

On-line adaptive tuning for PID controllers

H.-P.Huang, M.-L.Roan and J.-C.Jeng

Abstract: Adaptive and robust designs are two major approaches to deal with possible modelling errors in a control system. This paper proposes an adaptive control system for on-line tuning of PID controllers for SISO systems. The proposed system comprises of two adaptive loops. The first loop monitors and tunes the controllers on-line to ensure the system is robustly stable. This first loop enables the system to achieve robust stability over a wide range of possible modelling errors without trading off performance against nominal design. When modelling errors occur, the second adaptive loop conducts recursive on-line identification and retunes the controller. Because the nominal model is corrected on-line, directly following the on-line identification the system can continue as a new nominal system which is designed to have tight control performance. Numerical examples are presented to illustrate this adaptive control with two loops.

List of symbols

$C(s)$: transfer function of the IMC controller
 $G_p(s)$: transfer function of the plant
 $G_C(s)$: transfer function of the conventional controller
 k_p : gain
 $\ell(s)$: transfer function of the multiplicative modelling error
P: parameter vector
 s : Laplace transform variable
 t : time
 y : output
 \hat{y} : model output by the same input

Greek symbols

α : constant in the realisable PID controller
 δ : Sampling interval
 Δ : difference
 λ : eigen value
 ω : frequency
 ϕ : parameter sensitivity vector
 Φ : parameter sensitivity matrix
 τ : time constant
 σ : singular value

Superscripts

$\bar{\cdot}$: nominal
 $\hat{\cdot}$: estimated value
 T : transpose

Subscripts

0: initial
 D : derivative
 R : reset
 f : filter
 i : integral
 p : plant

1 Introduction

Dynamics of industrial plants are usually not completely known and are subject to change from time to time. In dealing with the control of such plants, robust control design and adaptive control design are two major approaches being used [1]. Each approach has its merits and inevitable disadvantages. For example, in robust control design, the system can remain stable subject to a set of modelling errors. However, owing to these modelling errors, the nominal performance of the system is traded off to secure robust stability. As the set of possible modelling errors expands, the performance of the nominal system becomes more conservative. Overviews of robust designs of such controllers can be found in [2] and [3]. On the other hand, theories and applications of adaptive control, especially self-tuning control, have been extensively reported in survey papers and books [4–6]. In adaptive design, good system performance is maintained by conducting self-tuning when the system deviates from nominal. Nevertheless, in general, there is a fundamental conflict between implementing self-tuning and the regular objectives of control [7], because special inputs are required to achieve repeated signals for on-line identification as an important ingredient of the self-tuning system. The stability of these self-tuning systems is another important issue. Although this issue has been addressed by researchers [6] and despite a great deal of progress in dealing with bounded disturbances [8–11], the stability of these systems is still a challenge in practical systems [10, 12]. For the reasons mentioned, adaptive systems are usually so sophisticated that they have to be maintained by well-trained personnel.

To the merits of both approaches, a system with dual adaptive loops (DAL) to enhance a conventional single-loop PID control system is presented. The conventional PID controller is designed to be stable over a small range of uncertainties to ensure tight nominal performance. The robustness of performance and stability is ensured by the dual adaptive loops.

The first adaptive loop conducts on-line tuning of the PID controller by adjusting an IMC filter to ensure stability robustness before updating to a new nominal model. Since

© IEE, 2002

IEE Proceedings online no. 20020099

DOI: 10.1049/ip-cta:20020099

Paper first received 12th November 2001

The authors are with the Department of Chemical Engineering, National Taiwan University, Taipei, Taiwan 10617

adaptation is done without estimating changes to the nominal model, it is performed in a direct manner. The second adaptive loop identifies the changes to the nominal model and tunes the controller accordingly. Because on-line identification is used, correction of the nominal model is performed in an indirect manner, i.e. by updating the nominal model of the process first and then changing the controller. During on-line identification, dead time and other parameters in the process are estimated at the same time. As the estimation of dead time needs nonlinear least squares computations, a Newton-like algorithm is also presented. The dual-loop adaptive system is then applied to a single-loop system where the nominal model is first order with dead time.

2 PID controller design for nominal system

Since the 1980s, the IMC principle has been a useful tool for designing feedback controllers. The design methodology focuses mainly on the use of an IMC filter and the inverse of a nominal model for the open-loop process, i.e. $\bar{G}_p(s)$. In this way, the feedback controller of a closed-loop system can be synthesised as

$$G_C(s) = \frac{C(s)}{1 - \bar{G}_p(s)C(s)} \quad (1)$$

where $C(s)$ is obtained by

$$C(s) = [\bar{G}_p]^{-1}(s)F(s) \quad (2)$$

Here, \bar{G}_p represents a nominal model of G_p , $[\bar{G}_p]^{-1}$ represents the invertible part of \bar{G}_p , and $F(s)$ is an IMC filter.

As a linear system with general dynamics, G_p is considered to be represented by a model in terms of the following transfer function:

$$G_p(s) = \frac{k_p N(s)}{D(s)} e^{-\theta s} \quad (3)$$

where $N(s)$ and $D(s)$ are rational polynomials of s . For most processes with high-order dynamics, their dynamics are usually modelled as the first-order dynamics plus dead time, that is,

$$G_p(s) = \frac{k_p e^{-\theta s}}{\tau s + 1} \quad (4)$$

In other words, all the higher order lags in (3) are lumped together to form an effective dead time.

