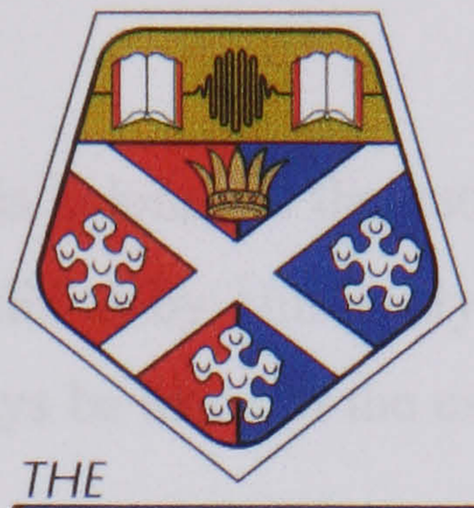
PID Controller Tuning Methods for Process Industry Applications.

James Crowe



Industrial Control Centre
Department of Electronic and Electrical Engineering
University of Strathclyde
Graham Hills Building
50 George Street
Glasgow, G1 1QE
Scotland, UK

Thesis submitted to the
University of Strathclyde
For the degree of
Doctor of Philosophy

March, 2004

Copyright:

Dedication:

To my wife Fiona and our sons Calum and Harrison; for tolerating too many lost weekends.

Acknowledgements:

I would like to thank my supervisors Prof. Michael A. Johnson and Prof. Michael J. Grimble for their untiring enthusiasm and support throughout the years. I would especially like to thank Prof. Michael A. Johnson for sharing his knowledge of Control Engineering and Mathematics and for all of the discussions that have shaped this thesis.

Abstract

A discussion of identification methods employed in process control applications is carried out. The identification methods discussed range from explicit modelling techniques based on the relay experiment and the Phase-Locked Loop methods of non-parametric system identification, through to the implicit modelling techniques of sub-space identification and the model-free methods used in Iterative Feedback Tuning.

For a given range of gain and phase margins, graphical methods are developed that show the viable gain margin and phase margin design pairings that are achievable by the use of a PI controller as the compensation element in a closed loop control system. Two further graphical methods that allow the parameters of a PID controller to be determined such that gain and phase margin design specifications can be met are discussed.

Iterative tuning methods that allow the design of PI controllers to meet gain and phase margin specifications are developed. An extension of the iterative tuning method that allows the design of PI controllers to meet maximum sensitivity and phase margin design specifications is also discussed.

The Phase-Locked Loop (PLL) method of system identification is used to carry out the closed loop identification and tuning of cascade connected control systems. The closed loop identification of multivariable systems using the PLL method of system identification and the design of a decentralised control system based on an extension to the exact gain and phase margin design method is discussed.

The Iterative Feedback Tuning (IFT) method of restricted structure controller design is discussed. A new method, Controller Parameter Cycling (CPC), is introduced. The CPC method of controller tuning allows the determination of both the cost function gradient and Hessian from experiments that are carried out on the closed loop system. Thus, improved numerical techniques can be used by the CPC method over those employed in the IFT method.

0 Preface

0.1 Motivation

In process industries the PID control algorithm implemented in pneumatic, analogue electronic or digital formats is used predominately for the control of single loop, cascade or multiple loop and multivariable processes. The use of microprocessor based PID controllers has seen an increase in the number of manufacturers who offer an auto-tune function as standard on their products. The majority of commercially produced auto-tuners utilise the relay experiment of Astrom and Hagglund (1984) to identify the phase crossover point of the frequency response of the process that is to be controlled. Tuning of the PID controller is then carried out using the data obtained from the relay experiment and the application of a rule based method or a simple parametric model of the process is produced and the PID controller is tuned from the basis of the model.

The Phase-Locked Loop (PLL) method of nonparametric system identification (Crowe, 1998; Crowe and Johnson, 1998; Johnson and Crowe, 1998) was developed as a direct result of research carried out on the relay experiment of Astrom and Hagglund (1984). The objective of the research was to provide a nonparametric identification method that would:

i) have the ease of use of the relay experiment,

ii) supply accurate estimates of the phase crossover point, and

iii) provide more accurate estimates of the phase crossover point in the presence of measurement noise or process disturbance, than the relay experiment.

It soon became apparent that the PLL method of system identification offered a greater flexibility in its use than did the relay experiment.

The motivation for the research contained in this Thesis was to determine how the PLL method of nonparametric system identification could be utilised such that material improvements in the functionality offered by auto-tuners based on the PLL method would be achieved over that offered by relay based auto-tuners.

0.2 Achievements of the Research

The work contained in this thesis builds and significantly extends the research into the Phase-Locked Loop method of non-parametric system identification as given by Crowe (1998).

An extension to the Phase-Locked Loop (PLL) method of non-parametric system identification has been developed to allow the open loop identification of type 1 processes.

For an unknown linear time invariant process, connected in closed loop, a graphical method was developed showing the viable gain and phase margin design pairings that are achievable by using a PI controller. An enumeration technique was used to develop graphical methods that show the range of gain and phase crossover frequencies at which, using a set of derived equations, the parameters of a PID controller can be determined to achieve a specific gain and phase margin design. A theorem relating to the enumeration method was produced.

A second graphical technique relating the PID controller gain parameter, k_p, to the gain and phase crossover frequencies was developed. The use of the method allows the remaining integral and derivative gain terms, k_i and k_d, to be calculated from a set of equations such that a specific gain and phase margin design can be achieved.

An iterative method was developed that allows the parameters of a PI controller to be determined such that a specified gain margin and phase margin design can be achieved. The iterative approach was extended such that the parameters of a PI controller could be found such that a specific maximum sensitivity and phase margin can be obtained. A theorem relating to the convergence of the iterative method was developed.

A method was developed that allows the tuning of a cascade connected control loop to be carried out with the cascade system remaining in closed loop. The method allows for a test to be carried out ensuring that the cascade system will remain stable when the inner controller parameters are updated online. The application of the Phase-Locked Loop method of non-parametric system identification was extended to include the closed loop identification of multivariable processes. A method was proposed for

the extension of the Fung *et al* (1998) exact gain and phase margin tuning method for use with multivariable processes.

A model-free method for tuning restricted structure controllers was developed. The method known as Continuous Parameter Cycling (CPC) allows the gradient and Hessian of a cost function to be determined from experiments that are carried out on the closed loop system. The CPC method thus allows improved numerical routines to be used over those used in Iterative Feedback Tuning. Propositions relating to the extraction of the gradient and Hessian data and proofs of those propositions were developed.

Two Journal and eight conference papers have been produced and published or have been accepted for publication that include the results of the research, they are:

J. Crowe and M.A. Johnson, 1999, A New Non-Parametric Identification Procedure for Online Controller Tuning, American Control Conference, (3337 – 3341), San Diego, U.S.A., 2 – 4, June.

J. Crowe and M.A. Johnson, 2000, Automated PI Controller tuning Using a Phase Locked Loop Identifier Module, IECON 2000, IEEE International Conference on Industrial Electronics, Control and Instrumentation, Nagoya, Japan, 22 – 28, October.

J. Crowe and M.A. Johnson, 2000, Open and Closed Loop Process Identification by a Phase Locked Loop Identifier Module, ADCHEM 2000, IFAC International Symposium on Advanced Control of Chemical Processes, Pisa, Italy, 14 – 16, June.

Crowe, J., M. A. Johnson and J. Wilkie, 2001, Recent developments in PID control for process control applications, CPACT conference in Advances in Process Analytics and Control, Glasgow, UK, 3-4, April.

J. Crowe and M.A. Johnson, 2001, Automated PI control tuning to meet Classical Performance Specifications Using a Phase Locked Loop Identifier, American Control Conference, Arlington, Virginia, USA, 25 – 27, June.

J. Crowe and M.A. Johnson, 2001, PID Tuning for Classical Robustness Specifications by Enumeration Methods, IECON 2001, IEEE International Conference on Industrial Electronics, Control and Instrumentation, Denver, USA, 29 Nov.– 02 Dec.

J. Crowe and M.A. Johnson, 2002, Automated Maximum Sensitivity and Phase Margin Specification Attainment in PI Control, Asian Journal of Control, Vol. 4, No. 4, December.

J. Crowe and M.A. Johnson, 2002, Toward Autonomous PI Control Satisfying Classical Robustness Specifications, IEE Proceedings: Control Theory and Applications, Vol. 149, No. 1, January 2002.

Crowe, J., M. A. Johnson and M. J. Grimble, 2003, On the Closed Loop Identification of Systems within Cascade Connected Control Strategies, European Control Conference, University of Cambridge, UK, 1 – 4 September.

Crowe, J., M. A. Johnson and M. J. Grimble, 2003, PID Parameter Cycling to Tune Industrial Controllers – A new model-free approach, SYSID, 13th IFAC Symposium on System Identification, Rotterdam, The Netherlands, 27 – 29 August.

0.3 Layout of the Thesis

Chapter 1 begins with a categorisation of identification methods that are used in process industries into explicit, implicit and model-free techniques. Under the heading of explicit modelling methods the relay experiment and the Phase-Locked Loop (PLL) methods of non-parametric system identification are discussed, along with some extensions to the use and operation of the PLL technique of system identification. Under the heading of implicit modelling methods a brief introduction to sub-space identification is given. This acts as a bridge between the explicit modelling methods discussed and the model-free techniques that are employed in Iterative Feedback Tuning and Continuous Parameter Cycling.

The development of graphical methods to aid in the selection of gain margin and phase margin designs for PI and PID controllers is carried out in Chapter 2. For a PI controller and an unknown process, a procedure is developed that allows all of the achievable pairings of gain and phase margin designs, in a given range of values, to be presented graphically. Two semi-graphical methods were developed for an unknown process in closed loop with an unknown PID controller that allow the controller parameters to be determined such that a specific gain and phase margin design can be achieved. In the first method, by using an enumeration technique a graphical representation of candidate gain and phase crossover frequencies at which, using a set of derived equations, the parameters of a PID controller can be calculated that results in a specific gain and phase margin design being met. For the second graphical technique a method was developed that relates the PID controller gain parameter, k_p, to the gain and phase crossover frequencies. By using the resulting graph and a set of derived equations the remaining integral and derivative gain terms for the PID controller are calculated such that a specified gain and phase margin is achieved.

In Chapter 3 iterative design methods are employed to determine the parameters of PI controllers to meet classical measures of robustness. The iterative design of a PI controller, in closed loop with an unknown process, to meet a specific gain and phase margin is detailed. The iterative design method is further developed to allow the design of a PI controller such that a maximum sensitivity and gain margin specification can be achieved.

The application of the Phase-Locked Loop (PLL) method of system identification is extended in Chapter 4 to carry out the closed loop identification of cascade and multivariable control systems. The tuning of a cascade connected control system operating in closed loop is detailed. The method allows tests to be carried out that ensure that the closed loop cascade system will remain stable when the inner controller parameters are updated. An extension to the exact gain and phase margin design method due to Fung *et al* (1998) that allows a gain and phase margin design to be carried out on a multivariable process is discussed.

Chapter 5 begins with a discussion of the Iterative Feedback Method (IFT) of controller tuning due to Hjalmarsson *et al* (1994, 1998). IFT utilises a series of experiments on the closed loop system to extract the gradient of the cost function with

respect to the controller parameters. Hence, by the use of a stochastic estimation routine the controller parameters can be determined that will minimise the cost function. Due to the somewhat general problem definition used by Hjalmarsson *et al* the simplicity of the method tends to be lost. A deterministic version of IFT due to Mahathanakiet *et al* (2002) that does not obscure the simplicity of the IFT method is also discussed. A new model-free iterative design method for restricted structure controllers termed Continuous Parameter Cycling (CPC), is introduced and discussed. The CPC method uses a time varying perturbation of the controller parameters to produce a time varying cost function. The gradient and Hessian data are then extracted from the time varying cost function. The availability of the gradient and Hessian data then allow improved numerical routines to be used, over those employed in IFT, to determine the controller parameters that will minimise the cost function.

Chapter 6 contains conclusions on the work discussed in the preceding chapters. A discussion of future possible research directions resulting from the research reported in the thesis is also given. References close the thesis.

1 Closed Loop Identification Methods for Process Control Applications.

1.1 Introduction.

The object of the research presented in this thesis is to provide a suite of tools, operating in the frequency domain, that can be incorporated into an autotuner that will carry out the required:

i) Identification

ii) PID controller existence testing, and

iii) PID controller design.

Further, the autotuner should have as high a degree of autonomy as possible such that the operator only requires to enter the desired frequency domain specifications that the closed loop control system is required to obtain and the configuration of the control system, viz. Single loop, cascade loop, or multivariable.

The three term or Proportional, Integral and Derivative (PID) controller is used extensively within the process industries to provide regulatory control of single loop, multi-loop (cascade) and multi-input multi-output control schemes. The technology in which the PID controller has been implemented has undergone many changes, ranging from pneumatic, analogue electronic, direct digital control and most recently as an algorithm within a programmable electronic system. Where there has been less of a change is in the methods used to tune the PID controllers to give an acceptable degree of control system performance. A review of the literature on rule based tuning methods for PID controllers (O'Dwyer, 1998a; 1998b) shows that there is an extensive range of tuning rules available that would provide the required degree of control system performance, and so it could be thought that PID controller tuning would not pose a problem. However, in practice it is found that the majority of PID control loops installed in the process industries are poorly tuned or that they are adjusted manually (Hersh and Johnson, 1997). In a typical process plant there are several hundred PID controllers in use. These controllers are used to control processes that in general have relatively long time constants ranging from a few minutes to a few hours. Hence for the control practitioner it may not be possible to provide the time necessary to tune an individual loop to the required degree of

control system performance and hence a compromise solution is attained. A further complication to the problem of providing a satisfactory control performance is that a process model may be necessary to carry out the controller tuning or at least an understanding of the dynamics of the process to be controlled is required. This information may not be readily available and hence some form of process identification shall be required. In carrying out the modelling of the process and the design of the PID controller a high degree of skill and knowledge is required by the control practitioner. PID controller design in the process industries commonly uses one of the experiment-based methods due to Ziegler and Nichols (1942); now routinely implemented as an autotuning function by most PID controller manufacturers. These experiment-based methods are the process reaction curve and ultimate period controller tuning methods. The Ziegler-Nichols paper, which was published in 1942, stated as the motivation for a rule-based PID controller design method that *"the mathematics of control involves such a bewildering assortment of exponential and trigonometric functions that the average engineer cannot afford the time necessary to plough through them to a solution of his current problems"*. This has been a driving force behind much of PID rule based design for the process industries ever since. Both the process reaction curve and ultimate period methods required an experiment to be carried out to identify certain parameters of the model of the process to be controlled followed by the application of a rule based controller tuning parameter selection.

The utility of the Ziegler and Nichols methods is that the information supplied from relatively simple experiments allows the choice of the controller tuning parameters to be made by a rule based method. In recent years there has been a trend to introduce controller tuning methods that do not require a high degree of skill on the part of the control practitioner to implement a control scheme that gives an acceptable control performance. With the introduction of PID controllers based on electronic technology, Astrom and Hagglund (1984) introduced the first push-button auto-tune method. The method is based on the Ziegler and Nichols Ultimate Period method in which information relating to the process at the phase crossover frequency is required. In the Astrom and Hagglund method the experiment is carried out in closed loop with the controller replaced by a relay. The result of this is the generation

of a limit cycle at a frequency close to that of the phase crossover frequency of the process. By making measurements of the frequency and amplitude of the limit cycle, the data required by the Ziegler and Nichols Ultimate Period method is obtained. The advantages of the Astrom and Hagglund method is that the experiment is carried out in closed loop and that a stable limit cycle is achieved for the majority of processes met in process industries.

From the literature on PID controller design methods, a way to relate the different PID controller design methods is to group them by the requirement to have either explicit, data driven or implicit models of the process available to the control system designer. In the subsequent sections of this chapter the process model identification methods shown in Figure 1.1 shall be discussed.

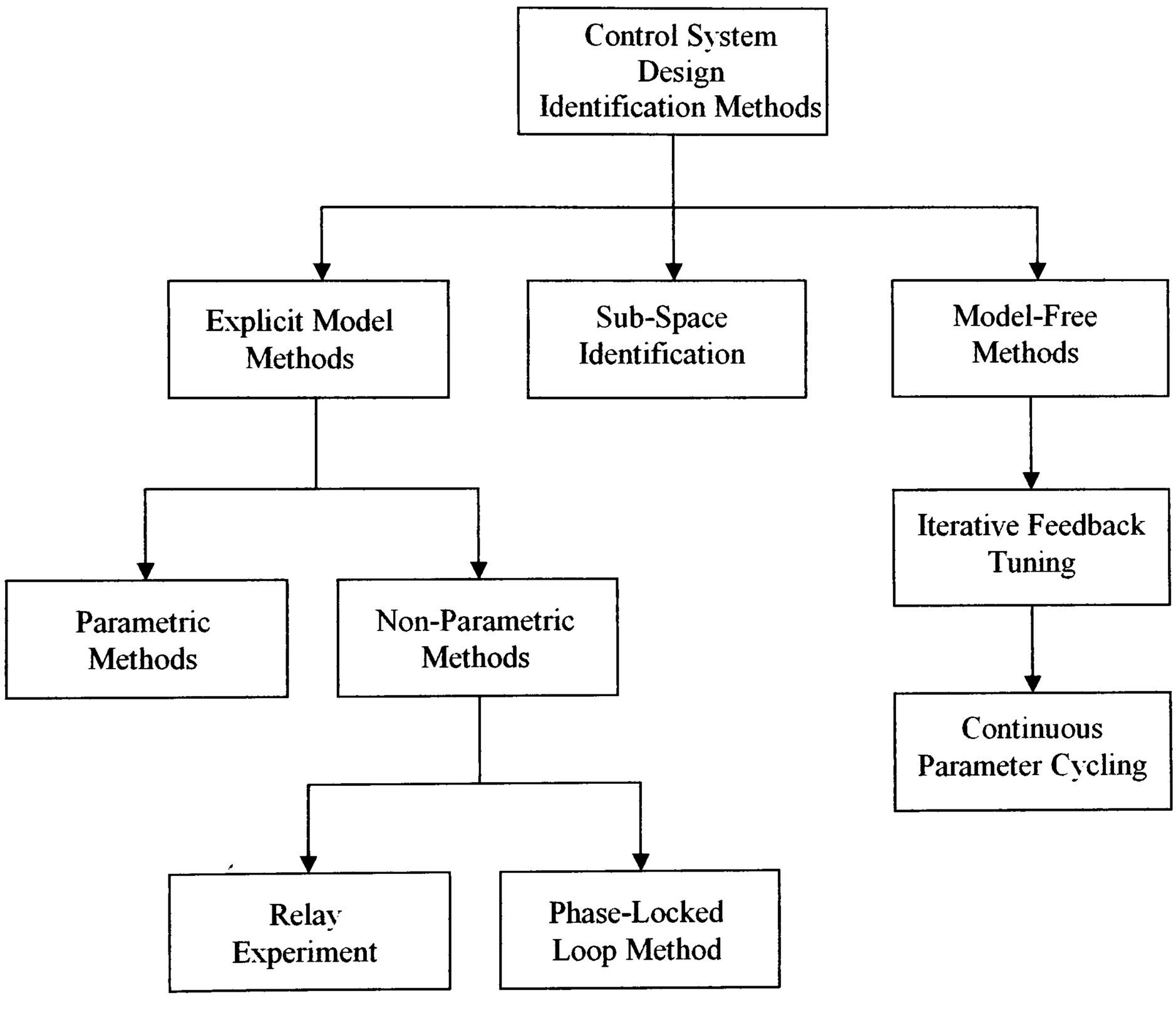


Figure 1.1: Closed Loop Identification Methods

The classification of the identification methods shown in Figure 1.1 begins by distinguishing between methods that are used to provide an explicit process model, either in parametric or non-parametric form, and methods that utilise either a data driven model or an implicit model of the process. Figure 1.1 serves to illustrate how the process modelling and controller synthesis methods discussed in the following, derive and utilise process identification methods ranging from explicit to sub-space or data driven to model free techniques.

In the following, under the heading of explicit models, only non-parametric single and multiple point methods of closed loop identification shall be discussed. The reason for this is that parametric system identification in closed loop is more difficult to implement in practice since not only does the control signal to the process require to be persistently exciting but the reference signal must also be persistently exciting to obtain reasonable estimates of the model parameters (Ljung, 1987; Soderstrom and Stoica, 1989). From Figure 1.1 it can be seen that the non-parametric identification methods that shall be discussed are the relay method and the phase-locked loop method. The relay method is an inherently closed loop method and is discussed in the following section. The phase-locked loop method of system identification shall be discussed in section 1.3 along with the developments made to the method.

In recent years there has been a great deal of research carried out in the area of model-free controller design methods. In the model-free controller design field two methods have been reported. In the first method reported the controller parameters are tuned such that a control performance cost function is minimised. By using response data collected from the closed loop process and re-injecting it back into the system, it is possible to provide estimates of the gradient of the cost function with respect to the controller parameters and hence, by the use of Netwon-like algorithms, the control performance cost function is minimised. This method of controller tuning was first reported by Hjalmarsson *et al* (1994, 1998) where the term Iterative Feedback Tuning (IFT) was used to describe the method. The IFT method is discussed in section 1.5 of this chapter. A known problem with Iterative Feedback Tuning is that it does not provide a means of determining an estimate of the Hessian of the gradient of the control performance cost function. The method of Continuous

Parameter Cycling, reported in Chapter 5 of this thesis, shall be shown to allow the estimation of both the gradient and the Hessian of the control performance cost function and hence allow the use of improved (over the IFT method) numerical routines to provide optimal control solutions.

The second method of so called *model-free* controller tuning reported in the literature uses subspace identification methods (Favoreel *et al*, 1998). The subspace matrices obtained from the process input and output data can be used to design controllers (Favoreel *et al*, 1999; Woodley *et al*, 2001; Kadali *et al*, 2003) without the intermediate step of explicitly identifying a process model, thus giving rise to the term *model-free approach* being used and adopted for subspace controller design methods. The subspace identification method is discussed briefly in section 1.4 of this chapter and serves as a link between explicit model methods and *model-free* approaches.

1.2 The Relay Method of Non-Parametric System Identification.

The connection of a relay in closed loop with a class of systems that can be found in process industries shall, in general, result in the generation of a limit cycle with a frequency that is close to the phase crossover frequency of the system. The configuration of the relay experiment is shown in Figure 1.2.

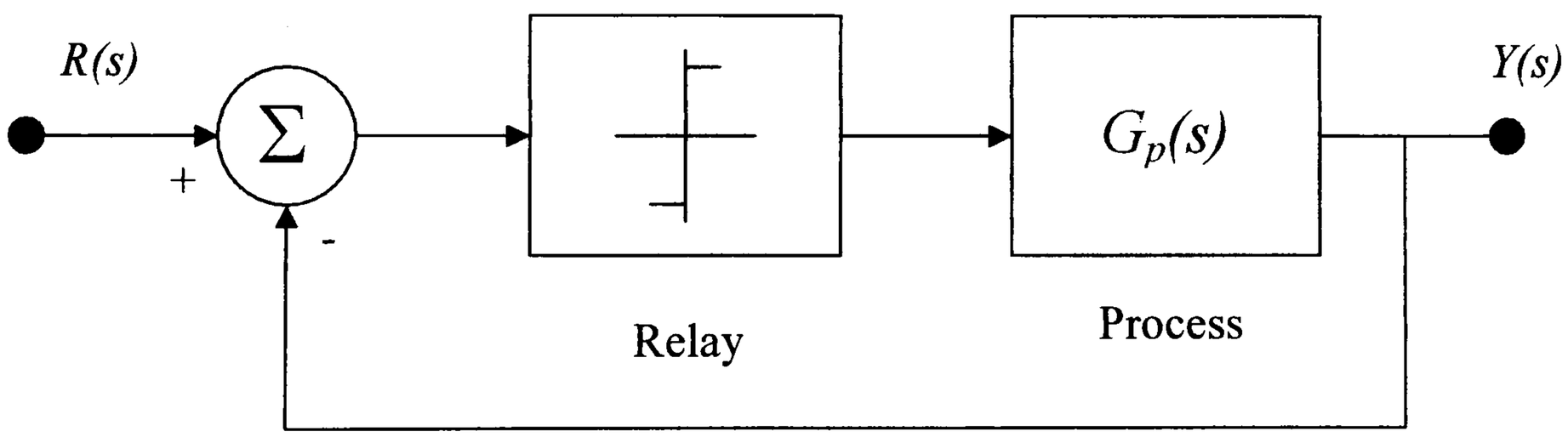


Figure 1.2: Relay Experiment Setup.

A relay exhibits a non-linear behaviour and as such the analysis of a system within which a relay is used is relatively difficult to carry out. It is usual in the case of a relay system to utilise the describing function method (Atherton, 1975) to derive

a linear system representation of the non-linear relay. With such a description of the relay available, linear methods can then be employed to carry out the analysis of the system. The describing function method is based on the assumption that there is only one sinusoidal component present in the system. Thus if the closed loop system exhibits a low pass characteristic at the phase crossover frequency then the describing function method shall return accurate estimates of the phase crossover frequency and the process magnitude at that frequency (Shen *et al*, 1996a). However if the system does not have a low pass characteristic at the phase crossover frequency then substantial odd harmonic components can circulate in the closed loop and thus degrade the accuracy of the estimates obtained from the use of the describing function method. To reduce the harmonic content of the relay output a relay with the following characteristic was proposed (Shen *et al*, 1996a):

$$y(t)=\begin{cases} +h & : \quad x(t)\geq \dfrac{h}{k} \\ kx(t) & : \quad -\dfrac{h}{k}<x(t)<\dfrac{h}{k} \\ -h & : \quad x(t)\leq -\dfrac{h}{k} \end{cases} \tag{1.1}$$

where $y(t)$ is the relay output, $x(t)$ is the relay input, h is the relay saturation limit and k is the gain of the linear section of the relay characteristic. It can be shown (Shen *et al*, 1996a) that if the gain, k, is chosen as

$$k=\frac{h}{A} \tag{1.2}$$

where A is the amplitude of the exciting sinusoid, there will be no harmonics other than the fundamental, present in the closed loop. However, there are a number of difficulties associated with the practical application of the Shen *et al* method. The choice of the gain k is system dependent and thus its value cannot be determined *a priori*. A further difficulty is that the value of k must be set slightly larger than the theoretical value given by equation (1.2) if a stable limit cycle is to be established.

A similar proposal to that of Shen *et al*, viz. that there are no harmonics present in the relay output other than the fundamental, is given by Lee *et al* (1995). The method of Lee *et al* differs from that of Shen *et al* in that a non-linear element is added to the output of the relay. The function of the non-linear element is to provide a sinusoidal excitation of the process of peak amplitude

$$|u(t)| = \frac{4h}{\pi} \tag{1.3}$$

where h is the relay height, and the frequency of the sinusoid is equal to that of the relay fundamental frequency. The practical implementation of the Lee *et al* method uses a standard relay experiment to determine the initial value for the excitation frequency and thereafter the non-linear function is used to extract the fundamental frequency of the relay oscillations. In order to reduce the harmonic content of the signals circulating in the closed loop to a minimum the switching of the frequency value is carried out at the zero point of the system excitation.

The relay experiment is a very simple and elegant means of identifying a process at its phase crossover frequency. However in practical applications of the method there is the possibility of a static load disturbance occurring during an experiment. The occurrence of a static load disturbance during a relay experiment causes the relay output to have an unequal mark-to-space ratio, assuming that the process is type zero. Hence, under a static load disturbance condition the identification of the process phase crossover data will have a large error.

Shen *et al* (1996b) observed that for a biased relay given by

$$y(t) = \begin{cases} h + \delta & : \quad x(t) > 0 \\ h - \delta & : \quad x(t) \leq 0 \end{cases} \tag{1.4}$$

where $x(t)$ is the relay input and h and δ are the relay height and bias respectively, when excited by a sinusoid plus bias signal given by

$$x(t) = A \sin \omega_0 t + \Delta a \tag{1.5}$$

gives rise to the same Fourier coefficients of the output waveform as a standard relay when it is excited by a sinusoid plus bias with the exception of a term δ. The term δ is present in the case of a biased relay excited by a sinusoid plus bias. Hence it is possible to utilise this term to restore equality to the mark-to-space ratio of the relay output by making a suitable choice of δ; Shen *et al* (1996b) give this value as

$$\delta = -h \frac{\Delta a}{A} \tag{1.6}$$

where h is the relay height, A is the peak amplitude of the system output and Δa is the value of the static load disturbance term. A difficulty in the practical implementation of this method lies in the fact that the load disturbance must be

detected during the identification so that its effect can be removed. A possible means of detecting the effect of a load disturbance would be to monitor the mark-to-space ratio of the relay output and to apply the bias, given by equation (1.6) to the standard relay, when an unequal mark-to-space ratio was detected.

A further problem arises in the practical application of the relay method of system identification when noise is present in the output of the system being identified. The presence of noise causes the relay to switch spuriously, thus giving rise to inaccuracies in the estimate of the phase crossover point data. Astrom and Hagglund (1995) proposed the use of a relay with hysteresis to reduce the effects of noise on the switching of the relay. The value of the relay hysteresis is set such that it is greater than the peak value of the noise signal present in the system output. The describing function of a relay with hysteresis has both a real and imaginary component and thus no longer has zero as an argument, hence there will be a systematic error present in the estimation of the system phase crossover data. An obvious means of reducing this error is to make the hysteresis value as small as possible.

The relay experiment of Astrom and Hagglund (1984) can be used to determine an estimate of the phase crossover data for a process. It is also possible to connect filters in cascade with the relay such that points other than the phase crossover point can be identified (Astrom and Hagglund, 1995). In particular if an integrator is used as the filter then a limit cycle at a frequency for which the phase shift of the process is $-\frac{\pi}{2}$(rad) will result. More generally the relay with hysteresis can be used to identify points on the frequency response curve of a process other than the phase crossover point. Loh *et al* (2001) detail a method for processes that can be adequately modelled as

$$G_p(s) = \frac{Ke^{-Ls}}{s(1+sT)} \tag{1.7}$$

whereby, an iterative method is used to determine points on the frequency response curve of the process that have either a desired magnitude or phase angle. If a point with a certain magnitude is required to be identified then the iterative method specifies how the relay height should be changed to achieve that identification.

Similarly if a point with a certain phase angle is to be identified the method details the required changes to the relay hysteresis value.

One of the benefits of the relay experiment is that a known point on the frequency response curve of a process can be identified. In the literature there are a number of extensions to the relay experiment that allow the identification of multiple points on the frequency response curve of a process. To identify multiple points of the frequency response of a process Bi *et al* (1997) utilise a parasitic relay that is locked to a multiple of the standard relay period. The parasitic relay experiment setup is shown in Figure 1.3 where the parasitic relay characteristic is described by

$$u_2(k)=\begin{cases} u_2(0)=\alpha h & \\ u_2(k)=-\alpha h\times sign(u_2(k-1)) & \text{if} \quad u_1(k-1)>0 \\ & \text{and} \quad u_1(k)<0 \\ u_2(k)=u_2(k-1) & \text{otherwise} \end{cases}$$

where α is a multiplier for the standard relay height h. The relay height of the parasitic relay must be chosen carefully so that the limit cycle is not moved too far from the phase crossover point frequency, 0.1 to 0.3 is the stated range for α.

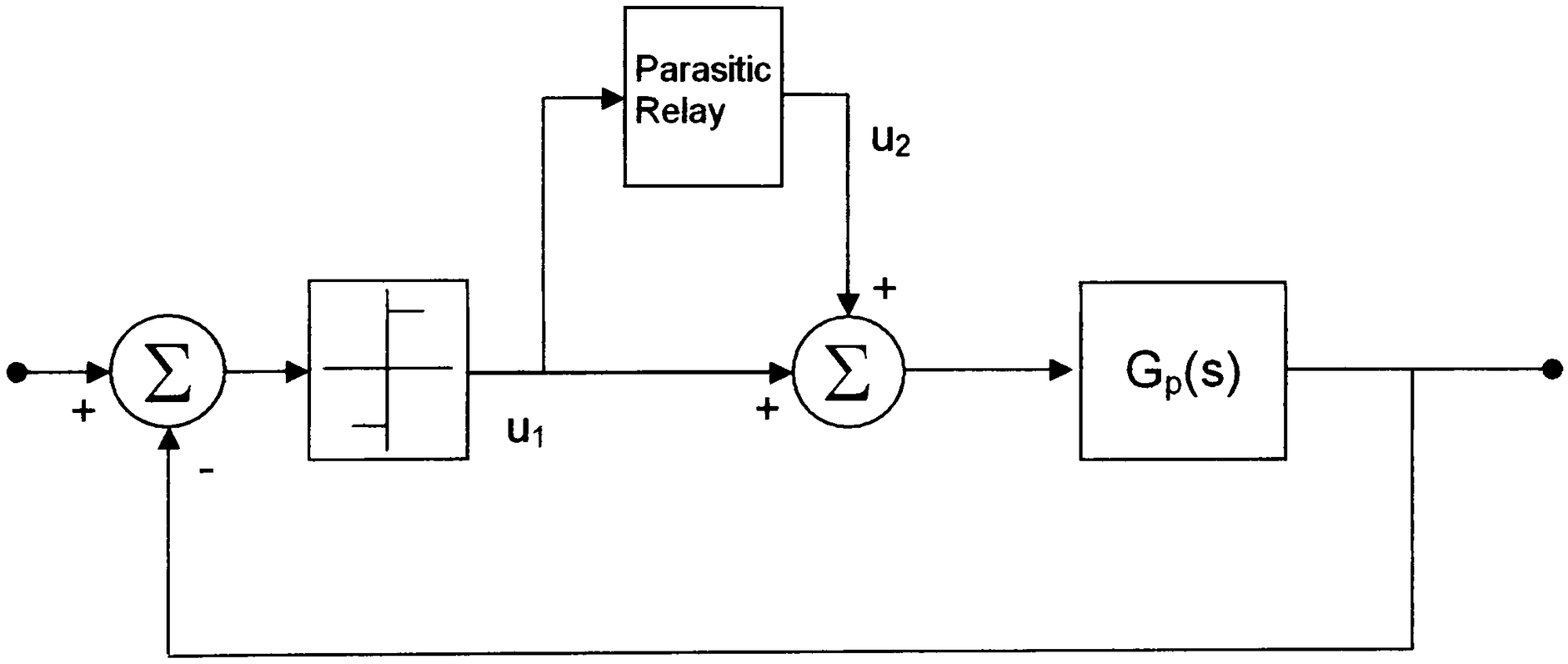


Figure 1.3: Parasitic Relay Experiment Setup.

It is also stated that it is possible to connect more than one parasitic relay with the standard relay without introducing too great an error in the frequency of oscillation to determine the phase crossover point. With this configuration it is possible to obtain an excitation of the process at the phase crossover point frequency and sub-multiples of that frequency. Each excitation frequency also generates odd harmonics of its own

frequency; hence the process can be excited at frequencies both above and below the phase crossover frequency. Since the process is excited by the test frequencies simultaneously, Fourier or Spectral analysis methods must be employed to extract the frequency response estimates of the process.

The method of Bi *et al* (1997) requires the addition of an extra hardware component to facilitate the generation of harmonics that allow the identification of points other than the phase crossover frequency of a process. Wang *et al* (1997a) describe a method whereby the data from a standard relay experiment can be used to identify multiple points of the frequency response of a system. The method begins by recording the process excitation and output signals until the closed loop is in steady state. The data are then multiplied by an exponential term of the form

$$f(t) = e^{-\alpha t}$$

this ensures that the data tend to zero as $t \to \infty$; a necessary step for the next stage of the method. The process input and output data are then Fourier transformed and an estimate of the shifted frequency response of the process is given by

$$\hat{G}(j\omega + \alpha) = \frac{\hat{Y}(j\omega + \alpha)}{\hat{U}(j\omega + \alpha)}$$

where $\hat{Y}(j\omega + \alpha)$ and $\hat{U}(j\omega + \alpha)$ are estimates of the shifted Fourier transforms of the process output and input signals respectively. It is possible to recover a non-shifted version of the estimate of the frequency response of the process by first inverse Fourier transforming the shifted estimate, $\hat{G}(j\omega + \alpha)$, such that the impulse response

$$\tilde{g}(t) = \hat{g}(t)e^{-\alpha t}$$

is obtained, thus the frequency response estimate can be recovered from the following

$$\hat{G}(j\omega) = \int_{-\infty}^{\infty} \left(\tilde{g}(t)e^{\alpha t} \right) e^{-j\omega t} dt$$

The methods of Bi *et al* and Wang *et al* provide estimates of multiple points on the frequency response curve of a process. However with these methods it is not possible to determine the phase angle or magnitude of these points *a priori*.

A method was proposed by Schei (1994) in which a relay and a filter element, in this case an integrator, were connected around a process that was operating in closed loop. With this method there is no requirement to change between controlling the loop by means of the existing controller and the relay as in the relay experiment of Astrom and Hagglund. The experimental setup for the method of Schei is shown in Figure 1.4.

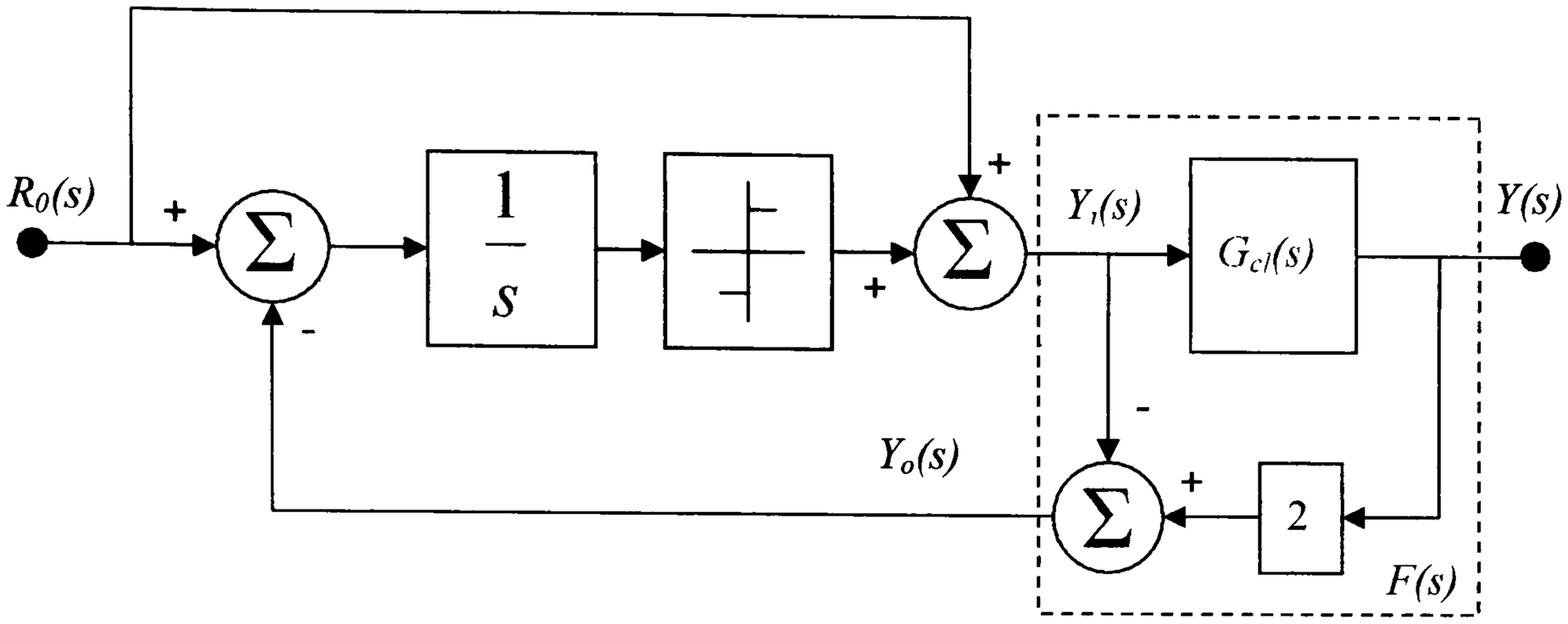


Figure 1.4: The Schei Relay Experiment, Monitoring of Phase Margin

The relay experiment of Schei allows the set point term $R_0(s)$ to be applied to the closed loop system, $G_{cl}(s)$, during testing such that there is no modification to the set point by the experiment setup. With this configuration Schei (1994) shows that it is possible to determine the phase margin of the forward transfer function of the closed loop system, $G_{cl}(s)$.

If the circuit of Figure 1.4 is reconfigured to be that shown in Figure 1.5 then it can be shown (Schei, 1994) that the gain margin of the forward transfer function of the system connected in closed loop, $G_{cl}(s)$, can be determined.

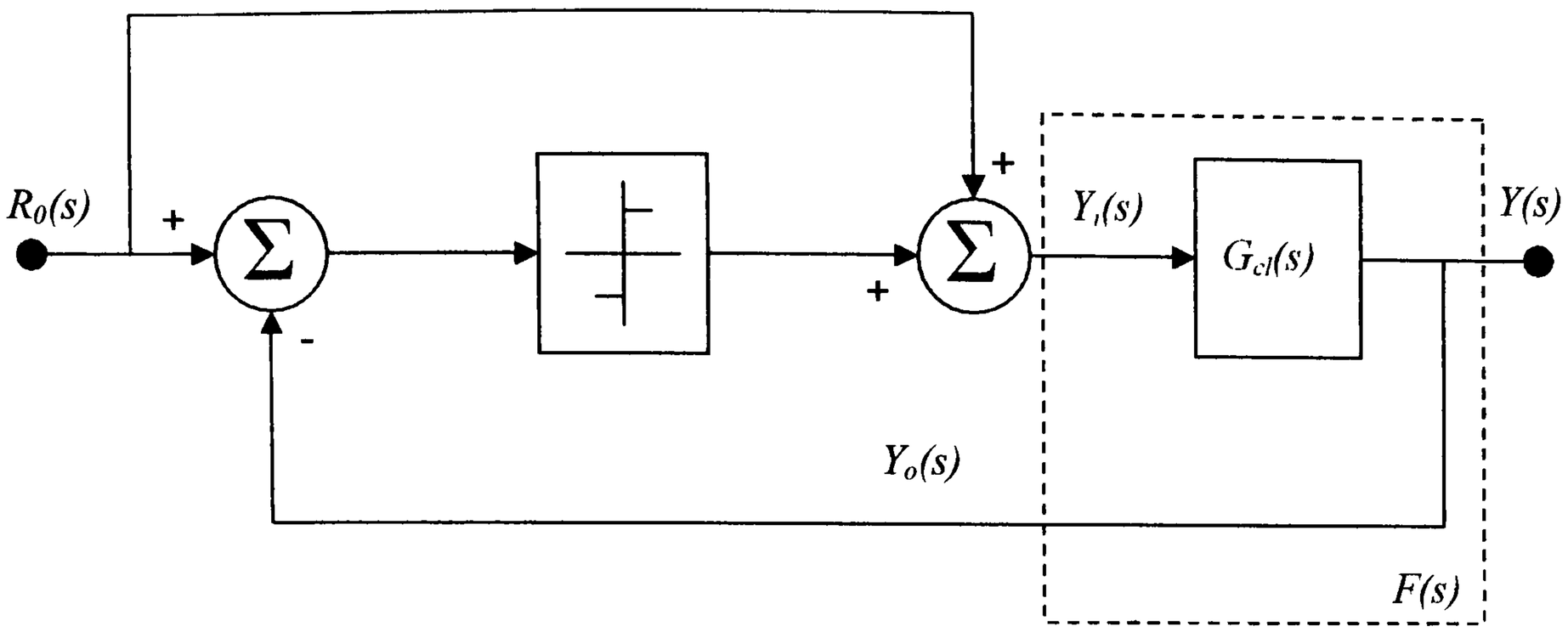


Figure 1.5: The Schei Relay Experiment, Monitoring of Gain Margin

The relay experiment due to Schei (1994) can thus be seen as a tool that allows the closed loop performance of a control loop to be relatively easily monitored. The methodology used by Schei has been extended by de Arruda and Barros (2003). In their method it is possible to identify either specified magnitude values of the forward path transfer function or specific sensitivity values for the system connected in closed loop, shown in Figure 1.6 as $G_{cl}(s)$.

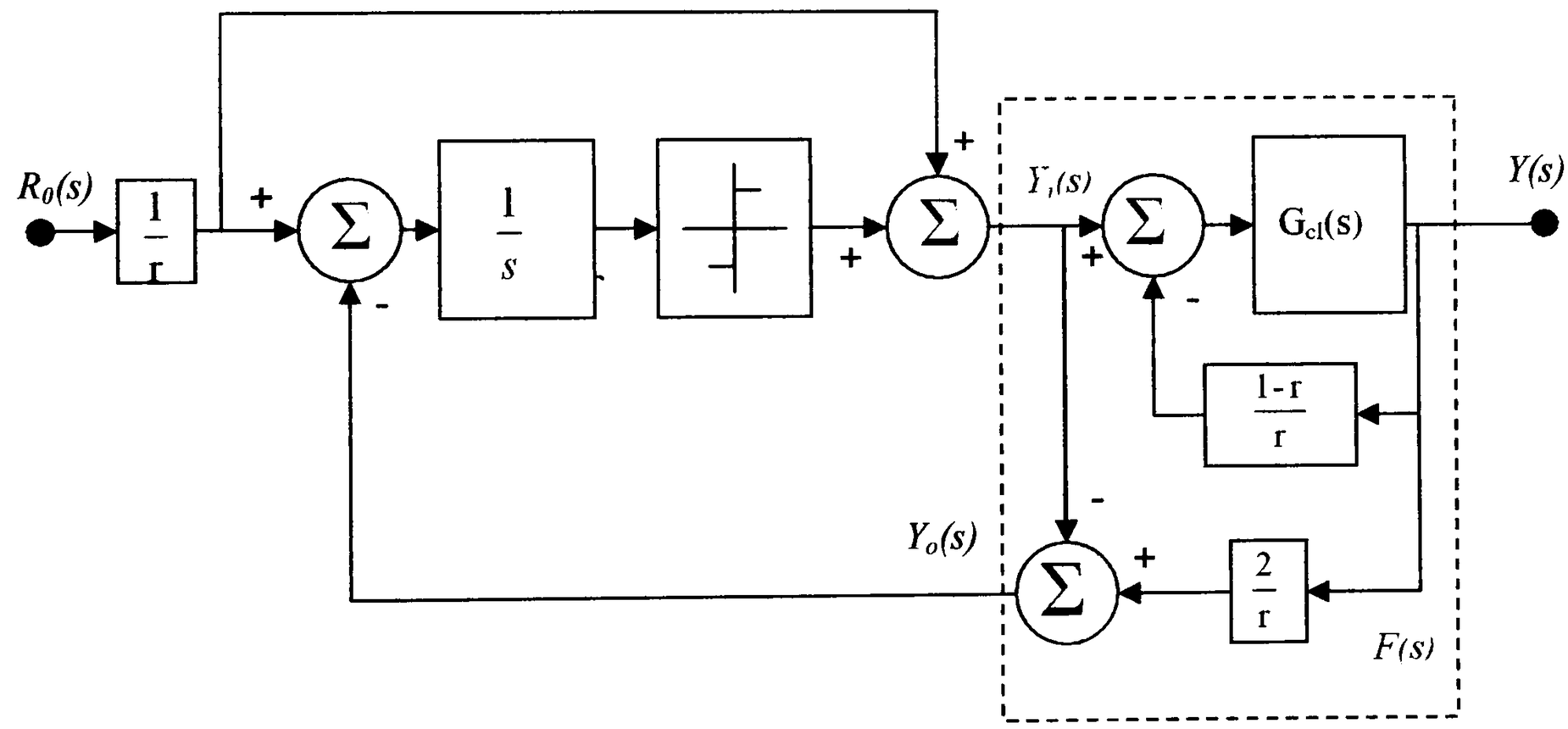


Figure 1.6: Relay Experiment to Identify Forward Loop Transfer Function

The positive constant term, r, shown in Figure 1.6 is used to select the point on the forward transfer function that is to be identified.

Figure 1.7 shows the experimental setup that is proposed by de Arruda and Barros (2003) to identify points on the sensitivity function of a system connected in closed loop.

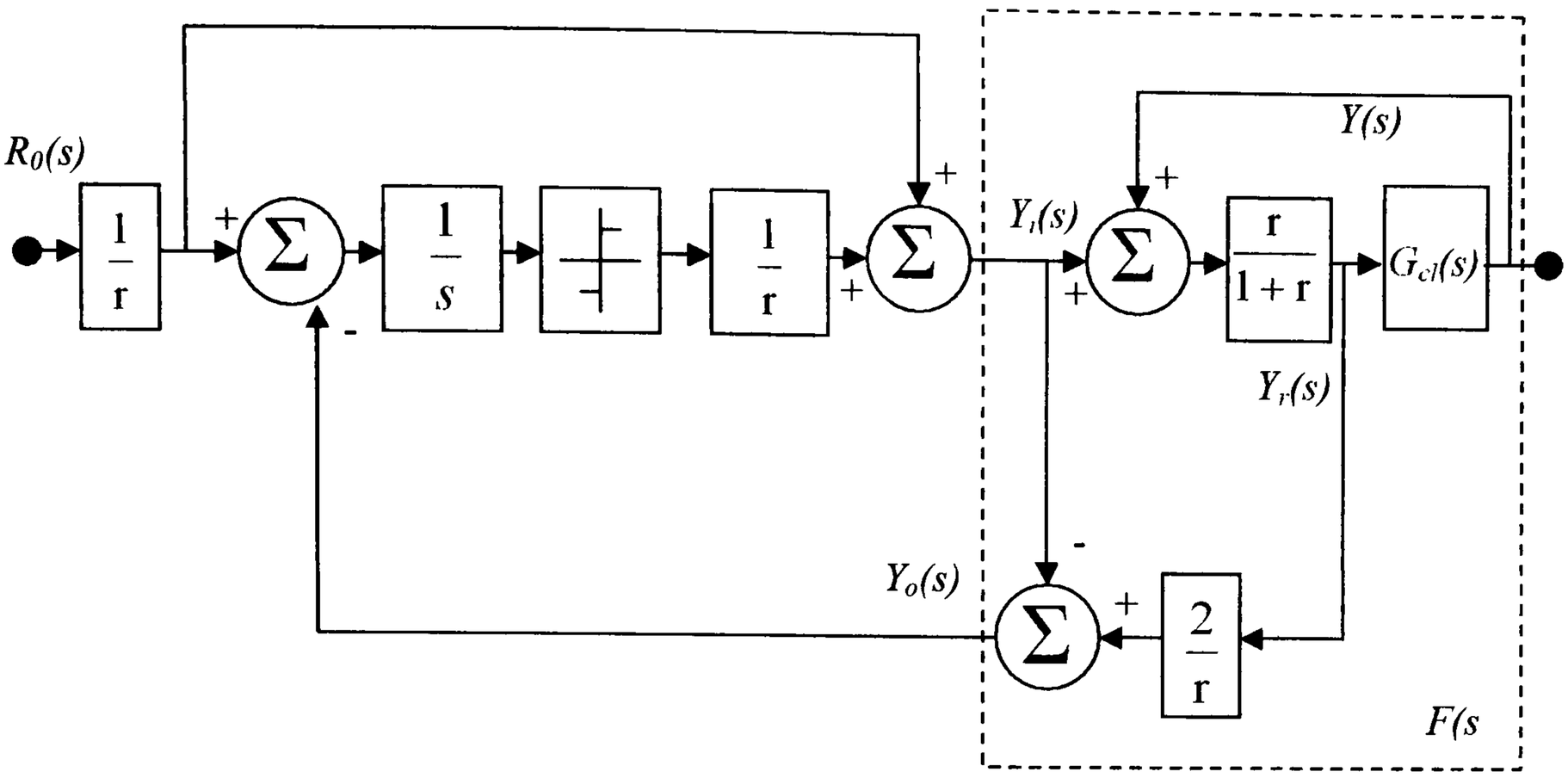


Figure 1.7: Relay Setup for Sensitivity Identification

From the above discussion, it can be seen that by the application of relays, data that are required for non-parametric identification of systems in closed loop can be obtained. The benefits that are obtained by the use of relays to obtain this data are that the time taken to carry out the required identifications tends to be low and that the experimental setup is relatively easy to implement, both of which are essential when this type of work is being carried out on an operational plant or process. There is however the problem caused by the use of the describing function approach for providing a linearised model of the non-linear relay characteristic affecting the accuracy of the results obtained. However, as discussed previously, there are a number of methods available to overcome this problem.

1.2.1 Summary Conclusions on the Relay Experiment.

The relay experiment has been a fertile source of research for a number of years, as evidenced by the large body of publications produced on the subject. Modifications to the relay characteristic such that the error due to the use of the describing function method of linearising the relay characteristic is reduced have

been reported (Shen *et al*, 1996a; Lee *et al*, 1995). Improvements in the accuracy of the relay experiment in the presence of measurement noise have been discussed (Astrom and Hagglund, 1995), as has the use of relays when process disturbances are present (Shen *et al*, 1996b). The use of relays has been extended from their application in single loop identification and tuning to the identification and tuning of cascade loop and multi-input multi-output systems (Hang *et al*, 1994; Shen and Yu 1994; Zhuang and Atherton, 1994; Palmor *et al*, 1995; Wang, 1997b). The relay experiment is a method that allows one point on the frequency response curve of a process to be identified. The extension of the relay experiment to the estimation of several frequency response points has been reported in the literature (Hagglund and Astrom, 1991; Bi *et al*, 1997; Wang *et al*, 1997a). The use of the relay experiment to determine the exact parameters of process models, of the type first and second order plus dead time, have appeared in the literature along with analytical methods to determine the output of a closed loop relay system for higher order systems (Kaya and Atherton, 2001; Panda and Yu, 2002).

From this the question arises, "What potential future development can be achieved with the relay experiment?" Current research on the relay experiment would appear to be focusing on using the shape of the closed loop system response (Panda and Yu, 2002) to determine low order model parameters. These models are then utilised in some form of auto-tuning method to derive the parameters for PID controllers. However the question of how accurate a representation of the frequency response of the model obtained is in comparison to the actual process frequency response is not answered. It would appear that research in the relay experiment has now moved from obtaining accurate non-parametric identification data to using the relay to obtain an accurate parametric model of the process.

While relay experiment research now appears to be moving into parametric system identification, it would seem that parametric system research is progressing to model-free or data driven approaches. The relay experiment will continue to be used in proprietary auto-tune controllers since the information it supplies is sufficiently accurate to allow a rule based design to provide a satisfactory level of control system performance. Research into the relay experiment can be broadly categorised as:

i) Modifications to the relay characteristic

ii) Operation of the relay experiment in noise or with process disturbances present

iii) Multiple point estimation on the process frequency response curve

iv) Cascade loop and multi-input, multi-output control system tuning methods.

From the above it can be seen that there have been many avenues of research followed and that the relay experiment is still an active area for research. However, the evidence of the review above is that the initial simplicity of the Astrom and Hagglund (1984) method is now being lost and that additional complexity is being added to provide further utility from the method (Schei, 1994; de Arruda and Barros, 2003). Further the published literature seems to show little evidence that the modified or more complex relay experiment methods are making a transition from theory to serious industrial application.

1.3 The Phase-Locked Loop Method of Non-parametric System Identification.

The relay experiment of Astrom and Hagglund (1984) is an elegant and fascinatingly simple method of identifying key system parameters on which to base PID tuning rules or to provide data for control loop monitoring purposes. Research into methods of overcoming the known operational problems of the relay experiment in the presence of noise and improving the accuracy of the identified data was discussed in the preceding section. The salient features of the relay experiment are that:

i) It is easy to perform

ii) It is carried out in closed loop

iii) It returns relatively accurate results, and

iv) Identification time is not protracted

It was the intention of the initial Phase-Locked Loop method of system identification research to provide an identification method with all of the benefits of the relay experiment but also to provide the added flexibility of being able to identify any point on the frequency response curve of a system. The initial Phase-Locked

Loop (PLL) method of system identification (Crowe and Johnson, 1998; Johnson and Crowe, 1998; Crowe, 1998) represents a clear departure from the relay experiment.

1.3.1 Phase-Locked Loop Identifier: Fundamental Theory and Properties.

The research into the Phase-Locked Loop method of system identification began in 1996. The results of the research were patented in 1998 (UK patent application number 9802358.3) with the first publication on the method following shortly there after (Crowe and Johnson, 1998; Johnson and Crowe, 1998). The initial spur for the development of the Phase-Locked Loop method of system identification was to answer the question "How could the relay experiment be carried out without using a relay?" The PLL is shown in block diagram form in Figure 1.8. The main components of the identification module, as shown in Figure 1.8, have the following functions:

i) A feedback structure using a phase or gain reference at an input comparator.

ii) A digital model of a Voltage Controlled Oscillator (VCO) that generates a process sinusoidal excitation path and a sinusoidal reference path.

iii) A digital signal processing unit or Phase Sensitive Detector to extract the actual measured system phase or gain for supply to the comparator.

iv) A digital integrator unit to ensure the identifier unit converges to the given system phase or gain reference.

The digital process identifier, the signal processor block of Figure 1.8, comprises two conceptual processes. The inner process is that of a sine wave experiment. The outer process is a digital control loop containing two sub-processes:

i) The extraction of phase and gain data from the output of the multiplier, and

ii) The update process and the convergence of the overall digital control loop.

This outer process or loop can be considered as a closed loop stability problem and is discussed in the following.

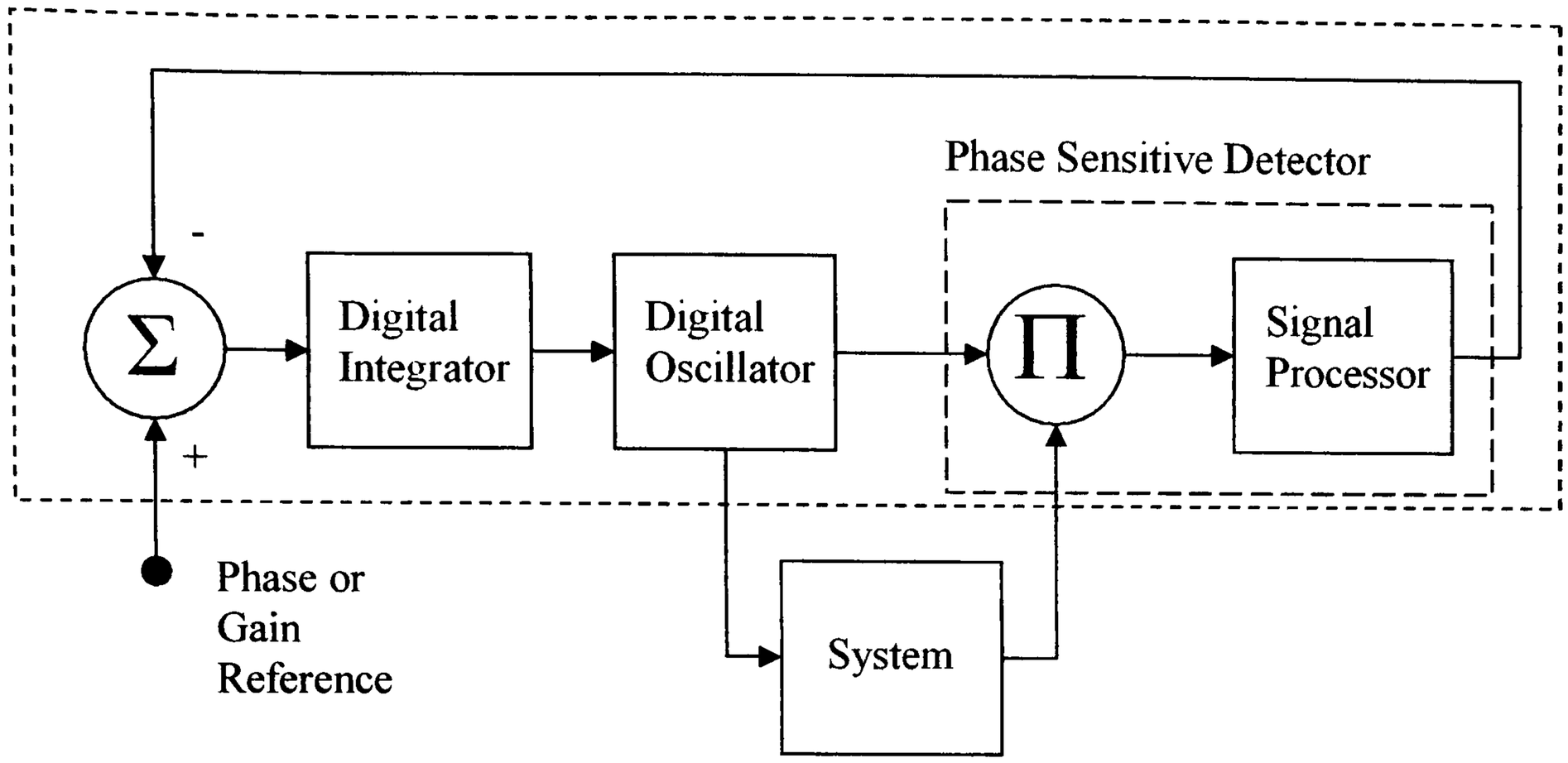


Figure 1.8: Phase-Locked Loop Method Conceptual Diagram

The inner sine wave experiment

The outer digital loop acts as a supervisor for the inner loop comprising the sine wave experiment. The outer loop will update only when the phase shift (or process gain) estimate has steadied and is giving a consistent sequence of values. The speed at which these values settle will depend on the transient characteristics of the particular system. A second aspect of the inner sine wave experiment is that it is not practical to allow the k^{th} experiment to go to completion. This is accommodated in the analysis by defining:

$$\phi_k = \lim_{t \to \infty} \phi_{kt} \tag{1.8}$$

where ϕ_{kt} is the estimate of ϕ_k at time, t. It is usual to truncate the estimation process at some $\bar{t} < \infty$ and this estimate is denoted, $\phi_{k\bar{t}}$.

Phase and gain data extraction in the digital identifier module

The duty of extracting phase (or gain) information from the sine wave experiment is performed by the Phase Sensitive Detector block of Figure 1.8 and this is a simple peaks and troughs operation. Whilst this enables the extraction of phase and gain data to be accomplished efficiently and accurately the question of loop convergence needs to be answered.

Identifier Convergence Theory

If the inner process is considered to contribute to the outer digital process on completion of the sine wave experiment, the outer loop can then be given the discrete system representation of Figure 1.9. The integrator takes the common z-domain form shown in the Figure 1.9. The digital oscillator is modelled as a simple digital gain block with gain K_{VCO}. The analysis of closed loop convergence is a quasi-steady state analysis. For this reason Figure 1.9 does not include the sine wave test loop. It is assumed that the sequence of converged outputs of the Φ_{DIG} block can be obtained.

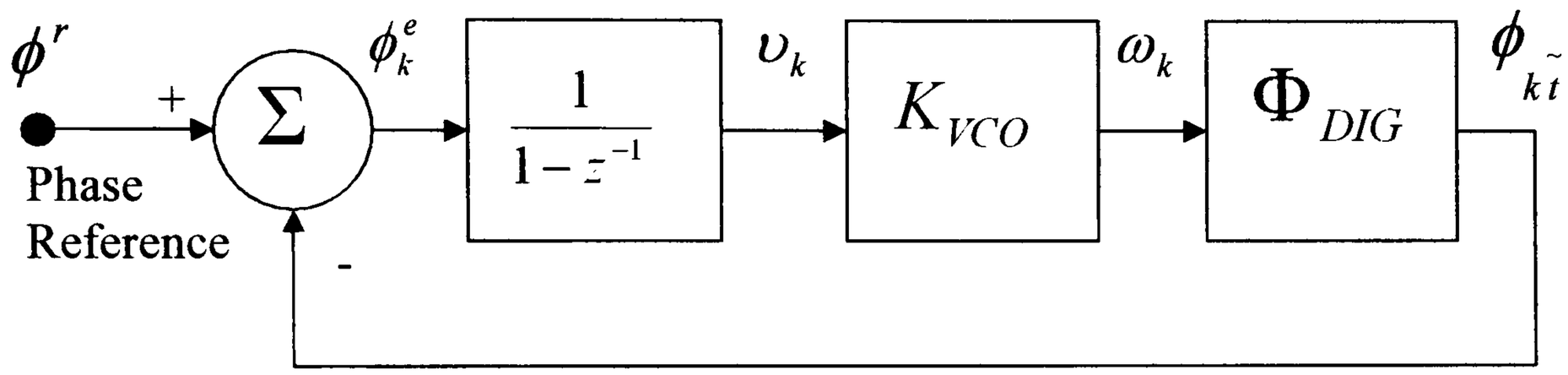


Figure 1.9: Outer Loop Digital Identifier System

Lemma 1 Fixed Point Lemma

If ω_* is a fixed point of the identification scheme, then

$$\phi(\omega_*) = \phi^r \qquad \bullet$$

Proof

As Crowe (1998).

The importance of the result is threefold:

i) If the routine converges then it converges to the reference phase value.

ii) Whether $\omega_* = \omega_r$ (for which $\phi^r = \phi(\omega_r)$) depends on the properties of the system but this result can be engineered by careful algorithm construction.

iii) An analogous result and interpretation for a system gain reference is straightforward.

For a fixed point frequency, the algorithmic relationship of the phase-locked loop identifier (Figure 1.9) can be used to give:

$$(\omega_* - \omega_{k+1}) = (\omega_* - \omega_k) + K_{VCO}\eta_{k\tilde{i}}(\phi^r - \phi_k) \tag{1.9}$$

where

$$\eta_{k\tilde{i}} = \left[1 - \left(\frac{\phi_{k\tilde{i}} - \phi_k}{\phi^r - \phi_k}\right)\right]$$

Theorem 1.1: Sufficient Conditions for Convergence

The iterative process (1.9) satisfies

$$\left|(\omega_* - \omega_{k+1})\right| \leq \left[\prod_{j=0}^{k} L_j\right]\left|(\omega_* - \omega_0)\right|$$

where

$$L_j = \left|1 - K_{VCO}\eta_{k\tilde{i}}\frac{d\phi(\omega_{j^*})}{d\omega}\right|$$

and

$$\omega_j \leq \omega_{j^*} \leq \omega_*$$

i) If there exists $\bar{k}$ such that for all $k > \bar{k}$, $L_k < 1$ then $|\omega_* - \omega_k| \to 0$ as $k \to \infty$

ii) If $L_{\max} = \max\{L_l\}_{j=0}^{\infty} < 1$, then $|\omega_* - \omega_k| \to 0$ as $k \to \infty$

From clauses i) and ii) it follows for L_j that

$$\left|1 - K_{VCO}\eta_{k\tilde{i}}\frac{d\phi(\omega_{j^*})}{d\omega}\right| < 1$$

Proof of Theorem 1.1

For a proof of Theorem 1.1 refer to Crowe (1998).

Remarks

Clause i) allows the Phase-Locked Loop identifier to *wander* before converging to the fixed point solution. Clause ii) is a special case of Clause i). In the case where

$\tilde{t} \to \infty$ then $\phi_{k\tilde{t}} = \phi_k$ and $\eta_{k\tilde{t}} = 1$. The key to the speed of convergence lies in the appropriate selection of $\tilde{t}$ (the cut-off point for the accuracy within Φ_{DIG}) and the selection of K_{VCO}. From the convergence analysis the bounds over which the digital controlled oscillator gain may be varied and convergence maintained are obtained from the following. Theorem 1.1 gives

$$\left| 1 - K_{VCO} \eta_{k\tilde{t}} \frac{d\phi(\omega_{j^*})}{d\omega} \right| < 1$$

hence

$$-1 < 1 - K_{VCO} \eta_{k\tilde{t}} \frac{d\phi(\omega_{j^*})}{d\omega} < 1$$

and

$$-2 < -K_{VCO} \eta_{k\tilde{t}} \frac{d\phi(\omega_{j^*})}{d\omega} < 0$$

$$0 < K_{VCO} \eta_{k\tilde{t}} \frac{d\phi(\omega_{j^*})}{d\omega} < 2$$

thus

$$0 < K_{VCO} < \frac{2}{\eta_{k\tilde{t}} \frac{d\phi(\omega_{j^*})}{d\omega}}$$

The Phase-Locked Loop method of system identification has been shown to provide a high degree of identification accuracy (Johnson and Crowe, 1998; Crowe, 1998) and is able to give more accurate results than the relay experiment of Astrom and Hagglund. However, in all of the simulation examples carried out, the relay experiment takes less time to achieve the identification albeit to a lower degree of accuracy. The main advantage that the Phase-Locked Loop method has over the relay experiment is that there is no restriction as to the point on the frequency response curve that is to be identified. The user can specify any phase angle or any magnitude value as a reference input to the identifier module. A further advantage of the Phase-Locked Loop method is that the intermediate points that are identified, as the error between the reference value and the actual value reduces, are accurate and can be used as estimates of the frequency response of the system being identified.

In the literature there have been accounts of other variants of the Phase-Locked Loop technique (Balestrino *et al*, 2000; Clarke and Park, 2003). In Balestrino *et al* the term Sinusoidal AutoTune Variation (SATV) is used to describe a Phase-

Locked Loop method. Comparing Figure 1.8 with Figure 1.10 it can be seen that structurally there is no difference between the Phase-Locked Loop method and the method described in Balestrino *et al* as SATV. Balestrino *et al* initialise the sinusoidal oscillator at a frequency close to the phase crossover frequency of the process being identified by the initial use of a relay experiment. However for the identification of points other than the phase crossover point no initialisation method is cited and it is assumed that there is no means to initialise the method for points other than the phase crossover point. In the Balestrino *et al* method the extraction of phase information is carried out using synchronous detection; a technique known to give good noise rejection properties (Soderstrom and Stoica, 1989).

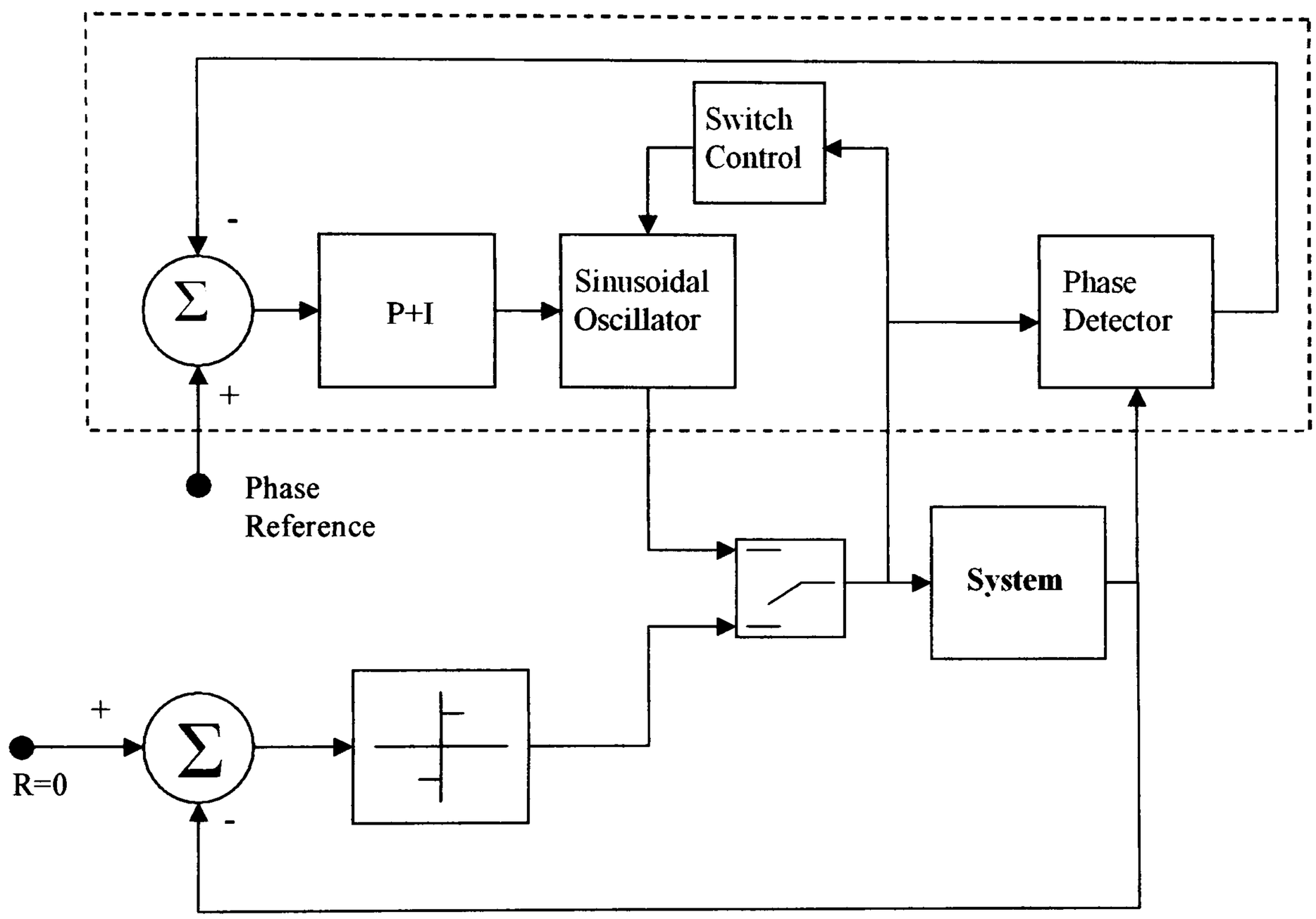


Figure 1.10: SATV Implementation Diagram (Balestrino *et al*, 2000).

Figure 1.11 shows the block diagram of a synchronous detector. As can be seen from the figure, the synchronous detector recovers the system phase angle and gain estimates by multiplying the system response by sine and cosine signals and then low pass filtering the resultant product terms.

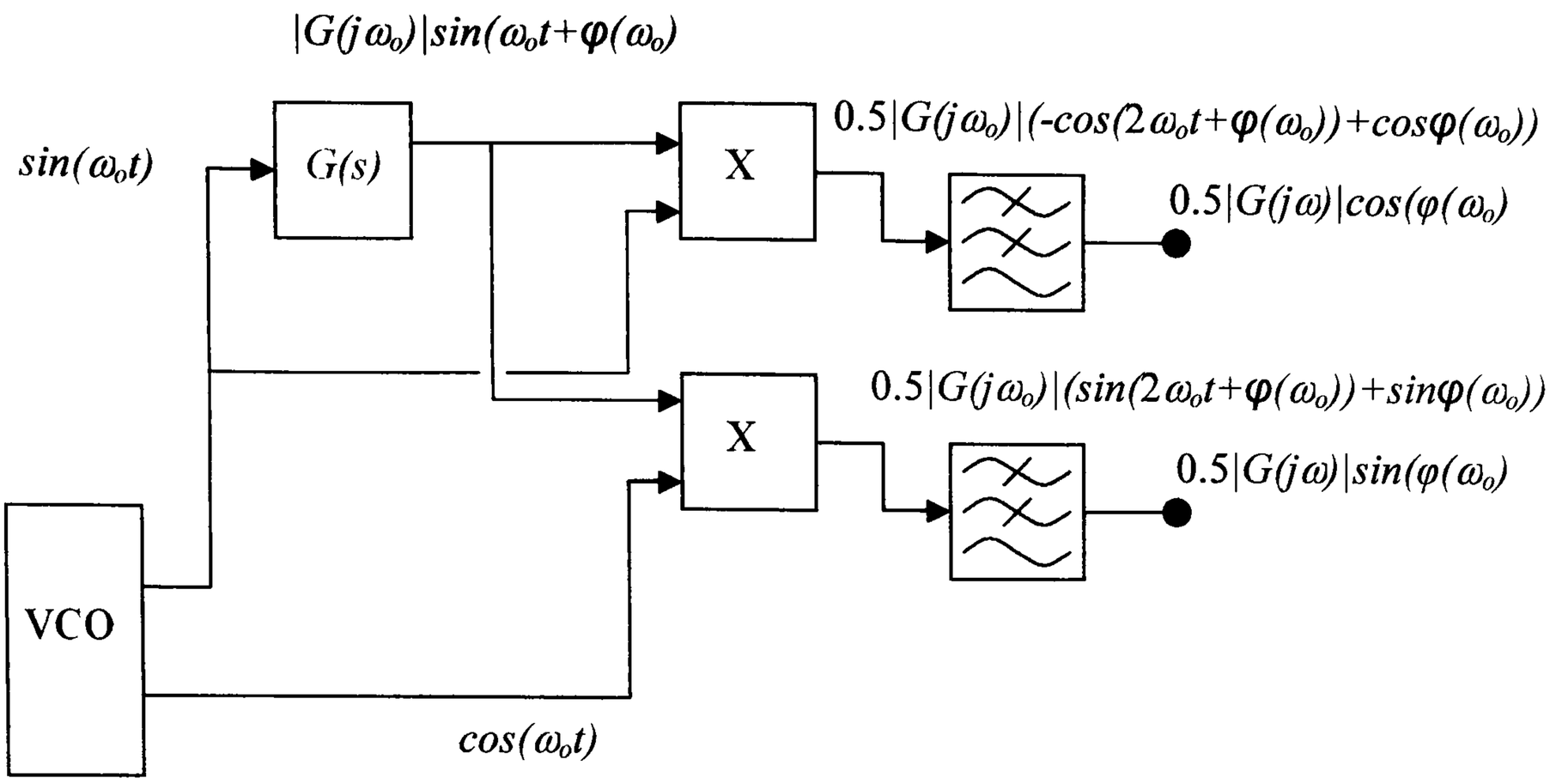


Figure 1.11: Block Diagram of a Synchronous Detector.

Clarke and Park (2003) use the term "phase/frequency estimator" for the system shown in Figure 1.12.

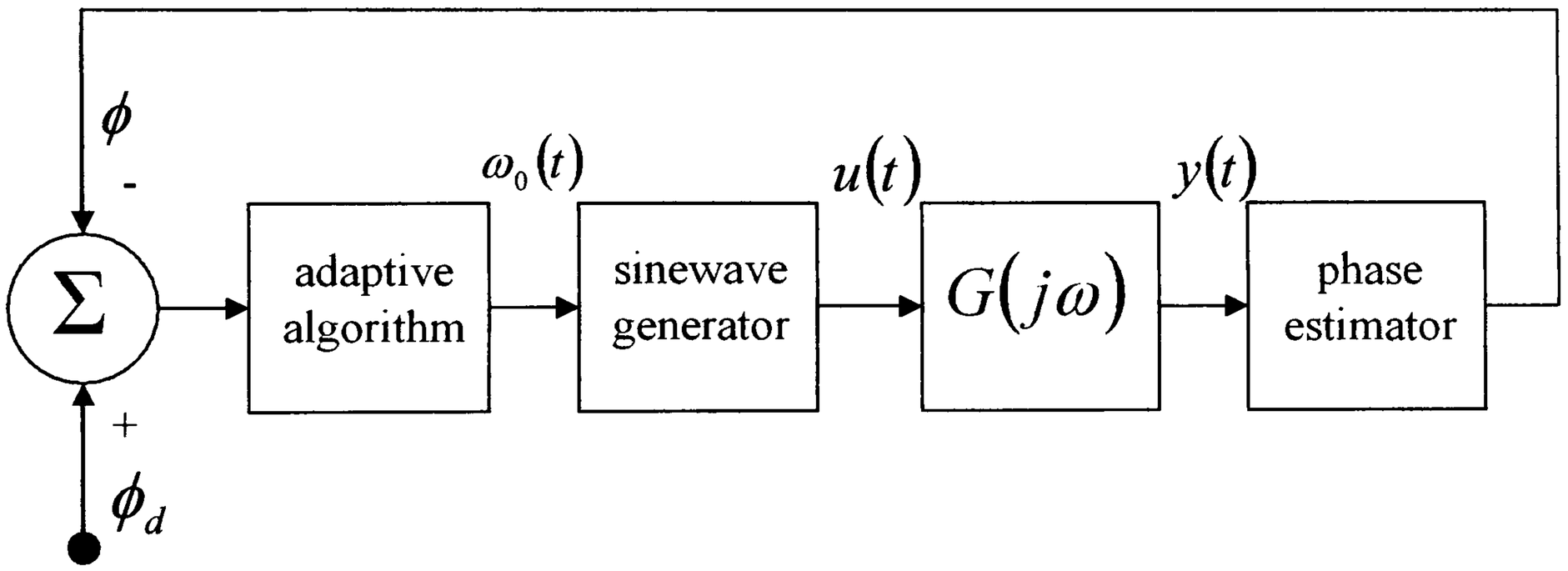


Figure 1.12: Conceptual diagram of a phase/frequency estimator (Clarke and Park, 2003).

The adaptive algorithm block in Figure 1.12 implements

$$\frac{d\omega_0(t)}{dt} = -Ke(t) = K\left(Arg(G(j\omega_0) - \phi_d\right)$$

and is thus seen to be an integrator with gain *K*. If the system shown in Figure 1.12 is compared with that in Figure 1.8 then no structural difference can be identified, so far as the functions of the individual blocks are concerned. However it should be borne in mind that the Phase-Locked Loop identifier of Crowe and Johnson operates

in discrete time whereas that of Clarke and Park operates in continuous time. In the Clarke and Park method, phase detection is carried out by the use of a Hilbert transform. A Hilbert transform is a filter that has unity gain for all frequencies and a phase shift of $-\frac{\pi}{2}$(rad) at all positive frequencies and a phase shift of $\frac{\pi}{2}$(rad) at all negative frequencies. The data extraction method using the Hilbert transform is explained by means of the following. Assume that a process is excited by a signal

$$S_i(t) = \sin(\omega_o t)$$

If the process is given by *G(s)*, then the resultant steady state output will be

$$S_o(t) = |G(j\omega_o)| \sin(\omega_o t + \phi(\omega_o))$$

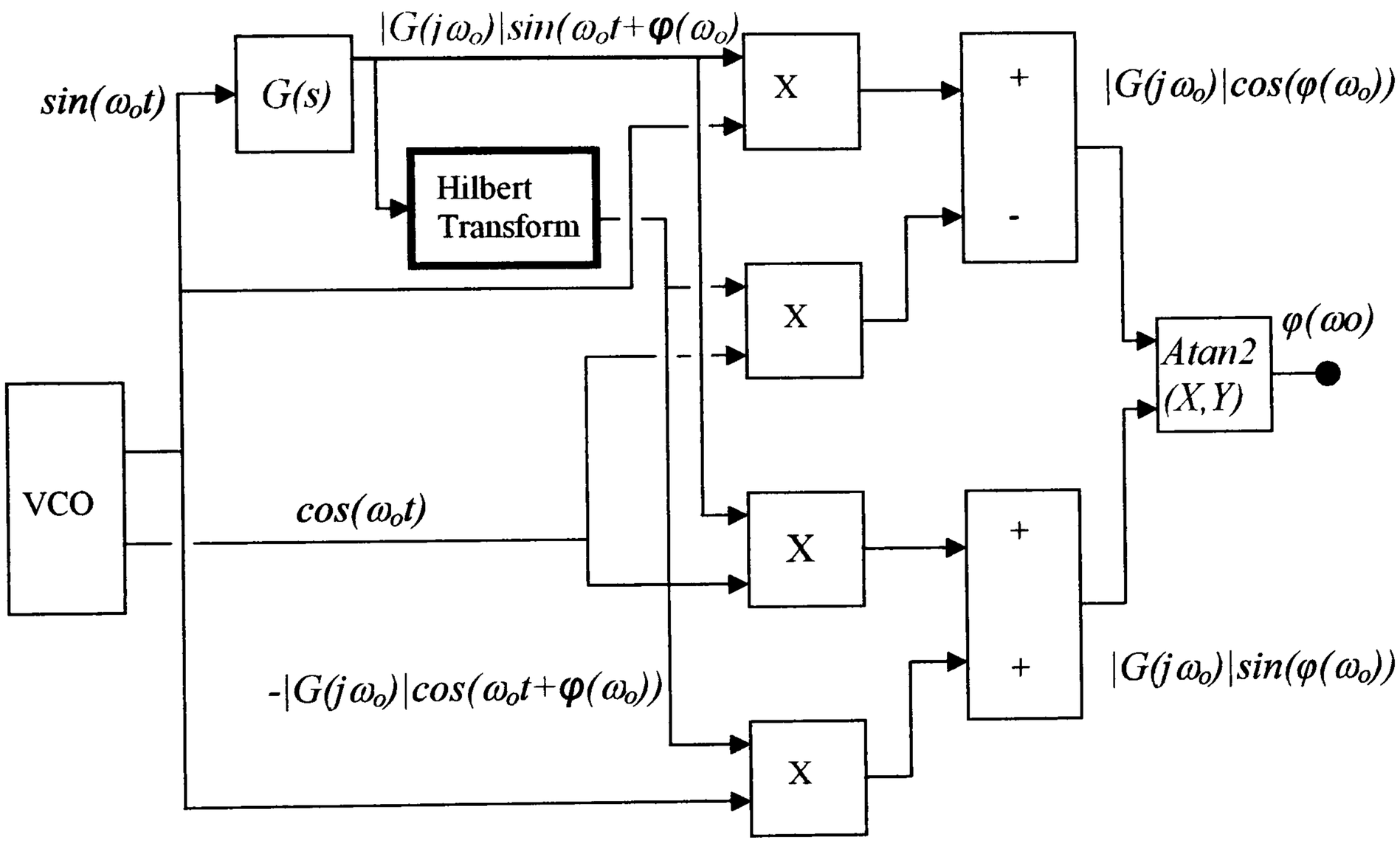


Figure 1.12: Conceptual Diagram of a Hilbert Transform Based Phase Detector.

With reference to Figure 1.12 it can be seen that

$$|G(j\omega_o)| \cos\phi(\omega_o) = |G(j\omega_o)|(-(-\cos(\omega_o t + \phi(\omega_o))\cos\omega_o t) + \sin(\omega_o t + \phi(\omega_o))\sin\omega_o t) \quad (1.10)$$

$$|G(j\omega_o)| \sin\phi(\omega_o) = |G(j\omega_o)|(\sin(\omega_o t + \phi(\omega_o))\cos\omega_o t$$

$$+\left(-\cos\left(\omega_o t+\phi(\omega_o)\right)\sin\omega_o t\right) \tag{1.11}$$

From equations (1.10) and (1.11) it is seen that the phase angle and magnitude of the frequency response of *G(s)* at the excitation frequency, ω_o, are readily obtainable. In Clarke and Park (2003) it was demonstrated that the Hilbert transform phase detector gave an improved performance with regard to noise rejection than did zero crossing detection methods and hence delivers an improved performance.

From the work of Balestrino *et al* and Clarke and Park it can be seen that structurally there is no difference between the Phase-Locked Loop identification methods that were first reported by Crowe and Johnson (1998, 1999). The main contribution of Balestrino *et al* was to use the simple expedient of initialising the Phase-Locked Loop identifier at a frequency close to the phase crossover frequency by the use of a relay experiment. The main contribution of the work by Clarke and Park was to carry out an assessment of possible phase sensitive detectors. The detectors that were compared were the zero crossing, synchronous demodulation and Hilbert Transform phase sensitive detectors. The candidate phase sensitive detectors were compared in terms of linearity, excitation amplitude dependence, harmonic content and noise rejection properties. The phase sensitive detector that gave the best performance against the selection criteria was the Hilbert Transform phase sensitive detector.

The Phase and Magnitude Detector Employed by Crowe and Johnson.

In the Phase-Locked Loop method of Crowe and Johnson the estimation of phase and gain data is carried out using a maximum and minimum peak value detection method. The process is excited by a cosine signal and the VCO generates not only the cosine excitation signal but a sinusoidal signal, both these signals after scaling, are used as multipliers of the process response. The extraction of the phase and magnitude data is then carried out using the maximum and minimum values of the multiplier outputs. This method employs the result that if a signal of the form

$$v(t)=|G(j\omega_o)|\left(\sin\left(2\omega_o t+\phi(\omega_o)\right)-\sin\phi(\omega_o)\right)$$

the output of the multiplier, is sampled at its maximum and minimum values then

$$|G(j\omega_o)|\sin\phi(\omega_o) = -\frac{v(t_{max}) + v(t_{min})}{2} \tag{1.12}$$

Similarly if a signal of the form

$$v(t) = |G(j\omega_o)|(\cos(2\omega_o t + \phi(\omega_o)) + \cos\phi(\omega_o))$$

is sampled at its maximum and minimum values then

$$|G(j\omega_o)|\cos\phi(\omega_o) = \frac{v(t_{max}) + v(t_{min})}{2} \tag{1.13}$$

Figure 1.13 shows the phase detector block diagram and details of the results (1.12) and (1.13) are given by Crowe (1998). Inspection of equations (1.12) and (1.13) show that the phase and gain data are easily derived.

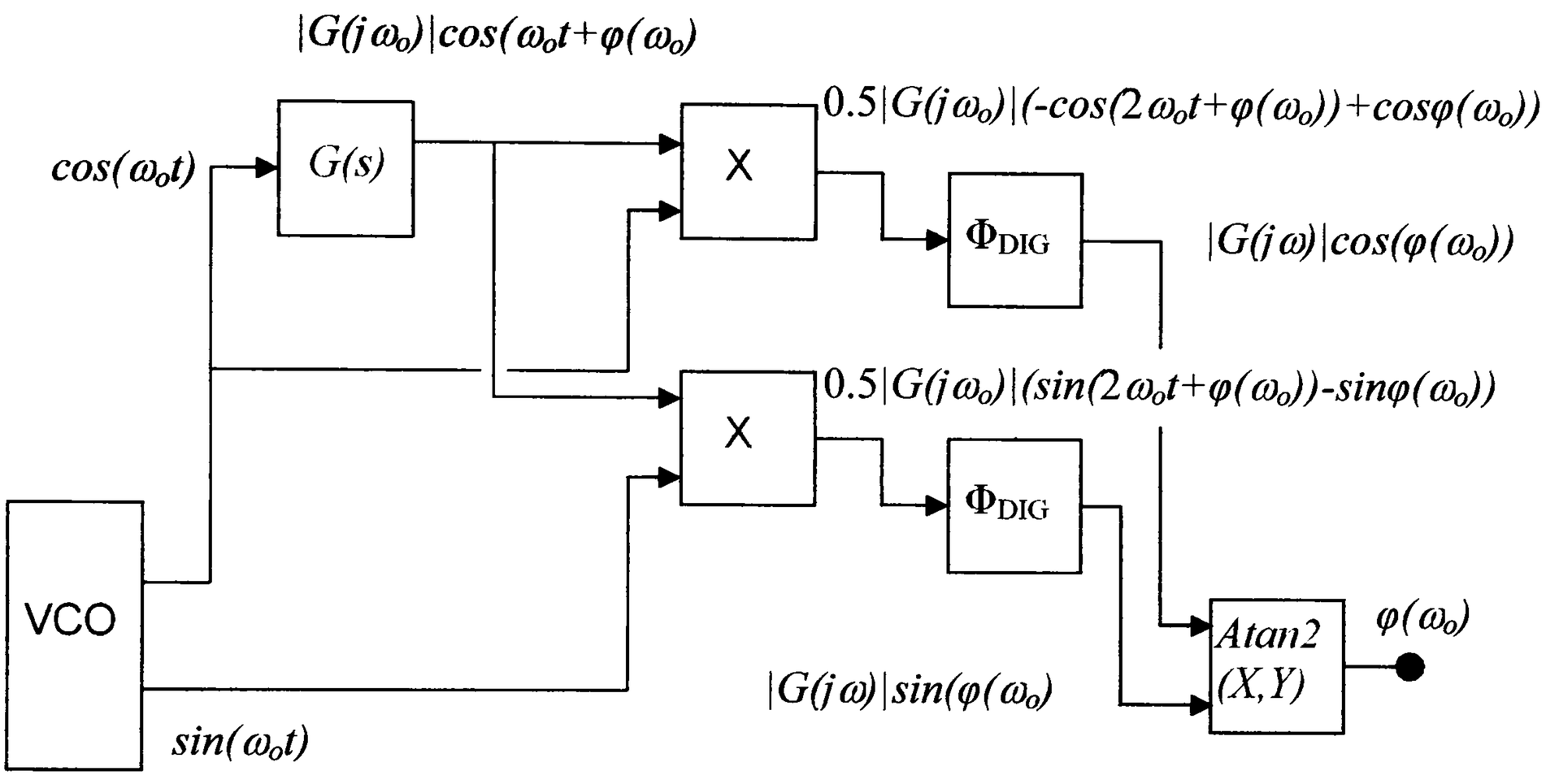


Figure 1.13: Block Diagram of the PLL Phase Detector Used by Crowe and Johnson.

1.3.2 Phase-Locked Loop Method for Identification of Type 1 Processes.

The Phase Locked Loop method of process identification, has been applied to a number of process identifications carried out both in open and closed loop configurations. A criticism that has been levelled at the method of Crowe and Johnson of Phase Locked Loop identification is that it is not possible to identify processes that are of type 1, viz. integrating type processes (Clarke and Park, 2003). The initial concept for the Phase Locked loop method was that it should be, as far as

possible, a direct replacement for the relay experiment but be more flexible in its application to process identification. Thus it was always assumed that the phase locked loop method would be used to identify processes connected in closed loop and thus in its present form can be used to identify type 1 processes in closed loop. However, in an effort to overcome this criticism and to allow open loop identification of type 1 systems to be carried out, a modification to the digital signal processor of the Crowe and Johnson Phase Locked Loop identifier is proposed.

In the Crowe and Johnson PLL identifier the Voltage Controlled Oscillator (VCO) is only allowed to update when the estimates of the process magnitude and phase angle have settled to within a specified tolerance. At that point in time, irrespective of where the VCO output waveform is in its cycle, the VCO frequency is updated and a signal at the updated frequency is initiated. The effect of this is twofold:

i) An offset is produced in the output of type 1 processes, and

ii) The offset produces an additional term at the excitation frequency in the output of the multiplier within the phase detector.

The additional term in the multiplier output prevents the estimates of magnitude and phase angle from settling and so the Crowe and Johnson PLL identifier fails to identify the type 1 process to a sufficient degree of accuracy. The cause of the offset is due to the fact that an integer number of excitation cycles have not been passed to the process and hence the integral term in the process has an output that will be different from zero.

