A FREQUENCY DOMAIN BASED SELF-TUNING PID CONTROLLER

J.V. Ringwood[*] and A. O'Dwyer[**]

[*]School of Electronic Engineering
Dublin City University
Glasnevin, Dublin 9, IRELAND

[**]Dept. of Control Systems and Electrical Eng.
Dublin Institute of Technology
Kevin St., Dublin 8, IRELAND

Abstract.   Traditionally, both explicit and implicit self-tuning controllers have employed time domain techniques for the identification and tracking of plant and controller parameters. The use of the frequency domain provides concise information on the dynamics of the process which has led to its wide acceptance as a domain for controller design. This paper demonstrates a method employing recursive, on-line measurement of the process frequency response, with a straightforward calculation of PID controller parameters. The computational effort involved is comparable with that of a time domain technique.

Keywords. Self-tuning regulator, Frequency domain, PID control

1. INTRODUCTION

   PID self-tuning algorithms utilising time domain identification techniques are widely reported in the literature. Traditionally, such methods incorporate some form of time domain identification based on a parameterised model with a set of design equations relating controller parameters to plant parameters; examples include methods by Banyasz and Keviczky (1982) and Tjokro and Shah (1985). One problem, however, with such techniques is the nesessity to impose a model structure on the system, which introduces approximation, even when best-fit parameters for such models are available. A further difficulty is the on-line identification of time delay for parameterised models. An explicit delay term cannot be incorporated into linear identification schemes (overparameterised models become impractical for more than 2 to 3 steps delay) and nonlinear schemes have achieved limited success (Durbin (1985)).

Another significant factor in designing PID self-tuners is the difficulty of relating PID parameters to process transfer function parameters. In many cases, recourse is made to the frequency domain, resulting in complex design equations due to time/frequency domain changes (Kofahl and Isermann (1985)). Some researchers have looked at the frequency domain as a suitable starting point for PID self-tuning. Off-line techniques are reported by Astrom and Hagglund (1984) and Tachibana (1984). More recently, the on-line case has been examined by Astrom and Wittenmark (1991) and Kallen and Wittenmark (1993), both using simple time domain models to evaluate gain and phase. A somewhat similar approach is taken by Lamaire *et al* (1991). A paper by Balchan and Lie (1987) describes an adaptive controller based on measurement of the *closed-loop* frequency response.

In this paper, an attempt is mode to measure the process frequency response directly. In particular, the gain and frequency at the phase crossover point ($\phi = -\pi$) is of interest. To determine this frequency, a simple algorithm is used to perform adaptation of the frequency variable, based on process phase measurements. Gain and phase measurements are performed based on numerical integration of the Discrete-Time Fourier Transform (DTFT). A selection of suitable techniques, which offer a certain intuitive appeal, are given in Section 2.1. Controller design, which utilises the closed-loop Ziegler-Nichols (1942) relations, is presented in Section 4.

2. PROCESS FREQUENCY RESPONSE MEASUREMENT

2.1 Recursive Fourier Transform Calculations:
   The system frequency response is calculated based on measurements of the Fourier Transform of input and output signals. For an open loop system, with an input signal n(t) and an output signal c(t), the plant frequency response is trivially evaluated as:

$$G(j\omega) = C(j\omega)/N(j\omega) \qquad (1)$$

with $C(j\omega)$ and $N(j\omega)$ being the Fourier Transforms of c(t) and n(t), respectively. A recursive technique for calculating the transforms is appropriate. One such method is to use the Discrete Time Fourier Transform (DTFT), defined as follows:

$$F(\omega) \;=\; T\sum_{k=0}^{\infty} f(kT)\, e^{-j\omega kT} \qquad (2)$$

This transform has the advantage that a new term may be added as new data points become available; a further advantage is that the frequency variable is continuous, which allows more accurate calculation of the phase crossover frequency. The DTFT could be modified by including tapering on the data window at the start and current evaluation points of the summation; this proposal would reduce spectral leakage. The inclusion of a non-rectangular data window would however increase the computational complexity of the calculation. An alternative recursive method for finding the transforms is to apply a numerical integration technique to the Fourier transform.. An example of suitable techniques is the Adams-Moulton set, as discussed by Johnson and Reiss (1982). The first four of this set are as follows:

$$F_{k+1} - F_k = Tx_{k+1} \tag{3}$$

$$F_{k+2} - F_{k+1} = \tfrac{T}{2}(x_{k+2} + x_{k+1}) \tag{4}$$

$$F_{k+3} - F_{k+1} = \tfrac{T}{12}(5x_{k+3} + 8x_{k+2} + x_{k+1}) \tag{5}$$

$$F_{k+4} - F_{k+3} = \tfrac{T}{24}(9x_{k+4} + 19x_{k+3} - 5x_{k+2} + x_{k+1}) \tag{6}$$

$$\text{where} \qquad x_k = f(kT)e^{-j\omega kT}$$

Equations (3) and (4) may be readily identified as the backward difference and trapezoidal rule (bilinear transform) respectively. Assuming a start from k=0 and zero initial conditions, the first four terms of the integrals in (3) and (4) become:

$$I_3 = T[x_0 + x_1 e^{-j\omega T} + x_2 e^{-j\omega 2T} + x_3 e^{-j\omega 3T}] \tag{7}$$

$$I_3 = T[\tfrac{1}{2}x_0 + x_1 e^{-j\omega T} + x_2 e^{-j\omega 2T} + \tfrac{1}{2}x_3 e^{-j\omega 3T}] \tag{8}$$

Note that (7) displays a DTFT. However, (8) demonstrates a DTFT with a data window which is tapered at each end. Higher order techniques exaggerate this windowing effect.

2.2 Beat Frequencies

From the definition of the DTFT in equation (2), it can be seen that product terms arise between sinusoidal signals in f(kT) and the exponential term. Since an average (or sum) of the product of sinusoids of different frequencies is zero, the only term which is non-zero is the product term involving a sinusoid at the DTFT frequency. This $\sin^2(\omega kT)$ term may be recast into a $\tfrac{1}{2}(1 - \cos(2\omega kT))$ term, involving a *beat* frequency at twice the DTFT frequency. A difference equation for the phase of the system evaluated using the DTFT can be found as:

$$\phi_k(\omega) = \phi_{k-1}(\omega) - \tan^{-1}\left\{\frac{\cos(\Phi) - \cos(2\omega kT + \Phi)}{\sin(\Phi) + \sin(2\omega kT + \Phi)}\right\} \tag{9}$$
$$- \tan^{-1}\left\{\frac{\cos(2\omega kT) - 1}{\sin(2\omega kT)}\right\}$$

After convergence, $\phi_k = \phi_{k-1} = \Phi$ (on average), but the phase measurement continues to vary according to the latter two terms in (9) which involve the beat frequency. However, as the DTFT frequency approaches the phase crossover frequency, where $\Phi \to -\pi$, a trivial calculation shows that these terms go to zero. It may be demonstrated that an attenuation inversely proportional to the difference between the DTFT frequency and the phase crossover frequency is achieved.

Low pass filters on gain and phase estimates are used to reduce the effect of beat frequencies. These are based on first order differences and have a cut-off frequency below 2ω. Alternatively, band pass filters or filters with a variable cut-off frequency could be employed for improved performance.

2.3 Data Forgetting

An important feature of either of the recursive schemes outlined above is that new terms are constantly being added as time progresses. This may lead to two difficulties:

(a) The size of the DTFT's may become very large, and
(b) The algorithm may become insensitive to changes in the process dynamics or evaluation frequency, due to the magnitude difference between the new terms being added and the current size of the transform.

The magnitude difference in (b) is typically of the order of 10^6. A form of data forgetting may be implemented to maintain a reasonable balance between the orders of magnitude of the transforms and their increments. An example of such a method involves weighting the data values by progressively smaller amounts as they recede in time. A forgetting factor, λ, is introduced as follows:

$$F_{k+1}(\omega) = \lambda F_k + g(x) \tag{10}$$

The first order DTFT with a rectangular data window has the form:

$$F_{k+1}(\omega) = \lambda F_k(\omega) + Tx_k \tag{11}$$

with $0 < \lambda \le 1$.

2.4 Identification in Closed Loop

To aid identification in closed-loop, an excitation signal at the appropriate (Fourier transform) frequency is added to the control signal. This signal, while not having any adverse effects on the regulation properties of the system, would seem to be sufficient to allow consistent identification of the open-loop frequency response in closed-loop, based on a related analysis by Wellstead (1986). The amplitude, A_o, of the sinusoidal excitation signal should be commensurate with the amplitude of the measurement noise at d(t). This excitation signal is preferable, from a regulation point of view, to the sharp-edged excitation signals associated with time-domain identification.