To synthesise a controller via IMC for G_p in (4), a nominal model and filter are chosen as follows:

$$\bar{G}_p(s) = \frac{\bar{k}_p}{\bar{\tau}s + 1} \frac{-0.5\bar{\theta}s + 1}{0.5\bar{\theta}s + 1} \quad (5)$$

$$F(s) = \frac{1}{\tau_f s + 1}$$

The corresponding IMC-PID controller is

$$G_c = k_c \frac{(1 + (1/\tau_R s) + \tau_D s)}{\alpha s + 1} \quad (6)$$

where

$$\begin{aligned} k_c &= \frac{\tau + 0.5\theta}{\tau_f + \theta} \\ \tau_R &= \tau + 0.5\theta \\ \tau_D &= \frac{\tau}{2\tau + \theta} \\ \alpha &= \frac{0.5\tau_f\theta}{\tau_f + \theta} \end{aligned} \quad (7)$$

Because \bar{G}_p in (5) is used, model mismatch is inevitable even though the model for G_p in (4) is perfect. This model mismatch is given as follows:

$$\begin{aligned} \ell(s) &= \frac{G_p}{\bar{G}_p} - 1 \\ &= \frac{k_p}{\bar{k}_p} \frac{N(s)(\bar{\tau}s + 1)}{D(s)} \frac{1 + 0.5\bar{\theta}s}{1 - 0.5\bar{\theta}s} e^{-\theta s} - 1 \end{aligned} \quad (8)$$

In the presence of this model mismatch, the time constant of the IMC filter, τ_f , has important effects on the performance and robustness of the closed-loop system. Rivera *et al.* [13] discussed this issue in detail with regard to this IMC-PID controller. They show that the controller gives an optimal value for the ISE performance measure at $\tau_f/\theta \cong 0.65$. The setting for $\bar{\tau}_f$ in a non-adaptive system has to ensure robust stability over all possible modelling errors and, thus, must not use small values of τ_f/θ ratio. As a result, the nominal performance has to be traded off against stability.

With this DAL scheme, the system allows τ_f to have lower values which, in turn, gives tighter nominal performance. If modelling errors exist or changes to the model occur, the first loop of this DAL adaptive system starts to detune the controller on-line to ensure robust stability. As detuning the controller degrades the performance of the system, the second adaptive loop begins estimating the changes to the system to retune the controller. Thus the system will resume as a new nominal system that is designed to give tighter performance.

3 Direct adaptation of τ_f for stability robustness

In this Section, direct adaptation to achieve robust stability by adjusting an IMC filter is discussed. In the IMC design, the time constant of the filter, τ_f , has to be chosen properly to provide stability robustness subject to modelling mismatch. However, any choice of constant value for τ_f has to deal with a trade-off between performance and stability robustness. Thus, it is desirable to start with a smaller value of τ_f which gives satisfactory nominal responses and then to change the value of τ_f on-line to ensure robustness when model mismatches jeopardise the stability of the system.

To develop the adaptive law for this purpose, first, a nominal value for τ_f should be assigned. Besides the minimum ISE measure, the overshoot produced by the system is also an important factor to be considered. Fig. 1 shows the resonance peak gain and overshoot of this IMC-PID control system. According to Fig. 1, a value of 0.5 for $\bar{\tau}_f/\bar{\theta}$ will produce no more than 5% overshoot in the nominal case.

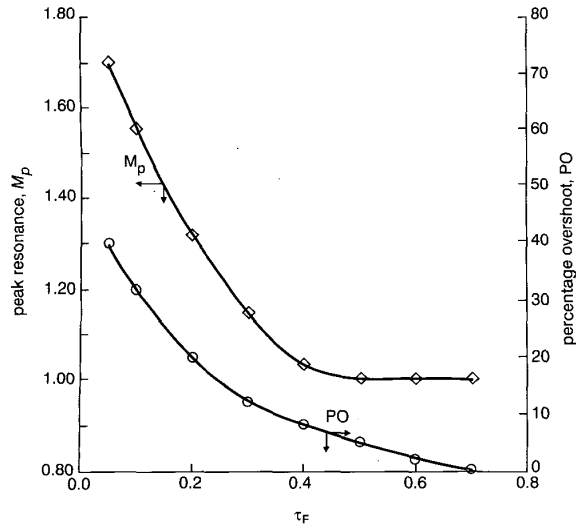


Fig. 1 Peak resonance (M_p) and percentage overshoot (PO) for an IMC-PID control system

$\tau_f = \tau_f/\theta$ where θ is dead time in the first-order model plus dead time

With this setting for τ_f , the nominal system will have a gain margin equal to 3. Note that the gain margin of the system is given by

$$GM = 2 \left(\frac{\bar{\tau}_f}{\bar{\theta}} + 1 \right) \quad (9)$$

Next, we look at how τ_f should be changed to keep the system robust. The following two properties will be useful for deriving the adaptive law for τ_f and their proofs are given in the Appendix (Section 9).

Property 1: For $G_p(s)$ in (4) and an IMC controller design based on (5), the initial slope of the system output, after the effective dead time, is given by

$$\lim_{t \rightarrow 0} \dot{y}(t + \theta) = \frac{1}{\tau_f} \frac{k_p \bar{\tau}}{k_p \bar{\tau}} \quad (10)$$

Notice that the result is derived based on a model of (4). For a higher order system, the resulting effective dead time in terms of θ in this model will be larger than the true one. As a result, the output, y , at $t = \theta_+$ will have non-zero slope. Thus, with this slope, (4) can apply to this case only approximately.