A simple modification of the update timing logic within the digital signal processor such that an update of the VCO frequency is only allowed at the end of a complete excitation cycle remedies the problem. The modified Crowe and Johnson PLL identifier is used to identify the critical point data for the process

$$G_p(s) = \frac{1}{s(s+1)^2}$$

The evolution of the phase angle, magnitude and frequency are shown in Figures 1.15, 1.16 and 1.17 respectively. The simulation was carried out using Matlab/Simulink™ with an update and stopping tolerance of 0.002. The theoretical values for the magnitude and frequency at the critical point or phase crossover point

for $G_p(s)$ are 0.5 and 1.0 (rad.s^{-1}), respectively. From Figure 1.14 it can be seen that the desired phase angle is *accurately* attained after approximately 208 (s). The estimated phase angle was -3.1404 (rad) which is within the required stopping tolerance and gives an error in the estimated phase angle of 0.038%

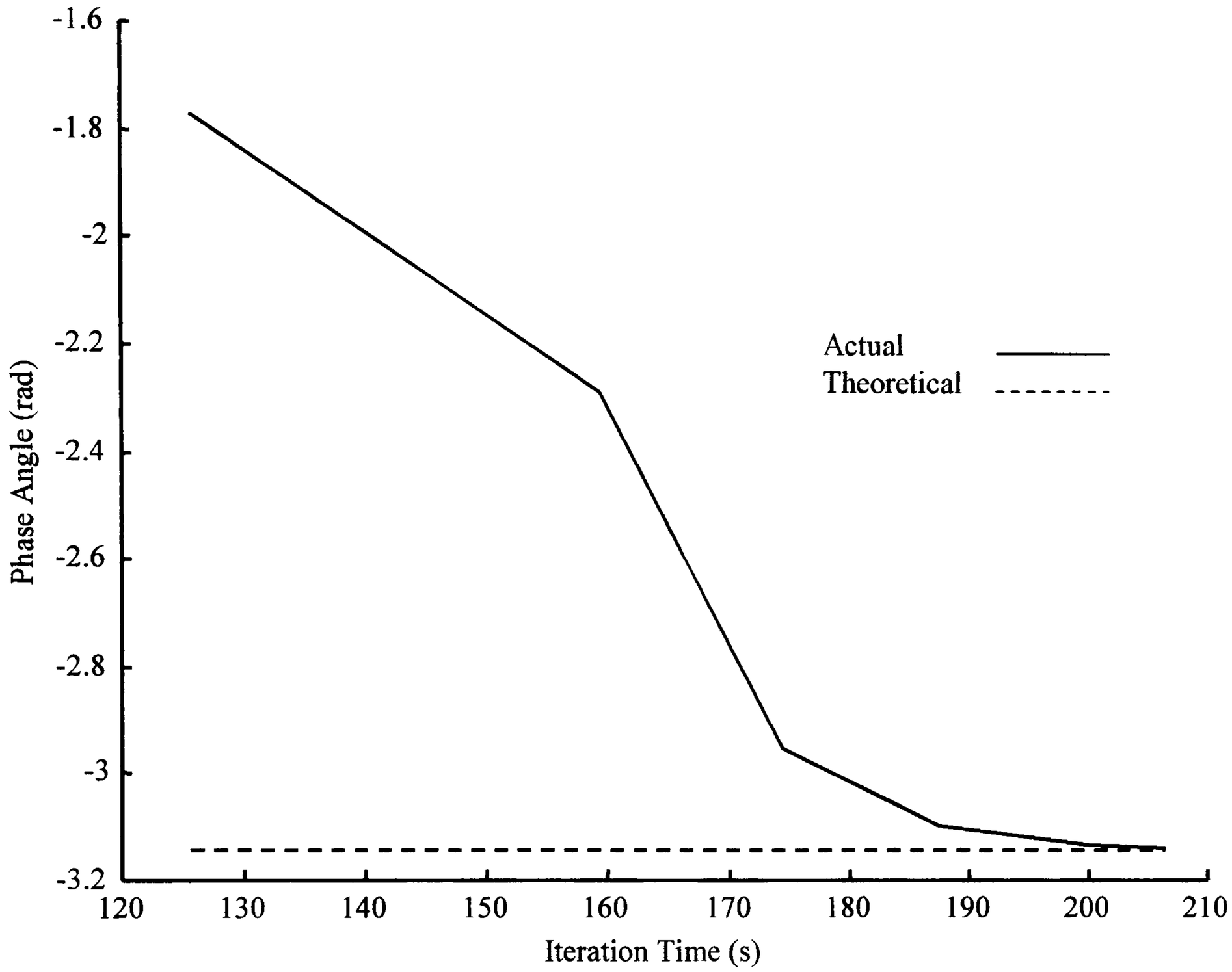


Figure 1.14: Evolution of Phase Angle with Time – Type 1 Process.

The evolution of the magnitude response can be seen from Figure 1.15, with the phase crossover point magnitude found as 0.5022. The percentage error between the theoretical value of phase crossover point magnitude, 0.5, and that found using the Crowe and Johnson PLL identification method is 0.44%.

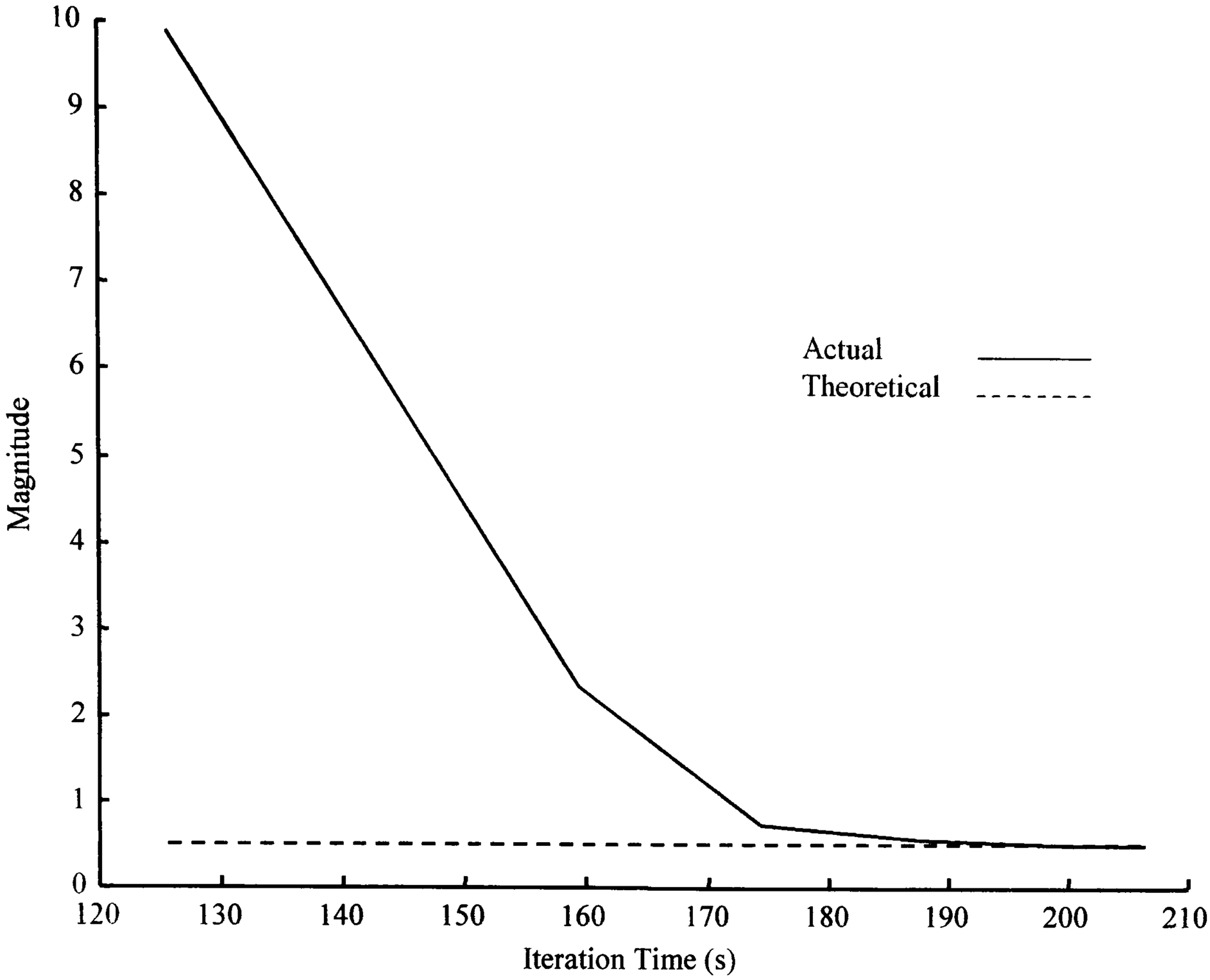


Figure 1.15: Evolution of Identified Magnitude – Type 1 Process.

The evolution of the excitation frequency is shown in Figure 1.16, with the final value of excitation frequency being 0.9977 (rad.s^{-1}). The error between this value and the theoretical value of 1.0 (rad.s^{-1}) is 0.23%. From this and other simulations carried out the modification to the Crowe and Johnson PLL identifier now allows the open loop identification of type 1 systems. However, closed loop identification remains the preferred method of carrying out identification for process control applications either with the controller being known or unknown.

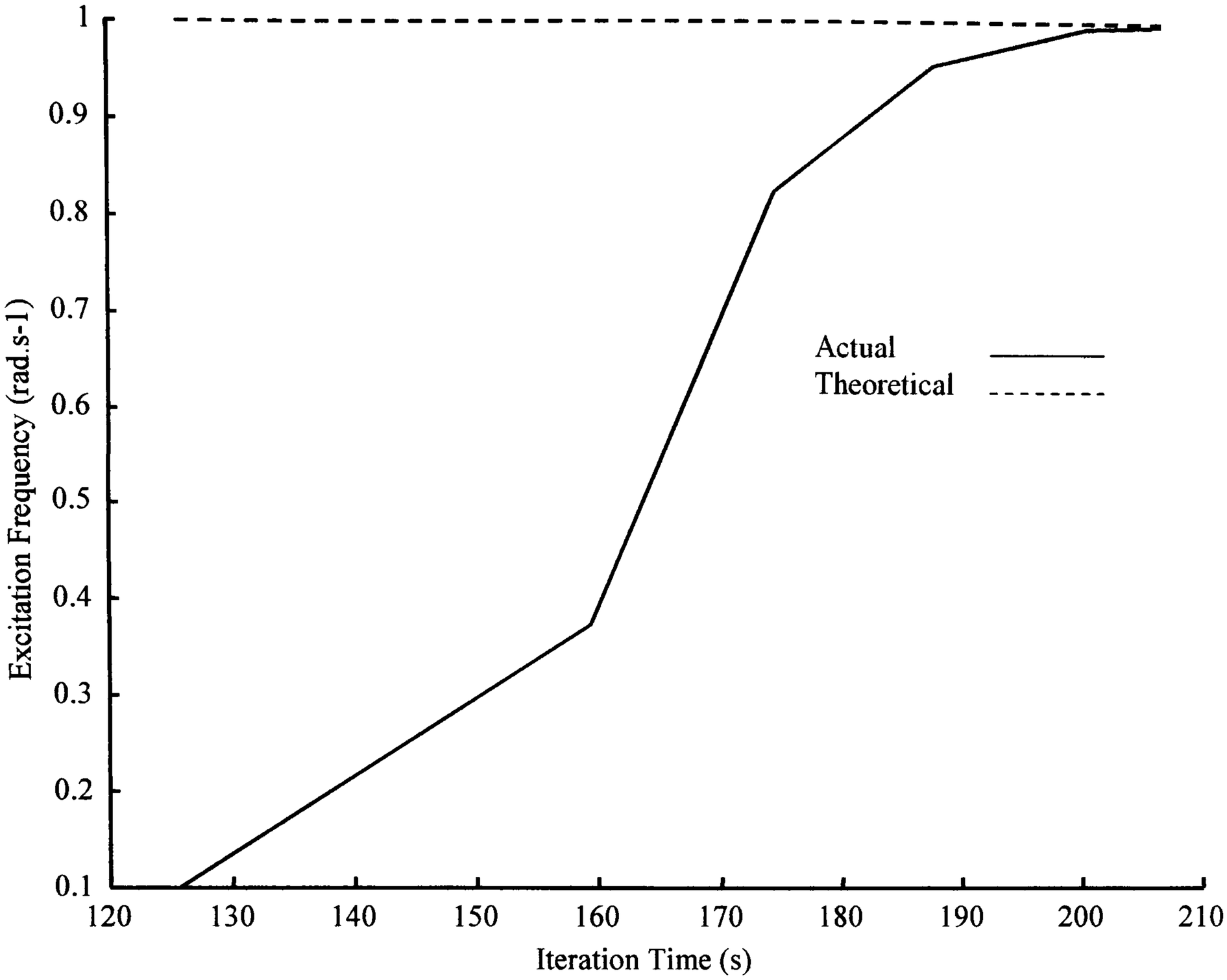


Figure 1.16: Evolution of Excitation Frequency– Type 1 Process.

1.3.3 Closed Loop Phase-Locked Loop Identification Methods.

In almost all practical situations, process control system identification will occur in closed loop. Two different identification situations can occur:

i) The transfer function of the controller and the process are both unknown, or

ii) The transfer function of the controller is known but that of the process is unknown.

The use of the Phase-Locked Loop method of process identification shall now be discussed where it shall be shown that the identification method, when applied to a closed loop system, has a similar ease of implementation as the relay experiment. In Figure 1.17 an unknown process $G_p(s)$ in a unity feedback configuration with an unknown controller $G_c(s)$ is shown.

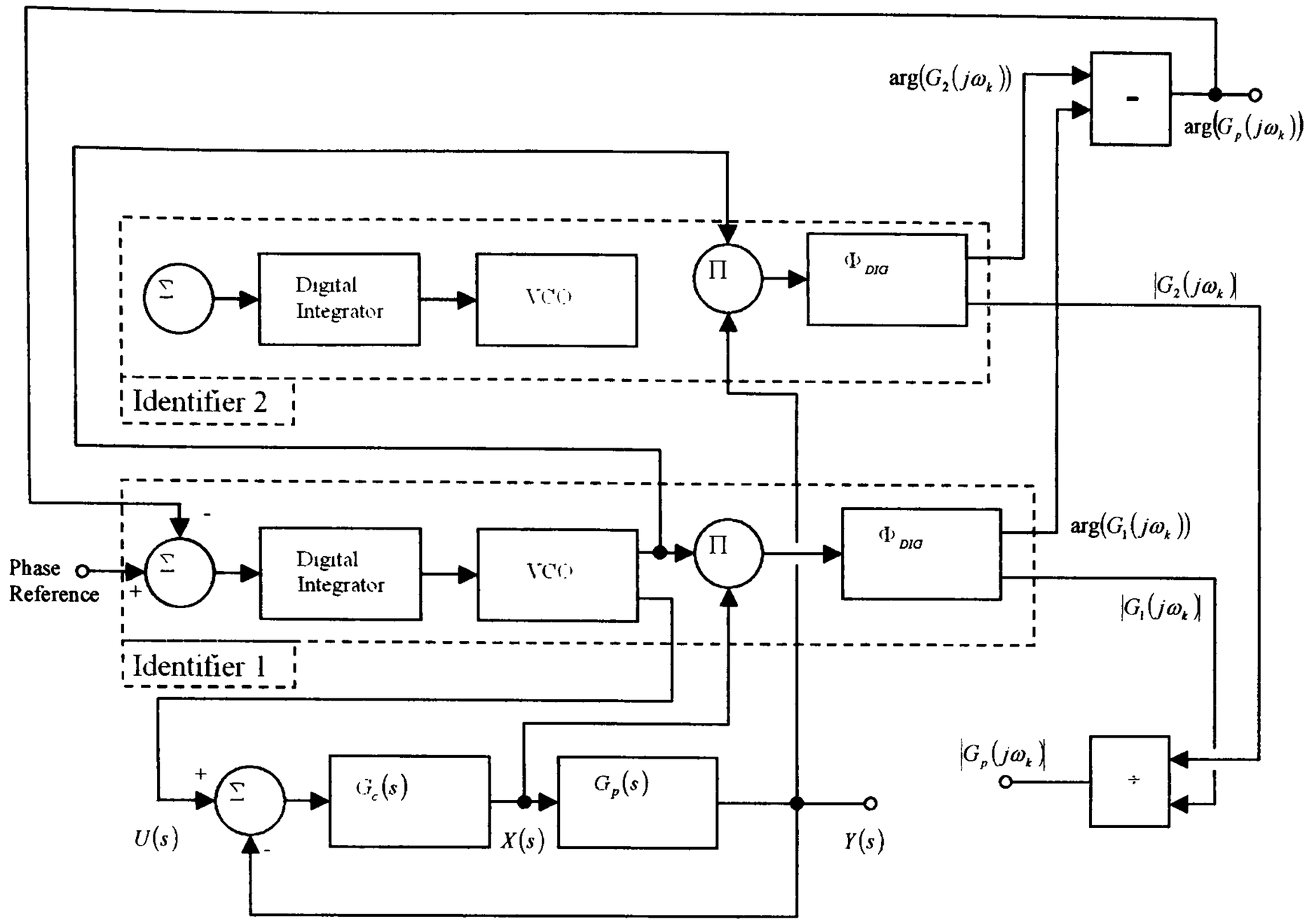


Figure 1.17: PLL Identification of an Unknown Process in Closed Loop with an Unknown Controller.

From Figure 1.17 it can be seen that two Phase-Locked Loop identifiers are used to carry out the identification of the unknown process, $G_p(s)$. The controller, $G_c(s)$, is also assumed to be unknown. It should be noted that there is only one excitation signal, generated by Identifier 1 and that the multiplier in Identifier 2 receives its VCO signal from Identifier 1. By this construct both PLL identifiers are synchronised to one excitation signal. Two identifications are carried out simultaneously, the first identification between $U(s)$, the reference input, and $X(s)$, the controller output. The second identification is carried out between $U(s)$ and $Y(s)$, the process output.

Using the identity $s = j\omega$, it is readily shown that (Crowe and Johnson, 2000b)

$$\arg\left(\frac{Y(j\omega)}{U(j\omega)}\right) - \arg\left(\frac{X(j\omega)}{U(j\omega)}\right) = \arg\left(G_p(j\omega)\right) \tag{1.14}$$

$$\left|\frac{Y(j\omega)}{U(j\omega)}\right| \div \left|\frac{X(j\omega)}{U(j\omega)}\right| = \left|G_p(j\omega)\right| \tag{1.15}$$

Hence by performing simultaneous identifications between the reference input and the controller output and between the reference input and the process output, it is possible to identify the process when it is connected in a closed loop configuration with an unknown controller.

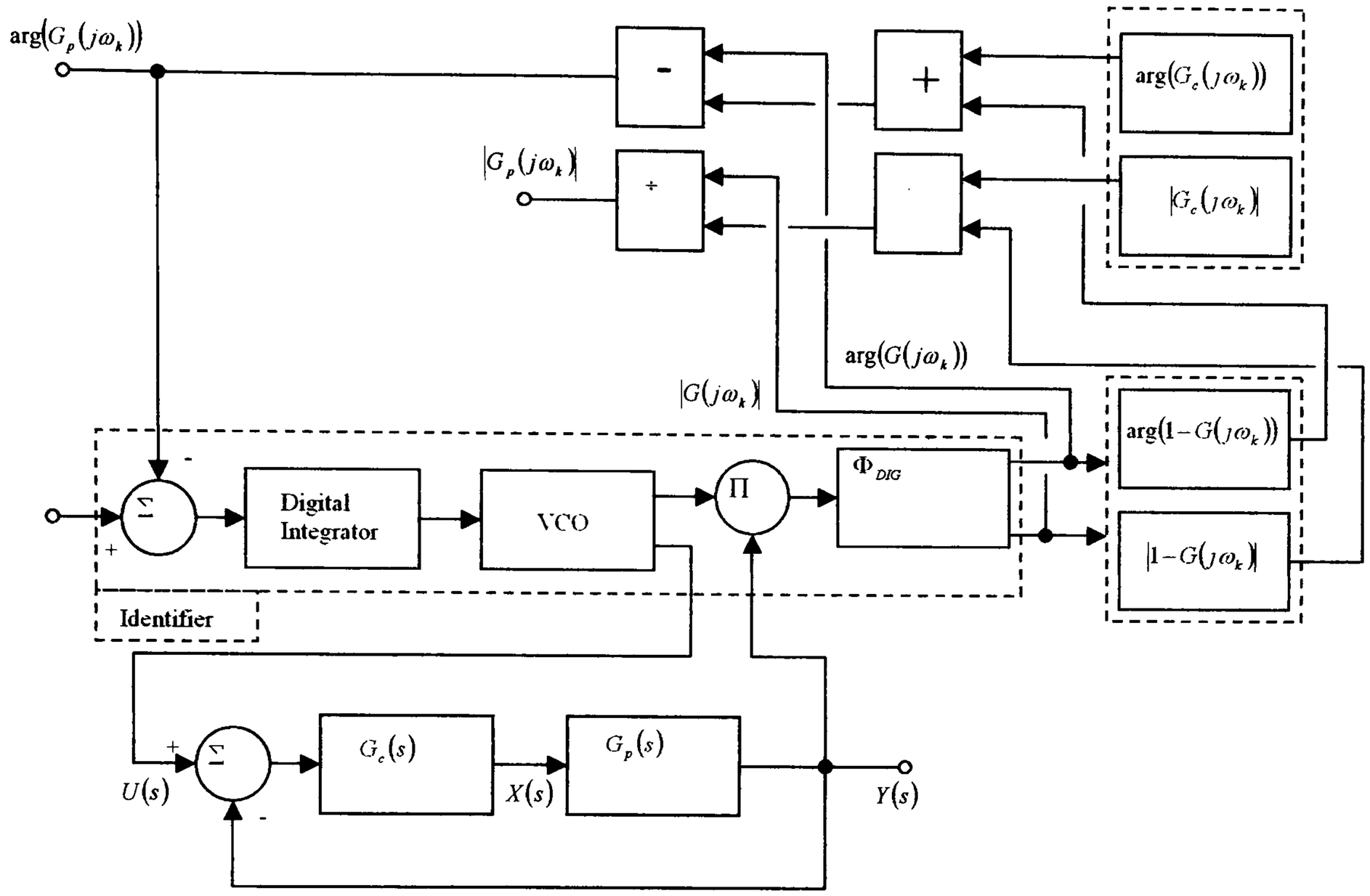


Figure 1.18: PLL Identification of an Unknown Process in Closed Loop with a Known Controller.

From Figure 1.18 it can be seen that only one Phase-Locked Loop identifier is used to carry out the identification. Since the controller transfer function and excitation frequency are known it is possible to determine the controller magnitude and phase angle at each excitation frequency. With this knowledge and the identification of the closed loop system between the reference input, *U(s)*, and the process output, *Y(s)*, it follows from Figure 1.18, setting $s = j\omega$ that the closed loop transfer function is (Crowe and Johnson, 2000b)

$$G(j\omega) = \frac{Y(j\omega)}{U(j\omega)} = \frac{G_c(j\omega)G_p(j\omega)}{1 + G_c(j\omega)G_p(j\omega)} \tag{1.16}$$

using equation (1.16)

$$G_p(j\omega) = \frac{G(j\omega)}{\left(1 - G(j\omega)\right)G_c(j\omega)} \tag{1.17}$$

to simplify the following notation let

$$\phi(\omega) = \arg\left(G(j\omega)\right)$$

then equation (1.16) can be represented by

$$G(j\omega) = |G(j\omega)|(\cos\phi(\omega) + j\sin\phi(\omega))$$

hence

$$|1 - G(j\omega)| = \sqrt{\left(1 - |G(j\omega)|\cos\phi(\omega)\right)^2 + \left(|G(j\omega)|\sin\phi(\omega)\right)^2} \tag{1.18a}$$

$$\arg\left(1 - G(j\omega)\right) = \tan^{-1}\left[\frac{|G(j\omega)|\sin(\phi(\omega))}{1 - |G(j\omega)|\cos(\phi(\omega))}\right] \tag{1.18b}$$

Thus if the closed loop transfer function is identified and the transfer function of the controller is known, then it is possible to identify the process as:

$$|G_p(j\omega)| = \frac{|G(j\omega)|}{|1 - G(j\omega)||G_c(j\omega)|} \tag{1.19}$$

$$\arg(G_p(j\omega)) = \arg(G(j\omega)) - \arg(1 - G(j\omega)) - \arg(G_c(j\omega)) \tag{1.20}$$

Thus by carrying out an identification between the reference input and the process output and with full knowledge of the controller it is possible to identify the process when it is connected in a closed loop configuration.

1.3.4 Use of the Phase-Locked Loop Method on Processes with Measurement Noise and Non-linearity.

In the applications of the Phase-Locked Loop method of system identification that have been discussed previously in this thesis, there has been no mention of the practical difficulties that may be encountered when using the method to identify a physical process.

For any physical process, the measurements taken from that process will be contaminated by noise to some degree. If no effective means of removing or

reducing the measurement noise is utilised then the data obtained from a noise contaminated measurement will be inaccurate. In Crowe (1998) a method of reducing the effect of measurement noise on the operation of the Phase-Locked Loop identifier is discussed. The method discussed utilises a Kalman filter to mitigate the effects of measurement noise on the data obtained from the Phase-Locked Loop identifier. A shortcoming of the technique is the increased identification time required to carry out the identification when measurement noise is present. However, the data obtained from the Phase-Locked Loop identification using a Kalman filter for measurement noise reduction is accurate.

In any physical process, and depending on the operating point of that process, there will be some degree of non-linearity. In the following a non-linear process shall be represented by a Hammerstein structure, viz. a static non-linearity followed by a dynamic linear model. In the majority of control applications within the process industries, the final control element is usually a control valve. A control valve can have a number of non-linear characteristics associated with it. Thus the non-parametric identification of a control valve with an associated non-linearity, in combination with a linear process model can adequately represent the type of problems to be found in the identification of physical processes. Saturation is a non-linear characteristic such that an increasing input signal produces no increase in the output signal. This type of non-linearity is usually the result of specifying too small a control valve or process pipe diameter, thus restricting the maximum flow through the valve to be below the required flow. When the control signal to a valve is cycled through its full range and the valve position is recorded then it can be seen that there is hysteresis in the characteristic of the valve position. The hysteresis effect can be reduced by using a valve positioner in the control loop. A third effect that can be caused by control valves is a dead zone non-linearity. This type of non-linearity is characterised by requiring a certain change in the input signal before any change in output signal is detected, thereafter there is a linear relationship between input signal and output signal. All process non-linearities are not attributable solely to the control valves, there are processes that are inherently non-linear. The titration curve used to carry out the neutralisation of an acid by an alkali exhibits a strong non-linearity.

In the case of a process that exhibits control valve induced non-linearity in its characteristic, then maintenance of the control valve should first be carried out prior to any process identification or controller tuning being carried out. If this is not done then the identification and tuning of the process will help to mask the underlying valve problem that may have severe consequences in terms of safety and lost production in the longer term. The maintenance of the control valve will also serve to reduce the non-linearity of the process.

When a process includes a non-linearity then care must be taken in the application of the Phase-Locked Loop identifier. Firstly, the magnitude of the excitation signal must be carefully chosen such that the non-linear process is identified at an operating point that is representative of that to be used in the physical process. If the process were to be controlled over a number of defined operating points, then identification at each of the defined operating points would have to be carried out. Secondly, the identification configuration must be carefully chosen. Consider the following example of the identification of the phase crossover frequency of the combination of a dead zone non-linearity and a linear time invariant process. The characteristic of the dead zone non-linearity used in the example is given by

$$y(t)=\begin{cases} x(t)-h & : \quad x(t)\geq h \\ 0 & : \quad -h<x(t)<h \\ x(t)+h & : \quad x(t)\leq -h \end{cases}$$

where *y(t)* is the output of the dead zone non-linearity, *x(t)* is the input to the dead zone non-linearity and *2h* is the width of the dead zone. The example linear time invariant process is given by

$$G_p(s)=\frac{10}{(1+0.2s)(1+0.7s)(1+s)}$$

It can be shown that the describing function of the dead zone non-linearity is given by

$$N(A,h)=\left(1-\frac{h}{A}\right)\left(1-\frac{2}{\pi}\sin^{-1}\left(\frac{h}{A}\right)\right)$$

where A is the peak value of the excitation sinewave and h is as described above.

From the describing function it can be seen that its value is real and depends on the magnitude of the excitation signal. Consider the open loop identification of the phase crossover frequency of the combination of the dead zone non-linearity and the process, $G_p(s)$ using the Phase-Locked Loop method. The results of the identification are given in Table 1.1.

Table 1.1: Open Loop Identification of Non-linear Process

	Magnitude	Phase angle (rad)	Frequency ($rad.s^{-1}$)	Time (s)
Theoretical	0.6425	-3.1416	3.6841	-
Actual	0.6682	-3.1414	3.674	237

The dead zone width is 0.2 and the peak excitation magnitude is unity. The theoretical values shown in Table 1.1 were calculated using the describing function of the non-linearity in combination with the process $G_p(s)$. As can be seen from Table 1.1 there is an error of -0.27% in the identified frequency and an error of +4% in the identified magnitude. The error in the identification is due to the harmonics of the excitation signal that are generated by the dead zone non-linearity. The error could be reduced by using a filter within the Phase-Locked Loop identifier as discussed by Clarke and Park (2003) or by using Fourier data extraction techniques as discussed in Crowe (1998).

If now the same combination of dead zone and process, $G_p(s)$ is identified in closed loop with the controller given by

$$G_c(s) = 0.79\left(1 + \frac{1}{0.85s} + 0.21s\right)$$

the following results, as shown in Table 1.2, are obtained

Table 1.2: Closed Loop Identification of Non-linear Process

	Magnitude	Phase angle (rad)	Frequency ($rad.s^{-1}$)	Time (s)
Theoretical	0.6927	-3.1416	3.6841	-
Actual	0.7227	-3.1410	3.6647	220

From the results of Table 1.2 it can be seen that there is an error of -0.53% in the identified frequency and an error of +4% in the identified magnitude. The identification error can be reduced by the use of a filter or by Fourier data extraction techniques as stated above.

As can be seen by comparing the results given in Tables 1.1 and 1.2, there is a difference both in the Theoretical and identified values of the magnitude of the combined process at the phase crossover frequency. The describing function for the dead zone non-linearity is purely real and hence has a phase angle of zero. Thus the phase crossover frequency of both the open and closed loop identifications is in good agreement with the theoretical values. However, The describing function of the dead zone non-linearity has a dependence on the magnitude of the excitation. In the open loop identification the excitation magnitude is set at unity but in the closed loop identification the magnitude of the excitation depends on the frequency response of the control sensitivity function, given by

$$C(s)=\frac{G_c(s)}{1+G_p(s)N(A,h)G_c(s)}$$

At the phase crossover frequency the controller output signal magnitude is approximately 1.75 although the closed loop reference signal has a magnitude of unity. Hence the identification configuration has a direct effect on the identified data.

Since the combined process will be operated in closed loop, a closed loop identification strategy should be employed. This will ensure that the non-linearity will be excited at magnitudes that are at the appropriate values for the closed loop and that any magnitude dependence of the non-linear elements of the combined process is included in the identification to reduce the effect of the harmonics of the excitation frequency.

The above brief discussion shows that where there is a significant non-linearity present in the process to be identified, then the identification should be carried out in closed loop. In order to increase the identification accuracy either a filter or Fourier data extraction techniques should be employed.

1.3.5 Summary Conclusions on the Phase-Locked Loop Method.

The Phase-Locked Loop (PLL) method of system identification (Crowe and Johnson, 1998; Johnson and Crowe, 1998; Crowe, 1998) was the direct result of research into non-parametric system identification methods and in particular the relay experiment of Astrom and Hagglund (1984). The application of the PLL method of system identification retains the simplicity of, but does not have any of the failings of the relay experiment. In simulated identification trials the PLL method has shown that accurate estimates of the required frequency response curve points are produced, regardless of the process frequency response characteristics. The PLL method of system identification has been extended to the closed loop identification of single-input single-output systems as reported by Crowe and Johnson (2000b) and Crowe *et al* (2001). Further extensions of the applicability of the PLL method to the identification of cascade systems and multi-input multi-output system are discussed subsequently in Chapter 4. In all of the identification trials carried out, where comparisons have been carried out against the relay experiment, the PLL method has taken a longer time to find the estimate of the point on the frequency response curve; albeit to a higher degree of accuracy. This may not be too great a failing since all of the identification data that is produced as the PLL approaches the required estimate is accurate and can be used to gain further information about the frequency response of the process being identified. Coupled to this is the ease with which a desired point can be found, both from the standpoint of the connection of the PLL to the system and the selection of a desired frequency response point or points to be identified, makes the PLL a viable tool for non-parametric system identification.

1.4 Subspace Identification.

Much of modern control theory is based on having a state space or transfer function representation of the system to be controlled available so that a controller design may be carried out. If such a model is not available then an identification of the process must be carried out before a controller design can be undertaken. Thus an

explicit model of the process is either available or must be identified. In subspace identification the model of the process that is obtained already has the necessary structure to allow many of the modern design methods such as LQG, H_∞ and Predictive control to be carried out using input and output data, from the system to be controlled, and subspace identification. Thus subspace identification is also termed a *model-free* approach in the literature since it is a data driven approach.

Assume that the states of a linear time invariant system are being supplied from a state estimation carried out by a Kalman filter. The state equations for the system can then be written as

$$x_{k+1} = Ax_k + Bu_k + Ke_k \tag{1.21}$$

$$y_k = Cx_k + Du_k + e_k \tag{1.22}$$

where the process inputs, outputs and states are given respectively by u_k, y_k and x_k. The steady state gain of the Kalman filter is given by K and e_k is an unknown white noise sequence with covariance given by

$$S = E\left\{e_k e_k^T\right\}$$

If it is assumed that there are n-states, l-inputs and m-outputs then the matrices are $A \in R^{n\times n}$, $B \in R^{n\times l}$, $C \in R^{m\times n}$, $D \in R^{m\times l}$ and $K \in R^{n\times m}$.

If the measurements of the inputs and outputs from the system u_k and y_k for $k=\{0, 1, \ldots, 2i+j-2\}$ are available, then the data block Hankel matrices for u_k, represented by U_p and U_f, with i-block rows and j-block columns are defined as

$$U_p = \begin{bmatrix} u_0 & u_1 & \cdots & u_{j-1} \\ u_1 & u_2 & \cdots & u_j \\ \cdots & \cdots & \cdots & \cdots \\ u_{i-1} & u_i & \cdots & u_{i+j} \end{bmatrix}$$

$$U_f = \begin{bmatrix} u_i & u_{i+1} & \cdots & u_{i+j-1} \\ u_{i+1} & u_{i+2} & \cdots & u_{i+j} \\ \cdots & \cdots & \cdots & \cdots \\ u_{2i-1} & u_{2i} & \cdots & u_{2i+j-2} \end{bmatrix}$$

where the subscripts p and f refer to past and future. Each block element in the above data Hankel matrices is a column vector of inputs, viz. $u_i = \begin{bmatrix} u_{i0} & u_{i1} & \cdots & u_{il-1} \end{bmatrix}^T$.

Similar data block Hankel matrices for y_k, represented as Y_p and Y_f, can be defined. The past and future state sequences are defined as

$$X_p = \begin{bmatrix} x_0 & x_1 & \cdots & x_{j-1} \end{bmatrix}$$

$$X_f = \begin{bmatrix} x_i & x_{i+1} & \cdots & x_{i+j-1} \end{bmatrix}$$

The matrix input-output equations used in subspace identification (Favoreel *et al*, 1998) are obtained by recursive substitution of equations (1.21) and (1.22)

$$Y_p = \Gamma_i X_p + H_i U_p + H_i^S E_p$$

$$Y_f = \Gamma_i X_f + H_i U_f + H_i^S E_f$$

These equations represent the effect of the state x_k, the deterministic input u_k and the unknown stochastic input e_k on the outputs y_k. The system related matrices are defined by

$$\Gamma_i = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{i-1} \end{bmatrix}$$

$$H_i = \begin{bmatrix} D & 0 & \cdots & 0 \\ CB & D & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ CA^{i-2}B & CA^{i-3}B & \cdots & D \end{bmatrix}$$

$$H_i^S = \begin{bmatrix} I & 0 & \cdots & 0 \\ CK & I & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ CA^{i-2}K & CA^{i-3}K & \cdots & I \end{bmatrix}$$

where Γ_i is the extended observability matrix, H_i and H_i^S are the lower triangular Toeplitz matrices containing the impulse response of the system due to the deterministic input u_k and the unknown stochastic input e_k respectively.

The subspace identification problem can also be stated as follows: given the past inputs and outputs W_p and the future inputs U_f, find an optimal prediction of the future outputs Y_f. If a linear predictor is used, then

$$\hat{Y}_f = L_w W_p + L_u U_f$$

where $$W_p = \begin{bmatrix} Y_p \\ U_p \end{bmatrix}$$

The least squares prediction $\hat{Y}_f$ of Y_f can be found by the solution to the following least squares problem:

$$\min_{L_w, L_u} \left\| Y_f - \begin{bmatrix} L_w & L_u \end{bmatrix} \begin{bmatrix} W_p \\ U_f \end{bmatrix} \right\|_F^2$$

where the subscript F denotes the Frobenius norm. For a matrix, $C = [c_{jk}]$, the Frobenius norm is given by

$$\|C\| = \sqrt{\sum_{j=1}^{n} \sum_{k=1}^{n} c_{jk}^2}$$

The solution to this problem is the orthogonal projection of the row space of Y_f into the row space spanned by W_p and U_f, defined as (Favoreel *et al*, 1998)

$$\hat{Y}_f = Y_f \Big/ \begin{bmatrix} W_p \\ U_f \end{bmatrix} \tag{1.23}$$

$$= \underbrace{Y_f /_{U_f} W_p}_{L_w W_p} + \underbrace{Y_f /_{W_p} U_f}_{L_u U_f}$$

$$= Y_f \begin{bmatrix} W_p^T & U_f^T \end{bmatrix} \left(\begin{bmatrix} W_p \\ U_f \end{bmatrix} \begin{bmatrix} W_p^T & U_f^T \end{bmatrix} \right)^{-1}$$

The numerical implementation of the projection, equation (1.23), can be carried out in a numerically robust manner by using a QR-decomposition (Favoreel *et al*, 1998).

An implementation of subspace identification was briefly discussed in the above section. Subspace methods of controller design are termed *model-free* methods in that they are data driven, using input and output data from the system to be controlled to allow a controller to be designed without the explicit step of model identification being carried out. The literature on the use of subspace methods for controller tuning shows successful applications for LQG (Favoreel *et al*, 1998; Favoreel *et al*, 1999), H_∞ (Woodley *et al*, 2001) and predictive control (Kadali *et al*, 2003) design methods. Subspace methods of controller tuning illustrate the transition from explicit model methods to the *model-free* method of Iterative Feedback Tuning.

1.5 Iterative Feedback Tuning

A short introduction to Iterative Feedback Tuning is given in this section to complete the overview of the *model-free* controller tuning methods. A brief reference to the frequency domain version of Iterative Feedback Tuning due to Kammer *et al* (2000) is also given. The IFT method shall be revisited in Chapter 5 where new and original contributions to the method are detailed.

In IFT the derivative of a control performance cost function with respect to the controller parameters is obtained by carrying out experiments on the closed loop system (Hjalmarsson *et al*, 1994; Hjalmarsson *et al*, 1998) and using the data recorded from those experiments in further *special* experiments. Thus after each set of experiments have been completed, an un-biased estimate of the cost function gradient is available. The estimate of the gradient is then used in a stochastic estimation algorithm that by repeated application of the method yields the controller parameters that give the optimal value of the cost function. The discussion of Iterative Feedback Tuning given in Hjalmarsson *et al*, (1994) and; Hjalmarsson *et al*, (1998) is based on a somewhat broad problem formulation. The salient features of that method are:

i) a system description involving a stochastic process output disturbance

ii) a two degrees of freedom control law

iii) the use of a stochastic optimisation approach, and

iv) a restricted structure control law

The benefit of the method to industrial control practitioners can be seen from the fact that no process model is required during the tuning process and that all of the experiments are carried out in closed loop. At each iteration of the Iterative Feedback Tuning method the current controller parameters remain fixed and are only updated at the end of an iteration. By repeated application of the Iterative Feedback Tuning method the performance of the closed loop system should improve as the optimal value of the cost function is approached.

A frequency domain version of the IFT method has been described by Kammer *et al* (2000). In Kammer *et al* use is made of Parsevals Theorem to transfer the cost function to the frequency domain. The transformed cost function is then differentiated with respect to the controller parameters. The derivative terms in the cost function gradient expression are recovered by the use of spectral analysis of the closed loop signals and it is assumed that there is full knowledge of the controller available.

The Kammer *et al* method is the frequency domain version of the Hjalmarsson *et al* method. However the Kammer *et al* method does have a benefit over the Hjalmarsson *et al* method in that it is possible to determine estimates of the Hessian of the cost function and so improved numerical methods can be implemented to provide the controller parameter update.

1.6 Summary Conclusions.

A discussion of the relay experiment of Astrom and Hagglund (1984) in relation to recent international research activity was given. The conclusion from this discussion is that the latest research results tend to be increasing the application areas of the relay experiment at the expense of the simplicity of the method. Additionally there appears to be no literature that supports the view that an increase in industrial use of the new relay experiment application areas is being made.

The Phase-Locked Loop (PLL) method of system identification was discussed and an extension to the PLL method allowing the open loop identification of type 1 processes was given. The identification of single-input single-output processes connected in closed loop was detailed. The literature is beginning to show increased interest in the PLL method of system identification and a discussion of the contribution of this research was given.

Subspace identification was discussed briefly, acting as a bridge between the explicit modelling techniques of the relay experiment and the PLL method and the model-free techniques used in Iterative Feedback Tuning (IFT) and Continuous Parameter Cycling. A brief introduction to IFT was given.

2 Testing for the Existence of PID Controllers that can Achieve Specified Classical Robustness Measures.

2.1 Introduction.

The design of a PID controller can be carried out relatively easily if there is an accurate model of the process available for the designer to perform an off-line controller design and simulation of the resulting closed loop control system. However, especially in the process industries, an accurate model of the process to be controlled is seldom available and an environment supporting the identification, design and simulation cycle is rarely to be had. It is then not surprising that few control loops in process industries are operating satisfactorily and that some processes are controlled manually (Hersh and Johnson, 1997).

The data obtained from carrying out a step test on the open loop process can be used to determine first or second order plus dead time process models (Seborg *et al*, 1989). Similarly frequency response data can be obtained relatively simply from the process, by the use of the Ziegler and Nichols ultimate period method or by a relay experiment, to allow the design of PID controllers to be carried out. For a large class of industrial plant these methods will provide sufficiently accurate process data to allow the design of a PID controller to be carried out that will give an acceptable level of control system performance. However there is reluctance on the part of process owners to allow tests to be carried out on process equipment that may result in either lost production or in the production of off-specification product.

Faced with the problem of having to design a controller to meet certain design requirements, how is the control practitioner to carry out this task when no process model is available and the process owner is reluctant to allow tests to be carried out? Additionally there is the problem of determining whether the required design requirements can in fact be met prior to the design being carried out. The first of these problems can be alleviated by the use of closed loop testing to reduce the production of off-specification product, so long as the test signals are not of too great a magnitude and the testing is not over an extended period of time. Thus it would be possible to determine a process model and carry out the design. However, the

problem still exists of knowing whether the design requirements can in fact be attained prior to using the model in the design. What is required is a method that allows the designer to see what designs are achievable for a particular process and allows the freedom to choose a candidate control design from those that meet the requirements of the design.

A review of the literature on PID controller tuning to achieve gain margin and phase margin robustness measures gives rise to Figure 2.1; this shows how the various methods can be categorised.

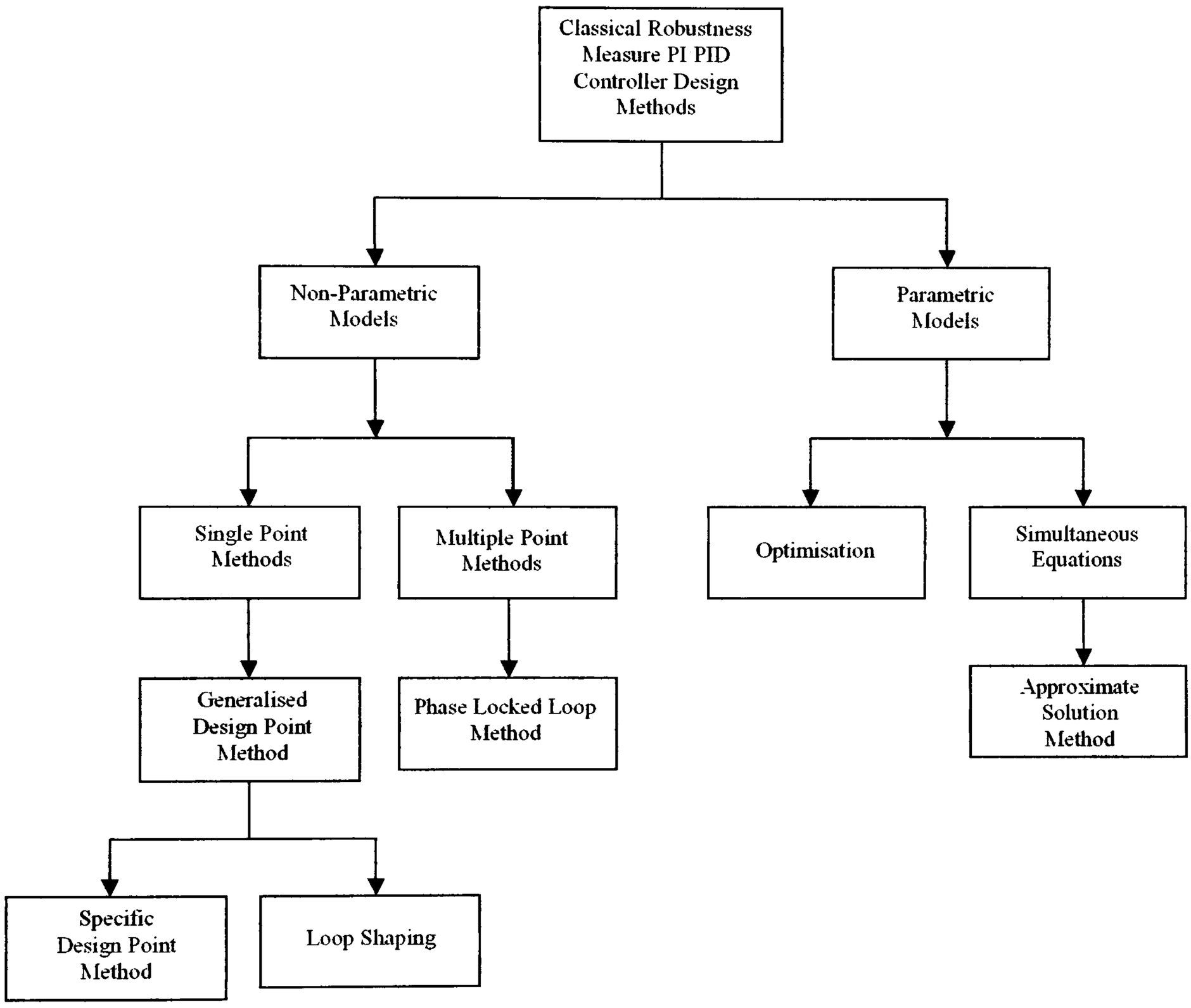


Figure 2.1: Categorisation of PID Controller Tuning Methods by Model Type Used.

With reference to Figure 2.1 the first distinction that is made between PID controller tuning methods is whether the process is given by a parametric or non-parametric model representation. A number of PID controller tuning techniques have been developed that require the availability of a parametric model of the process to

allow the design of a PID controller to meet the required gain margin, phase margin or maximum sensitivity specifications and in Figure 2.1 the classification of these methods is considered under either optimisation methods and methods that either solve a set of simultaneous equations exactly or provide an approximate solution.

Simultaneous Equations

An accurate parametric process model is required for the gain margin and phase margin tuning method proposed by Fung *et al* (1998). In this method the set of simultaneous equations relating to the specification of the gain margin and phase margin of the forward path transfer function of the compensated system are solved by a graphical means. This method shall be discussed subsequently in this chapter. An extension of the method is proposed in Tan *et al* (1999) such that the parameters of a PID controller can be established that will return a required gain and phase margin design. The method as discussed does however lose the benefit of a graphical interpretation of the results since the method simply returns a set of controller parameters. There is a trade-off required by the method in that five unknown parameters are required to be determined by the method, those being the three PID controller parameters and the gain crossover and phase crossover frequencies of the forward path of the compensated process. However the system of equations derived at the gain and phase crossover points of the compensated process only provide four equations and thus a possibly infinite set of solutions. The trade-off that is used is to set the frequency of the phase crossover point of the compensated system as a constant times the phase crossover frequency of the uncompensated process. The constant used is in the range 0.5 to 2. By this means the number of unknown parameters is reduced to four and a unique solution can be obtained.

Approximate Solution Method

The method employed by Fung *et al* and extended by Tan *et al* allows both the gain margin and phase margin to be met, however an accurate model of the process to be controlled is required to be available. The method discussed in Ho *et al*

(1995) uses an approximate parametric model of the process to be controlled. The process model used is first order plus dead time and both the PI controller and the process model are given in time constant form. The equation set used by Ho *et al* differs from that used by Fung *et al*. The difference comes from the way that Ho *et al* expresses the gain margin and phase margin conditions for the transfer function of the compensated forward path of the system to be controlled. For the gain margin and phase margin equations the complex nature of these equations is expressed using angle and magnitude criteria whereas Fung *et al* split the equations into their real and imaginary components. The equations resulting from the method used by Ho *et al* are non-linear due to the presence of an inverse tangent function. This does not present any great difficulty in itself since it would be possible to solve the equation set using a numerical method. However the goal of Ho *et al* was to achieve an analytic solution and to achieve that goal an approximation for the inverse tangent function is introduced. Thus by approximating the inverse tangent function by a linear function, an analytic solution for the PI controller parameters can be found. Ho *et al* extended the PI tuning method to PID controllers by using a second order model plus dead time, expressed in time constant form, of the process to be controlled. The PID controller is given as an interacting type. Ho *et al* then utilises the model pole with the largest time constant to cancel the controller zero introduced by the derivative term. Thus by setting the controller derivative time constant to be equal to the largest time constant of the model poles the problem collapses to that of designing a PI controller as previously posed and solved by Ho *et al*. This method of controller tuning to meet gain and phase margin specifications has a number of potential problems. A model is used to represent the process to be controlled, if the model is a good representation of the process in the frequency range of interest then the method will give good results. However, the poorer the fit of the model to the process then the worse will be the resulting controller design with respect to achieving the required design specification. The approximation to the inverse tangent function is valid only for a limited set of conditions and so if the conditions are violated then the method will not perform as well as expected.

Optimisation

The methods due to Fung *et al*, Tan *et al* and Ho *et al* are used to design PID controllers such that a specific gain and phase margin are achieved. These methods are characterised by the requirement to have a parametric system model available to allow a set of simultaneous equations to be solved either exactly or by using approximations as shown in Figure 2.1.

The maximum sensitivity of a system can be used as a guide to the closed loop time domain behaviour of the system (Astrom *et al*, 1998). The method developed by Astrom *et al* (1998) utilises a parametric model of the process. The PI controller that is designed is specified in terms of the proportional and integral term gains, k_p and k_i respectively. The method begins by defining a circle centre (-1, 0) and radius $1/M_s$ on the Nyquist diagram of the frequency response of the compensated system, where M_s is the desired maximum sensitivity. The circle represents a constraint on the frequency response curve such that the distance from the circle centre to any point on the frequency response curve cannot be less than $1/M_s$, thus if the parameters of a PI controller can be found such that the frequency response curve of the compensated system is tangent to the circle then the maximum sensitivity of the closed loop system will be M_s. The method can be augmented to include a constraint on the peak overshoot by defining a circle C, whose centre and radius are such that the peak overshoot, M_p circle, and maximum sensitivity, M_s circle are contained within C. The function that is to be optimised to provide the largest value of k_i, is given by

$$f(k_p, k_i, \omega) = \left| C + \left(k_p - j\frac{k_i}{\omega} \right) G_p(j\omega) \right|^2 \tag{2.1}$$

with the constraint that

$$f(k_p, k_i, \omega) \geq \left(\frac{1}{M_s} \right)^2 \tag{2.2}$$

The constraint (2.2) has the geometrical interpretation that for a fixed ω, equation (2.2) represents an ellipse in the k_p-k_i plane. If the value of ω is now iterated over the range $0 \leq \omega < \infty$, then it is found that the ellipses form an envelope that defines the

boundaries of the set of PI controller parameters that satisfy the constraint on maximum sensitivity. This situation is illustrated in Figure 2.2

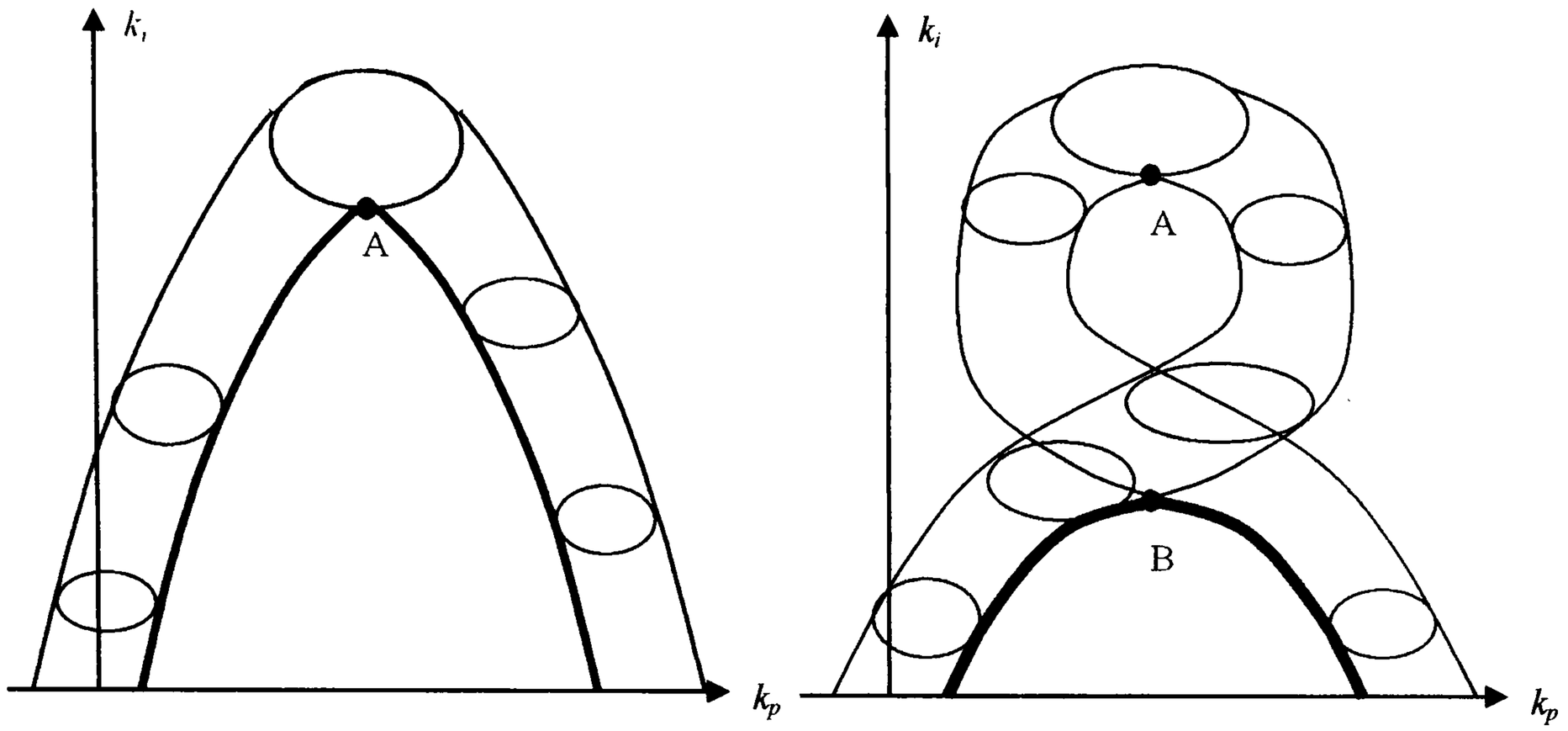


Figure 2.2: Envelope generated by the sensitivity constraint

The left hand envelope shown in Figure 2.2 has a continuous derivative. In carrying out the generation of the ellipses a situation can arise where a corner occurs in the envelope. The corner corresponds to multiple points on the frequency response curve of the compensated system that are tangent to the $1/M_s$ circle; thus giving rise to the possibility of there being multiple sets of controller parameters that yield a solution to the problem. An envelope with a corner is illustrated by the right hand diagram of Figure 2.2. In Figure 2.2 stabilising solutions are represented by the bold curves. The optimisation of equation (2.1) subject to the constraint (2.2), following some simplifications based on the total differential of $f(k_p, k_i, \omega)$, results in an equation that can be solved numerically to derive the frequency at which the controller parameters are to be calculated. The optimisation method was extended to allow the determination of the parameters of a PID controller to meet a maximum sensitivity constraint by Panagopoulos *et al* (1999). In this extension the constraint condition is augmented to include constraints on the curvature and rate of change of the phase angle of the frequency response of the forward transfer function of the compensated system. By including a constraint on the curvature, ensuring that it is negative, guarantees that the frequency response of the forward path of the compensated

system is tangent to the $1/M_s$ circle at only one point. The constraint calling for the rate of change of phase angle of the frequency response of the forward path transfer function of the compensated system to be negative is used to ensure that at high frequency there is no undesirable phase lead present. The constraints on curvature and phase angle do however increase the complexity of the optimization problem and in certain circumstances the ratio of integral to derivative controller parameters is less than four, hence the PID controller cannot be implemented in series form (Astrom and Hagglund, 1995).

The above methods all rely to a certain degree on having an accurate parametric model of the process available. If there is no parametric model of the process available then, as shown in Figure 2.1, methods have been developed to utilise a non-parametric model of the process in the design of a controller that will return a required gain margin, phase margin or maximum sensitivity. In the literature there is a distinction drawn between non-parametric methods that require only one point of the frequency response of the transfer function of the process to be identified and methods that require multiple points.

Generalised Design Point Methods

In single point tuning methods a particular point is identified on the frequency response of the process transfer function. The aim of the design is to be able to move that point, by the use of a compensation element, to some other desired identified point. The ultimate frequency method of Ziegler and Nichols (1942) is a single point method that relies on being able to identify the particular point on the frequency response curve of the process at which the phase crossover occurs. By identifying this point and then by the application of a rule based method the parameters of a PID controller can be chosen that will in general give an adequate degree of control for the closed loop system. The generalisation of that method and the frequency domain interpretation of moving a general point on the frequency response curve of the process to a desired point of the compensated frequency response curve is due to Astrom and Hagglund (1984).

Specific Design Point Method

In the Astrom and Hagglund method, specific design points are able to be achieved by the use of PI or PID controllers as the compensation elements of the closed loop system. The design points that are used are the gain margin and phase margin. However, the gain margin and phase margin cannot both be achieved by the same controller design as in the methods provided by Ho *et al*, Fung *et al* and Tan *et al*.

Loop Shaping

In addition to achieving a specific design point, single point non-parametric methods can be used to shape the frequency response of the forward transfer function of the compensated process such that a design specification that minimises the maximum sensitivity locally can be attained. The method described by Astrom and Hagglund (1995) causes the frequency response of the forward path of the compensated system to have a gradient that is orthogonal to the line $1+G_p(j\omega_s)G_c(j\omega_s)$, where $G_p(j\omega_s)$ and $G_c(j\omega_s)$ are the frequency responses of the process and the controller, respectively, at the design point frequency ω_s.

In an effort to improve on the design results achieved by single point design point methods, multiple point methods have been reported. By identifying two points on the frequency response of the process by means of a relay experiment and a relay experiment with hysteresis Shin *et al* (1997) employ a pole placement strategy to attain a desired damping ratio for the closed loop system. Multiple point non-parametric identification is also used in the phase locked loop design point methods that are described in the following sections of this chapter.

From the preceding discussion it can be seen that there are a number of competing PID controller design methods available. The problem associated with these methods is that until a design has been carried out it is not possible to know whether or not that design will be achievable. In the following sections of this chapter, methods shall be described that show the gain and phase margin design pairings that are achievable for an unknown process. The tool that shall be used to

allow this to be undertaken is the Phase-Locked Loop identifier discussed in Chapter 1. In section 2.2 an automated method is described that returns, in a graphical format, the range of achievable gain and phase margins that a PI controller acting on an unknown process can provide. A discussion of a method that allows achievable gain and phase margin ranges for an unknown process controlled by a PID controller to be calculated is carried out in section 2.3. Conclusions close the chapter.

2.2 Classical Robustness Measures: Automated Existence Testing.

Determining the controller parameters for a PI controller based on the specification of the gain margin or phase margin is a classical method of controller design (Wilkie *et al*, 2002). In general there is no direct relationship between the gain margin of a process and the time domain response of the closed loop system. In Ogata (1997) it is stated that the connections between gain and phase margin and the time domain response of the closed loop system are laborious and not of much practical use. However, it is found in practice that a satisfactory time domain closed loop response is usually obtained with gain margins in the range 2 to 5 and phase margins in the range 30° to 60° (Astrom and Hagglund, 1995). The gain margin and phase margin are decoupled measures of robustness and are used in connection with single input single output systems, having no direct applicability to multi-input multi-output systems. If a process given by $G_p(s)$ is controlled by a controller $G_c(s)$ then the gain margin and phase margin are given, respectively, by

$$GM = \frac{1}{\left|G_p(j\omega_{-\pi})G_c(j\omega_{-\pi})\right|} \tag{2.3}$$

$$\phi_{PM} = \pi + \arg\left(G_p(j\omega_1)G_c(j\omega_1)\right) \tag{2.4}$$

where $\omega_{-\pi}$ is the frequency at which

$$\arg\left(G_p(j\omega_{-\pi})G_c(j\omega_{-\pi})\right) = -\pi \tag{2.5}$$

and ω_1 is the frequency at which

$$\left|G_p(j\omega_1)G_c(j\omega_1)\right| = 1 \tag{2.6}$$

The gain margin and phase margin are used to establish the stability of the closed loop system and also its robustness to process changes.

In the following, automated methods shall be discussed that allow the existence of the classical robustness pairings of gain margin and phase margin to be shown graphically.

2.2.1 Automated Existence Testing of Gain Margin and Phase Margin Pairings Achievable by PI Controllers.

In the method due to Fung *et al* (1998) a parametric model of the process is assumed to be available. The method allows a test to be carried out that shows if a PI controller can provide a specified gain margin and phase margin for the given model. The method due to Fung *et al* shall now be detailed.

If it assumed that the process $G_p(s)$ is to be controlled using the controller $G_c(s)$ given by

$$G_c(s) = k_p + \frac{k_i}{s}$$

Then, using the substitution $s=j\omega$, it can be seen that the forward transfer function of the system is given by

$$G_c(j\omega)G_p(j\omega) = \left(k_p - j\frac{k_i}{\omega}\right)\left(G_{pR}(\omega) + jG_{pI}(\omega)\right) \tag{2.7}$$

where $G_{pR}(\omega)$ and $G_{pI}(\omega)$ represent the real and imaginary components of $G_p(j\omega)$, respectively. Using equations (2.3), (2.5) and (2.7) it follows that

$$\begin{bmatrix} G_{pR}(\omega_{-\pi}) & \dfrac{G_{pI}(\omega_{-\pi})}{\omega_{-\pi}} \\ G_{pI}(\omega_{-\pi}) & -\dfrac{G_{pR}(\omega_{-\pi})}{\omega_{-\pi}} \end{bmatrix} \begin{bmatrix} k_p \\ k_i \end{bmatrix} = \begin{bmatrix} -\dfrac{1}{GM} \\ 0 \end{bmatrix} \tag{2.8}$$

Solving equation (2.8) for k_p and k_i gives

$$k_p = -\frac{\cos\phi_p(\omega_{-\pi})}{GM\left|G_p(j\omega_{-\pi})\right|} \tag{2.9}$$

$$k_i = -\frac{\omega_{-\pi}\sin\phi_p(\omega_{-\pi})}{GM\left|G_p(j\omega_{-\pi})\right|} \tag{2.10}$$

where $\phi_p(\omega_{-\pi})$ and $|G_p(j\omega_{-\pi})|$ represent the phase angle and magnitude of the frequency response of the process at the phase crossover frequency.

From equations (2.4), (2.6) and (2.7) it follows that

$$\begin{bmatrix} G_{pR}(\omega_1) & \dfrac{G_{pI}(\omega_1)}{\omega_1} \\ G_{pI}(\omega_1) & -\dfrac{G_{pR}(\omega_1)}{\omega_1} \end{bmatrix} \begin{bmatrix} k_p \\ k_i \end{bmatrix} = \begin{bmatrix} -\cos\phi_{PM} \\ -\sin\phi_{PM} \end{bmatrix} \tag{2.11}$$

Solving equation (2.11) for k_p and k_i gives

$$k_p = -\frac{\cos(\phi_{PM} - \phi_p(\omega_1))}{|G_p(j\omega_1)|} \tag{2.12}$$

$$k_i = \frac{\omega_1 \sin(\phi_{PM} - \phi_p(\omega_1))}{|G_p(j\omega_1)|} \tag{2.13}$$

where $\phi_p(\omega_1)$ and $|G_p(j\omega_1)|$ represent the phase angle and magnitude of the frequency response of the process at the gain crossover frequency.

If a PI controller exists that can achieve both the required gain margin and phase margin then equations (2.9) and (2.12) must be equal as must equations (2.10) and (2.13). Motivated by the forms of equation pairs (2.9), (2.10) and (2.12), (2.13) Fung *et al* then formed two complex functions such that

$$f_1(\omega) = -\frac{\cos\phi_p(\omega)}{GM|G_p(j\omega)|} - j\frac{\omega\sin\phi_p(\omega)}{GM|G_p(j\omega)|} \tag{2.14}$$

$$-\frac{\pi}{2} > \arg(G_p(j\omega)) > -\pi$$

$$f_2(\omega) = -\frac{\cos(\phi_{PM} - \phi_p(\omega))}{|G_p(j\omega)|} + j\frac{\omega\sin(\phi_{PM} - \phi_p(\omega))}{|G_p(j\omega)|} \tag{2.15}$$

$$-\frac{\pi}{2} > \arg(G_p(j\omega)) > -\pi + \phi_{PM}$$

Thus by plotting functions (2.14) and (2.15) on the same complex plane, if there is an intersection, this corresponds to the real and imaginary parts of f_1 and f_2 being equal. Hence the parameters k_p and k_i of the PI controller are equal and can be read directly from the graph. Additionally, if there are multiple solutions then these solutions are all present on the graph as intersections of the $f_1(\omega)$ and $f_2(\omega)$ curves. This allows a

choice to be made as to the particular set of controller parameters that are to be used to implement the controller.

The method of Fung *et al* shall now be used to design a PI controller for the process given by

$$G_p(s) = \frac{(1-0.2s)}{(1+s)^3}$$

The controller shall be of the form

$$G_c(s) = k_p + \frac{k_i}{s}$$

The process has a right-half plane zero and is used to illustrate some of the points of the implementation of the Fung *et al* method. The design requirement is to have a gain margin of 3 and a phase margin of 45°. Figure 2.2 shows the graph resulting from plotting the functions f_1 and f_2. From the graph it can be seen that there are two intersections of the f_1 and f_2 curves. The intersections relate to controller parameters that satisfy the design requirement and are shown in Table 2.1. From Table 2.1 it can be seen that the controller parameters both achieve the design objective. In Astrom and Hagglund (1995) it is shown that the integral of the error, following a disturbance, is minimised by having the largest possible value of integral gain, k_i. Following from this, the particular choice of controller parameters that would be used to implement the design can be based on achieving the desired design requirements and also choosing to minimise the integral of the error such that an improvement in the disturbance rejection properties of the closed loop system is obtained.

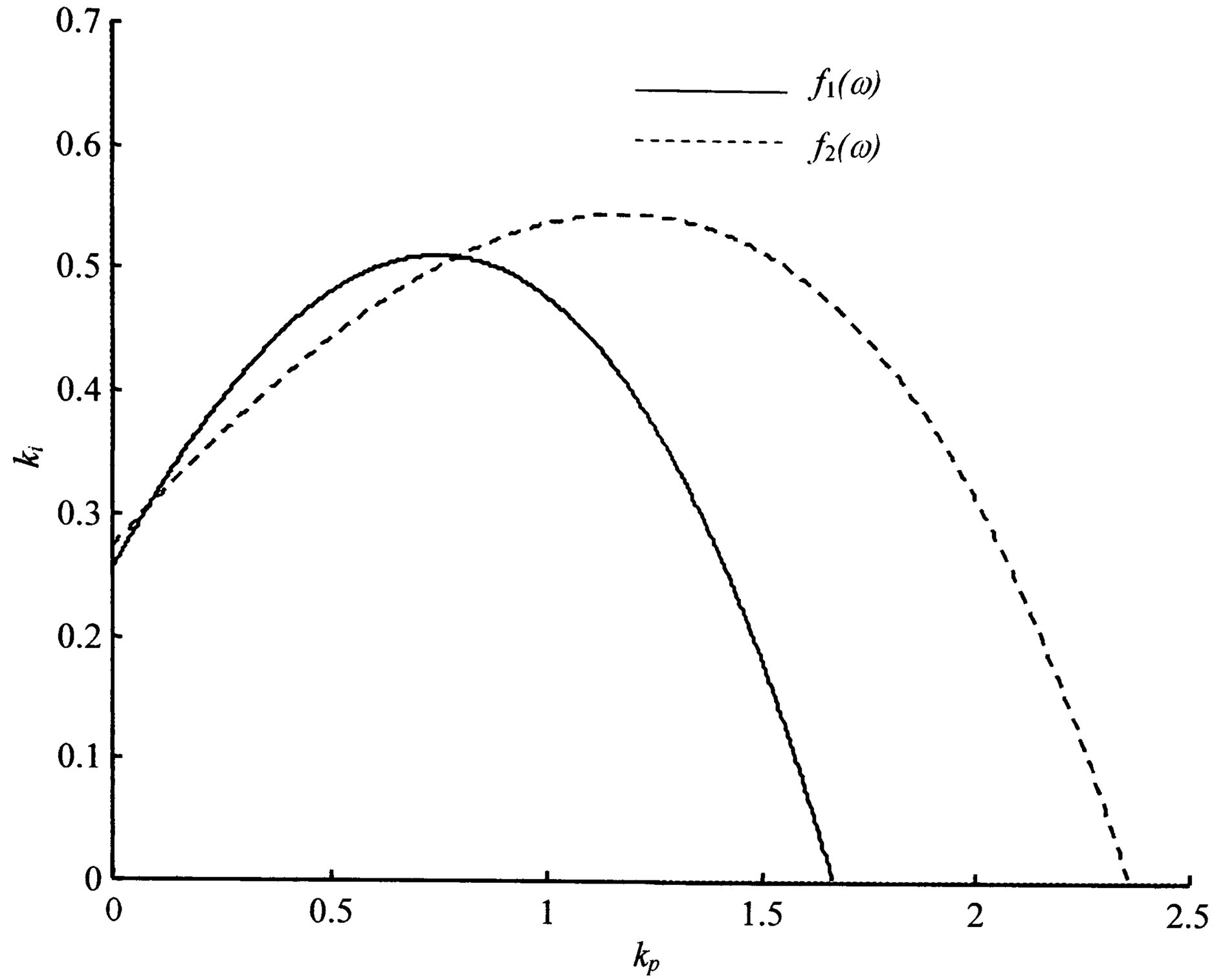


Figure 2.2: Exact Gain Margin and Phase Margin Method

To maximise the size of the integral term for this example, the controller chosen would have the parameters given by Controller 2 of Table 2.1.

Table 2.1: PI Controller Parameters

Process: $G_p(s) = \dfrac{(1-0.2s)}{(1+s)^3}$

Controller 1				**Controller 2**			
k_p		k_i		k_p		k_i	
0.0755		0.3026		0.7914		0.5121	
Gain Margin		Phase Margin (degree)		Gain Margin		Phase Margin (degree)	
Required	Actual	Required	Actual	Required	Actual	Required	Actual
3	3	45	45	3	3	45	45

The gain crossover frequency and phase crossover frequency when using Controller 2 parameters were determined as 0.471 (rad.s^{-1}) and 1.01 (rad.s^{-1}), respectively.

The method of Fung *et al* shows, by means of graphical results, all of the possible PI controller parameters that satisfy the design objective of achieving a specified gain margin and phase margin. The controller designer only knows that a particular gain margin and phase margin design is achievable at the end of the design. Hence a redesign may have to be undertaken, once more with no guarantee that the particular design will be viable. This may not be too much of an inconvenience since a model of the process is required for the method and hence the design is able to be carried out off-line. However, if there is no process model available this represents a major problem. Due to the use of the Phase-Locked Loop identifier it is possible to extend the method of Fung *et al* so that by carrying out non-parametric identifications of the process to be controlled all of the possible viable gain margin and phase margin pairings can be determined. The results shall be given in a graphical form for ease of presentation and use by control practitioners. The significance of this is that an online design method for PI controllers can be implemented allowing the possibility of providing the controller designer with a set of viable gain and phase margin pairings prior to carrying out the design. By this means a choice as to the desired versus the achievable gain and phase margin design specifications can be carried out prior to the final design being attempted.

The tool that is used to carry out the non-parametric identifications utilised by the method is the Phase-Locked Loop identifier that was discussed in Chapter 1. In the method of Fung *et al* a model of the process is assumed to be available to carry out the design method. In the method to be described a non-parametric identification of the process is carried out on-line in closed loop. From the identification data a model is produced that relates frequency to the controller parameters k_p and k_i at that frequency and is calculated using equations (2.9), (2.12), (2.10) and (2.13). Assume that the process to be identified is, $G_p(s)$, then the phase angle range over which the identifications are carried out is defined by

$$-\pi < \arg\left(G_p(j\omega)\right) < -\frac{\pi}{2} + \frac{\pi}{3} \tag{2.16}$$

The number of identifications that are carried out in this phase angle range is user selectable. The trade-off that must be made relates the number of identifications carried out to the accuracy of the resulting gain-phase margin pairing graph produced by the method and obviously the time required to carry out the method. The data determined from the identification step is then split into two sets. One set of data relates to the approximate phase angle range

$$-\pi < \arg\left(G_p(j\omega)\right) < -\frac{\pi}{2} \tag{2.17}$$

and the other to the phase angle range

$$-\pi + \frac{\pi}{3} < \arg\left(G_p(j\omega)\right) < -\frac{\pi}{2} + \frac{\pi}{3} \tag{2.18}$$

The choice of the frequency range is governed by the choice of the maximum value of phase margin that is to be used in the design. In this case the maximum phase margin is 60°, hence the use of $\pi/3$ in (2.18). In the method of Fung *et al* curves relating to the equations

$$f_1(\omega) = k_{p_{-\pi}}(\omega) + jk_{i_{-\pi}}(\omega)$$

and

$$f_2(\omega) = k_{p_1}(\omega) + jk_{i_1}(\omega)$$

are plotted and the intersections, if any, give the solution set for the controller parameters to achieve the required design specification. The new method follows this basic idea. The values of the curves f_1 and f_2 are calculated at the chosen phase angles within the phase angle range (2.17) and (2.18) respectively. This gives a rough estimation of the f_1 - f_2 curves over the required frequency ranges. The curves so obtained are only *accurate* at the specified frequency points and their values are required at intermediate points. This is carried out using cubic-spline interpolation. The method then continues by comparing the data from the f_1 - f_2 curves, over the desired gain margin and phase margin ranges, and recording any intersections. The results of the search are then displayed by means of a graph relating viable gain margin and phase margin pairings. The algorithm for the method is now detailed:

Algorithm 2.1: Viable Gain Margin and Phase Margin Pairings

Step 1: Initialisation.

Select the desired gain margin, *GM*, and phase margin, ϕ_{PM}, ranges.

Select the number of identification points in the range $-\pi < \arg\left(G_p(j\omega)\right) < -\frac{\pi}{2} + \phi_{PM}$

Step 2: Identification and Data Splitting Step.

Identify the process at the required phase angles and record the corresponding frequency and magnitude data.

Split the data to correspond to the frequency ranges $-\pi < \arg\left(G_p(j\omega)\right) < -\frac{\pi}{2}$ and $-\pi + \phi_{PM} < \arg\left(G_p(j\omega)\right) < -\frac{\pi}{2} + \phi_{PM}$ for use in calculating, respectively, the $f_1(\omega)$ and $f_2(\omega)$ curves.

Step 3: Search Step.

Choose a search interval $R_{GM} = \left(GM_{\min}, GM_{\max}\right)$ and $R_{\varphi_{pm}} = \left(\phi_{PM_{\min}}, \phi_{PM_{\max}}\right)$.

Search through the space $R_{GM} \times R_{\phi_{PM}}$ to find locations where $f_1(\omega) = k_{p_{-\pi}}(\omega) + jk_{i_{-\pi}}(\omega)$ is equal to $f_2(\omega) = k_{p_1}(\omega) + jk_{i_1}(\omega)$

Record the locations as the candidate solution gain-phase margin pairs.

●

2.2.2 Case Studies for the Automated Existence Testing of Gain Margin and Phase Margin Pairings

The viable phase and gain margin pairing method shall now be applied to two processes. In the case studies the processes that shall be used are given by

$$G_1(s) = \frac{(1 - 0.2s)}{(1 + s)^3}$$

$$G_2(s) = \frac{1}{(s + 1)^6}$$

The process $G_1(s)$ is non-minimum phase, having a right half plane zero. It is included to provide a link with the example showing how the Fung *et al* method is used earlier in this Chapter. The process $G_2(s)$ is a high order non-oscillatory process and is representative of a large class of system that are met in the process industries.

In the following case studies all of the required identifications are carried out in closed loop using the Phase-Locked Loop identification method.

Case Study 2.1.

The gain margin and phase margin ranges are selected as 2 to 5 for the gain margin and 30° to 60° for the phase margin. The process identifications were selected to be carried out at phase angles in the range -30° to -180° and sixteen equally spaced points on the frequency response curve of the process were identified.

Table 2.2: Identification Results for Case Study 2.1

$G_1(s)=\frac{(1-0.2s)}{(1+s)^3}$		$G_c(s)=1.922+\frac{0.5303}{s}$	
Magnitude	**Phase Angle (rad)**	**Excitation Frequency (rad.s^{-1})**	**Time (s)**
0.9605	-0.5240	0.1647	233
0.9318	-0.6978	0.2213	395
0.8953	-0.8722	0.2791	524
0.8519	-1.0473	0.3386	676
0.8030	-1.2215	0.40	839
0.7498	-1.3960	0.4641	938
0.6930	-1.5704	0.5309	1029
0.6335	-1.7449	0.6016	1123
0.5697	-1.9204	0.6770	1160
0.4953	-2.0943	0.7567	1220
0.4537	-2.2690	0.8431	1311
0.3954	-2.4436	0.9360	1376
0.3381	-2.6179	1.0389	1464
0.2904	-2.7921	1.1497	1538
0.2428	-2.9666	1.2744	1634
0.2013	-3.1414	1.4127	1699

The identifications were carried out in closed loop with the controller tuning parameters being derived from the results of a relay experiment and the application of Ziegler and Nichols ultimate period tuning rules for a PI controller. The results of the identification are shown in Table 2.2. The time taken to carry out the identification of all sixteen points took approximately 28 minutes. This identification time is not considered to be excessive since highly accurate estimates of the process frequency response curve are obtained and the identification was completed in closed loop. Figure 2.3 shows the identified data compared with the actual Nyquist curve for the process $G_1(s)$.

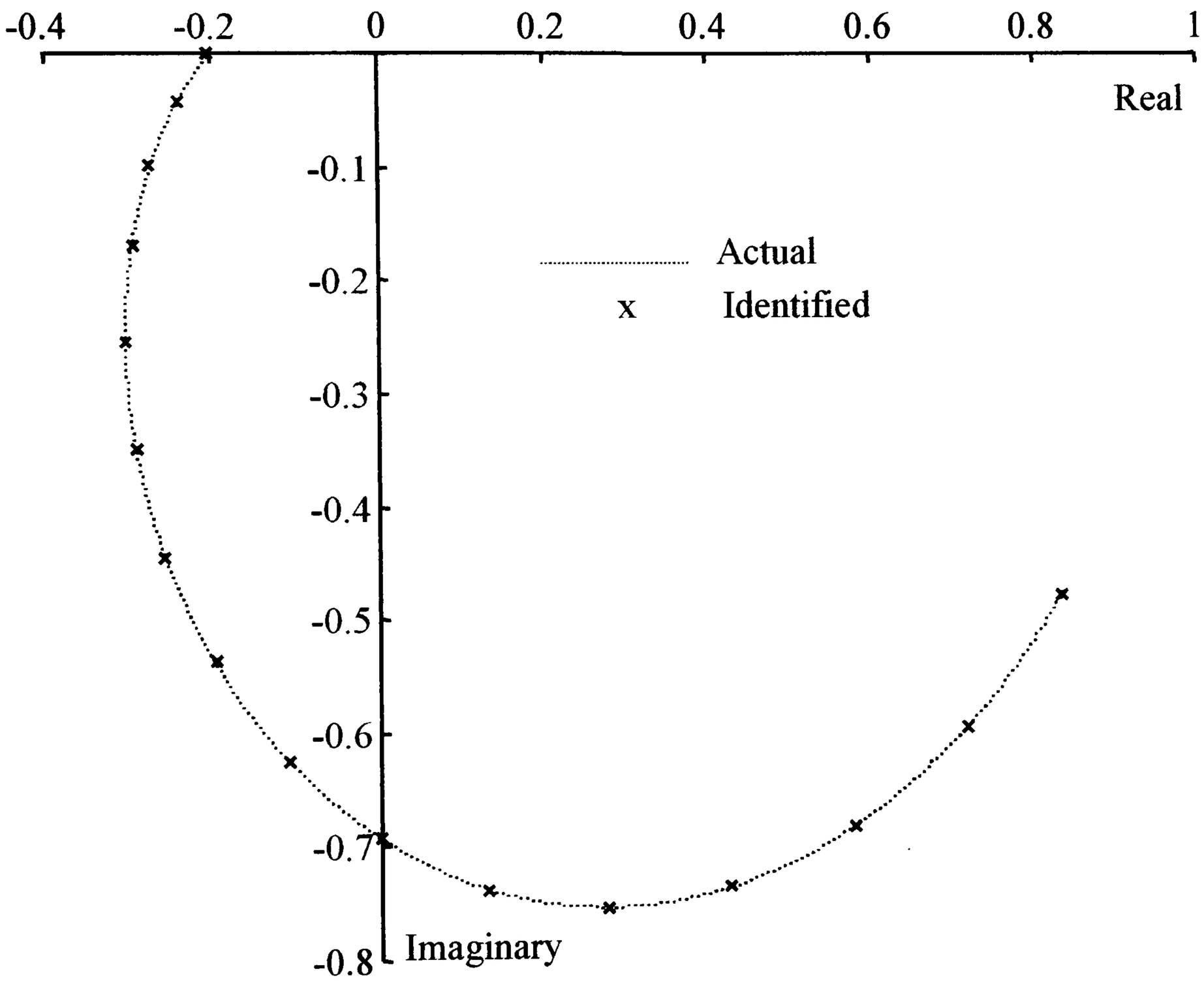


Figure 2.3: Actual and Identified Frequency Response for Process $G_1(s)$.

From Figure 2.3 it can be seen that there is good agreement between the identified and actual frequency response of the process. Figure 2.4 shows the results from the application of the viable gain and phase margin method for the process $G_1(s)$. In Figure 2.4 the viable pairings of gain margin and phase margin are represented by the

shaded region of the graph. From Figure 2.4 it is possible to see that there are certain regions within which it is not possible to design a PI controller such that a viable gain margin and phase margin pairing will be attained. Verification of the method has been carried out in part by using the Fung *et al* method and testing several gain and phase margin pairings to ensure that the Fung *et al* results agree with the results of Algorithm 2.1. The solution of the particular specification used in the previous example (gain margin of 3 and phase margin of 45°) is shown by the intersection of the hairlines in Figure 2.4.

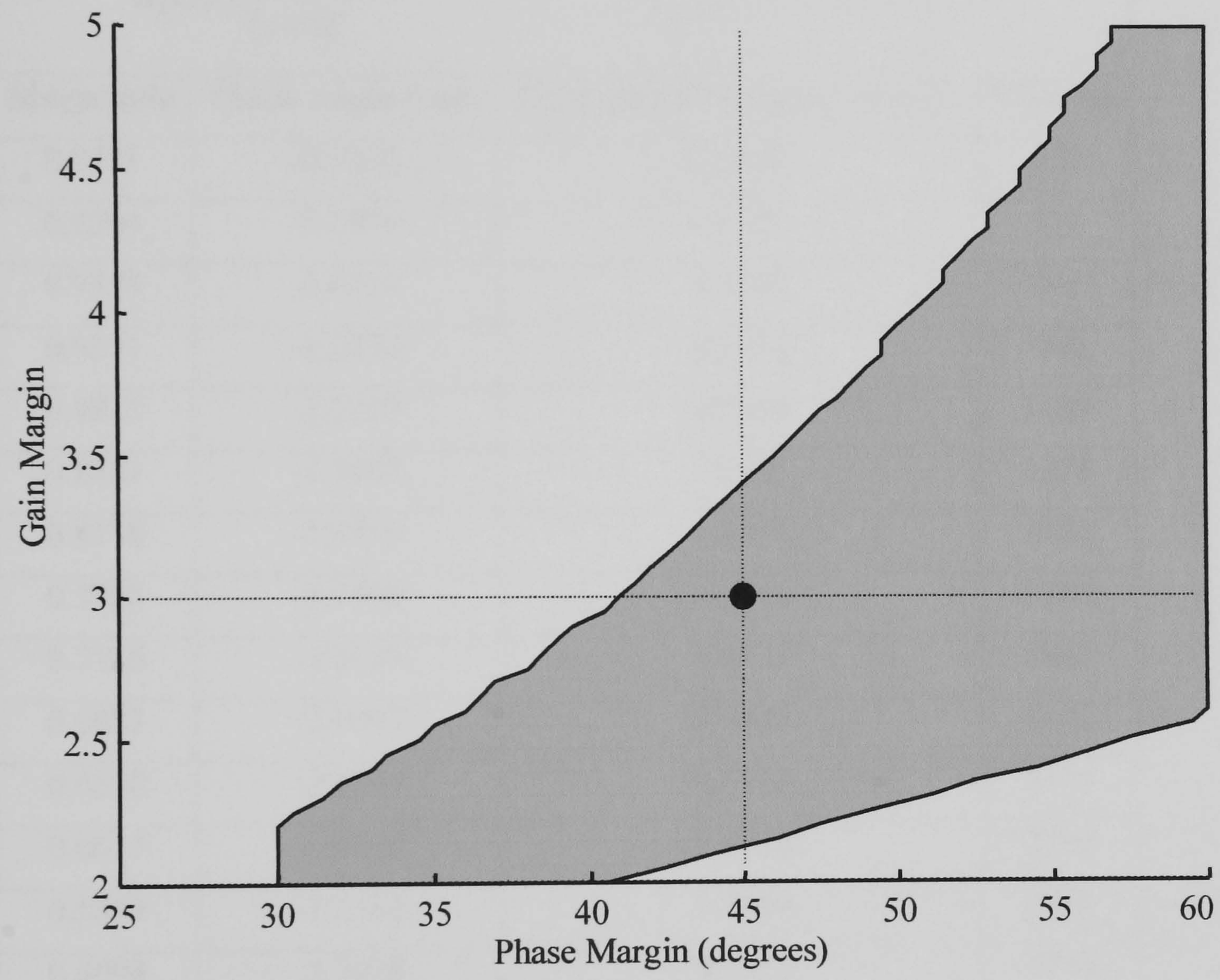


Figure 2.4: Viable Gain and Phase Margin Pairings for $G_1(s)$

Case Study 2.2.

In this case study the process $G_2(s)$ is used in closed loop with a controller the parameters of which were determined from a relay experiment and the application of Ziegler and Nichols ultimate period method rules for a PI controller. The range of the

gain margin and phase margin over which the viable gain margin and phase margin method was applied was 2 to 5 for the gain margin and 30° and 60° for the phase margin. The process identifications were selected to be carried out at phase angles in the range -30° to -180° and sixteen equally spaced points on the frequency response curve of the process were identified. Table 2.3 details the results of the identification section of the application.

Table 2.3: Identification Results for Case Study 2.2

$G_2(s) = \frac{1}{(s+1)^6}$		$G_c(s) = 0.9372 + \frac{0.1077}{s}$	
Magnitude	**Phase Angle (rad)**	**Excitation Frequency (rad.s^{-1})**	**Time (s)**
0.9771	-0.5232	0.0874	399
0.9594	-0.6976	0.1167	534
0.9378	-0.8729	0.1464	743
0.9111	-1.0474	0.1762	996
0.8835	-1.2213	0.2064	1189
0.8481	-1.3963	0.2369	1378
0.8116	-1.5710	0.2679	1583
0.7722	-1.7458	0.29932	1760
0.7326	-1.9199	0.3317	1986
0.6923	-2.0947	0.3639	2151
0.6530	-2.2693	0.3972	2335
0.6017	-2.4430	0.4312	2537
0.5519	-2.6183	0.4664	2635
0.5096	-2.7928	0.5021	2852
0.4658	-2.9666	0.5391	3051
0.4228	-3.1413	0.5762	3171

As can be seen from Table 2.3 the time to carry out the identification was approximately 53 minutes. This time is not considered to be excessive since the identification was carried out in closed loop and the excitation magnitude can be maintained at a relatively low level. The data obtained from the identification gives

an accurate estimate of the frequency response of the system $G_2(s)$ as can be seen from Figure 2.5.

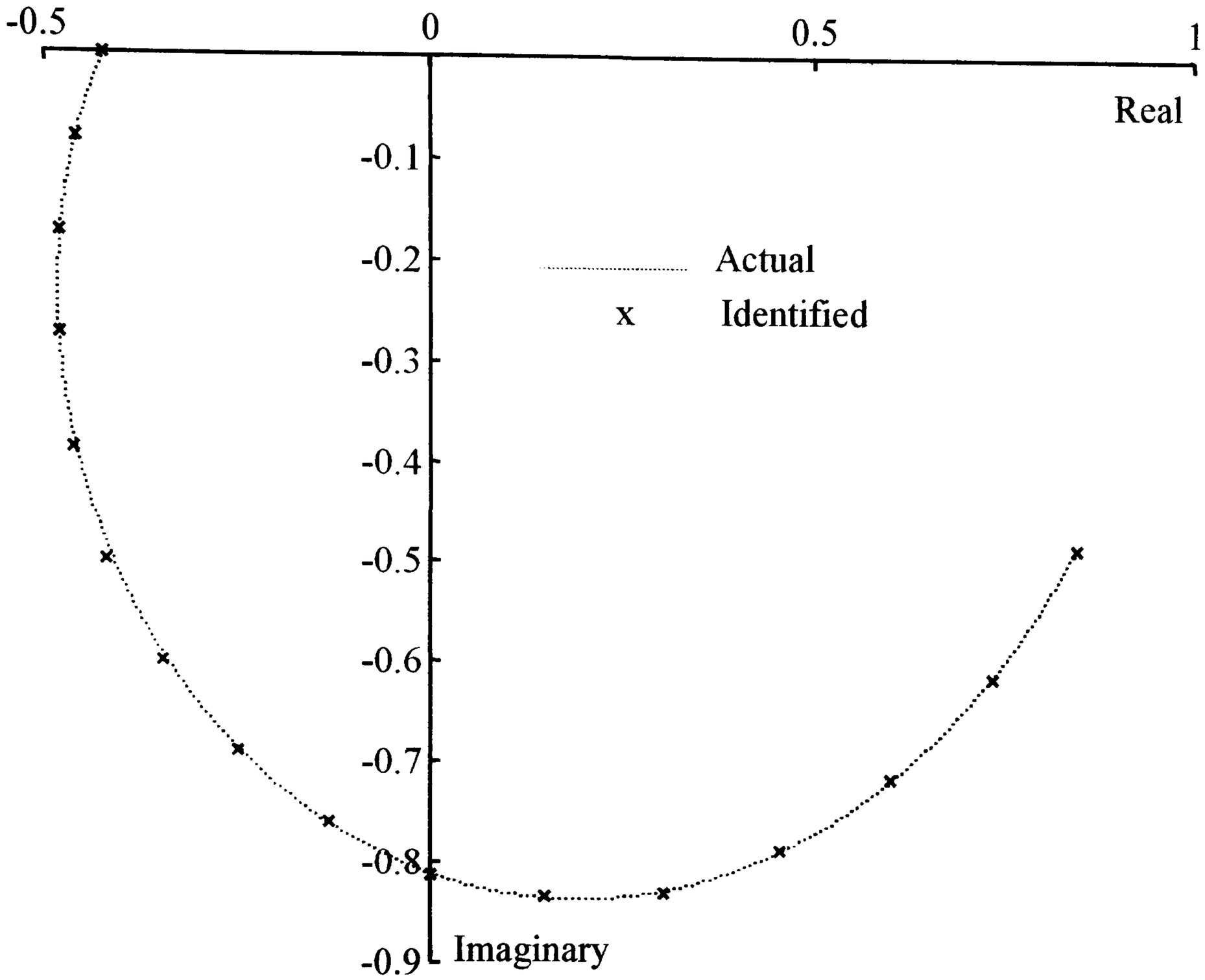


Figure 2.5: Actual and Identified Frequency Response for Process $G_2(s)$.