A further practical addition of band-pass filters with moveable centre frequency is included to concentrate calculations on the frequency range of interest. This helps to improve the disturbance and noise rejection properties of the adaptation algorithm. A Butterworth design is used with transfer function:

$$G_{bp}(z) = \frac{z^2 - 1}{z^2 - (1+\alpha)\beta z + \alpha} \tag{12}$$

where

$$\beta = \cos(\omega T)/\cos(\tfrac{\omega_{bw}T}{2})$$

α is a parameter determined from the equivalent low-pass design and depends only on the filter bandwidth, ω_{bw}, and the sampling period, T. ω is the centre frequency of the band-pass filter.
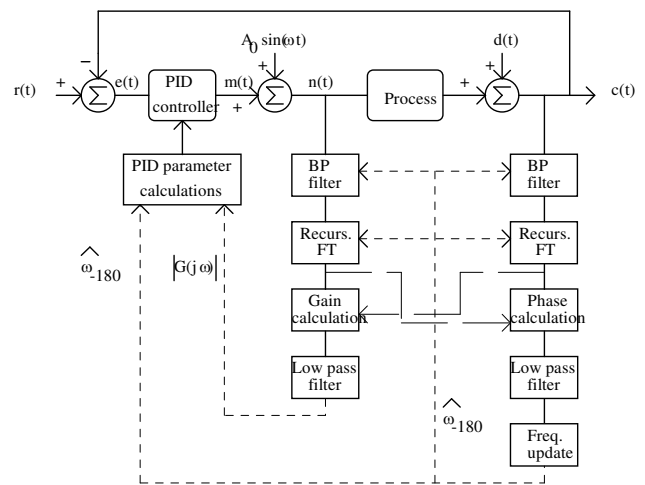

Fig. 1. Block diagram of the closed loop system.

3. FREQUENCY UPDATING

3.1 Update Method

The procedure for controller tuning discussed in Section 1 demonstrates that adjustment must be made to the evaluation frequency of the Fourier transform until the phase crossover frequency is calculated. It is proposed to extrapolate from previous phase and frequency values to determine the phase crossover frequency. Gradient algorithms, which allow updating of the frequency based on the slope of the phase versus frequency curve, are appropriate for a large class of plants in which phase lag increases continuously as frequency increases. One such algorithm is the Least Mean Squares (LMS) algorithm, as described by Widrow and Stearns (1985):

$$\omega_{i+1} = \omega_i - 2\mu\, e_i \frac{\partial e_i}{\partial w_i} \qquad (13)$$

where ω_{i+1} = new estimate of the phase crossover frequency, ω_i = current estimate of the phase crossover frequency, μ = adaptation constant and $e_i = -\pi - \phi_i$ = phase error (with ϕ_i = current phase estimate). If the transfer function of the plant is unknown, then one approximation for $\partial e_i / \partial \omega_i$ is:

$$\frac{\partial e_i}{\partial \omega_i} = -\frac{\partial \phi_i}{\partial \omega_i} \approx -\frac{\phi_i - \phi_{i-1}}{\omega_i - \omega_{i-1}} \qquad (14)$$

In these circumstances, the algorithm becomes

$$\omega_{i+1} = \omega_i - 2\mu(\pi + \phi_i)(\phi_i - \phi_{i-1})/(\omega_i - \omega_{i-1}) \qquad (15)$$

Other more computationally intensive gradient algorithms that may be used include the steepest descent algorithm, the Gauss-Newton algorithm and the Levenberg-Marquardt algorithm (Ljung, 1987); in general, these algorithms would facilitate faster adaptation than would the LMS algorithm. An alternative approach to that discussed above is to use a number of data points and fit a high order polynomial for the phase to the data. The parameters of the polynomial could be found using an estimation strategy such as least squares. The simplest algorithm of this type would be to fit a straight line to two data points; the updated estimate of the phase corssover frequency is then given by:

$$\omega_{i+1} = \omega_i - \frac{\delta}{m}(\pi + \phi_i) \qquad (16)$$

where $m = (\phi_i - \phi_{i-1})/(\omega_i - \omega_{i-1})$ and $0 < \delta \le 1$. δ may be considered as an uncertainty factor that reflects the general non-linear nature of the phase response. If no *a priori* knowledge of the plant is available, a value of $\delta = 0.7$ gives a reasonable trade off between speed of convergence (towards $\phi_i = -\pi$) and phase response non-linearity. This algorithm is used in the simulation work in Section 5.