Property 2: For $G_p(s)$ in (3), which is open-loop stable and is strictly proper, there exists τ_f^* such that $\forall \tau_f \geq \tau_f^*$, and the IMC controller can stabilise the closed-loop system.

According to property 1, a nominal system will yield an initial slope equal to $1/\tau_f$ in response to a step change. Any changes in k_p or τ will disturb the initial slope of y from its nominal value. Notice that the complementary sensitivity function of this IMC-PID control system is given by

$$\mathbf{T} = \frac{e^{-\theta s}}{(\tau_f + 0.5\theta)s + e^{-\theta s}} \quad (11)$$

When θ is small, the bandwidth of the system is approximately given by [13]

$$\omega_b \cong \frac{1}{\theta + \tau_f} \quad (12)$$

Since τ_f is assigned to be 0.5 times θ , we have

$$\omega_b \cong \frac{1}{1.5\tau_f} \propto \dot{y}(t = \theta_+) \quad (13)$$

To keep the initial slope of the response unchanged regardless of the existence of model errors, the bandwidth of the system must remain constant. Accordingly, the performance of the closed-loop system will remain robust to the model mismatch.

It is, then, clear that τ_f must be adjusted to compensate for the change in the initial slope in order to achieve robust performance. To develop an algorithm for this purpose, let us define the following:

τ_f^0 = the initial value of $\tau_f = 0.5\theta^0$,
 τ_f^1 = a new target value to which τ_f is to be adjusted,
 τ_f^* = the nominal designed value of $\tau_f = 0.5\bar{\theta}$.

Here, θ^0 is dead time in the nominal model, $\bar{\theta}$ is effective dead time found in the actual response. Then, the algorithm for adapting τ_f is obtained as follows:

$$\tau_f^1 = \tau_f^* \tau_f^0 \dot{y}|_{t=\bar{\theta}} \quad (14)$$

provided that \dot{y} can be measured or estimated.

In practice, the estimation of \dot{y} by taking numerical derivative will be sensitive to measuring noise. To estimate \dot{y} at $t = \bar{\theta}$, starting from the time instant at the effective dead time, three or more values of y separated by a small sampling interval (i.e. δ) are taken. These data are then fitted to a straight line. The slope of this line is then taken as the estimate of \dot{y} at $t = \bar{\theta}$. This part of the adaptation is made at the very first stage right after the system's effective dead time.

On the other hand, achieving stability of the system requires examining whether there is diverging or persistently oscillating errors in tracking the set-point. According to property 2, whenever modelling errors jeopardise the stability of the system, by increasing τ_f to a larger value, the system can eventually be stabilised. In contrast to modelling errors which jeopardise the stability of a closed-loop system, in some cases, modelling errors may make the system move towards a more favourable response than its nominal one. If so, changing τ_f will not be necessary in such cases. Thus, adaptation of τ_f is necessary only if the system fails to track the set-point well and only if significant modelling errors are detected. This part of the adaptation is made until the response approaches the set-point.

An adapting algorithm is thus given as follows:

$$\tau_f = \tau_f^1 \left(1 + \int_{t_r}^t |e(t)| \left| \frac{y(t) - \hat{y}(t|t-\delta)}{\hat{y}(t)} \right| dt \right) \quad (15)$$

Here, t_r is the moment when the response first approaches its target value, $e(t) = r - y(t)$, and $\hat{y}(t|t-\delta)$ is the model output using all the data up to $t - \delta$. In other words, $y(t) - \hat{y}(t|t-\delta)$ is a one-step ahead predictive error which is used to detect the existence of modelling errors.

The adaptive law given in (15) has the following implications:

1. When the nominal model is perfect, the value of τ_f will remain unchanged. Thus, the desired nominal performance can be obtained.
2. When modelling errors exist but tracking errors are small, model mismatch will not seriously jeopardise the control performance. In such cases, the value of τ_f will only be detuned slightly.
3. The value of τ_f will stop increasing if the PID control

can enable the system output to reach and stay at the constant set-point.

To summarise this direct adaptive law used to achieve system robustness, τ_f is adjusted according to the following:

$$\tau_f = \tau_f^1 \left(1 + \int_{t_{0.9}}^t |e(t)| \left| \frac{y(t) - \hat{y}(t|t - \delta)}{\hat{y}(t)} \right| \right) dt \quad (16)$$

where

$$\begin{aligned} \tau_f^1 &= \tau_f^0 = 0.5\bar{\theta} \quad \text{for } t < t_{0.1} \\ \tau_f^1 &= 0.5\bar{\theta}\tau_f^* \hat{y}(t_{0.1}) \quad \text{for } t \geq t_{0.1} \end{aligned} \quad (17)$$

Here, $t_{0.1}$ and $t_{0.9}$ denote the times at which $y(t)$ reaches the values of 10% and 90% of the set-point, respectively. $\bar{\theta}$ is an estimated value of the time delay obtained through regression analysis around the data in the neighbourhood of $y(t_{0.1})$.

