95% of control algorithms used in industrial processes is PID or PID modification and most of these is badly tuned. The consequence is that the dynamic performance is poor and in the worst cases even instability might occur. Correct tuning is made difficult by several problems which are at the same time the reasons for the gap between the control engineering practice and the control theory. Only a few of these are:

On the practical side:
- Dynamics of the process is known only roughly.
- Dynamic properties can change with time (valve sticking, wearness, etc.).
- The algorithm of the used PID modification is not known (because of the intellectual property rights, the documentation are non-algorithmic-level and superficial).
- For the above reasons the "academic" tuning methods cannot be applied.
- Industrial implementation of algorithms established in control theory encounters difficulties.

On the theoretical side:
- Most of the methods, thriving mathematical accuracy, start form assumption which are not satisfied in practice.
- Methods built on idealized models are preferred.
- Physical constraints are neglected (involving the constraints, the inherently linear models become non-linear).
- The methods "in focus" are favoured.

The paper defines a constrained PID algorithm which can be readily implemented in practice as well as discusses the limitations of PID-based algorithms and the possibilities of model-based design.

The Set of PID Algorithms

In spite of the two decades of industrial application and the intensive academic research providing the

*Correspondence concerning this article should be addressed to T. Chovan (chovan@fmt.vein.hu)

theoretical bases, chemical processes are dominated by PID or PID-based controllers [1]. The main reasons for this dominancy are the role of PID controllers in the classical control technologies, their position in the engineering curriculum, their availability in DCS's and not at least the efficiency of their application. On the top of these, certain model-based techniques, depending on the process model, often result in PID algorithms and therefore can be implemented as PID controllers. Still the research and application of model-based control algorithms are rather important, first of all, in cases of processes where the application of PID is not efficient. The study of model-based control algorithms is getting more and more intensive as the technological possibilities are opening. At the same time the model of the controlled process gains more importance in the analysis.

The input of the PID algorithm is the control error (e), the output is the control signal (u), and its continuous time ($t \geq 0$) model is:

$$u(t) = K\left(e(t) + \frac{1}{T_I}\int_0^t e(\tau)d\tau + T_D\frac{de(t)}{dt}\right) + u_S\,, \qquad (1)$$

where

K, T_I, T_D are the parameters (gain, integral and differential time constants)

u_S is the steady-state control signal corresponding to $w(t)$ setpoint

Design of the controller involves the determination of the three parameters, while the value of u_S is often set to zero or sometimes to other constant (the I-term assures the settling without steady-state error). In case of more complex algorithms (e.g. for batch processes) the u_S can be estimated more accurately:

$$u_S = u_0 + FF(w, z, \ldots)\,, \qquad (2)$$

where

u_0 is constant (in batch processes it can be used for initialization in the different phases)

$FF(w, z, \ldots)$ is a feed-forward term based on the setpoint and the measured disturbance(s) ($z, \ldots$).

In the process control systems usually different (I) modifications are implemented. Transfer functions of the common solutions are the following:

Parallel PID (P-PID):

$$G_{PIDP} = \frac{u(s)}{e(s)} = K\left(1 + \frac{1}{T_I s} + \frac{T_D s}{\alpha T_D s + 1}\right), \qquad (3)$$

Serial PID (S-PID):

$$G_{PIDS} = K\left(1 + \frac{1}{T_I s}\right)\cdot\frac{T_D s}{\alpha T_D s + 1}\,, \qquad (4)$$

Filtered parallel PID:

$$G_{PIDF} = G_{PIDF}\cdot\frac{1}{T_F s + 1}\,, \qquad (5)$$

where

$\alpha \in [0.1, 0.5]$, constant

T_F is the time constant of the first order filter, it must be determined during the design.

The above controllers are in continuous time. Discretizing with an appropriate sampling time the corresponding discrete PID algorithms can be obtained. Using a sampling time, orders of magnitude less than the characteristic time constant of the process, the discrete PID approximate the results the corresponding continuous algorithm with the required accuracy. The time constant for the great majority of chemical processes is several orders of magnitude larger than the (hardware) sampling time of 100 msec or 1 sec, realized in the process control systems. In case of relatively high sampling time, the discrete PID algorithms require special analysis.

