Proceeding of the 2004 American Control Conference
Boston, Massachusetts June 30 - July 2, 2004

A Method for PID Controller Tuning Using Nonlinear Control Techniques*

Prashant Mhaskar, Nael H. El-Farra and Panagiotis D. Christofides†
Department of Chemical Engineering
University of California, Los Angeles, CA90095

Abstract— **In this work, we propose a two-level, optimization-based method for deriving tuning guidelines for proportional-integral-derivative (PID) controllers that take explicitly into account the presence of nonlinear behavior. The central idea behind the proposed method is the selection of the PID controller tuning parameters so as to best "emulate" the control action and closed–loop response under a given nonlinear controller for a broad set of initial conditions and set-point changes. The first level involves using classical tuning guidelines (typically derived on the basis of linear approximations, running open or closed–loop tests) to obtain reasonable bounds on the tuning parameters in order to satisfy various design criteria such as stability, performance and robustness. These bounds are in turn incorporated as constraints on the optimization problem solved at the higher level to yield tuning parameter values that improve upon the values obtained from the first level to better emulate the closed–loop behavior under the nonlinear controller. The efficacy of the proposed tuning method is demonstrated through application to a nonlinear chemical reactor example.**

Key words: PID controller, Tuning methods, Nonlinear Control, Process control.

I. INTRODUCTION

The majority (over 90%) of the regulatory loops in the process industries use conventional proportional-integral-derivative (PID) controllers. Owing to the abundance of PID controllers in practice and the varied nature of processes that the PID controllers regulate, extensive research studies have been dedicated to the analysis of closed–loop properties of PID controllers and to devising new and improved tuning guidelines for the PID controllers, focusing on closed–loop stability, performance and robustness (see, for example, [1], [2], [3], [4], [5], [6], [7], the survey papers [8], [9] and books [10], [11], [12]). Most of the tuning rules are based on obtaining linear models of the system, either through running step tests or by linearizing a nonlinear model around the operating steady-state, and then computing values of the controller parameters that incorporate stability, performance and robustness objectives in the closed–loop system.

While the use of linear models for the PID controller tuning makes the tuning process easy, the underlying dynamics of many processes are often highly complex, due, for example, to the inherent nonlinearity of the underlying chemical reaction, or due to operating issues such as actuator constraints, time–delays and disturbances. Ignoring the inherent nonlinearity of the process when setting the values of the controller parameters may result in the controller's inability to stabilize the closed–loop system and may call for extensive re-tuning of the controller parameters.

The shortcomings of classical controllers in dealing with complex process dynamics, together with the abundance of such complexities in modern–day processes, have motivated a significant and growing body of research work within the area of nonlinear process control over the past two decades, leading to the development of several practically–implementable nonlinear control strategies that can deal effectively with a wide range of process control problems such as nonlinearities, constraints, uncertainties, and time–delays (see, for example, [13], [14], [15], [16], [17] and the books [18], [19], [20], [21]). While process control practice has the potential to gain from these advances through the direct implementation of the developed nonlinear controllers, an equally important direction in which process control practice stands to gain from these developments lies in investigating how nonlinear control techniques can be utilized for the improved tuning of classical PID controllers. This is an appealing goal because it allows control engineers to potentially take advantage of the superior stability and performance properties provided by nonlinear control without actually forsaking the ubiquitous conventional PID controllers or re-designing the control system hardware.

There has been some research effort towards incorporating nonlinear control tools in the design of PID controllers including, for example, [22], where it is shown that controllers resulting from nonlinear model-based control theory can be put in a form that looks like the PI or PID controllers for first and second-order systems. Other examples include [23], where adaptive PID controllers are designed using a backstepping procedure and [24], where a self-tuning PID controller is derived using Lyapunov techniques. In these works, even though the resulting controller has the same structure as that of a PID controller, the controller parameters (gain: K_c, integral time–constant: τ_I, and the derivative time–constant: τ_D) are not constant but functions of the error or process states. While such analysis provides useful analogies between nonlinear control tools and PID controllers, implementation of these control designs would require changing the control hardware in a way that values of the tuning parameters can continuously change while the process is in operation.