4 Indirect adaptation for nominal performance

According to the direct adaptive adjustment of τ_f depicted above, the value of τ_f will float. In cases where the change of the initial derivative of the responses cannot be clearly identified, the value of τ_f will continue increasing as long as modelling errors exist. To bring this τ_f back to its nominal value, the nominal model has to be updated on-line. Updating the nominal model and the controller using on-line algorithms is the main goal of indirect adaptive control. To estimate changes of the process model using on-line algorithms is known as on-line identification. Usually, least squares methods, especially linear least squares, are used for this purpose. In general, adaptive control with on-line identification experiences bottlenecks. First, on-line identification conflicts with the control objectives. As the control system brings the system back to its target as quickly as possible, the identification process perturbs the system by means of external signals (such as a sequence of set-point changes or some specially designed driving signals) in order to generate persistent excitations. Second, on-line identification itself encounters technical bottlenecks. One of the difficulties is in the estimation of dead time. Quite a bit of research on adaptive control with respect to unknown dead time has been reported in the literature [14–17]. Disadvantages of these methods include the large number of parameters needed and the convergence of the algorithms. As a matter of fact, estimation of dead time together with other system parameters is a nonlinear optimisation problem, so that linear least squares algorithms are not directly applicable. As a result, recursive algorithms for on-line identification and adaptive control are difficult to apply. In this dual-loop adaptive system, the indirect adaptive control includes on-line estimation of a nominal model of (4) and on-line computation of the controller parameters in (6).

For on-line identification, a model output is computed according to

$$\frac{\hat{y}(s; \mathbf{P})}{u(s)} = \frac{\hat{k}_p e^{-\hat{\theta}s}}{\hat{\tau}s + 1} \quad (18)$$

where $\mathbf{P} = \{\hat{k}_p, \hat{\tau}, \hat{\theta}\}$ is the parameter set of the model. Then, \mathbf{P} is determined by minimising as follows:

$$J(\tau_f; \hat{\mathbf{P}}) = \int_0^{\tau_f} [y(t) - \hat{y}(t; \hat{\mathbf{P}})]^2 dt \quad (19)$$

To derive an on-line estimation algorithm for $\hat{\mathbf{P}}$, it is assumed that the model output, \hat{y} is differentiable with respect to the parameter vector, \mathbf{P} . In other words, it is assumed that there always exists a small positive number, ϵ , such that $o(\epsilon^2)$ is negligible and the following holds:

$$\hat{y}\left(t; \mathbf{P} + \epsilon \frac{\Delta \mathbf{P}}{|\Delta \mathbf{P}|}\right) \approx \hat{y}(t; \mathbf{P}) + \epsilon \phi(t; \mathbf{P})^T \frac{\Delta \mathbf{P}}{|\Delta \mathbf{P}|} + o(\epsilon^2) \quad (20)$$

where $\phi(t; \mathbf{P})^T$ is the l th row vector of Φ with $l = t/\delta$. δ is a sampling interval.

Let

$$\mathbf{Y}(N) = \begin{bmatrix} y(1\delta) \\ y(2\delta) \\ \vdots \\ y(N\delta) \end{bmatrix}$$

$$\hat{\mathbf{Y}}(N, \mathbf{P}^{(i)}) = \begin{bmatrix} \hat{y}(1\delta; \mathbf{P}^{(i)}) \\ \hat{y}(2\delta; \mathbf{P}^{(i)}) \\ \vdots \\ \hat{y}(N\delta; \mathbf{P}^{(i)}) \end{bmatrix}$$

$$\Phi(N; \mathbf{P}^{(i)}) = \begin{bmatrix} \frac{\partial \hat{y}(1\delta; \mathbf{P}^{(i)})}{\partial p_1} & \frac{\partial \hat{y}(1\delta; \mathbf{P}^{(i)})}{\partial p_2} & \cdots & \frac{\partial \hat{y}(1\delta; \mathbf{P}^{(i)})}{\partial p_s} \\ \frac{\partial \hat{y}(2\delta; \mathbf{P}^{(i)})}{\partial p_1} & \frac{\partial \hat{y}(2\delta; \mathbf{P}^{(i)})}{\partial p_2} & \cdots & \frac{\partial \hat{y}(2\delta; \mathbf{P}^{(i)})}{\partial p_s} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial \hat{y}(N\delta; \mathbf{P}^{(i)})}{\partial p_1} & \frac{\partial \hat{y}(N\delta; \mathbf{P}^{(i)})}{\partial p_2} & \cdots & \frac{\partial \hat{y}(N\delta; \mathbf{P}^{(i)})}{\partial p_s} \end{bmatrix}$$

The formulation of $\hat{\mathbf{Y}}(N, \mathbf{P}^{(i)})$ for G_p of first-order dynamics plus dead time for identification has been given by Huang and Cheng [18]. Computation of $\hat{\Phi}_i$ for the model of (4) needs the sensitivity function of \hat{y} with respect to each of the parameters, i.e. $\partial \hat{y}(t, p_i) / \partial p_i$. In practice, these sensitivity functions can be obtained by finding the numerical derivatives of $\hat{y}(t, p_i)$ with respect to \mathbf{P}_i by the following:

$$\frac{\partial \hat{y}(t, p_i)}{\partial p_i} \approx \frac{\hat{y}(t, p_i + \delta) - \hat{y}(t, p_i)}{\delta} \quad (21)$$

where δ is a small perturbation to the parameter p_i , and $\hat{y}(t, p_i)$ can be obtained by integrating the differential equation as described by the model in (18) using $u(t)$ as the deriving input.