3.2 Update Regularity

The principal consideration is to allow the phase and gain measurements to settle (given that recursive techniques are used in the estimation), while retaining a reasonable rate of convergence to the $\phi_i = -\pi$ point. In practice, it has been found that it takes approximately 100 samples for a trapezoidal integration technique to settle using a forgetting factor of $\lambda = 0.97$. The choice of forgetting factor is determined by a tradeoff between convergence speed and noise immunity. A value as low as 0.8 can be used in a noise-free environment, giving rapid convergence and response to time varying systems, while a value

closer to 0.99 is required to average out the effect of noise. The update regularity should therefore be chosen in unison with λ.

4. CONTROL SYSTEM DESIGN

4.1 PID Controller Setting

In the continuous time domain, the Ziegler-Nichols tuning rules may be implemented, knowing the gain margin and the phase crossover frequency. In the discrete time domain, Kofahl *et al.* (1985) have specified appropriate tuning rules based on the continuous time Ziegler-Nichols rules. The digital controller is defined by :

$$m(k) = m(k-1) + q_0 \cdot e(k) + q_1 \cdot e(k-1) + q_2 \cdot e(k-2) \qquad (17)$$

with

$$q_o = K_c[1 + \frac{T}{2T_i} + \frac{T_d}{T}] \qquad (18)$$

$$q_1 = -K_c[1 + \frac{2T_d}{T} - \frac{T}{2T_i}] \qquad (19)$$

$$q_2 = K_c \frac{T_d}{T} \qquad (20)$$

where T is the sample period and K_c, T_i and T_d are the proportional gain, integral time constant and derivative time constant, respectively, of a corresponding analogue controller. Kofahl and Isermann (1985) suggest the following tuning rules:

$$K_c < 0.6K_u \;\;,\;\; T_i = 0.5T_u \;\;,\;\; T_d = 0.12T_u \qquad (21)$$

where K_u = ultimate gain and T_u = ultimate period.

4.2 PID Caution Control

The PID controller defined in equation (17) is not implemented until the phase crossover frequency is found and the process gain evaluated at this point. A form of caution control is used until the phase crossover frequency is found. This is done by reducing K_c in relation to the distance between the current phase value and $-\pi$; this will guarantee safe control. A suitable relation between K_c and ϕ_i has been found to be:

$$K_c = (K_c^* / |G_p(j\omega)|)e^{-\gamma|\phi_i + \pi|} \qquad (22)$$

γ is a design parameter which determines the degree to which the controller will be detuned. A value of $\gamma = 2$ has been found to be appropriate. At initialisation, small values of K_c and T_d and a large value of T_i are assigned. These values guarantee safe control. K_c^* is the nominal controller gain.

5. RESULTS

The performance of the proposed algorithm is demonstrated using simulation tests. The model used for the process is:

$$G_p(z) = \frac{0.11138z^{-1} + 0.09911z^{-2}}{1 - 1.684z^{-1} + 0.7047z^{-2}} z^{-2} \qquad (23)$$

for a sampling period of 0.2 secs. The following values for the design parameters were used:

Forgetting factor, λ	0.95
LP filter time constant	10 secs.
Frequency update every	500 samples
BPF parameter, α	-0.7387

BPF bandwidth, ω_{bw} 3 rads/sec
Caution control parameter, γ 2

The trapezoidal integration method in equation (4) was used for Fourier transform evaluation. The use of higher order methods is normally only merited for relatively large sampling periods.

Fig.2 shows the variation in frequency, gain and phase at startup. Note that the frequency update has been left purposely slow to facilitate clear demonstration of the gain and phase convergence at the different frequencies and the operation of the caution control. Update regularities of less than 100 samples have been found to be adequate. In spite of the filtering of the gain and phase estimates, the effect of beat frequencies is apparent from Fig.2. This effect is seen to diminish as the phase approaches $-\pi$ (see Section 2.2) and could be further reduced by increasing the order of the low pass filter. Fig.3 demonstrates the detuning effect of the caution control, with a progressive increase in controller gains as the phase crossover frequency is approached.
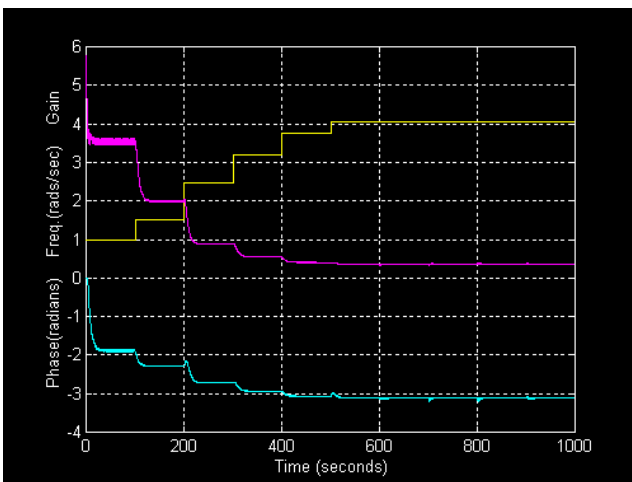