It is well known that in feedback loops, the zero steady-state error is maintained by the integrating term, therefore the I-term must included in most of the cases. At the same time, since the physical control signal is constrained, the application of the I-term can lead to saturation (wind-up) which is treated by different “backward integration” algorithms.

PID blocks of the process control systems allow realization of a large variety of PID modifications by using different configuration parameters. This solution, however, makes the correct application of PID algorithms more difficult in itself, since it may require the specification further several tens of parameters above the three or four tuning parameters.

Model-Based Algorithms

The fundamental problem of feedback control is that the effect of the actual control output – especially in case of higher order systems with dead-time – is delayed in time. The small change induces higher control output which ultimately can even cause instability. The mathematical model of the process allows estimating the future effect of the control output and this way determining the optimal output. The model predictive controllers (MPC) solving the optimal control problem over a discrete prediction horizon determine the optimal future values of discrete time control outputs. The first element is then realized and the calculation is repeated in every sampling period. Industrial application of MPC has two decades of history and software tools (e.g. RMPCT) for considerably supporting the design have been introduced. MPC superposed on PID loops can be efficiently used, first of all, for multivariable (MIMO) problems. In case of simple SISO problems the performance of MPC is comparable to that of a PID, however its calculation requirements and implementation cost can be significantly higher [2].

One of the simplest model-based design methods is the direct synthesis technique [3]. Its basic idea is that the dynamics of the closed loop is defined and the controller providing this response is calculated backward using the known process model. In case of simple process models, very often a PID variant, which can be readily implemented on any DCS, is obtained as a result. The results of the design for a few simple processes are summarized in Table 1, where the closed loop is defined as a first order filter (with dead-time) and T_C is the time constant of the closed loop.

Controllers applying the internal model control (IMC) principle are very popular in academic studies. Their essence is a feed-forward term containing the inverse of the process model. The control offset coming from the model error is corrected by feeding back the filtered model error. Depending on the process model, often a PID algorithm, which can be used in the classical feedback scheme, is obtained in this case too. Applying the IMC method on the processes in *Table 1* and using first order filters, the same results given in the table are obtained [3].

Investigating the results in Table 1 it can be concluded that up to second order systems the linear-model-based methods also result in PID algorithms. It is well known too that a large number of simple chemical processes can be modeled as first (or second) order system with dead-time. These facts support the widely accepted experience that a considerable part of chemical process control problems can be solved by different PID variants.

For systems with dead-time, the Smith predictor which can also be well inserted into IMC structures lives its renaissance. In case of batch systems it is practical to specify the PID algorithm by phases and often more complex solutions have to be applied (e.g. dual-mode control [4]).

Table 1 Model-based PID algorithms

Process model	***Direct synthesis or IMC***		
	$K \cdot K_C$	T_I	T_D
$\frac{K}{T\,s+1}$	$\frac{T}{T_C}$	T	0
$\frac{K}{(T_1\,s+1)(T_2\,s+1)}$	$\frac{T_1+T_2}{T_C}$	T_1+T_2	$\frac{T_1 \cdot T_2}{T_1+T_2}$
$\frac{K}{s}$	$\frac{1}{T_C}$	∞	0
$\frac{K}{s(T\,s+1)}$	$\frac{1}{T_C}$	∞	T
$K\frac{e^{-T_H s}}{T\,s+1}$	$\frac{T}{T_C+T_H}$	T	0
$\frac{K e^{-T_H s}}{(T_1\,s+1)(T_2\,s+1)}$	$\frac{T_1+T_2}{T_C+T_H}$	T_1+T_2	$\frac{T_1 \cdot T_2}{T_1+T_2}$

Investigating the results in Table 1 it can be concluded that up to second order systems the linear-model-based methods also result in PID algorithms. It is well known too that a large number of simple chemical processes can be modeled as first (or second) order system with dead-time. These facts support the widely accepted experience that a considerable part of chemical process control problems can be solved by different PID variants.

For systems with dead-time, the Smith predictor which can also be well inserted into IMC structures lives its renaissance. In case of batch systems it is practical to specify the PID algorithm by phases and often more complex solutions have to be applied (e.g. dual-mode control [4]).