Let

$$\mathbf{V}_i(N; \mathbf{P}^{(i)}) = [\mathbf{Y}(N) - \hat{\mathbf{Y}}(N; \mathbf{P}^{(i)})]^T [\mathbf{Y}(N) - \hat{\mathbf{Y}}(N; \mathbf{P}^{(i)})] \quad (22)$$

where the superscript i designates that the referred value is taken at the i th iteration. For convenience, $\hat{\mathbf{Y}}(N; \hat{\mathbf{P}}^{(i)})$ will be abbreviated as $\hat{\mathbf{Y}}_i(N)$ and $\Phi(N, \mathbf{P}^{(i)})$ as $\Phi_i(N)$ in the following.

Then, we have

$$\begin{aligned} \Delta \mathbf{V}_i(N) &= \mathbf{V}_{i+1}(N) - \mathbf{V}_i(N) \\ &= [\mathbf{Y}(N) - \hat{\mathbf{Y}}(N; \mathbf{P}_{i+1})]^T [\mathbf{Y}(N) - \hat{\mathbf{Y}}(N; \mathbf{P}_{i+1})] \\ &\quad - [\mathbf{Y}(N) - \hat{\mathbf{Y}}(N; \mathbf{P}_i)]^T [\mathbf{Y}(N) - \hat{\mathbf{Y}}(N; \mathbf{P}_i)] \end{aligned} \quad (23)$$

If we let

$$\mathbf{P}^{(i+1)} = \mathbf{P}^{(i)} + \rho_i \Delta \mathbf{P}^{(i)} \quad (24)$$

$$\Delta \mathbf{P}^{(i)} = \{[\Phi_i(N)]^T [\Phi_i(N)]\}^{-1} \{[\Phi_i(N)]^T [\mathbf{Y}(N) - \hat{\mathbf{Y}}_i(N)]\} \quad (25)$$

and

$$\rho_i = \frac{\epsilon}{|\Delta \mathbf{P}^{(i)}|} \quad (26)$$

where Φ_i designates $\Phi(N, \mathbf{P}^{(i)})$. It is thus found that

$$\Delta \mathbf{V}_i(N) \leq 0, \quad \forall 0 \leq \rho_i \leq 2 \quad (27)$$

Because $\lambda(\Phi_i(\Phi_i^T \Phi_i)^{-1} \Phi_i^T) \geq 0$, it is evident that $\Delta \mathbf{V}_i(N) \leq 0$. Similar recursive identification problem formulated in the above can be found elsewhere [19]. Notice that if, for all i , $\Phi(N, \mathbf{P}^{(i)})$ is of full rank, then

$$\Delta \mathbf{V}_i(N) < 0 \quad \forall i \quad (28)$$

For on-line estimation, the computation can be conducted using a rectangular data window as proposed by Huang and Wang [20] or by using an extending data window which adopts new data at each sampling instant. If we use an extending data window, the estimation starts with a value N_0 for N , and an initial guess $\mathbf{P}^{(0)}$ for \mathbf{P} . Then, \mathbf{P} is updated iteratively using (24) through (25) until a minimal value of \mathbf{V} of that data window is reached or a specified number of iterations is reached. This completes a computation cycle. As new input-output data come in, the computation cycle starts all over again with the newly computed \mathbf{P} and the value of N increased by one. The computation cycle is repeated until $\Phi(N; \mathbf{P}^{(i)})$ fails to be of full rank. A criterion to justify if this above matrix is of full rank is to check for its minimum singular value. If this minimum singular value is less than a prescribed limit, $\underline{\sigma}_{stop}$, then the computation for identification will be stopped. The indirect adaptive control modifies the controller parameters when a converged model is available. To determine if the identification has converged, the norm of parameter changes to be updated is compared with a specified tolerance. If this norm lies within this tolerance, the identification result will be used to update the controller parameters.

5 A dual adaptive loop (DAL) scheme for adaptive control

Section 3 addressed adaptive control by adaptively changing τ_f . By such an adaptation, the system will have a floating τ_f and obviously the system cannot have as good a performance as that expected from the nominal system. As shown in Section 4, the indirect type of adaptive control alone cannot guarantee robust stability during the period of identification. However, since the changed parameters have been identified and updated, the performance of the system can be resumed later on. In order to enjoy the advantages of both types of control, a DAL system for adaptive control is thus considered. In this DAL system, the indirect and direct adaptive mechanisms are implemented through two different loops. The overall system is shown in Fig. 2 and is implemented according to the following procedure.

1. Whenever a set-point change is made to the system, the direct adaptive loop adjusts the value of τ_f according to (16) and (17) in Section 3 to ensure robustness of stability and performance.
2. At the same time, the indirect loop starts to collect input

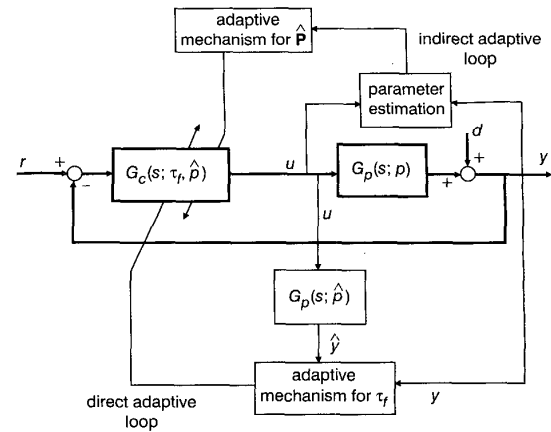


Fig. 2 Dual adaptive loop (DAL) system for robust and adaptive control

and output data to estimate the system parameters.