From the data obtained in the identification phase of the method the viable gain and phase margin pairs were determined. The viable pairings for gain margin and phase margin are represented in Figure 2.6 by the shaded region. Verification of the results shown in Figure 2.6 have been carried out for a number of the viable pairings using the method of Fung *et al* with the results being in good agreement.

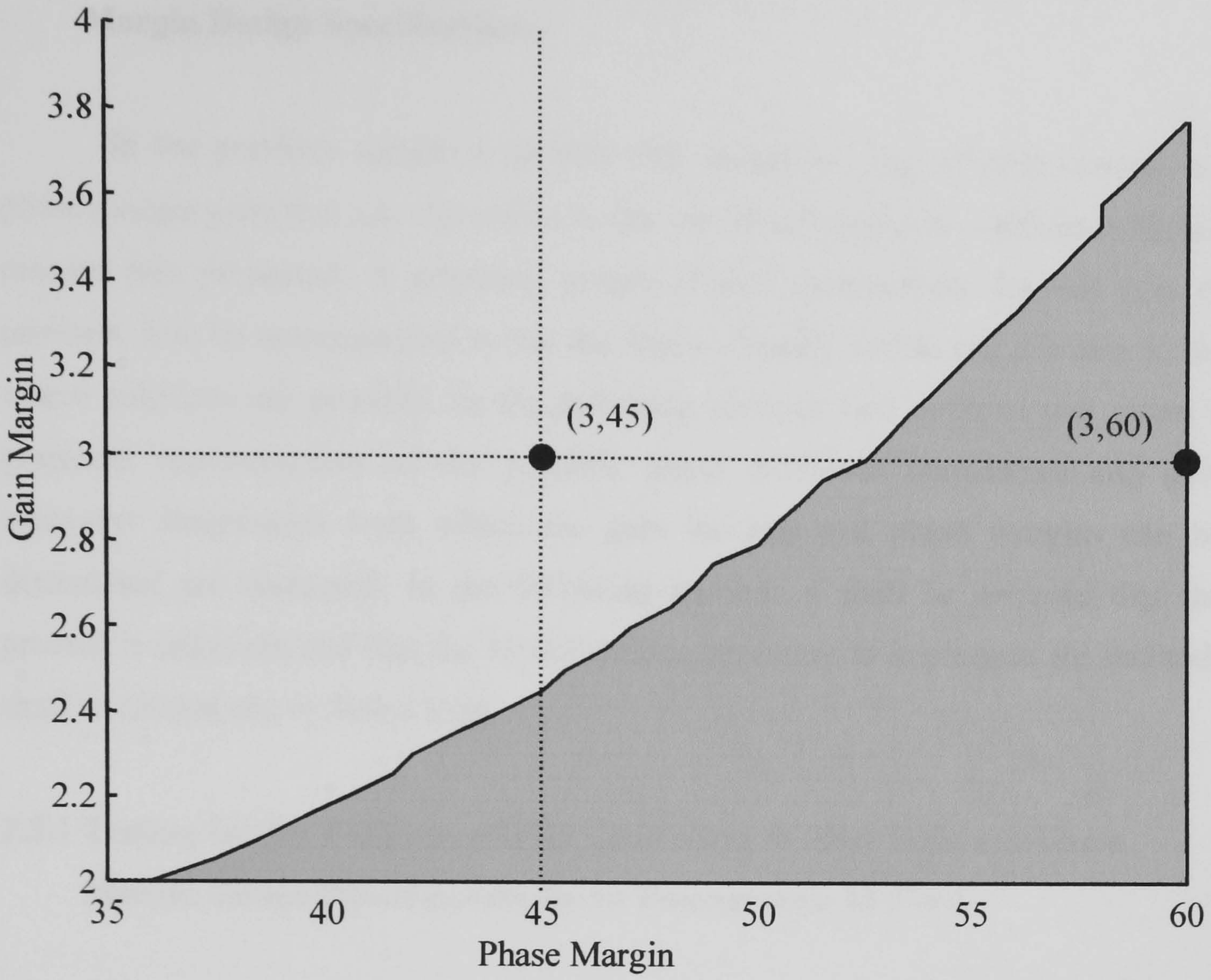


Figure 2.6: Viable Gain and Phase Margin Pairings for $G_2(s)$

As can be seen from Figure 2.6 if a gain margin and phase margin design are attempted for a gain margin of 3 and a phase margin of 45° then no PI controller can be found that will fulfil that requirement. However if the phase margin is increased to 60°, as shown in Figure 2.6, then that design can be achieved by the use of a PI controller.

The benefit of this method is that for an unknown process a map of achievable gain and phase margins can be obtained. With this knowledge, designs will only be carried out for viable gain and phase margin pairings. Additionally the identifications that are carried out are in closed loop and hence cause the least disruption to the process.

2.3 Testing for the Existence of PID Controllers to Meet Gain and Phase Margin Design Specifications.

In the previous section a method that returns the map of gain margin and phase margin pairs that are achievable by the use of a PI controller with an unknown process was presented. A graphical means of data presentation, for this type of problem, is to be recommended in that the data are readily visible and it is easy to see where solutions are possible. In the following sections two methods that return a graphical representation of the possible phase crossover frequencies and gain crossover frequencies from which the gain margins and phase margins can be determined are discussed. In the following sections it shall be assumed that the process is unknown and that the identifications necessary to implement the methods shall be carried out in closed loop.

2.3.1 Testing for the Existence of PID Controllers to Meet Gain and Phase Margin Design Specifications by an Enumeration Method.

In this section the PI controller method due to Fung *et al* (1998) shall first be generalised and then an extension of the generalised method shall be described that can be used to determine the parameters of PID controllers to meet gain margin and phase margin specifications.

In the PI controller tuning method of Fung *et al* (1998) a graphical approach is taken to solving the combined equation set given by (2.8) and (2.11) as

$$\begin{bmatrix} G_{pR}(\omega_{-\pi}) & \dfrac{G_{pI}(\omega_{-\pi})}{\omega_{-\pi}} \\ G_{pI}(\omega_{-\pi}) & -\dfrac{G_{pR}(\omega_{-\pi})}{\omega_{-\pi}} \\ G_{pR}(\omega_1) & \dfrac{G_{pI}(\omega_1)}{\omega_1} \\ G_{pI}(\omega_1) & -\dfrac{G_{pR}(\omega_1)}{\omega_1} \end{bmatrix} \begin{bmatrix} k_p \\ k_i \end{bmatrix} = \begin{bmatrix} \dfrac{-1}{GM} \\ 0 \\ -\cos\phi_{PM} \\ -\sin\phi_{PM} \end{bmatrix} \tag{2.19}$$

Equation (2.19) is derived from the Nyquist geometry of the forward transfer function of the compensated process at the phase crossover point and the gain

crossover point. The process to be controlled is given by $G_p(s)$ and the PI controller has the structure

$$G_c(s) = k_p + \frac{k_i}{s}$$

Solutions for k_p and k_i were found from equation (2.13) such that

$$k_p = \frac{-\cos\phi_p(\omega_{-\pi})}{GM\left|G_p(j\omega_{-\pi})\right|} \tag{2.20a}$$

$$k_i = -\frac{\omega_{-\pi}\sin\phi_p(\omega_{-\pi})}{GM\left|G_p(j\omega_{-\pi})\right|} \tag{2.20b}$$

$$k_p = -\frac{\cos(\phi_{PM} - \phi_p(\omega_1))}{\left|G_p(j\omega_1)\right|} \tag{2.21a}$$

$$k_i = \frac{\omega_1\sin(\phi_{PM} - \phi_p(\omega_1))}{\left|G_p(j\omega_1)\right|} \tag{2.21b}$$

The key observation made by Fung *et al* was that when equations (2.20a) and (2.21a) were equal and when equations (2.20b) and (2.21b) were equal then the PI controller satisfying the given gain margin and phase margin specification had been found. The novel representation and interpretation was given by defining two complex functions such that

$$f_1(\omega_{-\pi}) = K_{1P}(\omega_{-\pi}) + jK_{1I}(\omega_{-\pi}) \qquad f_1 : R^1 \to C$$

$$f_2(\omega_1) = K_{2P}(\omega_1) + jK_{2I}(\omega_1) \qquad f_2 : R^1 \to C$$

with

$$K_{1P}(\omega_{-\pi}) = \frac{-\cos\phi_p(\omega_{-\pi})}{GM\left|G_p(j\omega_{-\pi})\right|} \tag{2.22a}$$

$$K_{1I}(\omega_{-\pi}) = -\frac{\omega_{-\pi}\sin\phi_p(\omega_{-\pi})}{GM\left|G_p(j\omega_{-\pi})\right|} \tag{2.22b}$$

$$K_{2P}(\omega_1) = -\frac{\cos(\phi_{PM} - \phi_p(\omega_1))}{\left|G_p(j\omega_1)\right|} \tag{2.23a}$$

$$K_{2I}(\omega_1) = \frac{\omega_1\sin(\phi_{PM} - \phi_p(\omega_1))}{\left|G_p(j\omega_1)\right|} \tag{2.23b}$$

The complex functions f_1 and f_2 were drawn as curves in the C space. When $f_1(\omega_{-\pi}) = f_2(\omega_1)$ then this intersection was interpreted as a solution point prescribing a potential PI controller gain pair, k_p, k_i which satisfied the given specification.

In the Fung *et al* method the functions $f_1(\omega)$ and $f_2(\omega)$ do not necessarily have to have the form prescribed by equations (2.22a), (2.22b), (2.23a) and (2.23b). Two variations are possible; if equation (2.19) is re-ordered as

$$\begin{bmatrix} G_{pR}(\omega_{-\pi}) & \dfrac{G_{pI}(\omega_{-\pi})}{\omega_{-\pi}} \\ G_{pR}(\omega_1) & \dfrac{G_{pI}(\omega_1)}{\omega_1} \\ G_{pI}(\omega_{-\pi}) & -\dfrac{G_{pR}(\omega_{-\pi})}{\omega_{-\pi}} \\ G_{pI}(\omega_1) & -\dfrac{G_{pR}(\omega_1)}{\omega_1} \end{bmatrix} \begin{bmatrix} k_p \\ k_i \end{bmatrix} = \begin{bmatrix} \dfrac{-1}{GM} \\ -\cos\phi_{PM} \\ 0 \\ -\sin\phi_{PM} \end{bmatrix} \tag{2.24}$$

Then it can be shown that

$$k_{1p} = \frac{\omega_1 G_{pI}(\omega_{-\pi})\cos\phi_{PM} - \dfrac{\omega_{-\pi}}{GM} G_{pI}(\omega_1)}{\omega_{-\pi} G_{pR}(\omega_{-\pi}) G_{pI}(\omega_1) - \omega_1 G_{pR}(\omega_1) G_{pI}(\omega_{-\pi})} \tag{2.25a}$$

$$k_{1i} = \omega_{-\pi}\omega_1 \frac{\dfrac{G_{pR}(\omega_1)}{GM} - G_{pR}(\omega_{-\pi})\cos\phi_{PM}}{\omega_{-\pi} G_{pR}(\omega_{-\pi}) G_{pI}(\omega_1) - \omega_1 G_{pR}(\omega_1) G_{pI}(\omega_{-\pi})} \tag{2.25b}$$

$$k_{2p} = \frac{\omega_1 G_{pR}(\omega_{-\pi})\sin\phi_{PM}}{\omega_{-\pi} G_{pI}(\omega_{-\pi}) G_{pR}(\omega_1) - \omega_1 G_{pR}(\omega_{-\pi}) G_{pI}(\omega_1)} \tag{2.26a}$$

$$k_{2i} = \omega_{-\pi}\omega_1 \frac{G_{pI}(\omega_{-\pi})\sin\phi_{PM}}{\omega_{-\pi} G_{pI}(\omega_{-\pi}) G_{pR}(\omega_1) - \omega_1 G_{pR}(\omega_{-\pi}) G_{pI}(\omega_1)} \tag{2.26b}$$

The particular solutions for k_p and k_i are found by solving equation (2.24) as

$$\begin{bmatrix} A_{1.} \\ A_{3.} \end{bmatrix} \begin{bmatrix} k_p \\ k_i \end{bmatrix} = \begin{bmatrix} \dfrac{-1}{GM} \\ -\cos\phi_{PM} \end{bmatrix} \tag{2.27a}$$

$$\begin{bmatrix} A_{2.} \\ A_{4.} \end{bmatrix} \begin{bmatrix} k_p \\ k_i \end{bmatrix} = \begin{bmatrix} 0 \\ -\sin\phi_{PM} \end{bmatrix} \tag{2.27b}$$

where the $A_{1.}$, $A_{2.}$, $A_{3.}$ and $A_{4.}$ are the row entries of equation (2.19). If now functions are defined such that

$$f_1(\omega_{-\pi},\omega_1)=(k_{1p}(\omega_{-\pi},\omega_1),k_{1i}(\omega_{-\pi},\omega_1))$$
$$=k_{1p}(\omega_{-\pi},\omega_1)+jk_{1i}(\omega_{-\pi},\omega_1)$$
$$f_2(\omega_{-\pi},\omega_1)=(k_{2p}(\omega_{-\pi},\omega_1),k_{2i}(\omega_{-\pi},\omega_1))$$
$$=k_{2p}(\omega_{-\pi},\omega_1)+jk_{2i}(\omega_{-\pi},\omega_1)$$

where $f_i : R\times R\rightarrow C$, i=1,2

The condition for a possible solution can be given by a norm condition that is equivalent to the two functions intersecting.

$$\|f_1(\omega_{-\pi},\omega_1)-f_2(\omega_{-\pi},\omega_1)\|=0 \qquad (2.28)$$

This restatement of the Fung *et al* method leads to an enumeration method whereby the $(\omega_{-\pi},\omega_1)$ space is systematically traversed and the norm condition of equation (2.28) is tested to find possible PI controllers (Crowe and Johnson, 2001b).

The second variation of the method can be found if equation (2.19) is given by

$$\begin{bmatrix} G_{pR}(\omega_{-\pi}) & \dfrac{G_{pI}(\omega_{-\pi})}{\omega_{-\pi}} \\ G_{pI}(\omega_1) & -\dfrac{G_{pR}(\omega_1)}{\omega_1} \\ G_{pI}(\omega_{-\pi}) & -\dfrac{G_{pR}(\omega_{-\pi})}{\omega_{-\pi}} \\ G_{pR}(\omega_1) & \dfrac{G_{pI}(\omega_1)}{\omega_1} \end{bmatrix}\begin{bmatrix} k_p \\ k_i \end{bmatrix}=\begin{bmatrix} \dfrac{-1}{GM} \\ -\sin\phi_{PM} \\ 0 \\ -\cos\phi_{PM} \end{bmatrix} \qquad (2.29)$$

From equation (2.29) it can be shown that

$$k_{3p}=\frac{\dfrac{\omega_{-\pi}}{GM}G_{pR}(\omega_1)+\omega_1 G_{pI}(\omega_{-\pi})\sin\phi_{PM}}{\omega_{-\pi}G_{pI}(\omega_{-\pi})G_{pI}(\omega_1)+\omega_1 G_{pR}(\omega_{-\pi})G_{pR}(\omega_1)} \qquad (2.30a)$$

$$k_{3i}=\omega_{-\pi}\omega_1\frac{\dfrac{G_{pI}(\omega_1)}{GM}-G_{pR}(\omega_{-\pi})\sin\phi_{PM}}{\omega_{-\pi}G_{pI}(\omega_{-\pi})G_{pI}(\omega_1)+\omega_1 G_{pR}(\omega_{-\pi})G_{pR}(\omega_1)} \qquad (2.30b)$$

$$k_{4p} = -\frac{\omega_1 \dfrac{G_{pR}(\omega_{-\pi})}{GM}\cos\phi_{PM}}{\omega_{-\pi}G_{pI}(\omega_{-\pi})G_{pI}(\omega_1)+\omega_1 G_{pR}(\omega_{-\pi})G_{pR}(\omega_1)} \tag{2.31a}$$

$$k_{4i} = -\omega_{-\pi}\omega_1 \frac{G_{pI}(\omega_{-\pi})\cos\phi_{PM}}{\omega_{-\pi}G_{pI}(\omega_{-\pi})G_{pI}(\omega_1)+\omega_1 G_{pR}(\omega_{-\pi})G_{pR}(\omega_1)} \tag{2.31b}$$

The particular solutions for k_p and k_i are found by solving equation (2.29) as

$$\begin{bmatrix} A_{1.} \\ A_{4.} \end{bmatrix}\begin{bmatrix} k_p \\ k_i \end{bmatrix} = \begin{bmatrix} \dfrac{-1}{GM} \\ -\sin\phi_{PM} \end{bmatrix} \tag{2.32a}$$

$$\begin{bmatrix} A_{2.} \\ A_{3.} \end{bmatrix}\begin{bmatrix} k_p \\ k_i \end{bmatrix} = \begin{bmatrix} 0 \\ -\cos\phi_{PM} \end{bmatrix} \tag{2.32b}$$

where again $A_{1.}$, $A_{2..}$, $A_{3..}$ and $A_{4.}$ are the row entries of equation (2.19). If in this case functions are defined such that

$$\begin{aligned} f_3(\omega_{-\pi},\omega_1) &= (k_{3p}(\omega_{-\pi},\omega_1), k_{3i}(\omega_{-\pi},\omega_1)) \\ &= k_{3p}(\omega_{-\pi},\omega_1) + jk_{3i}(\omega_{-\pi},\omega_1) \\ f_4(\omega_{-\pi},\omega_1) &= (k_{4p}(\omega_{-\pi},\omega_1), k_{4i}(\omega_{-\pi},\omega_1)) \\ &= k_{4p}(\omega_{-\pi},\omega_1) + jk_{4i}(\omega_{-\pi},\omega_1) \end{aligned}$$

where $f_i : R \times R \to C$, i=3,4

The condition for a possible solution can be given by a norm condition that is equivalent to the two functions intersecting, viz.

$$\|f_3(\omega_{-\pi},\omega_1) - f_4(\omega_{-\pi},\omega_1)\| = 0 \tag{2.33}$$

The enumeration method described for PI controllers is a generalisation of the method described by Fung *et al* (1998). It is possible to extend the enumeration method to find the parameters of PID controllers to meet specified gain and phase margins (Crowe and Johnson, 2001b). The PID controller that shall be used is given by

$$G_c(s) = k_p + \frac{k_i}{s} + sk_d \tag{2.34}$$

and the process is represented by, $G_p(s)$. In the subsequent analysis the substitution

$$s = j\omega$$

shall be used. Thus the process, $G_p(s)$ can be represented by

$$G_p(j\omega) = G_{pR}(\omega) + jG_{pI}(\omega)$$

At the phase crossover frequency of the forward path of the compensated system, $\omega_{-\pi}$, the design requirement is to ensure that

$$G_p(j\omega_{-\pi})\left(k_p + j\left(\omega_{-\pi}k_d - \frac{k_i}{\omega_{-\pi}}\right)\right) = -\frac{1}{GM} \tag{2.35}$$

where GM is the required gain margin. Representing equation (2.35) by its real and imaginary components gives

$$\begin{bmatrix} G_{pR}(\omega_{-\pi}) & \dfrac{G_{pI}(\omega_{-\pi})}{\omega_{-\pi}} & -\omega_{-\pi}G_{pI}(\omega_{-\pi}) \\ G_{pI}(\omega_{-\pi}) & -\dfrac{G_{pR}(\omega_{-\pi})}{\omega_{-\pi}} & \omega_{-\pi}G_{pR}(\omega_{-\pi}) \end{bmatrix} \begin{bmatrix} k_p \\ k_i \\ k_d \end{bmatrix} = \begin{bmatrix} -\dfrac{1}{GM} \\ 0 \end{bmatrix} \tag{2.36}$$

At the gain crossover frequency of the forward path of the compensated system, ω_1 , the design requirement is to ensure that

$$G_p(j\omega_1)\left(k_p + j\left(\omega_1 k_d - \frac{k_i}{\omega_1}\right)\right) = -\cos\phi_{PM} - j\sin\phi_{PM} \tag{2.37}$$

By considering the real and imaginary components of equation (2.37) the following representation results

$$\begin{bmatrix} G_{pR}(\omega_1) & \dfrac{G_{pI}(\omega_1)}{\omega_1} & -\omega_1 G_{pI}(\omega_1) \\ G_{pI}(\omega_1) & -\dfrac{G_{pR}(\omega_1)}{\omega_1} & \omega_1 G_{pR}(\omega_1) \end{bmatrix} \begin{bmatrix} k_p \\ k_i \\ k_d \end{bmatrix} = \begin{bmatrix} -\cos\phi_{PM} \\ -\sin\phi_{PM} \end{bmatrix} \tag{2.38}$$

Equations (2.30) and (2.32) can be combined to give

$$\begin{bmatrix} G_{pR}(\omega_{-\pi}) & \dfrac{G_{pI}(\omega_{-\pi})}{\omega_{-\pi}} & -\omega_{-\pi}G_{pI}(\omega_{-\pi}) \\ G_{pI}(\omega_{-\pi}) & -\dfrac{G_{pR}(\omega_{-\pi})}{\omega_{-\pi}} & \omega_{-\pi}G_{pR}(\omega_{-\pi}) \\ G_{pR}(\omega_1) & \dfrac{G_{pI}(\omega_1)}{\omega_1} & -\omega_1 G_{pI}(\omega_1) \\ G_{pI}(\omega_1) & -\dfrac{G_{pR}(\omega_1)}{\omega_1} & \omega_1 G_{pR}(\omega_1) \end{bmatrix} \begin{bmatrix} k_p \\ k_i \\ k_d \end{bmatrix} = \begin{bmatrix} -\dfrac{1}{GM} \\ 0 \\ -\cos\phi_{PM} \\ -\sin\phi_{PM} \end{bmatrix} \tag{2.39}$$

With an obvious notation (2.39) is given by

$$A\kappa = Y_D \tag{2.40}$$

Equation (2.39) is under-determined, the unknowns are $(k_p, k_i, k_d, \omega_{-\pi}, \omega_1)$, and hence there are an infinite number of solutions.

The purpose of the PID controller design is to produce a set of controller parameters that stabilise the process under control and also provide a desired gain and phase margin. If equation (2.40) is going to provide a solution to the design problem then the number of unknowns must be reduced. In the enumeration method to follow, the method that is used to reduce the number of unknowns is to traverse the $(\omega_{-\pi}, \omega_1)$ space systematically using different subsets of equation (2.40) thus reducing the number of unknowns to three (k_p, k_i, k_d). The subsets of equation (2.40) and their explicit solutions are given by

$$\begin{bmatrix} G_{pR}(\omega_{-\pi}) & \dfrac{G_{pI}(\omega_{-\pi})}{\omega_{-\pi}} & -\omega_{-\pi}G_{pI}(\omega_{-\pi}) \\ G_{pI}(\omega_{-\pi}) & -\dfrac{G_{pR}(\omega_{-\pi})}{\omega_{-\pi}} & \omega_{-\pi}G_{pR}(\omega_{-\pi}) \\ G_{pR}(\omega_1) & \dfrac{G_{pI}(\omega_1)}{\omega_1} & -\omega_1 G_{pI}(\omega_1) \end{bmatrix} \begin{bmatrix} k_{1p} \\ k_{1i} \\ k_{1d} \end{bmatrix} = \begin{bmatrix} -\dfrac{1}{GM} \\ 0 \\ -\cos\phi_{PM} \end{bmatrix} \tag{2.41a}$$

$$\begin{bmatrix} A_{1.} \\ A_{2.} \\ A_{3.} \end{bmatrix} \begin{bmatrix} k_{1p} \\ k_{1i} \\ k_{1d} \end{bmatrix} = \begin{bmatrix} -\dfrac{1}{GM} \\ 0 \\ -\cos\phi_{PM} \end{bmatrix} \tag{2.41b}$$

The solution of equation (2.41b) is given by

$$k_{1p} = -\frac{\cos\phi_p(\omega_{-\pi})}{GM\left|G_p(j\omega_{-\pi})\right|}$$

$$k_{1i} = \frac{\omega_{-\pi}\cos\phi_p(\omega_{-\pi})\cos\phi_p(\omega_1)\left(1+\dfrac{\omega_1}{\omega_{-\pi}}\tan\phi_p(\omega_{-\pi})\tan\phi_p(\omega_1)\right)}{GM\left|G_p(j\omega_{-\pi})\right|\left(\dfrac{\omega_{-\pi}}{\omega_1}-\dfrac{\omega_1}{\omega_{-\pi}}\right)\sin\phi_p(\omega_1)}$$

$$- \frac{\omega_{-\pi}\cos\phi_{PM}}{\left|G_p(j\omega_1)\right|\left(\frac{\omega_{-\pi}}{\omega_1}-\frac{\omega_1}{\omega_{-\pi}}\right)\sin\phi_p(\omega_1)}$$

$$k_{1d} = \frac{\cos\phi_p(\omega_{-\pi})\cos\phi_p(\omega_1)\left(1+\frac{\omega_{-\pi}}{\omega_1}\tan\phi_p(\omega_{-\pi})\tan\phi_p(\omega_1)\right)}{\omega_{-\pi}GM\left|G_p(j\omega_{-\pi})\right|\left(\frac{\omega_{-\pi}}{\omega_1}-\frac{\omega_1}{\omega_{-\pi}}\right)\sin\phi_p(\omega_1)}$$

$$- \frac{\cos\phi_{PM}}{\omega_{-\pi}\left|G_p(j\omega_1)\right|\left(\frac{\omega_{-\pi}}{\omega_1}-\frac{\omega_1}{\omega_{-\pi}}\right)\sin\phi_p(\omega_1)}$$

$$\begin{bmatrix} G_{pR}(\omega_{-\pi}) & \frac{G_{pI}(\omega_{-\pi})}{\omega_{-\pi}} & -\omega_{-\pi}G_{pI}(\omega_{-\pi}) \\ G_{pI}(\omega_{-\pi}) & -\frac{G_{pR}(\omega_{-\pi})}{\omega_{-\pi}} & \omega_{-\pi}G_{pR}(\omega_{-\pi}) \\ G_{pI}(\omega_1) & -\frac{G_{pR}(\omega_1)}{\omega_1} & \omega_1 G_{pR}(\omega_1) \end{bmatrix} \begin{bmatrix} k_{2p} \\ k_{2i} \\ k_{2d} \end{bmatrix} = \begin{bmatrix} -\frac{1}{GM} \\ 0 \\ -\sin\phi_{PM} \end{bmatrix} \qquad (2.42a)$$

$$\begin{bmatrix} A_{1.} \\ A_{2.} \\ A_{4.} \end{bmatrix} \begin{bmatrix} k_{2p} \\ k_{2i} \\ k_{2d} \end{bmatrix} = \begin{bmatrix} -\frac{1}{GM} \\ 0 \\ -\sin\phi_{PM} \end{bmatrix} \qquad (2.42b)$$

The solution of equation (2.42b) is given by

$$k_{2p} = -\frac{\cos\phi_p(\omega_{-\pi})}{GM\left|G_p(j\omega_{-\pi})\right|}$$

$$k_{2i} = \frac{\omega_{-\pi}\sin\phi_{PM}}{\left|G_p(j\omega_1)\right|\cos\phi_p(\omega_1)\left(\frac{\omega_{-\pi}}{\omega_1}-\frac{\omega_1}{\omega_{-\pi}}\right)}$$

$$+ \frac{\omega_{-\pi}\cos\phi_p(\omega_{-\pi})\left(\frac{\omega_1}{\omega_{-\pi}}\tan\phi_p(\omega_{-\pi})-\tan\phi_p(\omega_1)\right)}{GM\left(\frac{\omega_{-\pi}}{\omega_1}-\frac{\omega_1}{\omega_{-\pi}}\right)\left|G(j\omega_{-\pi})\right|}$$

$$k_{2d} = \frac{\sin\phi_{PM}}{\omega_{-\pi}\left|G_p(j\omega_1)\right|\left(\frac{\omega_{-\pi}}{\omega_1} - \frac{\omega_1}{\omega_{-\pi}}\right)\cos\phi_p(\omega_1)}$$

$$+ \frac{\cos\phi_p(\omega_{-\pi})\left(\frac{\omega_{-\pi}}{\omega_1}\tan\phi_p(\omega_{-\pi}) - \tan\phi_p(\omega_1)\right)}{\omega_{-\pi}GM\left(\frac{\omega_{-\pi}}{\omega_1} - \frac{\omega_1}{\omega_{-\pi}}\right)\left|G(j\omega_{-\pi})\right|}$$

$$\begin{bmatrix} G_{pR}(\omega_{-\pi}) & \frac{G_{pI}(\omega_{-\pi})}{\omega_{-\pi}} & -\omega_{-\pi}G_{pI}(\omega_{-\pi}) \\ G_{pR}(\omega_1) & \frac{G_{pI}(\omega_1)}{\omega_1} & -\omega_1 G_{pI}(\omega_1) \\ G_{pI}(\omega_1) & -\frac{G_{pR}(\omega_1)}{\omega_1} & \omega_1 G_{pR}(\omega_1) \end{bmatrix} \begin{bmatrix} k_{3p} \\ k_{3i} \\ k_{3d} \end{bmatrix} = \begin{bmatrix} -\frac{1}{GM} \\ -\cos\phi_{PM} \\ -\sin\phi_{PM} \end{bmatrix} \quad (2.43a)$$

$$\begin{bmatrix} A_{1.} \\ A_{3.} \\ A_{4.} \end{bmatrix} \begin{bmatrix} k_{3p} \\ k_{3i} \\ k_{3d} \end{bmatrix} = \begin{bmatrix} -\frac{1}{GM} \\ -\cos\phi_{PM} \\ -\sin\phi_{PM} \end{bmatrix} \quad (2.43b)$$

The solution of equation (2.43b) is given by

$$k_{3p} = -\frac{\cos(\phi(\omega_1) - \phi_{PM})}{\left|G_p(j\omega_1)\right|}$$

$$k_{3i} = \frac{\omega_1}{GM\left|G_p(j\omega_{-\pi})\right|\left(\frac{\omega_{-\pi}}{\omega_1} - \frac{\omega_1}{\omega_{-\pi}}\right)\sin\phi_p(\omega_{-\pi})}$$

$$- \frac{\left(\omega_1\cos(\phi_p(\omega_1) - \phi_{PM}) + \omega_{-\pi}\sin(\phi_p(\omega_1) - \phi_{PM})\tan\phi_p(\omega_{-\pi})\right)}{\left|G_p(j\omega_1)\right|\left(\frac{\omega_{-\pi}}{\omega_1} - \frac{\omega_1}{\omega_{-\pi}}\right)\tan\phi_p(\omega_{-\pi})}$$

$$k_{3d} = \frac{1}{\omega_1 GM\left|G_p(j\omega_{-\pi})\right|\left(\frac{\omega_{-\pi}}{\omega_1} - \frac{\omega_1}{\omega_{-\pi}}\right)\sin\phi_p(\omega_{-\pi})}$$

$$- \frac{\left(\omega_{-\pi}\cos(\phi_p(\omega_1) - \phi_{PM}) + \omega_1\sin(\phi_p(\omega_1) - \phi_{PM})\tan\phi_p(\omega_{-\pi})\right)}{\omega_{-\pi}\omega_1\left|G_p(j\omega_1)\right|\left(\frac{\omega_{-\pi}}{\omega_1} - \frac{\omega_1}{\omega_{-\pi}}\right)\tan\phi_p(\omega_{-\pi})}$$

$$\begin{bmatrix} G_{pI}(\omega_{-\pi}) & -\dfrac{G_{pR}(\omega_{-\pi})}{\omega_{-\pi}} & \omega_{-\pi}G_{pR}(\omega_{-\pi}) \\ G_{pR}(\omega_1) & \dfrac{G_{pI}(\omega_1)}{\omega_1} & -\omega_1 G_{pI}(\omega_1) \\ G_{pI}(\omega_1) & -\dfrac{G_{pR}(\omega_1)}{\omega_1} & \omega_1 G_{pR}(\omega_1) \end{bmatrix} \begin{bmatrix} k_{4p} \\ k_{4i} \\ k_{4d} \end{bmatrix} = \begin{bmatrix} 0 \\ -\cos\phi_{PM} \\ -\sin\phi_{PM} \end{bmatrix} \tag{2.44a}$$

$$\begin{bmatrix} A_{2.} \\ A_{3.} \\ A_{4.} \end{bmatrix} \begin{bmatrix} k_{4p} \\ k_{4i} \\ k_{4d} \end{bmatrix} = \begin{bmatrix} 0 \\ -\cos\phi_{PM} \\ -\sin\phi_{PM} \end{bmatrix} \tag{2.44b}$$

The solution of equation (2.44b) is given by

$$k_{4p} = -\frac{\cos(\phi_p(\omega_1) - \phi_{PM})}{|G_p(j\omega_1)|}$$

$$k_{4i} = \frac{\omega_{-\pi}\cos\phi_{PM}\left(\dfrac{\omega_1\tan\phi_p(\omega_{-\pi})}{\omega_{-\pi}\tan\phi_p(\omega_1)} - 1\right)\sin\phi_p(\omega_1)}{|G_p(j\omega_1)|\left(\dfrac{\omega_{-\pi}}{\omega_1} - \dfrac{\omega_1}{\omega_{-\pi}}\right)} + \frac{\omega_{-\pi}\sin\phi_{PM}\left(1 + \dfrac{\omega_1\tan\phi_p(\omega_{-\pi})\tan\phi_p(\omega_1)}{\omega_{-\pi}}\right)\cos\phi_p(\omega_1)}{|G_p(j\omega_1)|\left(\dfrac{\omega_{-\pi}}{\omega_1} - \dfrac{\omega_1}{\omega_{-\pi}}\right)}$$

$$k_{4d} = \frac{\cos\phi_{PM}\left(\dfrac{\omega_{-\pi}\tan\phi_p(\omega_{-\pi})}{\omega_1\tan\phi_p(\omega_1)} - 1\right)\sin\phi_p(\omega_1)}{\omega_{-\pi}|G_p(j\omega_1)|\left(\dfrac{\omega_{-\pi}}{\omega_1} - \dfrac{\omega_1}{\omega_{-\pi}}\right)} + \frac{\sin\phi_{PM}\left(1 + \dfrac{\omega_{-\pi}\tan\phi_p(\omega_{-\pi})\tan\phi_p(\omega_1)}{\omega_1}\right)\cos\phi_p(\omega_1)}{\omega_{-\pi}|G_p(j\omega_1)|\left(\dfrac{\omega_{-\pi}}{\omega_1} - \dfrac{\omega_1}{\omega_{-\pi}}\right)}$$

The enumeration approach is formulated as the following algorithm.

Algorithm 2.2: Enumeration Search Method

Step 1: Design Specification.

Select the desired gain margin, GM, and phase margin, ϕ_{PM}, for the control design.

Compute $Y_D = \begin{bmatrix} -\dfrac{1}{GM} & 0 & -\cos\phi_{PM} & -\sin\phi_{PM} \end{bmatrix}^T$

Step 2: Solution Step.

Solve equation (2.41a) for explicit formulas for k_{1p}, k_{1i} and k_{1d}.

Solve equation (2.42a) for explicit formulas for k_{2p}, k_{2i} and k_{2d}.

Solve equation (2.43a) for explicit formulas for k_{3p}, k_{3i} and k_{3d}.

Solve equation (2.44a) for explicit formulas for k_{4p}, k_{4i} and k_{4d}.

Step 3: Search Step.

Choose a search interval $R_{-\pi} = (\omega_{\min}, \omega_{\max})$ and $R_1 = (\omega_{\min 1}, \omega_{\max 1})$.

Search through the space $R_{-\pi} \times R_1$ to find locations where

$$-\frac{\cos(\phi_p(\omega_1) - \phi_{PM})}{|G_p(j\omega_1)|} = -\frac{\cos\phi(\omega_{-\pi})}{GM|G_p(j\omega_{-\pi})|}$$

Record the locations as the candidate solution frequencies.

Step 4: PID Parameter Evaluation.

Substitute the candidate solution frequencies into the equations of Step 2 and record the controller parameters. ●

It is easily seen that the enumeration method is a generalisation of the procedure introduced by Tan *et al* (1999). The algorithm that forms the enumeration method returns the parameters of a PID controller to meet a specified gain margin and phase margin. A proof shall now be given that shows that the PID controller parameters found in Step 4 of the algorithm when coupled with the candidate solution frequencies found in Step 3 of the algorithm do in fact allow a solution of equation (2.39).

Theorem 2.1: PID Controller Parameters to give a Desired Gain Margin and Phase Margin.

Assuming that a specified gain margin and phase margin are achievable for a particular process that is to be controlled by the use of a PID controller and if two frequencies exist, $\omega_{-\pi}$ and ω_1, such that

$$-\frac{\cos(\phi_p(\omega_1)-\phi_{PM})}{|G_p(j\omega_1)|}=-\frac{\cos\phi_p(\omega_{-\pi})}{GM|G_p(j\omega_{-\pi})|} \tag{2.45}$$

then the controller parameters found from the solutions of

$$\begin{bmatrix} G_{pR}(\omega_{-\pi}) & \frac{G_{pI}(\omega_{-\pi})}{\omega_{-\pi}} & -\omega_{-\pi}G_{pI}(\omega_{-\pi}) \\ G_{pI}(\omega_{-\pi}) & -\frac{G_{pR}(\omega_{-\pi})}{\omega_{-\pi}} & \omega_{-\pi}G_{pR}(\omega_{-\pi}) \\ G_{pR}(\omega_1) & \frac{G_{pI}(\omega_1)}{\omega_1} & -\omega_1 G_{pI}(\omega_1) \end{bmatrix}\begin{bmatrix} k_p \\ k_i \\ k_d \end{bmatrix}=\begin{bmatrix} -\frac{1}{GM} \\ 0 \\ -\cos\phi_{PM} \end{bmatrix}, \text{ or}$$

$$\begin{bmatrix} G_{pR}(\omega_{-\pi}) & \frac{G_{pI}(\omega_{-\pi})}{\omega_{-\pi}} & -\omega_{-\pi}G_{pI}(\omega_{-\pi}) \\ G_{pI}(\omega_{-\pi}) & -\frac{G_{pR}(\omega_{-\pi})}{\omega_{-\pi}} & \omega_{-\pi}G_{pR}(\omega_{-\pi}) \\ G_{pI}(\omega_1) & -\frac{G_{pR}(\omega_1)}{\omega_1} & \omega_1 G_{pR}(\omega_1) \end{bmatrix}\begin{bmatrix} k_p \\ k_i \\ k_d \end{bmatrix}=\begin{bmatrix} -\frac{1}{GM} \\ 0 \\ -\sin\phi_{PM} \end{bmatrix}, \text{ or}$$

$$\begin{bmatrix} G_{pR}(\omega_{-\pi}) & \frac{G_{pI}(\omega_{-\pi})}{\omega_{-\pi}} & -\omega_{-\pi}G_{pI}(\omega_{-\pi}) \\ G_{pR}(\omega_1) & \frac{G_{pI}(\omega_1)}{\omega_1} & -\omega_1 G_{pI}(\omega_1) \\ G_{pI}(\omega_1) & -\frac{G_{pR}(\omega_1)}{\omega_1} & \omega_1 G_{pR}(\omega_1) \end{bmatrix}\begin{bmatrix} k_p \\ k_i \\ k_d \end{bmatrix}=\begin{bmatrix} -\frac{1}{GM} \\ -\cos\phi_{PM} \\ -\sin\phi_{PM} \end{bmatrix}, \text{ or}$$

$$\begin{bmatrix} G_{pI}(\omega_{-\pi}) & -\dfrac{G_{pR}(\omega_{-\pi})}{\omega_{-\pi}} & \omega_{-\pi}G_{pR}(\omega_{-\pi}) \\ G_{pR}(\omega_1) & \dfrac{G_{pI}(\omega_1)}{\omega_1} & -\omega_1 G_{pI}(\omega_1) \\ G_{pI}(\omega_1) & -\dfrac{G_{pR}(\omega_1)}{\omega_1} & \omega_1 G_{pR}(\omega_1) \end{bmatrix} \begin{bmatrix} k_p \\ k_i \\ k_d \end{bmatrix} = \begin{bmatrix} 0 \\ -\cos\phi_{PM} \\ -\sin\phi_{PM} \end{bmatrix}$$

allied with the frequencies $\omega_{-\pi}$ and ω_1 shall provide a solution to

$$\begin{bmatrix} G_{pR}(\omega_{-\pi}) & \dfrac{G_{pI}(\omega_{-\pi})}{\omega_{-\pi}} & -\omega_{-\pi}G_{pI}(\omega_{-\pi}) \\ G_{pI}(\omega_{-\pi}) & -\dfrac{G_{pR}(\omega_{-\pi})}{\omega_{-\pi}} & \omega_{-\pi}G_{pR}(\omega_{-\pi}) \\ G_{pR}(\omega_1) & \dfrac{G_{pI}(\omega_1)}{\omega_1} & -\omega_1 G_{pI}(\omega_1) \\ G_{pI}(\omega_1) & -\dfrac{G_{pR}(\omega_1)}{\omega_1} & \omega_1 G_{pR}(\omega_1) \end{bmatrix} \begin{bmatrix} k_p \\ k_i \\ k_d \end{bmatrix} = \begin{bmatrix} -\dfrac{1}{GM} \\ 0 \\ -\cos\phi_{PM} \\ -\sin\phi_{PM} \end{bmatrix} \tag{2.46}$$

Proof

Partition equation (2.46) as

$$\left[\begin{array}{ccc} G_{pR}(\omega_{-\pi}) & \dfrac{G_{pI}(\omega_{-\pi})}{\omega_{-\pi}} & -\omega_{-\pi}G_{pI}(\omega_{-\pi}) \\ G_{pI}(\omega_{-\pi}) & -\dfrac{G_{pR}(\omega_{-\pi})}{\omega_{-\pi}} & \omega_{-\pi}G_{pR}(\omega_{-\pi}) \\ G_{pR}(\omega_1) & \dfrac{G_{pI}(\omega_1)}{\omega_1} & -\omega_1 G_{pI}(\omega_1) \\ \hline G_{pI}(\omega_1) & -\dfrac{G_{pR}(\omega_1)}{\omega_1} & \omega_1 G_{pR}(\omega_1) \end{array}\right] \begin{bmatrix} k_p \\ k_i \\ k_d \end{bmatrix} = \left[\begin{array}{c} -\dfrac{1}{GM} \\ 0 \\ -\cos\phi_{PM} \\ \hline -\sin\phi_{PM} \end{array}\right] \tag{2.47}$$

If $\begin{bmatrix} G_{pR}(\omega_{-\pi}) & \dfrac{G_{pI}(\omega_{-\pi})}{\omega_{-\pi}} & -\omega_{-\pi}G_{pI}(\omega_{-\pi}) \\ G_{pI}(\omega_{-\pi}) & -\dfrac{G_{pR}(\omega_{-\pi})}{\omega_{-\pi}} & \omega_{-\pi}G_{pR}(\omega_{-\pi}) \\ G_{pR}(\omega_1) & \dfrac{G_{pI}(\omega_1)}{\omega_1} & -\omega_1 G_{pI}(\omega_1) \end{bmatrix}$, is non-singular for any chosen $\omega_{-\pi}$ and ω_1 then

$$\begin{bmatrix} \tilde{k}_p \\ \tilde{k}_i \\ \tilde{k}_d \end{bmatrix} = \begin{bmatrix} G_{pR}(\omega_{-\pi}) & \dfrac{G_{pI}(\omega_{-\pi})}{\omega_{-\pi}} & -\omega_{-\pi}G_{pI}(\omega_{-\pi}) \\ G_{pI}(\omega_{-\pi}) & -\dfrac{G_{pR}(\omega_{-\pi})}{\omega_{-\pi}} & \omega_{-\pi}G_{pR}(\omega_{-\pi}) \\ G_{pR}(\omega_1) & \dfrac{G_{pI}(\omega_1)}{\omega_1} & -\omega_1 G_{pI}(\omega_1) \end{bmatrix}^{-1} \begin{bmatrix} -\dfrac{1}{GM} \\ 0 \\ -\cos\phi_{PM} \end{bmatrix}$$

Hence, if

$$\begin{bmatrix} G_{pI}(\omega_1) & -\dfrac{G_{pR}(\omega_1)}{\omega_1} & \omega_1 G_{pR}(\omega_1) \end{bmatrix} \begin{bmatrix} \tilde{k}_p \\ \tilde{k}_i \\ \tilde{k}_d \end{bmatrix} = -\sin\phi_{PM} \tag{2.48}$$

then a solution to equation (2.46) has been found. It can be shown that

$$\begin{bmatrix} G_{pI}(\omega_1) & -\dfrac{G_{pR}(\omega_1)}{\omega_1} & \omega_1 G_{pR}(\omega_1) \end{bmatrix} \begin{bmatrix} \tilde{k}_p \\ \tilde{k}_i \\ \tilde{k}_d \end{bmatrix}$$

$$= \frac{\cos\phi(\omega_1)\cos\phi_{PM}}{\sin\phi(\omega_1)} - \frac{|G_p(j\omega_1)|\cos\phi(\omega_{-\pi})}{GM|G_p(j\omega_{-\pi})|\sin\phi(\omega_1)} \tag{2.49}$$

$$= \frac{GM|G_p(j\omega_{-\pi})|\cos\phi(\omega_1)\cos\phi_{PM} - |G_p(j\omega_1)|\cos\phi(\omega_{-\pi})}{GM|G_p(j\omega_{-\pi})|\sin\phi(\omega_1)}$$

$$+ \frac{GM|G_p(j\omega_{-\pi})|\sin\phi(\omega_1)\sin\phi_{PM} - GM|G_p(j\omega_{-\pi})|\sin\phi(\omega_1)\sin\phi_{PM}}{GM|G_p(j\omega_{-\pi})|\sin\phi(\omega_1)}$$

$$= \frac{GM|G_p(j\omega_{-\pi})|\cos(\phi(\omega_1)-\phi_{PM}) - |G_p(j\omega_1)|\cos\phi(\omega_{-\pi})}{GM|G_p(j\omega_{-\pi})|\sin\phi(\omega_1)} - \sin\phi_{PM} \tag{2.50}$$

From the equality

$$-\frac{\cos(\phi_p(\omega_1)-\phi_{PM})}{|G_p(j\omega_1)|} = -\frac{\cos\phi_p(\omega_{-\pi})}{GM|G_p(j\omega_{-\pi})|}$$

it follows that

$$GM|G_p(j\omega_{-\pi})|\cos(\phi_p(\omega_1)-\phi_{PM}) = |G_p(j\omega_1)|\cos\phi_p(\omega_{-\pi})$$

and hence

$$GM\left|G_p(j\omega_{-\pi})\right|\cos(\phi_p(\omega_1)-\phi_{PM})-\left|G_p(j\omega_1)\right|\cos\phi_p(\omega_{-\pi})=0 \qquad (2.51)$$

substitution of equation (2.51) into equation (2.50) results in

$$\frac{GM\left|G_p(j\omega_{-\pi})\right|\cos(\phi(\omega_1)-\phi_{PM})-\left|G_p(j\omega_1)\right|\cos\phi(\omega_{-\pi})}{GM\left|G_p(j\omega_{-\pi})\right|\sin\phi(\omega_1)}-\sin\phi_{PM}$$

$$=-\sin\phi_{PM} \qquad (2.52)$$

The same result, equation (2.45), is achieved regardless of the particular partition chosen, viz. equation, (2.41a), (2.42a), (2.43a) or (2.44a) can be used in the proof.

●

Comment

i) This result, equation (2.45), is stated in Tan *et al* (1999) although no proof is given.

2.3.2 Case Studies for the Design of a PID Controller by an Enumeration Method

The enumeration method shall now be utilised in the design of a PID controller given by

$$G_c(s)=k_p+\frac{k_i}{s}+sk_d$$

The controller shall be designed such that a gain margin of 3 and a phase margin of 60° shall be obtained for the candidate processes given by

$$G_3(s)=\frac{e^{-s}}{(s+1)(s+3)^2}$$

$$G_4(s)=\frac{e^{-2s}}{(s^2+s+5)(s+1)}$$

The process $G_3(s)$ is used to represent a large class of processes that are to be found in the process control industry. An oscillatory process with a time delay is represented by the process $G_4(s)$. All of the identifications were carried out with the

system in closed loop with a known controller. The identifications were carried out using the Phase-Locked Loop identifier method.

The PID controller, $G_c(s)$, can provide a maximum phase lag of $-\frac{\pi}{2}$ (rad) and a maximum phase lead of $\frac{\pi}{2}$(rad). If the phase margin to be designed is given by ϕ_{PM} then the search for a candidate controller should start from the frequency at which

$$\phi_p(\omega_{\min}) = -\frac{\pi}{2} + \phi_{PM} \tag{2.53}$$

Similarly the frequency at which the search for a candidate controller should stop is given by

$$\phi_p(\omega_{\max}) = -\frac{3\pi}{2} + \phi_{PM} \tag{2.54}$$

Thus the frequency range over which the search for a candidate controller will be carried out is found by identifying the process $G_p(s)$ such that equations (2.53) and (2.54) are true.

Case Study 2.3

The process is that given by $G_3(s)$ and the design gain margin and phase margin are to be 3 and 60° respectively. The frequency ranges for both R_1 and $R_{-\pi}$ are given by

$$R_{-\pi} = R_1 = (\omega_{\min}, \omega_{\max})$$

where $\omega_{\min}$ is 0.1973 (rad.s^{-1}) and $\omega_{\max}$ is 1.6407 (rad.s^{-1}). The results from the application of the enumeration method are shown in Figure 2.7. The form of the graph shown in Figure 2.7 requires an explanation. Figure 2.7 represents a contour map of the 1-Norm

$$L = |Y_4 - \sin\phi_{PM}|$$

where Y_4 is calculated using equation (2.48). The graph is generated by applying the rule that if a 1-Norm calculated at a particular pair of enumeration frequencies is

greater than a preset tolerance, in this case 0.001, then the 1-Norm is set equal to the preset tolerance.

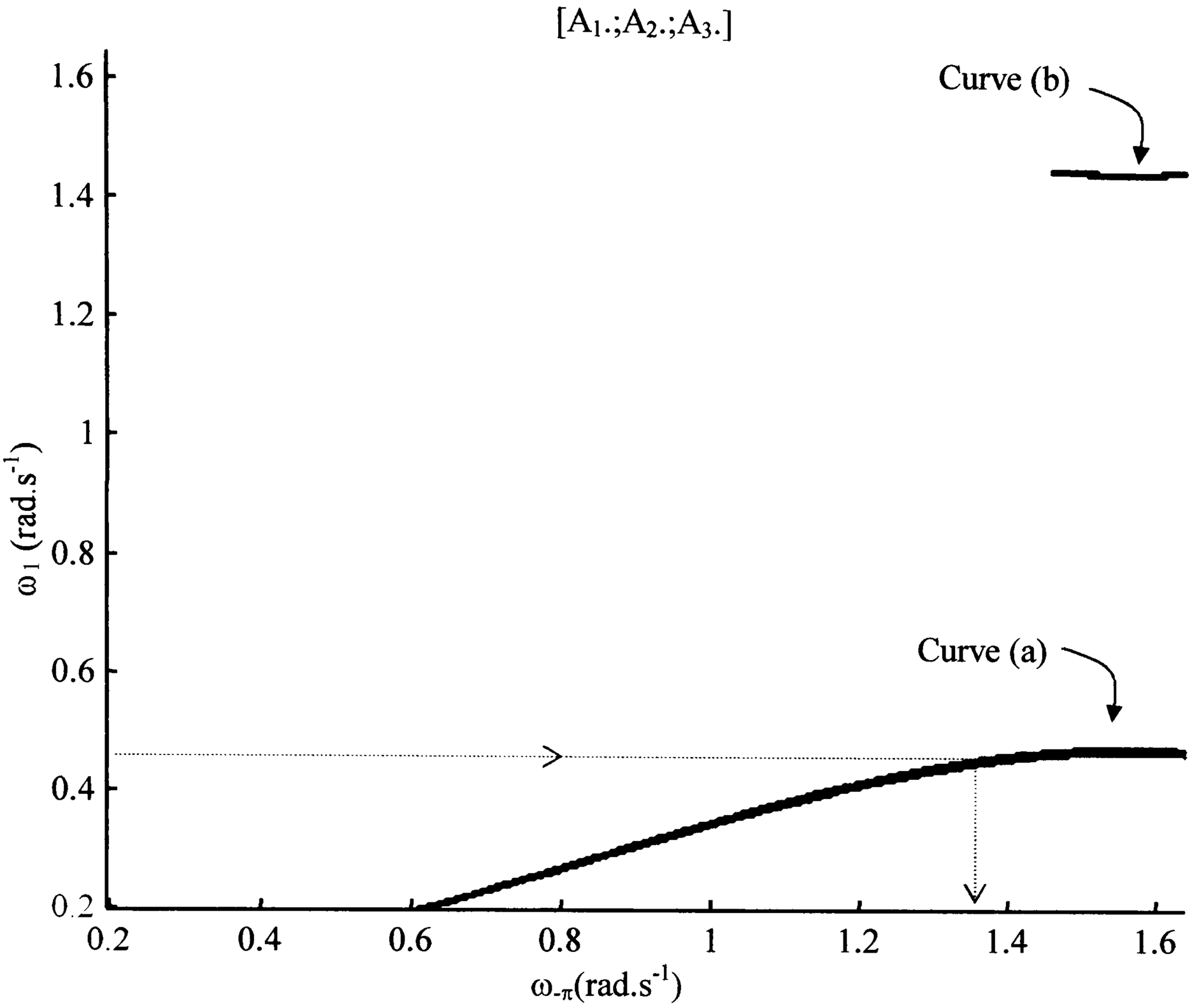


Figure 2.7: Candidate frequencies for controller parameter calculation.

A contour map is then generated for the calculated 1-Norm values. Hence Figure 2.7 gives the frequencies at which it is possible to calculate the parameters of a PID controller that will ensure that the specified design is met. To illustrate the enumeration method the results of a choice of frequencies where,

$\omega_1 = 0.45$ (rad.s^{-1}) and $\omega_{-\pi} = 1.3552$ (rad.s^{-1}) are shown in Table 2.4.

Table 2.4: Candidate values for the compensated process $G_3(s)G_c(s)$

Candidate phase and gain crossover frequencies

Phase Crossover Frequency (rad.s^{-1})	Gain Crossover Frequency (rad.s^{-1})
1.3552	0.45

Candidate PID Controller Parameters

k_p	k_i	k_d
6.0833	4.0693	2.203

Achieved Gain Margin and Phase Margin

Gain Margin		Phase Margin (deg.)	
Required	Achieved	Required	Achieved
3	3	60	60

The results presented in Table 2.4 are for the partition given by [A_1. ;A_2. ;A_3.] and are the controller parameters that return the lowest value of the 1-Norm

$$L = \left| Y_4 - \sin \phi_{PM} \right|$$

The results for partitions [A_1. ;A_2. ;A_4.], [A_1. ;A_3. ;A_4.] and [A_2. ;A_3. ;A_4.] returned exactly the same results. From Figure 2.7 it can be seen that there are two curves that provide candidate frequencies at which the PID controller parameters can be calculated to give the desired gain and phase margin specification. The upper curve in Figure 2.7 does however return candidate frequencies that have little difference between the phase crossover frequency and the gain crossover frequency. Thus, the PID controller parameters that are calculated using candidate frequencies from the upper curve are characterised as having large values for the k_i and k_d terms. Large integral gain is desirable for disturbance rejection (Astrom and Hagglund, 1995) however, the large derivative gain tends to increase the rise time of the closed loop response of the system. For this reason the lower curve is chosen to provide the candidate design frequencies.

Case Study 2.4

The process is that given by $G_4(s)$ and the design gain margin and phase margin are to be 3 and 60° respectively. The frequency ranges for both R_1 and $R_{-\pi}$ are given by

$$R_{-\pi} = R_1 = (\omega_{\min}, \omega_{\max})$$

where $\omega_{\min}$ is 0.1640 (rad.s^{-1}) and $\omega_{\max}$ is 1.2223 (rad.s^{-1}). The results from the application of the enumeration method are shown in Figure 2.8.

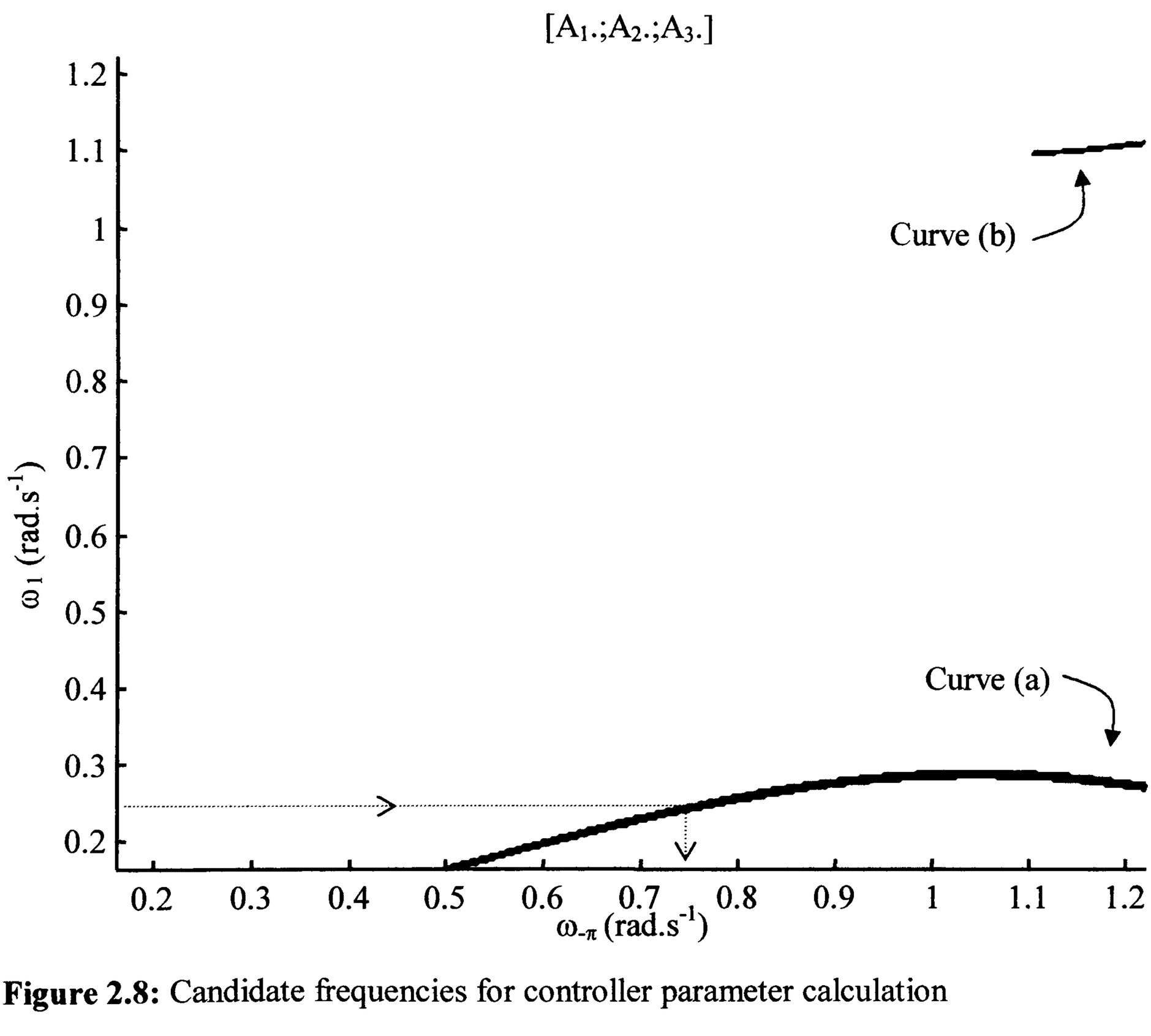


Figure 2.8: Candidate frequencies for controller parameter calculation

Similar graphs are returned from the enumeration method for the other partitions, of equation (2.39) and the controller parameters generated are exactly the same. From the figure it can be seen that there are two curves for the candidate design

frequencies. The upper curve results in integral and derivative gain terms that are relatively large due to the gain and phase margin frequencies being very close together. The results of the enumeration method for this particular design are given in Table 2.5. From the table it can be seen that the design requirements have been met. The results quoted are for the controller parameters that return the lowest value of the 1-Norm

$$L = \left| Y_4 - \sin \phi_{PM} \right|$$

Table 2.5: Candidate values for the compensated process $G_4(s)G_c(s)$

Candidate phase and gain crossover frequencies

Phase Crossover Frequency ($rad.s^{-1}$)	Gain Crossover Frequency ($rad.s^{-1}$)
0.7752	0.25

Candidate PID Controller Parameters

k_p	k_i	k_d
1.3689	1.2532	0.4158

Achieved Gain Margin and Phase Margin

Gain Margin		Phase Margin (deg.)	
Required	Achieved	Required	Achieved
3	3	60	60

2.3.3 Testing for the Existence of PID Controllers to Meet Gain and Phase Margin Design Specifications by a Graphical Means.

In the PI controller tuning method of Fung *et al* there was a match between the number of unknowns (k_p, k_i, $\omega_{-\pi}$, ω_1) and the number of available equations, there is therefore the possibility of there being a unique solution, no solution or multiple solutions. If the equation set is now extended to include the controller term k_d then the situation arises where there are five unknowns (k_p, k_i, k_d, $\omega_{-\pi}$, ω_1) and only four equations. Thus there are in general an infinite number of solutions to the

problem. To resolve this problem Tan *et al* set the frequency $\omega_{-\pi}$, of the compensated process, such that

$$\omega_{-\pi} = \alpha\omega_c \tag{2.55}$$

where $\alpha \in [0.5,2]$ and ω_c is the phase crossover frequency of the process to be controlled. Following the design method of Tan *et al*, from this point onwards, there is a loss of the graphical interpretation in the PI controller tuning method due to Fung *et al*. The PID controller that shall be used is given by

$$G_c(s) = k_p + \frac{k_i}{s} + sk_d \tag{2.56}$$

and the process is represented by, $G_p(s)$. In the subsequent analysis the substitution

$$s = j\omega$$

shall be used. Thus the process, $G_p(s)$ can be represented by

$$G_p(j\omega) = G_{pR}(\omega) + jG_{pI}(\omega)$$

where $$G_{pR}(\omega_{-\pi}) = |G_p(j\omega_{-\pi})|\cos\phi_p(\omega_{-\pi})$$

and $$G_{pI}(\omega_{-\pi}) = |G_p(j\omega_{-\pi})|\sin\phi_p(\omega_{-\pi})$$

At the phase crossover frequency of the forward path of the compensated system, $\omega_{-\pi}$, the design requirement is to ensure that

$$G_p(j\omega_{-\pi})\left(k_p + j\left(\omega_{-\pi}k_d - \frac{k_i}{\omega_{-\pi}}\right)\right) = -\frac{1}{GM} \tag{2.57}$$

where GM is the desired gain margin. Equation (2.57) has the solution

$$k_{p_{-\pi}} = \frac{-\cos\phi_p(\omega_{-\pi})}{GM|G_p(j\omega_{-\pi})|} \tag{2.58}$$

$$\left(\omega_{-\pi}k_d - \frac{k_i}{\omega_{-\pi}}\right) = \frac{\sin\phi_p(\omega_{-\pi})}{GM|G_p(j\omega_{-\pi})|} = X_{-\pi} \tag{2.59}$$

Similarly at the gain crossover frequency of the forward path of the compensated system, ω_1, the design requirement is to ensure that

$$G_p(j\omega_1)\left(k_p + j\left(\omega_1 k_d - \frac{k_i}{\omega_1}\right)\right) = -\cos\phi_{PM} - j\sin\phi_{PM} \tag{2.60}$$

where ϕ_{PM} is the desired phase margin.

Solving equation (2.60) yields the following

$$k_{p_1} = \frac{-\cos\left(\phi_p(\omega_1) - \phi_{PM}\right)}{\left|G_p(j\omega_1)\right|} \tag{2.61}$$

$$\left(\omega_1 k_d - \frac{k_i}{\omega_1}\right) = \frac{\sin\left(\phi_p(\omega_1) - \phi_{PM}\right)}{\left|G_p(j\omega_1)\right|} = X_1 \tag{2.62}$$

The positive solutions of equations (2.59) and (2.62) then give the controller parameters k_d and k_i as

$$k_d = \left(\frac{X_1}{\omega_{-\pi}} - \frac{X_{-\pi}}{\omega_1}\right)\left(\frac{\omega_1}{\omega_{-\pi}} - \frac{\omega_{-\pi}}{\omega_1}\right)^{-1} \tag{2.63}$$

$$k_i = \left(\omega_{-\pi} X_1 - \omega_1 X_{-\pi}\right)\left(\frac{\omega_1}{\omega_{-\pi}} - \frac{\omega_{-\pi}}{\omega_1}\right)^{-1} \tag{2.64}$$

There is still the resolution of the two equations for k_p, since there can only be one value of k_p. The method advocated by Tan *et al* is that a search is begun downwards from the phase crossover frequency, $\omega_{-\pi}$, of the forward path of the compensated process until a frequency is found, ω_1, such that

$$k_{p_1} = k_{p_{-\pi}}$$

$$\frac{-\cos\left(\phi_p(\omega_1) - \phi_{PM}\right)}{\left|G_p(j\omega_1)\right|} = \frac{-\cos\phi_p(\omega_{-\pi})}{GM\left|G_p(j\omega_{-\pi})\right|} \tag{2.65}$$

When the frequency ω_1 is found such that equation (2.65) is true, it is then a simple matter to calculate the remaining controller parameters using equations (2.63) and (2.64) for k_d and k_i respectively.

In the design method of Tan *et al* the graphical representation of the PI controller design method given by Fung *et al* is lost. It is however possible to provide a graphical method of finding a suitable ω_1 frequency such that the design of the PID controller can be carried out, in part, using a graphical means.

In the following method the process shall be considered to be unknown and the identifications required shall be carried out in closed loop. The Phase-Locked Loop identifier shall be used to carry out the identifications required by the method. Introduce two functions such that

$$f_a(\omega) = \frac{-\cos\phi_p(\omega)}{GM\left|G_p(j\omega)\right|}$$

and

$$f_b(\omega) = \frac{-\cos(\phi_p(\omega) - \phi_{PM})}{\left|G_p(j\omega)\right|}$$

these are of course the equations previously found for $k_{p_{-\pi}}$ and k_{p_1}. An identification process phase angle range is now specified, with the corresponding lower and upper frequencies given by:

ω_{min} is defined such that

$$\phi_p(\omega_{\min}) = -\frac{\pi}{2} + \phi_{PM}$$

and ω_{max} is defined such that

$$\phi_p(\omega_{\max}) = -\frac{3\pi}{2} + \phi_{PM}$$

and ϕ_{PM} is the design phase margin and $\phi_p(\omega)$ is the phase angle of the process. The phase angle range is now divided into an equal number of divisions and identification of the process is carried out. The data returned by the identification is the frequency and process magnitude for the particular phase angle. To allow continuous curves f_a and f_b to be drawn intermediate data are required. These data are furnished by the use of cubic-spline interpolation between the identified data points. Thus the curves f_a and f_b can now be drawn to a base of frequency. The design method is implemented by the use of the following algorithm.

Algorithm 2.4: Graphical Design of PID Controllers

Step 1: Initialisation.

Select the desired gain margin, *GM*, and phase margin, ϕ_{PM}.

Select the required closed loop bandwidth.

Select the number of identification points in the range

$$-\frac{3\pi}{2} + \phi_{PM} < \arg\left(G_p(j\omega)\right) < -\frac{\pi}{2} + \phi_{PM}$$

Step 2: Identification Step.

Identify the process at the required phase angles and record the corresponding frequency and magnitude data.

Use cubic-spline interpolation to generate the data to produce f_a and f_b curves.

Step 3: Controller design Step.

From the graph determine the phase crossover frequency corresponding to the closed loop bandwidth frequency (gain crossover frequency) and the PID controller parameter, k_p.

Use that data to calculate the remaining controller parameters k_i and k_d.

Step 4: Testing

Implement the controller design on the plant and test for acceptable control performance.

If control performance is acceptable, then stop

Else choose new gain crossover frequency and repeat at step 3.

●

The above algorithm shall now be used in the following case studies to demonstrate the use of the graphical design method.

2.3.4 Case Studies for the Gain and Phase Margin Design of PID Controllers by a Graphical Means.

The PID graphical design method shall now be utilised in the design of a PID controller given by

$$G_c(s) = k_p + \frac{k_i}{s} + sk_d$$

The controller shall be designed such that a gain margin of 3 and a phase margin of 60° shall be obtained for the candidate processes given by

$$G_3(s) = \frac{e^{-s}}{(s+1)(s+3)^2}$$

$$G_4(s) = \frac{e^{-2s}}{(s^2+s+5)(s+1)}$$

PID controllers are considered to give an acceptable control performance where the process to be controlled has dynamics that are of second order or can be adequately represented as second order (Astrom and Hagglund, 1995). Processes with dynamics greater than second order, oscillatory and processes with time delays are considered to be difficult to achieve an acceptable level of control system performance using PID control. In the following case studies, the processes are all third order with time delays. Additionally the model $G_4(s)$ has an oscillatory mode. By showing that the control performance achieved by the tuning methods when applied to these models is acceptable, it is expected that the method will perform well when used to tune a physical process.

The processes used and the design requirements in these case studies, are the same as were used in the previous case studies. This allows a comparison of the results of the two methods to be carried out.

Case Study 2.5

The two functions f_a and f_b are plotted over the range 0.1973 (rad.s^{-1}) to 1.6407 (rad.s^{-1}) and are shown in Figure 2.57.

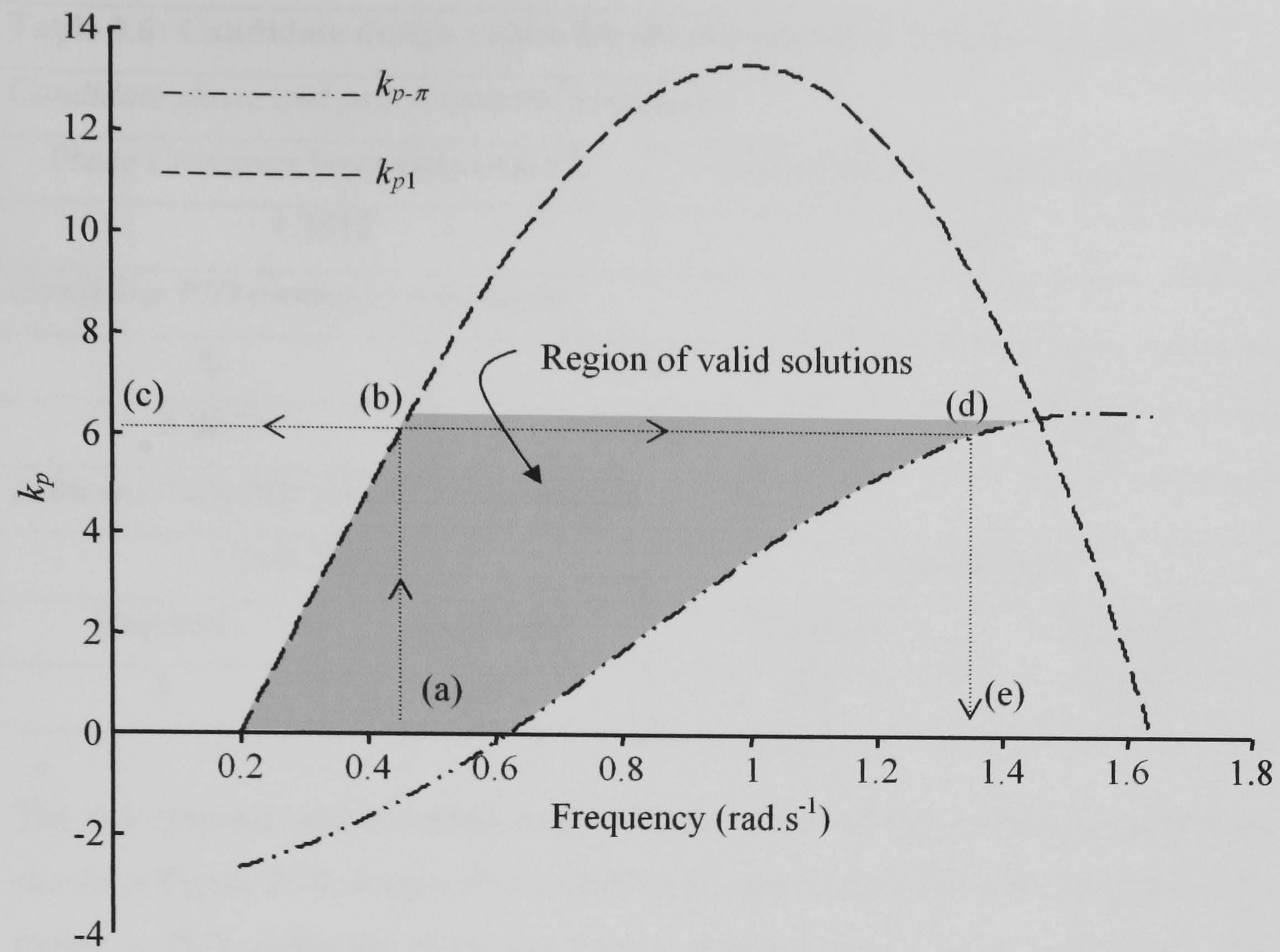


Figure 2.9: Candidate frequencies for PID controller design for process $G_3(s)$.

The function f_a corresponds with the controller parameter $k_{p\text{-}\pi}$ and the function f_b with the controller parameter k_{p1}. It can be seen from Figure 2.9 that there are an infinite number of frequency pairs $(\omega_{-\pi}, \omega_1)$ that could lead to viable PID controller parameters. To resolve this problem a design strategy must be formulated. If the closed loop bandwidth is first specified then this gives the required value of the gain crossover frequency ω_1, in Figure 2.9 this is the point marked (a), and in this case is chosen as 0.45 (rad.s^{-1}). A vertical line is now drawn from (a) to intersect with the f_b curve at the point marked (b) in the figure. From this point a horizontal line is drawn such that the controller parameter k_p can be read from the vertical axis and the intersection of this line with the f_a curve, point (d), gives the corresponding value of the phase crossover frequency, point (e). With the phase crossover and gain crossover frequencies determined it is now possible to calculate the remaining PID controller parameters namely k_i and k_d by using equations (2.59), (2.62), (2.63) and (2.64). The results of the design are shown in Table 2.6.

Table 2.6: Candidate design values for the compensated process $G_3(s)G_c(s)$

Candidate phase and gain crossover frequencies

Phase Crossover Frequency (rad.s^{-1})	Gain Crossover Frequency (rad.s^{-1})
1.3512	0.45

Candidate PID controller parameters

k_p	k_i	k_d
6.0833	4.0694	2.2033

Achieved Gain Margin and Phase Margin

Gain Margin		Phase Margin	
Required	Achieved	Required	Achieved
3	3	60	60

The step response and disturbance rejection properties of the closed loop system are shown in Figure 2.10 along with a controller designed using Ziegler-Nichols rules for tuning a PID controller with the data supplied from a relay experiment. The percentage overshoot, when the graph method designed controller is used, is approximately 12% which is considered to be rather large. However, it is an improvement over that obtained using the Ziegler-Nichols controller. The rise time of the Ziegler-Nichols tuned loop is faster than that of the graph method designed controller. Both controllers give a settling time, 5% criterion, of approximately 8 (s). The closed loop response using the Ziegler-Nichols base controller is more oscillatory than that obtained using the graph method controller. From the figure it can be seen that the step response, produced by using the graph based method controller, although acceptable for process control applications could be improved upon. In the majority of process control applications movement of the set point is not carried out too often and hence disturbance rejection is considered as a better measure of control system performance. A disturbance of magnitude 0.15 is added to the process output at time t = 30 (s), the resulting response can be seen from Figure 2.10. It can be seen that the effects of the disturbance have been removed by both controllers within approximately 10 (s), however the Ziegler-Nichols design based controller achieves a smaller integral square error figure than that of the graph based design controller. The closed loop response of the graph based design controller

could be improved if a wider bandwidth was chosen and a redesign carried out based on the wider bandwidth.

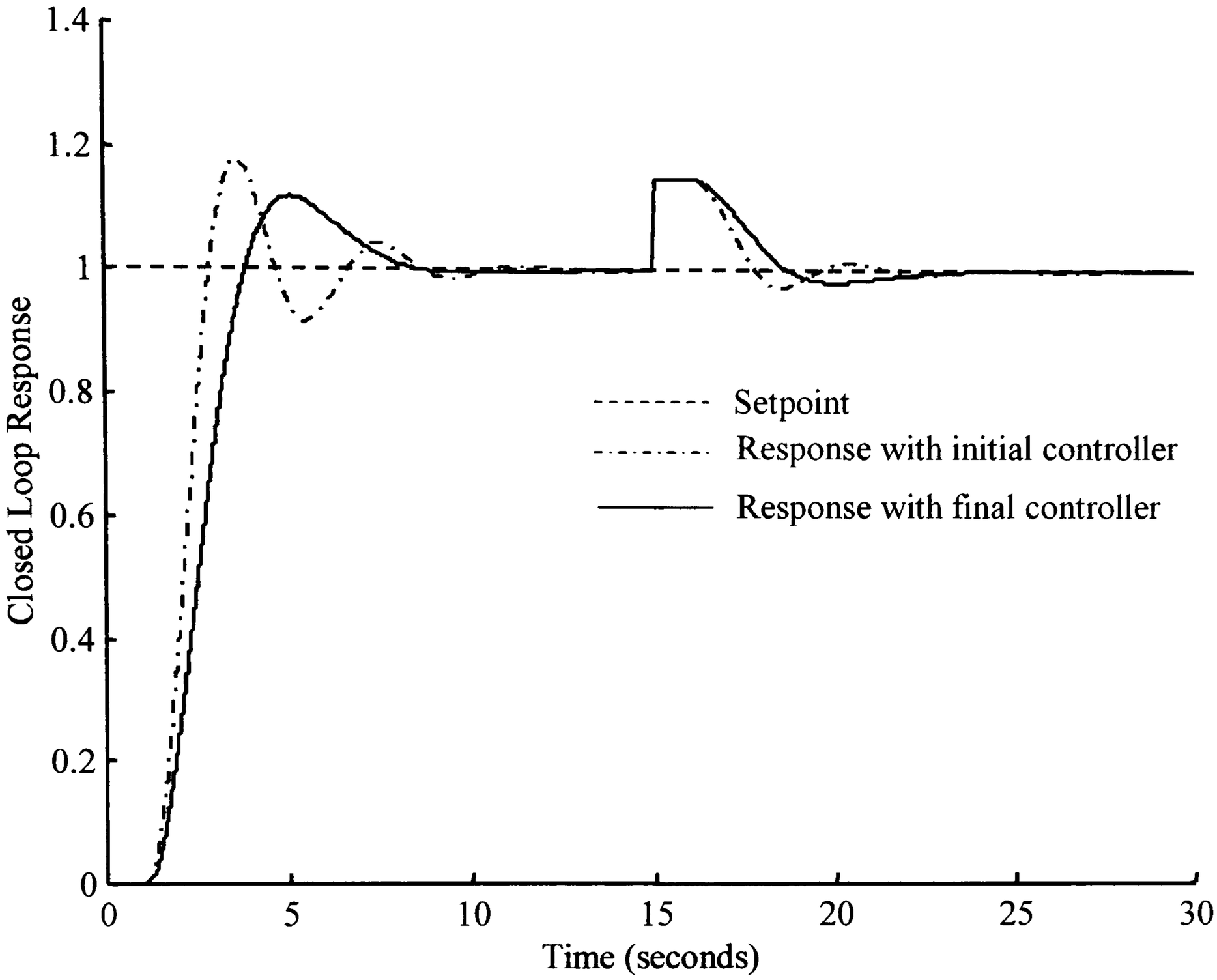


Figure 2.10: Closed loop response

Case study 2.6

The two functions f_a and f_b are plotted over the range 0.164 (rad.s^{-1}) to 1.2223 (rad.s^{-1}) for the process $G_4(s)$ and are shown in Figure 2.11.

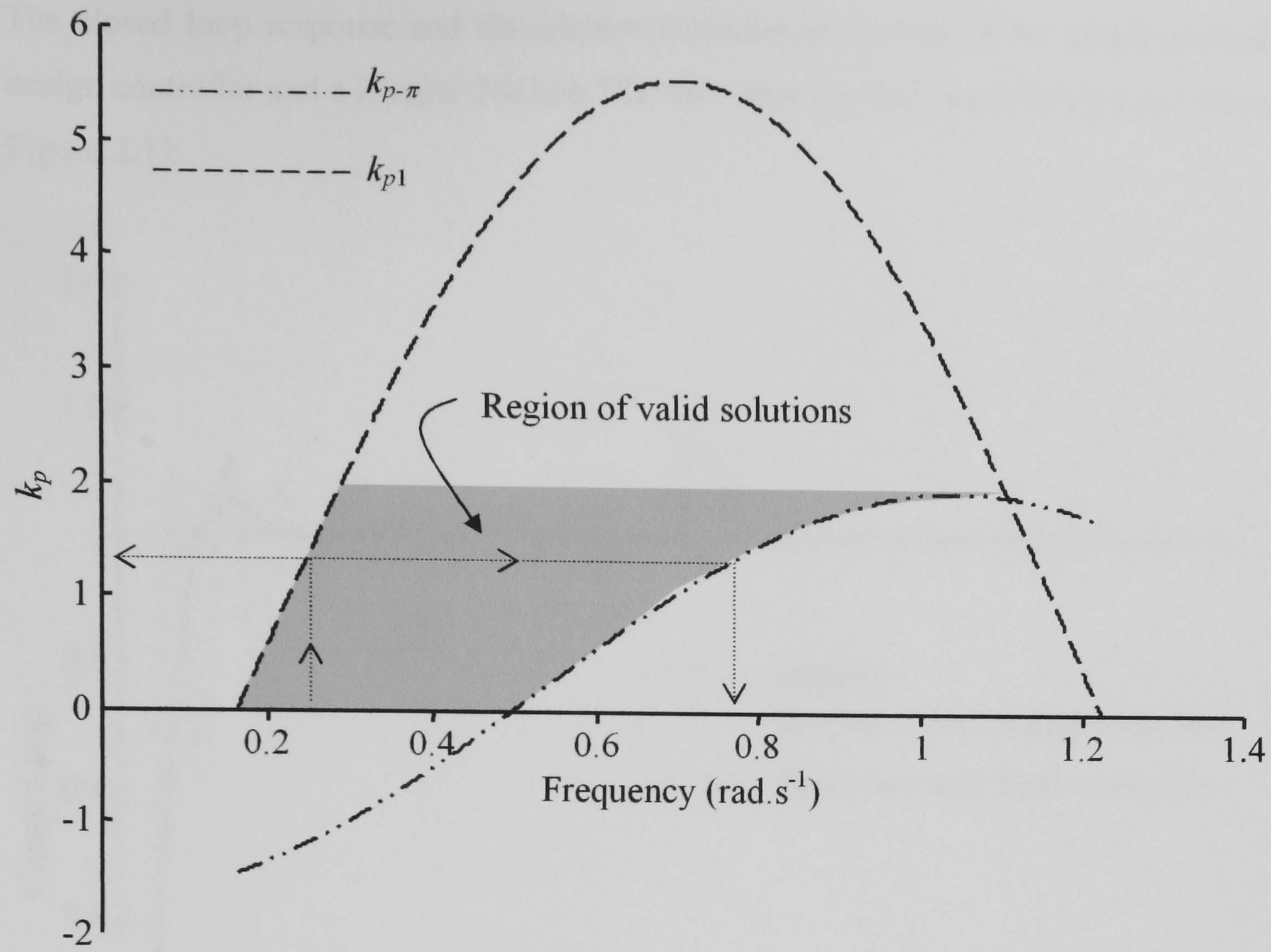


Figure 2.11: Candidate frequencies for PID controller design for process $G_4(s)$.

The closed loop bandwidth frequency was chosen to be 0.25 (rad.s^{-1}) from this the following design data was derived as shown in Table 2.7.

Table 2.7: Candidate design values for the compensated process $G_4(s)G_c(s)$

Candidate phase and gain crossover frequencies	
Phase Crossover Frequency (rad.s^{-1})	Gain Crossover Frequency (rad.s^{-1})
0.7752	0.25

Candidate PID controller parameters		
k_p	k_i	k_d
1.3689	1.2532	0.4158

Achieved Gain Margin and Phase Margin			
Gain Margin		Phase Margin	
Required	Achieved	Required	Achieved
3	3	60	60

The closed loop response and disturbance rejection properties of the graph method design controller and a Ziegler-Nichols PID rule base method controller are shown in Figure 2.12.

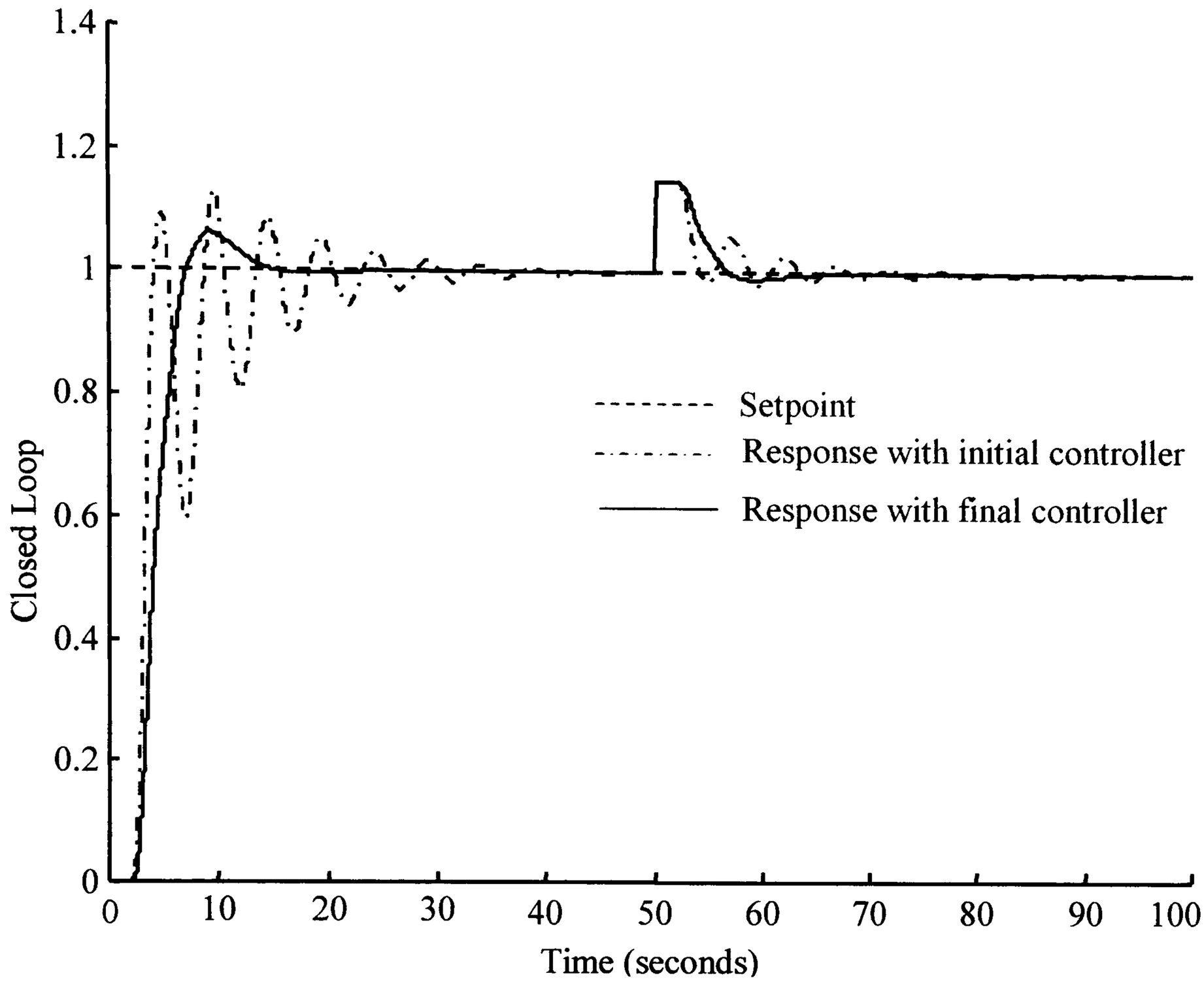


Figure 2.12: Closed loop response

The step response of the graph method design controller is much improved over that of the Ziegler-Nichols method controller both in terms of percentage overshoot and settling time. A disturbance of magnitude 0.15 was added to the process output at time t = 50(s). The graph method designed controller removed the effect of the disturbance in approximately 12(s) whereas the Ziegler-Nichols based design controller does not deal with the effect of the disturbance as effectively.

2.4 Summary Conclusions

Work based on an extension of the Fung *et al* (1998) exact gain and phase margin method was discussed that resulted in a technique that returns the achievable gain and phase margin pairings achievable by a PI controller acting on an unknown process. The identifications required by the technique are carried out in closed loop using the Phase-Locked Loop system identification module and the results are displayed in a graphical format. Additionally the PI controller parameters required for each gain and phase margin pairing are made available. Algorithms were presented that allow the semi-graphical design of PID controllers to meet specified gain and phase margins designs. In the first of the PID gain and phase margin design algorithms presented an enumeration technique is used to generate a graph showing candidate gain and phase crossover frequencies. Using a set of derived equations and the data from the graph the parameters of a PID controller can be calculated such that the design gain and phase margin can be met. In the second of the PID gain and phase margin semi-graphical design methods the method of Tan *et al* (1999) is extended. In this method a graph is generated such that the controller gain parameter, k_p, is graphed against candidate gain and phase crossover frequencies. By using the data from the graph and a set of derived equations the remaining parameters of a PID controller able to meet the given gain and phase margin specification are calculated.

3 The Design of PI Controllers to Meet Classical Robustness Measures.

3.1 Introduction.

In Chapter 2, for an unknown process, graphical methods were developed that show the gain and phase margin ranges that are achievable using PI or PID controllers. The viable gain margin and phase margin method for PI controllers developed in Chapter 2 also returns values for the controller parameters to obtain the required gain margin and phase margin pairing. However, the method only gives the controller parameters for the first solution point and as shown in Chapter 2 there is the possibility of multiple solutions yielding controller parameters. The accuracy of the results obtained by the viable gain margin and phase margin method depends on how many frequency points are identified and also on how many gain margin and phase margin points are chosen in their respective range of values. To overcome this problem and to allow the online determination of PI and PID controller parameters to give the exact gain margin and phase margin design specifications required, automated design techniques are discussed in section 3.3 and 3.4.

The classical frequency domain measures of gain and phase margin are used as measures of the relative stability of a system. Assume that a process given by $G_p(s)$ is controlled by a controller $G_c(s)$ then the gain margin (GM) and phase margin (ϕ_{PM}) are given, respectively, by

$$GM = \frac{1}{\left|G_p(j\omega_{-\pi})G_c(j\omega_{-\pi})\right|}$$

$$\phi_{PM} = \pi + \arg\left(G_p(j\omega_1)G_c(j\omega_1)\right)$$

where $\omega_{-\pi}$, the phase crossover frequency, is the frequency at which

$$\arg\left(G_p(j\omega_{-\pi})G_c(j\omega_{-\pi})\right) = -\pi$$

and ω_1, the gain crossover frequency, is the frequency at which

$$\left|G_p(j\omega_1)G_c(j\omega_1)\right| = 1$$

With a knowledge of the gain margin it is possible to calculate by how much the gain of the forward transfer function can be increased, at the phase crossover frequency, before the onset of instability. Similarly, knowing the phase margin of the forward

transfer function allows the calculation of how much phase lag can be introduced into the system, at the gain crossover frequency, prior to the onset of instability.

A further frequency domain robustness measure is given by the maximum sensitivity. The sensitivity function of a closed loop system is defined as the transfer function between the set point and the error. The maximum sensitivity is found at the frequency for which the magnitude of the sensitivity function has the greatest value. Thus, for a system having a forward transfer function given by

$$L(s) = G_p(s)G_c(s)$$

the sensitivity function is given by

$$S(s) = \frac{1}{1 + L(s)}$$

and the maximum sensitivity is then given by

$$M_s = \max_{\omega} |S(j\omega)|$$

The geometric significance of the maximum sensitivity is that it is the shortest distance from the (-1, 0) point, of a Nyquist plot, to the frequency response curve of the forward transfer function of the system. As a robustness measure the maximum sensitivity gives a measure of the maximum value of non-linearity that can be introduced into a closed loop system before the onset of instability. Khalil (1992) shows that if the non-linearity can be bounded by two straight lines with gradients

$$\alpha_1 = \frac{M_s}{(M_s + 1)}$$

$$\alpha_2 = \frac{M_s}{(M_s - 1)}$$

then the closed loop system will remain stable. The maximum sensitivity can be used to modify the closed loop time domain response of a system (Astrom *et al*, 1998) and hence lends itself to being a candidate parameter for the specification of the control performance that a PID controller is required to provide. In particular if the value of designed or achieved maximum sensitivity is reduced then the time domain response of the closed loop system becomes less oscillatory with an increase in the rise time. It should be noted that the observation in Astrom *et al* (1998) is based on experimental evidence; no theoretical basis is given to support the observation. Typically the range of values used for the maximum sensitivity to be achieved by a particular controller

design is in the range 1.2 to 2. A method to determine the parameters of a PI controller to meet a maximum sensitivity and phase margin design specification is discussed in section 3.4.

For all of the methods described in the following sections of the chapter it shall be assumed that the process to be controlled is unknown and that the controller is known. The identifications that are required are carried out using the Phase-Locked Loop identifier described in Chapter 1. In implementing the design methods of this chapter, certain prescribed points on the frequency response curve of the unknown process require to be identified. In section 3.2 the Phase-Locked Loop identifier configuration required to carry out these identifications is discussed.

The design equations, for the tuning methods discussed in this chapter that require to be solved, are discussed in section 3.1.1 and 3.1.2. Conclusions close the chapter.