Controller Design

The design of the control systems, in a broader sense, involves the selection of manipulated and measured variables based on the analysis of degree of freedom, sensitivity and dynamic behavior, as well as to select the control structure and method. More specifically the design means selecting the control algorithm and determining its parameters. This later, even now, is often solved by using classical methods (Ziegler-Nichols, Cohen-Coon, integral criteria, etc.) with appropriate computer aids and simulation tools. Based on simulation, the optimal parameters of the controller can also be found by different search methods. There are known several modified versions of the classical techniques. Model-based approaches (see e.g. Table 1), starting from different types of models, derive the control equations using the techniques of the linear control theory and applying suitable approximations (e.g. dead-time: Pade-approximation, nonlinearities: Taylor-series). In this case the identified process model and the control rule determine the control structure and the controller parameters too; separate tuning rules are not needed.

Building in the identification of the applied model, in the framework of classical schemes (gain scheduling, model reference, self tuning), adaptive algorithm can be constructed too. It is advisable to design the supervision of their operation in advance.

In the controller design several practical problems emerge, making more difficult the efficient application of academic results. Problems related to control valves, like hystheresis, sticking and nonlinear valve characteristic, are well known.

Since the problems of hystheresis and sticking must be solved by mechanical engineering techniques they are not considered in the design model. (Their indication, at the same time is a model-based diagnostic problem). Taking into account the strongly nonlinear valve characteristics is a prerequisite for the appropriate design.

Considering the practical controller design, an important element of the model is the allowable range of its variables, i.e. taking into account the related constraints. In mathematical sense, this changes the not constrained linear model into a nonlinear one and makes the detailed analysis more difficult (that is why it is often neglected in academic studies).

A number of publication confirm that using adequate models containing the corresponding constraints, the model-based algorithms are more efficient than PID controllers tuned with classical methods [5].

Constrained PI(D) Algorithm

Taking into account the physical constraints on the control variable the saturation (wind-up) effect can be eliminated. Especially in case of batch systems it is frequently occurs, that the requirement for fast settling generates such huge changes in the control output that cannot be realized. This may lead considerably high overshoots which prevent achieving good control performance. This was our main reason motivating the development a constrained PID variant.

To take the physical constraints of the control variable into account, two consecutive inverting of a standard PID algorithm is applied, as follows:

1. The inverse of a standard PID is formed in the transformed domain. This can be solved reciprocating the transfer function.

2. A constrained inverse of the inverse PID is formed after converting the inverse transfer function (in time domain) into a state-space model.

In details the following transformations are to be done. Using a P-PID controller ($\alpha = 0$) the starting transfer function is the following:

$$PID = \frac{u(s)}{e(s)} = K_C \left(1 + \frac{1}{T_I s} + T_D s \right), \tag{6}$$

that defines an improper object. Let us take its proper inverse:

$$PID^{-1} = \frac{e(s)}{u(s)} = \frac{T_I}{K_C} \cdot \frac{s}{1 + T_I s + T_I T_D s^2} , \tag{7}$$

Based on the transfer function, the inverse can be given as a time-domain input-output model:

$$T_I T_D \frac{d^2 e}{dt^2} + T_I \frac{de}{dt} + e = \frac{T_I}{K_C} \cdot \frac{du}{dt} , \tag{8}$$

Let us transform the input-output model into the following input-output equivalent state-space model (using the $x_1 = e, \quad x_2 = \frac{1}{T_I} \int e$ state definitions):

$$\frac{d\,x}{dt} = \begin{bmatrix} -\frac{1}{T_D} & -\frac{1}{T_D} \\ \frac{1}{T_I} & 0 \end{bmatrix} \cdot x + \begin{bmatrix} \frac{1}{T_D K_C} \\ 0 \end{bmatrix} \cdot u , \tag{9}$$

$$y (\equiv e) = [1 \;\; 0] \cdot \underline{x} , \tag{10}$$

The state-space model given by *Eq.(9-10)* is a proper inverse of a standard P-PID. The new C-PID algorithm is constructed by forming a constrained inverse of this model. To form a constrained inverse, let us consider the general scheme (*Fig.1*) of Globally Linearizing Control (GLC) [6]. The idea is that an originally nonlinear object can be transformed into a linear one by a state feedback compensator.