3. When the computed matrix, Φ , becomes full rank, identification computation starts along with increasing N . On-line estimation stops either when the parameter vector \mathbf{P} converges, or when the information contents of the matrix Φ become insufficient. In the latter case, the minimum singular value of Φ will smaller than an assigned limiting constant. In general, this small constant is taken as small as 1.0×10^{-3} .
4. The parameters of the controller are updated according to the identified nominal model, and the direct loop starts again using (6).

The weakness of the indirect adaptive loop is made up for by the direct adaptive loop, which guarantees stability of the system even when the resulting nominal model is not correct. The value of τ_f of the direct loop is changed back to its nominal designed value when on-line identification of the indirect loop is completed. As a result, τ_f will not drift if no further parameter changes occur.

6 Example—robust and adaptive PID control systems

Starting with a nominal model of (5) and using the IMC design to find C , the resulting G_c is a PID controller of the following form:

$$G_c(s) = k_c \frac{(1 + (1/\tau_R s) + \tau_D s)}{\alpha s + 1} \quad (29)$$

where the parameters are given as those of (7).

It is clear that the parameters in the PID controllers are functions of both τ_f and those parameters in \hat{G}_p . Therefore, by applying the adaptive mechanisms, the controller can be tuned on-line to obtain a self-tuning robust and adaptive control system. This self-tuned system can alleviate the difficulties and drawbacks pointed out by Chia and Cleveland [7].

The proposed DAL control system is demonstrated using the following examples, which use PID controllers.

6.1 Control with the proposed direct adaptive system

Example 1: Assume that the real plant and the nominal model are

$$G_p(s) = \frac{1.5}{2s + 1} \frac{229}{s^2 + 30s + 229} e^{-s} \quad (30)$$

$$\bar{G}_p(s) = \frac{1.4}{5s+1} e^{-s} \quad (31)$$

The above dynamic systems are simulated using the numerical Runge–Kutta method for integrating differential equations. The integration time sub-interval is taken as 0.005. The correct first-order model for $G_p(s)$ is found to be:

$$\bar{G}_p^*(s) = \frac{1.5}{2s+1} e^{-1.15s} \quad (32)$$

Because the given model has a modelling error compared with $G_p^*(s)$, the actual response has extensive oscillations as shown in Fig. 3.

If the direct loop is incorporated with the original IMC controller, τ_f is changed as shown in Fig. 4. Then the oscillations are attenuated. In this case, the modelling error is within reasonable limits, thus, after incorporating the direct adaptive loop, the performance is acceptable.

6.2 Control with the DAL adaptive system

Sometimes, deviations of the nominal parameters from the true ones are too large for the performance to be acceptable. Indirect adaptation via the system identification of

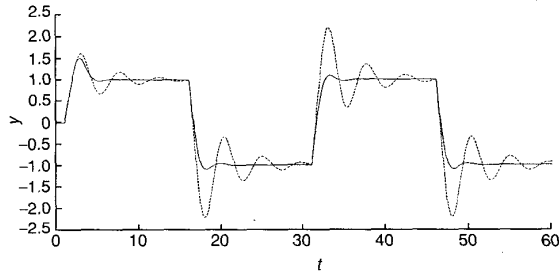


Fig. 3 Simulation results, example 1, system responses

— with direct adaptive control
 - - - without direct adaptive control

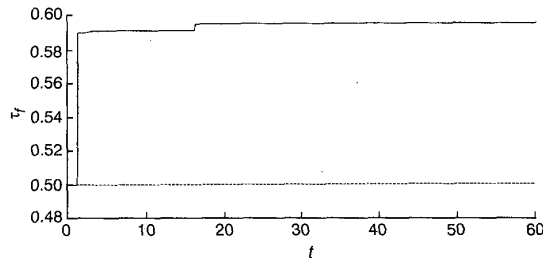


Fig. 4 Simulation results, example 1, values of τ_f

— with direct adaptive control
 - - - without direct adaptive control

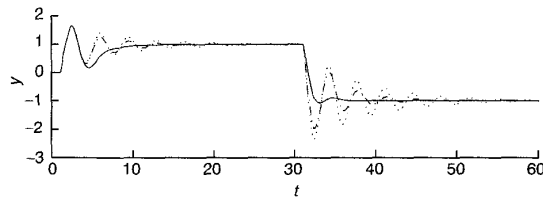


Fig. 5 Simulation results, example 2 without measurement noise, system responses

— with direct and indirect adaptive control
 - - - with direct adaptive control only
 without adaptive control

parameter sensitivity approach should be used in addition to a directed adaptation loop to obtain better performance. In the following, example 2 illustrates the advantage of adaptive control with dual loops over that of a direct adaptive loop alone.

Example 2: Consider the same real plant as in example 1, but that the nominal model with modelling error is given as

$$\bar{G}_p(s) = \frac{1.2}{3s+1} e^{-s} \quad (33)$$

Fig. 5 shows the responses of the three systems to step set-point changes, with the simulation parameters shown in Table 1.