3.1.1 Gain Margin and Phase Margin as Controller Design Specifications.

The objective for the controller design is to find a PI controller that satisfies given classical robustness measures of phase margin and gain margin. The design is concerned with finding a new PI controller, denoted by $G_{nc}(s)$, in cascade with an unknown process, denoted $G_p(s)$, such that the new forward path satisfies both the gain margin and phase margin specifications. Nyquist geometry shall be used to derive the required design equations. A new PI controller is introduced such that

$$G_{nc}(s) = k_p + \frac{k_i}{s} \tag{3.1}$$

The new controller can be considered to be a virtual controller, since at all times during the design phase the original controller is kept in the loop. The original controller would only be replaced or updated at the end of the design phase. The forward path of the new compensated system will be given by

$$G_{fp}(s) = G_p(s)G_{nc}(s) = G_p(s)\left(k_p + \frac{k_i}{s}\right) \tag{3.2}$$

In the frequency domain with $s = j\omega$, and introducing a Cartesian form for the unknown plant $G_p(j\omega) = G_{pR}(\omega) + jG_{pI}(\omega)$ obtain from equation (3.2)

$$G_{fp}(j\omega) = G_p(j\omega)\left(k_p + \frac{k_i}{j\omega}\right)$$

$$= \left(G_{pR}(\omega) + jG_{pI}(\omega)\right)\left(k_p + \frac{k_i}{j\omega}\right)$$

$$= \left(G_{pR}(\omega)k_p + \frac{G_{pI}(\omega)k_i}{\omega}\right) + j\left(G_{pI}(\omega)k_p - \frac{G_{pR}(\omega)k_i}{\omega}\right) \tag{3.3}$$

The specified gain margin is denoted by GM, thus the phase crossover point is given as

$$s_{-\pi} = -\frac{1}{GM} + j0$$

and this occurs at the forward path transfer function frequency of $\omega_{-\pi}$. Hence using equation (3.3)

$$G_{fp}(j\omega_{-\pi}) = \left(G_{pR}(\omega_{-\pi})k_p + \frac{G_{pI}(\omega_{-\pi})k_i}{\omega_{-\pi}}\right) + j\left(G_{pI}(\omega_{-\pi})k_p - \frac{G_{pR}(\omega_{-\pi})k_i}{\omega_{-\pi}}\right)$$

$$= -\frac{1}{GM} + j0 \tag{3.4}$$

Equating the real and imaginary parts of equation (3.4) gives

$$\begin{bmatrix} G_{pR}(\omega_{-\pi}) & \dfrac{G_{pI}(\omega_{-\pi})}{\omega_{-\pi}} \\ G_{pI}(\omega_{-\pi}) & -\dfrac{G_{pR}(\omega_{-\pi})}{\omega_{-\pi}} \end{bmatrix} \begin{bmatrix} k_p \\ k_i \end{bmatrix} = \begin{bmatrix} -\dfrac{1}{GM} \\ 0 \end{bmatrix} \tag{3.5}$$

The specified phase margin is denoted by ϕ_{PM} and this occurs at the forward path transfer function frequency ω_1 when $\left|G_{fp}(j\omega_1)\right| = 1$. The phase angle at this point is $\phi(\omega_1) = -\pi + \phi_{PM}$, thus the gain crossover point is given as

$$s_1 = e^{-j(\pi - \phi_{PM})} = -e^{j\phi_{PM}} = -\cos\phi_{PM} - j\sin\phi_{PM} \tag{3.6}$$

thus using equation (3.3) gives

$$G_{fp}(j\omega_1) = \left(G_{pR}(\omega_1)k_p + \frac{G_{pI}(\omega_1)k_i}{\omega_1}\right) + j\left(G_{pI}(\omega_1)k_p - \frac{G_{pR}(\omega_1)k_i}{\omega_1}\right)$$

$$= -\cos\phi_{PM} - j\sin\phi_{PM} \tag{3.7}$$

Equating the real and imaginary parts of equation (3.7) yields

$$\begin{bmatrix} G_{pR}(\omega_1) & \dfrac{G_{pI}(\omega_1)}{\omega_1} \\ G_{pI}(\omega_1) & -\dfrac{G_{pR}(\omega_1)}{\omega_1} \end{bmatrix} \begin{bmatrix} k_p \\ k_i \end{bmatrix} = \begin{bmatrix} -\cos\phi_{PM} \\ -\sin\phi_{PM} \end{bmatrix} \tag{3.8}$$

The equation set (3.5) and (3.8) are the key equation pairs needed in the specification of the new PI controller. Although these equations are linear in the PI controller gains k_p and k_i, the equations are non-linear being dependent on the unknown process transfer function $G_p(s)$ and the frequencies $\omega_{-\pi}$ and ω_1.

3.1.2 Maximum Sensitivity and Phase Margin as a PI Controller Design Specifications.

In this section the design is concerned with finding a new PI controller, denoted by $G_{nc}(s)$, in cascade with an unknown process, denoted $G_p(s)$, such that the new forward path satisfies the maximum sensitivity and phase margin specifications. The resulting design equations are exactly the same for the phase margin as are given in equation (3.8), hence only the design equations for the maximum sensitivity shall be developed in this section. Nyquist geometry shall be used to derive the required design equations.

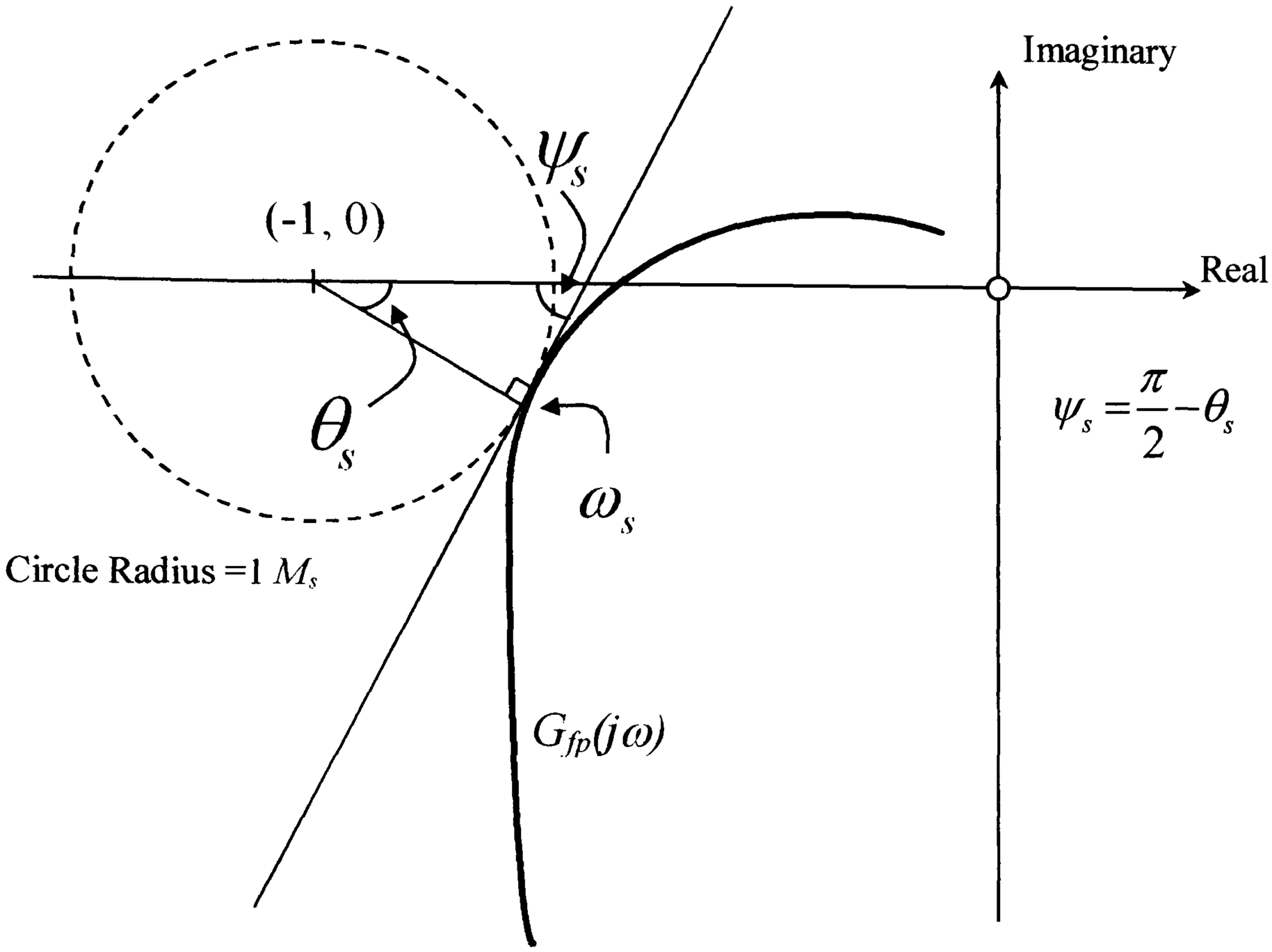


Figure 3.1: Tangency condition on maximum sensitivity

The new PI controller is introduced such that

$$G_{nc}(s) = k_p + \frac{k_i}{s} \tag{3.9}$$

The new controller can be considered to be a virtual controller, since at all times during the design phase the original controller is kept in the loop. The original controller would only be replaced or updated at the end of the design phase. The forward path of the new compensated system will be given by

$$G_{fp}(s) = G_p(s)G_{nc}(s) = G_p(s)\left(k_p + \frac{k_i}{s}\right) \tag{3.10}$$

In the frequency domain with $s = j\omega$, and introducing a Cartesian form for the unknown plant $G_p(j\omega) = G_{pR}(\omega) + jG_{pI}(\omega)$ obtain from equation (3.10)

$$G_{fp}(j\omega) = \left(G_{pR}(\omega)k_p + \frac{G_{pI}(\omega)k_i}{\omega}\right) + j\left(G_{pI}(\omega)k_p - \frac{G_{pR}(\omega)k_i}{\omega}\right) \tag{3.11}$$

It is desired to find a PI controller $G_{nc}(s)$ such that the compensated process meets a given maximum sensitivity, M_S specification. As can be seen from Figure 3.1, Nyquist geometry gives the following relationship for the maximum sensitivity point

$$s_{M_s} = -\left(1 - \frac{1}{M_S}\cos\theta_S\right) - j\frac{1}{M_S}\sin\theta_S \tag{3.12}$$

and this occurs at the forward path transfer function frequency of ω_s. Hence using equation (3.11)

$$G_{fp}(j\omega_s) = \left(G_{pR}(\omega_s)k_p + \frac{G_{pI}(\omega_s)k_i}{\omega_s}\right) + j\left(G_{pI}(\omega_s)k_p - \frac{G_{pR}(\omega_s)k_i}{\omega_s}\right)$$

$$= -\left(1 - \frac{1}{M_S}\cos\theta_S\right) - j\frac{1}{M_S}\sin\theta_S \tag{3.13}$$

Equating the real and imaginary parts of equation (3.13) gives

$$\begin{bmatrix} G_{pR}(\omega_s) & \dfrac{G_{pI}(\omega_s)}{\omega_s} \\ G_{pI}(\omega_s) & -\dfrac{G_{pR}(\omega_s)}{\omega_s} \end{bmatrix}\begin{bmatrix} k_p \\ k_i \end{bmatrix} = \begin{bmatrix} -\left(1 - \dfrac{1}{M_S}\cos\theta_S\right) \\ -\dfrac{1}{M_S}\sin\theta_S \end{bmatrix} \tag{3.14}$$

where ω_S is the frequency at which the *new* compensated forward path is tangent to the $1/M_S$ circle and θ_S is the angle between the negative real axis the -1 point and the point where the new compensated forward path locus is tangent to the $1/M_S$ circle.

3.2 Automated Identification for PI Controller Design.

In the methods discussed in the introduction to Chapter 2 a requirement was that an explicit process model was available to allow the controller design to be undertaken. It is however interesting to note that in almost all of the literature cited no mention is made as to how the process model is to be obtained. In this section the Phase-Locked Loop identification method, described in Chapter 1, is used to carry out the required identifications. The identifications required shall be carried out in closed loop with the process considered to be unknown.

In the automated PI controller design method to follow, gain margin, phase margin and maximum sensitivity specifications will be given and data sought for the unknown process at specific frequencies. To achieve these frequencies, the Phase-Locked Loop identifier structure of Figure 3.2 has to be modified according to the identification case.

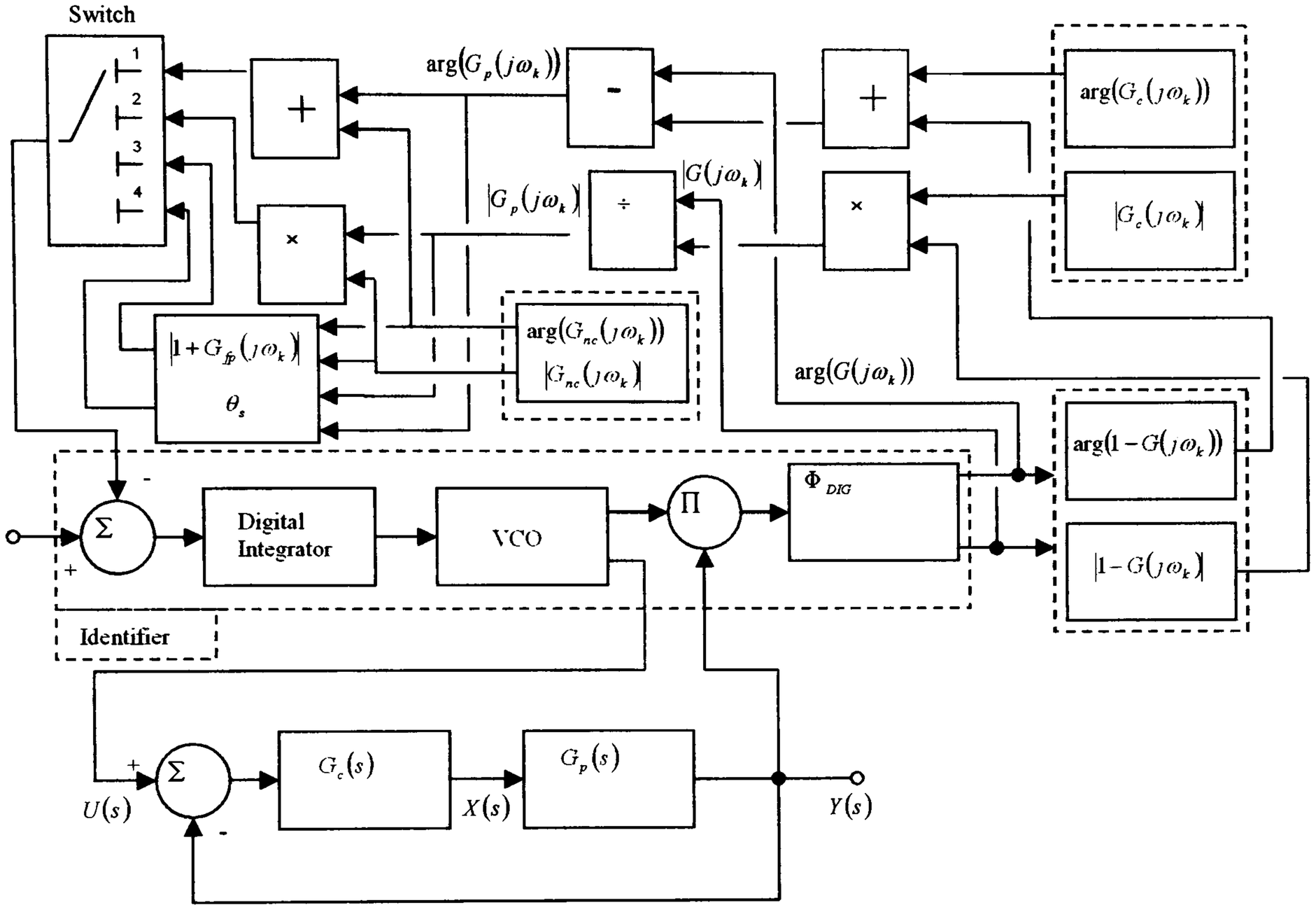


Figure 3.2: Closed loop identification – known controller.

However, the identification always retains the existing PID controller in the loop, and uses a parallel computation to place specifications on the new compensated forward path, $G_{fp}(s)=G_p(s)G_{nc}(s)$ to find the required frequencies.

In the following it shall be assumed that the process to be controlled is represented by $G_p(s)$ and that the controller is represented by $G_c(s)$. The compensated forward path of the closed loop system is represented by $G_{fp}(s)=G_p(s)G_c(s)$.

3.2.1 Gain Crossover Frequency Identification.

Using the substitution $s=j\omega$, the frequency at which

$$\left|G_{fp}(j\omega)\right| = \left|G_p(j\omega)G_{nc}(j\omega)\right| = 1$$

is required and is defined as the gain crossover frequency, ω_1. The Phase-Locked Loop identifier identifies $G_{CL}(j\omega)$ and uses

$$G_p(j\omega) = \frac{\left|G_{CL}(j\omega)\right|}{\left|1 - G_{CL}(j\omega)\right|\left|G_c(j\omega)\right|}$$

to compute $G_p(j\omega)$. In Figure 3.2 with the reference input set to 1 and the Switch in position 2, the identification of $G_p(j\omega)$ can then be used to evaluate the reference error equation,

$$e_k = 1 - \left|G_p(j\omega)\right| \times \left|G_{nc}(j\omega)\right|$$

since the controller $G_{nc}(s)$ is considered to be known. When this error is zero, the frequency found is ω_1 (Crowe and Johnson, 2002b; Crowe and Johnson, 2000a).

3.2.2 Phase Crossover Frequency Identification.

The frequency at which

$$\arg\left(G_{fp}(j\omega)\right) = \arg\left(G_p(j\omega)G_{nc}(j\omega)\right) = -\pi$$

is required and is defined as the phase crossover frequency, $\omega_{-\pi}$. Thus, the Phase-Locked Loop identifier identifies $G_{CL}(j\omega)$ and uses

$$\arg\left(G_p(j\omega)\right) = \arg\left(G_{CL}(j\omega)\right) - \arg\left(1 - G_{CL}(j\omega)\right) - \arg\left(G_c(j\omega)\right)$$

to compute $\arg\left(G_p(j\omega)\right)$. In Figure 3.2 with the reference input set to -π (rad) and the Switch in position 1, the identification of $\arg\left(G_p(j\omega)\right)$ can then be used to evaluate the reference error equation,

$$e_k = -\pi - \left(\arg\left(G_p(j\omega_k)\right) + \arg\left(G_{nc}(j\omega_k)\right)\right)$$

When this error is zero, the frequency $\omega_{-\pi}$ is found (Crowe and Johnson, 2002b; Crowe and Johnson, 2000a).

3.2.3 Maximum Sensitivity Frequency Identification.

The Nyquist geometry for the maximum sensitivity case is shown in Figure 3.1. The angle θ_S between the negative real axis, the –1 point and the point where the compensated forward path is tangent to the $1/M_S$ circle, is not known *a priori* and an estimate must be constructed. However, from Figure 3.1 it can be seen that, $\psi_S = \frac{\pi}{2} - \theta_S$, and the procedure for estimating θ_S uses this identity (Crowe and Johnson, 2001a; Crowe and Johnson, 2002a). From Figure 3.3, it can be seen that generally, $G_{fp}(j\omega)$ intersects the $1/M_S$ circle at two points, with frequencies ω_A and ω_B. These intersections occur where,

$$\left|\frac{1}{M_S}\right| = \left|1 + G_{fp}(j\omega_A)\right| = \left|1 + G_{fp}(j\omega_B)\right| \tag{3.15}$$

From these two locations the angles θ_1 and θ_2 can be found

$$\theta_1 = \tan^{-1}\left(\frac{\left|\mathrm{Im}\left(G_{fp}(j\omega_B)\right)\right|}{1 - \left|\mathrm{Re}\left(G_{fp}(j\omega_B)\right)\right|}\right) \tag{3.16a}$$

$$\theta_2 = \tan^{-1}\left(\frac{\left|\mathrm{Im}\left(G_{fp}(j\omega_A)\right)\right|}{1 - \left|\mathrm{Re}\left(G_{fp}(j\omega_A)\right)\right|}\right) \tag{3.16b}$$

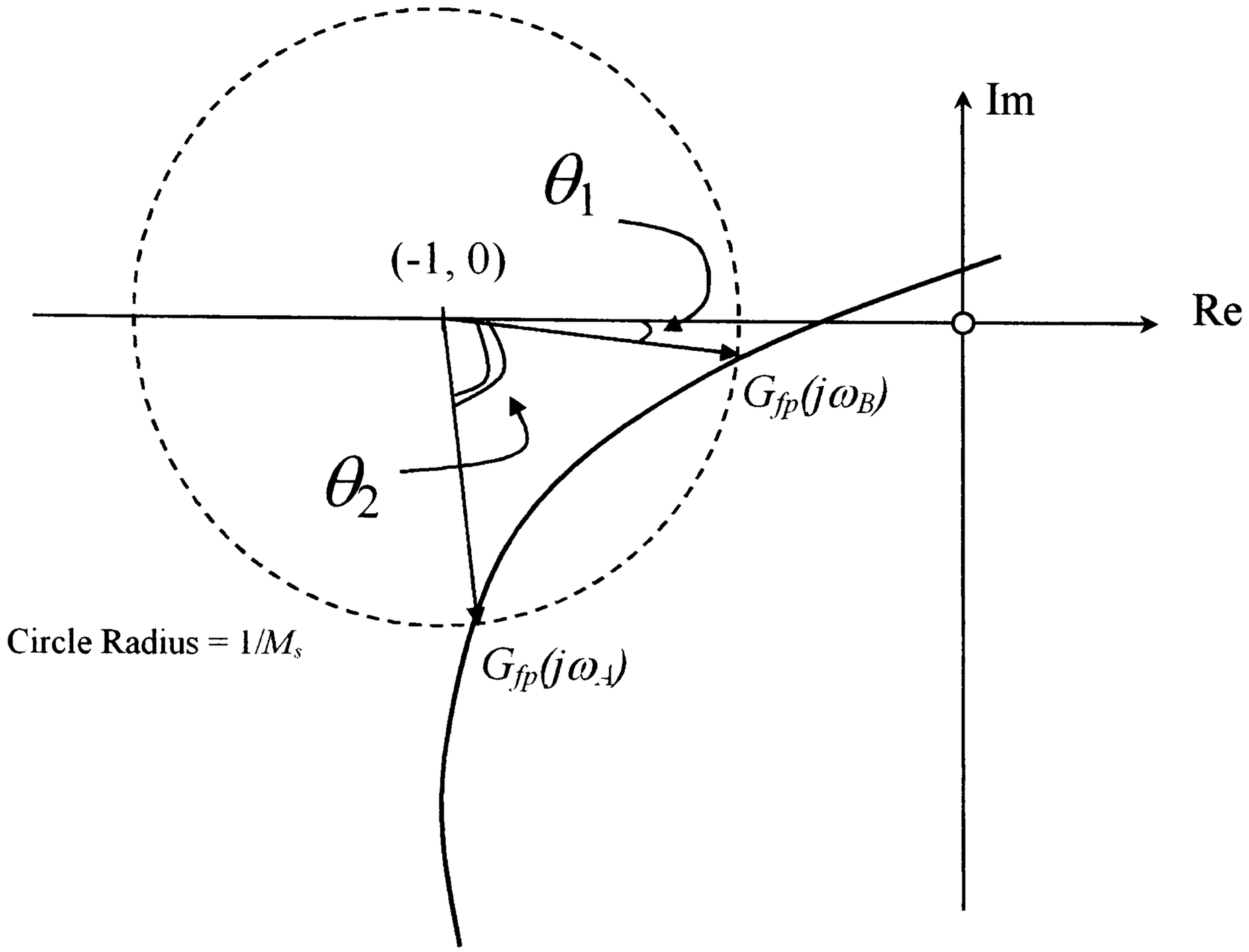


Figure 3.3: Intersection of the $G_{fp}(j\omega)$ locus and the 1/Ms circle

The values of θ_1, θ_2 are then used to calculate the angles, ψ_1 and ψ_2 that can be used to form a bound on θ_S. It follows that,

$$\psi_1 = \frac{\pi}{2} - \theta_1 \tag{3.17a}$$

$$\psi_2 = \frac{\pi}{2} - \theta_2 \tag{3.17b}$$

An estimate on the angle ψ_S is constructed using the geometric mean angle of ψ_1 and ψ_2,

$$\hat{\psi}_S = \sqrt{\psi_1 \psi_2} \tag{3.18}$$

and, from this,

$$\hat{\theta}_S = \left(\frac{\pi}{2}\right) - \hat{\psi}_S \tag{3.19}$$

It should be noted that the mean estimator for $\hat{\psi}_S$ favours the smaller of the two angles ψ_1 and ψ_2. The above explanation leads to the identification steps needed to find frequencies and system data to design for a maximum sensitivity specification. To implement this identification, two frequency identification steps are needed.

Step 1

The forward path data at the two frequencies ω_A and ω_B are needed. These are the intersection points of the Nyquist plot of $G_{fp}(s)$ and the $1/M_S$ circle (see Figure 3.3), and these points occur at the condition,

$$\left|\frac{1}{M_S}\right| = \left|1 + G_p(j\omega_A)G_{nc}(j\omega_A)\right| = \left|1 + G_p(j\omega_B)G_{nc}(j\omega_B)\right| \qquad (3.20)$$

Thus, in Figure 3.2 with the Switch in position 3 and a reference input of $1/M_s$ they are found using the Phase-Locked Loop identifier module and a reference error given by

$$e_k = \left(\frac{1}{M_S}\right) - \left|1 + G_p(j\omega_k)G_{nc}(j\omega_k)\right| \qquad (3.21)$$

From the reference error equation (3.21) it is obvious that when the Phase-Locked Loop identifier drives the reference error to zero it will not automatically seek the second frequency at which the magnitude of the forward path equals $1/M_s$. To ensure that both frequencies are found the following method is used. To find the frequency ω_A the Phase-Locked Loop identifier is initialised at a frequency below ω_A and when finding ω_B the identifier is initialised at a frequency higher than ω_B. In the examples given later in the chapter when finding ω_A the initialising frequency chosen is the gain crossover frequency of the forward path transfer function $G_{fp}(s)$ and when finding ω_B the initialising frequency is the frequency at which

$$Arg\left(G_p(j\omega_{-\pi})\right) = -\pi$$

Using this reference error, the Phase-Locked Loop identification automatically produces values of $\left|G_{fp}(j\omega_A)\right|$, $\arg\left(G_{fp}(j\omega_A)\right)$ and $\left|G_{fp}(j\omega_B)\right|$, $\arg\left(G_{fp}(j\omega_B)\right)$.

Step 2

Using the estimate of $\hat{\theta}_S$, the frequency $\hat{\omega}_S$ can be found. This frequency is identified using the condition,

$$\hat{\theta}_S = \tan^{-1}\left(\frac{\left|\mathrm{Im}\left(G_p(j\omega)\times G_{nc}(j\omega)\right)\right|}{1-\left|\mathrm{Re}\left(G_p(j\omega)\times G_{nc}(j\omega)\right)\right|}\right) \tag{3.22}$$

In Figure 3.2 with the Switch in position 4 and a reference input of $\hat{\theta}_S$, the Phase-Locked Loop identifier module framework will be driven to find ω_S by using a reference error equation,

$$e_k = \left(\hat{\theta}_S\right) - \tan^{-1}\left(\frac{\left|\mathrm{Im}\left(G_p(j\omega_k)\times G_{nc}(j\omega_k)\right)\right|}{1-\left|\mathrm{Re}\left(G_p(j\omega_k)\times G_{nc}(j\omega_k)\right)\right|}\right) \tag{3.23}$$

As can be seen it is the extensive identification flexibility of the Phase-Locked Loop module that enables these various frequency points to be found.

3.3 Gain Margin and Phase Margin: Automated PI Controller Design.

The Phase-Locked Loop module is used in the automated PI control design procedure to perform the identification steps of the routine. The identification steps are performed in closed loop with the existing controller, $G_c(s)$ left in place. The process is assumed to be unknown and denoted by $G_p(s)$. The new PI controllers will be given by,

$$G_{nc}(s) = k_p + \frac{k_i}{s}$$

The forward path of the new compensated system will be given by,

$$G_{fp}(s) = G_p(s)G_{nc}(s) = G_p(s)\left(k_p + \frac{k_i}{s}\right)$$

Thus, by combining equations (3.5) and (3.8) a generic equation suite for the new PI compensator gains, k_p, k_i is given by

$$\begin{bmatrix} G_{pR}(\omega_{-\pi}) & \dfrac{G_{pI}(\omega_{-\pi})}{\omega_{-\pi}} \\ G_{pI}(\omega_{-\pi}) & -\dfrac{G_{pR}(\omega_{-\pi})}{\omega_{-\pi}} \\ G_{pR}(\omega_1) & \dfrac{G_{pI}(\omega_1)}{\omega_1} \\ G_{pI}(\omega_1) & -\dfrac{G_{pR}(\omega_1)}{\omega_1} \end{bmatrix} \begin{bmatrix} k_p \\ k_i \end{bmatrix} = \begin{bmatrix} \dfrac{-1}{GM} \\ 0 \\ -\cos\phi_{PM} \\ -\sin\phi_{PM} \end{bmatrix} \tag{3.24}$$

This can be written succinctly as,

$$[X]K = Y^D$$

$$\theta = \begin{bmatrix} k_p & k_i \end{bmatrix}^T$$

$$Y^D = \begin{bmatrix} -\dfrac{1}{GM} & 0 & -\cos\phi_{PM} & -\sin\phi_{PM} \end{bmatrix}$$

Using this equation suite and the Phase-Locked Loop identification module an automated PI controller design routine can be constructed.

Algorithm 3.1: Automated PI Controller Design for Desired Gain Margin and Phase Margin Specifications.

Step 1: Design Specification.

Select the desired gain margin, GM and phase margin ϕ_{PM}, for the control design.

Compute $Y^D = \begin{bmatrix} -\dfrac{1}{GM} & 0 & -\cos\phi_{PM} & -\sin\phi_{PM} \end{bmatrix}$

Step 2: Initialisation Step

Initialise counter $n = 0$

Choose initial PI controller gains (can be those already used with the closed loop system): $k_p(0), k_i(0)$

Step 3: Identification Step

Step 3a: Phase crossover frequency identification

Use Phase-Locked Loop to find $\omega_{-\pi}(n)$ for the forward path $G_{fp}(s) = G_p(s)G_{nc}(s)$

And use known $k_p(n), k_i(n)$ to solve for $G_{pR}(\omega_{-\pi}), G_{pI}(\omega_{-\pi})$

Step 3b: Gain crossover frequency identification

Use Phase-Locked Loop to find $\omega_1(n)$ for the forward path $G_{fp}(s) = G_p(s)G_{nc}(s)$

Use known $k_p(n), k_i(n)$ to solve for $G_{pR}(\omega_{-\pi}), G_{pI}(\omega_{-\pi})$

Step 3c: Convergence test step

Compute $Y(n) = \left[-\frac{1}{GM(n)} \quad 0(n) \quad -\cos\phi_{PM}(n) \quad -\sin\phi_{PM}(n) \right]$

If $\left\| Y^D - Y(n) \right\| < tol$, then STOP.

Step 4: Controller update calculation

Use $\omega_{-\pi}, \omega_1, G_{pR}(\omega_{-\pi}), G_{pI}(\omega_{-\pi}), G_{pR}(\omega_1), G_{pI}(\omega_1)$

Form $[X_n]K = Y^D$

Solve as $K = \left[X_n^T X_n\right]^{-1} X_n^T Y^D$

Update, $n := n+1$, $k_p(n+1) = K_1$, $k_i(n+1) = K_2$

Go to Step 3 ●

3.3.1 Gain Margin and Phase Margin Design Algorithm Convergence Proof

In the previous section an algorithm was proposed that returns the gains of a PI controller such that a specified gain margin and phase margin can be achieved by using that controller as the compensation element. In this section a proof is proposed, that shows that under a relatively weak set of conditions, the algorithm shall return controller gains that converge to a fixed point solution of the equation set (3.24).

The forward path transfer function of the compensated system is given by

$$G_{fp}(s) = G_p(s)G_{nc}(s)$$

where the new controller is

$$G_{nc}(s) = k_p + \frac{k_i}{s}$$

From this it can be seen that the independent variables of the problem are k_p and k_i. Hence define

$$K = \begin{bmatrix} k_p \\ k_i \end{bmatrix}$$

where $K \in \Re_+^2$

For a given $K \in \Re_+^2$ the Phase-Locked Loop identifier is used to find

$$\omega_{-\pi}\left(k_p, k_i\right)$$

and

$$\omega_1\left(k_p, k_i\right)$$

This is written as

$$\omega_\nu(K)=\begin{bmatrix}\omega_{-\pi}\\ \omega_1\end{bmatrix}$$

where $\omega_\nu(K):\Re_+^2 \to \Re_+^2$ and $\omega_\nu(K)$ is non-linear.

The next step is to calculate an updated value of K using least squares, hence define

$$X(\omega_\nu)=\tilde{X}(\omega_{-\pi},\omega_1)$$

where $X(\omega_\nu):\Re_+^2 \to \Re^2$. The updated controller gains are thus found from

$$\begin{aligned}K_{n+1} &= \left[X^T(\omega_\nu)X(\omega_\nu)\right]^{-1}X^T(\omega_\nu)Y^D\\ &= f(\omega_\nu) = f(\omega_\nu(K_n)) = g(K_n)\end{aligned}$$

where $f:\Re_+^2 \to \Re^2$, $\omega_\nu:\Re_+^2 \to \Re_+^2$ and $g:\Re_+^2 \to \Re^2$.

Theorem 3.1: Sufficient Conditions for Convergence

Assuming suitable problem set-up assumptions are valid, then for the gain margin and phase margin PI tuning algorithm the basic convergence relationship is

$$\|K_* - K_{n+1}\| \le \left(\prod_{j=0}^{n}\mu_j\right)\|K_* - K_0\|$$

where

$$\mu_j = \sup_{0<\alpha_j<1}\left\|g'\left(K_j + \alpha_j\left(K_* - K_j\right)\right)\right\|$$

The sufficient conditions for the convergence are:

(a) $\exists\ j_1 \ge 0$ such that $\forall\ j \ge j_1$ μ_j satisfies $|\mu_j| < 1$.

(b) $|\mu_j| < 1$ for all $j = 0,1,\cdots,\infty$.

Proof

i) $K_* = g(K_*)$ and $K_{n+1} = g(K_n)$

ii) $K_* - K_{n+1} = g(K_*) - g(K_n)$ and hence $\|K_* - K_{n+1}\| = \|g(K_*) - g(K_n)\|$

Applying Luenberger's mean value inequality result (Luenberger, 1969) gives

$$\|K_* - K_{n+1}\| \leq \mu_n \|K_* - K_n\|$$

where $\mu_n = \sup_{0<\alpha<1} \|g'(K_n + \alpha(K_* - K_n))\|$

iii) Repeated application of this inequality yields

$$\|K_* - K_{n+1}\| \leq \left(\prod_{j=0}^{n} \mu_j \right) \|K_* - K_0\|$$

where $\mu_j = \sup_{0<\alpha_j<1} \|g'(K_j + \alpha_j(K_* - K_j))\|$

iv) Proof of clause (a)

Write $$\prod_{j=0}^{n} \mu_j = \left(\prod_{j=j_1+1}^{n} \mu_j \right) \left(\prod_{j=0}^{j_1} \mu_j \right)$$

$$= k_{j_1} \left(\prod_{j=j_1+1}^{n} \mu_j \right)$$

Thus if $|\mu_j| < 1$ for all $j \geq j_1$ then as $n \to \infty$, $\prod_{j=0}^{n} \mu_j \to 0$ and convergence follows.

v) Clause (b) is a special case of clause (a). ●

This theorem is a classic contraction mapping result. Satisfaction of the result depends on a number of properties; the existence of a closed region, denoted $\Theta \subseteq \Re_+^2$, through which the iterates, θ_j travel, and conditions on the Jacobian, $\left[\frac{\partial g}{\partial \theta}\right] = g'(\theta)$ over this closed region which incorporates θ^*. Schwarz (1989) links the contraction property to the necessary and sufficient condition that the spectral radius of the Jacobian, $g'(\theta)$ must satisfy $\rho(g'(\theta))<1$ over the closed region, Θ. However, the particular difficulties associated with finding analytical conditions for the convergence of the autonomous PI algorithm include the function of a function implicit in the relation, $g(\theta)$, where the Jacobian can be expressed as,

$$g'(\theta) = \left[\frac{\partial g}{\partial \theta}\right]$$

$$\left[\frac{\partial g}{\partial \theta}\right]=\left[\frac{\partial f}{\partial \omega_v}\right]\times\left[\frac{\partial \omega_v}{\partial \theta}\right]$$

the complexity of the function *f(.)* and the indirect computational access to the function, ω_v.

3.3.2 Implementation and Case Study Results for the Gain and Phase Margin Design Method

The candidate processes were chosen to demonstrate the ability of the tuning method to achieve both a gain margin and phase margin specification using a PI controller; these repeat processes are given by

$$G_1(s)=\frac{1}{(s+1)^6},\ G_2(s)=\frac{e^{-s}}{(s+1)(s+3)^2}$$

PI controllers are known to give a satisfactory closed loop control system response where the dynamics of the process are of first order (Astrom and Hagglund, 1995). Where a process has dynamics greater than first order or includes a time delay, then it is difficult to tune a PI controller to give an acceptable level of closed loop system performance. By applying the tuning method to the models $G_1(s)$, a high order non-oscillatory process and $G_2(s)$, a third order process with a time delay and achieving an acceptable level of closed loop system response, when the method is applied to a physical process similar results will be obtained. The processes, $G_1(s)$ and $G_2(s)$ are representative of those found in the process industries.

Case Study 3.1

The required design specification for the compensated forward path for process $G_1(s)$ was chosen as a gain margin of 3 and a phase margin of 60^0. The controller parameters together with the gain margin, phase margin, phase crossover frequency and gain crossover frequency for the initial and final values derived from the design simulation carried out for the process $G_1(s)$ are shown in Table 3.1.

Table 3.1: Initial and achieved values for the compensated process $G_1(s)G_c(s)$

Initial Values

k_p	k_i	Gain Margin	Phase Margin (Degrees)	Phase Crossover Frequency ($rad.s^{-1}$)	Gain Crossover Frequency ($rad.s^{-1}$)
0.9372	0.1077	2.2	85.5	0.5309	0.1839
Achieved Values					
0.4265	0.1545	3.0	60.0	0.4312	0.1565

The initial PI controller parameters were derived using a relay experiment followed by the application of Ziegler-Nichols tuning rules for a PI controller. This method of determining the initial tuning parameters means that the tuning algorithm is starting from a relatively long way from the desired tuning parameters. The choice of gain margin and phase margin values were made so that the resulting compensated system would have good stability robustness and good robustness to process parameter variations.

The progress of the gain and phase margin design method is shown in the following graphs for gain margin, phase margin, k_p, k_i and μ_n. The graphs are plotted to a base of algorithm iteration number. In order to relate the iteration number to the time taken to complete an iteration use is made of the following data:

i) the average time taken to identify the gain crossover point of the compensated forward path was 321s).

ii) the average time taken to identify the phase crossover point of the compensated forward path was 187(s).

Using this data and Algorithm 3.1, it can be seen that each iteration of the algorithm takes an average of 508(s). The settling time (2% criterion) for the process $G_1(s)$ is approximately 12(s) hence each iteration of the algorithm takes on average 42 settling time periods for this particular process.

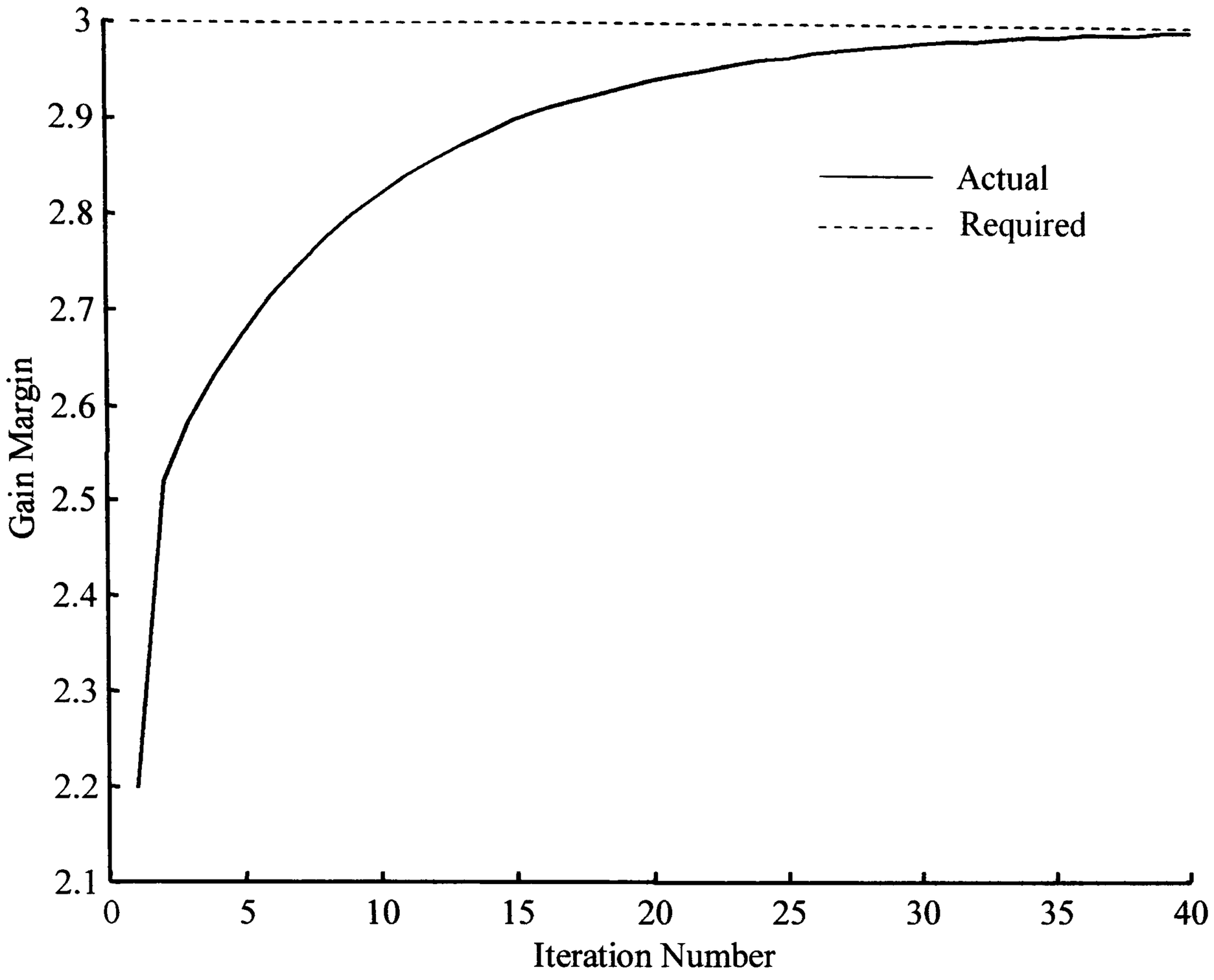


Figure 3.4: Evolution of gain margin

Figure 3.4 shows how the gain margin of the compensated forward path of the process $G_1(s)G_c(s)$ evolves as a new iteration of the algorithm is completed. It can be seen that after approximately twelve iterations of the algorithm, that the gain margin is within 5% of the desired value and is within 2% after approximately twenty-three iterations.

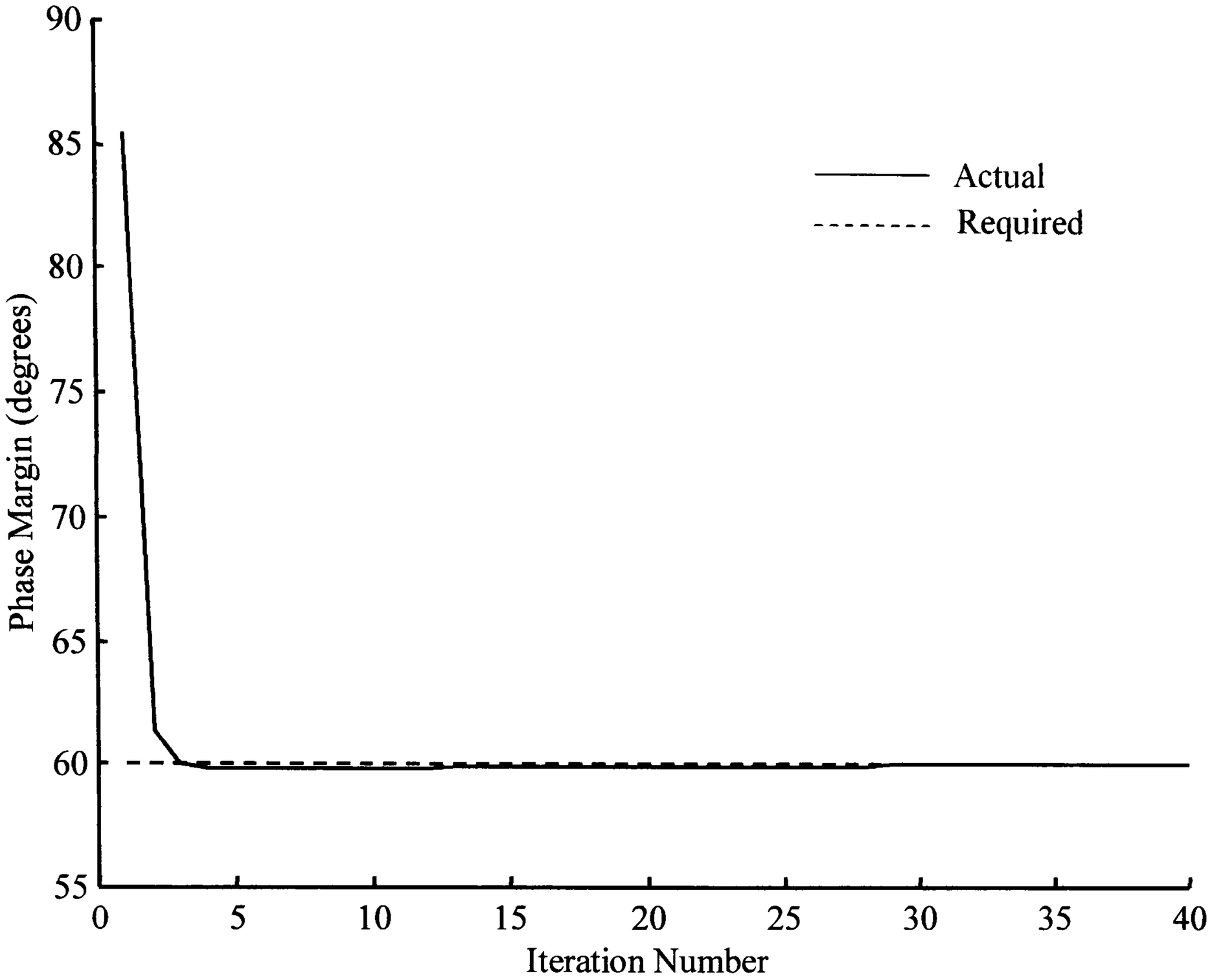


Figure 3.5: Evolution of phase margin

Correspondingly Figure 3.5 shows how the phase margin evolves as an iteration of the PI controller gain margin and phase margin algorithm is completed. From Figure 3.5 it can be seen that the phase margin is practically attained after five iterations of the algorithm. This type of behaviour, fast attainment of the desired phase margin, has been a feature of all of the simulations that have been carried out on representative processes from the process industries. Correspondingly, the attainment of the desired gain margin has been seen to be much slower, as seen by the number of iterations of the algorithm required to reach convergence.

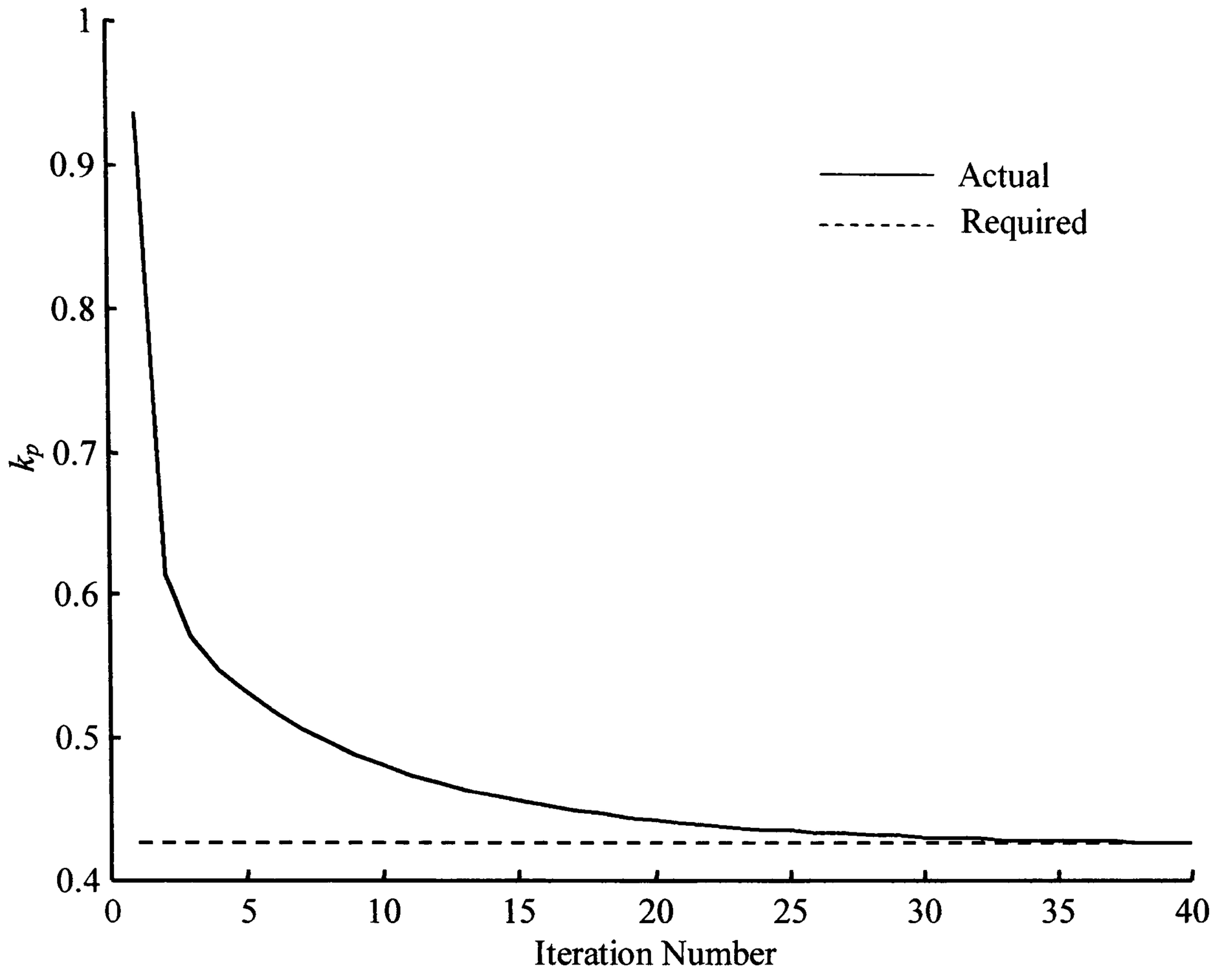


Figure 3.6: Evolution of PI controller parameter k_p

Figure 3.6 shows how the value of the PI controller parameter k_p evolves as the algorithm is executed. From this it can be seen that there is an initial large change in value followed by a more gentle convergence to the final value. The corresponding evolution of the PI controller gain term, k_i, is shown in Figure 3.7. Again as can be seen from the figure there is an initial rapid change in the parameter value followed by a more subdued change towards the final value.

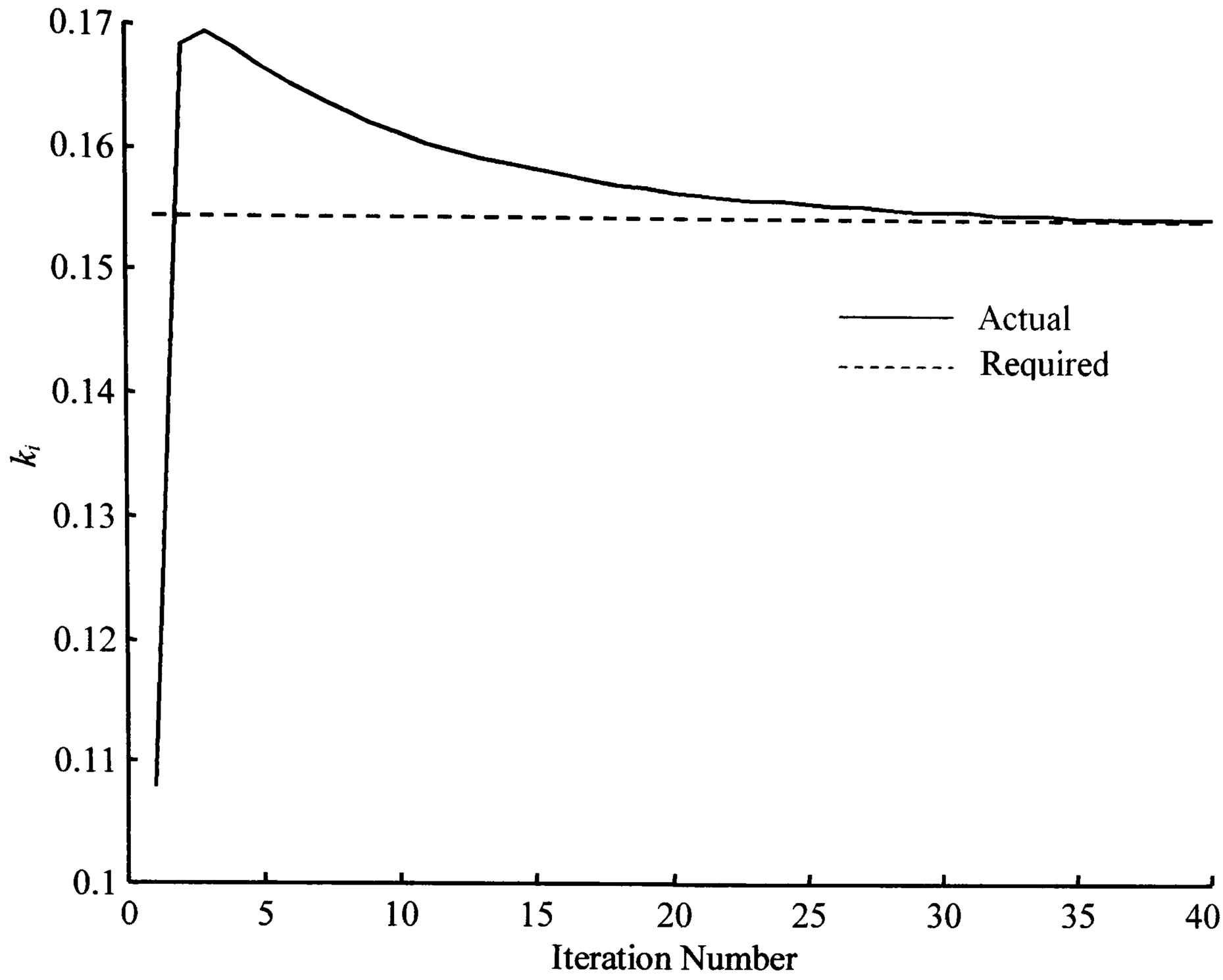


Figure 3.7: Evolution of PI controller parameter k_i

In the above the evolution of the controller parameters, and the resulting gain margin and phase margin, were shown. In Theorem 3.1 it was stated that the sufficient conditions for convergence are:

i) $\exists\ j_1 \geq 0$ such that $\forall\ j \geq j_1\ \ \mu_j$ satisfies $|\mu_j| < 1$.

ii) $|\mu_j| < 1$ for all $j = 0,1,\cdots,\infty$.

With the results of the simulation now available it will be possible to construct an estimate of μ for each iteration of the algorithm. Let the estimate of μ be given by

$$\hat{\mu}_n = \frac{\|K_* - K_{n+1}\|}{\|K_* - K_n\|} \leq \mu_n$$

where n is the iteration number. Knowing the value of K_* and K the estimate of μ can now be constructed. Figure 3.8 shows the evolution of μ with each iteration of the algorithm.

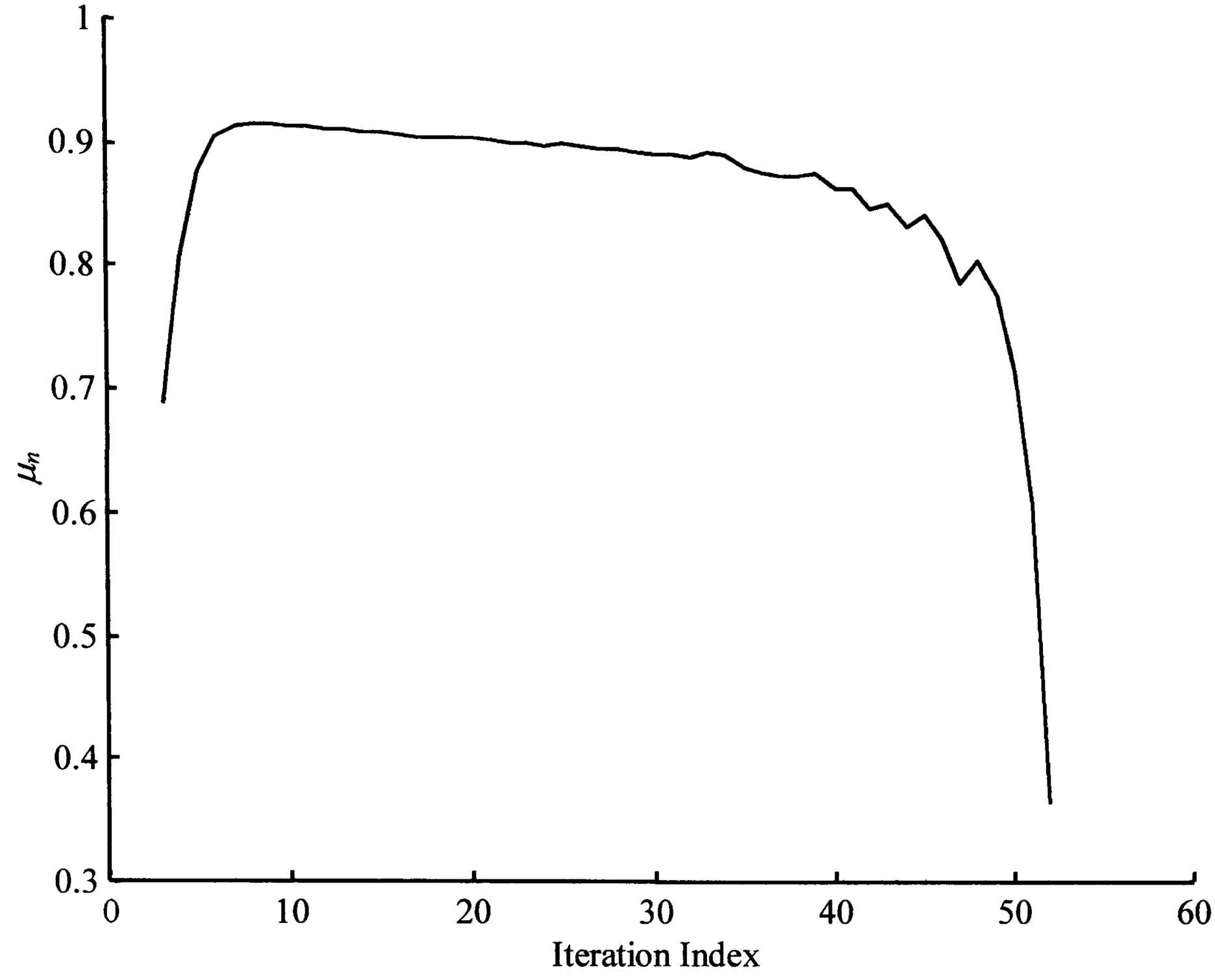


Figure 3.8: Evolution of μ_n.

From Figure 3.8 it can be seen that the estimate of μ satisfies both clause (a) and clause (b) of Theorem 3.1 for each iteration of the algorithm. It can also be seen from Figure 3.8 that there are more iterations shown than were shown in the convergence graphs for the controller parameters and the gain and phase margin graphs, this was carried out to ensure that the behaviour shown in Figure 3.8 was representative of the estimate of μ.

The closed loop step response and disturbance rejection properties of $G_1(s)$ in closed loop with the initial controller

$$G_{cI}(s) = 0.9372 + \frac{0.1077}{s}$$

and final controller $$G_{cF}(s) = 0.4265 + \frac{0.1545}{s}$$

are shown in Figure 3.9. A unit step is applied at time t = 0 (s) and a disturbance of magnitude 0.15 is applied at time t = 100 (s).

It can be seen that the initial controller gives a sluggish response with no overshoot whereas the final controller gives an under-damped response with an overshoot to the initial step of approximately 8%. However the response does have a shorter settling time and removes the disturbance effect more quickly than the initial *poorly* tuned controller.

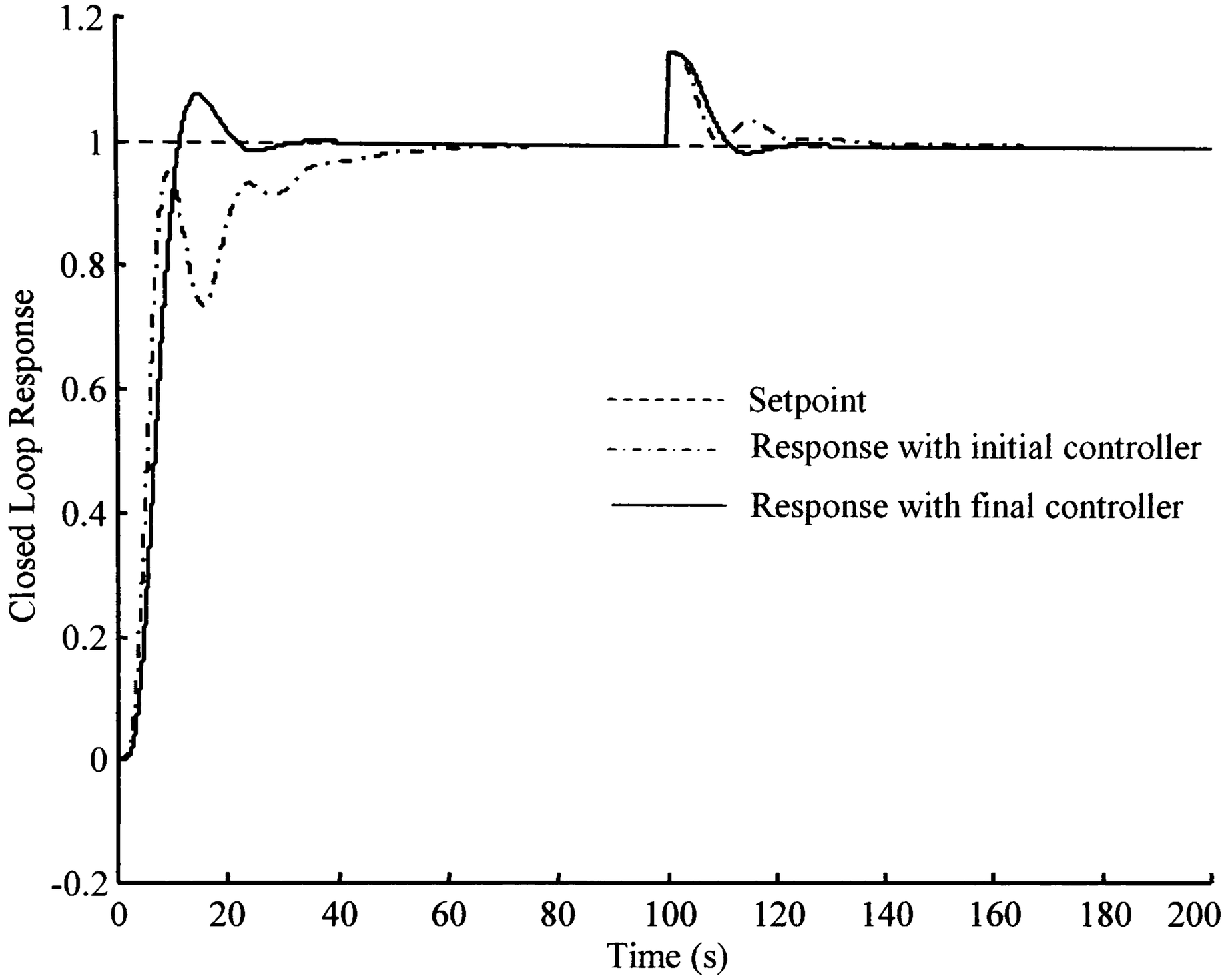


Figure 3.9: Closed loop response with initial and final PI controllers

Gain and phase margin specifications are better suited to a robustness specification rather than to a time domain closed loop response specification. Hence it is difficult to say, without knowing the initial tuning requirement, if the initial controller settings are better than the final settings as far as the time domain response of the closed loop system is concerned. However as far as the robustness of the closed loop system to process gain variations are concerned the final controller settings are much improved over the initial controller parameters.

Case Study 3.2

The initial controller settings for the following simulation example were based on Ziegler-Nichols tuning parameters for a PI controller with the data for the tuning rules supplied from a relay experiment. The initial and final PI controller tuning parameters are given in Table 3.2, along with the corresponding values of gain and phase margin and phase and gain crossover frequencies.

Table 3.2: Initial and achieved values for the compensated process $G_2(s)G_c(s)$

Initial Values					
k_p	k_i	Gain Margin	Phase Margin (Degrees)	Phase Crossover Frequency ($rad.s^{-1}$)	Gain Crossover Frequency ($rad.s^{-1}$)
7.064	1.938	2.33	92.0	1.2443	0.3046
Achieved Values					
3.5687	3.0777	3.0	60.0	1.0	0.3436

The progress of the gain and phase margin design method is shown in the following graphs for gain margin, phase margin, k_p, k_i and μ_n. The graphs are plotted to a base of algorithm iteration number. In order to relate the iteration number to the time taken to complete an iteration use is made of the following data:

iii) the average time taken t identify gain crossover point of the compensated forward path was 186(s).

iv) the average time taken t identify phase crossover point of the compensated forward path was 126(s).

Using this data and Algorithm 3.1, it can be seen that each iteration of the algorithm takes an average of 312(s). The settling time (2% criterion) for the process $G_2(s)$ is approximately 6(s) hence each iteration of the algorithm takes on average 52 settling time periods for this particular process.

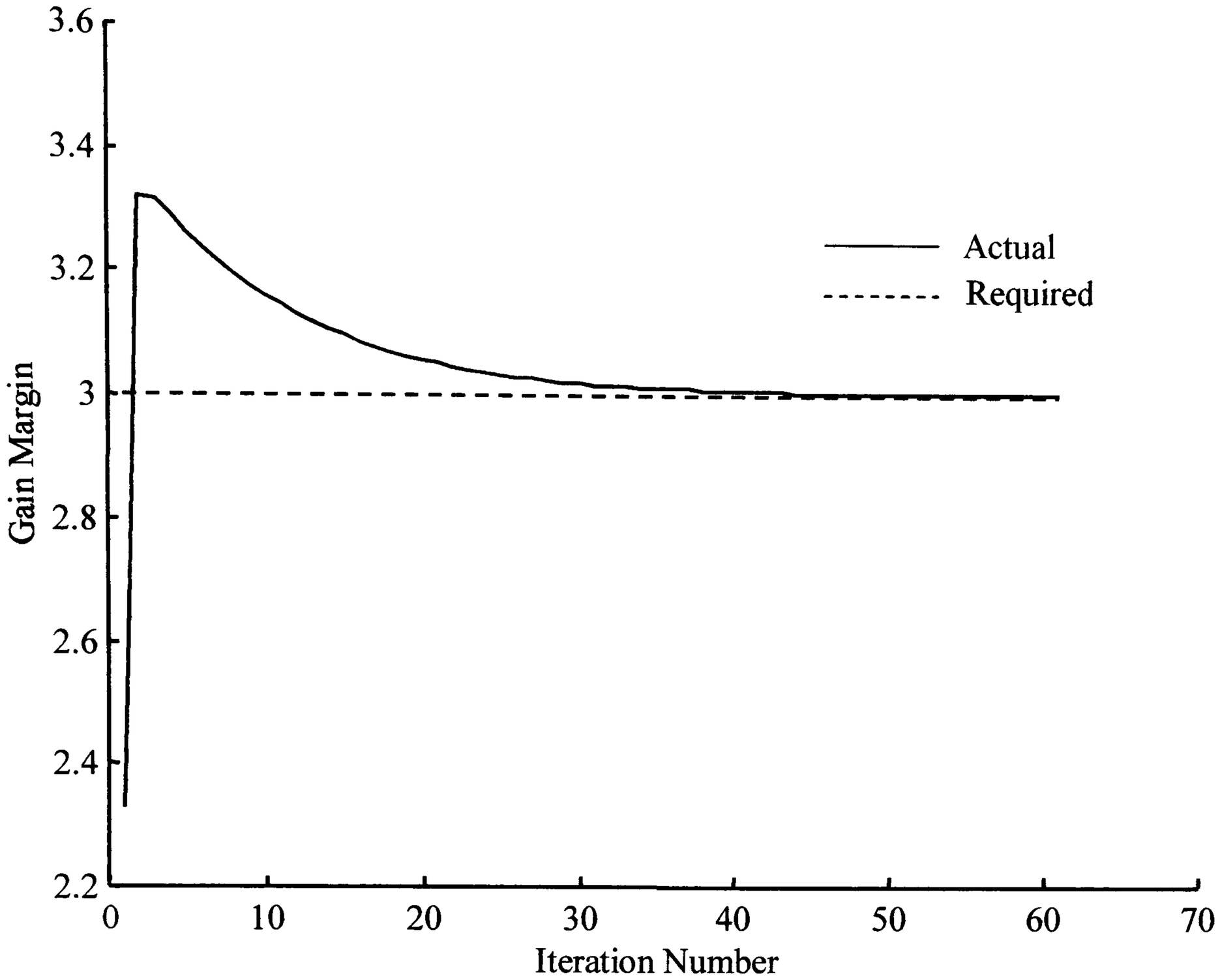


Figure 3.10: Evolution of the gain margin

The gain and phase margin for the design were chosen to be 3 and 60° respectively. From Figure 3.10 it can be seen that the gain margin is within 5% of the desired value within approximately ten iterations of the algorithm and within 2% after twenty-five iterations. Thus it can be seen that it takes a relatively large number of iterations to reach the desired value of gain margin

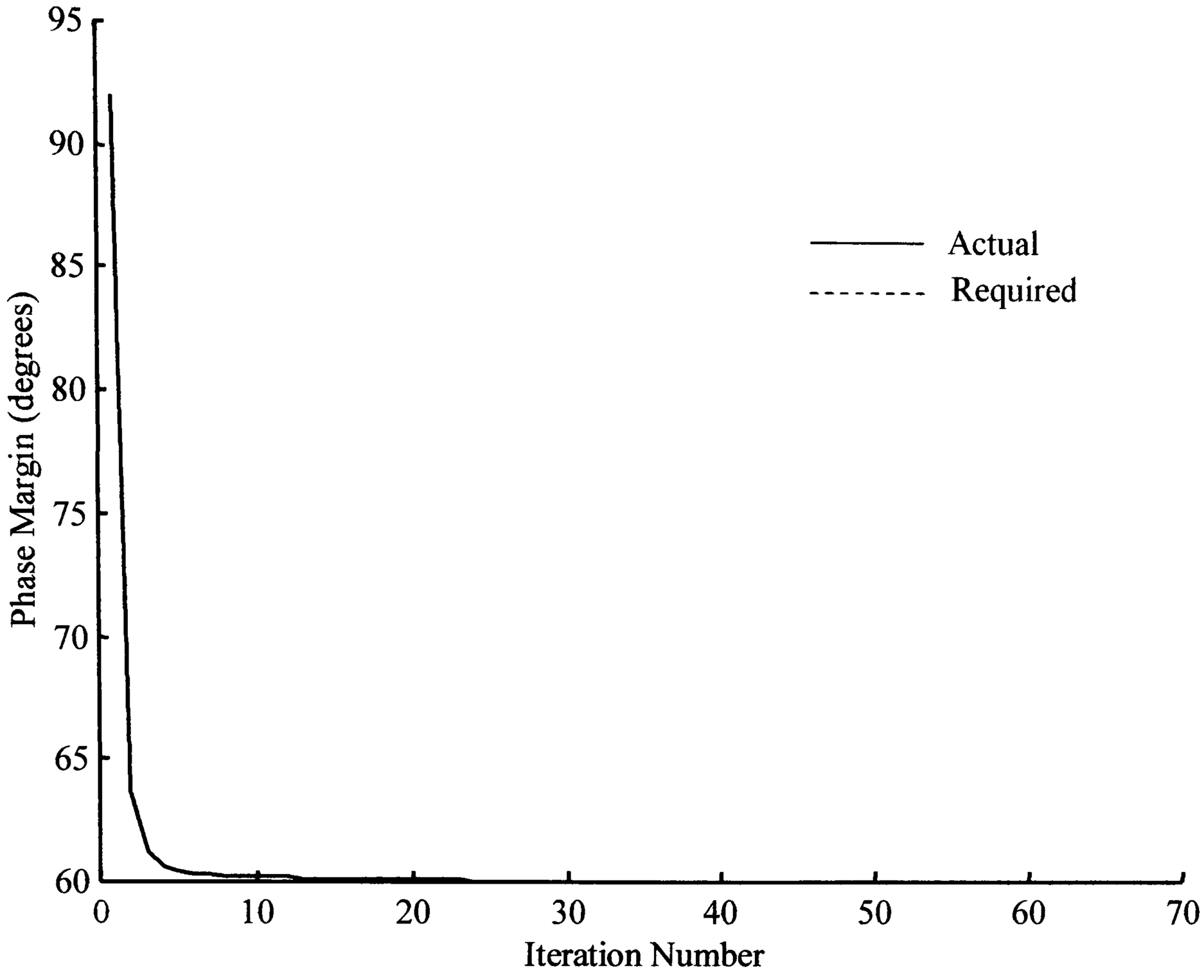


Figure 3.11: Evolution of the phase margin

Figure 3.11 shows that the phase margin is practically achieved after approximately five iterations. The evolution of the controller parameters is given in Figures 3.11 and 3.12 for k_p and k_i respectively.

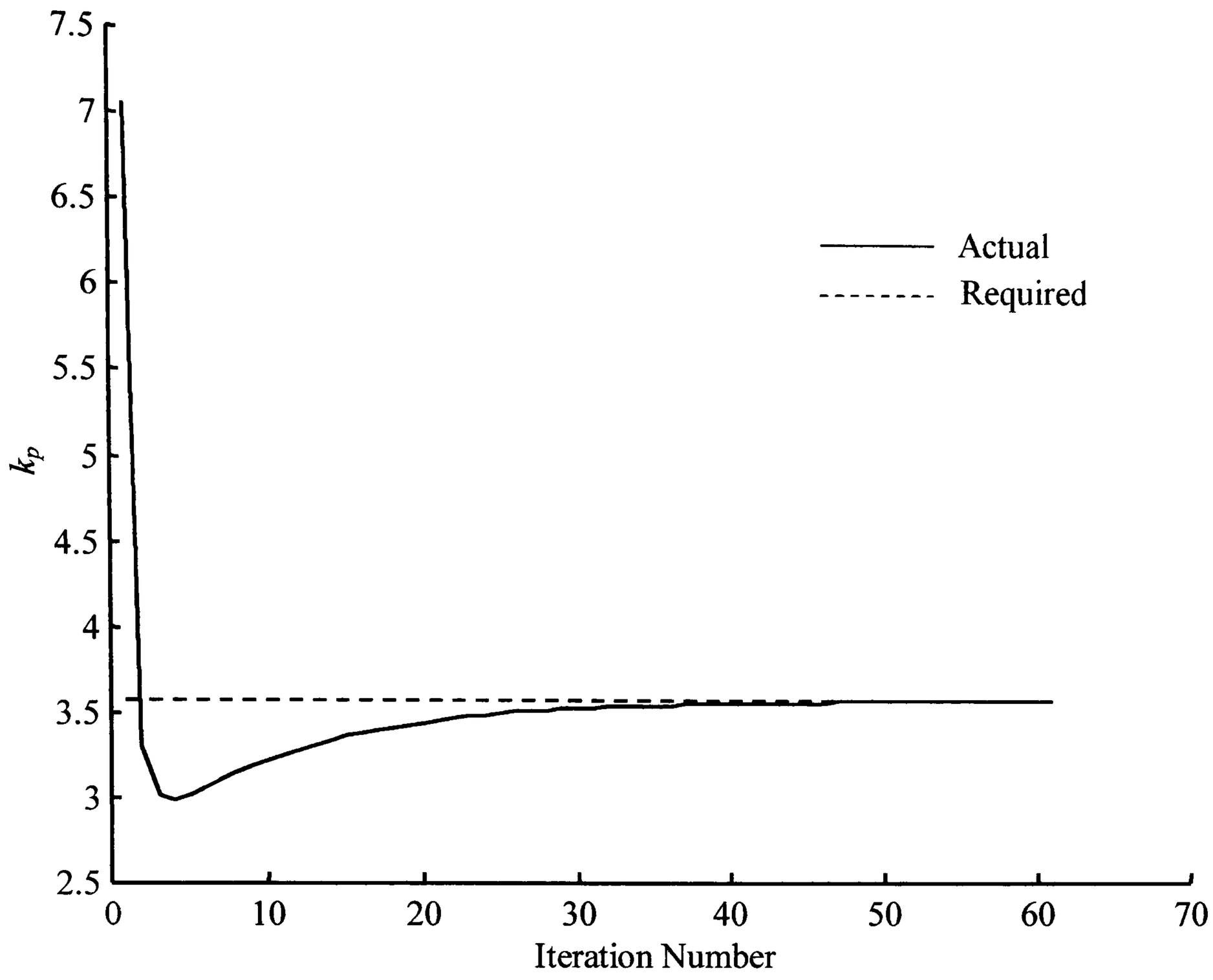


Figure 3.12: Evolution of the PI controller parameter k_p

As can be seen from Figure 3.12 there is an initial large change in the value of k_p followed by a very much slower change as the algorithm moves the controller parameter towards the converged value.

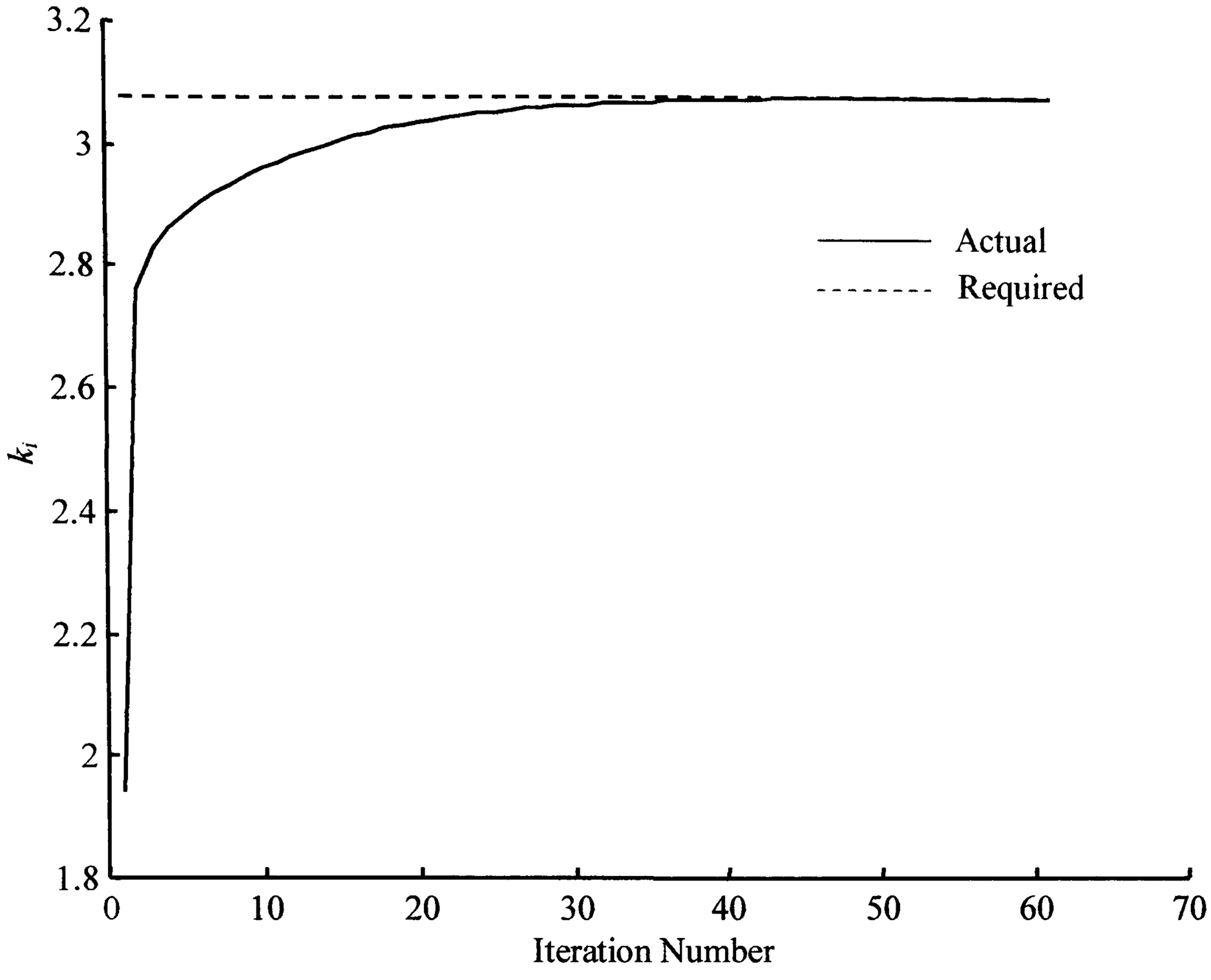


Figure 3.13: Evolution of the PI controller parameter k_i

From Figure 3.13 it can be seen that after an initial large change in the value of k_i, this is followed by a relatively slow change towards the converged value.

In a similar fashion to the previous case study the estimate of the parameter μ was constructed and is shown in Figure 3.14.

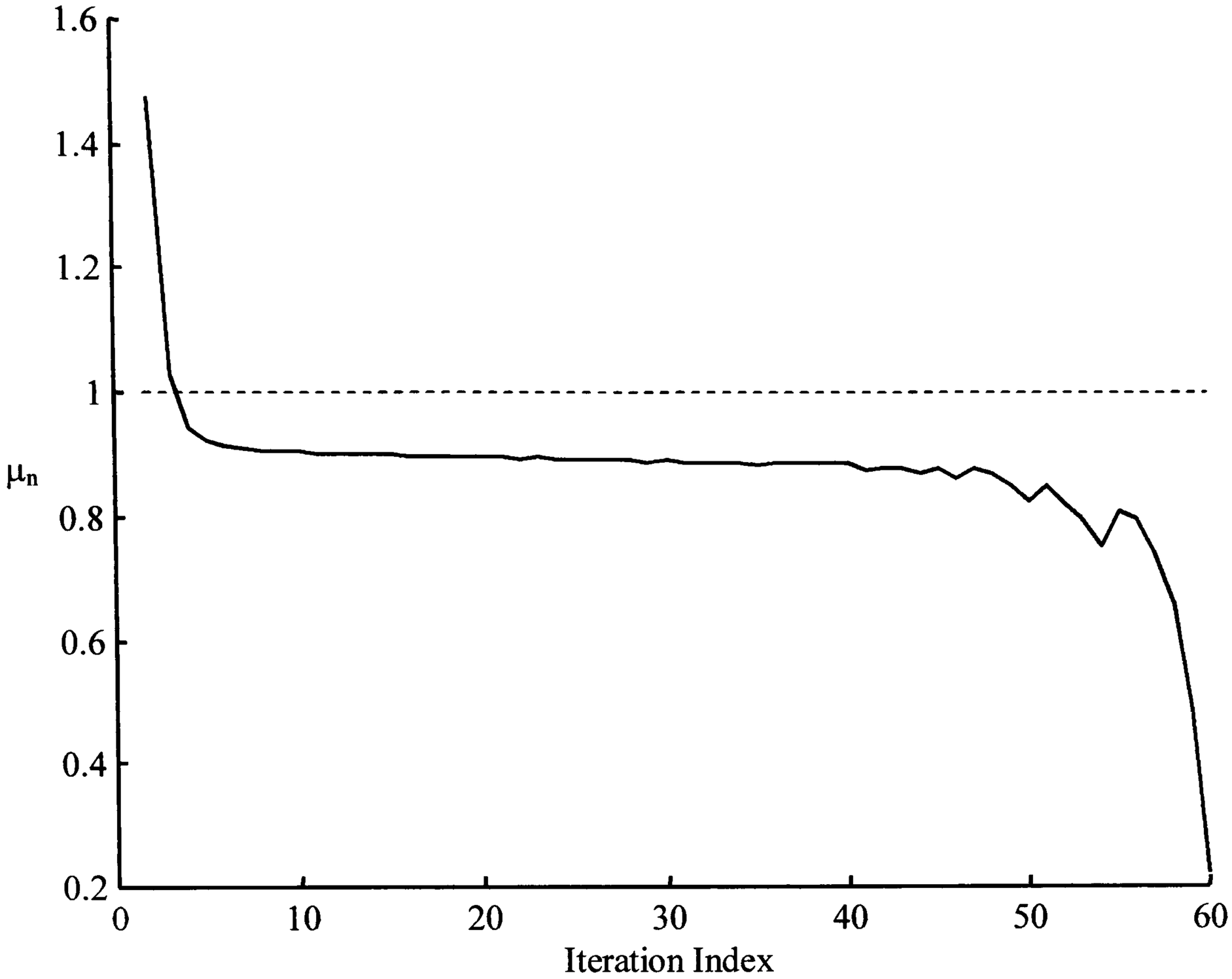


Figure 3.14: Evolution of μ_n.

From Figure 3.14 it can be seen that the evolution of the estimate of μ supports clause (a) of Theorem 3.1. Since after an initial few iterations where, $\hat{\mu}_n > 1$, after iteration five it can be seen that $\hat{\mu}_n < 1$.

By referring to Figure 3.15 it can be seen that the step response of the closed loop system with the final controller is improved over that obtained with the initial controller, as is the disturbance rejection property. As noted above it is difficult to compare the time domain responses of the initial and final closed loops since the design specifications are more to do with the stability robustness and robustness to process change properties than the time domain step or disturbance rejection properties of the two closed loops.

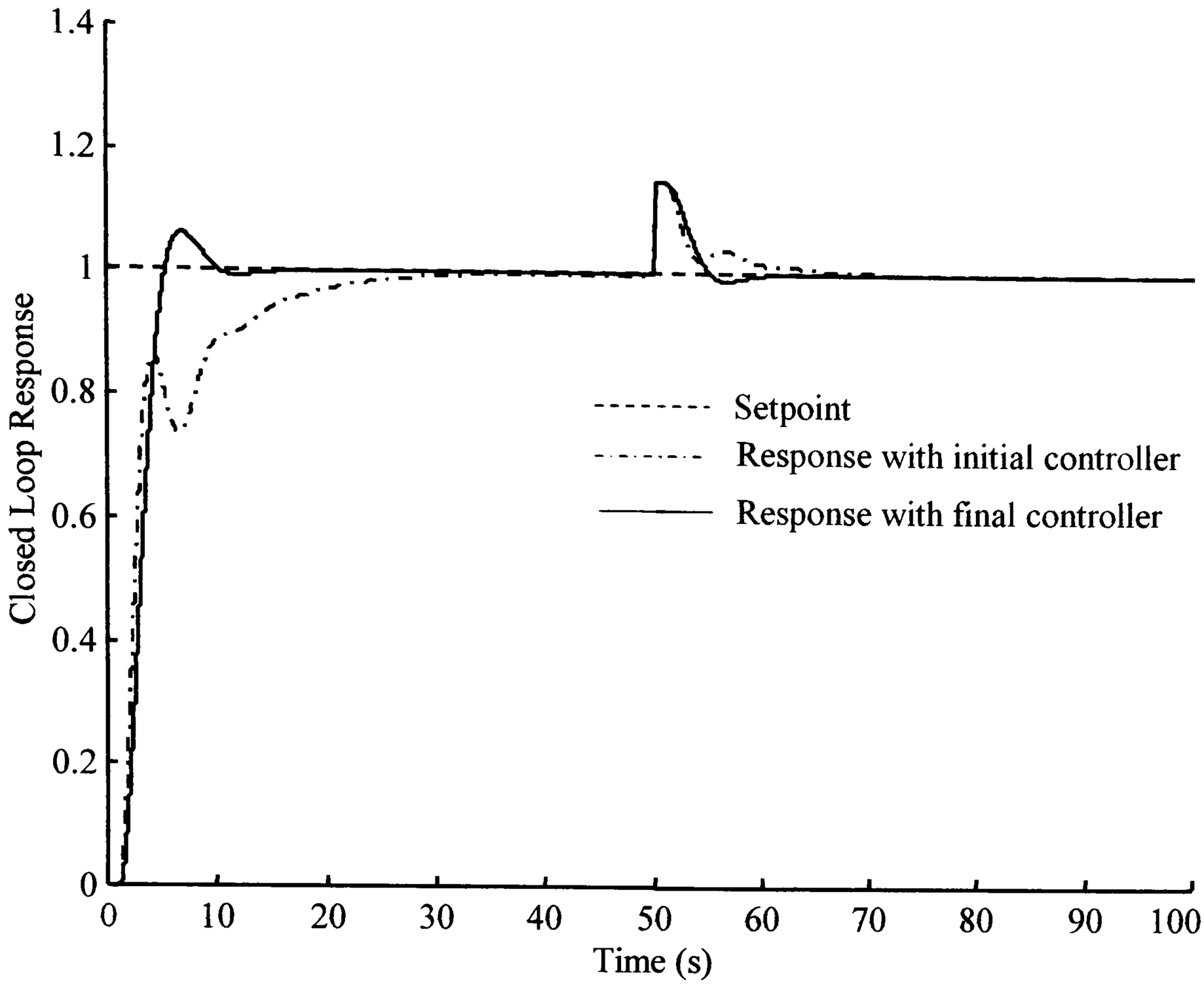


Figure 3.15: Closed loop response with initial and final PI controllers

The closed loop response of the final controller has less overshoot and an improved disturbance rejection over that of the initial controller. As can be seen from Table 3.2 there is an improvement in both the gain margin and phase margin of the compensated system using the final controller over that using the initial controller.

3.4 Maximum Sensitivity and Phase Margin: Automated PI Controller Design

In the automated PI controller design method to follow, phase margin and maximum sensitivity specifications will be given and data sought for the unknown process at specific frequencies. To find these frequencies, the Phase-Locked Loop identification method shown previously in Figure 3.1 shall be utilised according to the identification case. However, the identification always retains the existing PID controller in the loop, and uses a parallel computation to place specifications on the

new compensated forward path, $G_{fp}(s)=G_p(s)G_{nc}(s)$ to find the required frequency, where $G_{nc}(s)$ is a newly calculated controller.

Using equations (3.14) and (3.8) in the maximum sensitivity and phase margin design equations leads to a set of generic equations for the new PI compensator gains, k_p and k_i ,

$$\begin{bmatrix} G_{pR}(\omega_s) & \dfrac{G_{pI}(\omega_s)}{\omega_s} \\ G_{pI}(\omega_s) & -\dfrac{G_{pR}(\omega_s)}{\omega_s} \\ G_{pR}(\omega_1) & \dfrac{G_{pI}(\omega_1)}{\omega_1} \\ G_{pI}(\omega_1) & -\dfrac{G_{pR}(\omega_1)}{\omega_1} \end{bmatrix} \begin{bmatrix} k_p \\ k_i \end{bmatrix} = \begin{bmatrix} -\left(1-\dfrac{1}{M_S}\cos\theta_S\right) \\ -\dfrac{1}{M_S}\sin\theta_S \\ -\cos\phi_{PM} \\ -\sin\phi_{PM} \end{bmatrix} \tag{3.25}$$

With obvious notational identification this set of equations can be written compactly as,

$$[X_n]K = Y^D \tag{3.26}$$

The equations (3.25), (3.26) form the PI design component of the full algorithm. In the next section, the steps taken to ensure a viable sequence of iterates are explained and finally the full algorithm is given.

3.4.1 Phase Margin and Maximum Sensitivity Design Theory and Algorithm

In general there are an infinite number of values that θ_S can take between an upper and lower bound. First experimental trials with the automated algorithm lead to a realisation that it was essential to generate a sequence of points that bounded at least a subset of the possible values of θ_S. To ensure that this occurred, it was necessary to limit the size of updates that could be taken by the algorithm. Consequently, the algorithm was re-constructed so that the locus of $G_{fp}(j\omega)$ intersected the $1/M_S$ circle at *two* points, as in Figure 3.3. To ensure that this occurred, the term $1/M_S$ in the equation suite, (3.25) was replaced with a reduced value *d* given by,

$$d = \kappa_T \sqrt{\frac{1}{M_S}} \tag{3.27}$$

where $\kappa_T = \sqrt{\left(1 - \left|G_{fp}(j\omega_s)\right|\cos(\pi + \phi)\right)^2 + \left(\left|G_{fp}(j\omega_s)\right|\sin(\pi + \phi)\right)^2}$

and $\phi = \arg(G_p(j\omega_k))$ when

$$\hat{\theta}_S = \tan^{-1}\left(\frac{\left|\mathrm{Im}\left(G_{fp}(j\omega_k)\right)\right|}{1 - \left|\mathrm{Re}\left(G_{fp}(j\omega_k)\right)\right|}\right) \tag{3.28}$$

and ω_S is the frequency at which (3.28) occurs. The geometric mean for d is taken although the geometric mean could be replaced by a weighting factor on $1/M_S$. Using estimates for θ_S and d the equation set (3.25) was reformulated as,

$$\begin{bmatrix} G_{pR}(\omega_s) & \dfrac{G_{pI}(\omega_s)}{\omega_s} \\ G_{pI}(\omega_s) & -\dfrac{G_{pR}(\omega_s)}{\omega_s} \\ G_{pR}(\omega_1) & \dfrac{G_{pI}(\omega_1)}{\omega_1} \\ G_{pI}(\omega_1) & -\dfrac{G_{pR}(\omega_1)}{\omega_1} \end{bmatrix} \begin{bmatrix} k_p \\ k_i \end{bmatrix} = \begin{bmatrix} -\left(1 - d\cos\hat{\theta}_S\right) \\ -d\sin\hat{\theta}_S \\ -\cos\phi_{PM} \\ -\sin\phi_{PM} \end{bmatrix} \tag{3.29}$$

With this constructional device in place, the full algorithm description can be given.