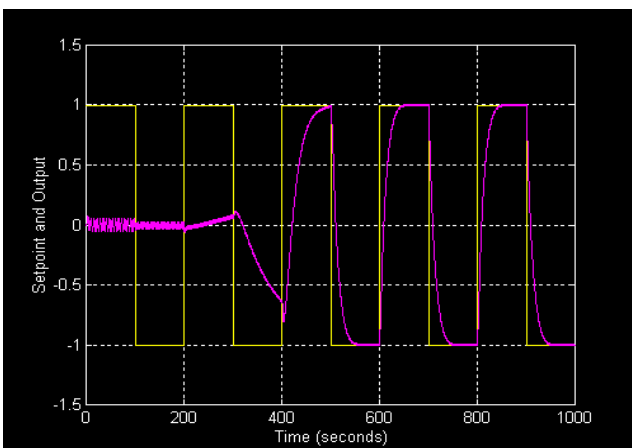Fig.2. Convergence on phase crossover frequency

Fig.3. Regulation properties of the system

6. CONCLUSIONS

A self-tuning controller has been presented, based on frequency-domain calculations. One advantage of this is that no process parameterisation is required. The computational effort, summarised is comparable with that of a time-domain self-tuner.

In addition, the algorithm contains design parameters not dissimilar to a time-domain algorithm. These generally involve a trade-off between speed of tuning and noise immunity. One feature of the technique in this paper is the easy addition of caution control, since a direct measure of the tuning error (i.e. phase error) is available.

The algorithm could also be extended to include explicit time delay estimation, since this effect (linear phase shift with frequency) can be resolved from the overall phase measurement. Such an extension is not possible with parametric time-domain schemes.

7. REFERENCES

Astrom, K.J. and T. Hagglund (1984), Automatic tuning of simple regulators with specifications on phase and amplitude margins, Automatica, Vol.20, No.5.

Balchen, J.G. and Lie, B. (1987) An adaptive controller based on continuous estimation of the closed loop frequency response, Modelling, Identification and Control, Vol.8, No.4, 223-240.

Banyasz, C. and Keviczky, L. (1982), Direct methods for self-tuning PID regulators, Proc. 6th IFAC Symp. on Identification and System Parameter Estimation, Washington DC.

Durbin, L.D. (1985), Deadtime approximations with adaptive deadtime compensation, Proc. ACC, Boston.

Hagglund, T. and K.J. Astrom (1991), Industrial adaptive controllers based on frequency response techniques, Automatica, Vol.27, No.4, 599-609.

Johnson, L.W. and R.D. Reiss (1982), Numerical Analysis, 2nd Edition, Addison-Wesley.

Kallen, P-O. and B. Wittenmark (1993), A frequency domain adaptive controller, Proc. IFAC World Congress, Sydney.

Kofahl, R. and R. Isermann (1985), A simple method for automatic tuning of PID-controllers based on process parameter estimation, Proc. of the American Control Conference (ACC 85), 1143-1148, Boston.

Lamaire, R.O., L. Valavani, M. Athens and G. Stein (1991), A frequency-domain estimator for use in adaptive control systems, Automatica, Vol.27, No.1.

Ljung, L. (1987), System Identification: Theory for the User, Prentice-Hall Inc., New Jersey.

Tachibana, Y. (1984) Identification of a system with dead time and its application to auto tuner, Elec. Eng. in Japan, Vol.104, No.2.

Tjokro, S. and S.L. Shah (1985), Adaptive PID control, Proc. ACC, Boston, 1528-1534.

Wellstead, P.E. (1986), Lecture Notes in Control and Information Sciences, Vol 79: Signal Processing for Control, (K. Godfrey, P. Jones Eds.), Springer-Verlag, 210-222.

Widrow, B. and S.D. Stearns (1985), Adaptive Signal Processing, Prentice-Hall Inc., New Jersey.

Ziegler, J.G. and N.B. Nichols (1942), Optimum settings for automatic controllers, Trans. ASME, Vol.64, No.11.