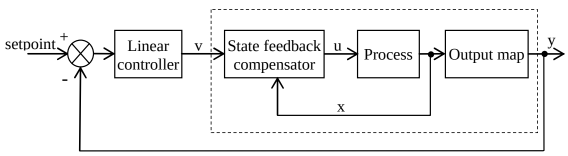

Fig.1 Globally linearizing control structure

The order of the linear input-output model, where the input is v, the output is y, is equal to the relative order of the state-space model, *Eq.(9-10)*. Based on the linearization technique, the constrained inverse is formed according to the scheme shown on *Fig.2* [7]. The variables are interpreted in the following way: the input of the inverse is the setpoint (w), its output is the manipulated variable (u). Let the relative order of of the state-space model, *Eq.(9-10)* be r. This means that the input of the process (w is not constrained, u is constrained) has a direct effect on the r-order derivative of the output ($d^r y / dt^r$). The not-constrained control output (v) is determined in such a way that the relationship between the setpoint (w) and the controlled variable (y) is defined by an r-order linear input-output model. The time constant of this linear model should be determined according to the time constants of the object given by *Eq.(9-10)*. Relatively small time constants result in aggressive interventions; the control output (v) is often reaches the physical constraints (in this case u takes its minimal or maximal value). With relatively large time constants the system capacity is not exploited resulting in slow control settlings.

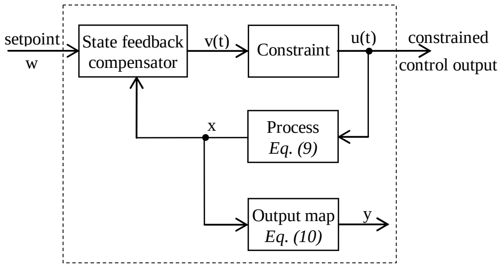

Fig.2 Formation of constrained inverse

To invert according to the given scheme, the the output of *Eq.(9-10)* is differentiated:

$$\frac{dy}{dt} = \frac{dx_1}{dt} = -\frac{1}{T_D} \cdot (x_1 + x_2) + \frac{1}{K_C T_D} \cdot u \,, \tag{11}$$

Since the first order derivative contains the control output explicitly, the relative order of the inverse PID is one. Hence the linear system can be defined in the following way:

$$T_F \cdot \frac{dy}{dt} + y = w \,, \tag{12}$$

Substituting *Eq.(11)* in place of the derivative, the value of the required control output is obtained (output of the feedback compensator):

$$v = K_C \cdot \left[\frac{T_D}{T_F} \cdot (w - x_1) + x_1 + x_2 \right], \tag{13}$$

The output constraints are treated as follows:

$$u = \begin{cases} v, & if \;\; v \in [u_{\min}, u_{\max}] \\ u_{\min}, & if \;\; v < u_{\min} \\ u_{\max}, & if \;\; v > u_{\max} \end{cases} \,, \tag{14}$$

where $u \in [u_{\min}, u_{\max}]$ is the physically allowed range of control output.

Summarizing the steps above, the scheme of the constrained PID (C-PID) algorithm can be constructed (see *Fig.3*). Initial values of the differential equations are set to zero error and to zero output difference.

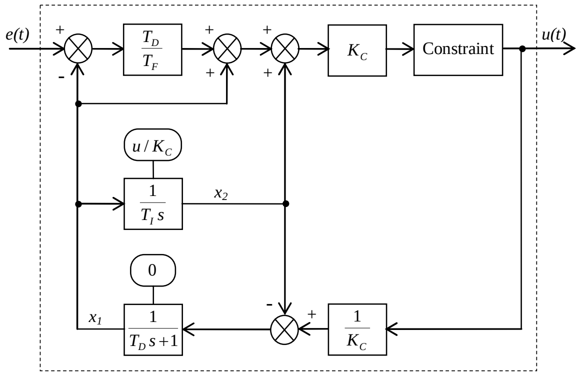

Fig.3 Scheme of the C-PID controller

Applying a similar reasoning or the limit value $T_D \to 0$ a C-PI algorithm can be elucidated too (see *Fig.4*). Here the relative order of the inverse is zero.