Algorithm 3.2: Automated PI Controller Design for Desired Maximum Sensitivity and Phase Margin Specifications.

Step 1: Design Specification.

Select the required maximum sensitivity M_S and phase margin ϕ_{PM} for the controller design.

Step 2: Initialisation Step.

Initialise counter $n = 0$

Choose the initial PI controller gains: $k_p(0)$ and $k_i(0)$

Set the convergence tolerance, *tol*

Step 3: Identification Step.

Step 3a: Gain Crossover Frequency Identification.

Use the Phase-Locked Loop to find $\omega_1(n)$ for the forward path $G_{fp}(s) = G_p(s)G_{nc}(s)$

Use $k_p(n)$, $k_i(n)$ to solve for $G_{pR}(\omega_1)$ and $G_{pI}(\omega_1)$

Step 3b: Maximum Sensitivity Frequency Identification.

Use the Phase-Locked Loop to identify $\omega_A(n)$ and $\omega_B(n)$ for the forward path $G_{fp}(s) = G_p(s)G_{nc}(s)$

Use $k_p(n)$ and $k_i(n)$ to solve for $G_{pR}(\omega_A)$, $G_{pI}(\omega_A)$, $G_{pR}(\omega_B)$ and $G_{pI}(\omega_B)$

Calculate $\theta_1(\omega_A)$, $\theta_2(\omega_B)$ and $\hat{\theta}_S(\theta_1, \theta_2)$

Use Phase-Locked Loop with $\hat{\theta}_S(\theta_1, \theta_2)$ to identify $\omega_S(n)$.

Calculate d using equation (3.27) and $\omega_s(n)$.

Step 3c: Convergence Test Step.

Compute $Y(n), Y^D$ where,

$$Y(n) = [-(1 - d\cos\hat{\theta}_S(n)), -d\sin\hat{\theta}_S(n), -\cos\phi_{PM}(n), -\sin\phi_{PM}(n)]^T$$

$$Y^D = [-(1 - (1/M_S)\cos\hat{\theta}_S(n)), -(1/M_S)\sin\hat{\theta}_S(n), -\cos\phi_{PM}, -\sin\phi_{PM}]^T$$

If $\|Y^D - Y(n)\| < tol$ then STOP.

Step 4: Controller Update Calculation.

Use ω_S, ω_1, $G_{pR}(\omega_S)$, $G_{pI}(\omega_S)$, $G_{pR}(\omega_1)$ and $G_{pI}(\omega_1)$

Form $[X_n]K = Y^D$

Solve as $K = [X_n^T X_n]^{-1} X_n^T Y^D$

Update n:= n + 1, $k_p(n+1) = K_1$, $k_i(n+1) = K_2$

Go to Step 3 ●

Remarks

i) The controller initialisation step can sometimes use the existing closed loop controller parameters.

ii) The main complexity of the algorithm resides in the inner iterative routines being run to find the non-parametric data for the process. These routines are subject to inner convergence tolerances. If these tolerances

are too stringent, then the Phase-Locked Loop module is slow to converge. Experience has shown that low values of convergence accuracy give satisfactory solutions.

3.4.2 Convergence Theorem

The algorithm for the design of a PI controller that achieves a desired maximum sensitivity and phase margin was discussed in the above section. A theorem and associated proof is proposed in this section that shows that under a relatively weak set of conditions that the algorithm will provide controller gains such that convergence to a fixed point solution of the equation set (3.25) is obtained.

The forward path transfer function of the compensated system is given by

$$G_{fp}(s) = G_p(s)G_{nc}(s)$$

where the new controller is given by

$$G_{nc}(s) = k_p + \frac{k_i}{s}$$

From the equation for the new controller it can be seen that the independent variables of the problem are k_p and k_i. Hence define

$$K = \begin{bmatrix} k_p \\ k_i \end{bmatrix}$$

where $K \in \Re_+^2$

For a given $K \in \Re_+^2$ the Phase-Locked Loop identifier is used to find

$$\omega_s(k_p, k_i)$$

and

$$\omega_1(k_p, k_i)$$

This is written as

$$\omega_v(K) = \begin{bmatrix} \omega_s \\ \omega_1 \end{bmatrix}$$

where $\omega_v(K): \Re_+^2 \to \Re_+^2$ and $\omega_v(K)$ is non-linear.

The next step is to calculate an updated value of K using least squares, hence define

$$X(\omega_v) = \tilde{X}(\omega_s, \omega_1)$$

where $X(\omega_v): \Re_+^2 \to \Re^2$. The updated controller gains are thus found from

$$K_{n+1} = \left[X^T(\omega_v)X(\omega_v)\right]^{-1} X^T(\omega_v)Y^D$$
$$= f(\omega_v) = f(\omega_v(K_n)) = g(K_n)$$

where $f: \Re_+^2 \to \Re^2$, $\omega_v: \Re_+^2 \to \Re_+^2$ and $g: \Re_+^2 \to \Re^2$.

Theorem 3.2: Sufficient Conditions for Convergence

Assuming that suitable problem set-up assumptions are valid, then for the maximum sensitivity and phase margin PI tuning algorithm the basic convergence relationship is

$$\|K_* - K_{n+1}\| \leq \left(\prod_{j=0}^{n} \mu_j\right) \|K_* - K_0\|$$

where $$\mu_j = \sup_{0<\alpha_j<1} \left\| g'\left(K_j + \alpha_j\left(K_* - K_j\right)\right)\right\|$$

The sufficient conditions for the convergence are:

(a) $\exists\ j_1 \geq 0$ such that $\forall\ j \geq j_1\ \ \mu_j$ satisfies $|\mu_j| < 1$.

(b) $|\mu_j| < 1$ for all $j = 0,1,\cdots,\infty$.

Proof

The proof follows exactly that given for Theorem 3.1 of section 3.3.1 of this chapter.

3.4.3 Implementation and Case Study Results

A prototype auto-tuner was implemented using Matlab™, and the simulation results for two candidate processes are reported. The candidate processes are given by

$$G_1(s)=\frac{1}{(s+1)^6} \text{ and } G_2(s)=\frac{e^{-s}}{(s+1)(s+3)^2}$$

The first process, $G_1(s)$, represents a broad class of systems found in the process control industries that are characterised by having a high order and non-oscillatory response. The second process to be studied, $G_2(s)$, represents non-oscillatory processes with time delays. The justification of the choice of model to demonstrate the tuning method has been discussed previously in section 3.3.2 of this thesis.

Case Study 3.3

The process to be used in this study is given by

$$G_1(s)=\frac{1}{(s+1)^6}$$

The desired phase margin and maximum sensitivity design that is to be attained is 60° and 1.7 respectively. The initial controller parameters were derived from the Ziegler-Nichols rules for tuning a PI controller and the data required by the method was obtained from the results of a relay experiment. Table 3.3 details the values of

i) The controller tuning parameters

ii) The maximum sensitivity and

iii) Phase margin and tangency angle

for the initial and final values of the design.

Table 3.3: Initial and converged controller results

k_p	k_i	Maximum Sensitivity	Phase margin (degrees)	Tangency angle θ_s (deg)
Process $G_1(s)G_c(s)$: Initial Values				
0.9372	0.1077	1.97	85.5	-
Process $G_1(s)G_c(s)$: Achieved Values				
0.4671	0.1591	1.70	60.0	18.0

The initial tuning of the PI controller, $G_c(s)$, resulted in the algorithm having to start relatively far from the desired design values of maximum sensitivity and gain

margin. The aim of the design was to achieve a controller that gives the closed loop system good stability robustness and disturbance rejection properties, hence the choice of design phase margin and maximum sensitivity, respectively.

The progress of the gain and phase margin design method is shown in the following graphs for maximum sensitivity, phase margin, k_p and k_i. The graphs are plotted to a base of algorithm iteration number. In order to relate the iteration number to the time taken to complete an iteration use is made of the following data:

v) the average time taken to identify the gain crossover point of the compensated forward path was 325(s).

vi) the average time taken to identify the intersection(s) of the required maximum sensitivity and the compensated forward path was 379(s).

Using this data and Algorithm 3.2, it can be seen that each iteration of the algorithm takes an average of 704(s). The settling time (2% criterion) for the process $G_1(s)$ is approximately 12(s) hence each iteration of the algorithm takes on average 58 settling time periods for this particular process.

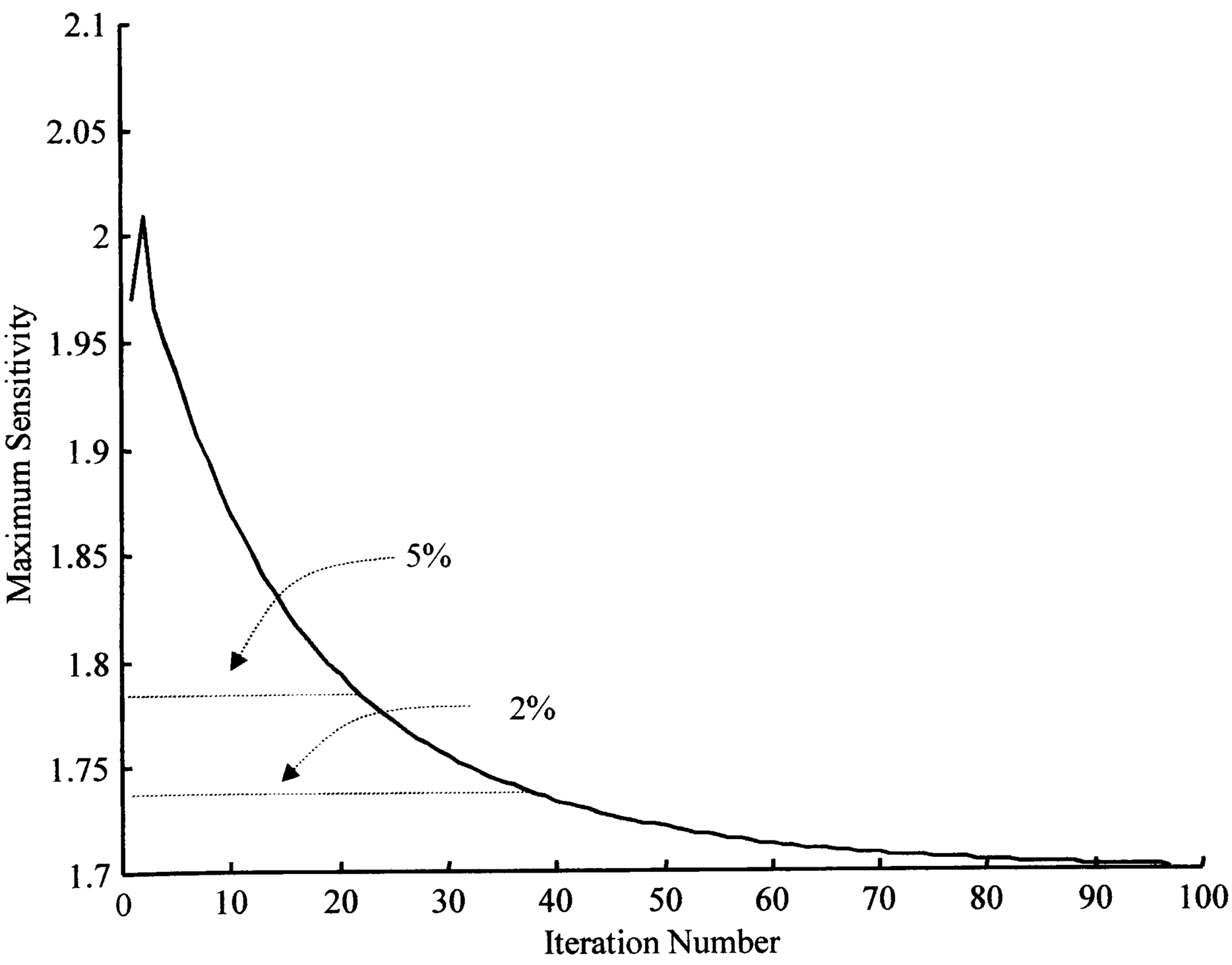


Figure 3.16: Evolution of the maximum sensitivity

Figure 3.16 shows how the maximum sensitivity of the new compensated forward path evolves during the progress of the algorithm. It can be seen that the design maximum sensitivity is within 5% of the desired value after 22 iterations and within 2% after 37 iterations of the algorithm. However, as can be seen from Figure 3.16, it takes a large value of further iterations to achieve the *exact* value required.

Figure 3.17 shows how the phase margin of the compensated forward path evolves during the progress of the algorithm. It can be seen that the required phase margin is *exactly* attained after approximately 65 iterations of the algorithm. This is a recurring feature of the results obtained using this algorithm, in that the required phase margin is *achieved* more rapidly than is the maximum sensitivity.

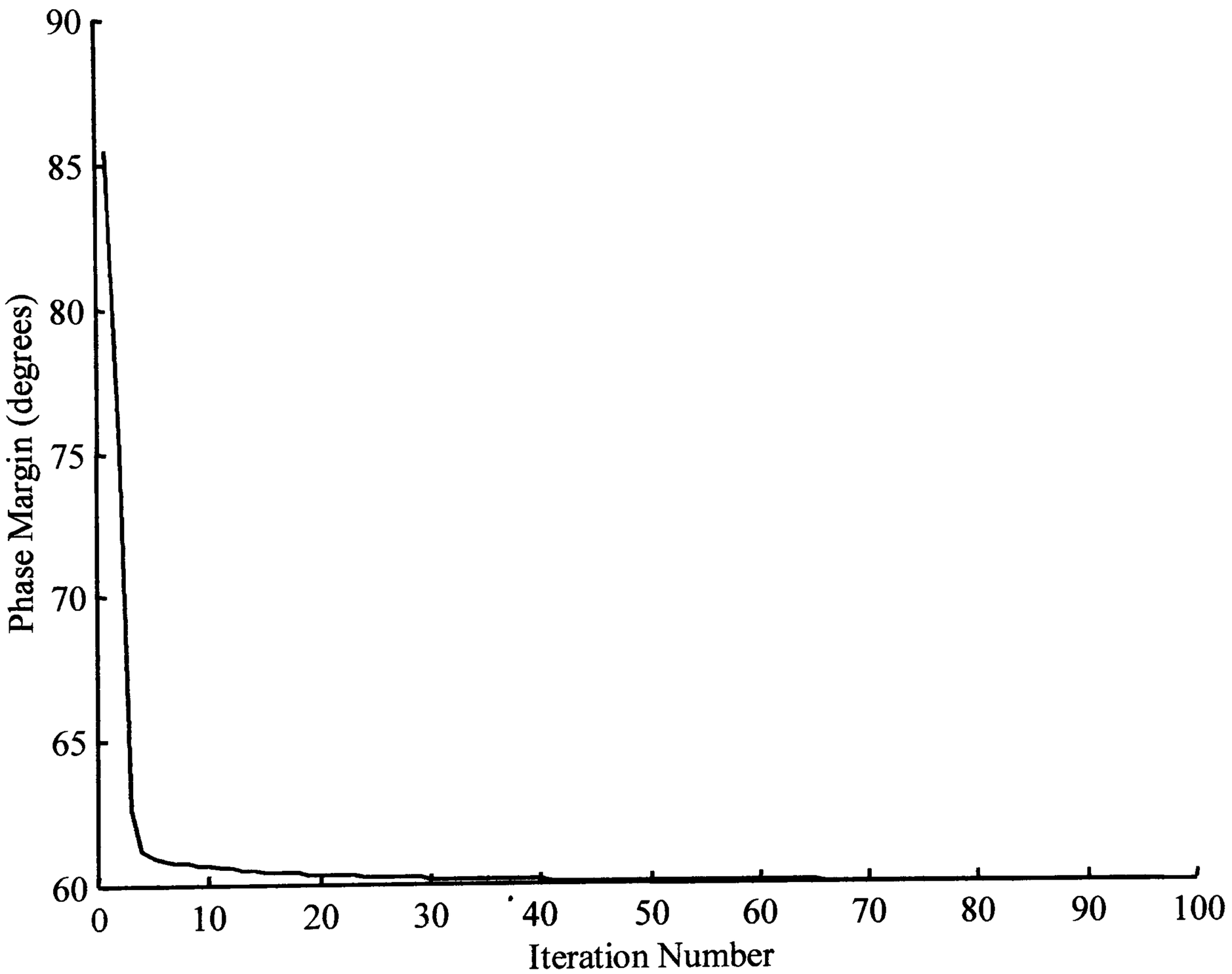


Figure 3.17: Evolution of the phase margin

Figures 3.16 and 3.17 are typical for the algorithm convergence showing a large number of iterations required to achieve the design values to a high degree of

accuracy. Figures 3.18 and 3.19 show how the controller parameters evolve with each iteration of the algorithm.

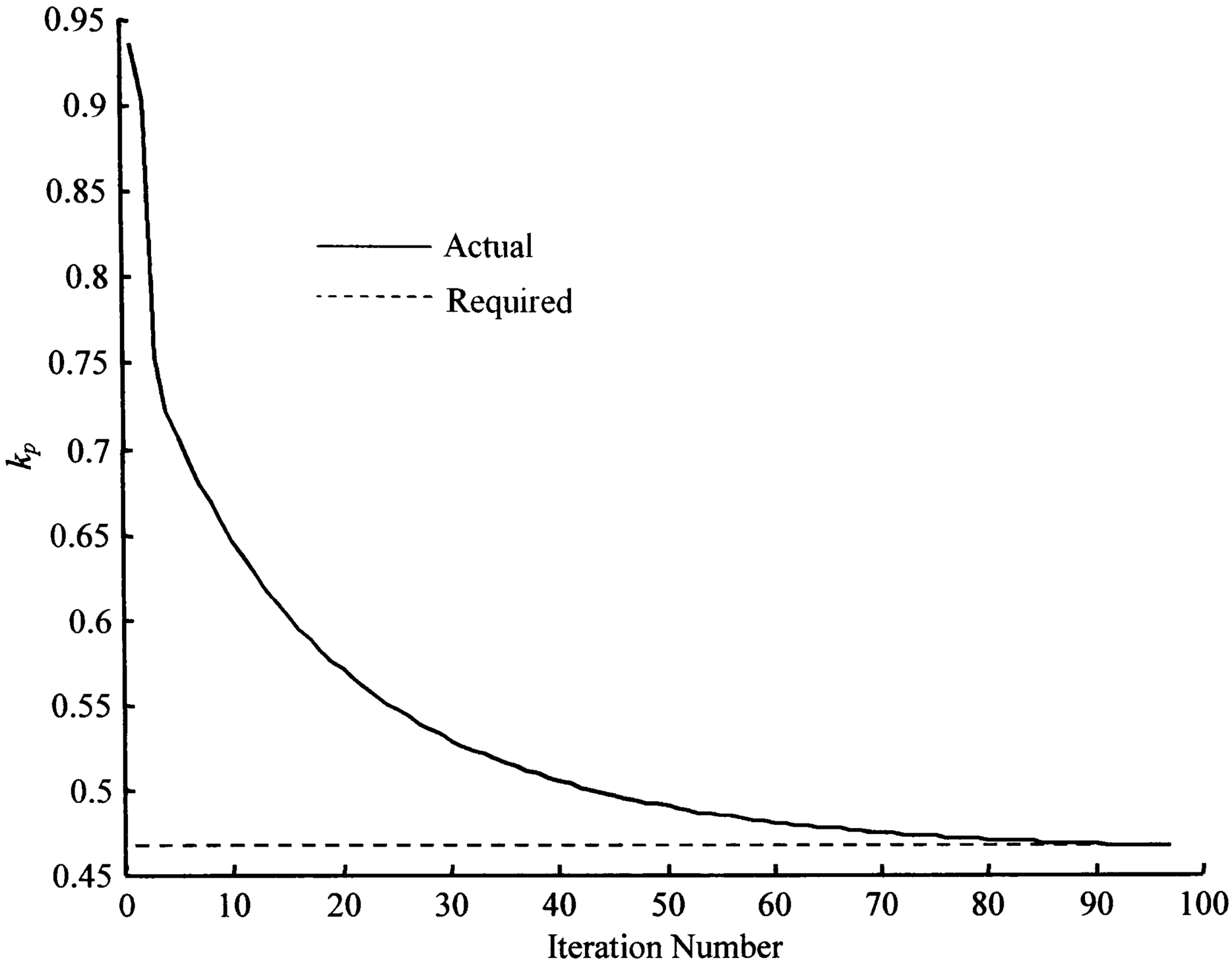


Figure 3.18: Evolution of controller parameter k_p

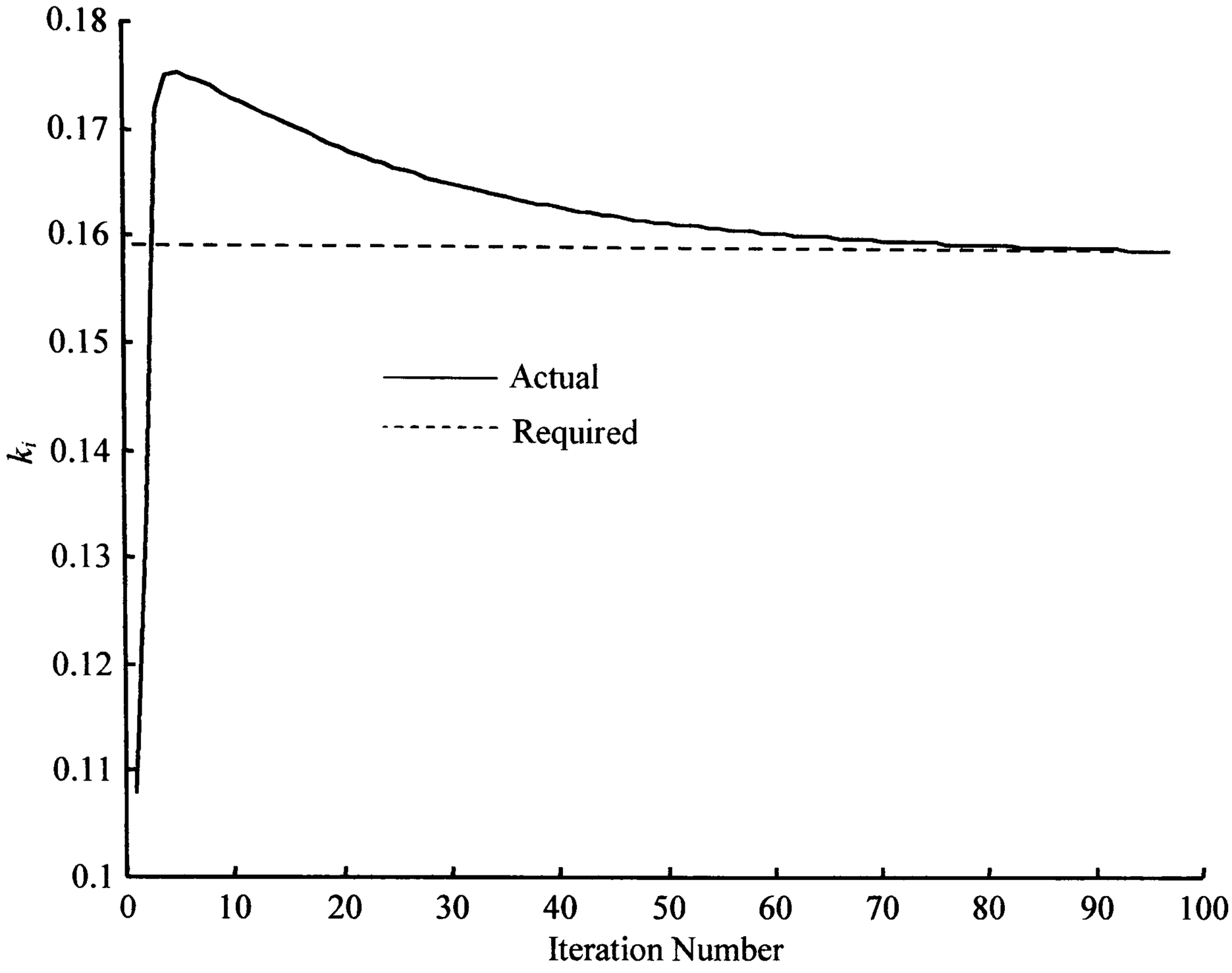


Figure 3.19: Evolution of the controller parameter k_i

In Figure 3.20 it is possible to compare the step and disturbance rejection performances of the initial and final PI controllers in closed loop with the process $G_1(s)$. A unit step is applied at time t = 0 (s) and the disturbance of magnitude 0.15 is applied at time t = 70 (s). As would be expected for this design the set point tracking capability of the designed controller, although acceptable for process control, could be improved upon. However the results do confirm, in this case, that if the maximum sensitivity is reduced then the closed loop response to a step input becomes less oscillatory with a reduced overshoot; as stated in Astrom *et al* (1999). The disturbance rejection is seen to be acceptable.

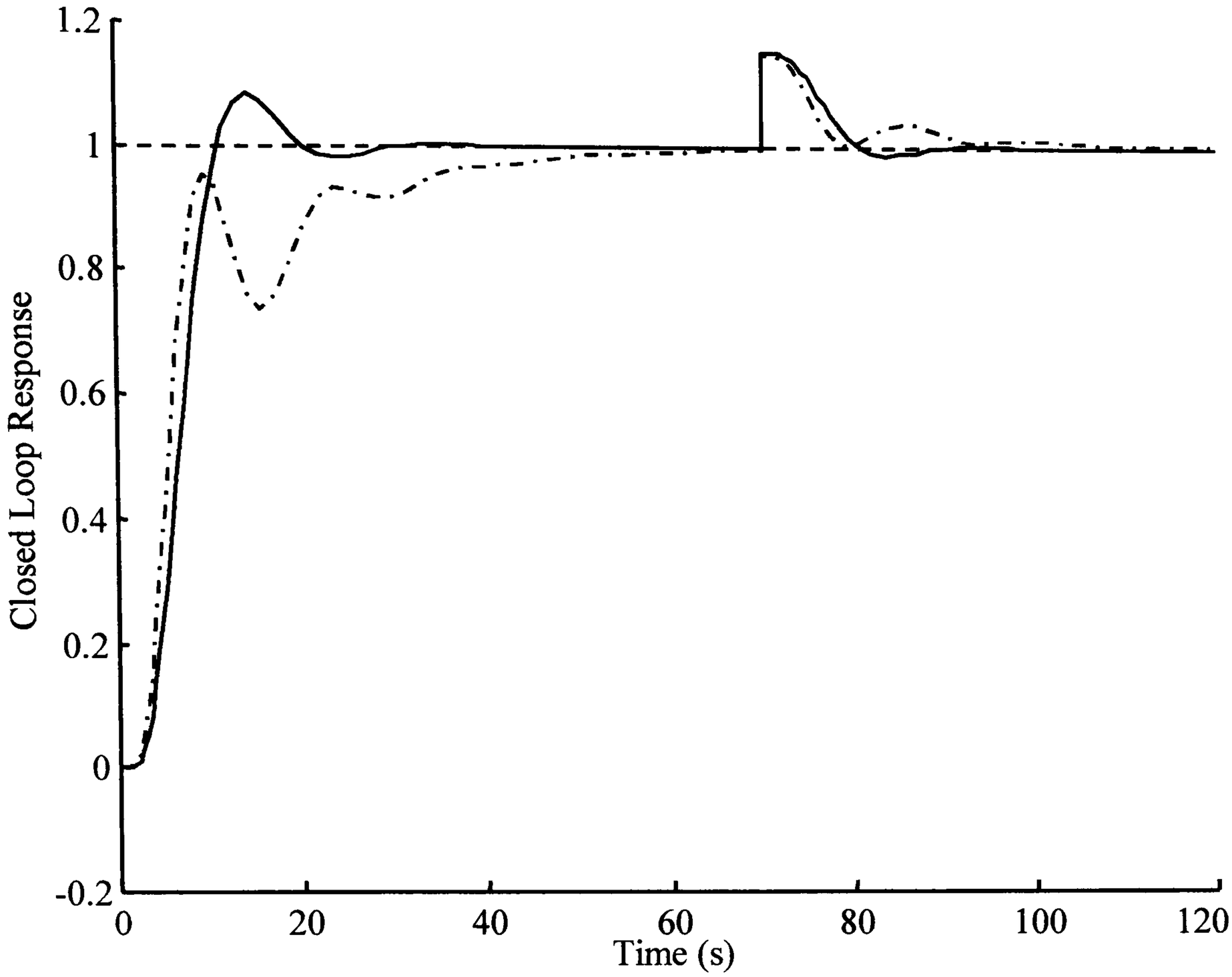


Figure 3.20: Closed loop response of the initial and final PI controllers

By reference to Figure 3.21 it can be seen that the desired design phase margin and maximum sensitivity have been achieved. The frequency range shown for the forward path of the system is 0.1 to 10 (rad.s^{-1}).

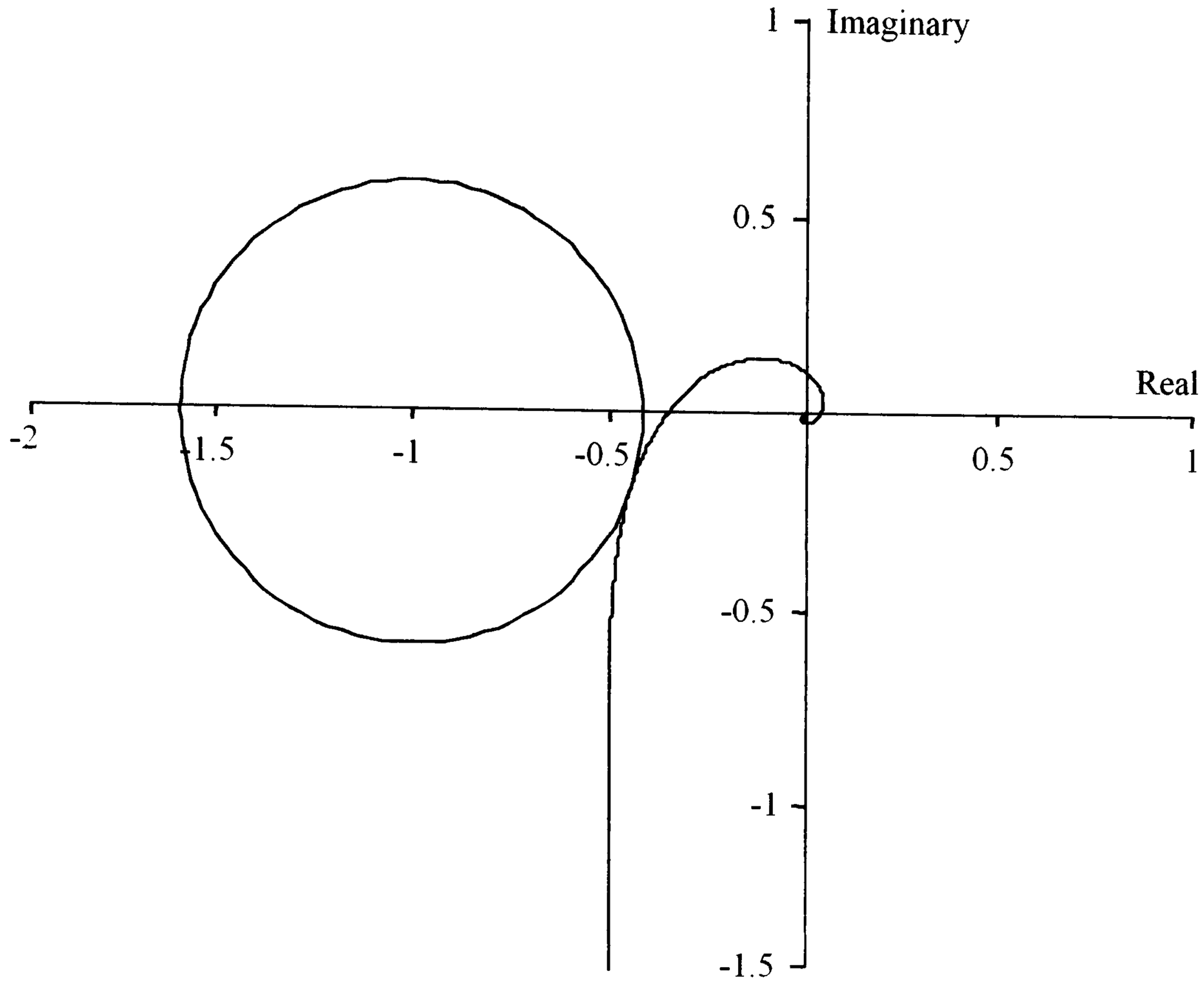


Figure 3.21: Nyquist diagram showing the forward path of the compensated process

Case Study 3.4

The process used for this case study is given by

$$G_2(s) = \frac{e^{-s}}{(s+1)(s+3)^2}$$

Process $G_2(s)$ was chosen since it is representative of processes found in the process industry that can be characterised as having a time delay and are non-oscillatory. The initial values for the parameters of the controller $G_c(s)$ were chosen so that the algorithm would be required to start from a relatively large distance from the final values. Table 3.4 details the initial and final values of the controller parameters and the phase margins and maximum sensitivities for the design. The desired maximum sensitivity specification and phase margin for this design are 1.7 and 60° respectively.

Table 3.4: Initial and converged controller results				
k_p	k_i	Maximum Sensitivity	Phase margin (deg)	Tangency angle θ_s (deg)
Process $G_2(s)G_c(s)$: Initial Values				
7.064	1.938	1.82	92.0	-
Process $G_2(s)G_c(s)$: Achieved Values				
4.1478	3.2371	1.70	60.0	16.9

The aim of the design was to achieve a controller that gives the closed loop system good stability robustness and disturbance rejection properties hence the choice of design phase margin and maximum sensitivity, respectively.

The progress of the gain and phase margin design method is shown in the following graphs for phase margin, maximum sensitivity, k_p and k_i. The graphs are plotted to a base of algorithm iteration number. In order to relate the iteration number to the time taken to complete an iteration use is made of the following data:

vii) the average time taken to identify the gain crossover point of the compensated forward path was 231(s).

viii) the average time taken to identify the intersection(s) of the required maximum sensitivity and the compensated forward path was 264(s).

Using this data and Algorithm 3.2, it can be seen that each iteration of the algorithm takes an average of 495(s). The settling time (2% criterion) for the process $G_2(s)$ is approximately 6(s) hence each iteration of the algorithm takes on average 82 settling time periods for this particular process.

The evolution of the phase margin and maximum sensitivity are shown in Figures 3.22 and 3.23 respectively.

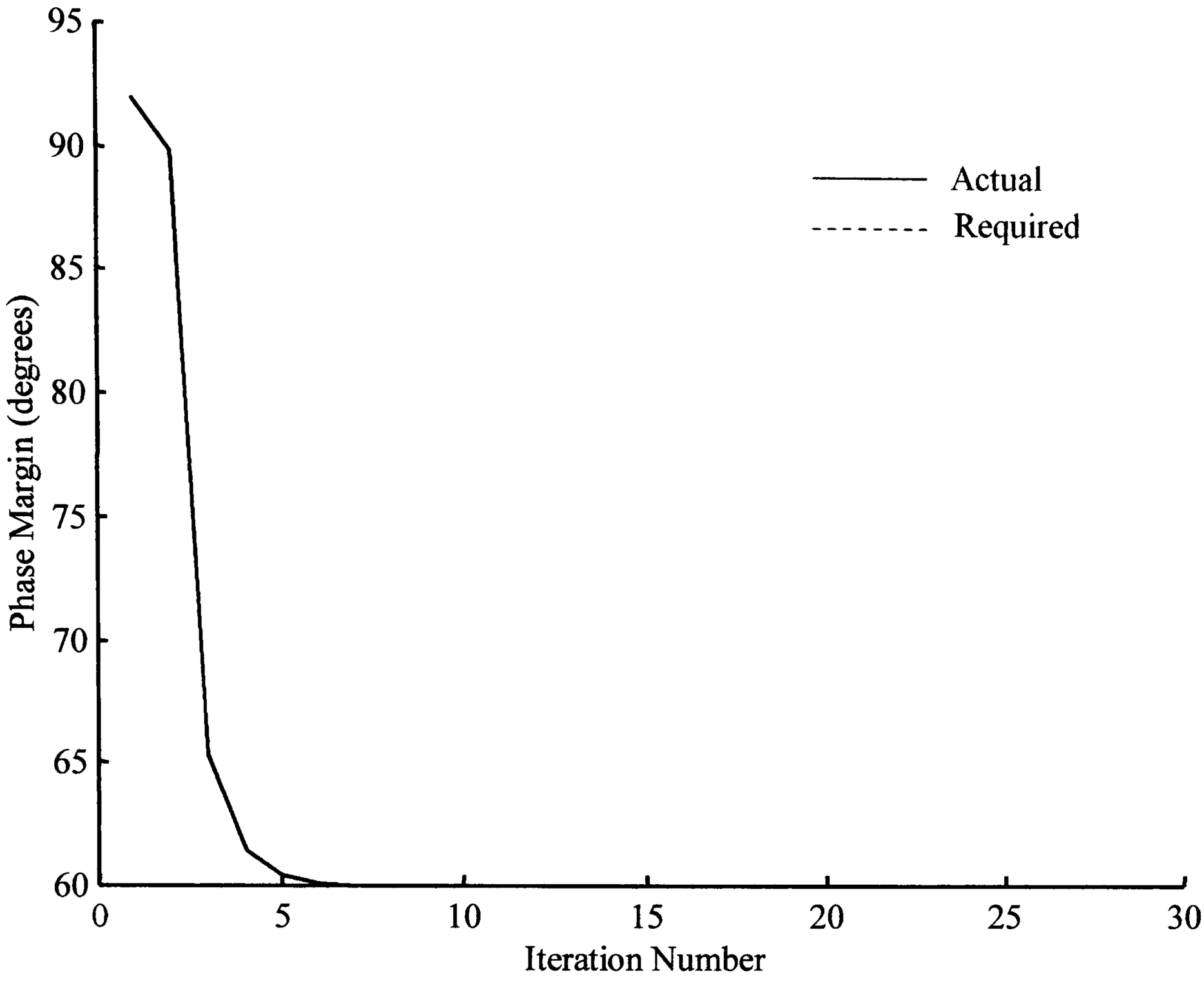


Figure 3.22: Evolution of the phase margin

From Figure 3.22 it can be seen that the phase margin design specification is *exactly* attained after a very few iterations of the algorithm.

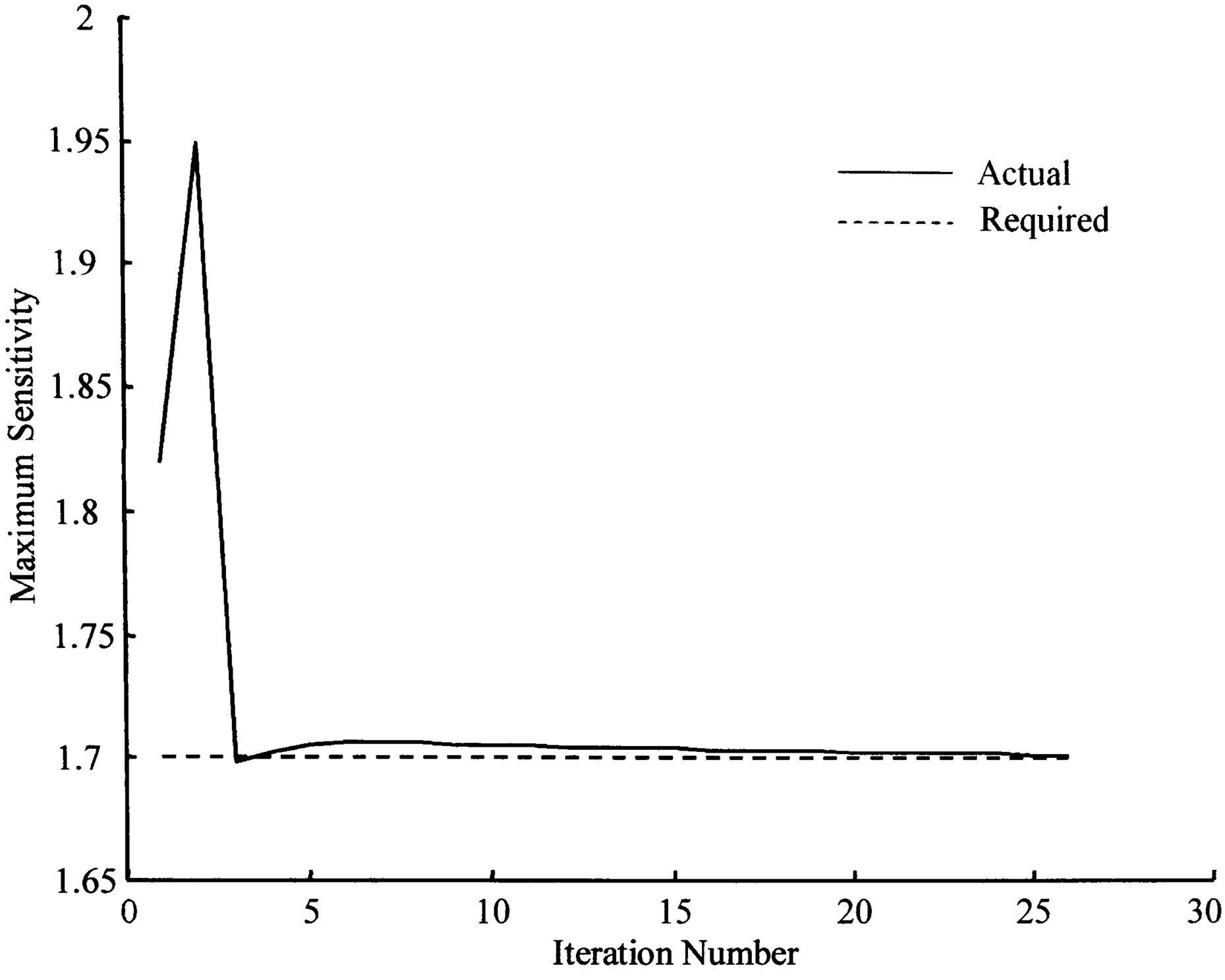


Figure 3.23: Evolution of the maximum sensitivity

The evolution of the maximum sensitivity is shown in Figure 3.23. It can be seen that after an initial large change in the sensitivity that there is a long slow movement towards the *exact* required value.

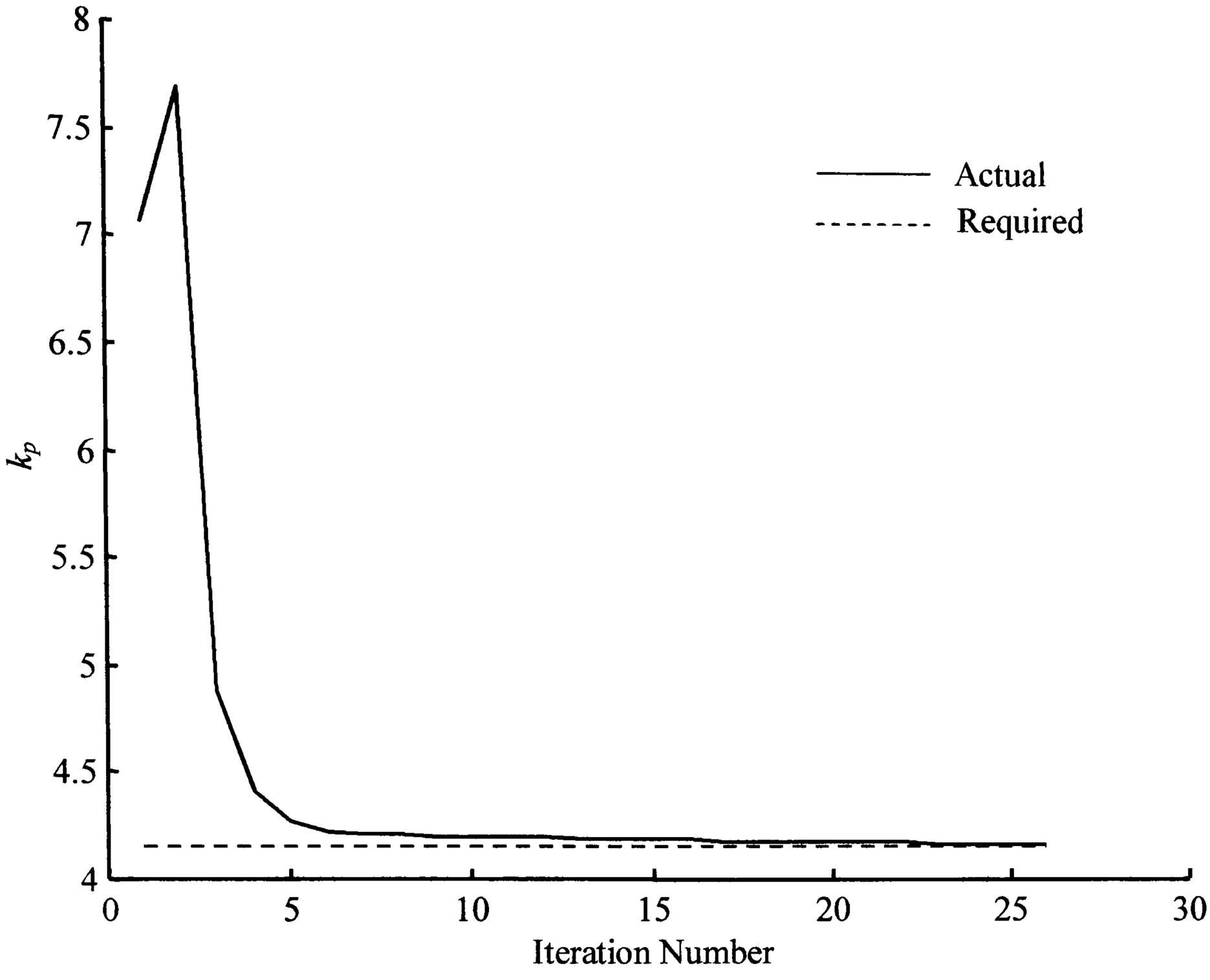


Figure 3.24: Evolution of the controller parameter k_p

From Figure 3.24 it can be seen that the rate of change of the parameter, $k_{p,}$ is practically zero after 10 iterations.

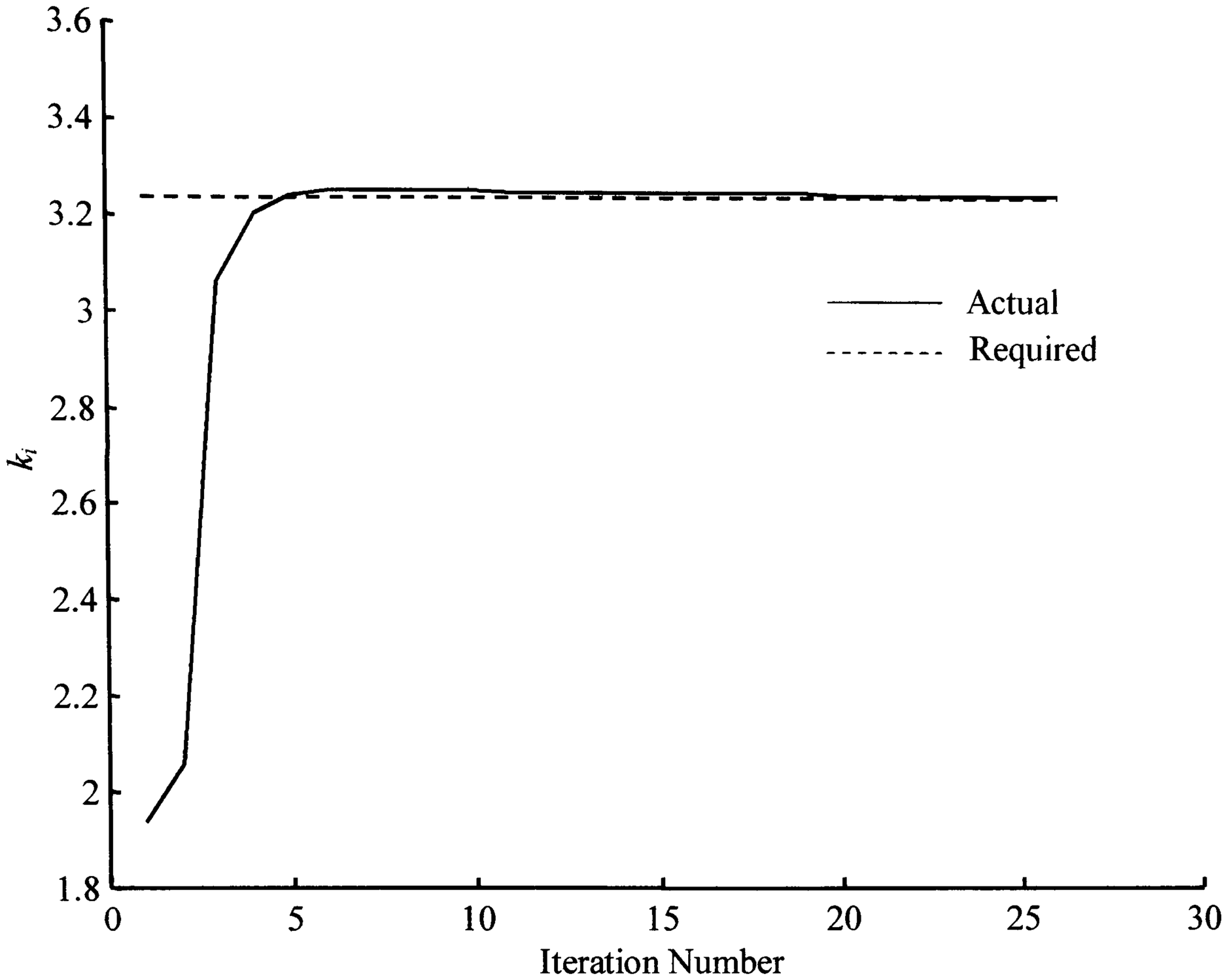


Figure 3.25: Evolution of the controller parameter k_i

From Figure 3.25 it can be seen that there is a similar reduction in the rate of change of the k_i parameter after approximately ten iterations of the algorithm. Hence in this case it would be considered that the algorithm had converged, in a practical sense, after approximately ten iterations.

The step and disturbance rejection properties of the closed loop system are shown in Figure 3.26. A unit step is applied as the set point to the closed loop system at time t = 0(s). A step disturbance in the process output is introduced at time t = 40(s). As can be seen the disturbance rejection properties of the closed loop are satisfactory as is the set point tracking.

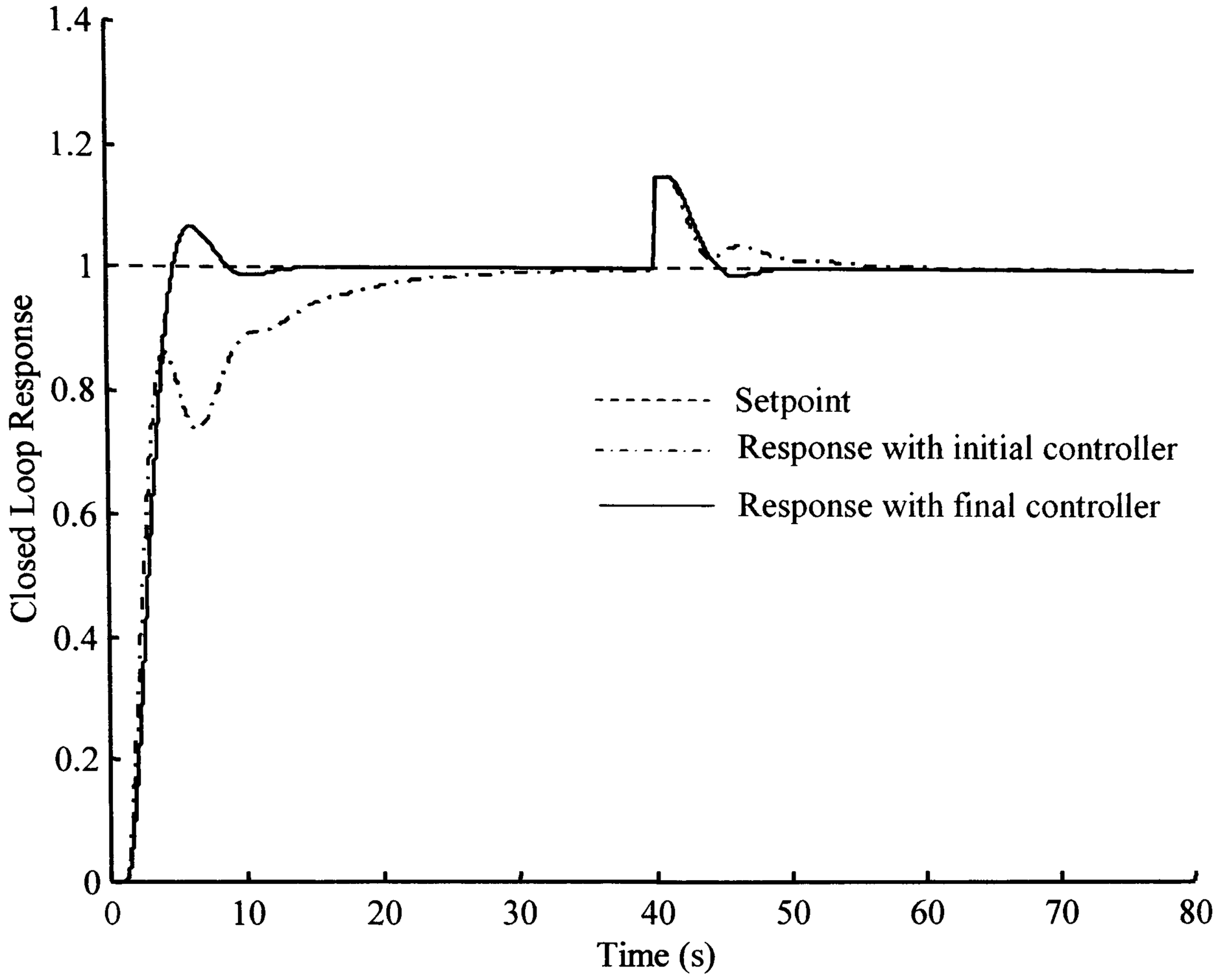


Figure 3.26: Closed Loop Response of the compensated system $G_2(s)$

The final figure of this section, Figure 3.27, shows that the maximum sensitivity and phase margin designs have been achieved. The frequency range used for the frequency response of the forward path of the compensated system shown in Figure 3.26 is 0.1 to 10 (rad.s^{-1}).

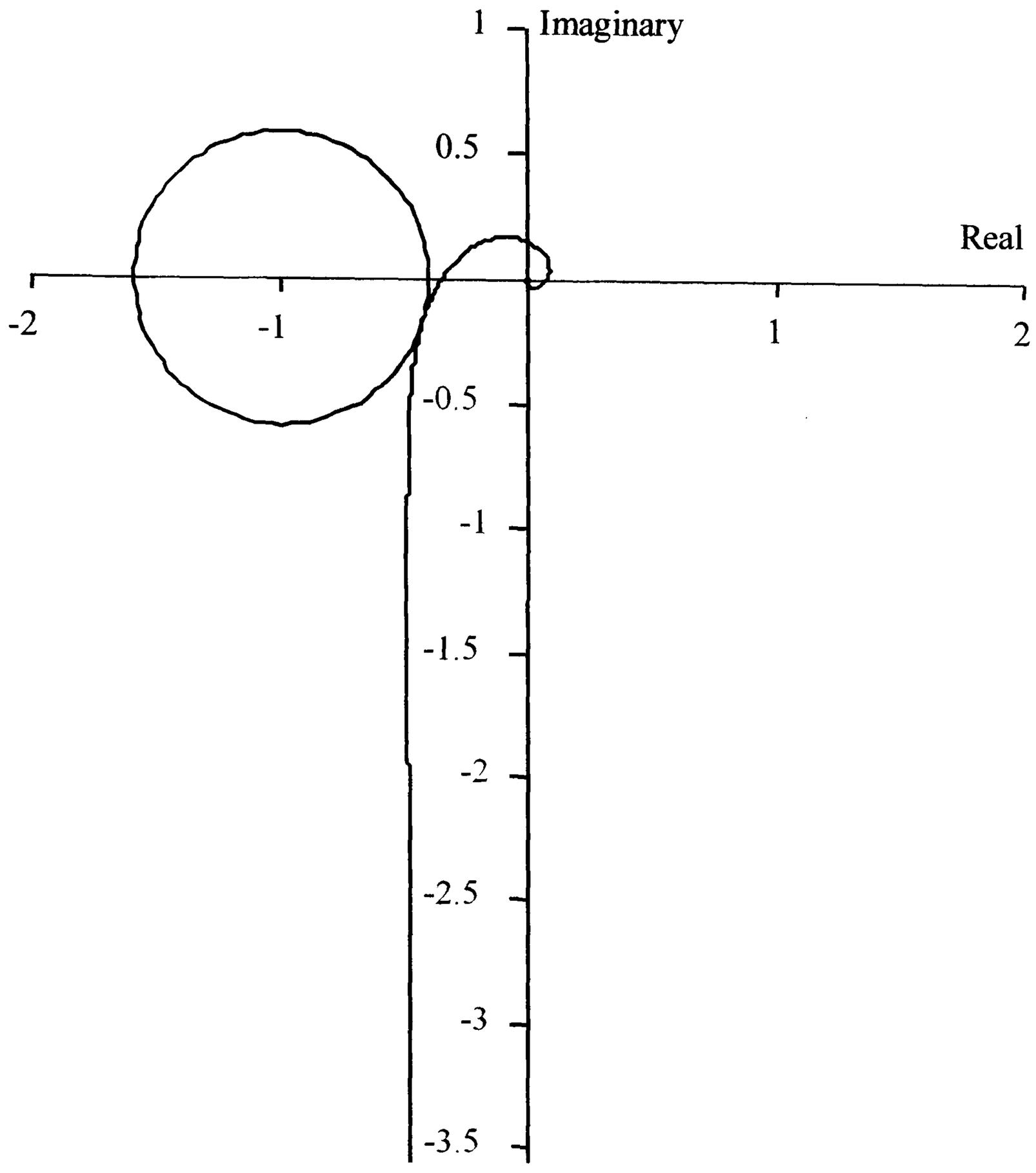


Figure 3.27 Nyquist diagram of the compensated forward path for process $G_2(s)$

3.5 Summary Conclusions.

The Phase-Locked Loop method of system identification in closed loop was extended to show how the module can be used in an automated PI controller design algorithm for identifying the critical features of a new compensated forward path of the process. Two automatic PI controller design algorithms were introduced to achieve:

i) Gain margin and phase margin specifications, and

ii) Maximum sensitivity and phase margin specifications.

In the maximum sensitivity and phase margin PI design algorithm considerable care was required to ensure that the design progressed through a sequence of convergent iterations.

4 Closed Loop Identification and Tuning of Cascade and Multi-Input Multi-Output Control Systems.

4.1 Introduction.

The autotuning of PID controllers using the relay experiment of Astrom and Hagglund (1984) has been extensively reported in the literature (Yu, 1999) and is widely applied in industrial practice as evidenced by the number of PID controller suppliers offering such a capability as *standard* in their PID controllers. The Astrom and Hagglund autotuner was applied to tuning single-input single-output (SISO) PID control systems. The extension of the relay experiment to the autotuning of advanced control strategies such as cascade (Hang *et al*, 1994) and multi-input multi-output systems (MIMO) (Shen and Yu, 1994; Wang *et al*, 1997b) has been widely researched and reported.

In the following section, the application of the Phase-Locked Loop identification method to the closed loop tuning of cascade connected systems shall be proposed. Following from this the closed loop identification and controller design of MIMO systems shall be investigated. Conclusions close the chapter.

4.2 The Cascade Control System Paradigm.

In process industries the manufacture of a product frequently follows a sequence of operations carried out in order. Thus, the next process in the sequence is generally fed directly from the output of the previous stage. Hence over a small section of the production process there will typically be one variable that is able to be manipulated, several process measurements and one output variable that is to be controlled. If the intermediate process variables are subject to disturbances then it is a logical development that inner control loops will be used to attenuate the effects of these disturbances on the outer process loop.

In a cascade connected control strategy a secondary or slave measurement is used such that the slave measurement indicates the presence of the disturbance effect more quickly than the measured output of the process. A slave controller is then used

to attenuate the disturbance effect at that point with the intention of reducing its effect on the primary or controlled variable of the process. If the slave controller design is such that the disturbance effects are sufficiently reduced, the outer or master controller can be tuned to provide the required reference tracking and robust stability properties for the closed loop system. A cascade connected control system is shown in Figure 4.1.

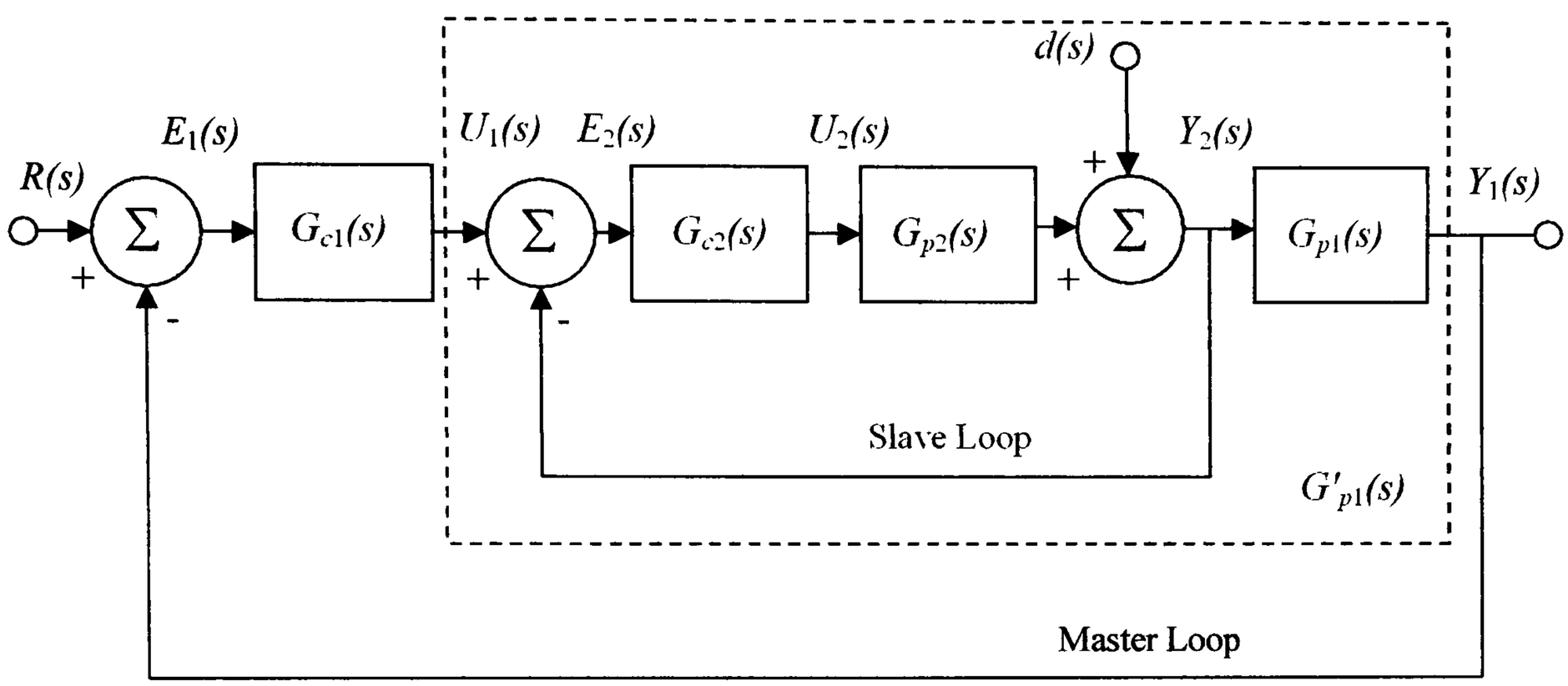


Figure 4.1: Cascade Connected Control System.

The supply disturbance is shown entering the system as *d(s)*. The cascade connected control system, shown in Figure 4.1, is characterised as having an inner or slave control loop and an outer or master control loop. The slave loop is nested within the master loop and the set point for the slave loop is derived from the controller output of the master loop. From Figure 4.1 it can be seen that the master controller is shown as $G_{c1}(s)$ and the slave controller is shown as $G_{c2}(s)$. The process controlled by the slave controller is denoted as $G_{p2}(s)$ and the outer loop process is shown as $G_{p1}(s)$.

From Figure 4.1 it can be seen that

$$Y_2(s) = G_{p2}(s)G_{c2}(s)(U_1(s) - Y_2(s)) + d(s) \tag{4.7}$$

and hence

$$Y_2(s) = T_2(s)U_1(s) + S_2(s)d(s) \tag{4.8}$$

where

$$T_2(s) = \frac{G_{p2}(s)G_{c2}(s)}{1 + G_{p2}(s)G_{c2}(s)} \tag{4.9}$$

and $$S_2(s) = \frac{1}{1 + G_{p2}(s)G_{c2}(s)} \tag{4.10}$$

From equations (4.8) and (4.10) it can be seen that the slave controller $G_{c2}(s)$ is used to attenuate the effects of the disturbance term via the slave loop sensitivity function $S_2(s)$.

An analysis of the outer or master loop shows that

$$Y_1(s) = G_{p1}(s)(T_2(s)G_{c1}(s)(R(s) - Y_1(s)) + S_2(s)d(s)) \tag{4.11}$$

hence $$(1 + T_2(s)G_{p1}(s)G_{c1}(s))Y_1(s) = T_2(s)G_{p1}(s)G_{c1}(s)R(s) + S_2(s)G_{p1}(s)d(s)$$

and
$$\begin{aligned} Y_1(s) = {} & (1 + T_2(s)G_{p1}(s)G_{c1}(s))^{-1} T_2(s)G_{p1}(s)G_{c1}(s)R(s) \\ & + (1 + T_2(s)G_{p1}(s)G_{c1}(s))^{-1} S_2(s)G_{p1}(s)d(s) \end{aligned} \tag{4.12}$$

Assuming that the controller $G_2(s)$ has sufficiently attenuated the slave loop disturbance term, $d(s)$, the master loop controller may be tuned to give an acceptable reference tracking and stability robustness for the closed loop system. The successful design of cascade controllers can thus be seen to require that the disturbance rejection properties, the reference tracking and robust stability requirements are specified. Hence the designs of the master and slave controllers have differing requirements (Eker and Johnson, 1996).

If a model of the disturbance is available and the disturbance term can be measured then a feed forward design can be employed. It can be shown that feed forward, combined with feedback control strategies offer an improvement in disturbance rejection over feedback only control strategies (Seborg *et al*, 1989). Consider Figure 4.2, where a disturbance term, $L(s)$, is seen entering the system. It is assumed that the disturbance term is measurable and that the transfer function, $G_D(s)$, of the disturbance is known. The feed back controller is given by $G_c(s)$ and the feed forward controller is given by $G_{ff}(s)$. The transfer function of the transmitter that measures the disturbance term is given by $G_T(s)$.

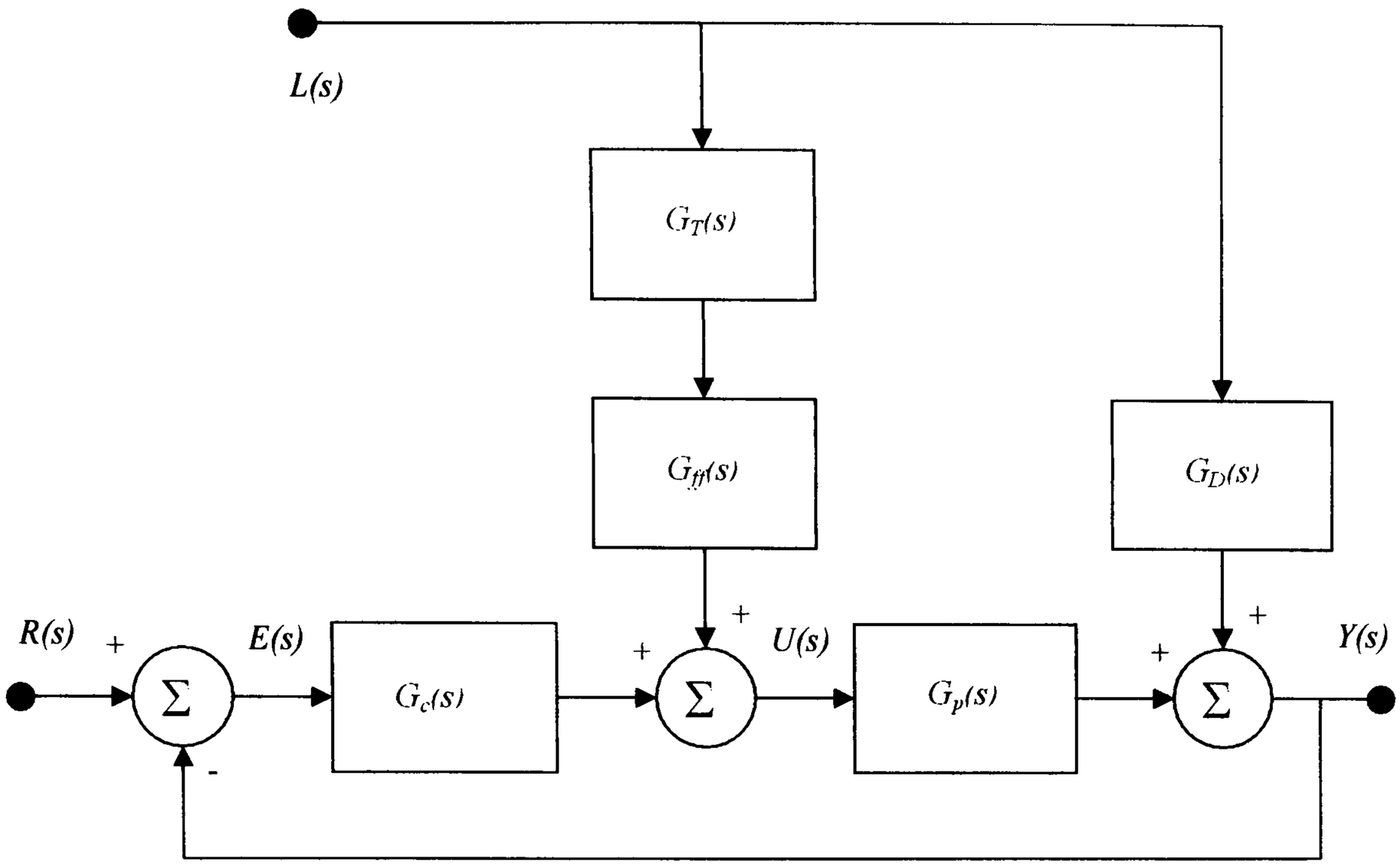


Figure 4.2: Combined Feed Back, Feed forward Control Scheme.

From Figure 4.2 it can be seen that

$$Y(s) = G_p(s)U(s) + G_D(s)L(s) \tag{4.13}$$

$$U(s) = G_c(s)E(s) + G_{ff}(s)G_T(s)L(s) \tag{4.14}$$

$$E(s) = R(s) + Y(s) \tag{4.15}$$

Thus substituting equations (4.15) and (4.14) into equation (4.13), obtain

$$Y(s) = G_p(s)\left(G_c(s)\left(R(s) - Y(s)\right) + G_{ff}(s)G_T(s)L(s)\right) + G_D(s)L(s) \tag{4.16}$$

Hence,

$$\left(1 + G_p(s)G_c(s)\right)Y(s) = G_p(s)G_c(s)R(s) + \left(G_p(s)G_{ff}(s)G_T(s) + G_D(s)\right)L(s) \tag{4.17}$$

It can be seen from equation (4.17) that the effect of the disturbance will be eliminated if the feed forward controller is

$$G_{ff}(s) = -\frac{G_D(s)}{G_p(s)G_T(s)} \tag{4.18}$$

Equation (4.18) may prove to be non-causal depending on the properties of the transfer functions of the disturbance, disturbance transmitter and the process. In such

situations, one solution is to use the static gains of the transfer function elements to design the feed forward controller.

4.3 Auto tuning of Cascade Control Loops.

Cascade control loops are commonly found in the process industries where there is a requirement to reduce the effects of process disturbances. With a large installed base of cascade connected control systems an effective means of tuning these controllers to provide the required degree of control system performance is required. The relay experiment of Astrom and Hagglund provides such a technique for single-input, single-output PID control loops. The extension of the relay experiment to the tuning of cascade control loops has been proposed by Hang *et al* (1994). In Hang *et al* (1994) a sequential tuning method is employed that first of all tunes the slave or inner loop of the cascade system and then the outer or master loop is tuned to complete the process. The method of Hang *et al* (1994) comprises the following steps to complete the cascade control system tuning process.

Algorithm 4.1: Cascade Control Loop Sequential Tuning.

Step 1: Initial Controller Configuration

The master control loop is set to manual mode.

Step 2: Slave Control Loop Tuning.

The slave controller is replaced by a relay and the slave loop is closed around the relay.

The frequency and magnitude of the resultant oscillations in the slave loop are recorded once a stable limit cycle has been established.

Ziegler-Nichols ultimate period method or Refined Ziegler and Nichols tuning rules for PID controller design are used to determine the slave loop controller parameters.

Step 3: Slave Loop Controller Implementation.

The slave controller is re-instated into the slave loop with the parameters set to those calculated in Step 2.

Step 4: Master Control Loop Tuning.

The master loop controller is replaced by a relay and the master loop is closed around the relay. The slave loop being already closed around the slave controller.

The frequency and magnitude of the resulting oscillations of the master loop are recorded once a stable limit cycle has been established.

Ziegler-Nichols ultimate period method or Refined Ziegler and Nichols tuning rules for PID controller design are used to determine the master loop controller parameters.

Step 5: Master Loop Controller Implementation.

The master controller is re-instated into the outer loop with the parameters set to those calculated in Step 4. ●

Remarks

After the above procedure has been carried out both the slave and master controllers are in closed loop.

Further fine tuning of the slave and master loops can be carried out using the relay experiment and Refined Ziegler and Nichols tuning rules due to Hang *et al* (1991).

4.3.1 Closed Loop Identification of Processes in Cascade Connected Control Systems

The method of Hang *et al* (1994) identifies the critical point for the slave loop and the critical point for the master loop, with the slave loop closed, and hence allows the use of single point tuning methods to provide the parameters of the master and slave PID controllers. To improve the performance of the tuning provided by this method either, an improvement in the identification method used can be made or an improved set of tuning rules can be utilised. In the following the Phase-Locked Loop method of system identification shall be used to identify:

i) The inner loop process, shown as $G_{p2}(s)$ in Figure 4.1

ii) The outer loop process, shown as $G_{p1}(s)$ in Figure 4.1

iii) The composite process, shown as $G'_{p1}(s)$ in Figure 4.1, and

By carrying out the above identifications the existence testing methods discussed in Chapter 2 and the gain margin and phase margin method or the maximum sensitivity

and phase margin design method of Chapter 3 can be applied to the design of cascade connected control systems. The application of this design methodology will allow:

i) A greater design freedom as to the method to be used, and

ii) Provide a means that shows that the design requirement is achievable.

From Figure 4.1, and assuming that the disturbance term is zero, it can be seen that (Crowe *et al*, 2003a)

$$U_2(s) = G_{c2}(s)\left(G_{c1}(s)\left(R(s) - Y_1(s)\right) - Y_2(s)\right) \tag{4.19}$$

$$Y_2(s) = G_{p2}(s)U_2(s) \tag{4.20}$$

$$Y_1(s) = G_{p1}(s)G_{p2}(s)U_2(s) \tag{4.21}$$

Substitute equations (4.20) and (4.21) into equation (4.19) to give

$$U_2(s) = G_{c2}(s)\left(G_{c1}(s)\left(R(s) - G_{p1}(s)G_{p2}(s)U_2(s)\right) - G_{p2}(s)U_2(s)\right)$$

hence $$\left(1 + G_{c2}(s)G_{p2}(s)\left(1 + G_{c1}(s)G_{p1}(s)\right)\right)U_2(s) = G_{c1}(s)G_{c2}(s)R(s)$$

and thus $$\frac{U_2}{R}(s) = \frac{G_{c1}(s)G_{c2}(s)}{1 + G_{c2}(s)G_{p2}(s)\left(1 + G_{c1}(s)G_{p1}(s)\right)} \tag{4.22}$$

Substitute for U_2 from equation (4.20) into equation (4.22) to give

$$Y_2(s) = \frac{G_{c1}(s)G_{c2}(s)G_{p2}(s)}{1 + G_{c2}(s)G_{p2}(s)\left(1 + G_{c1}(s)G_{p1}(s)\right)}R(s)$$

and hence $$\frac{Y_2}{R}(s) = \frac{G_{c1}(s)G_{c2}(s)G_{p2}(s)}{1 + G_{c2}(s)G_{p2}(s)\left(1 + G_{c1}(s)G_{p1}(s)\right)} \tag{4.23}$$

From equations (4.22) and (4.23) it can be seen that

$$\arg\left(G_{p2}(j\omega)\right) = \arg\left(\frac{Y_2}{R}(j\omega)\right) - \arg\left(\frac{U_2}{R}(j\omega)\right) \tag{4.24}$$

$$\left|G_{p2}(j\omega)\right| = \frac{\left|\frac{Y_2}{R}(j\omega)\right|}{\left|\frac{U_2}{R}(j\omega)\right|} \tag{4.25}$$

Equations (4.24) and (4.25) show that if two simultaneous identifications are carried out between the reference input and the output of the slave controller, $G_{c2}(s)$, and between the reference input and the output of the inner loop or slave process, $G_{p2}(s)$, then it is possible to identify the inner loop process represented by $G_{p2}(s)$.

Referring to Figure 4.1 it can be seen that

$$U_1(s) = G_{c1}(s)(R(s) - Y_1(s)) \tag{4.26}$$

$$Y_1(s) = G_{p1}(s)\frac{G_{c2}(s)G_{p2}(s)}{1 + G_{c2}(s)G_{p2}(s)}U_1(s) \tag{4.27}$$

Substitute (4.27) into (4.26) to give

$$U_1(s) = G_{c1}(s)R(s) - \frac{G_{c1}(s)G_{p1}(s)G_{c2}(s)G_{p2}(s)}{1 + G_{c2}(s)G_{p2}(s)}U_1(s) \tag{4.28}$$

hence $$(1 + G_{c2}(s)G_{p2}(s)(1 + G_{c1}(s)G_{p1}(s)))U_1(s) = (1 + G_{c2}(s)G_{p2}(s))G_{c1}(s)R(s) \tag{4.29}$$

and thus $$\frac{U_1(s)}{R(s)} = \frac{(1 + G_{c2}(s)G_{p2}(s))G_{c1}(s)}{1 + G_{c2}(s)G_{p2}(s)(1 + G_{c1}(s)G_{p1}(s))} \tag{4.30}$$

Substitute for U_1 from (4.30) into (4.27) to give

$$Y_1(s) = \frac{G_{p1}(s)G_{p2}(s)G_{c1}(s)G_{c2}(s)}{1 + G_{c2}(s)G_{p2}(s)(1 + G_{c1}(s)G_{p1}(s))}R(s) \tag{4.31}$$

hence $$\frac{Y_1(s)}{R(s)} = \frac{G_{p1}(s)G_{p2}(s)G_{c1}(s)G_{c2}(s)}{1 + G_{c2}(s)G_{p2}(s)(1 + G_{c1}(s)G_{p1}(s))} \tag{4.32}$$

From (4.32) and (4.30) it can be seen that

$$\arg(G'_{p1}(j\omega)) = \arg\left(\frac{Y_1}{R}(j\omega)\right) - \arg\left(\frac{U_1}{R}(j\omega)\right) \tag{4.33}$$

$$\left|G'_{p1}(j\omega)\right| = \frac{\left|\frac{Y_1}{R}(j\omega)\right|}{\left|\frac{U_1}{R}(j\omega)\right|} \tag{4.34}$$

From equations (4.33) and (4.34) it can be seen that the simultaneous identifications required to be carried out such that the composite process $G'_{p1}(s)$ can be identified are from the reference input to the output of the master controller, $G_{c1}(s)$, and from the reference input to the output of the outer loop process, $G_{p1}(s)$. It should be noted that the proposed identifications are carried out with no prior knowledge of the master or slave controllers being required.

In Crowe (1998) it is shown that the Phase-Locked Loop method of system identification can be used, albeit with an extended identification time, in the presence of step like load disturbances. Thus, there is no loss of generality in assuming that the disturbance term is zero during the course of the identification procedure. The necessary steps in carrying out an identification using the Phase-Locked Loop method are given in Algorithm 4.2.

Algorithm 4.2: Closed Loop Identification of Cascade Connected Control Systems Using the Phase-Locked Loop Method.

Step 1: Initialisation.

Choose the reference value, phase or gain, that locates the required point on the frequency response of the inner process.

Set the initial value of the integrator gain and choose the update and stopping tolerance values.

Step 2: Inner Process Identification.

Carry out simultaneous identifications between the master loop reference input and the slave loop controller output and between the master loop reference input and the slave loop output.

Record the results of the identification.

Step 3: Composite Process Identification.

Carry out simultaneous identifications between the master loop reference input and the master loop controller output and between the master loop reference input and the master loop output.

Record the results of the identification. ●

Remarks

The composite process refers to the series connection of the inner closed loop transfer function and the transfer function of the outer process.

In Chapter 1, Figure 1.18 the configuration of the Phase-Locked Loop Identifiers required to implement the identification is shown.

Case Study 4.1

The inner and outer processes of a cascade connected control system will be identified using the Phase-Locked Loop method. The structure of the system is that shown in Figure 4.1 where the transfer functions of the processes are given by

$$G_{p2}(s) = \frac{2e^{-0.7s}}{(s+2)(s+4)}$$

$$G_{p1}(s) = \frac{0.1e^{-0.1s}}{(s+0.1)^2(s+0.5)}$$

and the inner and outer controllers are of PI type and are given, initially, by

$$G_{c2} = 2.818 + \frac{1.401}{s}$$

$$G_{c1} = 0.1323 + \frac{0.0054}{s}$$

The parameters of the PI controllers were derived from the results of a relay experiment following the method of Hang *et al* (1994) and the application of the Ziegler and Nichols ultimate period tuning rules. The identification of the inner and outer processes shall be carried out with both the inner and outer loops closed and shall employ the Phase-Locked Loop identifier configuration shown in Figure 1.18 of Chapter 1.

Inner Process Identification

The first identification that shall be carried out is that of the inner process $G_{p2}(s)$ at the phase crossover point, viz. the point at which the process phase shift is $-\pi$ (rad). The simultaneous identifications that were carried out were:

i) From the reference input to the inner controller output, and

ii) From the reference input to the inner process output.

Table 4.1 details the results for the identification.

Table 4.1: Inner Process Identification Results.

$$G_{p2}(s)=\frac{2e^{-0.7s}}{(s+2)(s+4)}$$

Theoretical		
$\omega_{-\pi}$ (rad.s^{-1})	$\lvert G_{p2}(j\omega_{-\pi})\rvert$	Identification Time (s)
2.441	0.1352	-
Identified		
$\omega_{-\pi}$ (rad.s^{-1})	$\lvert G_{p2}(j\omega_{-\pi})\rvert$	Identification Time (s)
2.4425	0.1352	979

The initial identifier frequency was chosen as 0.1(rad.s^{-1}) and 0.2 was the peak magnitude of the excitation, the peak magnitude remained unaltered during the identification process. As can be seen from Table 4.1 the identified data is highly accurate, the phase crossover frequency is within 0.06% of the theoretical value and the identified magnitude is for all practical purposes exact. The time taken to carry out the identification was 979 (s), this may at first seem to be rather excessive however it should be noted that the Phase-Locked Loop method supplies the identification data of points on the frequency response curve of the process being identified as it approaches the desired or reference process identification point. Additionally, since the process was considered to be unknown the identification was initialised at a relatively low frequency of 0.1(rad.s^{-1}), compared with the phase crossover frequency of the process. The identification was carried out in closed loop with the transient response of the closed loop system having a marked effect on the identification time. This was due to the long settling time required at each change of identification frequency. The accuracy of the points identified by the Phase-Locked Loop identifier can be seen from Figure 4.3 where the Nyquist curve of the actual process is compared with the identified points of the inner process as the identified data approaches the reference phase shift of -π (rad). The frequency range over which the Nyquist diagram was plotted was from 0.1(rad.s^{-1}) to 2.4425(rad.s^{-1}). From Figure 4.3 it can be seen that there is an extremely good match between the identified points and the Nyquist curve of the actual process.

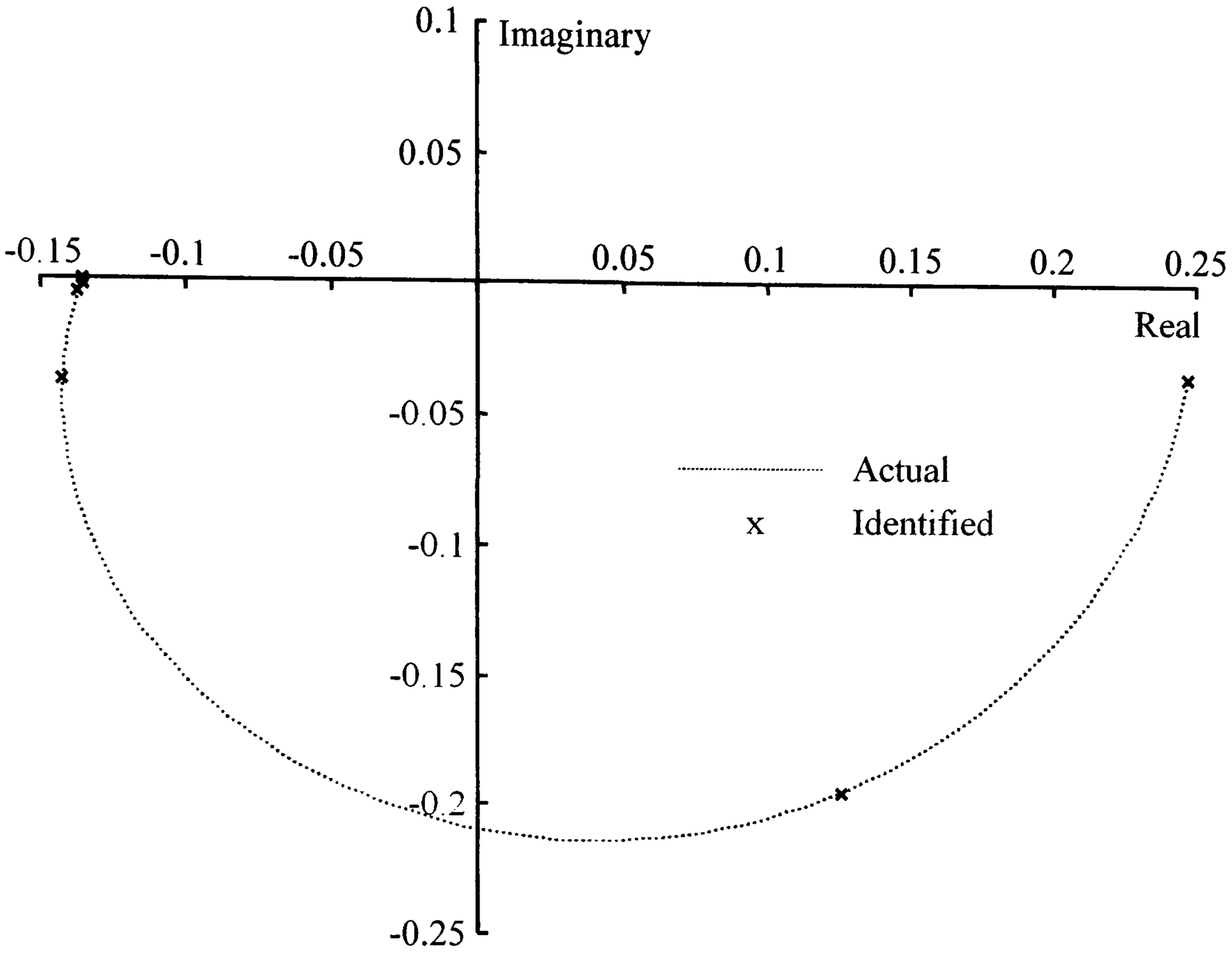


Figure 4.3: Inner Process Nyquist Diagram and Identified Data Points.

Composite Process Identification.

The second Identification that was carried out was the identification of the composite process, shown as $G'_{pl}(s)$ in Figure 4.1, at the phase crossover point. The Phase-Locked Loop identifier was initialised such that the excitation frequency was 0.1 (rad.s^{-1}) with a peak magnitude of 0.2. The excitation peak magnitude was not altered during the course of the identification. Table 4.2 details the results of the identification. As can be seen from Table 4.2 the identified phase crossover point of the composite process was identified to a high degree of accuracy, better than 0.05%. The time to achieve the identification was relatively long. However, as can be seen from Figure 4.4 there are other points accurately identified as the Phase-Locked Loop identifier converges to the desired reference point.

Table 4.2: Outer Process Identification Results.

$$G_{p1}^{'}(s)=\frac{G_{p2}(s)G_{c2}(s)}{1+G_{p2}(s)G_{c2}(s)}G_{p1}(s)$$

Theoretical		
$\omega_{-\pi}$ (rad.s^{-1})	$\left\|G_{p1}^{'}(j\omega_{-\pi})\right\|$	Identification Time (s)
0.2081	2.9193	-
Identified		
$\omega_{-\pi}$ (rad.s^{-1})	$\left\|G_{p1}^{'}(j\omega_{-\pi})\right\|$	Identification Time (s)
0.2080	2.9194	2206

The time to perform an identification using the Phase-Locked Loop identifier is dependent on the characteristics of the process being identified. In this case study the processes involved have relatively long time constants and the tuning of the controllers produce an oscillatory response to set point changes with an extended settling time. All of the above add to the time required to carry out the identification. However it should be noted that the data from the identification method is very accurate and could be used in a multi-point PID controller design method.

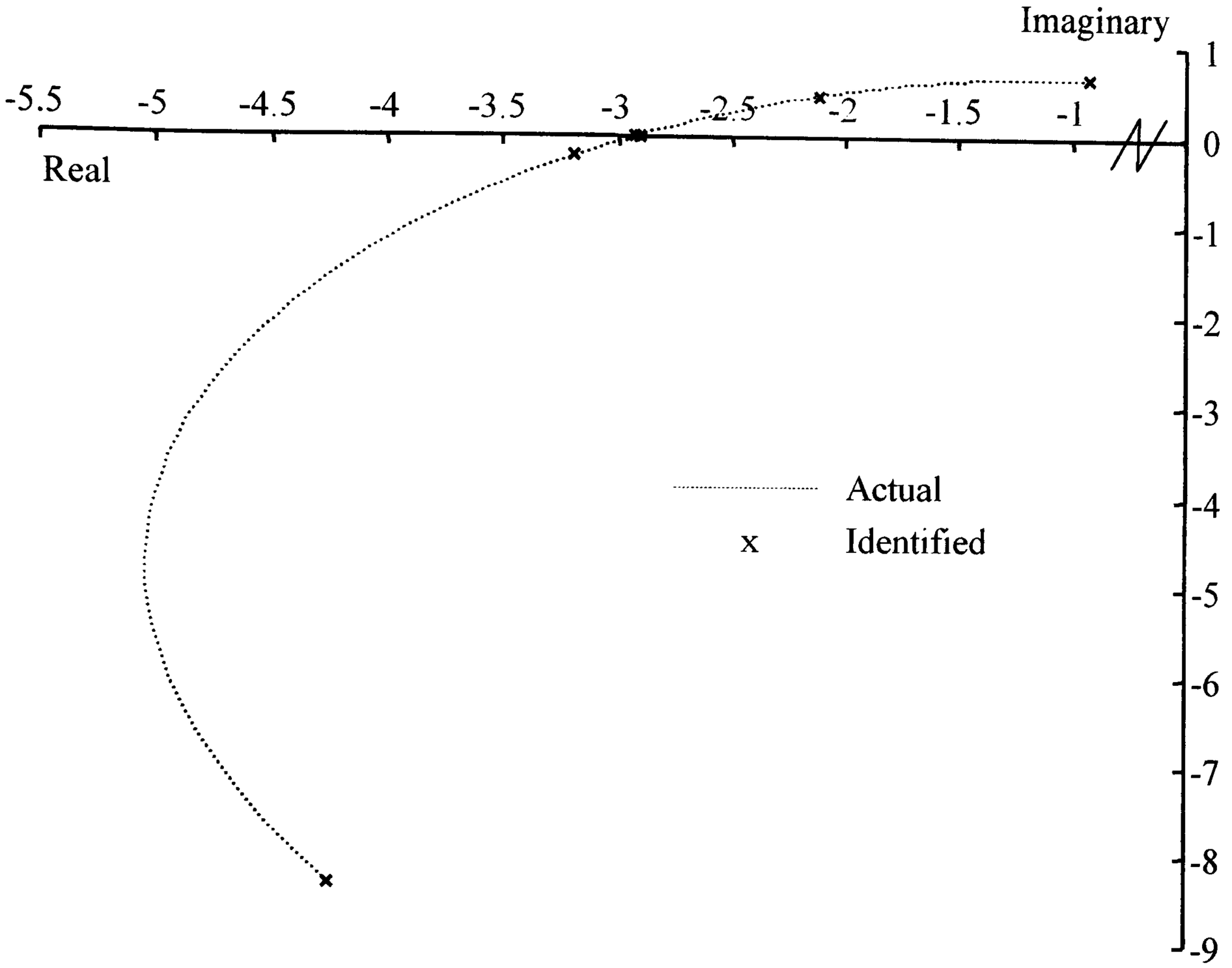


Figure 4.4: $G_{p1}^{'}(s)$ Nyquist Diagram and Identified Data Points.