Motivated by the above considerations, we propose in this

* Financial support by NSF, CTS-0129571, is gratefully acknowledged.
†Corresponding author. (e-mail: pdc@seas.ucla.edu)

work a two-level, optimization-based method for deriving tuning guidelines for PID controllers that take explicitly into account the presence of nonlinear behavior. The central idea behind the proposed approach is the selection of the PID controller tuning parameters so as to best "emulate" the control action and closed–loop response prescribed by a given nonlinear controller for a broad set of initial conditions and set-point changes. The first level involves using classical tuning guidelines (typically derived on the basis of linear approximations, running open or closed–loop tests) to obtain reasonable bounds on the tuning parameters to satisfy various design criteria such as stability, performance and robustness. These bounds are in turn incorporated as constraints on the optimization problem solved at the higher level to yield tuning parameter values that improve upon the values obtained from the first level to better emulate the closed–loop behavior under the nonlinear controller. The efficacy of the proposed tuning method is demonstrated through application to a nonlinear chemical reactor example.

II. Two-level PID Tuning Method

In this work, we consider continuous–time single–input single–output (SISO) nonlinear systems, with the following state-space description

$$\begin{array}{rcl} \dot{x}(t) & = & f(x(t)) + g(x(t))u(t) \\ y & = & h(x) \end{array} \tag{1}$$

where $x = [x_1 \cdots x_n]' \in \mathbb{R}^n$ denotes the vector of state variables and x' denotes the transpose of x, $y \in \mathbb{R}$ is the process output, $u \in \mathbb{R}$ is the manipulated input, $f(\cdot)$ is a sufficiently smooth nonlinear vector function with $f(0) = 0$, $g(\cdot)$ is a sufficiently smooth nonlinear vector function and $h(\cdot)$ is a sufficiently smooth nonlinear function with $h(0) = 0$. Throughout the paper, the notation $L_f h$ denotes the standard Lie derivative of a scalar function $h(\cdot)$ with respect to the vector function $f(\cdot)$, i.e., $L_f(x) = (\partial h/\partial x)f(x)$.

The basic idea behind the proposed approach is the design (but not implementation) of a nonlinear controller that achieves the desired process response, and then, the tuning of the PID controller parameters so as to best "emulate" the control action and closed–loop process response under the nonlinear controller, subject to constraints derived from classical PID tuning rules. First, we use classical PID tuning guidelines to come up with reasonable ranges for the PID tuning parameters. Then we compute off-line, through simulation, the control action of the nonlinear controller over the time that it takes to practically achieve the set-point change, and the corresponding process response. The PID controller tuning parameters are then computed by solving (off-line) an optimization problem that minimizes some measure of the difference between the control action and closed–loop process response under the nonlinear controller on one hand, and those obtained under the PID controller on the other, for a given initial condition and set-point change. The optimization is carried out subject to constraints that ensure that the PID tuning parameters are within acceptable ranges of the values derived from the classical tuning rules. This idea is described algorithmically below:

1) Construct a nonlinear process model and derive a linear model around the operating steady-state (either through linearization or by running step tests).
2) On the basis of the linear model, use classical tuning guidelines to determine bounds on the values of K_c, τ_I and τ_D.
3) Using the nonlinear process model and desired process response, design a nonlinear controller.
4) For a set-point change, compute off-line, through simulations, the input trajectory ($u_{nl}(t)$) 'prescribed' by the nonlinear controller over the time (t_{final}) that it takes to achieve the set-point change and the corresponding $y_{nl}(t)$.
5) Compute PID tuning parameters ($K_c,\ \tau_I\ and\ \tau_D$) as the solution to the following optimization problem

$$J = \int_0^{t_{final}} (y_{nl}(t) - y_{PID}(t))^2 + (u_{nl}(t) - u_{PID}(t))^2 dt \tag{2}$$

$$\begin{array}{rcl} s.t.\ u_{PID}(t) & = & K_c\left(e + \dfrac{\int_0^{t'} edt'}{\tau_I} + \tau_D\dfrac{de}{dt}\right) \\ e(t) & = & y_{sp} - y_{PID} \\ \dot{x}(t) & = & f(x(t)) + g(x(t))u_{PID}(t) \\ y_{PID} & = & h(x) \\ \alpha_1 K_c^c & \leq K_c & \leq \alpha_4 K_c^c \\ \alpha_2 \tau_I^c & \leq \tau_I & \leq \alpha_5 \tau_I^c \\ \alpha_3 \tau_D^c & \leq \tau_D & \leq \alpha_6 \tau_D^c \\ (K_c,\ \tau_I,\ \tau_D) & = & argmin(J) \end{array} \tag{3}$$

where y_{sp} is the desired set-point, y_{nl}, u_{nl} are the closed–loop process response and control action under the nonlinear controller respectively, and $0 \leq \alpha_i < 1,\ i = 1, 2, 3$ and $1 < \alpha_i < \infty,\ i = 4, 5, 6$ are design parameters.