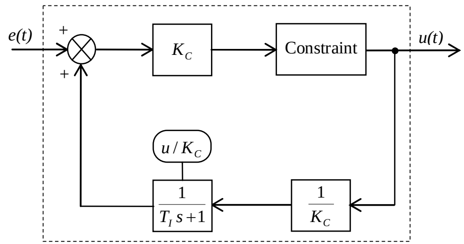

Fig.4 Scheme of the C-PI controller

The non-constrained transfer functions can be easily constructed and the following results are obtained:

C-PID:

$$G_{C-PID} = K_C \cdot \left(1 + \frac{1}{T_I\, s} + T_D\, s \right) \cdot \frac{1}{T_F\, s + 1} \,, \tag{15}$$

C-PI:

$$G_{C-PI} = K_C \cdot \left(1 + \frac{1}{T_I\, s} \right), \tag{16}$$

It can be seen that C-PID not reaching the constraints is equivalent to a parallel PID with a filter, *Eq. (5)*, while a C-PI to a normal PI controller. Hence it is clear that taking the constraints into account don't makes the basic algorithms more complicated. This fact has a great importance for practical realizations.

Design of Constrained PI(D) Algorithms

To determine the parameters of a C-PID it is practical to describe the controlled system as a second order object. The scheme of the closed loop for the non-constrained case is shown in *Fig.5*.

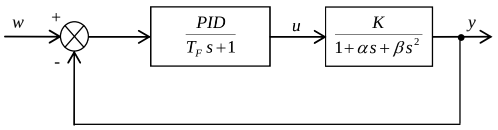

Fig.5 Non-constrained closed loop

This is equivalent to the closed loop given in Fig.6.

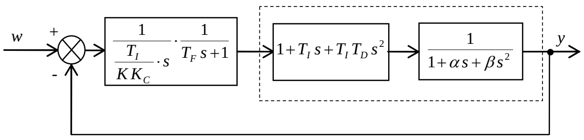

Fig.6 Equivalent loop

The framed part shows well that the controller compensates the dynamics of the process if the controller parameters are chosen according to the followings:

$$T_I = \alpha, \qquad T_D = \alpha / \beta\,. \tag{17}$$

The filter parameter T_F should be selected at the possible smallest value $(T_F \to 0)$ allowed by the measurement noises, and then setting the time constant of the closed loop to T_C, the controller gain can be given by the following expression:

$$K_C = \frac{\alpha}{K \cdot T_C}\,. \tag{18}$$

In the direct synthesis method the time constant of the closed loop is selected as a half or fifth of the time constant of the process, therefore the gain can be estimates as:

$$K_C = \frac{\gamma}{K}, \qquad \gamma \in [2,5]\,, \tag{19}$$

The two parameters of a C-PI controller can be determined also according to the above reasoning.

Testing of Constrained PI(D) Algorithms

The C-PID algorithm was physicaly tested on an electrical water heating system installed in our Process Engineering laboratory. The P&I diagram of the system is shown on *Fig.7;* its technical specification is given in an earlier publication [8]. The temperature of the water ($y \equiv T$) leaving the heater system is controlled by manipulating the heater performance ($u \in [0,10]$). The flowrate of the water and its feed temperature are considered as non-measured disturbances. The dynamic $u \to y$ relationship is chosen as a second order input-output model with dead-time that provides a structuraly adequate description. Open-loop experiments were conducted in order to determine the model parameters. The parameters were estimated by fitting to the measured data using Matlab (see *Fig.8*).