4.3.2 Cascade Controller Gain and Phase Margin Design.

The initial design of the PI controllers used in Case Study 4.1, $G_{c1}(s)$ and $G_{c2}(s)$, was carried out using a relay experiment and the application of the Ziegler and Nichols ultimate period method for PI controllers. Further the initial tuning followed the sequential tuning method of Hang *et al* (1994). In the following the design of PI controllers shall be carried out such that the disturbance rejection property of the initial Ziegler and Nichols PI controller design of the cascade system of Case Study 4.1 shall be improved upon. In Case Study 4.1 no prior knowledge of the controllers was required, however in the following algorithm it is assumed that the master controller is known. If it is not then it can be identified using the Phase-Locked Loop technique. All of the identification steps required to complete the design shall be carried out with the cascade system in closed loop. Additionally, a test shall be

employed to ensure that the controller tuning parameters can be implemented on line without the closed loop system becoming unstable. Motivated by the Hang *et al* (1994) sequential tuning method and the incorporation of a new identification tool, algorithm 4.3 shall be used to implement the identification and tuning method.

Algorithm 4.3: Closed Loop Identification and Tuning of a Cascade Connected System.

Step 1: Initialisation.

Choose the phase margin, ϕ_{PM} , and gain margin, *GM*, ranges that are required for the inner and outer loops.

Choose the number of points to be identified on the frequency response curves of the inner and composite processes.

Step 2: Inner Process Identification.

The inner process frequency response is identified at the required number of points in the phase reference range -π(rad) to $-\pi/2+\hat{\phi}_{PM}$ (rad), where $\hat{\phi}_{PM}$ is the largest value of the required phase margin range.

Step 3: Inner Loop Viable Gain Margin and Phase Margin Pairing.

The data supplied by Step 2 is used by the Viable Gain Margin and Phase Margin Pairing Algorithm, Algorithm 2.1 of chapter 2.

The required gain and phase margin are chosen on the basis of obtaining the largest value of integral gain available from the PI controller parameter set found (Astrom and Hagglund, 1995).

Step 4: Slave Loop Stability Test.

Use the Phase-Locked Loop method to identify the outer process $G_{p1}(s)$, such that the phase error is given by

$$e_k = -\pi - \arg\left(G_{c1}(j\omega)\right) - \arg\left(G'_{c2}(j\omega)\right) - \arg\left(G_{p2}(j\omega)\right)$$
$$- \arg\left(G_{p1}(j\omega)\right) + \arg\left(1 + G'_{c2}(j\omega)G_{p2}(j\omega)\right)$$

Using the data supplied from the above identification of $G_{p1}(s)$, the known controller parameters and the identification data for $G_{p2}(s)$ determine if

$$|G_{c1}(j\omega_{-\pi})|\frac{|G_{c2}^{'}(j\omega_{-\pi})G_{p2}(j\omega_{-\pi})|}{|1+G_{c2}^{'}(j\omega_{-\pi})G_{p2}(j\omega_{-\pi})|}|G_{p1}(j\omega_{-\pi})|<1$$

where $\omega_{-\pi}$ is the frequency at which

$$\arg(G_{p1}(j\omega_{-\pi}))=-\pi-\arg(G_{c1}(j\omega_{-\pi}))-\arg(G_{c2}^{'}(j\omega_{-\pi}))$$
$$-\arg(G_{p2}(j\omega_{-\pi}))+\arg(1+G_{c2}^{'}(j\omega_{-\pi})G_{p2}(j\omega_{-\pi}))$$

If $$|G_{c1}(j\omega_{-\pi})|\frac{|G_{c2}^{'}(j\omega_{-\pi})G_{p2}(j\omega_{-\pi})|}{|1+G_{c2}^{'}(j\omega_{-\pi})G_{p2}(j\omega_{-\pi})|}|G_{p1}(j\omega_{-\pi})|<1$$

Then go to Step 5

Else the design is continued with the outer loop on manual at Step 5 or go to step 3.

Step 5: Composite Process Identification.

The composite process frequency response is identified at the required number of points in the phase reference range -π(rad) to $-\pi/2+\hat{\phi}_{PM}$ (rad), where $\hat{\phi}_{PM}$ is the largest value of the required phase margin range.

Step 6: Outer Loop Viable Gain Margin and Phase Margin Pairing.

The data supplied by Step 5 is used by the Viable Gain Margin and Phase Margin Pairing Algorithm, Algorithm 2.1 of chapter 2.

From the data supplied by the gain and phase margin pairing algorithm a choice of gain and phase margin is made that gives an acceptable time domain response for the cascade connected loop. ●

Case Study 4.2

For this design exercise a gain margin and phase margin specification shall be used to determine the parameters of the inner and outer PI controllers. The first step in the design is to determine the gain margin and phase margin pairings that can be achieved using a PI controller in the inner control loop. The automated existence testing of gain margin and phase margin pairings achievable by PI controllers has been described in Chapter 2. For this case study the range of the gain margin was chosen to be from 2 to 5 with the range for the phase margin chosen as 30° to 60°. In order to utilise the automated testing method the inner process, $G_{p2}(s)$, must be

identified at a number of points on its frequency response. The inner process was identified at sixteen equally spaced points over a phase angle range of $-\pi$ (rad) to $-\pi/6$ (rad), using the Phase-Locked Loop identification technique. The results of the identification are shown in Figure 4.5

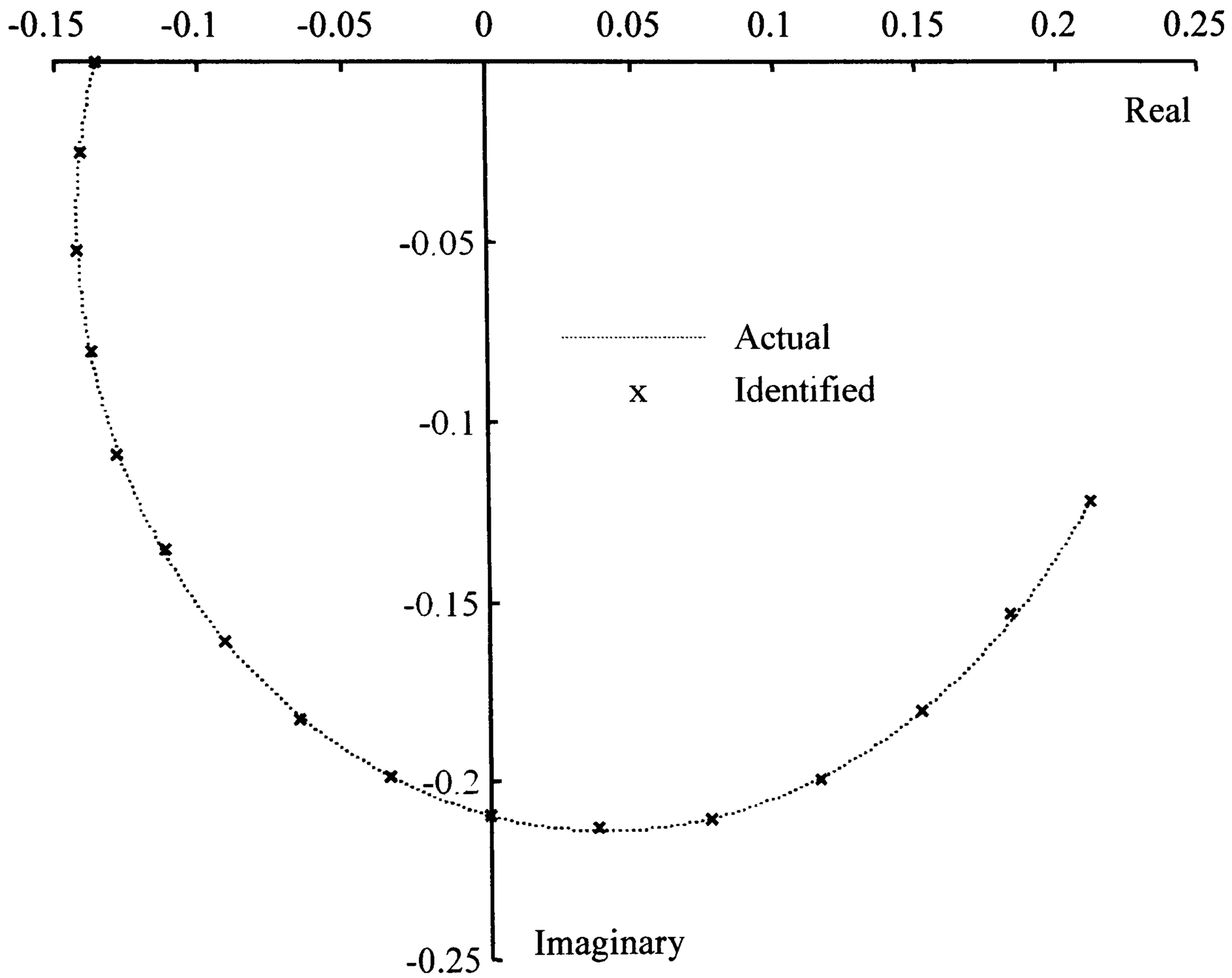


Figure 4.5: Nyquist Diagram for $G_{p2}(s)$ Showing Identified Data Points.

From Figure 4.5 it can be seen that the identified data is in very good agreement with the theoretical values. However the time taken to carry out the identification can be considered to be excessive. The average identification time per point was 196(s) with the identification of all sixteen points taking approximately 53 minutes. Although this may seem to be a long length of time to be carrying out tests on the system it should be remembered that the cascade control system is at all times during the identification operating with both the inner and outer loops closed. In addition to this the magnitude of the excitation signal can be kept to a relatively low level, thus reducing the upset to the process being controlled during the identification phase.

Also note that the identification method and the gain and phase margin pairing algorithm require very little input by an operator for the initialisation, thus expensive manpower is not needed during this phase of the design process. Table 4.3 shows the results from the identification, it should be noted that only the data at the identified points is presented since this is the only identification data supplied to the gain margin and phase margin pairing algorithm.

Table 4.3: Identification Results for Inner Process, $G_{p2}(s)$

$$G_{p2}(s) = \frac{2e^{-0.7s}}{(s+2)(s+4)}$$

Magnitude	Phase Angle (rad)	Excitation Frequency (rad.s^{-1})	Time (s)
0.2447	-0.5233	0.3628	607
0.2412	-0.6978	0.4853	1011
0.2366	-0.8722	0.6086	1152
0.2308	-1.0467	0.7340	1499
0.2244	-1.2220	0.8612	1681
0.2171	-1.3964	0.9881	1749
0.2092	-1.5709	1.1199	1994
0.2022	-1.7456	1.2535	2101
0.1937	-1.9196	1.3947	2257
0.1846	-2.0942	1.5288	2426
0.1762	-2.2689	1.6679	2569
0.1682	-2.4434	1.8242	2637
0.1586	-2.6184	1.9698	2801
0.1561	-2.7926	2.1223	2927
0.1436	-2.9673	2.2764	3060
0.1363	-3.1416	2.4392	3139

Figure 4.6 shows the results from the gain margin and phase margin pairing algorithm based on the data supplied from the identification of the inner process, $G_{p2}(s)$.

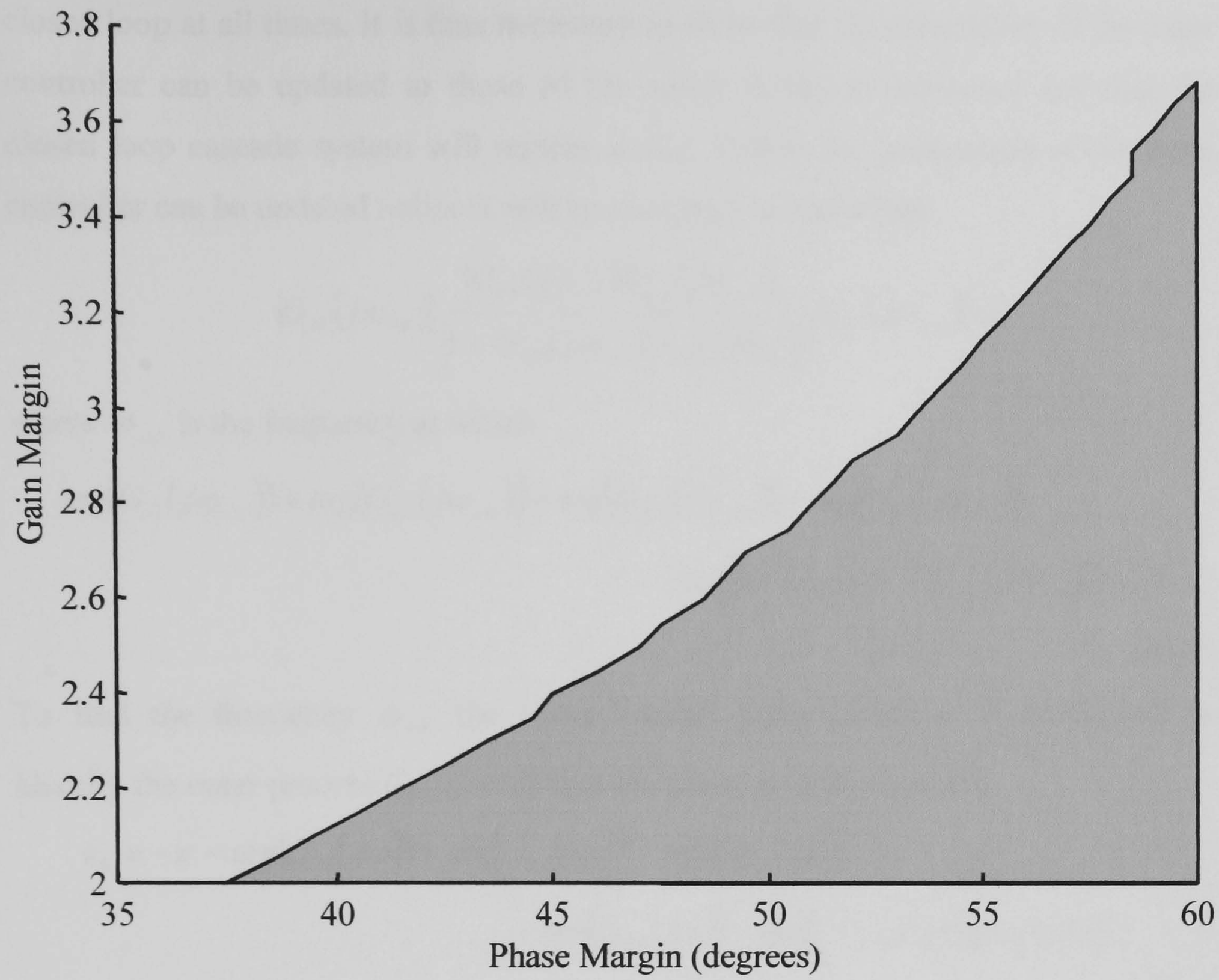


Figure 4.6: Viable Gain Margin and Phase Margin Pairings for the Process $G_{p2}(s)$.

The shaded region in Figure 4.6 represents the viable gain margin and phase margin pairings that can be achieved by a PI controller and the inner process, $G_{p2}(s)$. The next stage of the design is to choose the gain margin and phase margin pairing that will give the largest value of integral gain for the PI controller. This choice is made on the basis that the integral of the error is inversely proportional to the integral gain and thus will improve the disturbance rejection properties of the closed loop system (Astrom and Hagglund, 1995). From the data generated from the gain margin and phase margin viable pairing method, the gain margin and phase margin pairing that resulted in the largest value of integral gain was a gain margin of 2 and a phase margin of 60°. The resulting PI controller is given by

$$G_{c2}^{'}(s) = 2.8248 + \frac{2.9339}{s}$$

One of the objectives of this control system design is to be able to keep the system in closed loop at all times. It is thus necessary to show that the parameters of the inner controller can be updated to those of the newly designed controller and that the closed loop cascade system will remain stable. Before the parameters of the inner controller can be updated online it will be necessary to show that

$$\left|G_{c1}(j\omega_{-\pi})\right|\frac{\left|G_{c2}^{'}(j\omega_{-\pi})G_{p2}(j\omega_{-\pi})\right|}{\left|1+G_{c2}^{'}(j\omega_{-\pi})G_{p2}(j\omega_{-\pi})\right|}\left|G_{p1}(j\omega_{-\pi})\right|<1$$

where $\omega_{-\pi}$ is the frequency at which

$$\arg\left(G_{c1}(j\omega_{-\pi})\right)+\arg\left(G_{c2}^{'}(j\omega_{-\pi})\right)+\arg\left(G_{p2}(j\omega_{-\pi})\right)+\arg\left(G_{p1}(j\omega_{-\pi})\right)$$
$$-\arg\left(1+G_{c2}^{'}(j\omega_{-\pi})G_{p2}(j\omega_{-\pi})\right)=-\pi \tag{4.35}$$

To find the frequency $\omega_{-\pi}$ the Phase-Locked Loop Identifier is configured to identify the outer process $G_{p1}(s)$ such that the phase error is given by

$$e_k=-\pi-\arg\left(G_{c1}(j\omega)\right)-\arg\left(G_{c2}^{'}(j\omega)\right)-\arg\left(G_{p2}(j\omega)\right)$$
$$-\arg\left(G_{p1}(j\omega)\right)+\arg\left(1+G_{c2}^{'}(j\omega)G_{p2}(j\omega)\right) \tag{4.36}$$

Using the data from Table 4.3, the known controller parameters and an initial identification of $G_{p1}(s)$ at a frequency of 0.3628 (rad.s^{-1}) it can be seen that the initial frequency is too high since the calculated phase angle of the cascade connected system forward path is -3.8544 (rad). Thus an identification of the inner process at a range of lower frequencies is required to be carried out. Five points were chosen to be identified in the range 10° to 30°; Table 4.4 shows the results of the identification.

Table 4.4: Identification Results for Inner Process, $G_{p2}(s)$

$$G_{p2}(s) = \frac{2e^{-0.7s}}{(s+2)(s+4)}$$

Magnitude	Phase Angle (rad)	Excitation Frequency (rad.s^{-1})	Time (s)
0.2497	-0.1741	0.1203	538
0.2491	-0.2622	0.1810	708
0.2477	-0.3486	0.2404	844
0.2469	-0.4361	0.3016	974
0.2449	-0.5240	0.3622	1150

Cubic splines are used to interpolate between the data points shown in Table 4.4, thus two look-up tables are formed:

i) Phase angle vs. frequency, and

ii) Magnitude vs. frequency

Since during the identification process the frequency of excitation is always known this can be used to drive the look-up table to provide the frequency response data required for the process $G_{p2}(s)$. Since the controller parameters are known it is possible to calculate their frequency response for a given frequency. The Phase-Locked Loop identifier is now configured to identify the outer process $G_{p1}(s)$ and is given the phase reference, ϕ_{ref}, as shown in equation (4.36). The identification gave the following results

$$\left|G_{p1}(j\omega_{-\pi})\right| = 3.4285$$

$$\arg\left(G_{p1}(j\omega_{-\pi})\right) = -2.6672 \text{ (rad)}$$

at a frequency of

$$\omega_{-\pi} = 0.2093 \text{ (rad.s}^{-1}\text{)}$$

Using the above data and the known controller parameters and the identification data for the inner process, the magnitude of the forward path of the cascade system is now found as

$$\left|G_{c1}(j\omega_{-\pi})\right|\frac{\left|G_{c2}^{'}(j\omega_{-\pi})G_{p2}(j\omega_{-\pi})\right|}{\left|1+G_{c2}^{'}(j\omega_{-\pi})G_{p2}(j\omega_{-\pi})\right|}\left|G_{p1}(j\omega_{-\pi})\right|=0.3977$$

Since this value is less than unity it will be possible to update the controller online and the closed loop cascade system shall remain stable. Should it not be possible to update the controller online the outer control loop could be placed on manual and the design continued or a possible redesign of the inner controller would be required.

To compare the performance of the cascade control system operating with the new inner controller, $G_{c2}^{'}(s)$, against the performance with the original inner controller, $G_{c2}(s)$, the following test was carried out. A step input was applied to the closed loop cascade system with the original controller settings at time t=0(s), followed by a step load disturbance of magnitude 0.15 applied to the inner loop at time t=300(s). The integral of the square of the reference error of the master controller was then calculated, using

$$I_1=\int_{\tau=0}^{\infty}e_1^2(\tau)d\tau$$

where $e_1(t)$ is shown in Figure 4.2.

A similar test was carried out on the cascade system this time operating in closed loop with the inner controller using the updated PI controller, $G_{c2}^{'}(s)$. Table 4.5 shows the results of the comparison for the two tests.

Table 4.5: Integral Square Error (ISE) Comparison

$G_{c2}(s)=2.818+\frac{1.401}{s}$		$G_{c2}^{'}(s)=2.8248+\frac{2.9339}{s}$	
ISE		ISE	
Step Only	Step and Load Disturbance	Step Only	Step and Load Disturbance
12.64	13.84	10.46	10.72

From Table 4.5 it can be seen that the results using the new design for the inner controller results in an improvement in the overall performance of the cascade system as far as the load disturbance rejection property is concerned. Figure 4.7

allows a comparison to be carried out between the closed loop response of the cascade system using the original inner and outer controllers and that obtained when the inner controller is replaced by the updated controller, $G_{c2}^{'}(s)$.

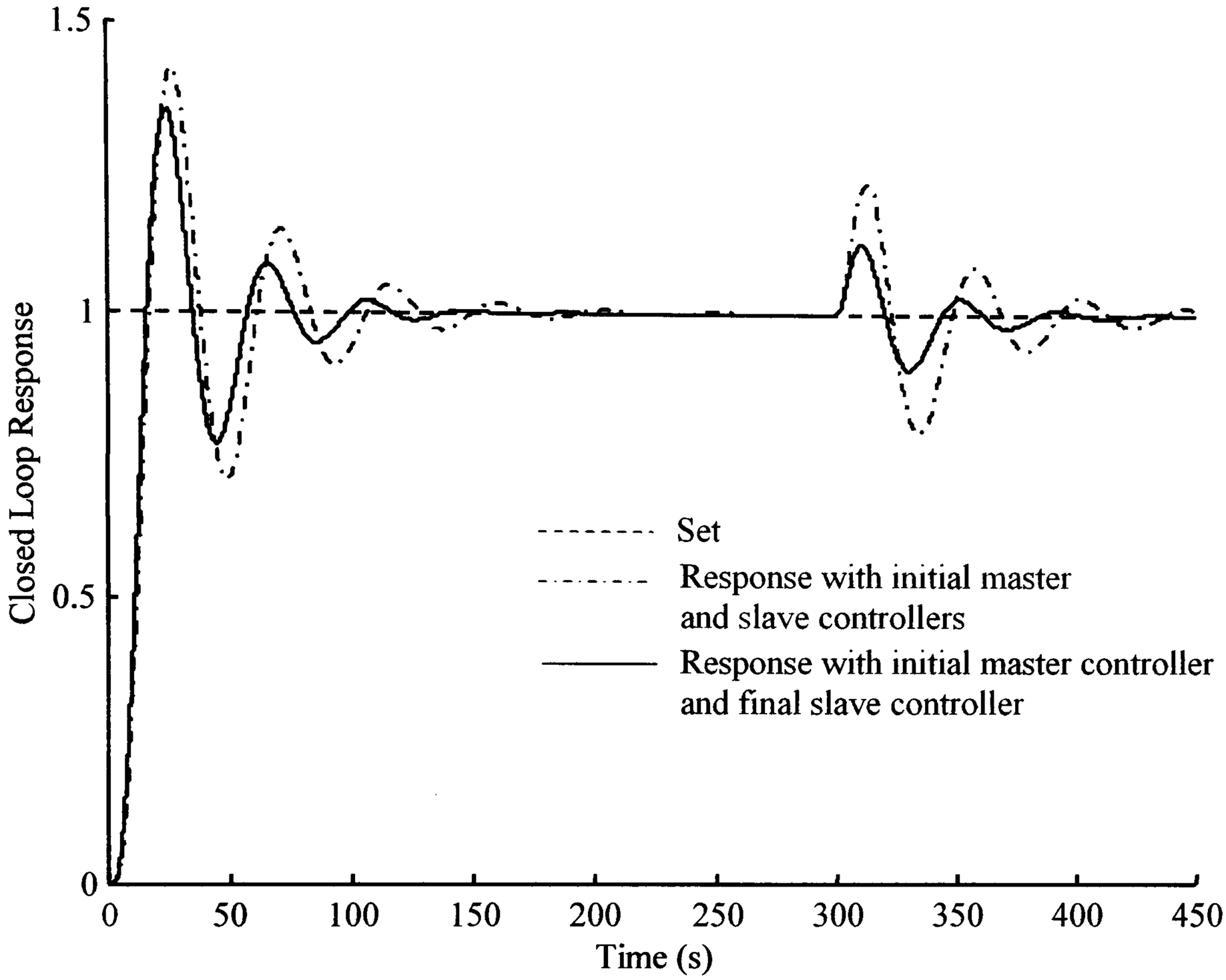


Figure 4.7: Cascade System Step and Load Disturbance Response with $G_{c2}^{'}(s)$.

Table 4.6 gives a frequency domain comparison between the original inner loop Ziegler and Nichols ultimate period method compensated forward path and the gain and phase margin compensated forward path.

Table 4.6: Gain and Phase Margin Data for the Inner Forward Path		
Forward Path	$G_{p2}(s)G_{c2}(s)$	$G_{p2}(s)G_{c2}^{'}(s)$
Gain Margin	2.39	2
Phase Crossover Frequency (rad.s^{-1})	2.2427	2.0144
Phase Margin (degree)	113	60
Gain Crossover Frequency (rad.s^{-1})	0.4633	0.8540

From Table 4.6 it can be seen that the design gain and phase margin values have been attained. The bandwidth, as defined by the gain crossover frequency, of the gain and phase margin compensated system can be seen to be increased over that of the original compensated forward path.

Having tuned the inner loop PI controller, a similar design process will now be carried out for the outer controller. However, the design intent for the master loop controller is to produce an improvement in the time domain response of the closed loop cascade system and to improve the stability robustness properties. In this case, the closed loop identification of the composite process shown as $G_{p1}^{'}(s)$ in Figure 4.2 is carried out. Again sixteen points on the frequency response curve of the composite process $G_{p1}^{'}(s)$ are identified for use by the gain and phase margin viable pairing method. The phase angle range chosen for the identification is from $-\frac{\pi}{6}$(rad) to $-\pi$ (rad). The results of the identification are shown in Table 4.7. The time taken to complete the identification of all sixteen points was approximately 4 hours and 36 minutes. The length of time taken to complete the identification may be considered to be lengthy however the identification was carried out in closed loop with low excitation magnitudes used hence the disruption to the process would be minimised. The excitation frequencies required to carry out the identification are very low and thus have a long period.

Table 4.7: Identification Results for Composite Process, $G_{p1}^{'}(s)$

$$G_{p1}^{'}(s) = \frac{G_{p2}(s)G_{c2}^{'}(s)}{1 + G_{p2}(s)G_{c2}^{'}(s)} G_{p1}(s)$$

Magnitude	Phase Angle (rad)	Excitation Frequency (rad.s^{-1})	Time (s)
19.004	-0.5232	0.0226	2527
18.259	-0.6977	0.0305	3623
17.335	-0.8724	0.0387	5025
16.263	-1.0469	0.0473	6365
15.068	-1.2214	0.0564	7599
13.783	-1.3960	0.0661	8680
12.443	-1.5707	0.0766	10067
11.082	-1.7450	0.0879	11391
9.7479	-1.9195	0.1004	12036
8.4338	-2.0942	0.1142	12726
7.2182	-2.2686	0.1295	13269
6.0598	-2.4430	0.1466	13820
5.0225	-2.6177	0.1660	14813
4.1069	-2.7929	0.1873	15253
3.3213	-2.9675	0.2120	15995
2.6382	-3.1420	0.2396	16527

Further the amount of operator input required whilst the identification is being carried out is minimal requiring only the input of the identification range and the number of points to be identified. The identification is then carried out autonomously by the Phase-Locked Loop identifier module. The results of the identification are shown graphically in Figure 4.8 where it can be seen that the identified data are in good agreement with the theoretical frequency response curve of the composite process, $G_{p1}^{'}(s)$.

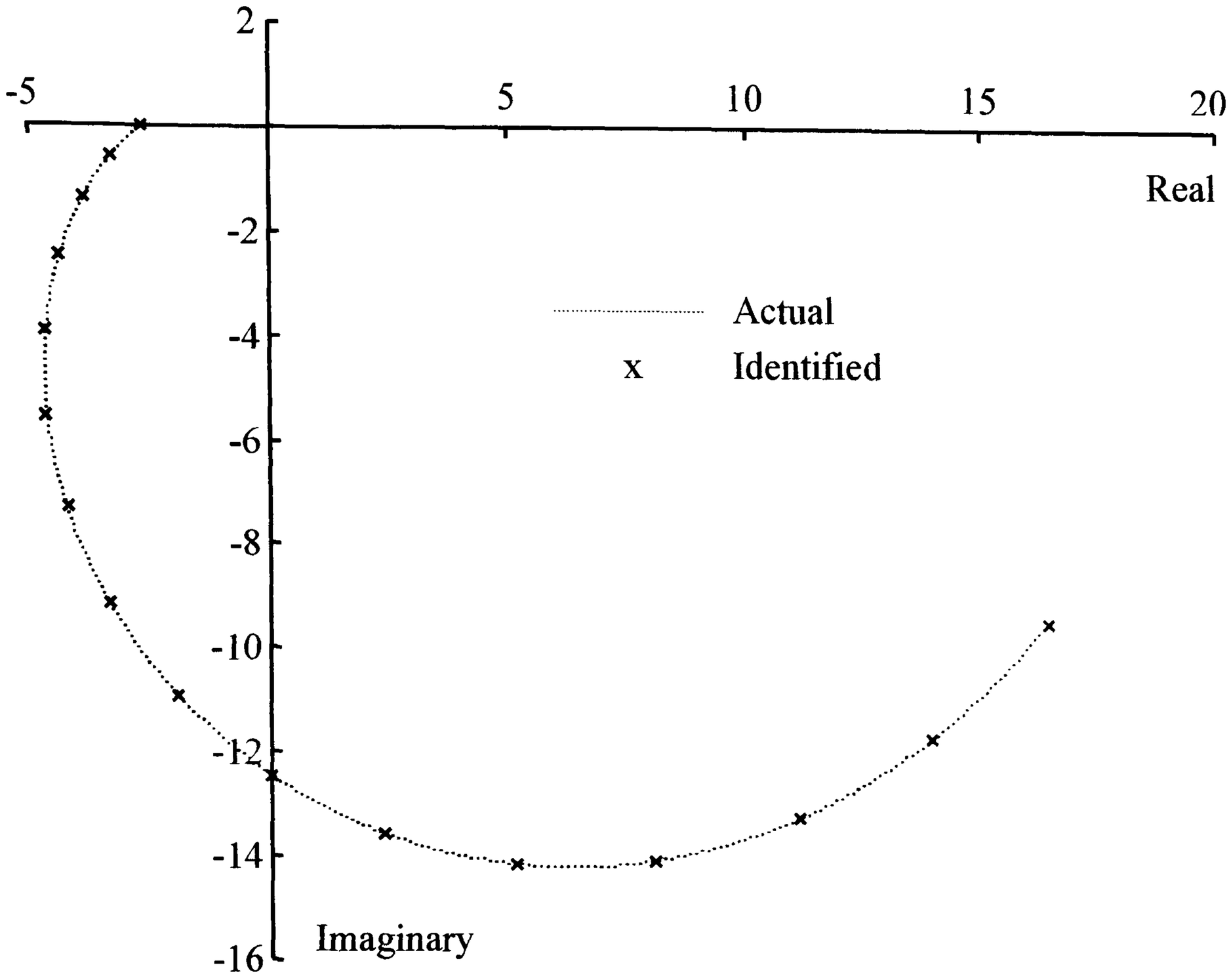


Figure 4.8: Frequency Response of Composite Process, $G_{p1}^{'}(s)$

When the identification phase is complete the data obtained is used by the gain margin and phase margin viable pairing routine. The results of this are shown in Figure 4.9. The shaded region of the figure represents the viable gain and phase margin pairings that can be achieved by a PI controller acting on the composite process, $G_{p1}^{'}(s)$.

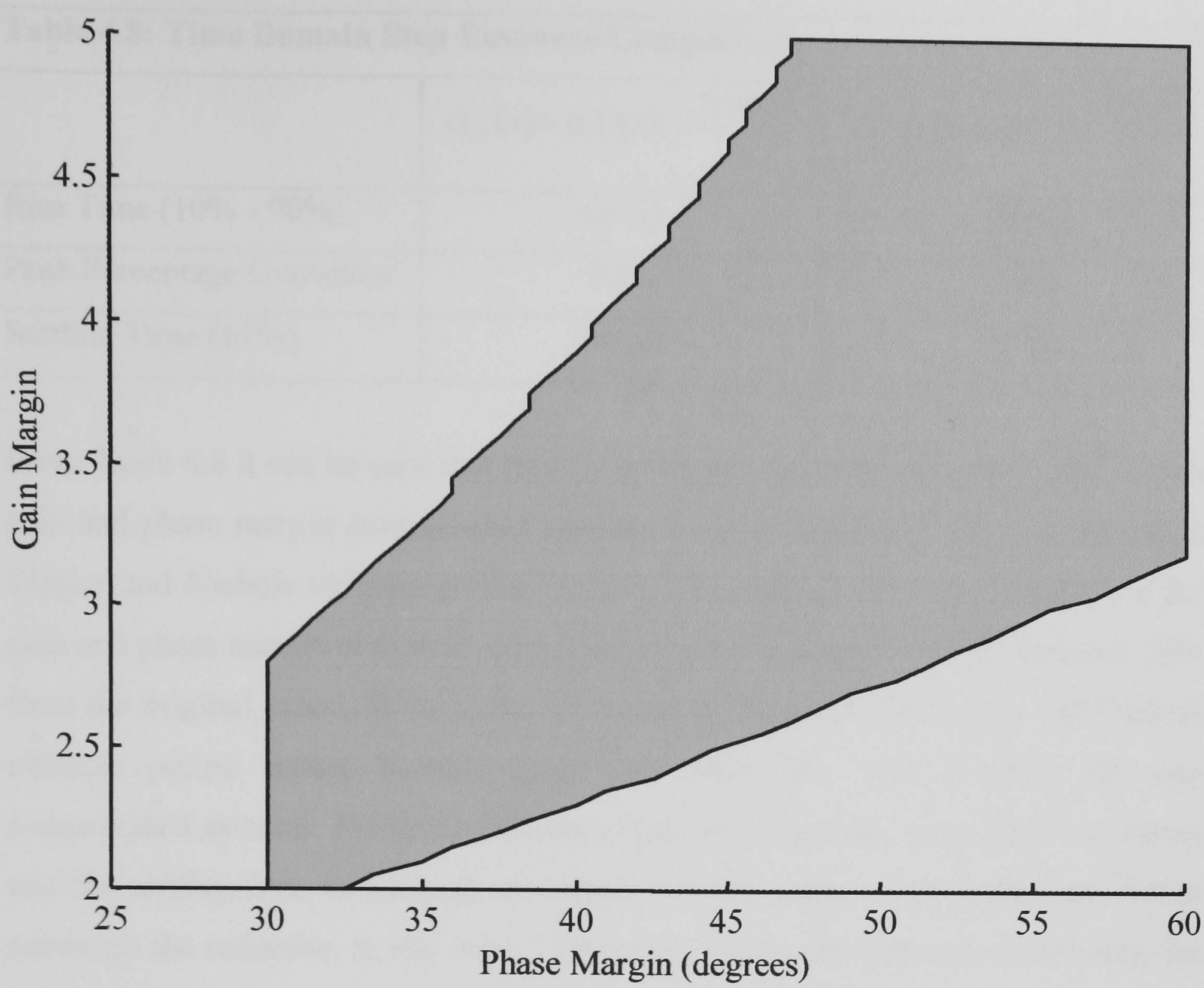


Figure 4.9: Gain and Phase Margin Pairings for the Composite Process, $G_{p1}^{'}(s)$

The design objective is to choose a viable gain margin and phase margin pairing that will produce an improvement in the time domain step response of the closed loop cascade system. To achieve this, the gain and phase margin pairing chosen was a gain margin of 4 and a phase margin of 60°. This choice of gain margin and phase margin resulted in a PI controller given by

$$G_{c1}^{'}(s) = 0.0712 + \frac{0.0031}{s}$$

The comparison test between the new outer loop PI controller and the original Ziegler and Nichols tuned controller was made on a similar basis as the inner controller comparison, viz. a step input applied at time t=0(s) and a load disturbance of magnitude 0.15 applied to the inner loop at time t=300(s). The results of the comparison test are shown in Table 4.8.

Table 4.8: Time Domain Step Response Comparison		
	$G_{c1}(s) = 0.1323 + \frac{0.0054}{s}$	$G'_{c1}(s) = 0.0712 + \frac{0.0031}{s}$
Rise Time (10% - 90%)	10 (s)	15 (s)
Peak Percentage Overshoot	35%	7.3%
Settling Time (±2%)	94 (s)	77 (s)

From Table 4.8 it can be seen that there is an improvement in the settling time of the gain and phase margin compensated system of approximately 17 (s) over the initial Ziegler and Nichols ultimate period method. The peak percentage overshoot of the gain and phase margin compensated system has been reduced by approximately 28% from the original value. From Table 4.8 it can be seen that the Ziegler and Nichols ultimate period tuning method gives the better rise time between the two compensated systems. However the improvements in the peak percentage overshoot and the settling time of the gain and phase margin compensated system are felt to outweigh the reduction in rise time. Figure 4.20 shows the comparison between the step and disturbance responses of the cascade systems. From Figure 4.20 it can be seen that the final tuning of the cascade system has a slower rise time and a reduced level of peak percentage overshoot to that of the cascade system using the outer loop PI controller tuned using Ziegler and Nichols ultimate period tuning rules. The load disturbance rejection properties show that the new tuning settings do give a faster rejection of the disturbance than the initial Ziegler and Nichols ultimate period version along with a reduction in the integral of the square of the reference error.

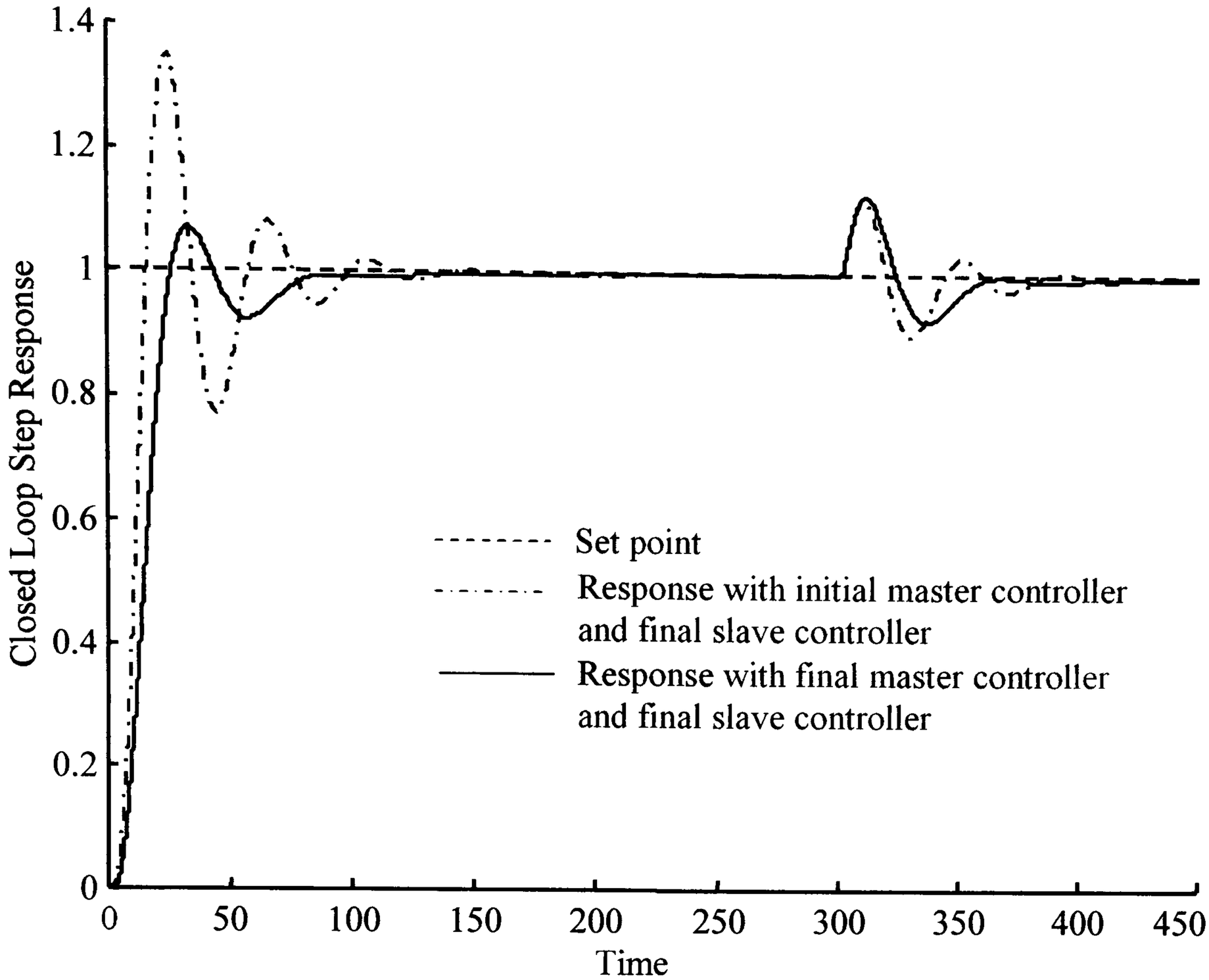


Figure 4.20: Closed Loop Response of Cascade Control System.

4.4 Identification and Tuning of Multivariable Control Systems.

In the process industries when there is a requirement to control interacting multi-input multi-output systems, decentralised control using PID controllers is often chosen as the control methodology (Palmor *et al*, 1995) over a full matrix multivariable control system that would deliver a potentially improved control performance. Decentralised control is still chosen since its simpler structure is easier to implement and understand by industry-based control practitioners. There is also a benefit to be derived from the reduced vulnerability of the control system to actuator and sensor failure since in decentralised control systems only one loop is affected by the failure. In the case of single-input single-output systems the relay experiment of Astrom and Hagglund has been used extensively as the basis for auto-tuning PID control systems. When the relay experiment is applied to the non-parametric

identification and tuning of multivariable control systems there are effectively three methods reported in the literature:

i) Independent relay feedback

ii) Sequential relay feedback, and

iii) Simultaneous decentralised relay feedback.

Independent Relay Feedback.

In independent relay feedback a relay is connected to each input in turn. The corresponding output is closed around the relay with all other loops left open. The identification process is continued until each loop has been identified. This method of multivariable system identification allows particular points of the system frequency response matrix to be identified on a column by column basis. Figure 4.21 shows how independent relay feedback identification can be implemented on a two-input, two-output system.

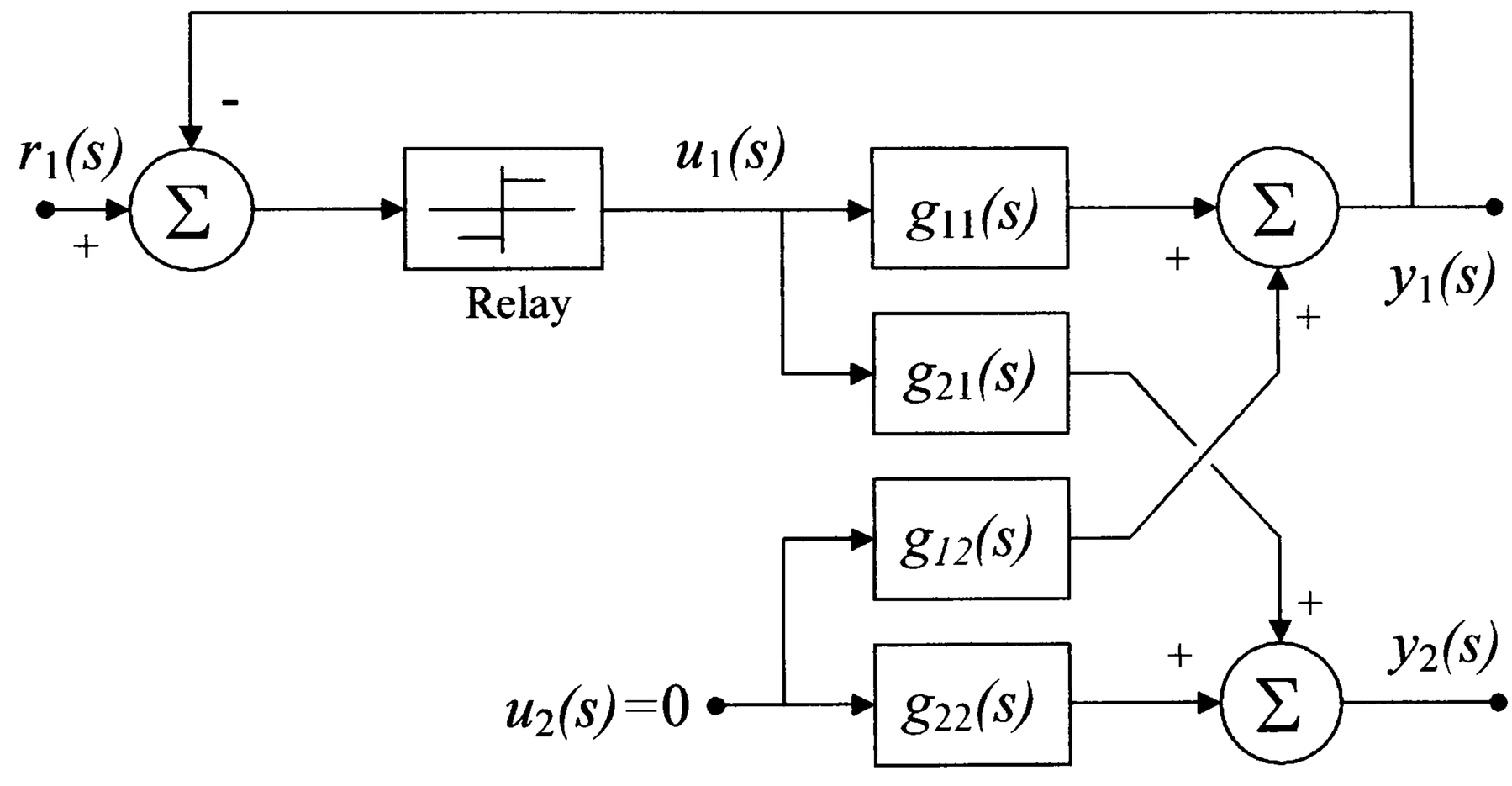


Figure 4.21: Independent Relay Feedback.

Sequential Relay Feedback.

In sequential relay feedback each loop is identified using a relay experiment and a PID controller is designed on the basis of the information supplied from that

relay experiment. As each loop is identified it is subsequently closed by the newly designed PID controller. The procedure of identification, PID controller design and loop closure around the new controller continues until all loops are closed. The procedure is then iterated until the controller parameters have converged. A popular choice of controller tuning technique is a rule based method such as the Ziegler and Nichols ultimate period method. Use of the Ziegler and Nichols ultimate period method of PID controller tuning can lead to unstable designs (Yu, 1999).Yu (1999) proposes that a detuning factor is applied to the Ziegler and Nichols ultimate period rules such that stable controllers may result from the design. Figure 4.22 shows the conceptual diagram of sequential relay feed back identification and tuning for a two-input, two-output system.

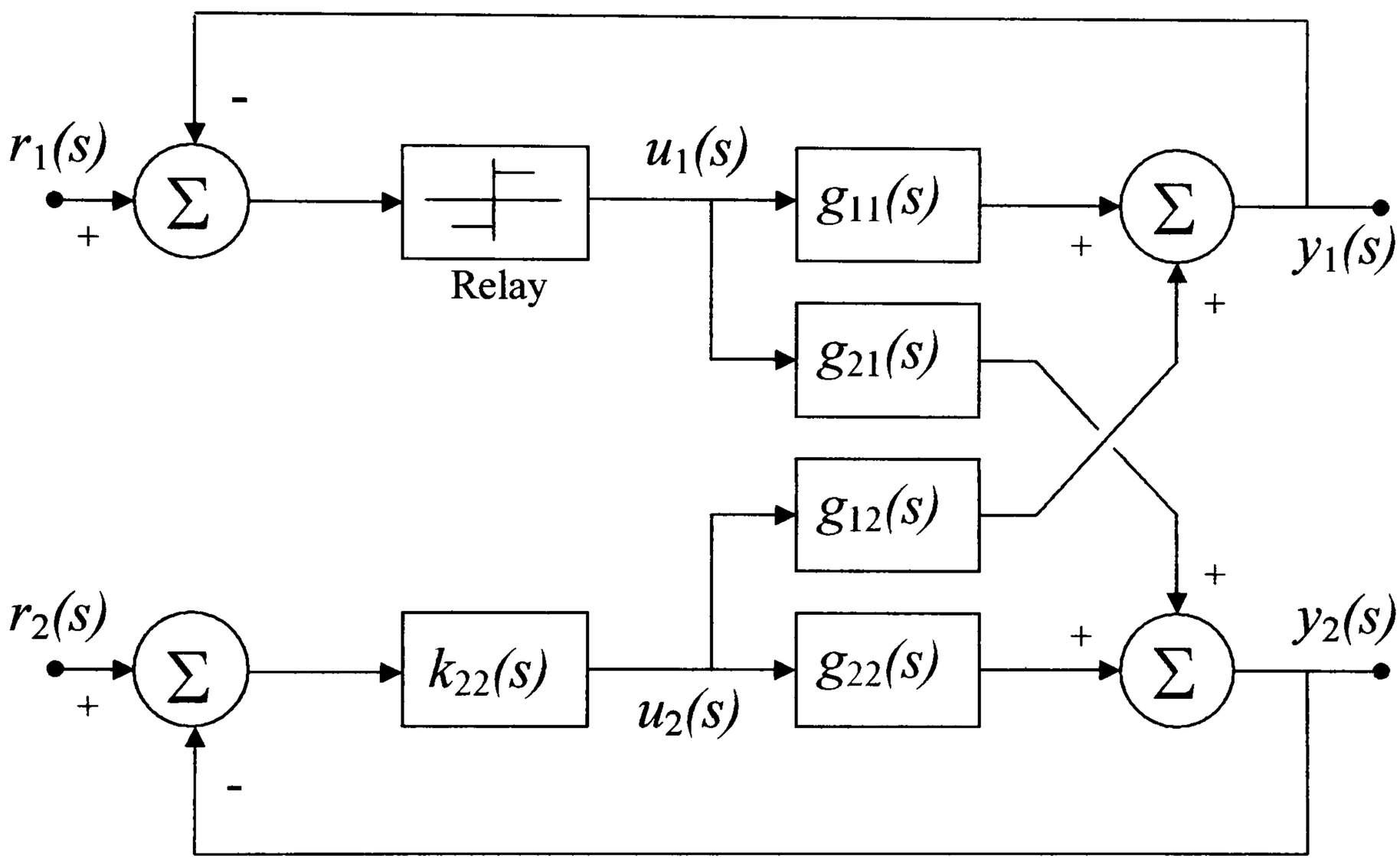


Figure 4.22: Sequential Relay Feedback Conceptual Diagram.

Simultaneous Decentralised Relay Feedback.

The conceptual diagram for decentralised relay feedback applied to a two-input, two-output system is shown in Figure 4.23. As can be seen from the figure both process

inputs are connected to relays and the corresponding process outputs are closed around the relays.

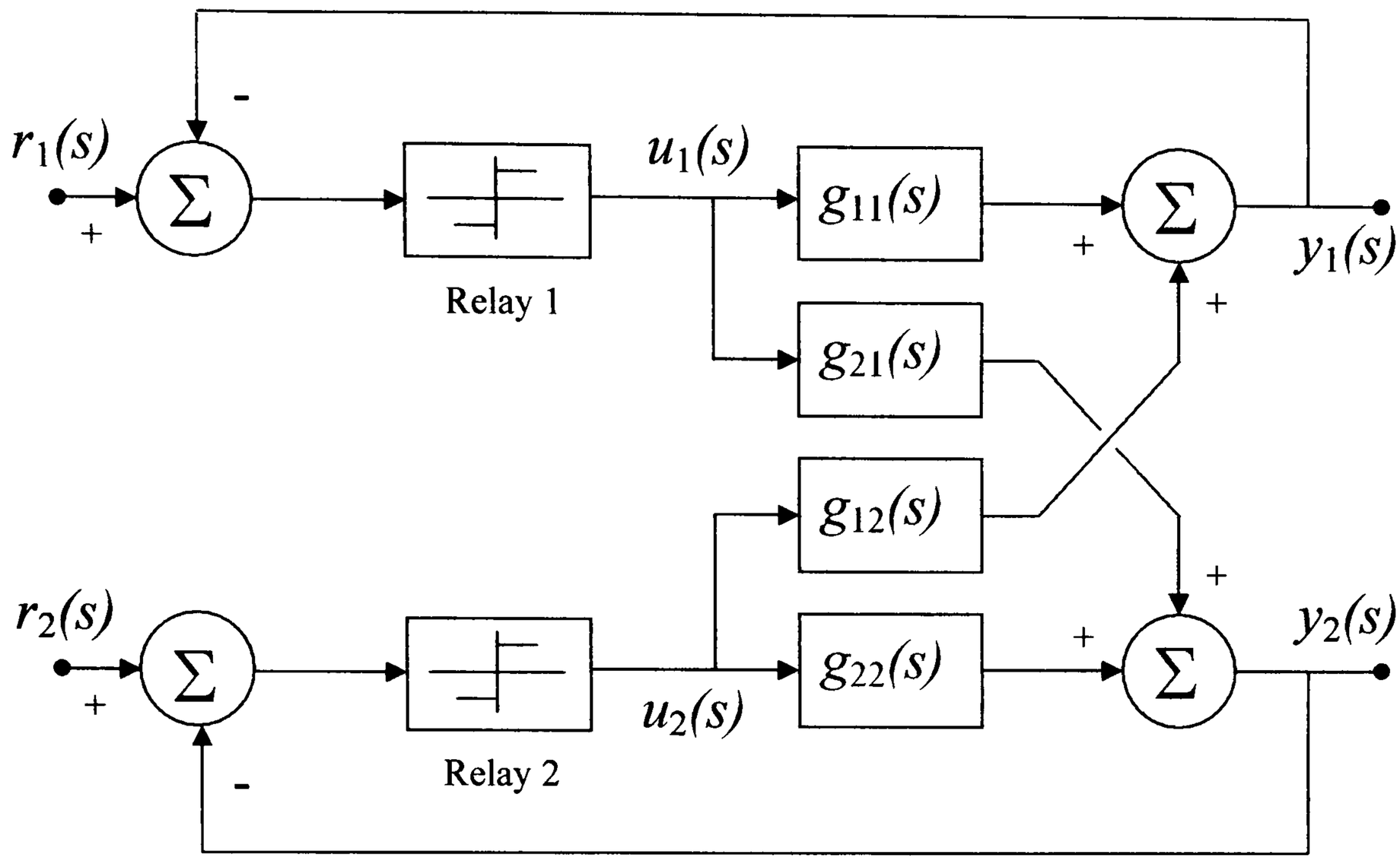


Figure 4.23: Conceptual Diagram for a two-input, two-output process.

Palmor *et al* (1995) discusses a method that allows the identification of a desired critical point in the gain space of a two-input, two-output multivariable system. In Figure 4.24 the curve shown as (1) represents the case where the two loops have little or no interaction, hence the system becomes unstable when any of the two loop gains, $K_i g_{ii}(0)$ i=1,2, exceeds their single-input single-output critical gains.

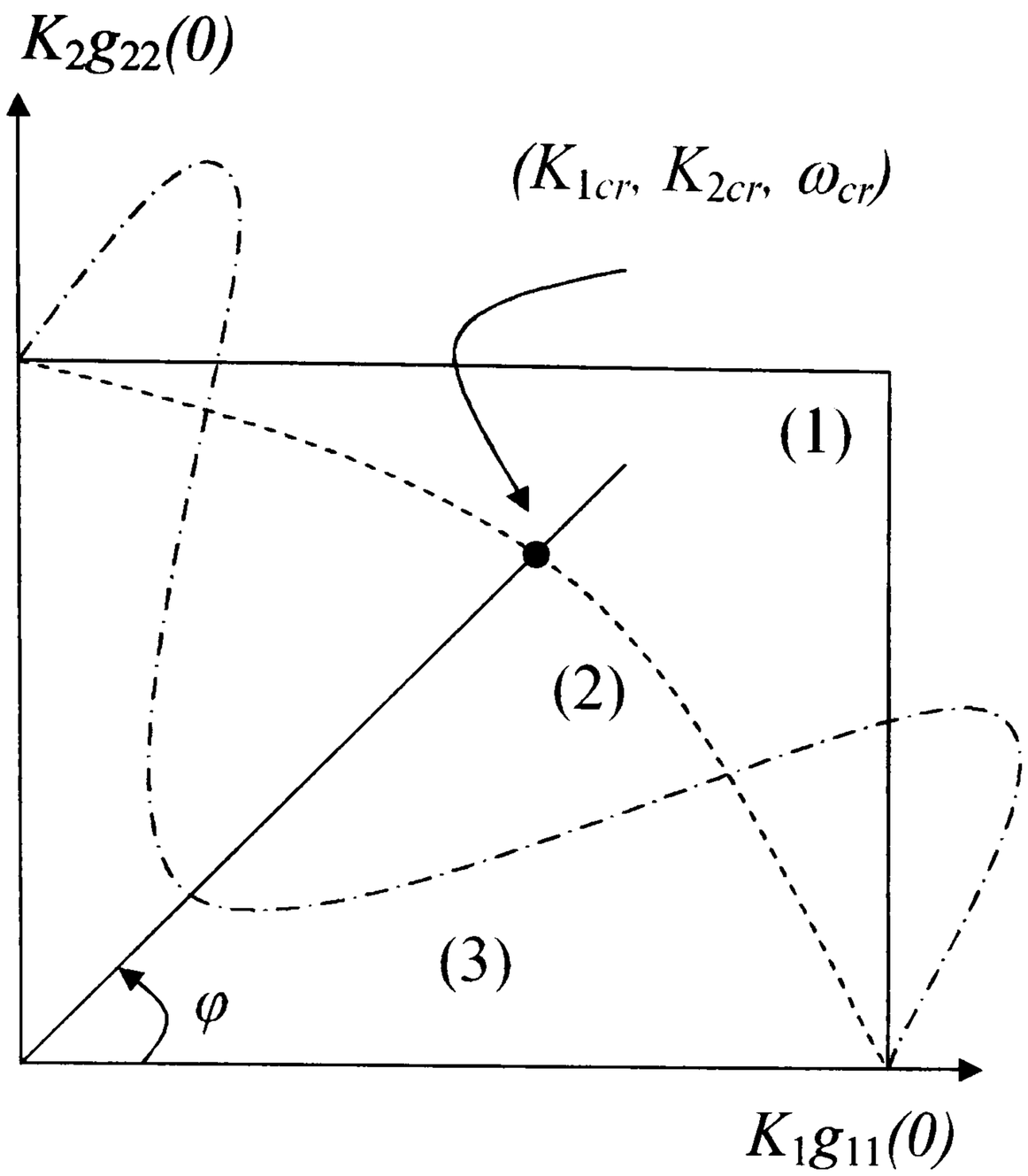


Figure 4.24: Stability Boundaries: Three Special Cases (Palmor *et al*, 1995)

The other two curves, shown as (2) and (3) in Figure 4.24, represent the general cases where coupling between the two loops exists.

Palmor *et al* (1995) state that stable limit cycles are generated, each with the same period, if the ratio of the relay heights is equal to a constant. Hence by modifying the relay height ratio it is possible to move the critical point along the stability boundary. Palmor *et al*, define a desired critical point in the gain space by selecting a particular angle, shown as ϕ in Figure 4.24, or by selecting a particular weighting factor. The weighting factor determines the weighting of loop two relative to loop one. If the weighting factor is chosen to be greater than unity then loop two will produce an improved control performance over that produced by loop one, by ensuring that loop two has a larger critical gain than that of loop one. The relationship between ϕ and the weighting factor, C_d, is given in (Palmor *et al*, 1995) as,

$$\tan\phi = C_d = \frac{K_{2_{cr}} g_{22}(0)}{K_{1_{cr}} g_{11}(0)} \tag{4.37}$$

where $K_{1_{cr}}$ and $K_{2_{cr}}$ are the critical gains of loop one and loop two respectively and $g_{11}(0)$ and $g_{22}(0)$ are the zero frequency gains of the leading diagonal elements of the two-input, two-output multivariable process. The zero frequency gains $g_{11}(0)$ and $g_{22}(0)$ require to be identified. To accomplish this Palmor *et al* carry out two experiments. With the relays connected in closed loop and the system oscillating at a critical frequency, the set point input of loop one is given a non-zero value with the set point input of loop two made equal to zero. The average values of the multivariable system input and output signals are then calculated, such that

$$\bar{y} = G(0)\bar{u}$$

where,
$$\bar{y} = \begin{bmatrix} \frac{1}{T}\int_0^T y_1(t)dt \\ \frac{1}{T}\int_0^T y_2(t)dt \end{bmatrix} \text{ and } \bar{u} = \begin{bmatrix} \frac{1}{T}\int_0^T u_1(t)dt \\ \frac{1}{T}\int_0^T u_2(t)dt \end{bmatrix} \tag{4.38}$$

and T is the period of the critical frequency. The experiment is carried out once again, this time with the set point input of loop one made equal to zero and the set point input of loop two made non-zero, this leads to

$$\begin{bmatrix} \bar{y}^1 & \bar{y}^2 \end{bmatrix} = G(0)\begin{bmatrix} \bar{u}^1 & \bar{u}^2 \end{bmatrix} \tag{4.39}$$

The superscripts in equation (4.39) represent the experiment number. From equation (4.39) it is possible to calculate the values for the zero frequency gains of the elements of the two-input, two-output multivariable process.

The next step in the procedure is to choose the desired critical point either by selecting a desired ϕ_d or by selecting a desired weighting factor, C_d. The relay heights are now selected, assume that the relay heights are chosen as h_1 and h_2 for relay 1 and relay 2 respectively. From this point the Palmor *et al* decentralised relay feedback method follows the following steps:

i) The first experiment is carried out using the relay heights h_1 and h_2 and the peak amplitudes of the limit cycles in loop one, a_1, and loop two, a_2, are recorded as is the limit cycle frequency.

ii) In the second experiment the relay heights are swapped over and the peak amplitudes of the limit cycles in loop one, a_1, and loop two, a_2, are recorded as is the limit cycle frequency. By using this technique the critical points identified should be on opposite sides of the desired critical point. From the results of the two relay experiments a new value for the relay height ratio is calculated from

$$\frac{h_1}{h_2} = \left| \frac{c}{C_d \left| \frac{g_{11}(0)}{g_{22}(0)} \right| - b} \right| \tag{4.40}$$

where $b = \dfrac{\left(\dfrac{a_1}{a_2}\right)_1 - \left(\dfrac{a_1}{a_2}\right)_2}{\left(\dfrac{h_1}{h_2}\right)_1 - \left(\dfrac{h_1}{h_2}\right)_2}$ and $c = \left(\dfrac{a_1}{a_2}\right)_1 - b\left(\dfrac{h_1}{h_2}\right)_1$ and the subscripts on the bracketed terms refer to the experiment number.

iii) The error between the desired critical point and the identified critical point is now tested. If $|\phi - \phi_d| < tol$ then stop.

iv) If the error tolerance is not met then the decentralised relay experiments are continued at step i) using the relay heights calculated in step ii).

When the desired critical point has been identified then the tuning of the decentralised controllers is carried out using a rule based PID controller tuning technique. It should be noted that when the desired critical point has been identified, there is only one critical frequency hence both decentralised PID controllers will have the same values for their integral and derivative time constants.

In the above, the methods all employ the steps of non-parametric identification followed by controller tuning using the application of a rule based PID method. Additionally all of the methods lead to the design of decentralised PID controllers. In the tuning method reported by Wang *et al* (1997b) a different approach is taken to determining the PID controller parameters. The identification of

the multivariable process is accomplished by the use of decentralised relay experiments. However, the aim of the identification is to provide data for two points on the frequency response curves of each element of the transfer function matrix of the process so that a first order plus dead time model of each element of the process transfer function matrix can be determined. The zero frequency gains of the elements of the transfer function matrix are required as are the critical frequency gains of the elements of the transfer function matrix. The critical frequency being the frequency of the resulting limit cycle oscillations set up by the decentralised relay experiment, with the assumption that there is only one frequency present. To carry out this identification Wang *et al* employ a biased relay in one of the loops and standard relays are placed in all of the remaining loops to form a decentralised relay experiment. By using a biased relay, asymmetrical waveforms are generated at the inputs and outputs of the multivariable process. If it is assumed that the multivariable process to be identified is given by $G_p(s)$ and it is of dimension m x m, then the required gains are extracted from the process input and output data using the following (Wang *et al*, 1997b).

$$U(0)=\begin{bmatrix} \int_0^{T_c} u_1(t)dt \\ \vdots \\ \int_0^{T_c} u_m(t)dt \end{bmatrix}$$

$$Y(0)=\begin{bmatrix} \int_0^{T_c} y_1(t)dt \\ \vdots \\ \int_0^{T_c} y_m(t)dt \end{bmatrix}$$

and

$$U(j\omega_c)=\begin{bmatrix}\int_0^{T_c} u_1(t)e^{-j\omega_c t}\,dt\\ \vdots\\ \int_0^{T_c} u_m(t)e^{-j\omega_c t}\,dt\end{bmatrix}$$

$$Y(j\omega_c)=\begin{bmatrix}\int_0^{T_c} y_1(t)e^{-j\omega_c t}\,dt\\ \vdots\\ \int_0^{T_c} y_m(t)e^{-j\omega_c t}\,dt\end{bmatrix}$$

hence obtain

$$Y(0)=G_p(0)U(0)$$

$$Y(j\omega_c)=G_p(j\omega_c)U(j\omega_c)$$

where T_c and ω_c are the critical period and critical frequency respectively.

As in the method of Palmor *et al* (1995), because the process inputs and outputs are vectors it is not possible to directly determine the transfer function matrix element gains from one experiment. To overcome this difficulty, Wang *et al* carry out *m*-1 further decentralised relay experiments, calculating the transfer function element gains as above for each experiment. Thus after *m*-decentralised relay experiments have been carried out, the following are obtained

$$\begin{bmatrix}Y^1(0) & \dots & Y^m(0)\end{bmatrix}=G_p(0)\begin{bmatrix}U^1(0) & \dots & U^m(0)\end{bmatrix}$$

$$\begin{bmatrix}Y^1(j\omega_c) & \dots & Y^m(j\omega_c)\end{bmatrix}=G_p(j\omega_c)\begin{bmatrix}U^1(j\omega_c) & \dots & U^m(j\omega_c)\end{bmatrix}$$

where the superscript denotes the experiment number.

From the above the transfer function gains at zero frequency, $G_p(0)$, and the critical frequency gains, $G_p(j\omega_c)$, are found. The final stage in the Wang *et al* method is the design of the controller for the multivariable system. Using the identification data for the frequency response of the transfer function matrix, obtained from the decentralised relay experiments, first order plus dead time models of the individual transfer function elements are fitted to the data. The objective of the Wang *et al* controller design method is to produce a de-coupled closed loop system and to have an acceptable control system response from the independent loops. Consider the

multivariable control system shown in Figure 4.25. The process, $G_p(s)$, is m-input m-output and the controller $K(s)$ is a full matrix controller.

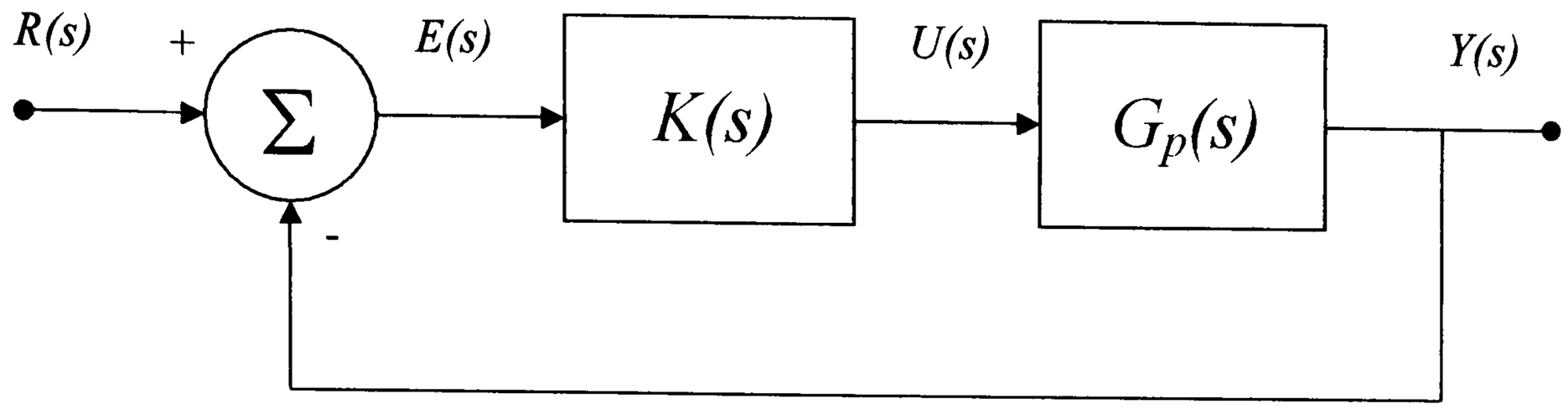


Figure 4.25: Multivariable Control System

For the system shown in Figure 4.25 it can be shown that

$$Y(s)=\left(I_{mxm}+G_p(s)K(s)\right)^{-1}G_p(s)K(s)R(s) \qquad (4.41)$$

where I_{mxm} is the m x m identity matrix.

It is known that the product or sum of two diagonal matrices returns a diagonal matrix as does the inverse of a diagonal matrix. Hence in equation (4.41) to ensure that the closed loop system is de-coupled the matrix $G_p(s)K(s)$ must be made diagonal, viz.

$$G_p(s)K(s)=diag[q_{ii}],\ i = 1, 2, \ldots, m$$

Assuming that de-coupling can be achieved, the design of the diagonal controllers of the full matrix controller can be carried out having knowledge of the leading diagonal elements of the transfer function matrix. In Wang *et al* the design of the controllers on the leading diagonal of the full matrix controller is carried out using the method proposed by Ho *et al* (1995) where analytical expressions are given that enable a PI controller to be designed to produce a specified gain margin and phase margin. With the diagonal controllers designed, it remains to determine the off-diagonal controllers. The off-diagonal controllers are designed such that for any column, i = 1 2, ... , m, the off-diagonal controllers are given in terms of their respective diagonal controllers, thus

$$\begin{bmatrix} k_{1,i}(s) \\ \vdots \\ k_{i-1,i}(s) \\ k_{i+1,i}(s) \\ \vdots \\ k_{m,i}(s) \end{bmatrix} = -\begin{bmatrix} g_{1,i}(s) & \cdots & g_{1,i-1}(s) \\ \vdots & \ddots & \vdots \\ g_{i-1,i}(s) & \cdots & g_{i-1,i-1}(s) \\ g_{i+1,i}(s) & \cdots & g_{i+1,i-1}(s) \\ \vdots & \ddots & \vdots \\ g_{m,i}(s) & \cdots & g_{m,i-1}(s) \end{bmatrix}$$

$$\begin{bmatrix} g_{1,i+1}(s) & \cdots & g_{1,m}(s) \\ \vdots & \ddots & \vdots \\ g_{i-1,i+1}(s) & \cdots & g_{i-1,m}(s) \\ g_{i+1,i+1}(s) & \cdots & g_{i+1,m}(s) \\ \vdots & \ddots & \vdots \\ g_{m,i+1}(s) & \cdots & g_{m,m}(s) \end{bmatrix}^{-1} \begin{bmatrix} g_{1,i}(s) \\ \vdots \\ g_{i-1,i}(s) \\ g_{i+1,i}(s) \\ \vdots \\ g_{m,i}(s) \end{bmatrix} k_{ii}(s)$$

The above off-diagonal controller design method is discussed in "Industrial Digital Control Systems" (Warwick and Rees, Eds., 1988) where the authors of the method are given as Boksenbom and Hood (1949). All of the above methods utilise a relay experiment or some combination of relay experiments to identify the multivariable process. In the following a decentralised controller design method is discussed that assumes the availability of a process model.

Gain margin and phase margin design is an accepted method of specifying frequency domain characteristics of a single-input single-output control system. In Ho *et al* (1995) the extension of gain margin and phase margin design to a multivariable process controlled by a decentralised PID control system is discussed. In classical gain and phase margin design two points are specified and the frequency response curve of the compensated forward path of the control system is required to pass through those points. In the method by Ho *et al* this idea is developed such that the Gershgorin Band is shaped such that it passes through two specified points. The two points are defined in a similar manner to the gain and phase margin of single-input single-output systems.

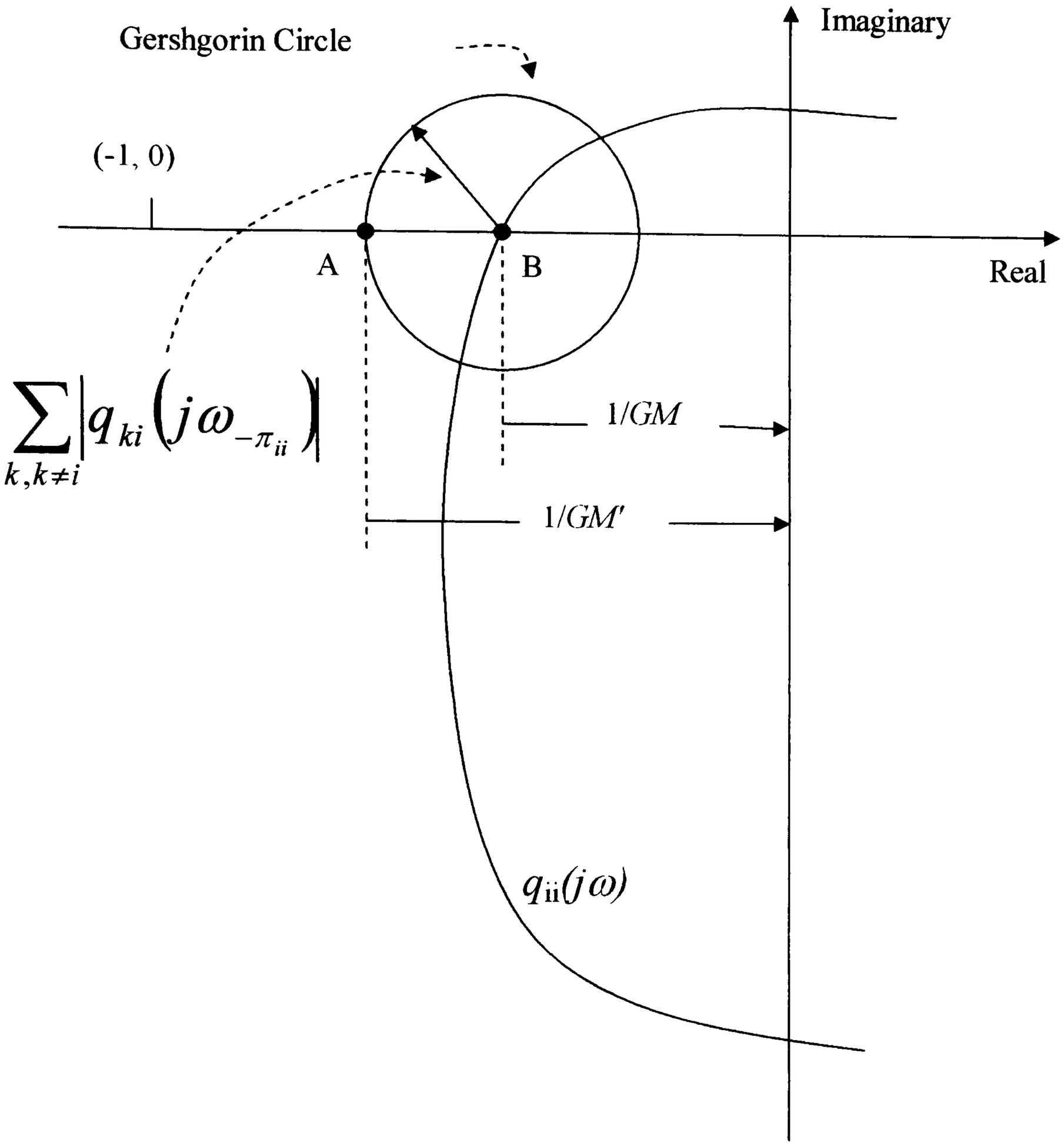


Figure 4.26: Typical Nyquist Diagram with Gershgorin Circle Shown at the Phase Crossover Frequency (Ho *et al*, 1995).

The diagram shown in Figure 4.26 is used to define the multivariable gain margin, *GM'*.

$$GM' = \frac{1}{\left|q_{ii}\left(j\omega_{-\pi_{ii}}\right)\right| + \sum_{k,k\neq i}\left|q_{ki}\left(j\omega_{-\pi_{ii}}\right)\right|} \tag{4.42}$$

where $\omega_{-\pi_{ii}}$ is the frequency at which $\arg\left(q_{ii}\left(j\omega_{-\pi_{ii}}\right)\right) = -\pi$ and $q_{ii}(j\omega)$ is the compensated forward path frequency response diagonal element, $q_{ii}(s)$. From equation (4.42) and Figure 4.26 it follows that the gain margin, GM, is given by

$$GM' = \frac{1}{\left|q_{ii}\left(j\omega_{-\pi_{ii}}\right)\right| + \sum_{k,k\neq i}\left|q_{ki}\left(j\omega_{-\pi_{ii}}\right)\right|}$$

$$GM' = \frac{GM}{1 + \dfrac{\sum_{k,k\neq i}\left|q_{ki}\left(j\omega_{-\pi_{ii}}\right)\right|}{\left|q_{ii}\left(j\omega_{-\pi_{ii}}\right)\right|}}$$

hence

$$GM = GM'\left(1 + \frac{\sum_{k,k\neq i}\left|q_{ki}\left(j\omega_{-\pi_{ii}}\right)\right|}{\left|q_{ii}\left(j\omega_{-\pi_{ii}}\right)\right|}\right) \qquad (4.43)$$

and thus

$$GM = GM'\left(1 + \frac{\sum_{k,k\neq i}\left|g_{ki}\left(j\omega_{-\pi_{ii}}\right)\right|}{\left|g_{ii}\left(j\omega_{-\pi_{ii}}\right)\right|}\right) \qquad (4.44)$$

where $GM = \dfrac{1}{\left|q_{ii}\left(j\omega_{-\pi_{ii}}\right)\right|}$ and equation (4.44) follows from equation (4.43) since decentralised control is used.

The diagram shown in Figure 4.27 is used to define the multivariable phase margin, ϕ'_{PM}.

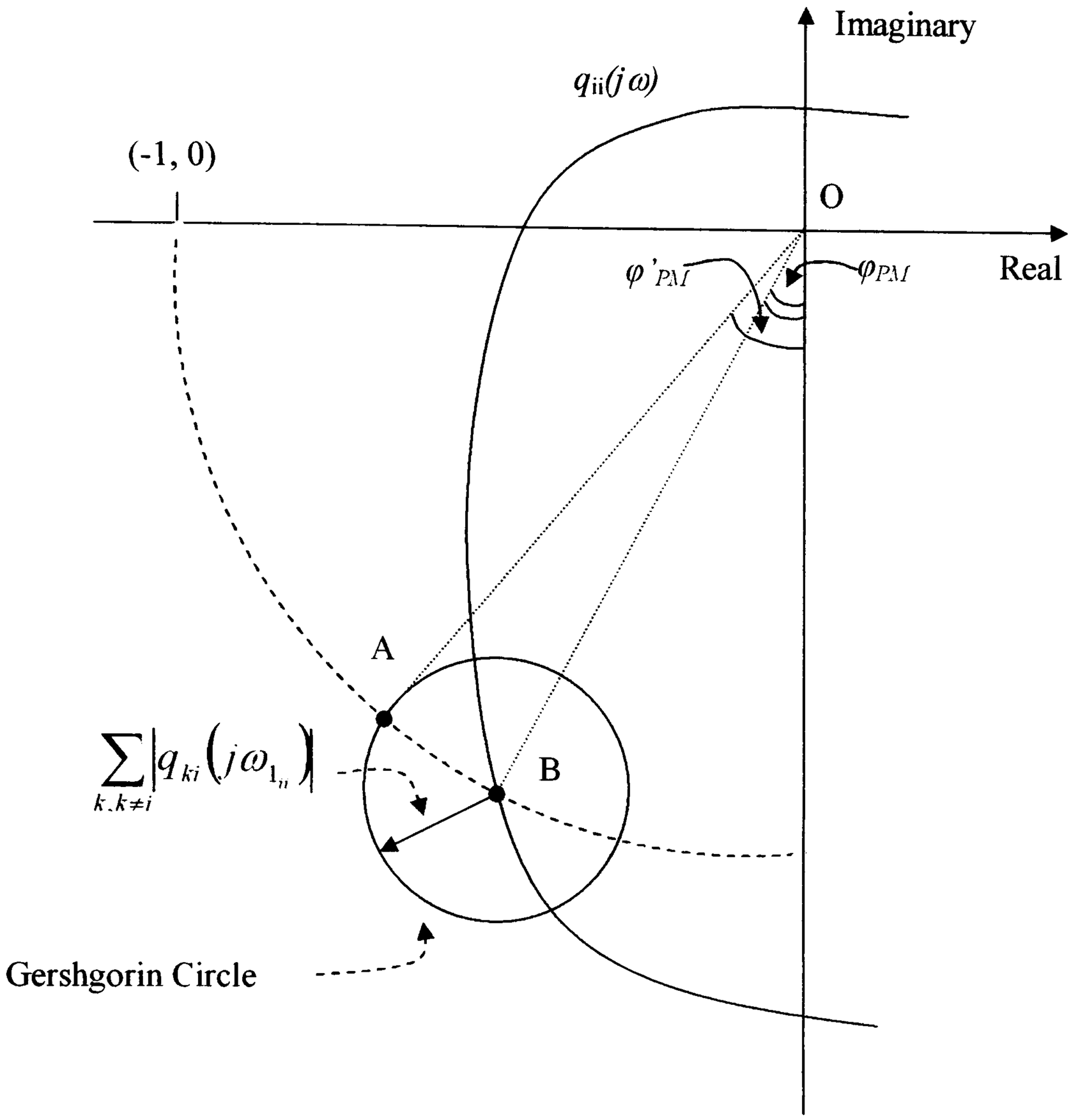


Figure 4.27: Typical Nyquist Diagram with Gershgorin Circle Shown at the Gain Crossover Frequency (Ho *et al*, 1995).

From Figure 4.27 it can be seen that *OA* is equal to *OB* and both are equal to unity. If a line is drawn from the centre of *AB* to the point *O*, then it can be seen that

$$\sin\left(\frac{\phi'_{PM}-\phi_{PM}}{2}\right)=\frac{\sum_{k,k\neq i}\left|q_{ki}\left(j\omega_{1_{ii}}\right)\right|}{2\left|q_{ii}\left(j\omega_{1_{ii}}\right)\right|}$$

hence

$$\phi_{PM}=\phi'_{PM}+2\sin^{-1}\left(\frac{\sum_{k,k\neq i}\left|q_{ki}\left(j\omega_{1_{ii}}\right)\right|}{\left|q_{ii}\left(j\omega_{1_{ii}}\right)\right|}\right) \tag{4.45}$$

Equation (4.45) can be written in the form

$$\phi_{PM} = \phi_{PM}^{'} + 2\sin^{-1}\left(\frac{\sum_{k,k\neq i}\left|g_{ki}\left(j\omega_{1_{ii}}\right)\right|}{\left|g_{ii}\left(j\omega_{1_{ii}}\right)\right|}\right) \tag{4.46}$$

where $\omega_{1_{ii}}$ is the frequency at which $\left|q_{ii}\left(j\omega_{1_{ii}}\right)\right| = 1$ and $q_{ii}(j\omega)$ is the compensated forward path frequency response of the diagonal element, $q_{ii}(s)$.

In the next stage of the method by Ho *et al* the decentralised controllers are designed. Specifying the multivariable gain margin, *GM'*, and phase margin, $\phi_{PM}^{'}$, for each loop, the PID controllers for each loop are then designed from the gain margin, *GM*, and phase margin, ϕ_{PM}, obtained from equations (4.44) and (4.46) employing the single-input single-output gain and phase margin PID controller design equations given in Ho *et al* (1995).

4.4.1 Closed Loop Identification of Multivariable Processes.

In the above, a discussion of the methods used to extend the application of the relay experiment to the non-parametric identification of particular points on the frequency response of the transfer function matrix of multivariable processes was undertaken along with a discussion of decentralised and full matrix PID controller design from the data obtained by those identifications. In the sequel an extension of the Phase-locked Loop method of non-parametric system identification, to the identification of multivariable processes connected in closed loop shall be carried out. It shall be shown that there is no restriction as to the controller configuration in that, decentralised or full matrix controllers may be employed. In Chapter 2, the exact gain and phase margin controller design method of Fung *et al* (1998) was discussed. An extension of the exact gain and phase margin controller design method to multivariable systems shall be developed.

Consider the multivariable system shown in Figure 4.28.

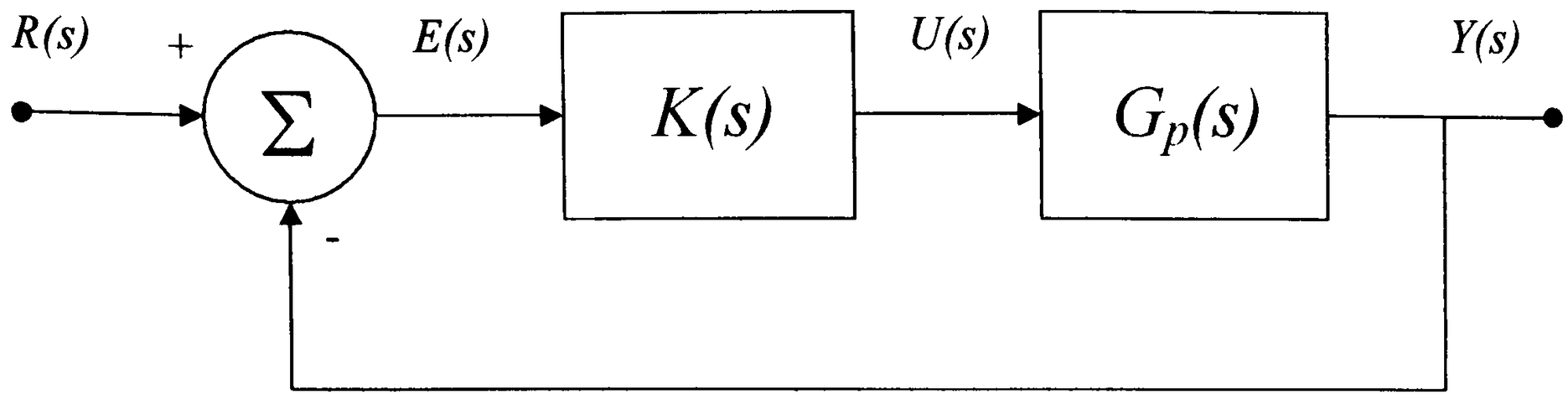


Figure 4.28: Multivariable Control System

In the sequel it will be assumed that the process transfer function matrix is square and is dimension m x m,

From Figure 4.28 it can be seen that

$$Y(s) = G_p(s)U(s) \tag{4.47}$$

$$U(s) = K(s)E(s) \tag{4.48}$$

$$E(s) = R(s) - Y(s) \tag{4.49}$$

From equations (4.47) to (4.49) it follows that

$$Y(s) = \left(I + G_p(s)K(s)\right)^{-1} G_p(s)K(s)R(s) \tag{4.50}$$

From equations (4.47) to (4.49) it can also be shown that

$$U(s) = \left(I + K(s)G_p(s)\right)^{-1} K(s)R(s) \tag{4.51}$$

In equation (4.50) let

$$T(s) = \left(I + G_p(s)K(s)\right)^{-1} G_p(s)K(s) \tag{4.52}$$

and in equation (4.51) let

$$C(s) = \left(I + K(s)G_p(s)\right)^{-1} K(s) \tag{4.53}$$

Substitution of equations (4.52) and (4.53) into equations (4.50) and (4.51) respectively, gives

$$Y(s) = T(s)R(s) \tag{4.54}$$

$$U(s) = C(s)R(s) \tag{4.55}$$

hence using equations (4.54) and (4.55) it follows that

$$T^{-1}(s)Y(s) = C^{-1}(s)U(s)$$

thus

$$Y(s) = T(s)C^{-1}(s)U(s) \tag{4.56}$$

but from equation (4.47)

$$Y(s) = G_p(s)U(s) \tag{4.57}$$

Thus from equations (4.56) and (4.57) it follows that

$$G_p(s) = T(s)Q^{-1}(s) \tag{4.58}$$

Proof of Equation (4.58)

$$\begin{aligned} T(s)Q^{-1}(s) &= (I + G(s)K(s))^{-1} G(s)K(s)\{(I + K(s)G(s))^{-1} K(s)\}^{-1} \\ &= (I + G_p(s)K(s))^{-1} G_p(s)K(s)K^{-1}(s)(I + K(s)G_p(s)) \\ &= (I + G_p(s)K(s))^{-1} G_p(s)(I + K(s)G_p(s)) \end{aligned} \tag{4.59}$$

Using Householders matrix identity

$$(B + CDE)^{-1} = B^{-1} - B^{-1}CD(D + DEB^{-1}CD)^{-1} DEB^{-1}$$

Comparing $(B + CDE)^{-1}$ to $(I + G_p(s)K(s))^{-1}$ gives $B = I$, $C = G_p(s)$, $D = I$ and $E = K(s)$, where I is the identity matrix.

Hence $$(I + G_p(s)K(s))^{-1} = I - G_p(s)(I + K(s)G_p(s))^{-1} K(s) \tag{4.60}$$

Substituting equation (4.60) into equation (4.59) gives

$$T(s)C^{-1}(s) = G_p(s)(I + K(s)G_p(s)) - G_p(s)(I + K(s)G_p(s))^{-1} K(s)G_p(s)(I + K(s)G_p(s))$$

$$\begin{aligned} &= G_p(s)\{I + K(s)G_p(s) - (I + K(s)G_p(s))^{-1} K(s)G_p(s)(I + K(s)G_p(s))\} \\ &= G_p(s)\{I + K(s)G_p(s) - (I + K(s)G_p(s))^{-1} K(s)G_p(s) \\ &\qquad - (I + K(s)G_p(s))K(s)G_p(s)K(s)G_p(s)\} \\ &= G_p(s)\{I + K(s)G_p(s) - \{(I + K(s)G_p(s))^{-1} \\ &\qquad + (I + K(s)G_p(s))^{-1} K(s)G_p(s)\}K(s)G_p(s)\} \\ &= G_p(s)\{I + K(s)G_p(s) - (I + K(s)G_p(s))^{-1}(I + K(s)G_p(s))K(s)G_p(s)\} \\ &= G_p(s)\{I + K(s)G_p(s) - K(s)G_p(s)\} \\ &= G_p(s) \end{aligned}$$

Thus it can be seen from equation (4.58) that if the matrices *T(s)* and *C(s)* can be identified then it is possible to construct an estimate of the process transfer function matrix, $G_p(s)$. Recall equation (4.54) in expanded form

$$\begin{bmatrix} y_1(s) \\ y_2(s) \\ \vdots \\ \vdots \\ y_m(s) \end{bmatrix} = \begin{bmatrix} t_{11}(s) & t_{12}(s) & \cdots & t_{1m}(s) \\ t_{21}(s) & t_{22}(s) & \cdots & t_{2m}(s) \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ t_{m1}(s) & t_{m2}(s) & \cdots & t_{mm}(s) \end{bmatrix} \begin{bmatrix} r_1(s) \\ r_2(s) \\ \vdots \\ \vdots \\ r_m(s) \end{bmatrix} \tag{4.61}$$

It is not possible to determine *T(s)* directly from equation (4.61). However if *m*-identifications are carried out such that for each identification *i*, $i = 1, \ldots, m$ the excitation applied to the multivariable closed loop system reference input is such that

$$r_j(s) = \begin{cases} r_i(s), j = i \\ 0, j \neq i \end{cases}$$

Then, after *m*-identifications the following equation can be constructed

$$\begin{bmatrix} t_{11}(s) & t_{12}(s) & \cdots & t_{1m}(s) \\ t_{21}(s) & t_{22}(s) & \cdots & t_{2m}(s) \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ t_{m1}(s) & t_{m2}(s) & \cdots & t_{mm}(s) \end{bmatrix} = \begin{bmatrix} y_1^1(s) & y_1^2(s) & \cdots & y_1^m(s) \\ y_2^1(s) & y_2^2(s) & \cdots & y_2^m(s) \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ y_m^1(s) & y_m^2(s) & \cdots & y_m^m(s) \end{bmatrix} \times$$

$$\begin{bmatrix} r_1(s) & 0 & 0 & 0 \\ 0 & r_2(s) & \cdots & 0 \\ 0 & 0 & \ddots & \vdots \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & 0 & r_m(s) \end{bmatrix}^{-1} \tag{4.62}$$

where the superscript refers to the identification test number. Thus from equation (4.62) it can be seen that the matrix, *T(s)*, will be given by

$$T(s)=\begin{bmatrix} \left(\frac{y_1^1(s)}{r_1(s)}\right) & \left(\frac{y_1^2(s)}{r_2(s)}\right) & \cdots & \left(\frac{y_1^m(s)}{r_m(s)}\right) \\ \left(\frac{y_2^1(s)}{r_1(s)}\right) & \left(\frac{y_2^2(s)}{r_2(s)}\right) & \cdots & \left(\frac{y_2^m(s)}{r_m(s)}\right) \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \left(\frac{y_m^1(s)}{r_1(s)}\right) & \left(\frac{y_m^2(s)}{r_2(s)}\right) & \cdots & \left(\frac{y_m^m(s)}{r_m(s)}\right) \end{bmatrix} \tag{4.63}$$

Using a similar reasoning and a similar excitation strategy it can be shown that the matrix $C(s)$ can be obtained from

$$C(s)=\begin{bmatrix} \left(\frac{u_1^1(s)}{r_1(s)}\right) & \left(\frac{u_1^2(s)}{r_2(s)}\right) & \cdots & \left(\frac{u_1^m(s)}{r_m(s)}\right) \\ \left(\frac{u_2^1(s)}{r_1(s)}\right) & \left(\frac{u_2^2(s)}{r_2(s)}\right) & \cdots & \left(\frac{u_2^m(s)}{r_m(s)}\right) \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \left(\frac{u_m^1(s)}{r_1(s)}\right) & \left(\frac{u_m^2(s)}{r_2(s)}\right) & \cdots & \left(\frac{u_m^m(s)}{r_m(s)}\right) \end{bmatrix} \tag{4.64}$$

By carrying out specific identifications on the closed loop system equations (4.58), (4.63) and (4.64) show that it is possible to construct an estimate of the frequency response of the multivariable process transfer function, $G_p(s)$. It should be noted that there is no requirement on having any knowledge of the controller structure or parameters to perform the closed loop identification.