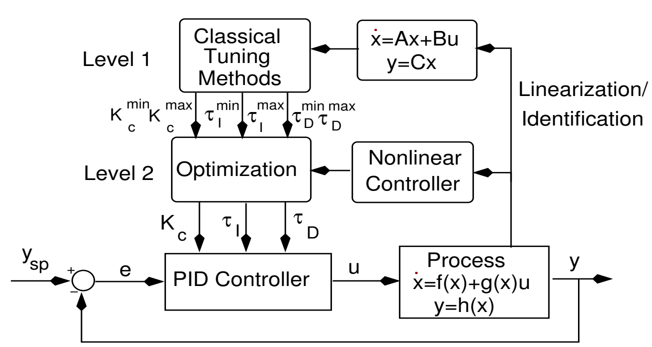

Fig. 1. Schematic representation of the implementation of the proposed two-level, optimization-based tuning method.

Remark 1: The optimization problem above computes values for K_c, τ_I, τ_D such that the closed–loop control action and process response under the PID controller is similar to that under the nonlinear controller. Note that the above optimization is carried out off-line and is part of the design procedure to compute values for the tuning parameters. Also, the above optimization problem can be carried out over a range of representative initial conditions and set-point changes to obtain PID tuning parameters that allow the PID controller to approximate, in an average (with respect to initial conditions and set-point changes) sense, the closed–loop response under the nonlinear controller.

Remark 2: The tuning methodology is not proposed as an *alternative* to existing tuning guidelines, but one that serves to *improve* upon existing tuning guidelines by imbuing them with desirable features provided by nonlinear control tools. If a given methodology is satisfactory, and it is desired that the tuning parameters not be changed appreciably, this can be enforced by using values of the design parameters, α_i, close to 1. The tuning parameters resulting from the solution to the optimization problem, while changed to mimic the nonlinear control action, will be close to the ones obtained from the classical tuning methods.

Remark 3: The proposed PID controller tuning method can be used to implement control for processes with uncertainties, time–delays, and manipulated input constraints. The main idea, once again, is the design of a nonlinear controller that accounts for process uncertainties, time–delay and manipulated input constraints. The PID tuning guidelines can serve to "carry over" the robustness and constraint handling properties of the nonlinear controllers in two ways: (1) through the objective function, by requiring the control action and closed–loop process response under PID control to mimic that under the nonlinear controller, and (2) manipulated input constraints can be incorporated directly as constraints in the optimization problem.

Remark 4: Note that the proposed tuning method is different from other methods that propose a nonlinear PID controller (see, for example, [22], [23]), which require changing the hardware of the PID controller because, in these designs, K_c, τ_I and τ_D are not constant but are functions of the error or process states, and therefore need to be continuously changed while the process is in operation. The tuning framework proposed in this work, while using nonlinear control techniques, provides tuning guidelines for *existing* PID controllers, by providing values for the controller parameters that stay fixed during implementation, and does not require redesigning or replacing the PID controller hardware.

Remark 5: Note that the derivative part of the PID controller is often implemented using a filter. This feature can be easily incorporated in the optimization problem by explicitly accounting for the filter dynamics. Constraints on the filter time-constant, τ_f, obtained empirically through knowledge of the nature of noise in the process, can be imposed to ensure that the filtering action restricts the process noise from being transferred over to the control action.