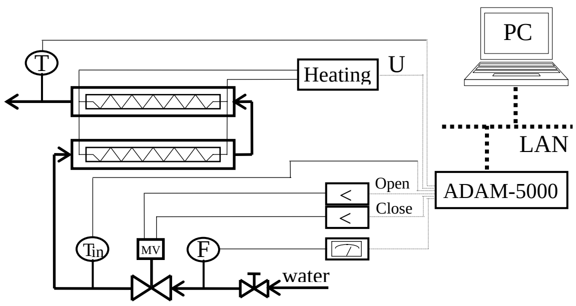

Fig.7 The laboratory system for testing

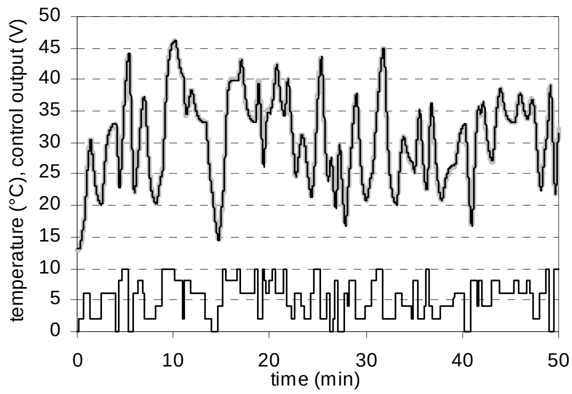

Fig.8 Identification of the process model

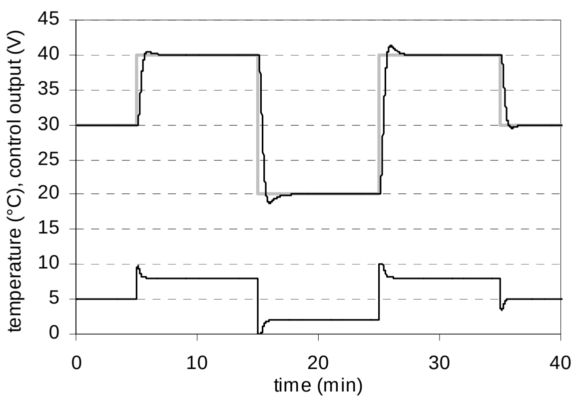

Fig.9 Simulation test of the P-PID controller

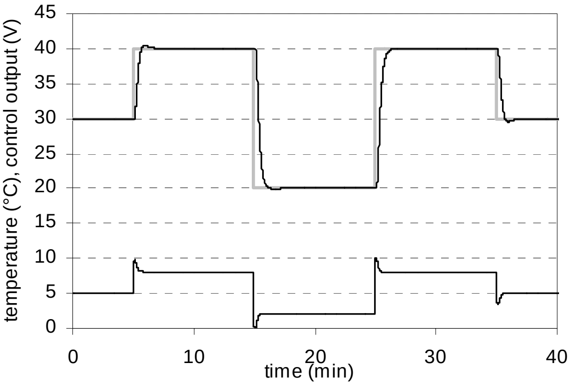

Fig.10 Simulation test of the anti wind-up PID controller

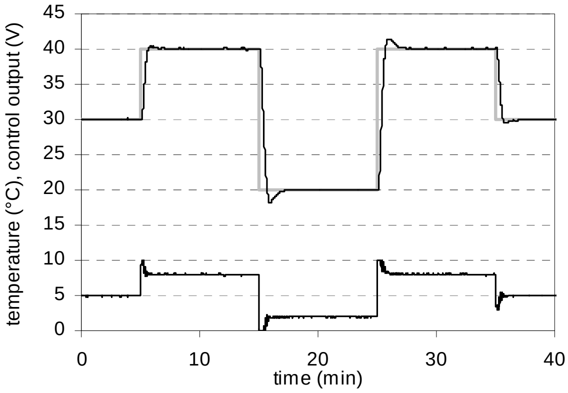

Fig.12 Physical test of the P-PID controller

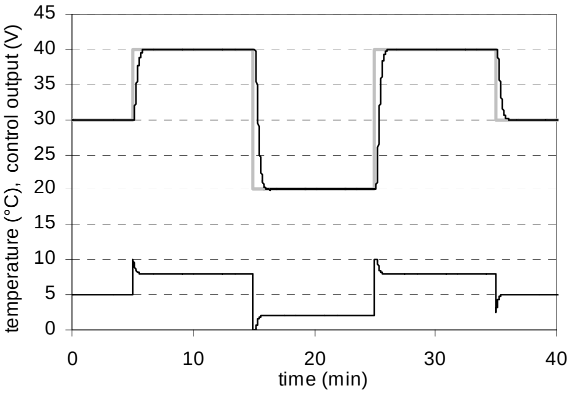

Fig.11 Simulation test of the C-PID controller

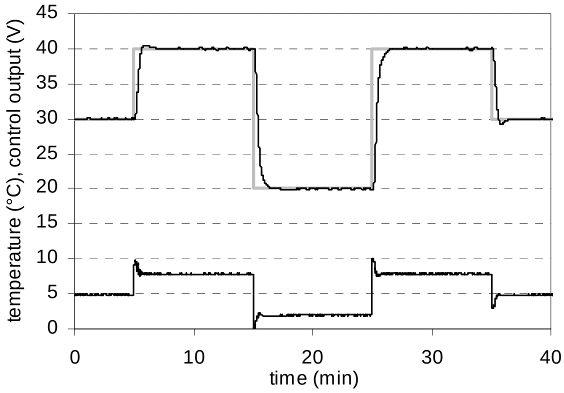

Fig.13 Physical test of the anti wind-up PID controller

The C-PID algorithm was compared to a standard P-PID algorithm as well as to an anti wind-up PID algorithm used in an industrial PLC. The PID parameters are determined in each cases by the direct synthesis method based on the identified process model. In the simulation studies the mathematical model of the heater system was the process. The studies presents servo problems, however the load disturnbance compensation studies qualitatively showed similar results. Simulation tests are illustrated on *Fig.9-11*. *Fig.9* shows well that in those time periods when the control output approaches its physical limits, significant overshoots can be observed after changing the setpoint. Overshoots can be considerably reduced by applying an anti wind-up compensator (*see Fig.10*). *Fig.11* justifies that the C-PID algorithm completely eliminates the overshoot.

The same tests were conducted on the laboratory physical system. The results are given on *Fig.12-14*. The physical experiments illustrates well the effect of measurement noises, still the relation of the different methods is the same in case of the physical tests as it was shown in the simulation studies.

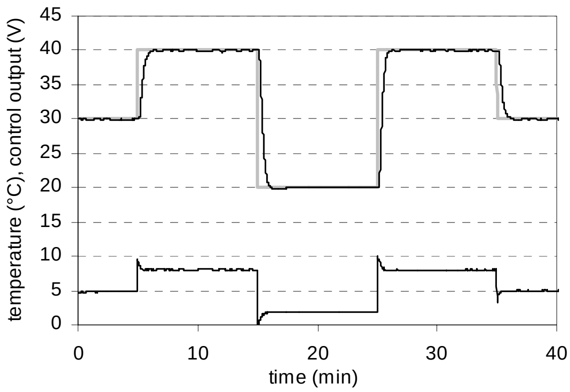

Fig.14 Physical test of the C-PID controller

Conclusions