4.4.2 The Phase-Locked Loop Method Applied to Closed Loop Multivariable Process Identification.

In the above, it is shown that by carrying out specific identifications it is possible to produce an estimate of the frequency response of a multivariable process transfer function matrix. In Figure 4.29 an unknown multivariable process, $G_p(s)$, is shown in a feedback configuration with an unknown controller, $K(s)$. It is assumed, for the purpose of description that the process is square and of dimension 2 x 2. The extension of the method to higher dimensions will be seen to be readily accomplished. From Figure 4.29 it can be seen that four Phase-Locked Loop identifiers are connected to the closed loop system. The excitation signal is shown connected to the system via a switch. The purpose of the switch is to connect the excitation signals to the particular reference inputs depending on the test number. If the test number is one, then the following reference inputs will be applied

$$R(s)=\begin{bmatrix} r_1(s) \\ 0 \end{bmatrix}$$

similarly if it is test number two then

$$R(s)=\begin{bmatrix} 0 \\ r_2(s) \end{bmatrix}$$

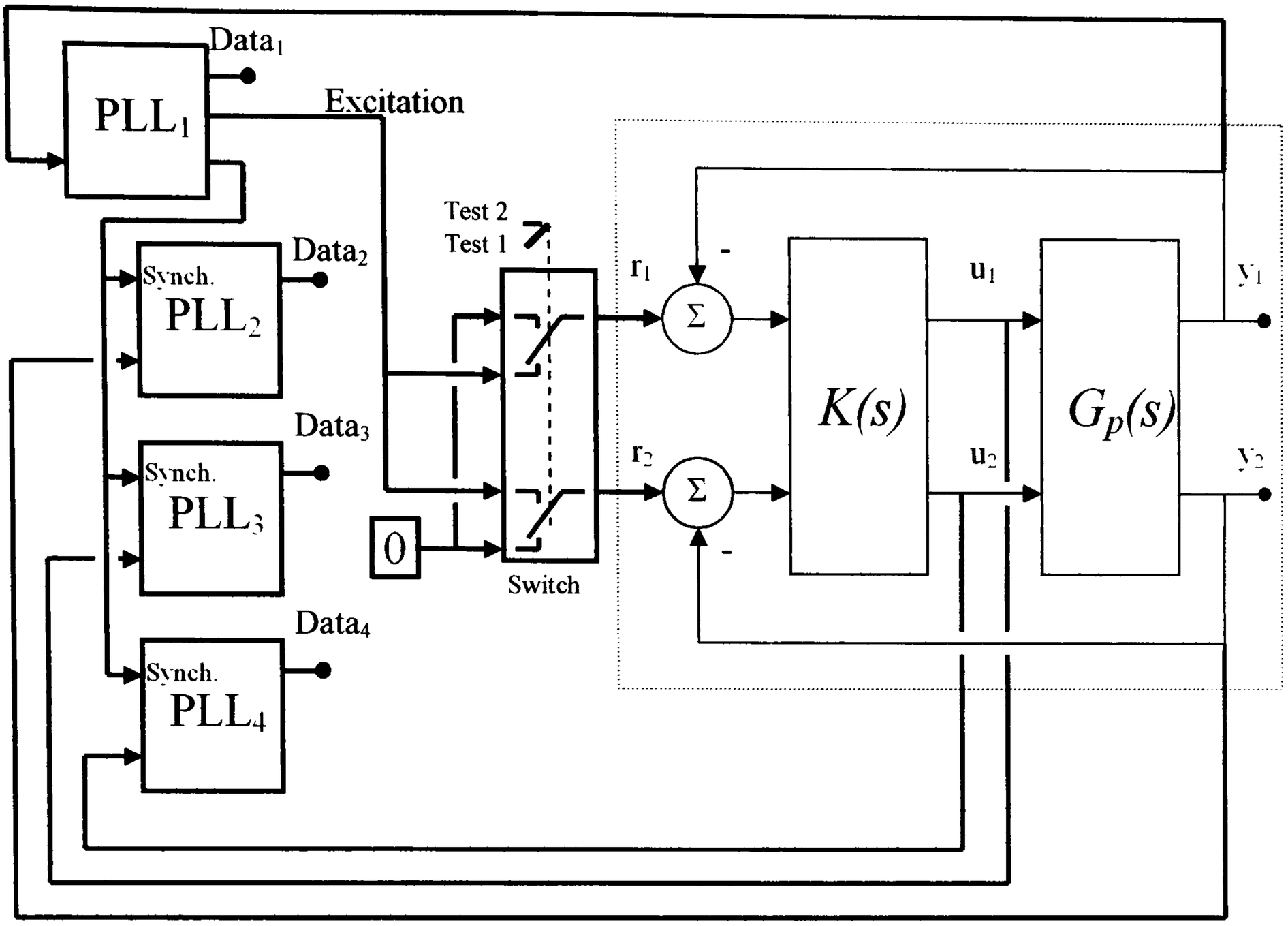


Figure 4.29: PLL Identification of an Unknown Multivariable Process in Closed Loop with an Unknown Controller.

Only one excitation signal is generated and applied to the closed loop system by the Phase-Locked Loop (PLL) identifiers. From Figure 4.29 it can be seen that there is a connection between the PLL identifiers shown as, *Synch*. The function of this connection is to transmit the excitation signal between all of the PLL identifiers. Hence by this construction the PLL identifiers are synchronised to a single excitation signal and thus all of the required identifications are carried out in parallel. At each of the identifications a column of the matrices *T(s)* and *C(s)* is obtained. Then, after each identification, the position of the switch is changed over and a new identification is initiated for the remainder of the *T(s)* and *C(s)* columns. When all of the elements of the *T(s)* and *C(s)* matrices have been identified an estimate of the frequency response of the multivariable process transfer function matrix is constructed. This process is repeated until the PLL identifiers have obtained a converged estimate of a particular point on the frequency response of a chosen

process transfer function element. It should be noted that all of the process transfer function elements are identified however one must choose a transfer function element for which a particular frequency response curve point is sought. The identification process is captured in Algorithm 4.4.

Algorithm 4.4: Multivariable Process Closed Loop Identification.

Step 1: Initialisation

Choose the reference phase angle.

Choose the multivariable process transfer function element that the reference phase angle refers to.

Choose the initial excitation frequency ω_k, excitation magnitude and stopping tolerance values.

Set the column index counter, *n*, to 1

Step 2: Excitation and Identification

Apply the identification excitation to reference input, *n*, with all other reference excitations set equal to zero.

Use the Phase-Locked Loop identifiers to identify between the reference input, n, and the multivariable process outputs to obtain the vector

$$T_{.n}(j\omega_k)=\left[\frac{y_1(j\omega_k)}{r_n(j\omega_k)} \quad \frac{y_2(j\omega_k)}{r_n(j\omega_k)} \quad \cdots \quad \frac{y_m(j\omega_k)}{r_n(j\omega_k)}\right]^T$$

Use the Phase-Locked Loop identifiers to identify between the reference input, *n*, and the inputs to the multivariable process to obtain the vector

$$C_{.n}(j\omega_k)=\left[\frac{u_1(j\omega_k)}{r_n(j\omega_k)} \quad \frac{u_2(j\omega_k)}{r_n(j\omega_k)} \quad \cdots \quad \frac{u_m(j\omega_k)}{r_n(j\omega_k)}\right]^T$$

Step 3: Increment

$n:=n+1$

If $n \neq m+1$ go to Step 2

else go to Step 4

Step 4: Process Transfer Function Matrix Evaluation

Form the matrix $T(j\omega_k)$ using the following

$$T(j\omega_k)=[T_{.1}(j\omega_k) \quad T_{.2}(j\omega_k) \quad \cdots \quad T_{.m}(j\omega_k)]$$

Form the matrix $C(j\omega_k)$ using the following

$$C(j\omega_k)=[C_{.1}(j\omega_k) \quad C_{.2}(j\omega_k) \quad \cdots \quad C_{.m}(j\omega_k)]$$

Obtain the matrix, $G(j\omega_k)$, from $G(j\omega_k)=T(j\omega_k)C(j\omega_k)^{-1}$

Step 5: Test for Convergence to the Reference Value

If $|$ *reference* $-arg(g_{uv}(j\omega_k))$ $|$ < Tol then Stop

The Phase-Locked Loop identifier system calculates the next excitation frequency, ω_k.

Reset n to 1

Go to Step 2 ●

Case Study 4.3: Identification of a Multivariable Process

In this case study a multivariable process connected in closed loop with a full matrix controller will be identified. The multivariable process (Wood and Berry, 1973) is given by the following transfer function matrix

$$G_p(s)=\begin{bmatrix} \dfrac{12.8e^{-s}}{(1+16.7s)} & \dfrac{-18.9e^{-3s}}{(1+21s)} \\ \dfrac{6.6e^{-7s}}{(1+10.9s)} & \dfrac{-19.4e^{-3s}}{(1+14.4s))} \end{bmatrix}$$

The controller transfer function matrix is given by (Wang *et al*, 1997b)

$$K(s)=\begin{bmatrix} 0.184\left(1+\dfrac{1}{3.92s}\right) & -0.0102\left(1+\dfrac{1}{0.445s}-0.804s\right) \\ -0.0674\left(1-\dfrac{1}{4.23s}+0.796s\right) & -0.066\left(1+\dfrac{1}{4.25s}\right) \end{bmatrix}$$

In carrying out the identification using the Phase-Locked Loop identification method the initial integrator gain was set to be 0.2, the initial identification frequency was chosen as 0.1 (rad.s^{-1}). A reference phase angle of -π (rad) for process transfer function element (1,1) was chosen. The choice of this element was made since the frequency required to obtain a phase angle of -π (rad) is the largest and hence all of the other elements will have phase angles that are greater than -π (rad) at the end of

the identification. The results of the identification are shown graphically in Figures 4.30 to 4.33 inclusive where the identification data is plotted against the corresponding Nyquist curve of the process transfer function element.

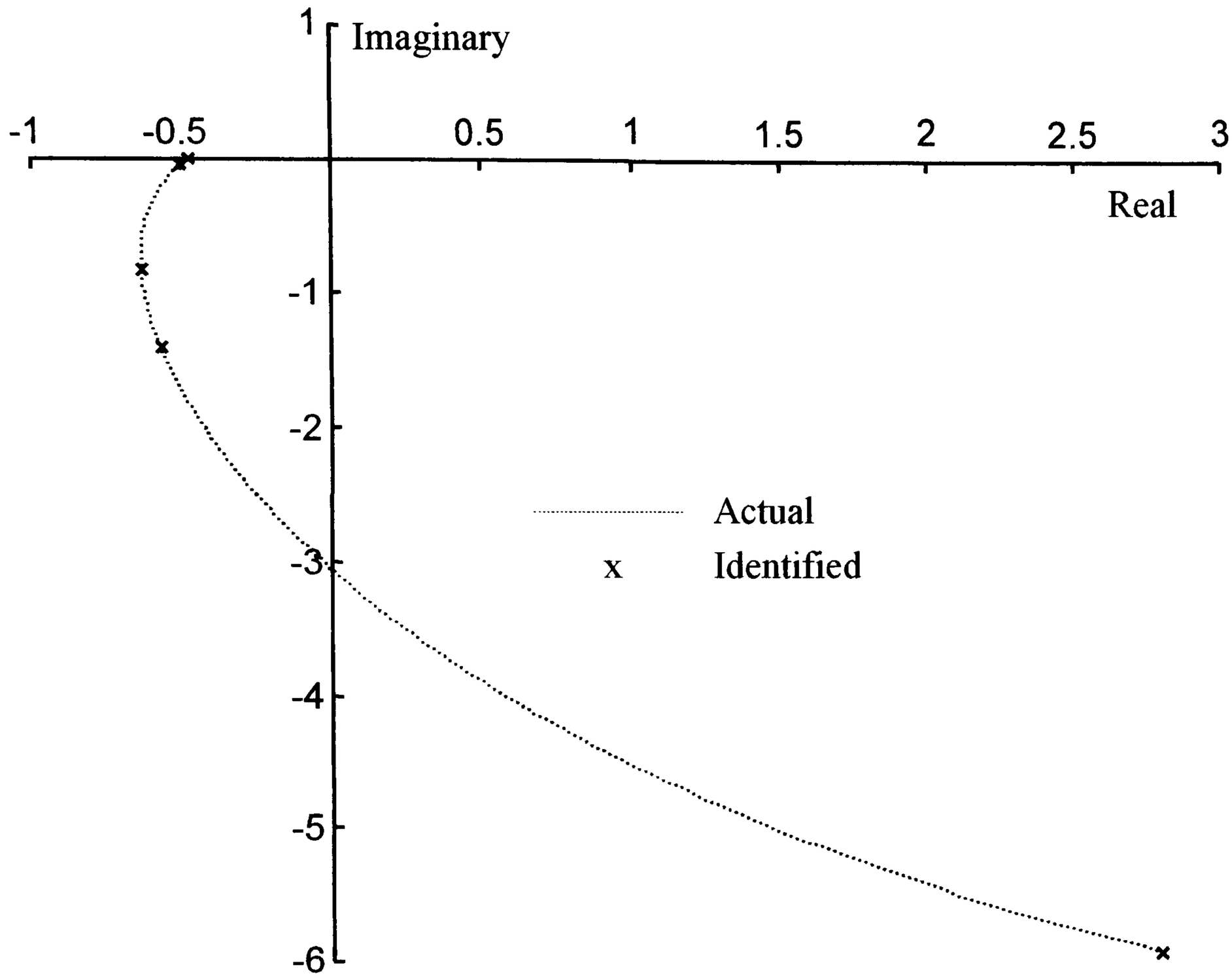


Figure 4.30: Nyquist Diagram for Process Transfer Function Element $g_{p_{11}}(s)$

It can be seen from Figure 4.30 that the identified data is in good agreement with the theoretical frequency response. Similarly Table 4.9 allows a numerical comparison to be carried out between the identified data and theoretical data calculated on the basis that the frequency at which the identified data was obtained is accurate data for the identification. The identification took approximately 30 minutes. This time is not considered to be excessive since the identification was carried out in closed loop and the intermediate points obtained from the identification are good estimates of the frequency response of the process transfer function.

Table 4.9: Identification Data for Transfer Function Element $g_{p_{11}}(s)$

Magnitude		Phase Angle (rad)		Frequency (rad.s^{-1})	Time (s)
Actual	Identified	Actual	Identified		
6.5759	6.5758	5.1519	5.1520	0.1	332
1.5158	1.5158	-1.9542	-1.9537	0.5021	690
1.0328	1.0328	-2.2297	-2.2288	0.7397	976
0.5014	0.5015	-3.0592	-3.0575	1.5276	1265
0.4765	0.4761	-3.1410	-3.1394	1.6074	1508
0.4759	0.4760	-3.1431	-3.1415	1.6095	1797

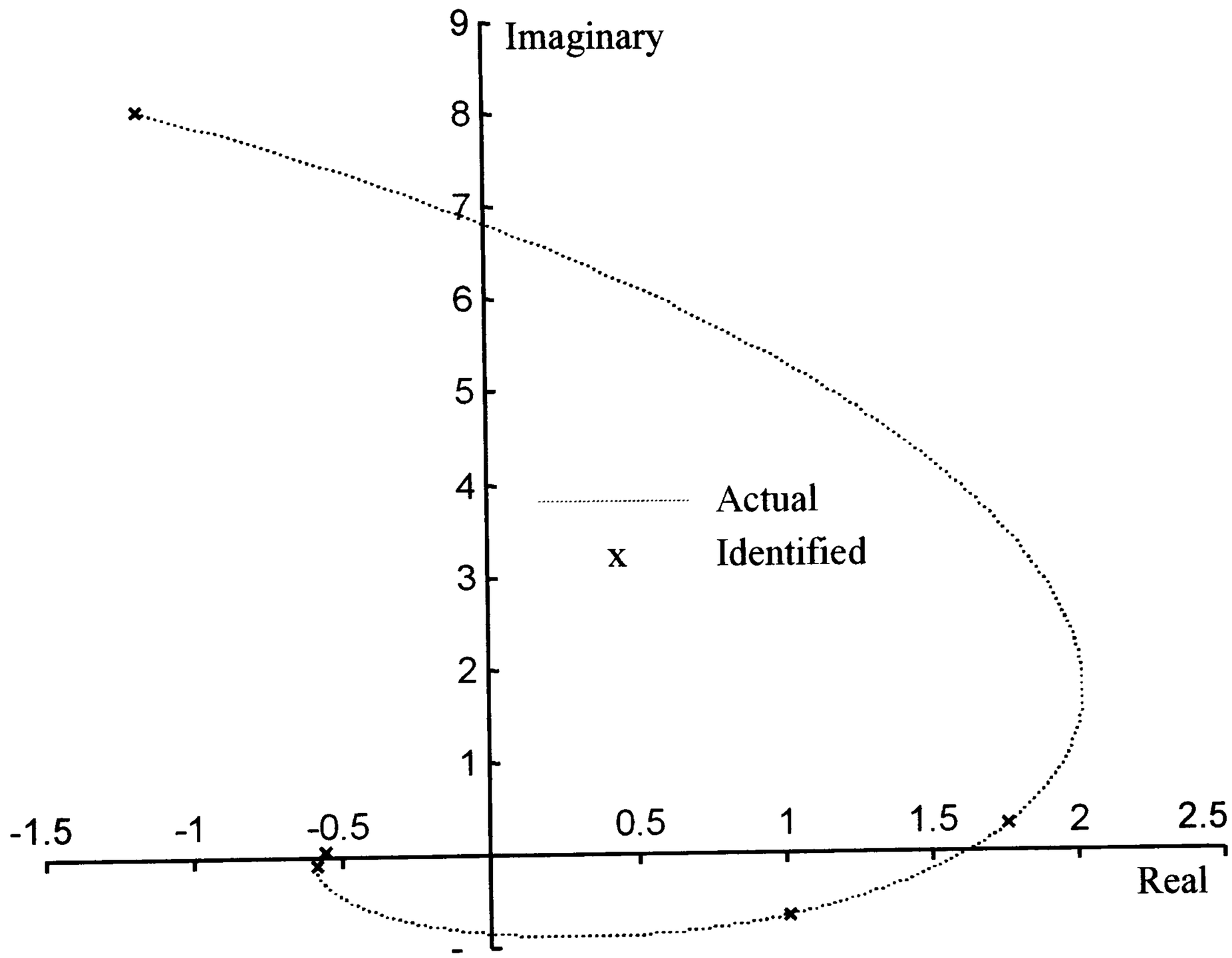


Figure 4.31: Nyquist Diagram for Process Transfer Function Element $g_{p_{12}}(s)$

Good agreement is obtained between the identified data and the theoretical frequency response curve of element (1,2) of the process transfer function matrix. Table 4.10 shows the numerical results from the identification.

Table 4.10: Identification Data for Transfer Function Element $g_{p_{12}}(s)$

Magnitude		Phase Angle (rad)		Frequency (rad.s^{-1})	Time (s)
Actual	Identified	Actual	Identified		
8.1257	8.1257	1.7152	1.7153	0.1	332
1.7845	1.7845	0.1591	0.1596	0.5021	690
1.2142	1.2142	-0.5839	-0.5831	0.7397	976
0.5889	0.5890	-2.9808	-2.9792	1.5276	1265
0.5597	0.5598	-3.2218	-3.2206	1.6074	1508
0.5589	0.5589	-3.2281	-3.2263	1.6095	1797

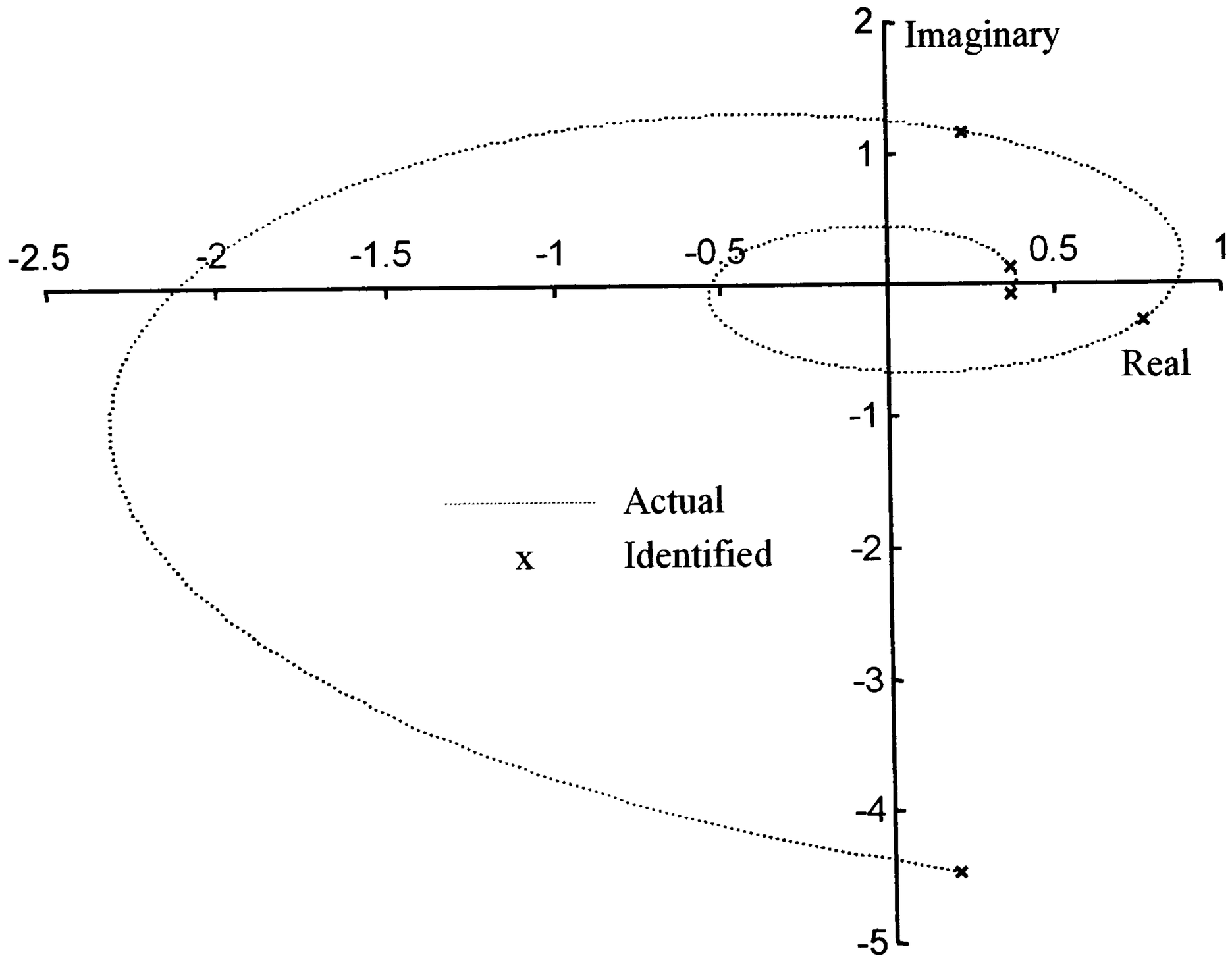


Figure 4.32: Nyquist Diagram for Process Transfer Function Element $g_{p_{21}}(s)$

Table 4.11: Identification Data for Transfer Function Element $g_{p_{21}}(s)$

Magnitude		**Phase Angle (rad)**		**Frequency (rad.s^{-1})**	**Time (s)**
Actual	Identified	Actual	Identified		
4.4618	4.4616	4.7548	4.7548	0.1	332
1.1863	1.1863	-4.9046	-4.9042	0.5021	690
0.8124	0.8124	-0.3419	-3.4126	0.7397	976
0.3957	0.3955	0.3624	0.3638	1.5276	1265
0.3761	0.3760	-0.1992	-0.1978	1.6074	1508
0.3756	0.3755	-0.2140	-0.2122	1.6095	1797

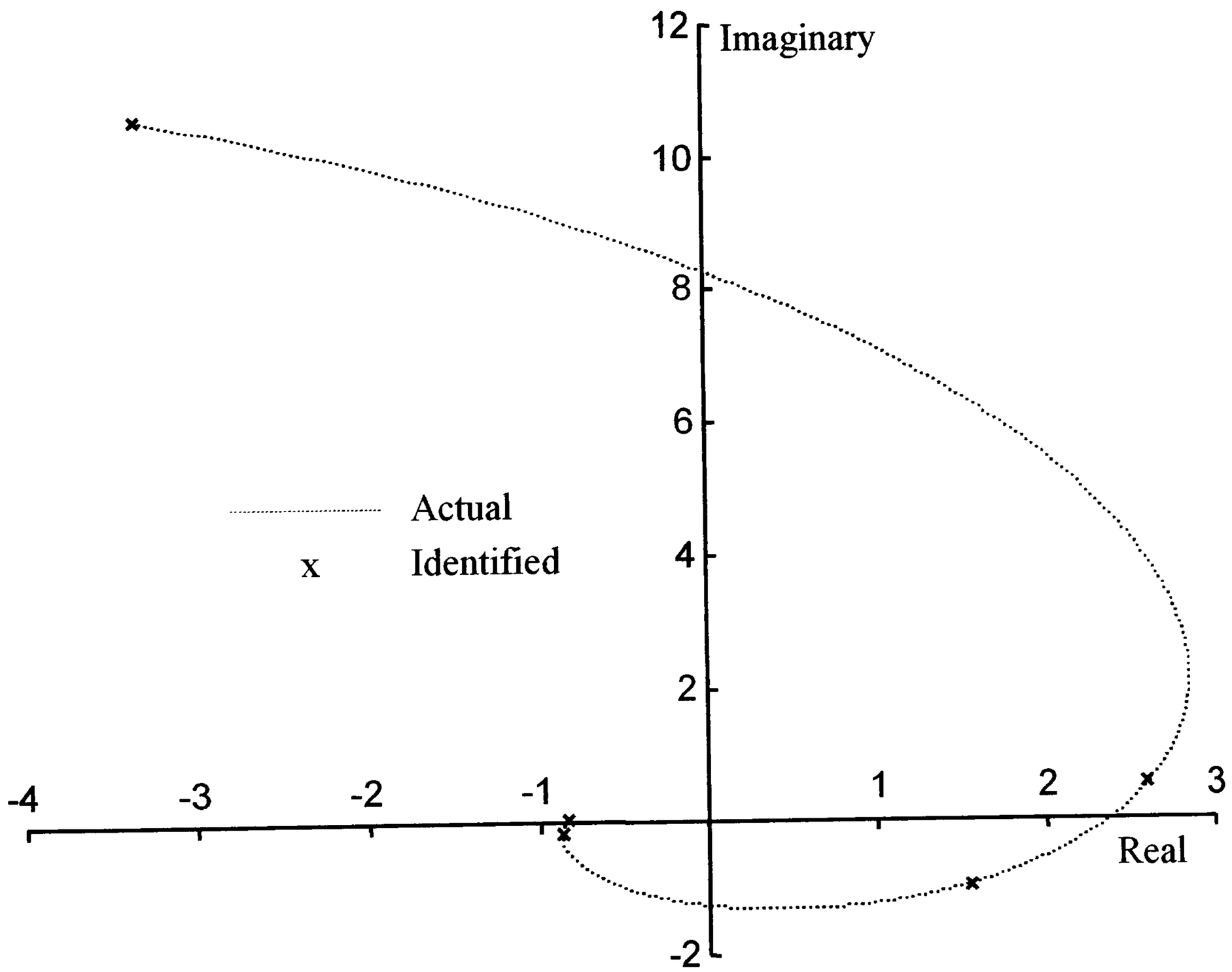


Figure 4.33: Nyquist Diagram for Process Transfer Function Element $g_{p_{22}}(s)$

Table 4.12: Identification Data for Transfer Function Element $g_{p_{21}}(s)$

Magnitude		**Phase Angle (rad)**		**Frequency (rad.s^{-1})**	**Time (s)**
Actual	Identified	Actual	Identified		
11.066	11.065	1.8778	1.8779	0.1	332
2.6580	2.6579	0.2020	0.2025	0.5021	690
1.8134	1.8133	-0.5546	-0.5539	0.7397	976
0.8810	0.8811	-2.9666	-2.9650	1.5276	1265
0.8374	0.8372	-3.2082	-3.2067	1.6074	1508
0.8363	0.8362	-3.2146	-3.2129	1.6095	1797

4.4.3 Gain and Phase Margin Controller Design for Multivariable Processes.

In the above the identification of a multivariable process in closed loop with a full matrix controller was carried out using the Phase-Locked Loop method. In this section the design of a decentralised controller for a multivariable process based on the specification of gain and phase margins shall be undertaken. The multivariable process shall be given by (Wood and Berry, 1973)

$$G_p(s)=\begin{bmatrix} \dfrac{12.8e^{-s}}{(1+16.7s)} & \dfrac{-18.9e^{-3s}}{(1+21s)} \\ \dfrac{6.6e^{-7s}}{(1+10.9s)} & \dfrac{-19.4e^{-3s}}{(1+14.4s))} \end{bmatrix}$$

The initial controller shall be a full matrix controller given by (Wang *et al*, 1997b)

$$K(s)=\begin{bmatrix} 0.184\left(1+\dfrac{1}{3.92s}\right) & -0.0102\left(1+\dfrac{1}{0.445s}-0.804s\right) \\ -0.0674\left(1-\dfrac{1}{4.23s}+0.796s\right) & -0.066\left(1+\dfrac{1}{4.25s}\right) \end{bmatrix}$$

Case Study 4.4

The extension of gain and phase margin design for multivariable processes was reported by Ho *et al* (1995). In this case study it shall be shown that if the

multivariable process can be identified then it is possible to use a modification of the Fung *et al* (1998) method to carry out a gain margin and phase margin design.

The design objective is to provide the parameters for a decentralised PI based control system for the Wood and Berry (1973) process, such that:

i) a gain margin of 5 and a phase margin of 60° will be achieved by controller $k_{11}(s)$, and

ii) a gain margin of 3 and a phase margin of 30° will be achieved by controller $k_{22}(s)$.

The first step will be to identify the multivariable process at a sufficient number of frequency response curve points. In this case thirteen points were chosen in the range -60° to -180° for the process transfer function matrix element $g_{p_{11}}(s)$. The results of the identification are shown in Tables 4.13 to 4.16 inclusive for each transfer function matrix element. To allow a comparison of the accuracy of the identified data with the actual plant data the identification results are shown graphically on Figures 4.34 to 4.37 inclusive.

Table 4.13: Identification Data for Transfer Function Element $g_{p_{11}}(s)$

Magnitude		**Phase Angle (rad)**		**Frequency (rad.s^{-1})**	**Time (s)**
Actual	Identified	Actual	Identified		
7.2669	7.3267	5.2303	5.2365	0.0858	4039
5.7221	5.7777	5.0574	5.0614	0.1184	6466
4.250	4.2962	4.8827	4.8872	0.1681	10092
3.0237	3.0597	4.7075	4.7119	0.2434	12515
2.1547	2.1802	-1.7482	-1.7455	0.3465	13036
1.5790	1.5978	-1.9231	-1.9192	0.4760	13658
1.2110	1.2261	-2.0986	-2.0934	0.6226	14007
0.9696	0.9802	-2.2739	-2.2699	0.7789	14943
0.8019	0.8124	-2.4508	-2.4425	0.9427	15778
0.6827	0.6897	-2.6252	-2.6187	1.1078	16786
0.5926	0.6016	-2.8012	-2.7922	1.2767	17450
0.5238	0.5308	-2.9746	-2.9674	1.4447	17826
0.4684	0.4740	-3.150	-3.1408	1.6158	18219

Table 4.14: Identification Data for Transfer Function Element $g_{p_{12}}(s)$

Magnitude		**Phase Angle (rad)**		**Frequency (rad.s^{-1})**	**Time (s)**
Actual	Identified	Actual	Identified		
9.1716	9.1731	1.8201	1.8207	0.0858	4039
7.0523	7.0552	1.5980	1.5971	0.1184	6466
5.1513	5.1534	1.3425	1.3434	0.1681	10092
3.6288	3.6287	1.0338	1.0354	0.2434	12515
2.5732	2.5774	-5.6153	-5.6117	0.3465	13036
1.8814	1.8806	0.2425	0.2447	0.4760	13658
1.4413	1.4422	-0.2207	-0.2175	0.6226	14007
1.1533	1.1530	-0.7048	-0.7005	0.7789	14943
0.9535	0.9526	-1.2068	-1.2023	0.9427	15778
0.8117	0.8096	-1.7096	-1.7042	1.1078	16786
0.7045	0.7057	-2.2220	-2.2153	1.2767	17450
0.6226	0.6222	-2.7304	-2.7219	1.4447	17826
0.5568	0.5575	-3.2471	-3.2389	1.6158	18219

Table 4.15: Identification Data for Transfer Function Element $g_{p_{21}}(s)$

Magnitude		**Phase Angle (rad)**		**Frequency (rad.s^{-1})**	**Time (s)**
Actual	Identified	Actual	Identified		
4.8204	4.8188	4.9306	4.9313	0.0858	4039
4.0425	4.0441	4.5428	4.5414	0.1184	6466
3.1618	3.1639	4.0352	4.0369	0.1681	10092
2.3278	2.3290	3.3690	3.3710	0.2434	12515
1.6893	1.6922	-3.7375	-3.7347	0.3465	13036
1.2491	1.2492	-4.7124	-4.7098	0.4760	13658
0.9622	0.9624	0.5005	0.5046	0.6226	14007
0.7720	0.7707	-0.6227	-0.6161	0.7789	14943
0.6393	0.6404	-1.7895	-1.7846	0.9427	15778
0.5447	0.5416	-2.9596	-2.9529	1.1078	16786
0.4730	0.4719	-4.1528	-4.1449	1.2767	17450
0.4183	0.4184	-5.3371	-5.3264	1.4447	17826
0.3741	0.3744	-0.2583	-0.2509	1.6158	18219

Table 4.16: Identification Data for Transfer Function Element $g_{p_{22}}(s)$

Magnitude		**Phase Angle (rad)**		**Frequency (rad.s^{-1})**	**Time (s)**
Actual	Identified	Actual	Identified		
12.2051	12.205	1.9938	1.9946	0.0858	4039
9.8149	9.8102	1.7460	1.7455	0.1184	6466
7.4072	7.4138	1.4583	1.4591	0.1681	10092
5.3226	5.3237	1.1185	1.120	0.2434	12515
3.8123	3.8207	-5.5541	-5.5510	0.3465	13036
2.8006	2.8005	0.2877	0.2896	0.4760	13658
2.1505	2.1493	-0.1859	-0.1827	0.6226	14007
1.7228	1.7199	-0.6769	-0.6734	0.7789	14943
1.4252	1.4267	-1.1838	-1.1789	0.9427	15778
1.2137	1.2119	-1.69	-1.6854	1.1078	16786
1.0537	1.0542	-2.2050	-2.1988	1.2767	17450
0.9314	0.9304	-2.7153	-2.7073	1.4447	17826
0.8330	0.8342	-3.2337	-3.2251	1.6158	18219

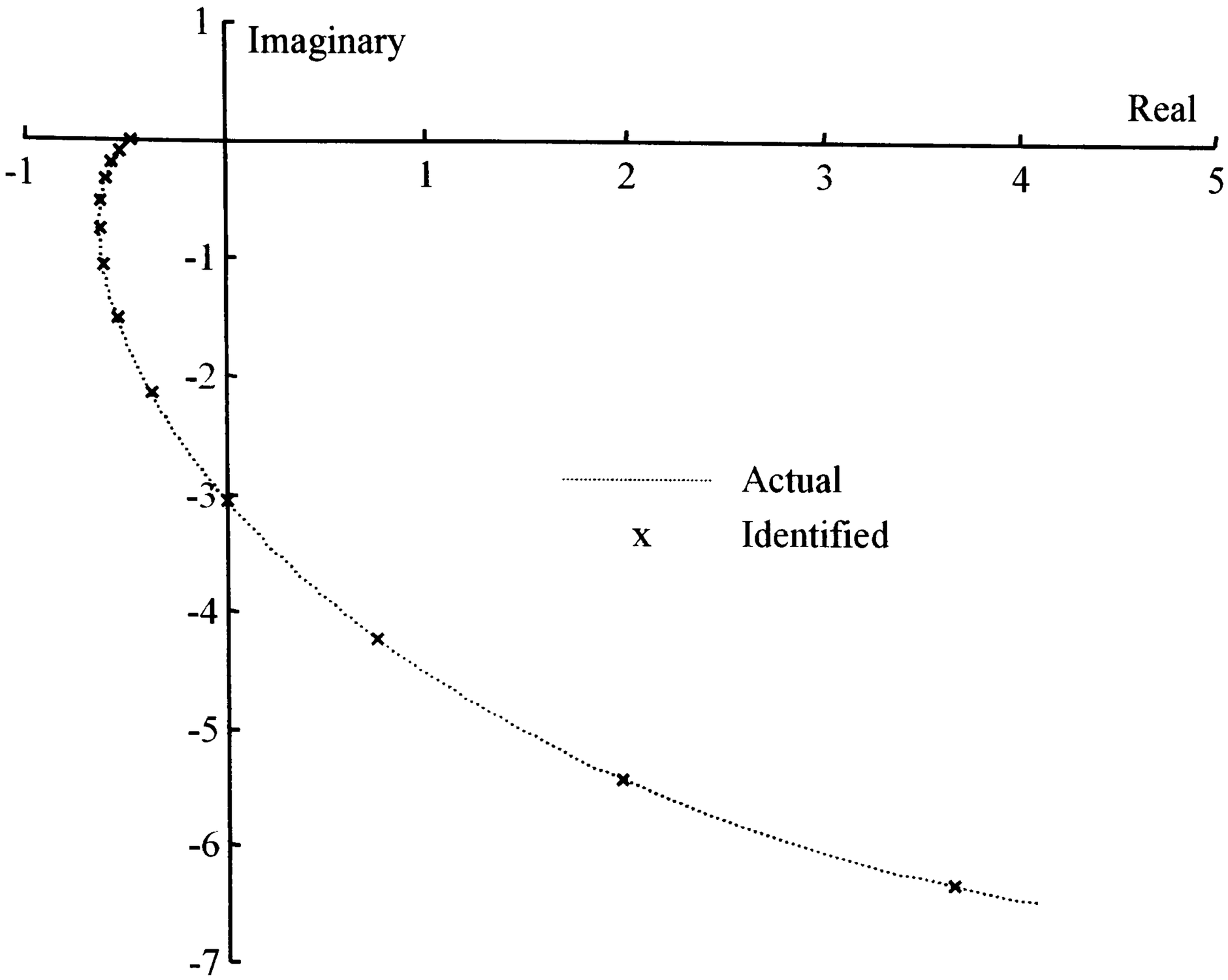


Figure 4.34: Frequency Response for Process Transfer Function Element $g_{p_{11}}(s)$

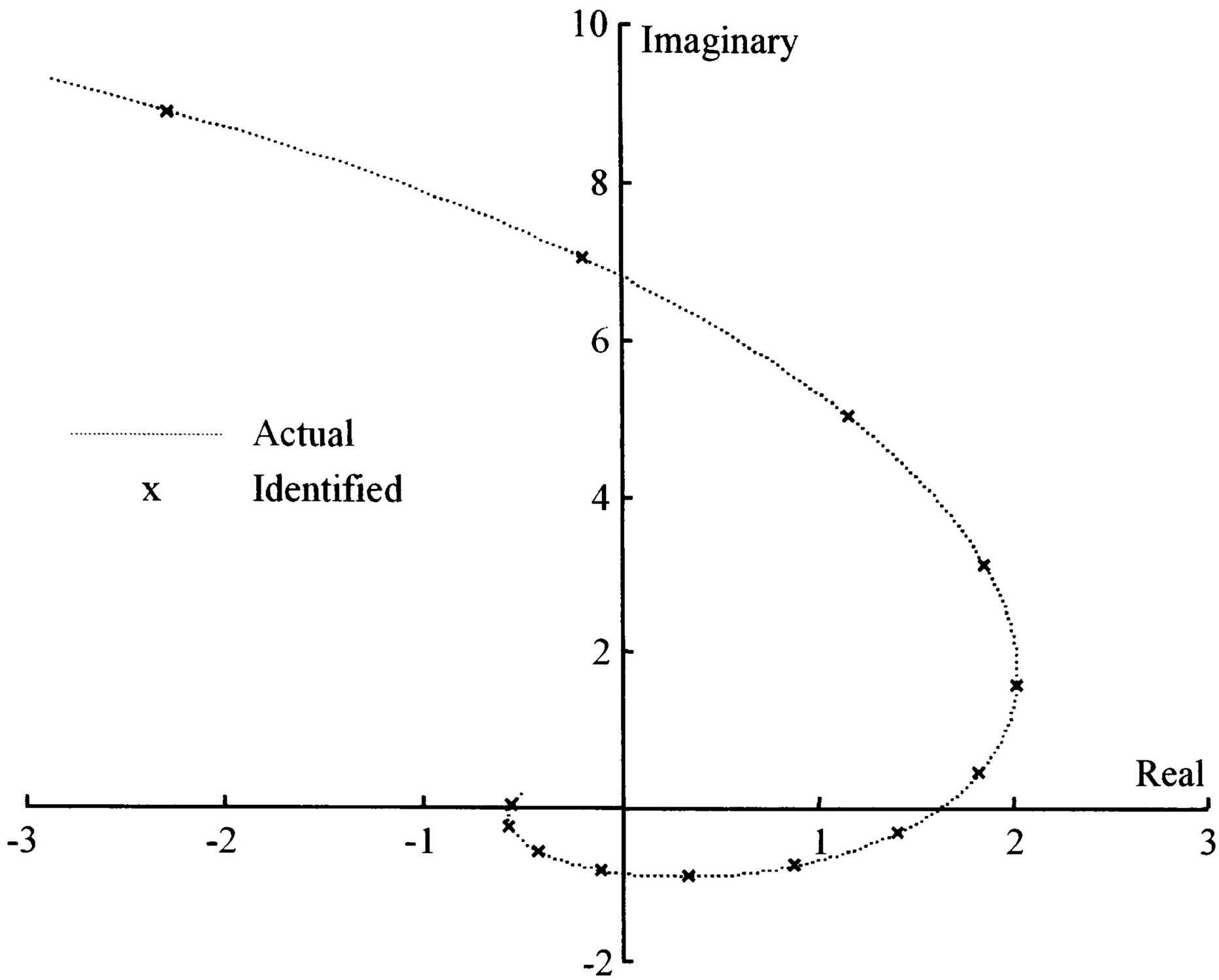


Figure 4.35: Frequency Response for Process Transfer Function Element $g_{p_{12}}(s)$

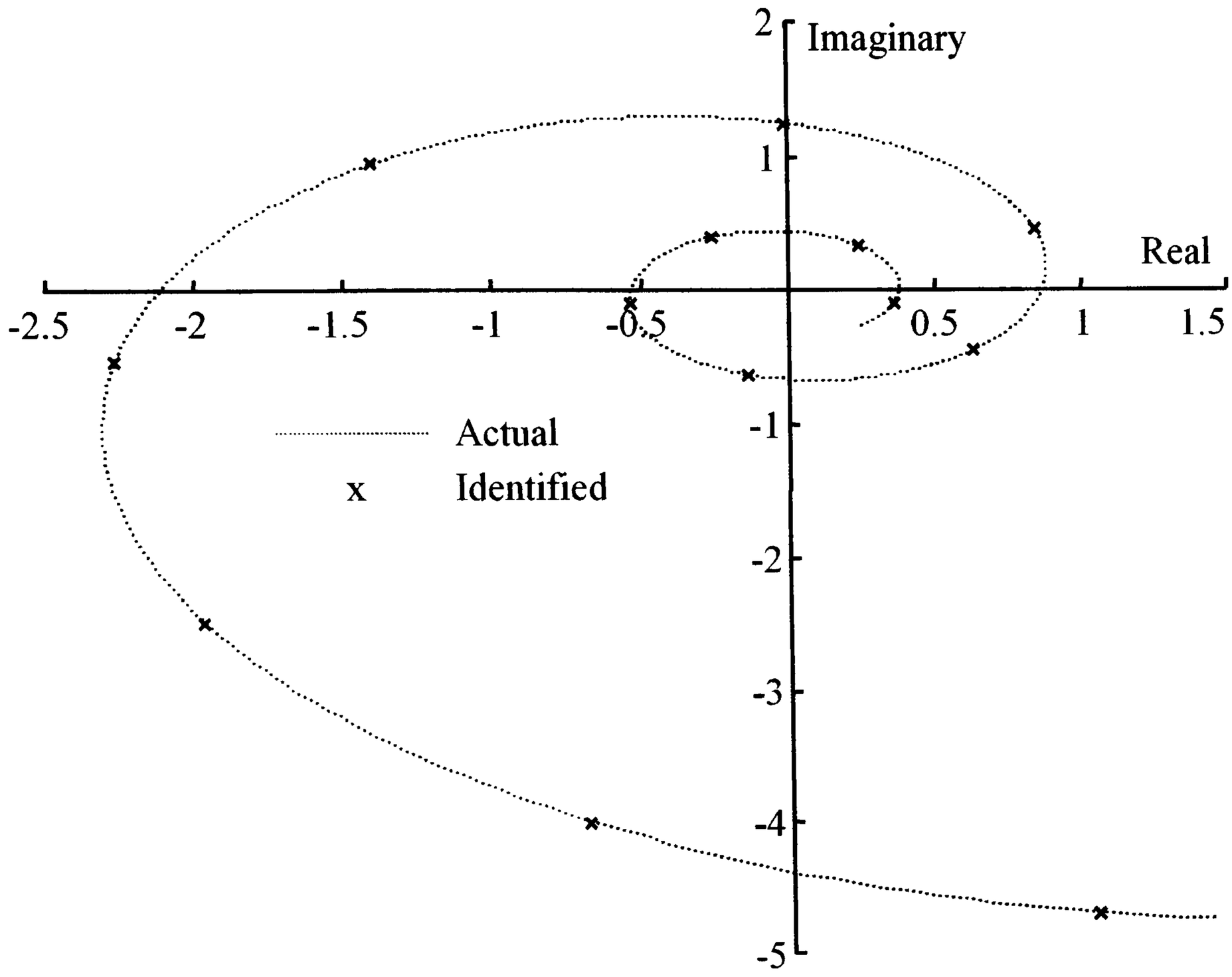


Figure 4.36: Frequency Response for Process Transfer Function Element $g_{p_{21}}(s)$

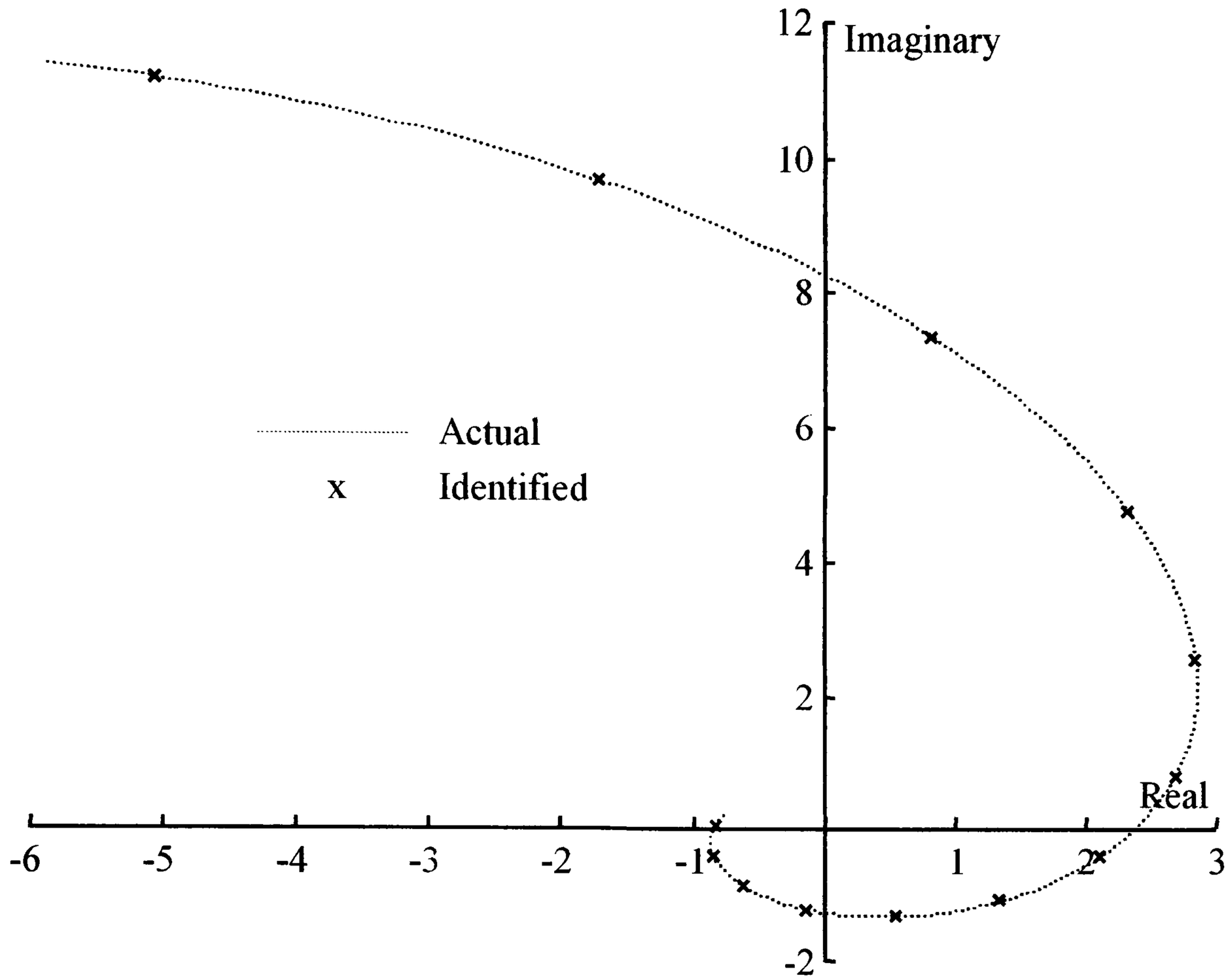


Figure 4.37: Frequency Response for Process Transfer Function Element $g_{p_{22}}(s)$

With the identification complete the controller design can now be undertaken. Consider the system shown in Figure 4.38. From Figure 4.38 it can be seen that

$$U_2(s) = K_{22}(s)(R_2(s) - Y_2(s)) \tag{4.65}$$

and

$$Y_2(s) = g_{22}(s)U_2(s) - g_{21}(s)U_1(s) \tag{4.66}$$

thus

$$(1 + g_{22}(s)K_{22}(s))U_2(s) = K_{22}(s)R_2(s) - g_{21}(s)K_{22}(s)U_1(s)$$

hence,

$$U_2(s) = \frac{K_{22}(s)}{(1 + g_{22}(s)K_{22}(s))}R_2(s) - \frac{g_{21}(s)K_{22}(s)}{(1 + g_{22}(s)K_{22}(s))}U_1(s) \tag{4.67}$$

Now

$$Y_1(s) = g_{11}(s)U_1(s) + g_{12}(s)U_2(s) \tag{4.68}$$

Substitution of equation (4.67) into equation (4.68) gives

$$Y_1(s) = g_{11}(s)U_1(s) + \frac{g_{12}(s)K_{22}(s)}{(1 + g_{22}(s)K_{22}(s))}R_2(s) - \frac{g_{12}(s)g_{21}(s)K_{22}(s)}{(1 + g_{22}(s)K_{22}(s))}U_1(s)$$

or

$$Y_1(s) = \left(g_{11}(s) - \frac{g_{12}(s)g_{21}(s)K_{22}(s)}{(1 + g_{22}(s)K_{22}(s))}\right)U_1(s) + \frac{g_{12}(s)K_{22}(s)}{(1 + g_{22}(s)K_{22}(s))}R_2(s)$$

(4.69)

Similarly it can be shown that

$$Y_2(s)=\left(g_{22}(s)-\frac{g_{12}(s)g_{21}(s)K_{11}(s)}{(1+g_{11}(s)K_{11}(s))}\right)U_2(s)+\frac{g_{21}(s)K_{11}(s)}{(1+g_{11}(s)K_{11}(s))}R_1(s)$$

(4.70)

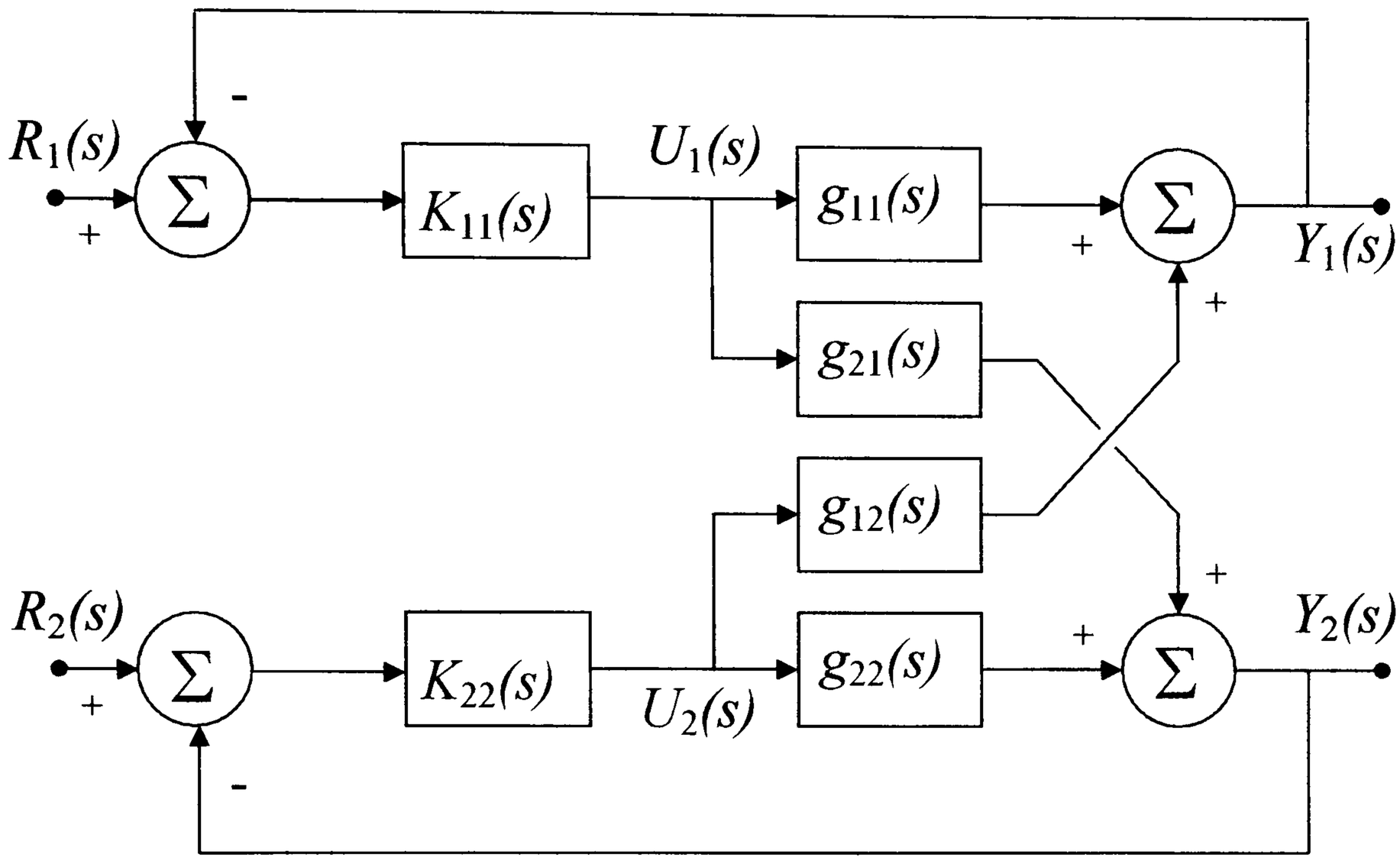


Figure 4.38: Decentralised 2-Input 2-Output Control System

Equation (4.69) can be considered to represent the system between $U_1(s)$ and $Y_1(s)$ in Figure 4.38 as the sum of the control signal $U_1(s)$ filtered by the transfer function

$$G_1(s)=g_{11}(s)-\frac{g_{12}(s)g_{21}(s)}{\left(\frac{1}{K_{22}(s)}+g_{22}(s)\right)} \qquad (4.71)$$

and a disturbance term. The disturbance term being given by

$$d_1(s)=\frac{g_{12}(s)}{\left(\frac{1}{K_{22}(s)}+g_{22}(s)\right)}R_2(s) \qquad (4.72)$$

Similarly the system between $U_2(s)$ and $Y_2(s)$ in Figure 4.38 can be considered as the sum of the control signal $U_2(s)$ filtered by the transfer function

$$G_2(s) = g_{22}(s) - \frac{g_{12}(s)g_{21}(s)}{\left(\frac{1}{K_{11}(s)} + g_{11}(s)\right)} \tag{4.73}$$

and a disturbance term

$$d_2(s) = \frac{g_{21}(s)}{\left(\frac{1}{K_{11}(s)} + g_{11}(s)\right)} R_1(s) \tag{4.74}$$

If the controllers $K_{11}(s)$ and $K_{22}(s)$ are known and there is identification data that allows the multivariable process transfer function elements to be adequately represented over a given frequency interval then it will be possible to carry out the required design using equations (4.71) and (4.73) as the target processes.

To implement the Fung *et al* (1998) method, or as it is more generally known, the Exact Gain and Phase Margin method on a 2-input 2-output system the following new algorithm was utilised.

Algorithm 4.4: Exact Gain and Phase Margin Design for a 2-input 2-Output Process.

Step 1: Initialisation.

The existing controller parameters are retained or use Exact Gain and Phase Margin method with $K_{11}(s) = K_{22}(s) = 1$.

Set the minimum and maximum values for the frequency range of interest.

Set loop one and loop two gain and phase margin desired values GM_1 ϕ_{PM_1} and GM_2 ϕ_{PM_2}.

n=0

Step 2: Controller $K_{11}(s)$ Tuning Parameters.

Form two complex functions such that

$$f_1(\omega) = -\frac{\cos\phi_p(\omega)}{GM_1\left|G_p(j\omega)\right|} - j\frac{\omega\sin\phi_p(\omega)}{GM_1\left|G_p(j\omega)\right|}$$

$$-\frac{\pi}{2} > \arg\left(G_p(j\omega)\right) > -\pi$$

$$f_2(\omega) = -\frac{\cos(\phi_{PM_1} - \phi_p(\omega))}{|G_p(j\omega)|} + j\frac{\omega \sin(\phi_{PM_1} - \phi_p(\omega))}{|G_p(j\omega)|}$$

$$-\frac{\pi}{2} > \arg(G_p(j\omega)) > -\pi + \phi_{PM}$$

where $\phi_p(\omega) = \arg(G_1(j\omega))$

and $|G_p(j\omega)| = |G_1(j\omega)|$

Plot the functions on the same complex plane, if there is an intersection, this corresponds to the controller parameters.

Note and store the controller parameters k_{p1} and k_{i1}.

Step 3: Controller $K_{22}(s)$ Tuning Parameters.

Form two complex functions such that

$$f_1(\omega) = -\frac{\cos\phi_p(\omega)}{GM_2|G_p(j\omega)|} - j\frac{\omega \sin\phi_p(\omega)}{GM_2|G_p(j\omega)|}$$

$$-\frac{\pi}{2} > \arg(G_p(j\omega)) > -\pi$$

$$f_2(\omega) = -\frac{\cos(\phi_{PM_2} - \phi_p(\omega))}{|G_p(j\omega)|} + j\frac{\omega \sin(\phi_{PM_2} - \phi_p(\omega))}{|G_p(j\omega)|}$$

$$-\frac{\pi}{2} > \arg(G_p(j\omega)) > -\pi + \phi_{PM}$$

where $\phi_p(\omega) = \arg(G_2(j\omega))$

and $|G_p(j\omega)| = |G_2(j\omega)|$

Plot the functions on the same complex plane, if there is an intersection, this corresponds to the controller parameters.

Note and store the controller parameters k_{p2} and k_{i2}.

Step 4: Tolerance Checking and Updating.

If $|k_{p2}(n) - k_{p2}(n-1)| < tol$ and $|k_{p1}(n) - k_{p1}(n-1)| < tol$ and $|k_{i2}(n) - k_{i2}(n-1)| < tol$ and $|k_{i1}(n) - k_{i1}(n-1)| < tol$ then STOP

Else $n := n+1$

go to Step 2, using the new controller parameters for both controllers. ●

Remarks

i) There is an assumption that both gain margins and phase margins can be met.

ii) There is an assumption that if a solution exists the method will converge towards that solution.

The above algorithm was used to design the decentralised control system for the Wood and Berry (1973) process. Two representative graphs, for the controller parameters, are shown in Figures 4.39 and 4.40.

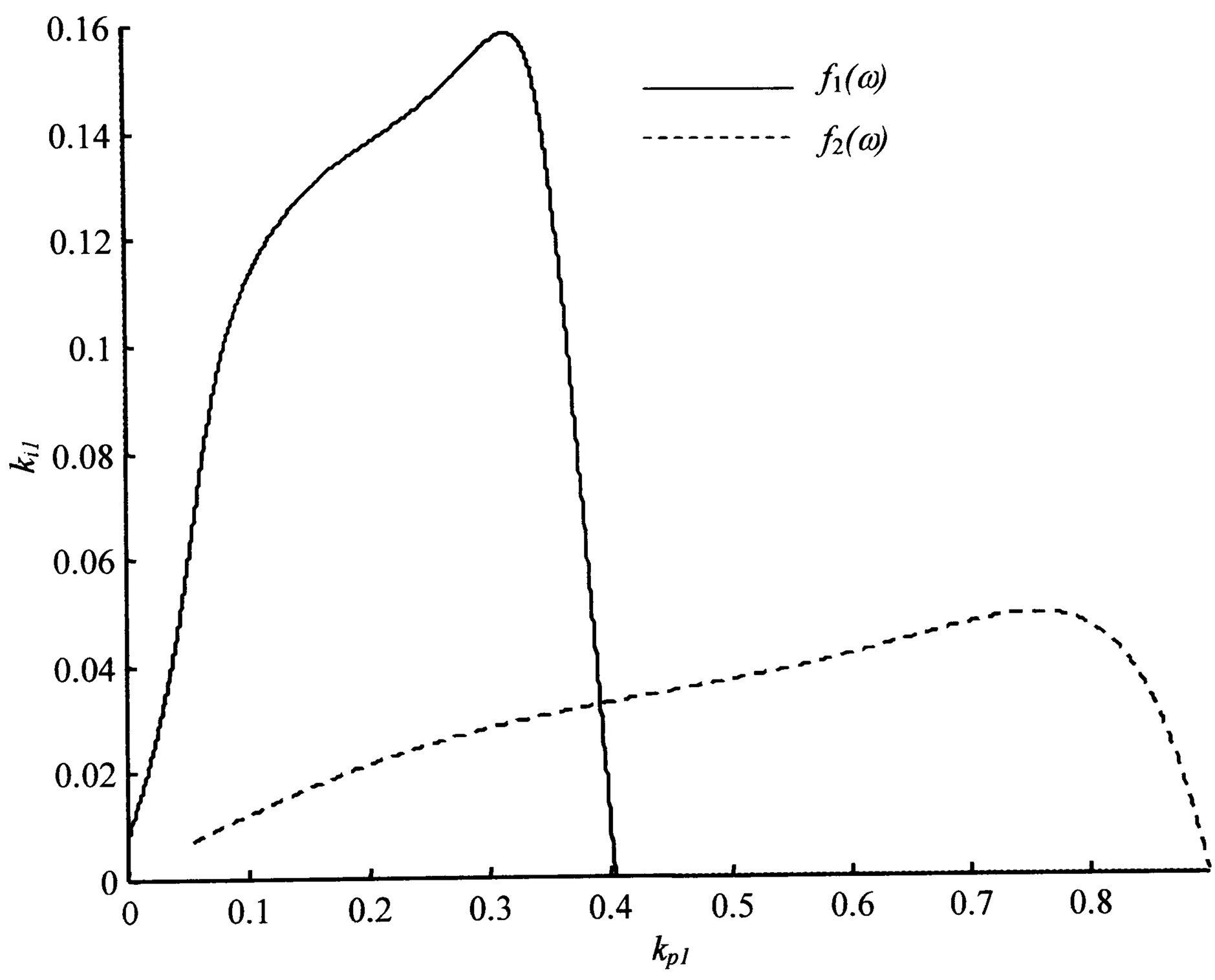


Figure 4.39: Exact Gain Margin and Phase Margin Loop One.

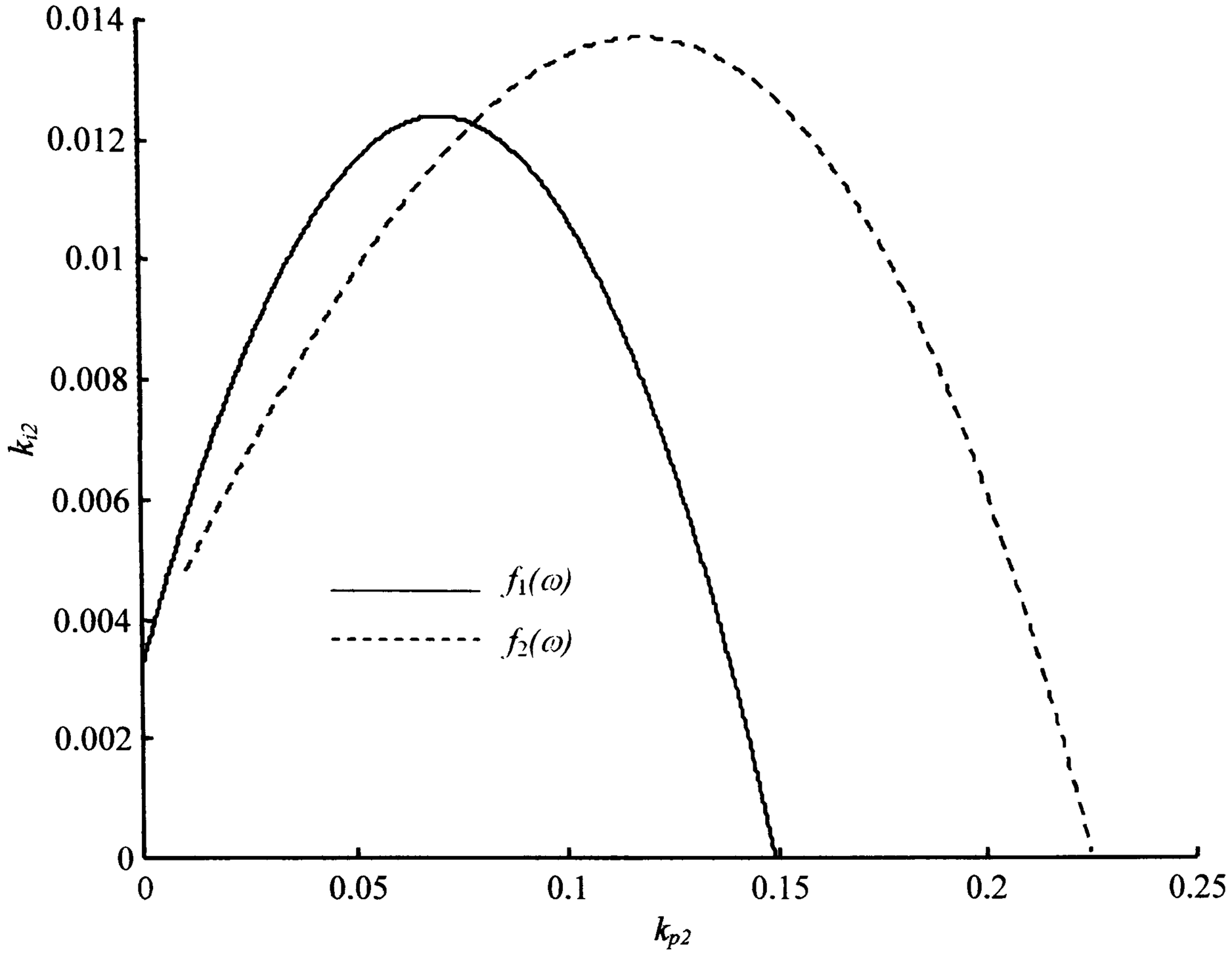


Figure 4.40: Exact Gain Margin and Phase Margin Loop Two.

The results from the tuning experiment are shown in a tabular format in Table 4.17.

Table 4.17: Exact Gain and Phase Margin Design Results

Iteration Index	Design Requirements (Loop One)		Design Requirements (Loop Two)	
	Gain Margin = 5 Phase Margin = 60°		Gain Margin = 3 Phase Margin = 30°	
	k_{p1}	k_{i1}	$-k_{p2}$	$-k_{i2}$
0	0.2295	0.0123	0.0779	0.005
1	0.3915	0.0338	0.0809	0.0144
2	0.3909	0.0318	0.0788	0.0123
3	0.3912	0.0325	0.0784	0.0123
4	0.3913	0.0326	0.0786	0.0123
5	0.3912	0.0325	0.0786	0.0123
6	0.3912	0.0326	0.0786	0.0123

From Table 4.17 it can be seen that the results converged after six iterations of the algorithm. To get the accuracy of the results shown in the Table the *Zoom* facility was used on the graphs produced. The initial tuning of the decentralised controllers was calculated using the method of Ho *et al* (1995) and requiring a gain margin of 5 and phase margin of 60° for loop one and a gain margin of 3 and a phase margin of 30° for loop two. To allow a comparison between the time domain performance of the Exact Gain and Phase Margin method and the Ho *et al* (1995) method the following step test was carried out. At time t = 0(s) a step input was applied to the reference input of loop one. At time t = 150(s) a step input was applied to the reference input of loop two. The results of the step tests are shown in Figures 4.41 and 4.42.

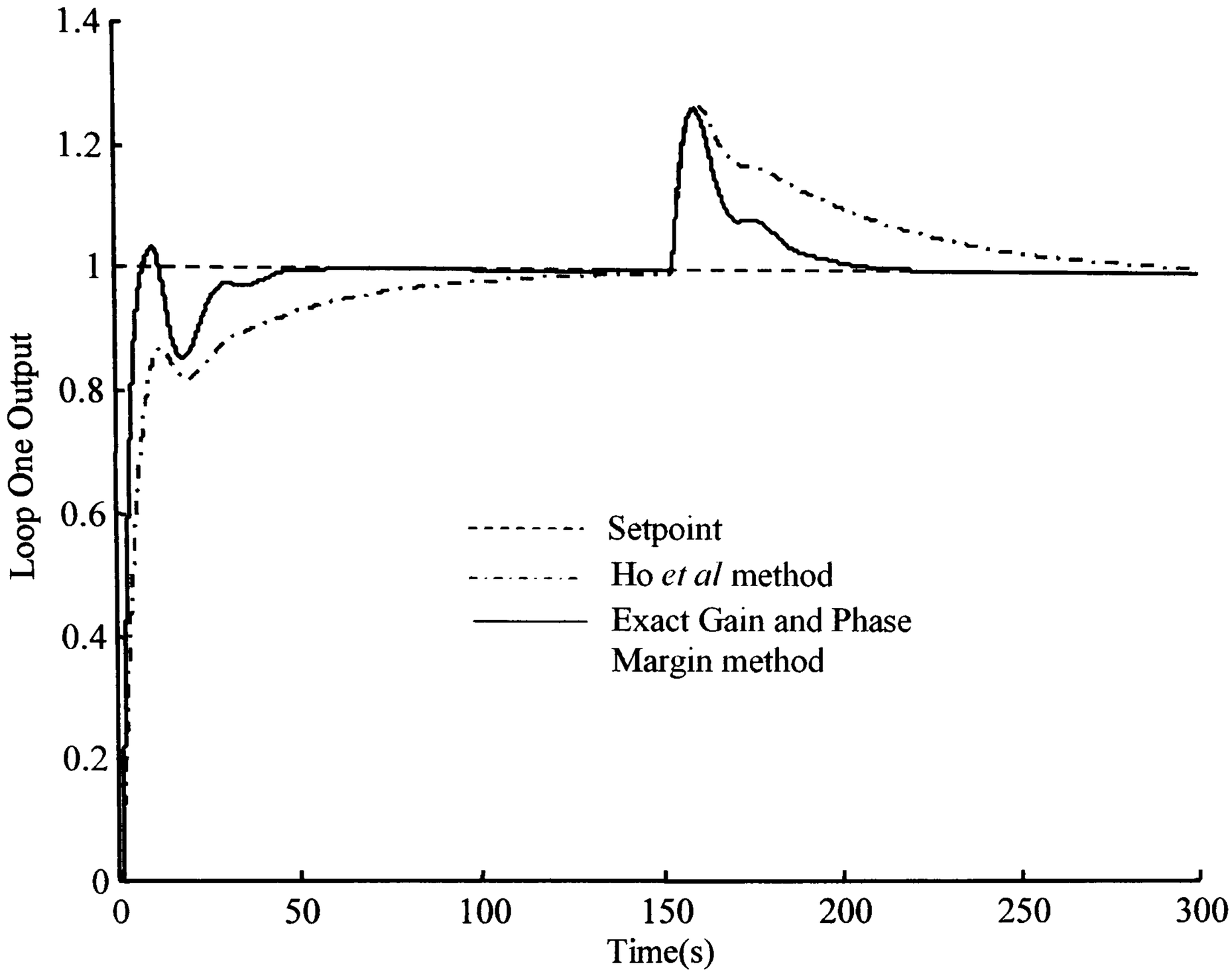


Figure 4.41: Step Response for Loop One; Ho vs. Exact Gain and Phase Margin.

From Figure 4.41 it can be seen that the response of Loop one when tuned using the exact gain and phase margin method gives a faster rise and settling time than that obtained using the Ho *et al* (1995) method. After the step input was applied to Loop two the interaction effects were removed more quickly by the controller tuned using exact gain and phase margin than that tuned using the Ho *et al* (1995) method. However neither method reduces the peak amplitude of the interaction term greatly.

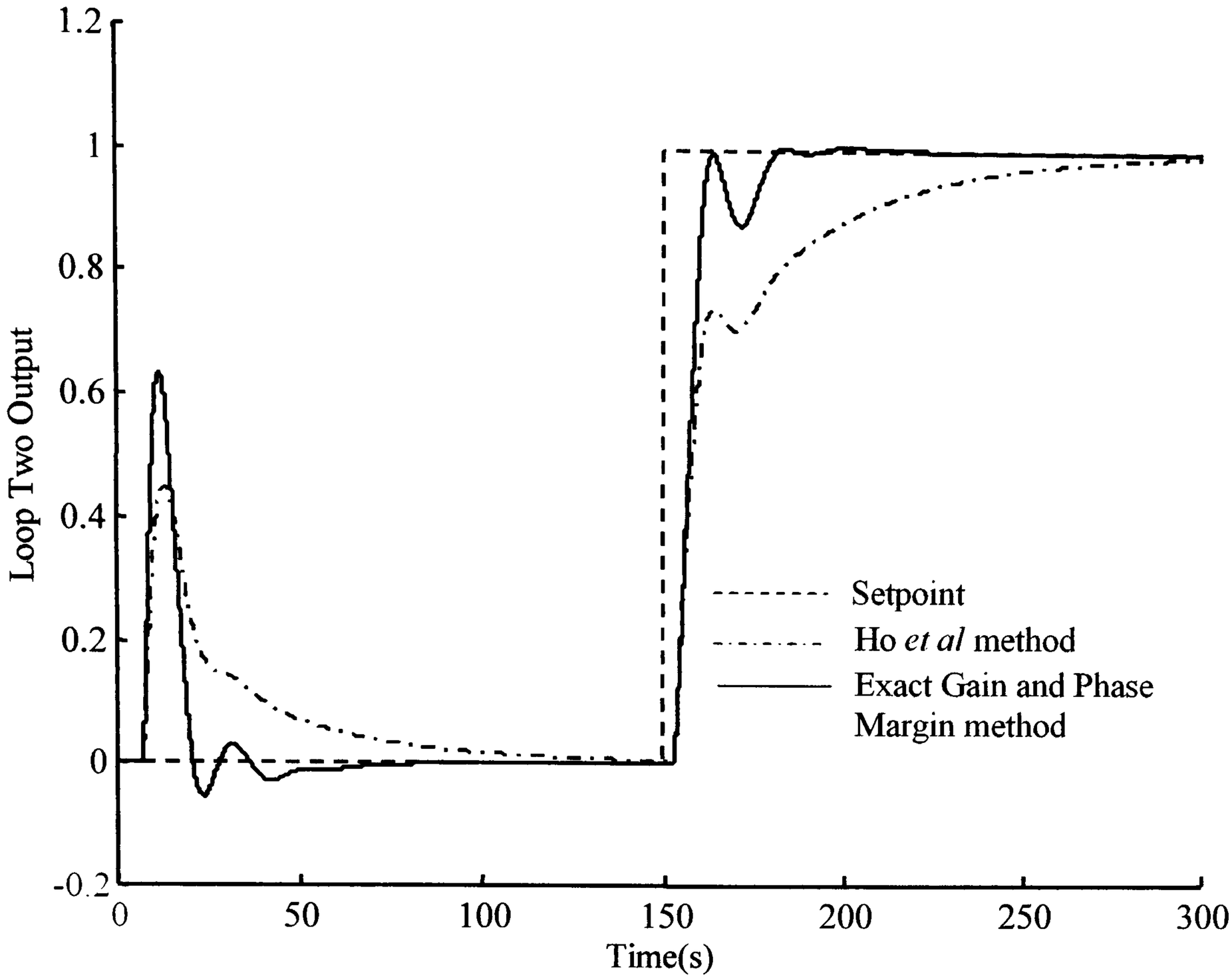


Figure 4.42: Step Response for Loop Two; Ho vs. Exact Gain and Phase Margin.

From Figure 4.42 it can be seen that there is a strong interaction between the two loops. The controller tuned using the Ho *et al* (1995) method is seen to be very slow at removing the effects of the interaction. However it does produce an interaction term with a reduced peak amplitude compared with that produced by the controller tuned using exact gain and phase margin design. When the step input is applied to loop two the response of the controller tuned using exact gain and phase margin is better as regards rise time and settling time to the controller tuned using the Ho *et al* (1995) method.

4.5 Summary Conclusions.

The sequential tuning method for cascade connected control systems due to Hang *et al* (1994) was discussed. The sequential tuning method of Hang *et al* is not carried out entirely in closed loop. The Hang *et al* method is only carried out in closed loop

after the initial tuning has been performed and subsequent re-tuning is required; only then is closed loop tuning carried out. To overcome this problem and allow closed loop tuning of cascade connected control systems to be performed a method was introduced that allows:

i) The closed loop identification of the processes within a cascade connected control system.

ii) The introduction of a test for closed loop stability when the inner controller parameters are to be updated.

The new method was employed to carry out the gain margin and phase margin design of a cascade connected control system. Further work is required to find a design method that combines the use of classical robustness measures with achieving an acceptable control system performance in the time domain for the closed loop cascade system.

The PLL method of system identification was extended to allow the identification of the elements of a transfer function matrix of a multivariable process connected in closed loop. The method is relatively easy to implement requiring only additional PLL identifiers and a switch to change over the excitation signal. The exact gain and phase margin PI controller design method due to Fung *et al* (1995) was extended to allow the design of de-centralised controllers for multivariable processes. A design example was carried out using the extended exact gain and phase margin design method.

5 Continuous Parameter Cycling Method of Model-Free Controller Design

5.1 Introduction

Today in the Process Industries, the first choice for a control algorithm is still usually PID although this situation is changing slowly (Blevins *et al*, 2002). The main reason for this lies in the fact that most industry based control practitioners have an understanding of how the closed loop performance of particular processes will be affected by changes in the PID controller parameters. Where there is a gap in the process knowledge, rudimentary parametric and non-parametric data may be determined by performing step tests, by the application of a relay experiment (Astrom and Hagglund, 1985; Yu, 1999) or using a phase locked loop test (Crowe and Johnson, 1998; 1999). If the control practitioner has such parametric or non-parametric data then rule based methods can be used to determine a PID controller (O'Dwyer, 1998a; 1998b).

Process operators are often under economic pressure to maximise the production from industrial processes and to reduce the cost of production. One means of helping to achieve these goals is to benchmark the existing controller against some form of performance index value (Harris, 1989; Thornhill *et al*, 1999). This type of approach entails the acceptance of a simple mathematical index as capturing desired control performance. For example, the Harris (1989) method uses the minimum variance criterion as a benchmark index function for controller performance assessment. It is a short step from minimum variance indices to LQ and LQG cost functions as the benchmark criterion. The use of LQ and LQG cost functions has lead to recent research focussing on:

i) More extended benchmarking formulations and

ii) Restricted structure control problems.

A key problem with both LQ/LQG benchmarking and restricted structure areas has been a requirement for an explicit parametric process model on which to base the necessary computations. In quite a different approach, particularly to the solution of restricted structure controller problems, Hjalmarsson and colleagues have performed cost function optimisation using only input and output data from the

process without the intermediate step of having to produce an explicit process model (Hjalmarsson *et al*, 1994; Hjalmarsson *et al* 1998). This technique is called Iterative Feedback Tuning (IFT). A frequency domain version is given by Kammer *et al* (2000). This avoidance of an explicit identified process model in the method has often led this approach to be termed a model-free approach.

The key step in Iterative Feedback Tuning is the computation of the cost function gradient so that the cost function can be optimised with respect to the controller parameters. This is not a new idea; in the early work on adaptive control systems Newton-like algorithms employing the gradient of a cost function with respect to the controller parameters were used (Talkin, 1961; McGrath *et al*, 1961; Narendra and McBride, 1964). Iterative Feedback Tuning is discussed in Section 5.2. Hjalmarsson and colleagues would also like to compute second order information residing in the cost function Hessian. Unfortunately, the Hjalmarsson IFT method is rather cumbersome at extracting the desired second order information, and this difficulty was one of the motivating reasons for looking again at the IFT method. Hence, this discussion also follows the theme of optimising a cost function using an implicit model or model-free approach. However instead of using special system inputs to compute gradients, controller parameter cycling is used to find the same quantities, and also produce the second order Hessian information in a much more direct manner. The theory of the new method that is used to minimise an LQ cost function online with respect to the parameters of a restricted structure controller is given in Section 5.3 of the chapter, along with a discussion of some implementation issues. In section 5.4, application examples are given for the design of restricted structure decentralised (PID) controllers, based on the controller parameter cycling method, for a multivariable process. Conclusions close the chapter.

5.2 Iterative Feedback Tuning

The seminal papers for the Iterative Feedback Tuning method due to Hjalmarsson *et al* (1994); Hjalmarsson *et al* (1998) use a system description involving a stochastic process output disturbance, a two-degree of freedom control law, a stochastic optimisation approach and a restricted structure control law. In IFT

the derivative of a control performance cost function with respect to the controller parameters is obtained by carrying out experiments on the closed loop system (Hjalmarsson *et al*, 1994; Hjalmarsson *et al*, 1998). For an iteration of the algorithm the current controller is utilised and is only updated at the end of the present iteration. Thus the data required for the method is able to be gathered while the process is operating in closed loop and the controller is updated on each iteration to improve the performance of the closed loop. Consider the closed loop system shown in Figure 5.1, where the system is controlled by a two-degrees of freedom controller, hence

$$U(s) = G_{cr}(s)R(s) - G_{cy}(s)Y(s) \tag{5.1}$$

where $R(s)$ is a deterministic reference signal. The disturbance input $v(t)$ is a zero mean stationary random process. The controller $G_c(s)=\{G_{cr}(s), G_{cy}(s)\}$ consists of a reference controller, $G_{cr}(s)$, and a feedback controller, $G_{cy}(s)$. The controller is characterised by a tuning parameter vector such that

$$\rho \in R^{n_p} \tag{5.2}$$

where n_p represents the number of tuning parameters of the two degrees of freedom controller, $G_c(s)$.

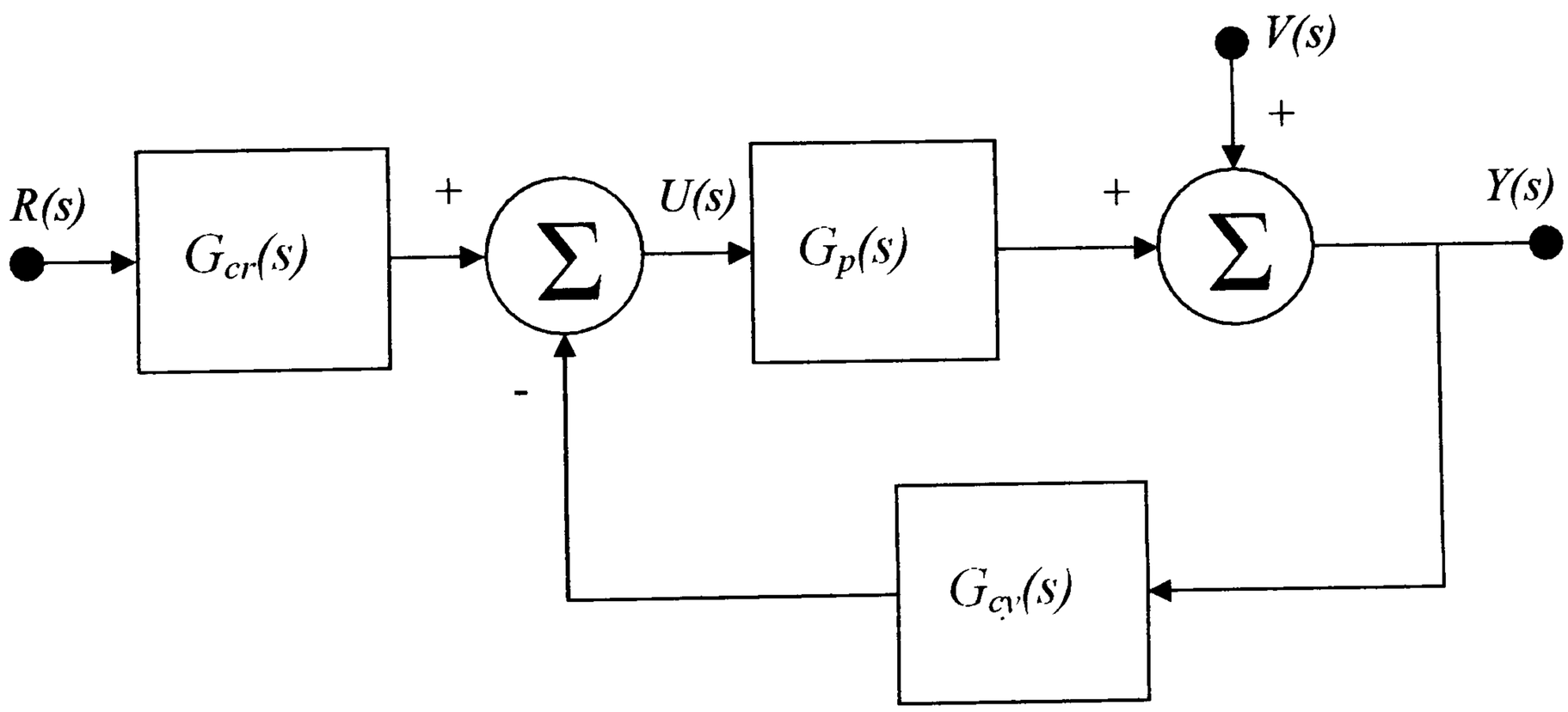


Figure 5.1: System Block Diagram of the Closed Loop.

The notation for the time domain signals y and u are given as $y_t(\rho)$ and $u_t(\rho)$. The subscript t denotes the dependence of these signals on time and the dependence on the controller parameters is given using the vector, ρ. Following the method in

Hjalmarsson *et al* (1998) a desired output response, $y_t^d(\rho)$, is defined. The error between the desired response and that produced by the closed loop system given in Figure 5.1 will be

$$\tilde{Y}_s(\rho) = Y_s(\rho) - Y^d(s) = \left(\frac{G_{cr}(s)G_p(s)}{1 + G_{cy}(s)G_p(s)} R(s) - Y^d(s) \right) + \frac{1}{1 + G_{cy}(s)G_p(s)} V(s) \tag{5.3}$$

where the subscript s denotes a dependency on the Laplace variable.

The cost function that is to be minimised is given by

$$J(\rho) = \frac{1}{2N} E\left\{ \sum_{t=1}^{N} \left(L_y \tilde{y}_t(\rho) \right)^2 + \lambda \sum_{t=1}^{N} \left(L_u u_t(\rho) \right)^2 \right\} \tag{5.4}$$

where E{.} denotes expectation with respect to *v(t)*.

The optimal controller parameter vector ρ that minimises $J(\rho)$ in equation (5.4) is defined as

$$\rho^* = \min_{w.r.t.\rho} J(\rho) \tag{5.5}$$

The cost function of equation (5.4) comprises two cost components. The first cost penalises any error between the closed loop response and the desired response and is weighted by the time domain filter term L_y. The second cost component penalises the control effort and is weighted by the time domain filter term L_u. In equation (5.4) the relative contribution between tracking the desired output and the control effort is adjusted by the term λ. In the sequel the filter terms L_y and L_u shall be assumed to be unity.

If the closed loop response and sensitivity functions of the closed loop system shown in Figure 5.1 are defined by

$$T_s(\rho) = \frac{G_p(s)G_{cr}(s)}{1 + G_p(s)G_{cy}(s)} \tag{5.6}$$

and

$$S_s(\rho) = \frac{1}{1 + G_p(s)G_{cy}(s)} \tag{5.7}$$

Then substitution of equations (5.6) and (5.7) into equation (5.4) and taking the expectation gives

$$J(\rho)=\frac{1}{2N}\sum_{t=1}^{N}\left(y_t^d-T(\rho)r_t\right)^2+\frac{1}{2}E\left\{(S(\rho)v)^2\right\}$$

$$+\lambda\frac{1}{2N}E\left\{\sum_{t=1}^{N}(u_t(\rho))^2\right\} \qquad (5.7)$$

The first term in equation (5.7) is the tracking error, the second term relates the contribution to the cost function of the disturbance and the final term relates to the control effort.