Remark 6: To allow for simple computations, approximations can be introduced in solving the optimization problem of Eqs.2-3. For instance, in the computation of the control action, the error, $e(t)$, may be approximated by simply taking the difference between the set-point, y_{sp}, and the process output under the nonlinear controller, $y_{nl}(t)$, leading to a simpler optimization problem, that can be solved easily using numerical solvers such as Microsoft Excel (for a given choice of the decision variables, the objective function can be computed algebraically and does not involve integrating the process dynamics). The justification behind this being that if the resulting value of $u_{PID}(t)$ is "close enough" to $u_{nl}(t)$, then this approximation holds (see Section III for a demonstration). If the solution of the optimization problem does not yield a sufficiently small value for the objective function (indicating that u_{PID}, and therefore, y_{PID} is significantly different from u_{nl}, and y_{nl}), this approximation may not be valid anymore. In this case, one could revert to using $e(t) = y_{sp} - y_{PID}(t)$ in the optimization problem, where y_{PID} is the closed–loop process response under the PID controller.

Remark 7: Finally, we note that the proposed method *does not* turn the PID controller into a nonlinear controller. The tuning method can only serve to improve upon the process response of the PID controller for operating conditions for which PID control action can be used to stabilize the process. If the process is highly nonlinear, or a complex process response is desired, it may be possible that the PID controller structure is not adequate, and in this case, the appropriate nonlinear controller should be implemented in the closed–loop to achieve the desired closed–loop properties.

III. Application to a Chemical Reactor Example

We consider a continuous stirred tank reactor where an irreversible, first-order reaction of the form $A \xrightarrow{k} B$ takes place. The inlet stream consists of pure species A at flow rate F, concentration C_{A0} and temperature T_{A0}. Under standard modeling assumptions, the mathematical model for the process takes the form

$$\begin{aligned}\dot{C}_A &= \frac{F}{V}(C_{A0} - C_A) - k_0 e^{\frac{-E}{RT_R}} C_A \\ \dot{T}_R &= \frac{F}{V}(T_{A0} - T_R) + \frac{(-\Delta H)}{\rho c_p} k_0 e^{\frac{-E}{RT_R}} C_A \\ &+ \frac{UA}{\rho c_p V}(T_j - T)\end{aligned} \tag{4}$$

where C_A denotes the concentration of the species A, T_R denotes the temperature of the reactor, T_j is the temperature of the fluid in the surrounding jacket, U is the heat–transfer coefficient, A is the jacket area, V is the volume of the reactor, k_0, E, ΔH are the pre-exponential constant, the activation energy, and the enthalpy of the reaction respectively, and c_p and ρ, are the heat capacity and fluid density in the reactor respectively. The values of all process parameters are given in Table I. At the nominal operating condition of $T_j^{nom} = 493.87\ K$, the reactor is operating at the unique, stable steady-state $(C_A^s,\ T_R^s) = (0.52, 398.97)$. The control objective is to implement set-point changes in the reactor temperature using the jacket fluid temperature, T_j, as the manipulated input, using a PID controller.

TABLE I
PROCESS PARAMETERS AND STEADY–STATE VALUES.

V	=	100.0	L
E/R	=	8000	K
C_{A0}	=	1.0	mol/L
T_{A0}	=	400.0	K
ΔH	=	2.0×10^5	J/mol
k_0	=	4.71×10^8	min^{-1}
c_p	=	1.0	$J/g.K$
ρ	=	1000.0	g/L
UA	=	1.0×10^5	$J/min.K$
F	=	100.0	L/min
C_A^s	=	0.52	mol/L
T_R^s	=	398.97	K
T_j^{nom}	=	493.87	K

To proceed with our PID controller tuning method, we initially design an input/output linearizing nonlinear controller. Note that the linearizing controller design is used in the simulation example only for the purpose of illustration, and any other nonlinear controller design deemed fit for the problem at hand can be used as part of the proposed controller tuning method.

Defining $x = [C_A - C_A^s,\ T_R - T_R^s]'$ and $u = T_j - T_j^{nom}$, the process of Eq.4 can be recast in the form of Eq.1 where the explicit form of $f(\cdot)$ and $g(\cdot)$ are omitted for brevity. Consider the control law given by:

$$u = \frac{\nu - y(t) - \gamma L_f h(x)}{\gamma L_g h(x)} \quad (5)$$

where $L_f h(x)$ and $L_g h(x)$ are the Lie derivatives of the function $h(x)$ with respect to the vector functions $f(x)$ and $g(x)$, respectively, γ, a positive real number, is a design parameter and ν is the set-point. Taking the derivative of the output in Eq.1 with respect to time, we get

$$\dot{y} = L_f h(x) + L_g h(x) u \quad (6)$$

Substituting the linearizing control law of Eq.5, we get

$$\dot{y} = \frac{\nu - y}{\gamma} \quad (7)$$

Under the control law of Eq.5, the controlled output y evolves linearly, to achieve the prescribed value of ν, with the design parameter γ being the time–constant of the closed–loop response.