In industrial applications several versions of PID controllers can be found. Because of the physical constraints on the control output only those supplemented with anti reset wind-up compensators can follow setpoint changes without overshoots. Significant overshoots can involve safety risk especially in control of batch systems. The paper presents the so called C-

PID algorithm which takes the physical constraints into account and provides settlings practically without overshoots. The algorithm does not make the standard PID algorithm more complex and it can be readily implemented in DCS's. For the C-PID design, considering the potential capacity of PID algorithms, it is practical to describe the object as a second order process with dead-time. In case of systems with large dead-times the use of a Smith predictor is suggested that does not limit the applicability of C-PID.

Acknowledgement

This project has been financially supported in part by the Chemical Engineering Institute Cooperative Research Center, III-2 project.

REFERENCES

1. LUYBEN, W. L.: Effect of Derivative Algorithm and Tuning Selection on the PID Control of Dead-Time Processes, Ind. Eng. Chem. Res., *2001*, 40, 3605-3611.
2. BÓDIZS Á.: Study of Model Predictive Control (in Hungarian), Ph.D. Theses, Veszprém, 1998.
3. SEEBORG, D. E., EDGAR, T. F., MELLICHAMP, D. A.: Process Dynamics and Control, Wiley, New York, 1989.
4. LIPTÁK, B. G. (ed.): Instrument Engineers' Handbook. Process Control, 3rd Ed., Chilton Book C., Radnor, PE, 1995.
5. NAGY, L.: Simulation and Control of Batch Reactors, Ph.D. Theses, Veszprém, 2005.
6. MADAR, J., ABONYI, J., SZEIFERT, F: Feedback Linearizing Control Using Hybrid Neural Networks Identified by Sensitivity Approach, Eng. Appl. of Artificial Intelligence, *2005*, 18, 343-351.
7. SZEIFERT, F., NAGY, L., CHOVÁN, T., ABONYI, J.: Constrained Inverse Model-based Control (*in preparation*).
8. BÓDIZS Á., SZEIFERT F., CHOVÁN T.: Convolution Model Based Predictive Controller for Nonlinear Process, Ind. Eng. Chem. Res., *1999*, 38, 154-161.