Because this model has a large modelling error compared with G_p^* , the IMC control alone gives even greater oscillations than those of Example 1 at the output. Then, a direct adaptive loop is added and the τ_f is adapted as shown in Fig. 6. Although the oscillations are attenuated, the performance is still unacceptable. Finally, the indirect adaptive loop is incorporated along with the direct adaptive loop. Fig. 7 shows, the estimation results of the parameters during the identification. The system takes about five units of time to finish the recursive identification. This can be seen by the sudden change of τ_f to a new value. The second step set-point change occurs at $t=30$, and since the modelling errors have been corrected, the system at this point shows extremely good performance.

In the previous simulations, it is assumed that no measuring errors exists in the sampled data. If random errors having a normal distribution $N(0, 0.1)$ are added to the sampled output, the results are as shown in Figs. 8–10. As can be seen, the DAL adaptive scheme still works well in the presence of measurement noise.

Table 1: Simulation parameters used in example 2

Sampling interval	0.1 time unit
Initial data length	20 points
ϵ	0.05
$\underline{\epsilon}_{stop}$	0.001

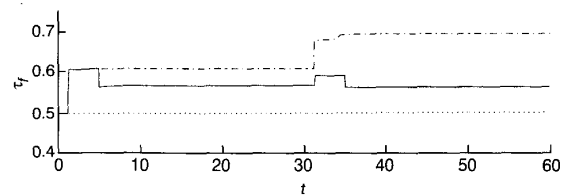


Fig. 6 Simulation results, example 2 without measurement noise, values of τ_f

— with direct and indirect adaptive control
 - - - with direct adaptive control only
 without adaptive control

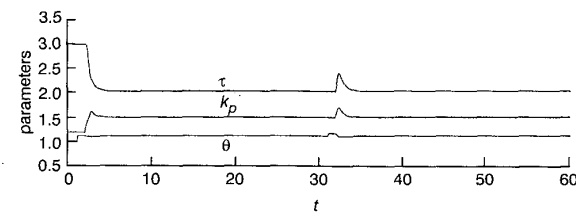


Fig. 7 Simulation results, example 2 without measurement noise, estimated plant parameters during identification

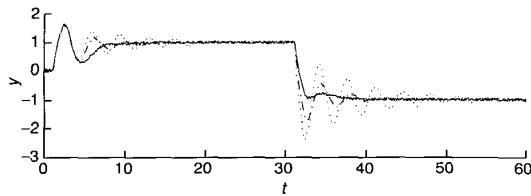


Fig. 8 Simulation results, example 2 with measurement noise, system responses

— with direct and indirect adaptive control
 - - - with direct adaptive control only
 ···· without adaptive control

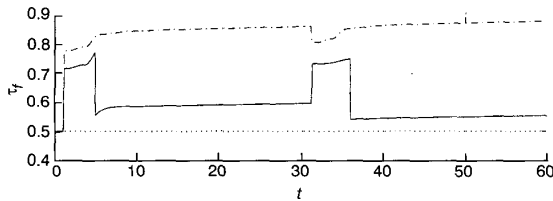


Fig. 9 Simulation results, example 2 with measurement noise, values of τ_f

— with direct and indirect adaptive control
 - - - with direct adaptive control only
 ···· without adaptive control

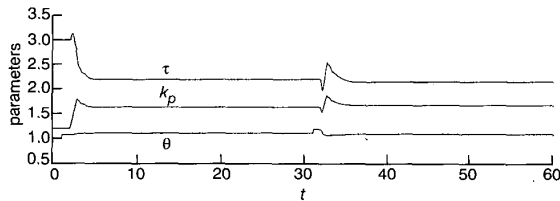


Fig. 10 Simulation results, example 2 with measurement noise, estimated plant parameters during identification

7 Concluding remarks

A DAL system for self-tuning adaptive control has been presented. The main advantages of this design include:

1. The system changes its controller parameters through normal closed-loop operation.
2. The system can achieve tighter control performance via a self-tuning procedure in the indirect adaptive loop and can achieve robust stability via the direct adaptive loop.

Simulation results show that the proposed DAL system can be used effectively on-line to auto-tune or self-tune PID controllers.

8 References

- 1 LANDAU, I.D.: 'From robust control to adaptive control', *Control Eng. Pract.*, 1999, 7, pp. 1113–1124
- 2 SKOGESTAD, S., and POSTLETHWAITE, I.: 'Multivariate feedback control—analysis and design' (John Wiley & Sons, 1996)
- 3 BHATTACHARYYA, S.P., CHAPPELLAT, H., and KEEL, L.H.: 'Robust control—the parametric approach' (Prentice-Hall, 1995)
- 4 SEBORG, D.E., and EDGAR, T.F.: 'Adaptive control strategies for process control: a survey', *AIChE J.*, 1986, 32, (6), pp. 881–913
- 5 KRSTIC, M., KANELLAKOPOULOS, I., and KOKOTOVIC, P.: 'Nonlinear and adaptive control design' (John Wiley & Sons, 1995)
- 6 IOANNOU, P.A., and SUN, J.: 'Robust adaptive control' (Prentice-Hall, 1996)
- 7 CHIA, T.C., and CLEVELAND, O.: 'Some basic approaches for self-tuning controllers', *Control Eng.*, 1992, pp. 49–52
- 8 ASTROM, K.J., and WITTENMARK, B.: 'Adaptive control' (Addison-Wesley, 1989)