It is well known that when a first–order closed–loop response, with a given time–constant, is requested for a linear first-order process, the method of direct synthesis yields a PI controller. Note that the relative order of the controller output, T_R, with respect to the manipulated input, T_j, in the example of Eq.4 is also one. This motivates using a PI controller and tuning the controller parameters to achieve the prescribed first-order response.

For the purpose of tuning the PI controller, the nonlinear process response generated under the nonlinear controller, using a value of $\gamma = 0.25$, was used in the optimization problem. Based on the parameter ranges for K_c and τ_I suggested by the IMC-based and Ziegler-Nichols tuning methods, the following constraints were derived and incorporated in the optimization problem: $0.1 \leq K_c \leq 15$ and $0.05 \leq \tau_I \leq 3$. The values of the parameters, computed using the IMC method, Ziegler-Nichols and the two-level PI tuning method are reported in Table II.

TABLE II
TUNING PARAMETERS

Tuning method	K_c	τ_I
IMC	1.81	0.403
Ziegler-Nichols	15.0	0.1667
Proposed method	5.38	2.1

The solid lines in Figs.2-3 show the closed–loop response of the output and the manipulated input under the nonlinear control of Eq.5. Note that the value of γ was chosen as 0.25 to yield a smooth, fast transition to the desired set-point. The optimization problem was solved approximately, using the closed–loop process response under the nonlinear controller to compute $e(t)$, and the objective function only included penalties on the difference between the control actions under the PI controller and the nonlinear controller (see Remark 6). The dashed–line shows the response of the PI controller tuned using the proposed optimization-based method. The result shows that the response under the PI controller is close to that under the nonlinear controller and demonstrates the feasibility of using a PI controller to generate a closed–loop response that mimics the response of the nonlinear controller.

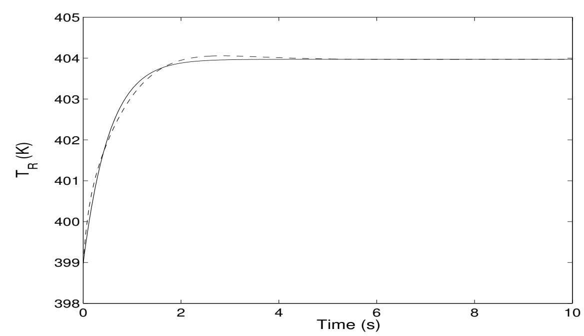

Fig. 2. Closed–loop output profile under a linearizing controller (solid line) and a PI controller tuned using the proposed method (dashed line).

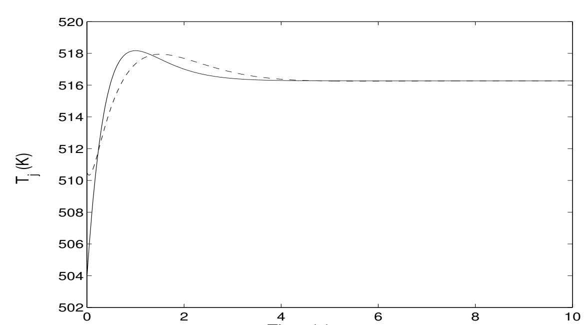

Fig. 3. Closed–loop manipulated input profile under a linearizing controller (solid line) and a PI controller tuned using the proposed method (dashed line).

In Fig.4, we present the closed–loop responses when the controller parameters computed using the IMC-based tuning rules and Ziegler-Nichols tuning rules are implemented. As can be seen, the transition to the new set-point under the PID controller tuned using the proposed method (dashed lines in Fig.4) is smoother when compared to a classical PI controller tuned using IMC tuning rules (solid line) and Ziegler-Nichols tuning rules (dotted line) that exhibit noticeable overshoot before settling at the desired set-point. The corresponding manipulated input profiles are shown in Fig.5.

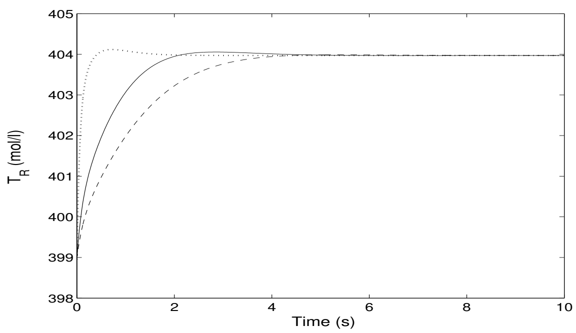

Fig. 4. Closed–loop output profile using IMC tuning rules for PI controller (solid line), using Ziegler-Nichols tuning rules (dotted line) and the proposed method (dashed line).

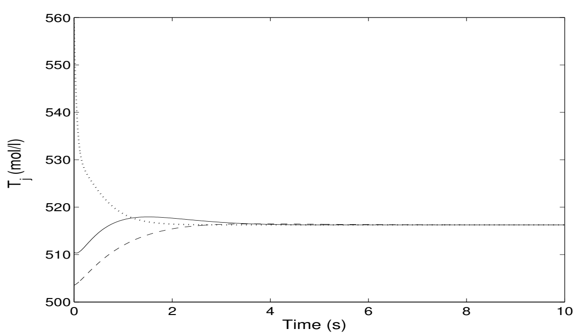

Fig. 5. Closed–loop manipulated input profile using IMC tuning rules (solid line), using Ziegler-Nichols tuning rules (dotted line) and the proposed method (dashed line).

We now demonstrate the application of the proposed method to the same system, but with C_A as the controlled variable and T_j as the manipulated variable. As in the previous case, we initially design an input/output linearizing nonlinear controller to yield a second-order linear input-output response in the closed–loop system of the form:

$$\tau_{cl}^2 \ddot{y} + \frac{2\xi}{\tau_{cl}} \dot{y} + y = \nu \tag{8}$$

where τ_{cl} and ξ are design parameters and were chosen as $\tau_{cl} = 0.2$ and $\xi = 1.05$ (implying that the closed–loop system is a slightly over-damped second-order system).

The following tuning methods were used for the first level: (1) IMC-based tuning rule, where a step test is run to approximate the system by a first-order + time–delay process, (2) IMC-based tuning rule, where the process is linearized around the operating steady-state to obtain a second-order linear model, and (3) Ziegler-Nichols tuning rules, where the parameters $K_{cu} = -20000$ and $P_u = 0.4$ are obtained using the method of relay auto tuning [25] (the tuning parameter values are reported in Table III). Based on the parameter ranges suggested by the first level tuning methods, the following constraints were used in the optimization problem set up to compute K_c, τ_I and τ_D: $-2000.0 \leq K_c \leq -1000$, $0.1 \leq \tau_I \leq 2$ and $0.02 \leq \tau_D \leq 0.15$. The derivative part of the controller was implemented using a first order filter with time–constant $\tau_f = 0.1$. The

TABLE III

TUNING PARAMETERS

Tuning method	K_c	τ_I	τ_D
IMC-I	-1342.7	0.95	0.0524
IMC-II	-1813.4	1.00	0.114
Ziegler-Nichols	-11753.0	0.2	0.05
Proposed method	-1229.0	1.38	0.13

solid lines in Figs.6-7 show the closed–loop response of the output and the manipulated input under the linearizing control design. The dashed–line shows the response of the PID controller tuned using the proposed method, which is close to the response of the nonlinear controller. As is clear from Fig.6, the resulting PID controller yields a response that is close enough to that of the nonlinear controller.

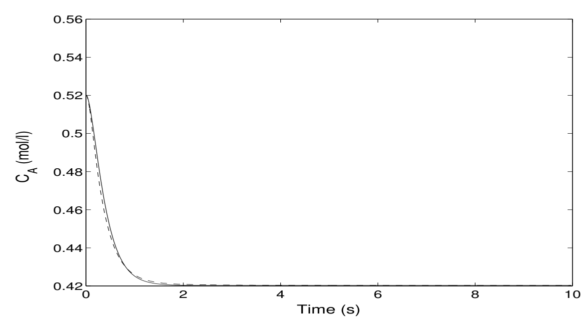

Fig. 6. Closed–loop output profile under a linearizing controller (solid line) and a PID controller tuned using the proposed method (dashed line).