- 9 NARENDRA, K.S., and ANNASWARY, A.M.: 'Stable adaptive systems' (Prentice-Hall, 1989)
- 10 ORTEGA, R., and TANG, Y.: 'Robustness of adaptive controllers—a survey', *Automatica*, 1989, 25, pp. 651–677
- 11 SASTRY, S., and BODSON, M.: 'Adaptive control: stability, convergence, and robustness' (Prentice-Hall, 1989)
- 12 BUTLER, H.: 'Model reference adaptive control: from theory to practice' (Prentice-Hall, 1992)
- 13 RIVERA, D.E., MORARI, M., and SKOGESTAD, S.: 'Internal model control 4. PID controller design', *Int. Eng. Chem. Process Design Dev.*, 1986, 25, pp. 252–265
- 14 BISWAS, K.K., and SINGH, G.: 'Identification of stochastic time delay systems', *IEEE Trans. Autom. Control*, 1978, 23, pp. 504–505
- 15 GERRY, J.P., VOGEL, E.F., and EDGAR, T.F.: 'Adaptive control of a pilot scale distillation column'. Proceedings of American Control Conference, 1983, San Francisco, USA, pp. 819–824
- 16 CHIEN, I.L., SEBORG, D.E., and MELLICHAMP, D.A.: 'Self-tuning controller for systems with unknown or varying time delays'. Proceedings of American Control Conference, 1984, San Diego, USA, pp. 905–912
- 17 LAMMERS, H.C., and VERBRUGGEN, H.B.: 'Simple self-tuning control of processes with a slowly varying time delay'. IEE Control Conference, 1985, Cambridge, UK, pp. 393–398
- 18 HUANG, H.P., and CHENG, C.C.: 'Transfer function matrix identification of MIMO dynamic systems', *Chin. I. Ch. E.*, 1989, 20, (1), pp. 31–39
- 19 LJUNG, L.: 'System identification—theory for the user' (Prentice-Hall, 1999, 2nd edn.)
- 20 HUANG, H.P., and WANG, G.B.: 'Control and estimation for perturbed nonlinear process with dead-time', *Chem. Eng. Commun.*, 1992, 117, pp. 117–141

9 Appendix

9.1 Proof of property 1

For a step change of set-point, we have $r(s) = 1/s$ and

$$\dot{y}(t = \theta_+) = \lim_{s \rightarrow \infty} s \{s y(s) e^{s\theta}\} \quad (34)$$

Because $G_c(s)$ is obtained by (6) and $y(s)$ is obtained by

$$y(s) = \frac{G_c(s)G_p(s)}{1 + G_c(s)G_p(s)} \frac{1}{s} \quad (35)$$

it is easy to see that

$$\dot{y}(t = \theta_+) = \frac{k_p \bar{\tau}}{\tau \alpha} \quad (36)$$

Furthermore, by the definitions of the PID parameters given in (7), the result is obtained as

$$\dot{y}(t = \theta_+) = \frac{k_p \bar{\tau}}{\tau_f k_p \tau} \quad (37)$$

□

9.2 Proof of property 2

According to the definition of ℓ ,

$$\ell(s) = \frac{G_p(s) - \tilde{G}_p(s)}{\tilde{G}_p(s)} \quad (38)$$

and because the steady state gains G_p and \tilde{G}_p are of the same sign, it is found that

$$\ell(0)F(0) \geq -1 \quad (39)$$

and

$$|\ell(j\omega)F(j\omega)| < \infty \quad \text{as } \omega \rightarrow \infty \quad (40)$$

According to (39), the Nyquist map of $\tilde{G}_+ F \ell$ always starts from a point on the real axis to the right of $(-1, 0)$. It can then be shown that for any $\ell(s)$, there exists a value, τ_f^{\min} , such that the Nyquist map of $\tilde{G}_+ F \ell(j\omega)$ will never cross the real axis at a point to the left of $(-1, 0)$ for all $\tau_f \geq \tau_f^{\min}$.

Since

$$|\bar{G}_+| = 1 \quad \forall \omega \in [0, \infty) \quad (41)$$

we have

$$|G_+ F \ell(j\omega)| \leq |\ell(j\omega)| |F(j\omega)| = \frac{|\ell(j\omega)|}{\sqrt{1 + \omega^2 \tau_f^2}} \quad (42)$$

Thus, for a given $\ell(s)$, assuming that the maximum value of

$$\frac{|\ell(j\omega)|}{\sqrt{1 + \omega^2 \tau_f^2}}$$

throughout $\omega \in [0, \infty)$ occurs at $\omega = \omega_0$, it is clear that

$$|G_+ F \ell(j\omega)| \leq \frac{|\ell(j\omega)|}{\sqrt{1 + \omega^2 \tau_f^2}} \leq \frac{|\bar{\ell}|}{\sqrt{1 + \omega_0^2 \tau_f^2}} \quad (43)$$

Here, $|\bar{\ell}|$ is the maximum of $|\ell(j\omega)|$ for $\forall \omega \in [0, \infty)$. For $\bar{\ell} \leq 1$, the above inequality holds for each $\tau_f \geq 0$. In the case where $\bar{\ell} > 1$, there is a value for τ_f^{min} of the following, i.e.

$$\tau_f^{min} = \frac{\sqrt{|\bar{\ell}|^2 - 1}}{\omega_0} \quad (44)$$

Here, for any $\tau_f \geq \tau_f^{min}$, the following result holds true:

$$|G_+ F \ell(j\omega)| \leq 1 \quad (45)$$

Thus, it can be concluded that, for each τ_f which is greater than τ_f^{min} obtained by (44), the system can be stabilised with the IMC-PID controller. \square