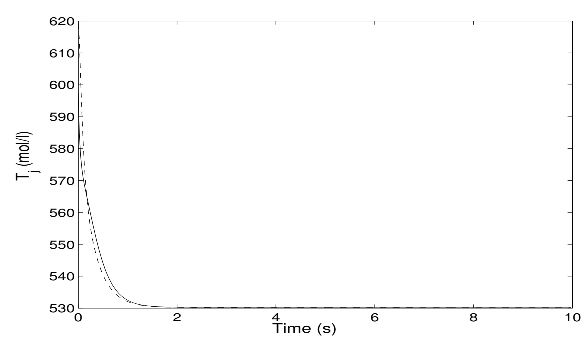

Fig. 7. Closed–loop manipulated input profile under a linearizing controller (solid line) and a PID controller tuned using the proposed method (dashed line).

In Fig.8, we present the closed–loop responses when the controller parameters computed using the classical tuning rules are implemented. The values suggested by

Ziegler-Nichols tuning lead to closed–loop instability. In the simulation, a smaller value for K_c and a larger τ_I were used. As can be seen, the transition to the new set-point using the proposed tuning method (solid lines in Fig.8) compares favorably to that obtained when using the IMC-based tuning rules-I and II (dash-dotted and dotted line) and the Ziegler-Nichols tuning rules (dashed line). The corresponding manipulated input profiles are shown in Fig.9.

In summary, we proposed a two-level tuning method where the first level involved using classical tuning guidelines to obtain reasonable bounds on the tuning parameters. These bounds were in turn incorporated as constraints on the optimization problem solved at the higher level to yield tuning parameter values that improve upon the values obtained from the first level to better emulate the closed–loop behavior under the nonlinear controller. For the cases studied, it was found that the proposed tuning method achieved a response that was closer to the one obtained under the nonlinear controller.

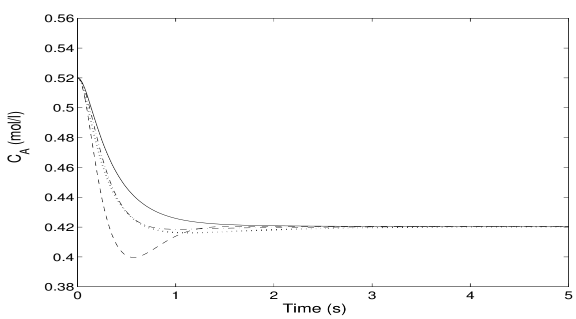

Fig. 8. Closed–loop output profile using IMC tuning rules I (dotted line), IMC tuning rules II (dash-dotted line), Ziegler-Nichols tuning rules (dashed line) and the proposed method (solid line).

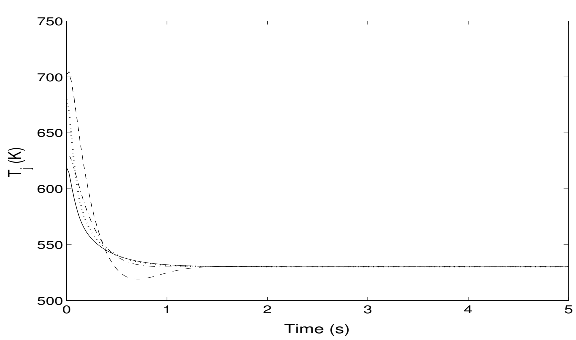

Fig. 9. Closed–loop manipulated input profile using IMC tuning rules I (dotted line), IMC tuning rules II (dash-dotted line), Ziegler-Nichols tuning rules (dashed line) and the proposed method (solid line).

REFERENCES

[1] J. G. Ziegler and N. B. Nichols, "Optimum settings for automatic controllers," *Trans. ASME*, vol. 64, pp. 759–768, 1942.

[2] D. E. Rivera, M. Morari, and S. Skogestad, "Internal model control. 4. PID controller design," *Ind. Eng. Chem. Proc. Des. & Dev.*, vol. 25, pp. 252–265, 1986.

[3] J. T. Lee, W. Cho, and T. F. Edgar, "Control system design based on a first order plus time delay model," *J. Proc. Contr.*, vol. 7, pp. 65–73, 1997.

[4] W. L. Luyben, "Effect of derivative algorithm and tuning selection on the PID control of dead-time processes," *Ind. & Eng. Chem. Res.*, vol. 40, pp. 3605–3611, 2001.

[5] H. Panagopoulos, K. J. Astrom, and T. Hagglund, "Design of PID controllers based on constrained optimization," *IEEE Proc. Contr. Theory Appl.*, vol. 149, pp. 32–40, 2002.

[6] S. Skogestad, "Simple analytic rules for model reduction and PID controller tuning," *J. Proc. Contr.*, vol. 13, pp. 291–309, 2003.

[7] K. J. Astrom and T. Hagglund, "Revisiting the Ziegler-Nichols step response method for PID control," *J. Proc. Contr., in press*, vol. 11, 2004.

[8] K. J. Astrom, T. Hagglund, C. C. Hang, and W. K. Ho, "Automatic tuning and adaption for PID controllers- A survey," *Contr. Eng. Pract.*, vol. 1, pp. 699–714, 1993.

[9] P. Cominos and N. Munro, "PID controllers: recent tuning methods and design to specifications," *IEEE Proc. Contr. Theory Appl.*, vol. 149, pp. 17–25, 2002.

[10] M. Morari and E. Zafiriou, *Robust Process Control*. New Jersey: Prentice Hall, 1989.

[11] S. Skogestad and I. Postelethwaite, *Multivariable Feedback Control*. New Jersey: Wiley, 1996.

[12] T. E. Marlin, *Process Control*. New Jersey: McGraw Hill, 2000.

[13] M. Arcak, A. R. Teel, and P. Kokotovic, "Robust nonlinear control of feedforward systems with unmodeled dynamics," *Automatica*, vol. 37, pp. 265–272, 2001.

[14] N. H. El-Farra and P. D. Christofides, "Coordinating feedback and switching for control of hybrid nonlinear processes," *AIChE Journal*, vol. 49, pp. 2079–2098, 2003.

[15] ——, "Bounded robust control of constrained multivariabe nonlinear processes," *Chem. Eng. Sci.*, vol. 58, pp. 3025–3047, 2003.

[16] Y. Lin and E. D. Sontag, "A universal formula for stabilization with bounded controls," *Syst. & Contr. Lett.*, vol. 16, pp. 393–397, 1991.

[17] C. Antoniades and P. D. Christofides, "Feedback control of nonlinear differential difference equation systems," *Chem. Eng. Sci.*, vol. 54, pp. 5677–5709, 1999.

[18] R. Sepulchre, M. Jankovic, and P. Kokotovic, *Constructive Nonlinear Control*. Berlin-Heidelberg: Springer-Verlag, 1997.

[19] R. A. Freeman and P. V. Kokotovic, *Robust Nonlinear Control Design: State-Space and Lyapunov Techniques*. Boston: Birkhauser, 1996.

[20] H. K. Khalil, *Nonlinear Systems*, 2nd ed. New York, NY: Macmillan Publishing Company, 1996.

[21] A. Isidori, *Nonlinear Control Systems: An Introduction*. Berlin-Heidelberg: Springer-Verlag., 2nd edition, 1989.

[22] R. A. Wright, C. Kravaris, and N. Kazantzis, "Model based synthesis of nonlinear PI and PID controllers," *AIChE J.*, vol. 47, pp. 1805–1818, 2001.

[23] A. R. Benaskeur and A. Desbiens, "Backstepping-based adaptive PID control," *IEEE Proc. Contr. Theory Appl.*, vol. 149, pp. 54–59, 2002.

[24] R. H. Chang, W and J. Hsieh, "A self tuning PID control for a class of nonlinear systems based on the Lyapunov approach," *J. Proc. Contr.*, vol. 12, pp. 233–242, 2002.

[25] K. J. Astrom and T. Hagglund, "Automatic tuning of simple regulators with specification on the gain and phase margins," *Automatica*, vol. 20, pp. 645–651, 1984.