The minimisation of equation (5.4) is carried out by obtaining an expression for its gradient with respect to the controller parameter vector, ρ. Thus, differentiating equation (5.4) gives

$$\frac{\partial J}{\partial\rho}(\rho)=\frac{1}{N}E\left\{\sum_{t=1}^{N}\tilde{y}_t(\rho)\frac{\partial\tilde{y}_t}{\partial\rho}(\rho)+\lambda\sum_{t=1}^{N}u_t(\rho)\frac{\partial u_t}{\partial\rho}(\rho)\right\} \qquad (5.8)$$

If the gradient of the cost function is available then the value of ρ that minimises the cost function, $J(\rho)$, can be obtained using the following Newton-like algorithm.

$$\rho_{i+1}=\rho_i-\gamma_i R_i^{-1}\frac{\partial J}{\partial\rho}(\rho_i) \qquad (5.9)$$

where γ_i is a positive real scalar that determines the step size and R is a positive definite matrix that approximates the Hessian of $J(\rho)$. It is not possible to solve equation (5.8) for the gradient since it involves taking the expectation of unknown quantities. It is shown by Robbins and Monro (1951) that it is possible to determine the value of ρ that will minimise the cost function, $J(\rho)$, using a stochastic approximation algorithm of the form given by (5.9) so long as there is an unbiased estimate of the gradient term available. In order to calculate the gradient given by equation (5.8) the following terms are therefore required to be available or made available by experiment:

i) $\tilde{y}_t(\rho)$ and $u_t(\rho)$

ii) the gradients $\frac{\partial\tilde{y}_t}{\partial\rho}(\rho)$ and $\frac{\partial u_t}{\partial\rho}(\rho)$

iii) unbiased estimates of the product terms $\tilde{y}_t(\rho)\frac{\partial\tilde{y}_t}{\partial\rho}(\rho)$ and $u_t(\rho)\frac{\partial u_t}{\partial\rho}(\rho)$

The innovation in the Hjalmarsson *et al* method was to define a set of tests such that the terms in the gradient equation could be determined from the closed loop system itself. Three tests on the system are carried out, the first and the third tests simply collect data from the closed loop system operating with the current controller. The second test consists of taking the recorded closed loop output from the first experiment subtracting it from the reference input and using that signal as the reference input for the second test. If for each of the three sets of test data of length N data is gathered and denoting the reference data as

$$\{r_i^j\}, j = 1,2,3$$

and the corresponding output data as

$$\{y^j(\rho_i)\}, j = 1,2,3$$

Then for test one obtain

$$r_i^1 = r$$

giving rise to the output

$$y^1(\rho_i) = T(\rho_i)r + S(\rho_i)v_i^1$$

For test two obtain

$$r_i^2 = r - y^1(\rho_i)$$

giving rise to the output

$$y^2(\rho_i) = T(\rho_i)(r - y^1(\rho_i)) + S(\rho_i)v_i^2$$

For test three obtain

$$r_i^3 = r$$

giving rise to the output

$$y^3(\rho_i) = T(\rho_i)r + S(\rho_i)v_i^3$$

The tests result in an exact realisation of

$$\tilde{y}(\rho_i) = y^1(\rho_i) - y^d$$

and it is shown in Hjalmarsson *et al* (1998) that

$$\frac{\hat{\partial y}}{\partial \rho}(\rho_i) = \frac{1}{G_{cr}(\rho_i)}\left[\left(\frac{\partial G_{cr}}{\partial \rho}(\rho_i) - \frac{\partial G_{cy}}{\partial \rho}\right)y^3(\rho_i) + \frac{\partial G_{cy}}{\partial \rho}(\rho_i)y^2(\rho_i)\right]$$

is an unbiased estimate of

$$\frac{\partial y}{\partial \rho}(\rho_i)$$

The three tests described give rise to corresponding control signals such that

$$u^1(\rho_i) = S(\rho_i)(G_{cr}(\rho_i)r - G_{cy}(\rho_i)v_i^1)$$

$$u^2(\rho_i) = S(\rho_i)(G_{cr}(\rho_i)(r - y^1(\rho_i)) - G_{cy}(\rho_i)v_i^2)$$

$$u^3(\rho_i) = S(\rho_i)(G_{cr}(\rho_i)r - G_{cy}(\rho_i)v_i^3)$$

The above control signals are used to derive the estimates of the input related data required for the solution of the gradient equation (5.8). The control signals result in an exact realisation of

$$u(\rho_i) = u^1(\rho_i)$$

while in Hjalmarsson *et al* (1998) it is shown that

$$\hat{\frac{\partial u}{\partial \rho}}(\rho_i) = \frac{1}{G_{cr}(\rho_i)}\left[\left(\frac{\partial G_{cr}}{\partial \rho}(\rho_i) - \frac{\partial G_{cy}}{\partial \rho}\right)u^3(\rho_i) + \frac{\partial G_{cy}}{\partial \rho}(\rho_i)u^2(\rho_i)\right]$$

is an unbiased estimate of

$$\frac{\partial u}{\partial \rho}(\rho_i)$$

Thus an estimate of the gradient, based on the test results obtained from the closed loop system, can be obtained from

$$\hat{\frac{\partial J}{\partial \rho}}(\rho) = \frac{1}{N}\sum_{t=1}^{N}\left(\tilde{y}_t(\rho)\hat{\frac{\partial y_t}{\partial \rho}}(\rho) + \lambda u_t(\rho)\hat{\frac{\partial u_t}{\partial \rho}}(\rho)\right)$$

Thus Hjalmarsson *et al* have constructed a method that allows the controller to have its parameters updated whilst still connected in closed loop with the system that allows the cost function, equation (5.4), to be minimised. All of this is carried out without the explicit generation of a process model and since the controller parameters are updated iteratively the term Iterative Feedback Tuning (IFT) has been used by the originators of the method to refer to this method of controller tuning.

5.2.1 A deterministic LQ optimal Control Problem

In the above discussion on Iterative Feedback Tuning, the simplicity of the method is obscured by the rather general formulation of the problem. The salient features of the

Hjalmarsson *et al*, (1994) and Hjalmarsson *et al* (1998) Iterative Feedback Tuning method are given as:

i) a system description involving a stochastic process output disturbance

ii) a two degrees of freedom control law

iii) use of a stochastic optimisation approach (Robbins and Munro, 1951), and

iv) a restricted structure control law

A simpler approach to the method was adopted by Mahathanakiet *et al* (2002) where a deterministic version of Iterative Feed back Tuning was formulated. Consider Figure 5.2 where the system is assumed to be single-input, single-output and the controller implements a one degree of freedom control law.

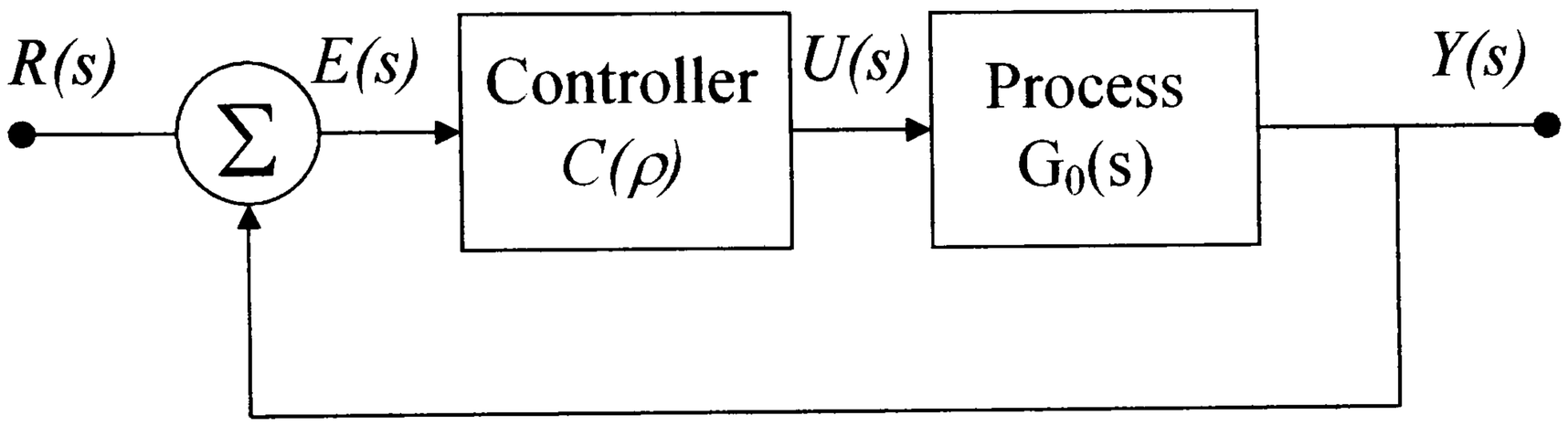


Figure 5.2: Unity feedback control loop

The controller $C(\rho)$ is given in terms of the controller parameter vector, $\rho \in R^{n_c}$, where n_c is the number of controller parameters. The notation for the time signals e, u, and y are given as, $e_t(\rho)$, $u_t(\rho)$ and $y_t(\rho)$. The subscript t denotes the dependence of these signals on time, and the dependence on the controller parameter is given using the vector, ρ. The Iterative Feedback Tuning optimisation problem is now formulated using the reference error and control signals of Figure 5.2, the LQ cost function to be minimised is given by

$$J(\rho) = \frac{1}{2T_f} \int_0^{T_f} \left((e_t(\rho))^2 + \lambda (u_t(\rho))^2 \right) dt \qquad (5.10)$$

Where, $e_t \in R$, $u_t \in R$, $\lambda \in R^+$, $\rho \in R^{n_c}$ and n_c is the number of controller parameters. The relative contribution between the error and control signals is adjusted by means of λ.

If equation (5.10) is now differentiated using Leibniz's Theorem for Differentiation of an Integral (Abramowitz and Stegun, 1972) the gradient function of the simplified problem is given by

$$\frac{\partial J}{\partial \rho} = \frac{1}{T_f}\int_0^{T_f}\left(\left(e_t(\rho)\frac{\partial e_t(\rho)}{\partial \rho}\right) + \lambda\left(u_t(\rho)\frac{\partial u_t(\rho)}{\partial \rho}\right)\right)dt \qquad (5.11)$$

The Generic Optimisation Problem

$$\min_{w.r.t.\,\rho \in R^{n_c}} J(\rho)$$

with

i) $\{\rho_i > 0,\ i = 1,\cdots,n_c\}$

ii) $C(\rho)$ closed loop stabilising.

This is seen to be a fixed structure or restricted structure LQ optimal control problem (Grimble and Johnson, 1988). Incorporating a limit process $T_f \to \infty$ yields the steady state optimisation problem. The condition ρ_i>0 ensures that the PID controller parameters are positive.

In order to find the optimal value of the cost function the gradient expression, equation (5.11), must be available. To construct the gradient expression the following time domain signals are required:

i) $e_t(\rho),\ u_t(\rho)$

ii) $\frac{\partial e_t(\rho)}{\partial \rho}$ and $\frac{\partial u_t(\rho)}{\partial \rho}$

The above time domain signals are found using the following closed loop system relationships. Consider Figure 5.2

$$Y_s(\rho) = \frac{G_0(s)C_s(\rho)}{1 + G_0(s)C_s(\rho)}R(s) = T_0(s)R(s) \qquad (5.12)$$

where $T_0(s)$ is the complementary sensitivity function.

$$E_s(\rho) = \frac{1}{1 + G_0(s)C_s(\rho)}R(s) = S_0(s)R(s) \qquad (5.13)$$

where $S_0(s)$ is the sensitivity function.

$$U_s(\rho)=\frac{C_s(\rho)}{1+G_0(s)C_s(\rho)}R(s)=S_0(s)C_s(\rho)R(s) \tag{5.14}$$

In equations (5.12) to (5.14) inclusive a simple notation for the Laplace transforms of the time signals $y_t(\rho)$, $e_t(\rho)$, and $u_t(\rho)$ was introduced. The subscript s denotes the dependence of these signals on the Laplace variable.

From equation (5.13) it can be shown that

$$\frac{\partial E_s(\rho)}{\partial\rho}=-(1+G_0(s)C_s(\rho))^{-2}G_0(s)\frac{\partial C_s(\rho)}{\partial\rho}R(s)$$

$$=\left[-\frac{1}{C_s(\rho)}\frac{\partial C_s(\rho)}{\partial\rho}\right]T_0(s)S_0(s)R(s)$$

$$=-\left[\frac{1}{C_s(\rho)}\frac{\partial C_s(\rho)}{\partial\rho}\right]T_0(s)E_s(\rho) \tag{5.15}$$

From equation (5.14) it can be shown that

$$\frac{\partial U_s(\rho)}{\partial\rho}=\frac{(1+G_0(s)C_s(\rho))\frac{\partial C_s(\rho)}{\partial\rho}-C_s(\rho)G_0(s)\frac{\partial C_s(\rho)}{\partial\rho}}{(1+G_0(s)C_s(\rho))^2}R(s)$$

$$=\frac{\frac{\partial C_s(\rho)}{\partial\rho}}{(1+G_0(s)C_s(\rho))^2}R(s)$$

$$=\left[\frac{1}{C_s(\rho)}\frac{\partial C_s(\rho)}{\partial\rho}\right]C_s(\rho)S_0^2(s)R(s)$$

$$=\left[\frac{1}{C_s(\rho)}\frac{\partial C_s(\rho)}{\partial\rho}\right]S_0(s)C_s(\rho)E_s(s) \tag{5.16}$$

In equations (5.15) and (5.16) there is a term of the form

$$G_{grad}(s)=\frac{1}{C_s(\rho)}\frac{\partial C_s(\rho)}{\partial\rho} \tag{5.17}$$

The calculation of $G_{grad}(s)$ is carried out as follows. Assume that in Figure 5.2 a PID controller is being used, thus

$$u_t(\rho)=K_p e_t(\rho)+K_i\int_0^t e_\tau(\rho)d\tau+K_d\frac{de_t(\rho)}{dt}$$

$$U_s(\rho) = \left(K_p + \frac{K_i}{s} + K_d s \right) E_s(\rho)$$

$$= \left(\rho_1 + \frac{\rho_2}{s} + \rho_3 s \right) E_s(\rho) \tag{5.18}$$

and hence

$$U_s(\rho) = C_s(\rho) E_s(\rho) \tag{5.19}$$

where

$$\rho \in \Re^{n_c}$$

and n_c is the number of controller parameters.

From equations (5.18) and (5.19) it can be seen that

$$C_s(\rho) = \left(\rho_1 + \frac{\rho_2}{s} + \rho_3 s \right)$$

hence

$$G_{grad}(s) = \frac{1}{C_s(\rho)} \frac{\partial C_s(\rho)}{\partial \rho}$$

$$G_{grad}(s) = \frac{1}{\left(\rho_1 + \frac{\rho_2}{s} + \rho_3 s \right)} \begin{bmatrix} 1 \\ 1 \\ \frac{1}{s} \\ s \end{bmatrix}$$

The time signals required to calculate the gradient given by equation (5.11) are derived from experiments on the closed loop system. Consider Figure 5.3 where it can be seen that the reference error $E^{(1)}(s)$, generated during the first experiment, is used as the reference input for the second experiment. Thus in the second experiment the following signals are available

$$Y^{(2)}(s) = T_0(s) E^{(1)}(s)$$

and

$$U^{(2)}(s) = S_0(s) C(\rho) E^{(1)}(s)$$

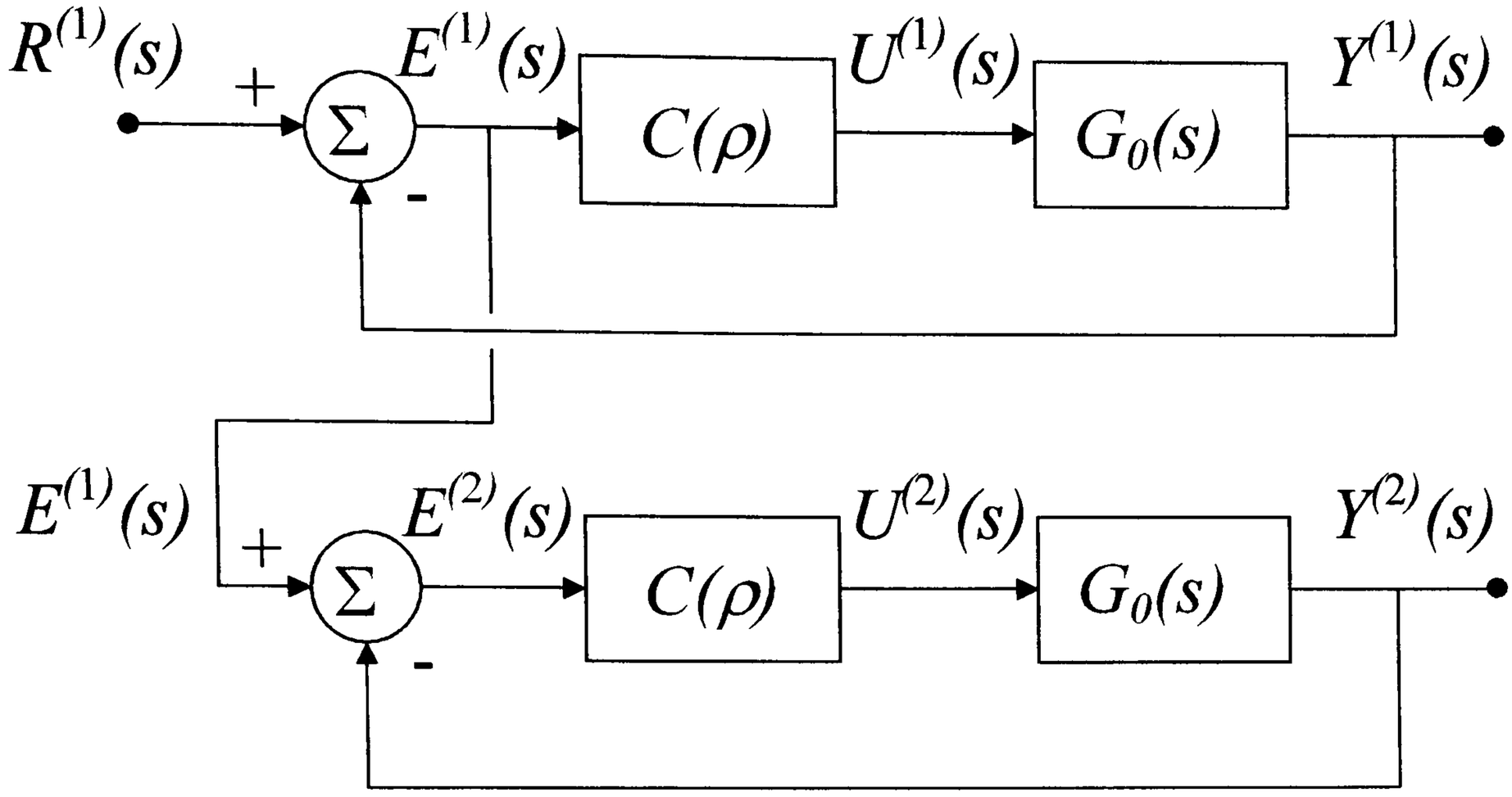


Figure 5.3: The Iterative Feedback Tuning Innovation

It is now possible to formulate an algorithm for the generation of the gradient as given by equation (5.11) as follows

Algorithm 5.1: The IFT Gradient Sub-Step Algorithm.

Step 1: Set up the controller at the k^{th} step

Use $\rho_k = \rho(k)$ in $C(\rho)$

Step 2: Response Generation

Run the closed loop system and record the time signals $e_t(\rho_k)$ and $u_t(\rho_k)$.

Run the closed loop using $e_t(\rho_k)$ as the reference input and record the time signals $y_t(\rho_k)$ and $u_t(\rho_k)$.

Step 3: Processing the Responses

Filter the signals $y_t(\rho_k)$ and $u_t(\rho_k)$ such that

$$\frac{\partial E_s(\rho)}{\partial \rho} = -G_{grad}(s)Y_s(\rho) \text{ and } \frac{\partial U_s(\rho)}{\partial \rho} = G_{grad}(s)U_s(\rho)$$

Step 4: Compute the Gradient

Use the time domain signals $e_t(\rho_k)$, $\left.\frac{\partial e_t(\rho)}{\partial \rho}\right|_{\rho=\rho(k)}$, $u_t(\rho_k)$ and $\left.\frac{\partial u_t(\rho)}{\partial \rho}\right|_{\rho=\rho(k)}$ to compute the gradient. ●

5.3 The Parameter Cycling Method of Tuning Industrial Controllers.

In the above two similar approaches to Iterative Feedback Tuning were discussed. In Hjalmarsson *et al* (1994) and Hjalmarsson *et al* (1998) the description of the Iterative Feedback Tuning method employed a stochastic system description with a two degree of freedom controller. However the general formulation of the problem led to the simplicity of the method being obscured. In an effort to add clarity to the Iterative Feedback Tuning method Mahathanakiet *et al* (2002) investigated the IFT method using a simple deterministic LQ cost function. Leibniz's Theorem for the Differentiation of an Integral (Abramowitz and Stegun, 1972) was used to derive the gradient function for the simplified problem. The gradient was then constructed using system responses and special input signals. In this section, a quite different approach is taken to the gradient and Hessian generation step.

5.3.1 Generating the Gradient and Hessian

A classical gradient computation would be based on perturbing the gain vector from $\rho(k)$ to $\rho(k)+\Delta\rho$ and calculating the respective cost functions $J(\rho(k))$ and $J(\rho(k)+\Delta\rho)$, so that numerical differences could be used to calculate an expression for the gradient, *viz.*

$$\frac{\partial J}{\partial \rho(k)} = \frac{J(\rho(k)+\Delta\rho(k)) - J(\rho(k))}{\Delta\rho(k)}$$

Further numerical perturbations can be used to calculate the Hessian information. It would then be possible to use a steepest descent or Newton method to calculate updated controller parameters. This method suffers from the problems of large numbers of gain perturbations and system response generations to calculate a gradient and a large number of gradient iterations to attain the (local) minimum. In the following a method of perturbing the gain is given such that estimates of both the gradient and the Hessian are obtained (Crowe *et al*, 2003b). This allows improved numerical routines, over steepest descent, to be used to reduce the number of iterations required to achieve convergence to the minimum of the cost function.

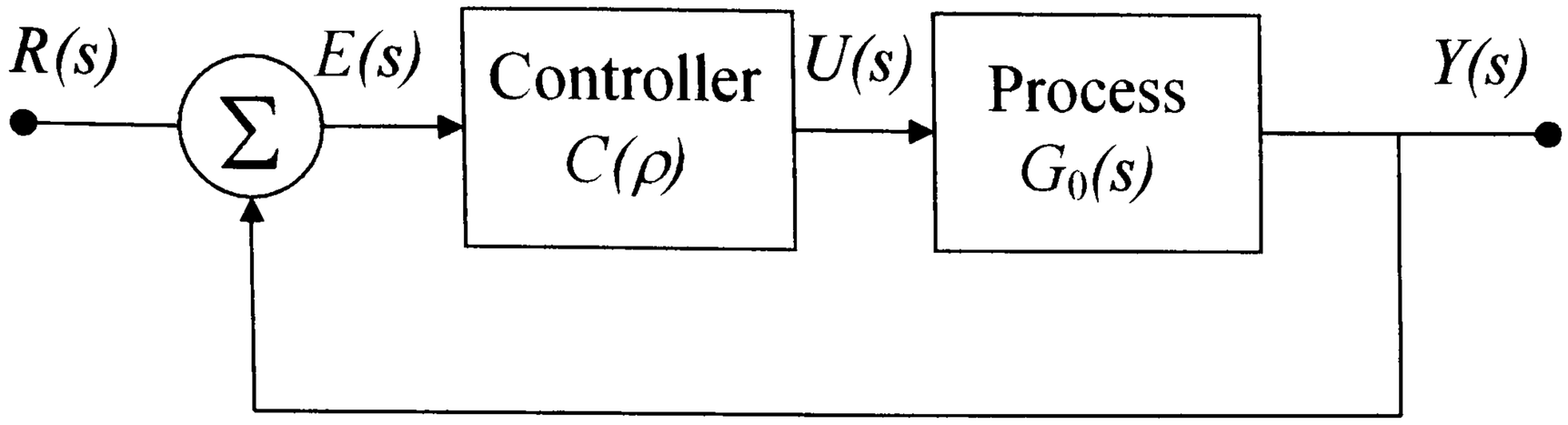


Figure 5.4: Unity feedback control loop

Consider the unity feedback system shown in Figure 5.4. It is assumed that the controller is characterised by means of a parameter vector, ρ, such that

$$\rho \in \Re^{n_c}$$

where n_c is the number of controller parameters.

If now it is assumed that the controller parameters are perturbed by an amount

$$\Delta\rho(t_\Delta) \in \Re^{n_c}$$

then Taylor's Theorem gives,

$$J(\rho + \Delta\rho(t_\Delta)) = J(\rho) + \Delta\rho(t_\Delta)^T \frac{\partial J}{\partial \rho} + R_2(\xi) \qquad (5.20)$$

$$J(\rho + \Delta\rho(t_\Delta)) = J(\rho) + \Delta\rho(t_\Delta)^T \frac{\partial J}{\partial \rho} +$$

$$\frac{1}{2}\Delta\rho(t_\Delta)^T H(\rho)\Delta\rho(t_\Delta) + R_3(\xi) \qquad (5.21)$$

where in equations (5.20) and (5.21)

$$\frac{\partial J}{\partial \rho} \in \Re^{n_c}$$

$$R_2(\xi) = \frac{1}{2}\Delta\rho(t_\Delta)^T H(\xi)\Delta\rho(t_\Delta)$$

and

$$R_3(\xi)$$

is a third order residual,

$$H_{ij} = \frac{\partial^2 J}{\partial\rho_i \partial\rho_j},\ H \in \Re^{n_c \times n_c}$$

and

$$\rho_i < \xi_i < \rho_i + \Delta\rho_i(t_\Delta)\ i = 1, \cdots, n_c$$

It should be noted that the perturbation of the controller parameters is time varying. Using these Taylor expansions, gradient and Hessian extraction is given by two new results. Sinusoidal function orthogonality plays a key role in the proof of the propositions. For the gradient result it is useful to define a set of integers that will define multiples of a fundamental gain perturbation frequency, ω_0. Introduce a set of n_c integers, denoted $\Im_G(n_c)$ with $\Im_G(n_c) = \{n_i,\ n_i \neq n_j, i, j \in [1,..n_c]\}$.

Proposition 5.1: Gradient extraction

Consider a time-varying controller gain vector perturbation

$$\Delta\rho(t_\Delta) \in \Re^{n_c}$$

where

$$\Delta\rho_i(t_\Delta) = \delta_i \sin n_i\omega_0 t_\Delta,\ \delta_i \in \Re,\ n_i \in \Im_G(n_c); i = 1,..,n_c.$$

Set

$$T_0 = \frac{2\pi}{\omega_0}$$

then for

$$i = 1,..,n_c,$$

$$\int_0^{T_0} J(\rho + \Delta\rho(t_\Delta))\sin n_i\omega_0 t_\Delta dt_\Delta = \left(\frac{\delta_i\pi}{\omega_0}\right)\frac{\partial J}{\partial\rho_i} + O(\delta_{\max}^2) \qquad (5.22)$$

where

$$\delta_{\max} = \max\{\delta_i, i = 1,..n_c\}$$ ●

Remarks

The result requires a suitable selection of the perturbation integration period, T_0, then the fundamental frequency is calculated as $\omega_0 = \frac{2\pi}{T_0}$.

Proof of Proposition 5.1

From Taylor's Theorem

$$J(\rho+\Delta\rho)=J(\rho)+\Delta\rho^T\frac{\partial J}{\partial\rho}+\frac{1}{2}\Delta\rho^T H(\xi)\Delta\rho$$

with $$H_{ij}(\xi)=\frac{\partial^2 J}{\partial\rho_i\partial\rho_j}\quad \rho<\xi<\rho+\Delta\rho$$

Thus if the perturbation is made time-varying,

$$J(\rho+\Delta\rho(t_\Delta))=J(\rho)+\sum_{i=1}^{n_c}\Delta\rho_i(t_\Delta)\frac{\partial J}{\partial\rho_i}+\frac{1}{2}\sum_{i=1}^{n_c}\sum_{j=1}^{n_c}H_{ij}(\xi(t_\Delta))\Delta\rho_i(t_\Delta)\Delta\rho_j(t_\Delta)$$

Let the perturbation be given by

$$\Delta\rho_i(t_\Delta)=\delta_i\sin n_i\omega_0 t_\Delta$$

then

$$J(\rho+\Delta\rho(t_\Delta))=J(\rho)+\sum_{i=1}^{n_c}\delta_i\frac{\partial J}{\partial\rho_i}\sin(n_i\omega_0 t_\Delta)$$

$$+\frac{1}{2}\sum_{i=1}^{n_c}\sum_{j=1}^{n_c}\delta_i\delta_j H_{ij}(\xi(t_\Delta))\sin(n_i\omega_0 t_\Delta)\sin(n_j\omega_0 t_\Delta)$$

Use $T_0=\frac{2\pi}{\omega_0}$, then

$$\int_0^{T_0}J(\rho+\Delta\rho(t_\Delta))\sin(n_i\omega_0 t_\Delta)dt_\Delta=J(\rho)\int_0^{T_0}\sin(n_i\omega_0 t_\Delta)dt_\Delta$$

$$+\delta_i\frac{\partial J}{\partial\rho_i}\int_0^{T_0}\sin^2(n_i\omega_0 t_\Delta)dt_\Delta+\sum_{\substack{j=1\\ j\neq i}}^{n_c}\delta_j\frac{\partial J}{\partial\rho_j}\int_0^{T_0}\sin(n_j\omega_0 t_\Delta)\sin(n_i\omega_0 t_\Delta)dt_\Delta$$

$$+\frac{1}{2}\sum_{i=1}^{n_c}\sum_{j=1}^{n_c}\delta_i\delta_j\int_0^{T_0}H_{ij}(\xi(t_\Delta))\sin^2(n_i\omega_0 t_\Delta)\sin(n_j\omega_0 t_\Delta)dt_\Delta \qquad (5.23)$$

Resolution of the identities in equation (5.23), gives

a) $$\int_0^{T_0}\sin(n_i\omega_0 t_\Delta)dt_\Delta=\left[\frac{-\cos(n_i\omega_0 t_\Delta)}{n_i\omega_0}\right]_0^{T_0}$$

$$=-\frac{\cos(2n_i\pi)}{n_i\omega_0}+\frac{1}{n_i\omega_0}=0 \qquad (5.24)$$

b) $$\int_0^{T_0} \sin^2(n_i \omega_0 t_\Delta) dt_\Delta = \left[\frac{t_\Delta}{2} - \frac{\sin(2n_i \omega_0 t_\Delta)}{4n_i \omega_0} \right]_0^{T_0 = \frac{2\pi}{\omega_0}}$$

$$= \frac{\pi}{\omega_0} - \frac{\sin(4n_i \pi)}{4n_i \omega_0} = \frac{\pi}{\omega_0}$$

c) $$\int_0^{T_0} \sin(n_j \omega_0 t_\Delta) \sin(n_i \omega_0 t_\Delta) dt_\Delta = \left[\frac{\sin((n_j - n_i)\omega_0 t_\Delta)}{2(n_j - n_i)\omega_0} - \frac{\sin((n_j + n_i)\omega_0 t_\Delta)}{2(n_j + n_i)\omega_0} \right]_0^{T_0 = \frac{2\pi}{\omega_0}}$$

with $n_j^2 \neq n_i^2$

$$= \frac{\sin(2(n_j - n_i)\pi)}{2(n_j - n_i)\omega_0} - \frac{\sin(2(n_j + n_i)\pi)}{2(n_j + n_i)\omega_0} = 0 \qquad (5.25)$$

d) $$\int_0^{T_0} H_{ij}(\xi(t_\Delta)) \sin^2(n_i \omega_0 t_\Delta) \sin(n_j \omega_0 t_\Delta) dt_\Delta < B < \infty$$

where B is a finite bound. Thus $R_2(\xi)$ is dominated by $\delta_{\max}^2$, where

$$\delta_{\max}^2 = \max_{i=1,\ldots,n_c} \{\delta_i^2\}$$

and is written as $$O(\delta_{\max}^2) \qquad (5.26)$$

Substitution of equations (5.23) to (5.26) inclusive, into equation (5.23) gives

$$\int_0^{T_0} J(\rho + \Delta\rho(t_\Delta)) \sin n_i \omega_0 t_\Delta dt_\Delta = \left(\frac{\delta_i \pi}{\omega_0} \right) \frac{\partial J}{\partial \rho_i} + O(\delta_{\max}^2)$$

●

Sinusoidal function orthogonality also plays a key role in the result for the extraction of the Hessian information. The Hessian is symmetric and has $n_c(n_c - 1)/2$ unknown elements. For an orthogonality based on a sine perturbation of the gain parameters and a cosine extraction of the Hessian, introduce a set of $n_c(n_c - 1)/2$ integers, denoted $\Im_H^{s\sim c}(n_c)$. Given the set of gain excitation multiples, $\Im_G(n_c)$ then the set $\Im_H^{s\sim c}(n_c)$ comprises integers, n_{ij} such that

i) $n_{ij} = n_i + n_j$

ii) $n_{ij} \neq n_{i_1 j_1}$

iii) $n_{ij} \neq n_{i_1} - n_{j_1}$ for all pairs $(i,j),(i_1,j_1)$ where $i = 1,..n_c; j = i,..,n_c$ and $i_1 = 1,...n_c; j_1 = i,..,n_c$.

Proposition 5.2: Hessian extraction

Consider a time varying controller gain vector perturbation $\Delta\rho_i(t_\Delta)$, such that

$$\Delta\rho_i(t_\Delta) = \delta_i \sin n_i\omega_0 t_\Delta$$

with

$$\delta_i \in \Re,\ n_i \in \Im_G(n_c); i = 1,..,n_c .$$

Set

$$T_0 = \frac{2\pi}{\omega_0}$$

Define a set of $n_c(n_c - 1)/2$ integers such that:

i) $n_{ij} = n_i + n_j$

ii) $n_{ij} \neq n_{i_1 j_1}$

iii) $n_{ij} \neq n_{i_1} - n_{j_1}$

for pairs (i, j) and (i_1, j_1) where $i = 1,..,n_c$; $j = 1,..,n_c$ and $i_1 = 1,..,n_c$; $j_1 = 1,..,n_c$

then,

$$\int_0^{T_0} J(\rho + \Delta\rho(t_\Delta))\cos n_{ij}\omega_0 t_\Delta dt_\Delta =$$

$$\left(-\frac{\delta_i\delta_j\pi\, f(i,j)}{4\omega_0}\right)\frac{\partial^2 J}{\partial\rho_i\partial\rho_j} + O(\delta_{\max}^3) \qquad (5.27)$$

where $$f(i,j) = \begin{cases} 1 & i = j \\ 2 & otherwise \end{cases}$$

●

Proof of Proposition 5.2

From Taylor's Theorem

$$J(\rho + \Delta\rho(t_\Delta)) = J(\rho) + \Delta\rho^T(t_\Delta)\frac{\partial J}{\partial\rho} + \frac{1}{2}\Delta\rho^T(t_\Delta)H_{ij}(\rho)\Delta\rho^T(t_\Delta) + R_3(\xi)$$

where $$\rho_i < \xi_i < \rho_i + \Delta\rho_i(t_\Delta)$$

If it is assumed that the perturbation is given by,

$$\Delta\rho_i(t_\Delta) = \delta_i \sin n_i\omega_0 t_\Delta$$

with $\delta_i \in \Re$, and $n_i \in \Im_G$, $i = 1,\ldots,n_c$, then

$$J(\rho + \Delta\rho(t_\Delta)) = J(\rho) + \sum_{i=1}^{n_c} \delta_i \sin(n_i\omega_0 t_\Delta)\frac{\partial J}{\partial \rho_i}$$

$$+ \frac{1}{2}\sum_{i=1}^{n_c}\sum_{j=1}^{n_c} \delta_i\delta_j H_{ij}(\rho)\sin(n_i\omega_0 t_\Delta)\sin(n_j\omega_0 t_\Delta) + R_3(\xi)$$

To find the $(i_1, j_1)^{th}$ element of H_{ij}, introduce the frequency $n_{i_1 j_1}$, where $i_1 = 1,..,n_c$ and $j_1 = 1,..,n_c$ hence using $T_0 = \frac{2\pi}{\omega_0}$;

$$\int_0^{T_0} J(\rho + \Delta\rho(t_\Delta))\cos(n_{i_1 j_1}\omega_0 t_\Delta)dt_\Delta =$$

$$J(\rho)\int_0^{T_0}\cos(n_{i_1 j_1}\omega_0 t_\Delta)dt_\Delta + \sum_{i=1}^{n_c}\delta_i\frac{\partial J}{\partial \rho_i}\int_0^{T_0}\sin(n_i\omega_0 t_\Delta)\cos(n_{i_1 j_1}\omega_0 t_\Delta)dt_\Delta$$

$$+ \frac{1}{2}\sum_{i=1}^{n_c}\sum_{j=1}^{n_c}\delta_i\delta_j H_{ij}(\rho)\int_0^{T_0}\sin(n_i\omega_0 t_\Delta)\sin(n_j\omega_0 t_\Delta)\cos(n_{i_1 j_1}\omega_0 t_\Delta)dt_\Delta$$

$$+ \int_0^{T_0} R_3(\xi(t_\Delta))\cos(n_{i_1 j_1}\omega_0 t_\Delta)dt_\Delta \qquad (5.28)$$

Resolution of the Identities in equation (5.28)

a) $$\int_0^{T_0}\cos(n_{i_1 j_1}\omega_0 t_\Delta)dt_\Delta = -\left.\frac{\sin(n_{i_1 j_1}\omega_0 t_\Delta)}{n_{i_1 j_1}\omega_0}\right|_0^{T_0=\frac{2\pi}{\omega_0}} = 0 \qquad (5.29)$$

b) $$\int_0^{T_0}\sin(n_i\omega_0 t_\Delta)\cos(n_{i_1 j_1}\omega_0 t_\Delta)dt_\Delta =$$

$$-\frac{1}{2}\left[\frac{\cos\left((n_i - n_{i_1 j_1})\omega_0 t_\Delta\right)}{(n_i - n_{i_1 j_1})\omega_0} + \frac{\cos\left((n_i + n_{i_1 j_1})\omega_0 t_\Delta\right)}{(n_i + n_{i_1 j_1})\omega_0}\right]_0^{T_0=\frac{2\pi}{\omega_0}}$$

when $n_i \neq n_{i_1 j_1}$

$$-\frac{1}{2}\left[\frac{\cos\left(\left(n_i - n_{i_1 j_1}\right)2\pi\right)}{\left(n_i - n_{i_1 j_1}\right)\omega_0} + \frac{\cos\left(\left(n_i + n_{i_1 j_1}\right)2\pi\right)}{\left(n_i + n_{i_1 j_1}\right)\omega_0}\right]$$

$$+\frac{1}{2}\left[\frac{1}{\left(n_i - n_{i_1 j_1}\right)\omega_0} + \frac{1}{\left(n_i + n_{i_1 j_1}\right)\omega_0}\right] = 0 \qquad (5.30)$$

when $n_i = n_{i_1 j_1}$

$$\int_0^{T_0} \sin\left(n_i \omega_0 t_\Delta\right)\cos\left(n_{i_1 j_1}\omega_0 t_\Delta\right)dt_\Delta$$

$$= \int_0^{T_0} \sin\left(n_i \omega_0 t_\Delta\right)\cos\left(n_i \omega_0 t_\Delta\right)dt_\Delta$$

$$= \frac{1}{2}\int_0^{T_0} \sin\left(2n_i \omega_0 t\Delta\right)dt_\Delta$$

$$= -\frac{1}{2}\frac{\cos\left(2n_i \omega_0 t_\Delta\right)}{2n_i\omega_0}\Bigg|_0^{T_0=\frac{2\pi}{\omega_0}} = -\frac{1}{4n_i\omega_0}\left[-\cos\left(4n_i\pi\right) + \cos\left(0\right)\right] = 0 \qquad (5.31)$$

c) $$I_{ij} = \int_0^{T_0} \sin\left(n_i \omega_0 t_\Delta\right)\sin\left(n_j \omega_0 t_\Delta\right)\cos\left(n_{i_1 j_1}\omega_0 t_\Delta\right)dt_\Delta$$

$$= \int_0^{T_0} \frac{1}{2}\left[\cos\left(\left(n_i - n_j\right)\omega_0 t_\Delta\right) - \cos\left(\left(n_i + n_j\right)\omega_0 t_\Delta\right)\right]\cos\left(n_{i_1 j_1}\omega_0 t_\Delta\right)dt_\Delta$$

assuming that $n_{i_1 j_1} \neq n_i + n_j$ and $n_{i_1 j_1} \neq n_i - n_j$, then

$$I_{ij} = \frac{1}{2}\left[\frac{\sin\left(\left(n_i - n_j - n_{i_1 j_1}\right)\omega_0 t_\Delta\right)}{2\left(n_i - n_j - n_{i_1 j_1}\right)\omega_0} + \frac{\sin\left(\left(n_i - n_j + n_{i_1 j_1}\right)\omega_0 t_\Delta\right)}{2\left(n_i - n_j + n_{i_1 j_1}\right)\omega_0}\right.$$

$$\left. - \frac{\sin\left(\left(n_i + n_j - n_{i_1 j_1}\right)\omega_0 t_\Delta\right)}{2\left(n_i + n_j - n_{i_1 j_1}\right)\omega_0} - \frac{\sin\left(\left(n_i + n_j + n_{i_1 j_1}\right)\omega_0 t_\Delta\right)}{2\left(n_i + n_j + n_{i_1 j_1}\right)\omega_0}\right]_0^{T_0=\frac{2\pi}{\omega_0}} = 0 \qquad (5.32)$$

whenever $n_{i_1 j_1} \neq n_i + n_j$ with $n_{i_1 j_1} \neq n_i - n_j$. This covers all of the non-selective cases.

Now choose $n_{i_1 j_1} = n_i + n_j$ with $n_{i_1 j_1} \neq n_i - n_j$, then

d) $$I_{ij} = \frac{1}{2}\int_0^{T_0} \cos\left(\left(n_i - n_j\right)\omega_0 t_\Delta\right)\cos\left(n_{i_1 j_1}\omega_0 t_\Delta\right)dt_\Delta$$

$$-\frac{1}{2}\int_0^{T_0}\cos^2\left(\left(n_i+n_j\right)\omega_0 t_\Delta\right)dt_\Delta$$

The first integral is zero since $n_{i_1 j_1} \neq n_i - n_j$

$$=-\frac{1}{2}\left[\frac{t_\Delta}{2}+\frac{\sin\left(2\left(n_i+n_j\right)\omega_0 t_\Delta\right)}{4\left(n_i+n_j\right)\omega_0}\right]_0^{T_0=\frac{2\pi}{\omega_0}}=-\frac{\pi}{2\omega_0} \qquad (5.33)$$

Substitution of equations (5.29) to (5.33) inclusive, into equation (5.28) gives

$$\int_0^{T_0} J\left(\rho+\Delta\rho(t_\Delta)\right)\cos n_{ij}\omega_0 t_\Delta dt_\Delta =$$

$$\left(-\frac{\delta_i\delta_j\pi\, f(i,j)}{4\omega_0}\right)\frac{\partial^2 J}{\partial\rho_i\partial\rho_j}+O(\delta_{\max}^3)$$

where $$f(i,j)=\begin{cases}1 & i=j\\ 2 & otherwise\end{cases}$$

●

The proofs of the propositions define the sets $\Im_G(n_c)$, $\Im_H^{s\sim c}(n_c)$, and it is useful to note that other variants of the propositions are possible. For example, cosine perturbation of the gain elements followed by cosine extraction of the Hessian elements can be constructed. In this case, the integer set denoted $\Im_H^{c\sim c}(n_c)$ which specifies the frequencies used by the cosine Hessian extraction functions has to be defined. Thus, $\Im_H^{c\sim c}(n_c)$ has $n_c(n_c-1)/2$ integers n_{ij} such that,

(i) $n_{ij} \notin \Im_G(n_c)$

(ii) $n_{ij} = n_i + n_j$

(iii) $n_{ij} \neq n_{i_1 j_1}$

(iv) $n_{ij} \neq n_{i_1} - n_{j_1}$ for all pairs $(i,j),(i_1,j_1)$ where $i=1,..n_c; j=i,..,n_c$ and $i_1=1,..n_c; j_1=i_1,..,n_c$.

Apart from these theoretical issues, other numerical considerations are necessary to convert the theory into working algorithms as outlined in the next section.

5.4 Implementation Issues

5.4.1 Numerical Selections for the Algorithm

Selection of T_f, T_0, ω_0 In the calculation of the cost function it is not practical to allow the online experiment to continue for an infinitely long time. Thus, the integration period is fixed to $T_f = 5*\tau(dominant)$, where it is assumed that the system has settled to within 1% of the steady state value after this time has elapsed, and then set $T_0 = T_f$. This gives the frequency of the basic perturbation frequency as, $\omega_0 = \frac{2\pi}{T_0}$.

Selection of Gain Perturbation Amplitudes.

The selection of the gain perturbation amplitudes $\delta_i \in \Re$ will be problem dependent. However there are several considerations here, firstly the perturbation should not cause closed loop instability, and secondly, the size can be made to minimise the disturbance to the system outputs. This latter issue will be of particular concern to production personnel when conducting online experiments. Finally, the perturbation size is directly linked to the accuracy of gradient and Hessian extraction as given in the propositions.

Selection of Orthogonality Integer Sets.

Constructing sets $\Im_G(n_c)$, $\Im_H^{S\sim C}(n_c)$, $\Im_H^{C\sim C}(n_c)$. A frequency multiple of ω_0 must be assigned to each gain parameter perturbation, $\Delta\rho_i(t_\Delta)$. The choice of frequency multiples is dependent on the choice of whether a sine or cosine controller gain perturbation function is chosen. If only the gradient is to be extracted, then the rules for the construction of $\Im_G(n_c)$ are straightforward. If, however the Hessian is also to be extracted then the construction of the pair $\Im_G(n_c)$, $\Im_H^{S\sim C}(n_c)$ or the pair $\Im_G(n_c)$,

$\mathfrak{I}_H^{C\sim C}(n_c)$ is a little more involved. Table 5.1 shows feasible frequency multiples for the case of a single PID controller, with three gains.

Thus, for a sine perturbation and a cosine extraction function for the Hessian, the duplication of a frequency between the $\mathfrak{I}_G(n_c)$ and the set $\mathfrak{I}_H^{S\sim C}(n_c)$ is permissible (see for example in Table 5.1, that $n_2 = n_{11} = 4$ in $\mathfrak{I}_H^{S\sim C}(n_c)$).

Table 5.1: Feasible frequency multiples

$\mathfrak{I}_G(n_c)$	$n_1 = 2$	$n_2 = 4$	$n_3 = 5$
$\mathfrak{I}_H^{s\sim c}(n_c)$	$n_{11} = 4$	$n_{12} = 6$	$n_{13} = 7$
	--	$n_{22} = 8$	$n_{23} = 9$
	--	--	$n_{33} = 10$
$\mathfrak{I}_G(n_c)$	$n_1 = 1$	$n_2 = 5$	$n_3 = 8$
$\mathfrak{I}_H^{c\sim c}(n_c)$	$n_{11} = 2$	$n_{12} = 6$	$n_{13} = 9$
	--	$n_{22} = 10$	$n_{23} = 13$
	--	--	$n_{33} = 16$

Selection of Integration Formula.

The time scales over which the cost function is calculated and the parameter perturbation signals evolve are separate. The cost function is calculated in real time, whereas the parameter perturbation signal uses the time-domain, t_Δ. Standard numerical integration formulas are used to compute the two extraction integrals,

$$\int_0^{T_0} J(\rho + \Delta\rho(t_\Delta))\sin n_i \omega_0 t_\Delta dt_\Delta$$

and

$$\int_0^{T_0} J(\rho + \Delta\rho(t_\Delta))\cos n_{ij} \omega_0 t_\Delta dt_\Delta .$$

These use the discrete time where $t_\Delta = kT$, $k = 0,1,2,\cdots,N$, and T is the integration step size or sampling interval. Clearly, step size, T has to be chosen to achieve maximal accuracy from the integration method and provide sufficient resolution for the maximum frequency of the perturbation and extraction signals.

Exploitation of Symmetry Properties.

From the proof of Proposition 5.1 recall that with a time-varying parameter perturbation, $\Delta\rho(t_\Delta) = \delta_i \sin(n_i\omega_0 t_\Delta)$, the cost $J(\rho + \Delta\rho(t_\Delta))$ is given by

$$J(\rho + \Delta\rho(t_\Delta)) = J(\rho) + \sum_{i=1}^{n_c} \delta_i \frac{\partial J}{\partial \rho_i} \sin(n_i\omega_0 t_\Delta)$$

$$+ \frac{1}{2}\sum_{i=1}^{n_c}\sum_{j=1}^{n_c} \delta_i\delta_j H_{ij}(\xi(t_\Delta))\sin(n_i\omega_0 t_\Delta)\sin(n_j\omega_0 t_\Delta) + \cdots$$

Thus it can be seen that the cost is composed of a sum of constant terms, cosinusoidal terms and sinusoidal terms. Since a sinusoidal function possesses odd symmetry and a cosinusoidal function even symmetry, the sum of odd symmetrical components and even symmetrical components returns an odd symmetrical component. Thus the cost function, $J(\rho + \Delta\rho(t_\Delta))$, will also have odd symmetry. If the cost is determined over half of the time period, T_0, then by exploiting the odd symmetry property the entire time-varying cost can be constructed. Similarly if the parameter perturbation is cosinusoidal then it can be shown that the resulting cost function will posses even symmetry. Thus only half the time-varying cost function is required to be determined for either sinusoidal or cosinusoidal parameter perturbation. The time-varying cost function relating to the third iteration of the case study example discussed later in the chapter is shown in Figure 5.5.

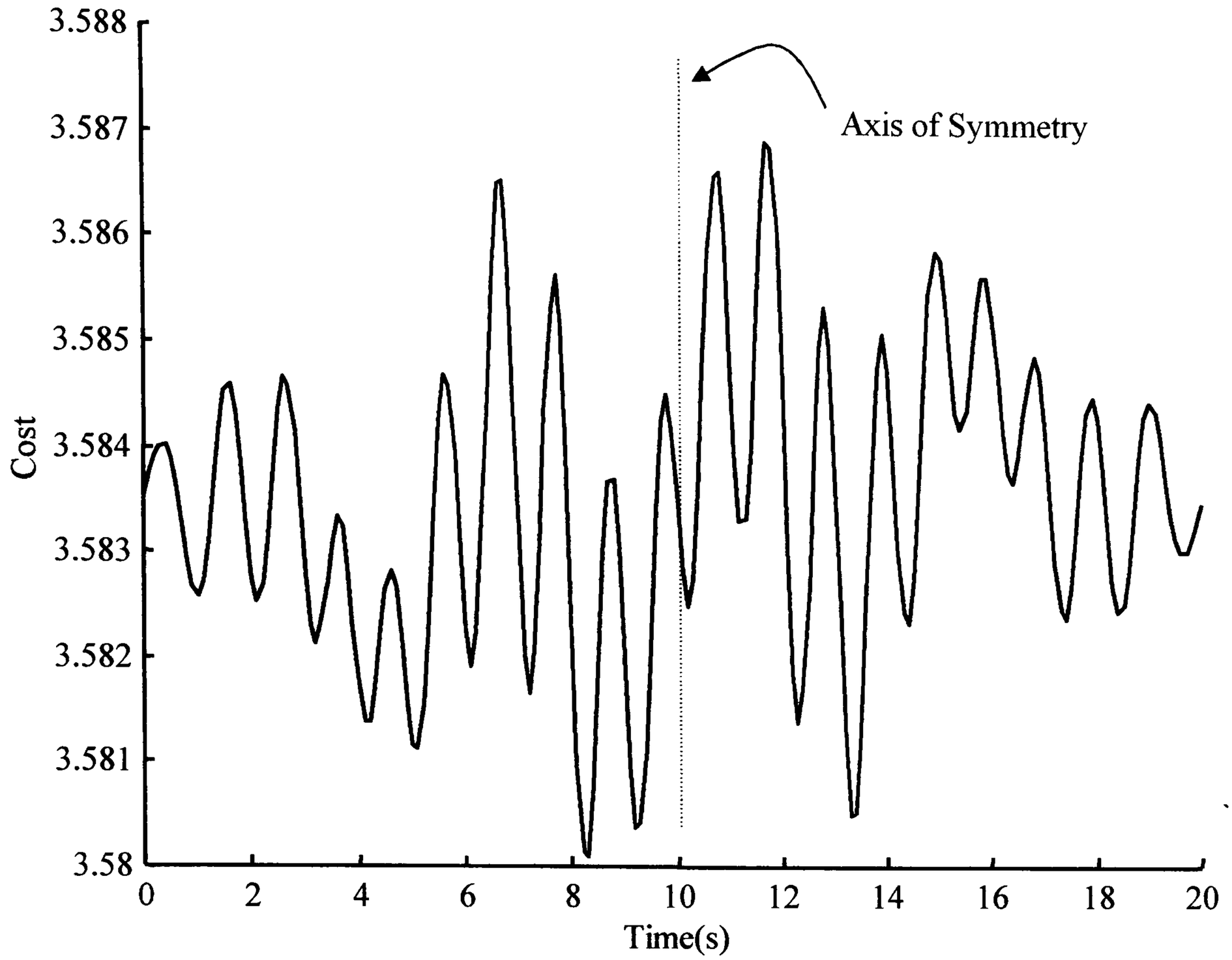


Figure 5.5: Time–varying Cost Function Showing Odd Symmetry.

From Figure 5.5 it can be seen that the axis of symmetry is about a vertical line drawn at the centre of the x-axis. It can be seen that only the first 10(s) of the time-varying cost function is required to be determined since symmetry can be employed to generate the remaining part of the cost function.

5.4.2 The Controller Parameter Cycling Algorithm

To create a new controller parameter cycling method, the optimisation Algorithm 1 is used with the estimates of the cost function gradient and Hessian from the theory of Propositions 1 and 2. In an application, the controller structure will be fixed a priori, and can be used to define the integer sets, $\mathfrak{I}_G(n_c)$, $\mathfrak{I}_H^{s\sim c}(n_c)$ or $\mathfrak{I}_G(n_c)$, $\mathfrak{I}_H^{c\sim c}(n_c)$.

The new algorithm finds estimates of both the gradient and the Hessian and uses these in a Newton algorithm to determine the next set of controller parameters whilst minimising the cost function. It is known that Newton type algorithms are not globally convergent and if the algorithm is initialised outside the region of convergence, the parameter updates are chosen to move in the direction of the convergence region. Similarly if the Hessian estimate is negative definite then this cannot be used and additional steps are used for the parameter updates prior to the Hessian estimate becoming positive definite. This has been discussed by Ljung (1987) and involves replacing the Hessian by, $H_{LM} = \frac{\partial J}{\partial \rho}\frac{\partial J}{\partial \rho}^T + \alpha I$ where $\alpha \in \Re^+$.

This is known as the Levenberg-Marquardt procedure. When the Hessian becomes positive definite, the Newton method reverts to using the Hessian estimate in the parameter updates. The algorithm is given as follows.

Algorithm 5.2: Optimisation by Controller Parameter Cycling

Step 1: Initialisation, $\rho \in \Re^{n_c}$

Choose cost weighting, λ, cost time interval, T_f

Set $T_0 = T_f$, compute $\omega_0 = 2\pi / T_0$

Choose perturbation sizes, $\{\delta_i, \quad i = 1,..,n_c\}$

Find sets, $\Im_G(n_c)$, $\Im_H^{s\sim c}(n_c)$, $\Im_G(n_c)$, $\Im_H^{c\sim c}(n_c)$ as appropriate

Determine N, set $T = T_0 / N$

Choose convergence tolerance, ε

Set loop counter, $k = 0$, choose $\rho(k)$

Step 2: Gradient and Hessian Calculation

Calculate gradient, $\frac{\partial J}{\partial \rho}(k)$ (Proposition 1)

And Hessian, $H_{ij}(k) = \frac{\partial^2 J}{\partial \rho_i \partial \rho_j}$, $H \in \Re^{n_c \times n_c}$ (Proposition 2)

If $\left\| \frac{\partial J}{\partial \rho}(k) \right\| < \varepsilon$ and $H(k) > 0 \in \Re^{n_c \times n_c}$ then stop

Step 3: Update Calculation

Select or calculate the update step size, γ_k.

If $H(k) > 0 \in \Re^{n_c \times n_c}$ compute $\rho(k+1) = \rho(k) - \gamma_k H(k)^{-1} \frac{\partial J}{\partial \rho}(k)$

Else compute $\rho(k+1) = \rho(k) - \left[\left[\frac{\partial J}{\partial \rho}(k) \right] \left[\frac{\partial J}{\partial \rho}(k) \right]^T + \alpha_k I \right]^{-1} \frac{\partial J}{\partial \rho}(k)$

Set $k = k+1$ and go to Step 2

Remarks

The implementation of Proposition 1 and 2 as numerical procedures involves exploiting the symmetry properties of the integrals to be evaluated and careful selection of the integration step size.

5.5 Application Results for the Controller Parameter Cycling Method.

An application of the method to a two input, two output system controlled by a decentralised PID controller system is reported in the sequel.

5.5.1 Multivariable Process - Algorithm Setup

The system to be controlled has the transfer function matrix (Zhuang and Atherton, 1994),

$$G(s) = \frac{1}{d(s)} \begin{bmatrix} 1.5s+1 & 0.15s+0.2 \\ 0.45s+0.6 & 0.96s+0.8 \end{bmatrix}$$

$$d(s) = 2s^4 + 8s^3 + 10.5s^2 + 5.5s + 1$$

This system is considered to be typical of those found in process industries (Zhuang and Atherton, 1994). The PID controllers to be tuned are of the parallel type and are given as,

$$K_{ii}(s) = K_{Pii}(s) + \frac{K_{Iii}(s)}{s} + K_{Dii}s$$

where $i = 1,2$.

Thus the number of parameters to be tuned, n_c, is six and the fixed structure controllers are to be tuned such that the LQ cost function to be minimised is given by

$$J = \int_0^{T_f} \left(e(t)^T e(t) + u(t)^T \Lambda u(t) \right) dt$$

Algorithm Set-Up.

In this case, unit weightings are given to the error terms, and equal weighting is given to each of the control terms using, $\Lambda = diag\{0.01,0.01\}$. The initial PID controller was derived using a relay experiment and Ziegler-Nichols rules. Throughout the tuning procedure the controller parameter perturbation size was, $\delta_i = \delta = 0.001$, $i = 1,...,6$. The cost function integration period was chosen to be $T_f = 20$, hence, $T_0 = T_f = 20$, and $\omega_0 = 0.1\pi$ (rad.s^{-1}). The choice of the gain perturbation frequencies cannot be made in isolation from the choice of the Hessian extraction frequencies. The gain perturbation frequencies must be chosen such that the orthogonality properties of the sinusoids used can be exploited in the extraction of the gradient and Hessian data. Algorithm 5.3 details the steps required in deriving a set of gain and Hessian extraction frequencies for a sinusoidal gain perturbation, whilst Algorithm 5.4 gives similar details for a cosinusoidal gain perturbation.

Algorithm 5.3: Selection of Frequencies for Hessian Element Extraction with Sinusoidal Gain Perturbation.

Step 1: Initialisation

Choose a set of gain perturbation integers

$$\Im_G(n_c) = \{I_1, I_2, \cdots, I_{n_c}\}$$

ordered such that

$$I_1 < I_2 < \cdots < I_{n_c}$$

and n_c is the number of controller parameters.

S_Flag = 0

D_Flag = 0

Step 2: Generation of Sum Terms

Construct an n_c x n_c matrix, S, such that

$$S(i,j) = I_i + I_j$$

where $i = 1,2,\cdots,n_c;\ j = 1,2,\cdots,n_c$

Step 3: Generation of Difference Terms

Construct an n_c x n_c matrix, D, such that

$$D(i,j) = |I_i - I_j|$$

where $i = 1,2,\cdots,n_c;\ j = 1,2,\cdots,n_c$

Step 4: Test for Extraction Frequency Conflict

Step 4a

For each of the leading diagonal elements of S,

$$\text{If } S(i,i) = S(i,j) \text{ then S\_Flag = 1}$$

where $i = 1,2,\cdots,n_c;\ j = 1,2,\cdots,n_c$ and $i \neq j$

$$\text{If } S(i,i) = D(i,j) \text{ then S\_Flag = 1}$$

where $i = 1,2,\cdots,n_c;\ j = 1,2,\cdots,n_c$ and $i \neq j$

If S_Flag = 1 then go to Step 1

Else record the frequency given by $S(i,i)$

Step 4b

For each of the purely upper triangular elements of S,

$$\text{If } S(i,j) = S(i_1,j_1)$$

$$\text{Then If } D(i,j) \neq S(i_1,j_1)$$

$$\text{Then D\_Flag = 1}$$

$$\text{Else (S\_Flag = 1 and D\_Flag = 0)}$$

where $i = 1,2,\cdots,n_c;\ j = 1,2,\cdots,n_c;\ i_1 = 1,2,\cdots,n_c;\ j_1 = 1,2,\cdots,n_c$

If S_Flag = 1 then go to Step 1

Else record the frequency given by $S(i,j)$

If D_Flag = 1 then record the frequency given by $S(i,j)$

Remarks

The generation of the gain perturbation frequencies can be carried out using nested *For Loops* in a coded version of the algorithm.

The algorithm can generate different sets of gain perturbation and Hessian extraction frequencies.

It can be seen from Algorithm 5.3 that the selection of the gain perturbation frequencies and the frequencies that are used to extract the Hessian elements cannot be considered in isolation.

Algorithm 5.4: Selection of Frequencies for Hessian Element Extraction with Cosinusoidal Gain Perturbation.

Step 1: Initialisation

Choose a set of gain perturbation integers

$$\Im_G(n_c) = \{I_1, I_2, \cdots, I_{n_c}\}$$

ordered such that

$$I_1 < I_2 < \cdots < I_{n_c}$$

and n_c is the number of controller parameters.

D_Flag = 0

P_Flag = 0

S_Flag = 0

Step 2: Generation of Sum Terms

Construct an n_c x n_c matrix, S, such that

$$S(i,j) = I_i + I_j$$

where $i = 1,2,\cdots,n_c$; $j = 1,2,\cdots,n_c$

Step 3: Generation of Difference Terms

Construct an n_c x n_c matrix, D, such that

$$D(i,j) = |I_i - I_j|$$

where $i = 1,2,\cdots,n_c$; $j = 1,2,\cdots,n_c$

Step 4: ***Test for Extraction Frequency Conflict - Gradient***

For each of the gain perturbation frequencies $\Im_G(n_c)$,

$$\text{If } I_i = S(i,j) \text{ or } I_i = D(i,j) \text{ then P\_Flag} = 1$$

where $i = 1,2,\cdots,n_c;\ j = 1,2,\cdots,n_c$

If P_Flag = 1 then got to STEP 1

Step 5: Test for Extraction Frequency Conflict - Hessian

Step 5a

For each of the leading diagonal elements of S,

$$\text{If } S(i,i) = S(i,j) \text{ then S\_Flag} = 1$$

where $i = 1,2,\cdots,n_c;\ j = 1,2,\cdots,n_c$ and $i \neq j$

$$\text{If } S(i,i) = D(i,j) \text{ then S\_Flag} = 1$$

where $i = 1,2,\cdots,n_c;\ j = 1,2,\cdots,n_c$ and $i \neq j$

If S_Flag = 1 then go to Step 1

Else record the frequency given by $S(i,i)$

Step 5b

For each of the purely upper triangular elements of S,

$$\text{If } S(i,j) = S(i_1,j_1)$$

$$\text{Then If } D(i,j) \neq S(i_1,j_1)$$

$$\text{Then D\_Flag} = 1$$

$$\text{Else (S\_Flag} = 1 \text{ and D\_Flag} = 0)$$

where $i = 1,2,\cdots,n_c;\ j = 1,2,\cdots,n_c;\ i_1 = 1,2,\cdots,n_c;\ j_1 = 1,2,\cdots,n_c$

If S_Flag = 1 then go to Step 1

Else record the frequency given by $S(i,j)$

If D_Flag = 1 then record the frequency given by $S(i,j)$ ●

Algorithm 5.3 was used to develop the Hessian extraction frequencies for a sinusoidal gain perturbation using the integer set, $\Im_G(n_c) = \{1,4,5,16,19,20\}$. The results from the use of the algorithm are given in Table 5.2.

Table 5.2: Hessian Extraction Frequency Integers

$\mathfrak{I}_G(n_c)$	$n_1=1$	$n_2=4$	$n_3=5$	$n_4=16$	$n_5=19$	$n_6=20$
$\mathfrak{I}_H^{S\sim C}(n_c)$	$n_{11}=2$	$n_{12}=5$	$n_{13}=6$	$n_{14}=17$	$n_{15}=18$	$n_{16}=19$
	-	$n_{22}=8$	$n_{23}=9$	$n_{24}=12$	$n_{25}=23$	$n_{26}=16$
	-	-	$n_{33}=10$	$n_{34}=11$	$n_{35}=14$	$n_{36}=25$
	-	-	-	$n_{44}=32$	$n_{45}=35$	$n_{46}=36$
	-	-	-	-	$n_{55}=38$	$n_{56}=39$
	-	-	-	-	-	$n_{66}=40$

5.5.2 Algorithm Results.

The simulation was carried out using Matlab/Simulink™. The gain perturbation and the calculation of the cost function was implemented using Simulink™, while the estimation of the gradient, Hessian and updated controller parameters was carried out using Matlab™. Sinusoidal perturbation of the controller parameters was used in this example. Figure 5.7 shows the evolution of the cost function over the period

$$t_\Delta = T_0$$

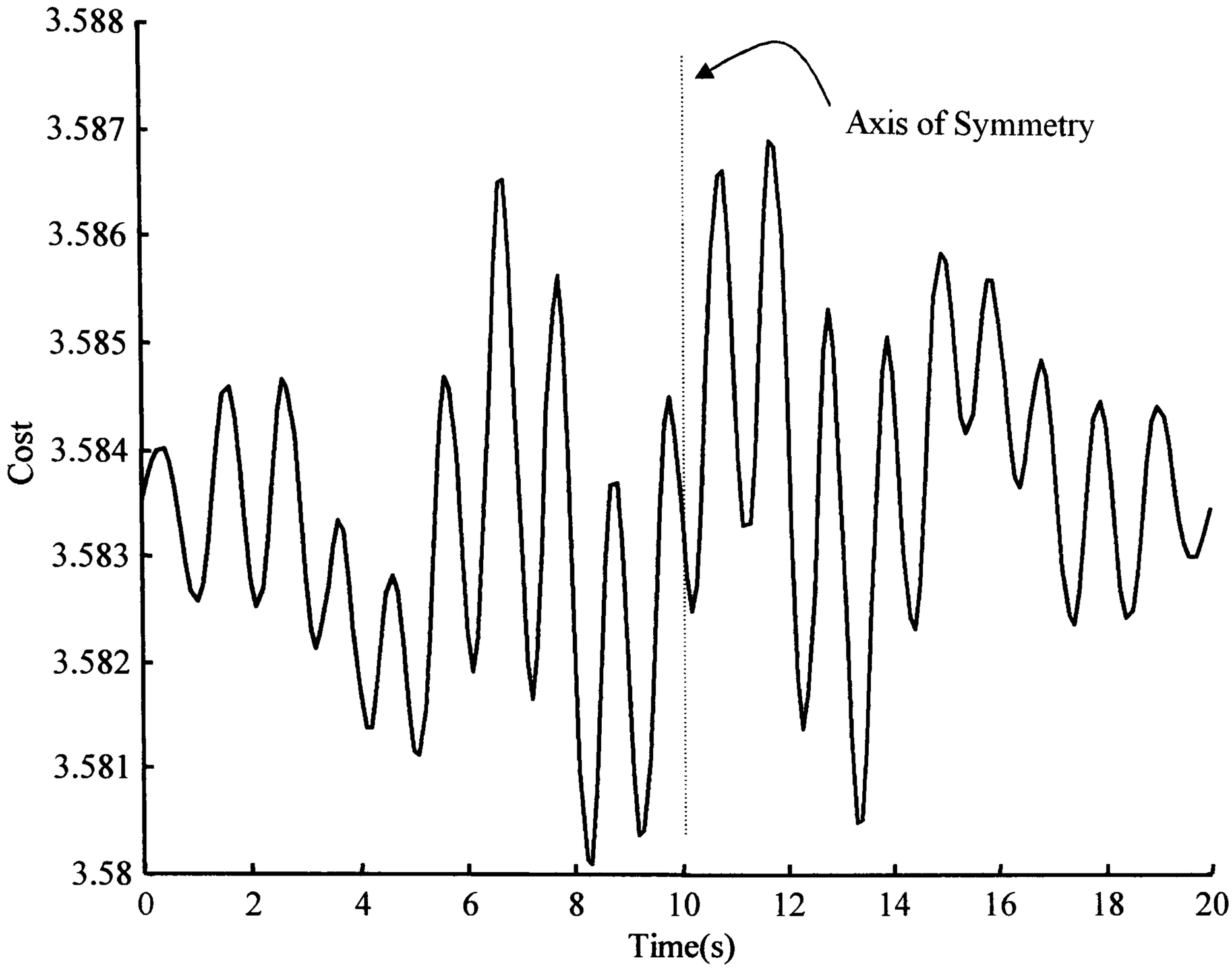


Figure 5.7: Evolution of Cost $J(\rho + \Delta\rho(t_\Delta))$

As can be seen from Figure 5.7 the cost exhibits odd symmetry as predicted since sinusoidal excitation of the controller parameters was used. Figure 5.7 was generated from the second iteration of Algorithm 5.2. The graphs of the cost function for subsequent iterations of the algorithm all have a similar form and show odd symmetry.

Figures 5.6 and 5.7 below show the 2-norm of the cost function gradient and the cost function values for the algorithm iterations.

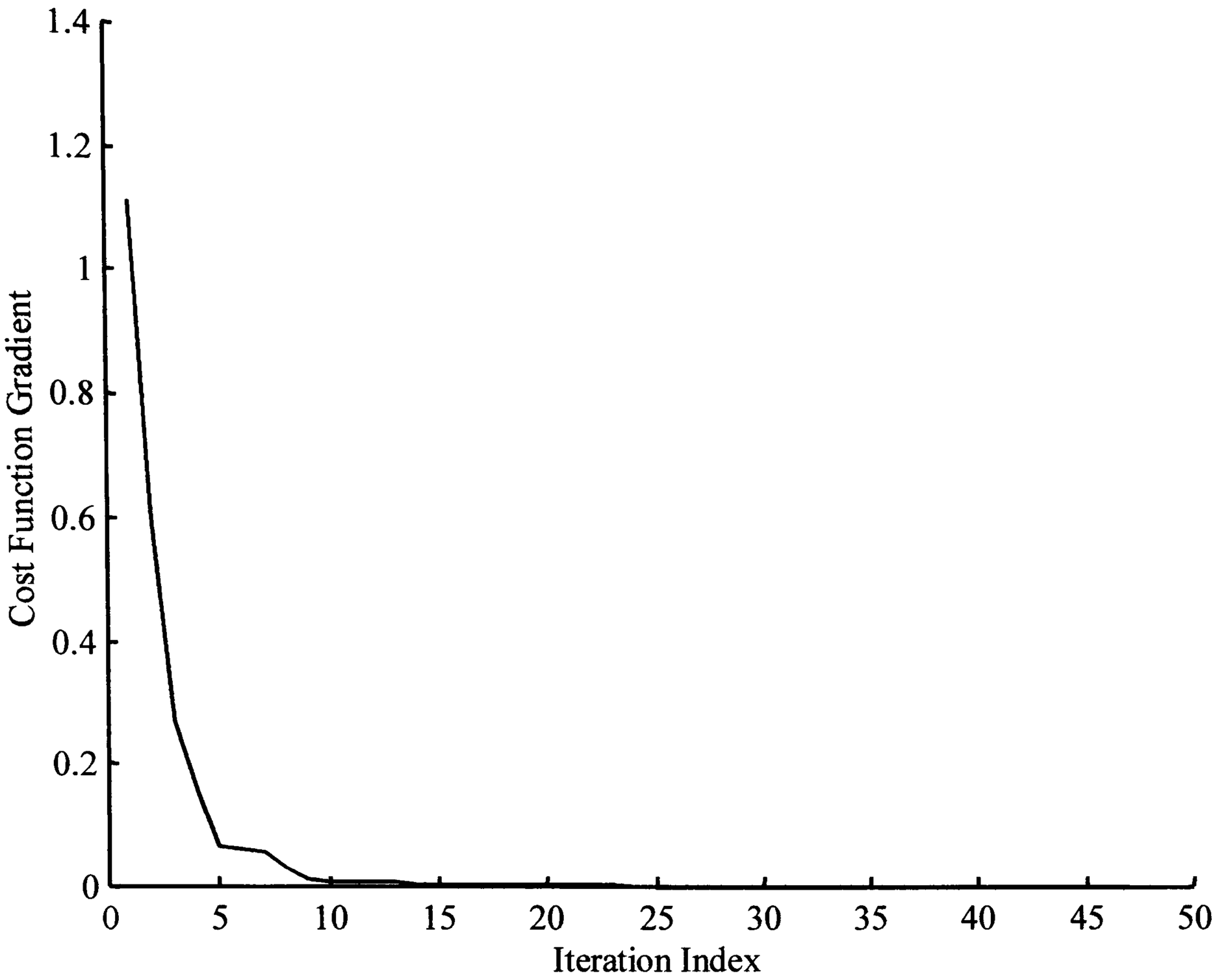


Figure 5.8: Evolution of the 2-Norm of the Cost Function Gradient.

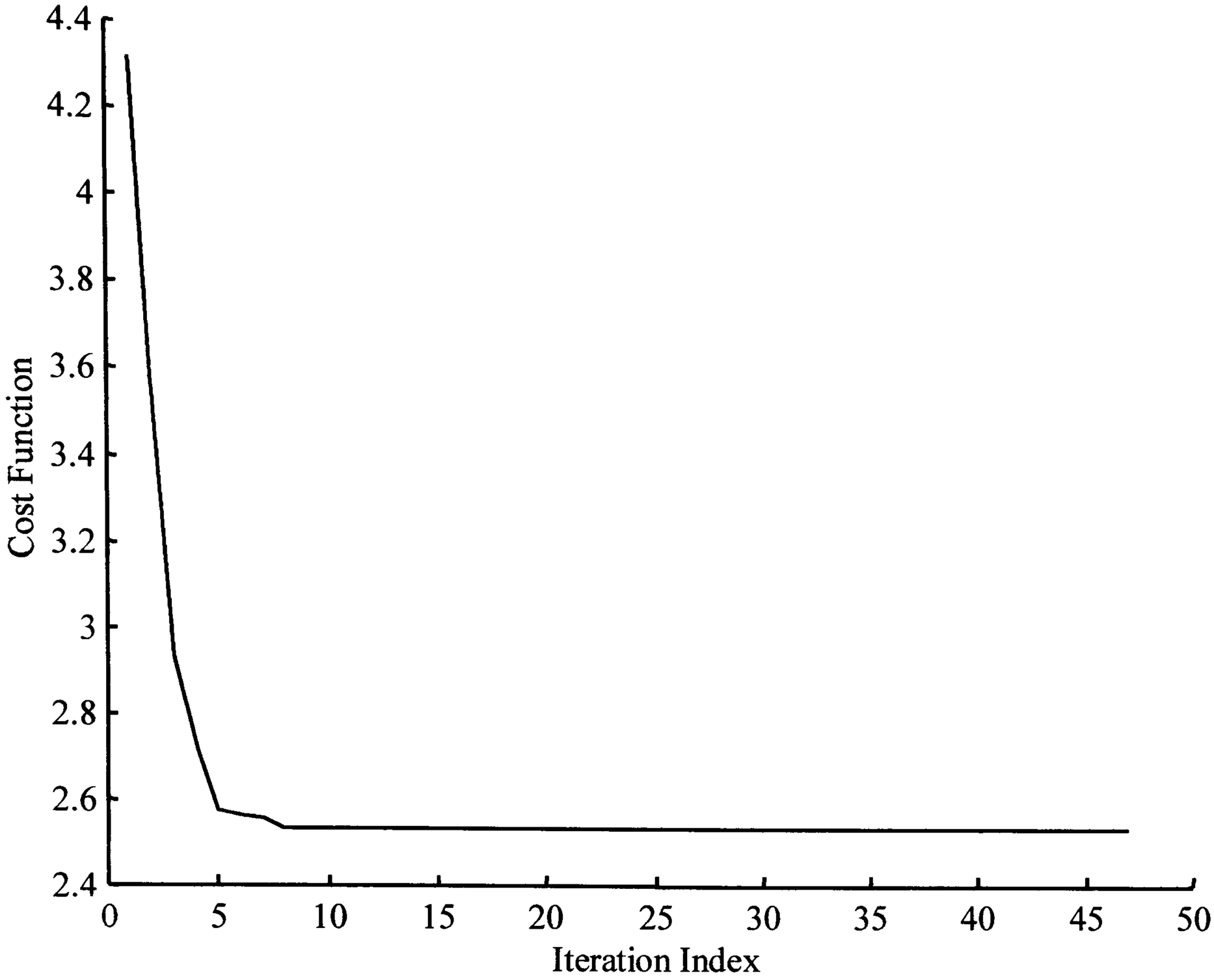


Figure 5.9: Evolution of the Cost Function.

For the first three iterations of the algorithm the Hessian estimate was negative definite and the Hessian was replaced by the Levenberg-Marquardt procedure with, $\alpha I = 0.01 I_6$. At the fourth iteration it was found that the algorithm returned negative values for the updated integral gain parameters for both controllers. Consequently the algorithm was adjusted using $\alpha I = 0.1 I_6$. After two further iterations the Hessian estimate was positive definite and was used in the Newton update. Since the Hessian estimate was corrupted by noise its use had to be conservative. To do this the controller parameter update size was limited using, $\gamma_k \in (0,1)$. For the next two iterations, $\gamma_k = 0.1$ and the values of the 2-norm of the cost function gradient and cost function value decreased. For iterations 10 and 11, $\gamma_k = 1$. After iteration 11, the cost function value and gradient 2-norm increased. The step size was reset at $\gamma_k = 0.1$ and a new iteration 11 performed where the cost function value and gradient 2-norm decreased; the step size $\gamma_k = 0.1$ was retained for the remaining iterations. From

Figure 5.8 and 5.7 the evolution of the cost function gradient 2-norm shows that the minimum of the cost function is found after approximately 15 iterations of the algorithm when the cost function gradient 2-norm is $O(10^{-3})$. Thus, the controller parameters at iteration 15 give the minimum value of the cost function. In order to ensure that further iterations of the algorithm do attain a minimum, a large number of additional iterations were performed. It can be seen from Figures 5.6 and 5.7 that the cost function values tend to an asymptotic value and that the cost function gradient 2-norm limits at zero. The evolution of the controller parameters are shown in Figures 5.10 to 5.12 inclusive.

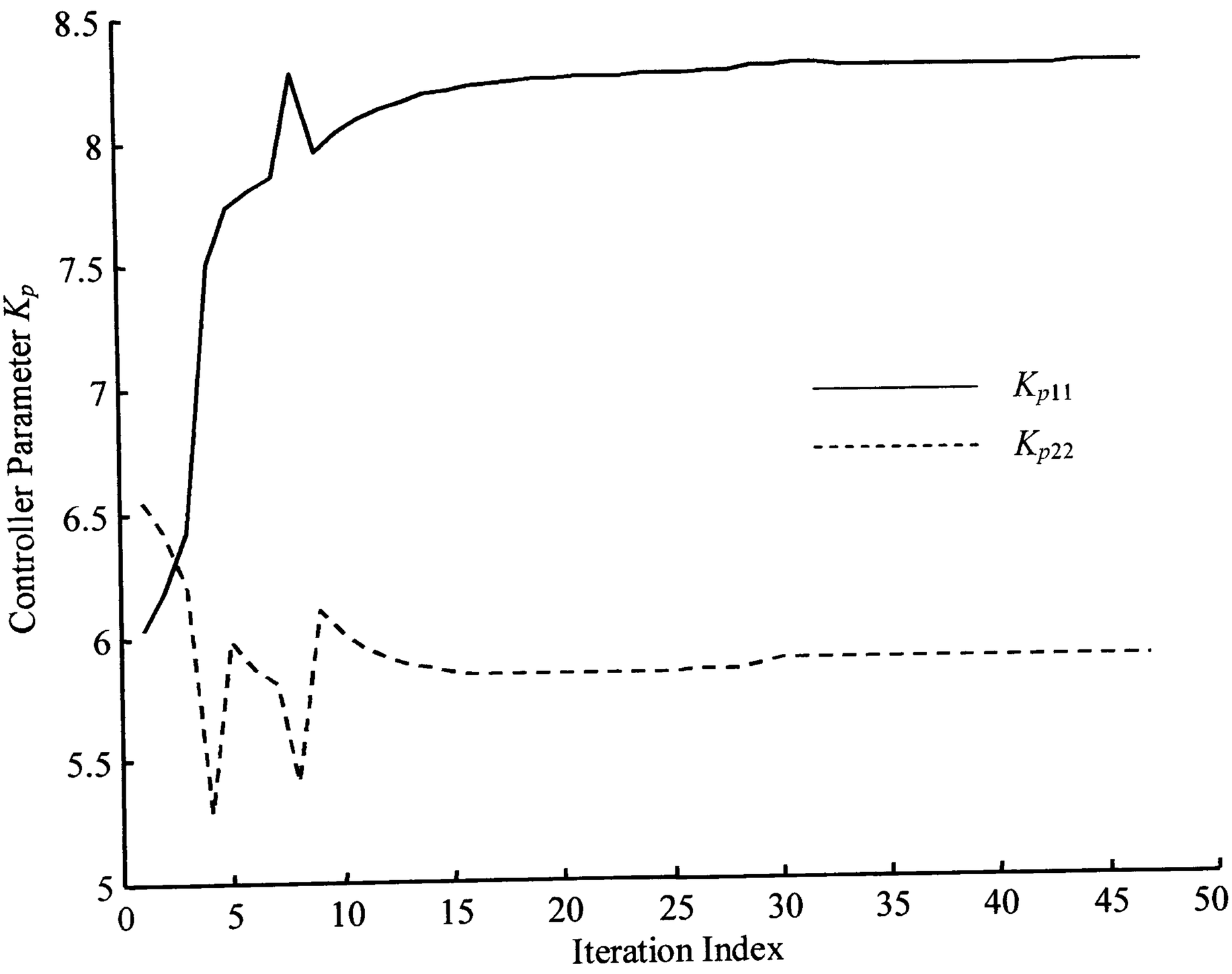


Figure 5.10: Evolution of the Controller Parameter K_p

As can be seen from Figure 5.10 the controller parameter K_{p22} tends to wander during the first ten or so iterations of the algorithm. However, this did not cause any problems with the stability of the closed loop system and as can be seen after

approximately ten iterations of the algorithm the movement in value of K_{p22} became less pronounced.

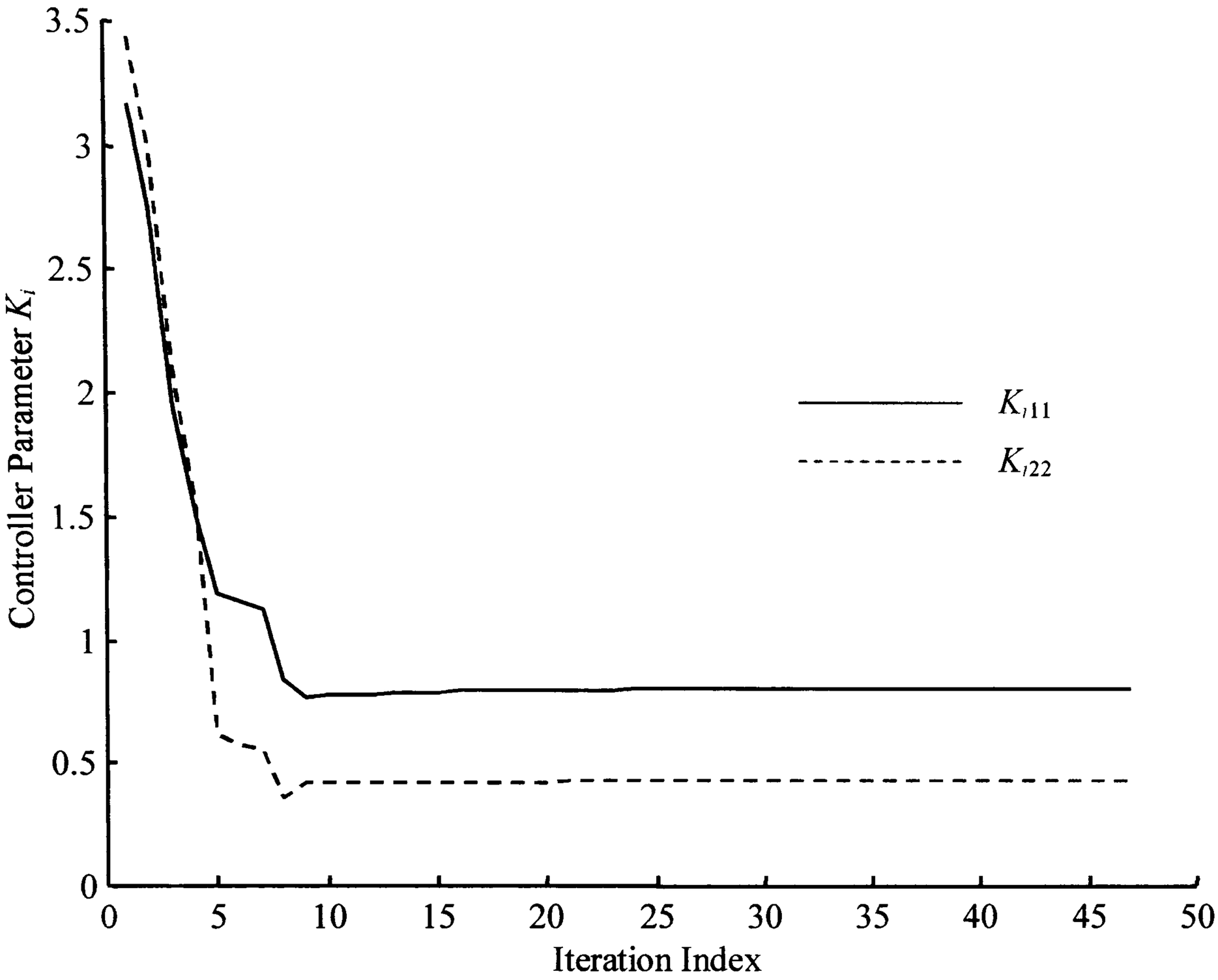


Figure 5.11: Evolution of the Controller Parameter K_i

From Figure 5.11 it can be seen that for both controllers the evolution of the integral gain, K_i, is relatively smooth. After an initial rapid reduction in the value of the integral gain for both controllers, the rate of change of K_i fell to a low value approximately after iteration ten of the algorithm.

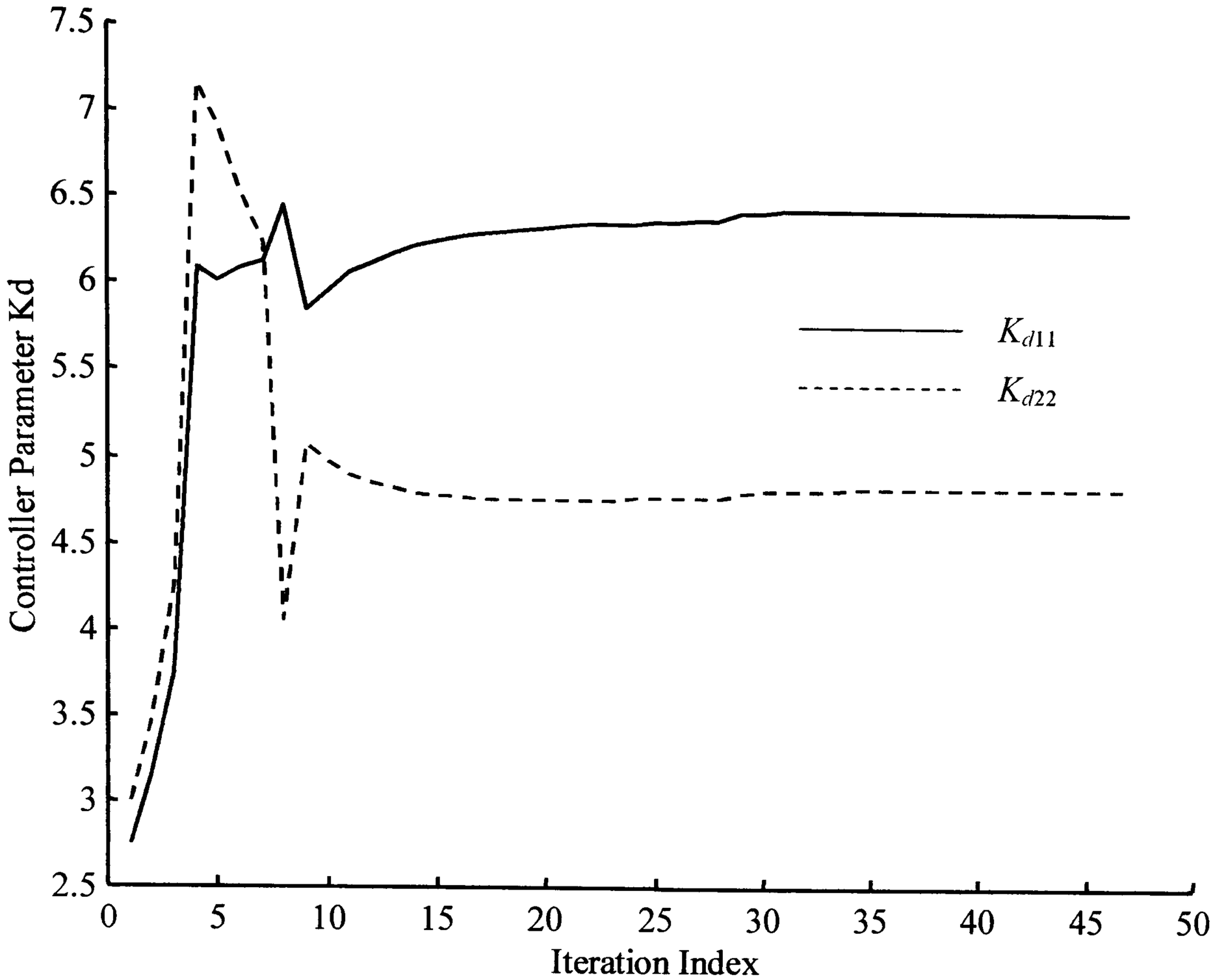


Figure 5.12: Evolution of the Controller Parameter K_d

From Figure 5.12 it can be seen that there is an initial rapid change in the values of the controller derivative gains. The controller parameter K_{d22} has a larger variation in its value than does K_{d11}, however the stability of the closed loop system was unaffected by the variations in the controller parameters during the course of the tuning algorithm. As can be seen from Figure 5.12 after approximately ten iterations of the algorithm the rate of change of the controller derivative gain parameters fell to a low value.

From Figures 5.8 and 5.9 it can be seen that the 2-Norm of the gradient of the cost function tends to zero and that the value of the cost function tends to an asymptote as the number of algorithm iterations increases, however the time response of the final version of the decentralised controller cannot be inferred from the cost function or gradient values. The block diagram of the decentralised control system used in the assessment of the initial and final control system performance is shown in Figure 5.13.

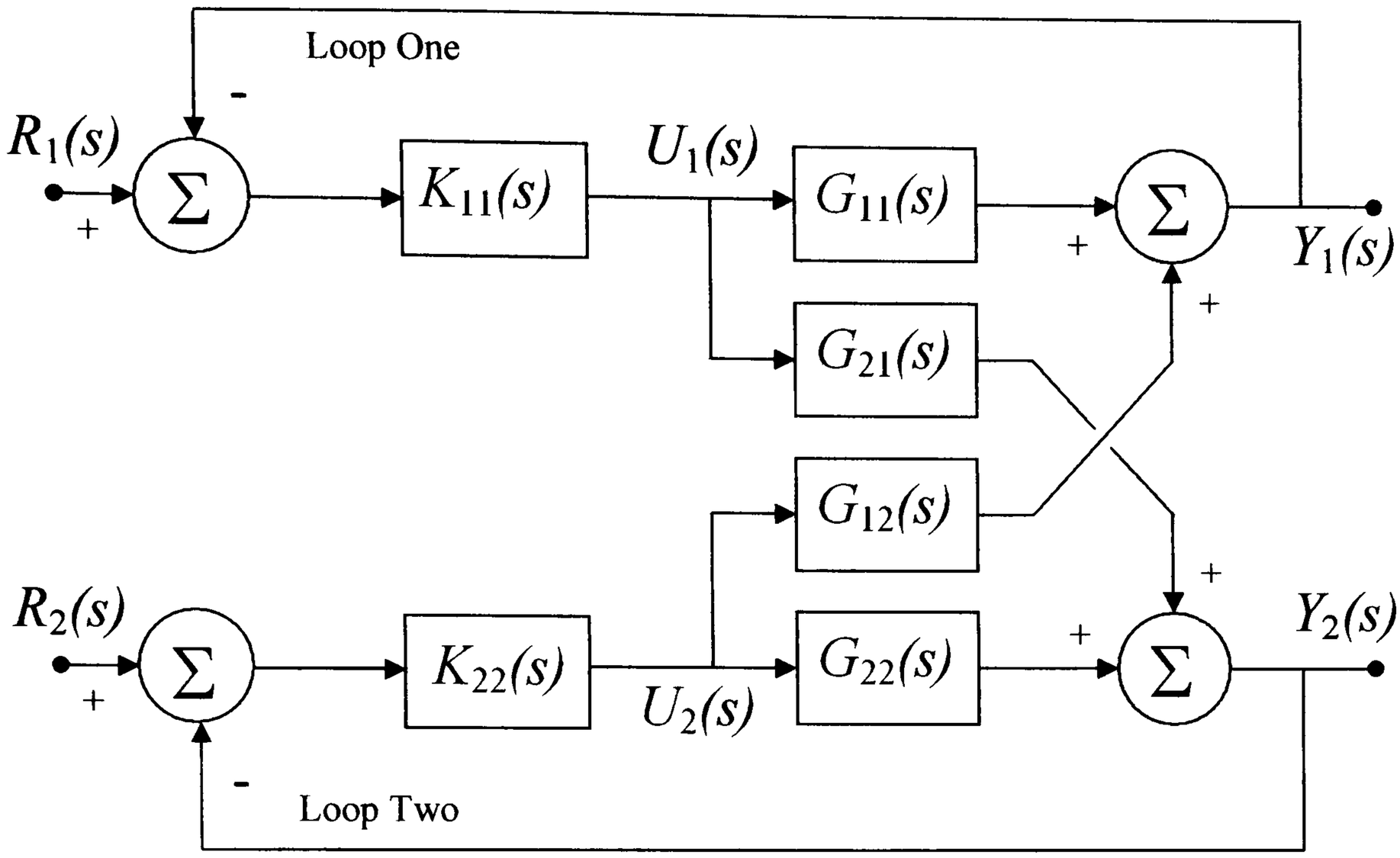


Figure 5.13: Decentralised Control System

A step input was applied to the reference input of Loop One at time t = 0(s) and a step input was applied to the reference input of Loop Two at time t = 100(s). Figure 5.14 compares the closed loop step response for the output of loop one of the system, with the controller parameters set to the initial Z-N values and the final Controller Parameter Cycling method tuned values. As can be seen from Figure 5.14 the Ziegler-Nichols tuned controller gives a peak percentage overshoot of approximately 50% compared with no overshoot from the Controller Parameter Cycling method tuned controller to the initial step input. Both controllers achieve similar rise times. The settling times achieved for the controllers are, based on a ±2% criterion, 12.5(s) for the Z-N controller and 4.5(s) for the Controller Parameter Cycling tuned controller. The Integral of the Square reference Error (ISE) was calculated during the simulation for both the Z-N tuned controllers and the Controller Parameter Cycling (CPC) tuned controllers. For the initial step input applied to the reference input of Loop One the Z-N tuned controllers had an ISE of 1.251. The corresponding value for the system using the CPC tuned controllers was 0.9012 for Loop One.

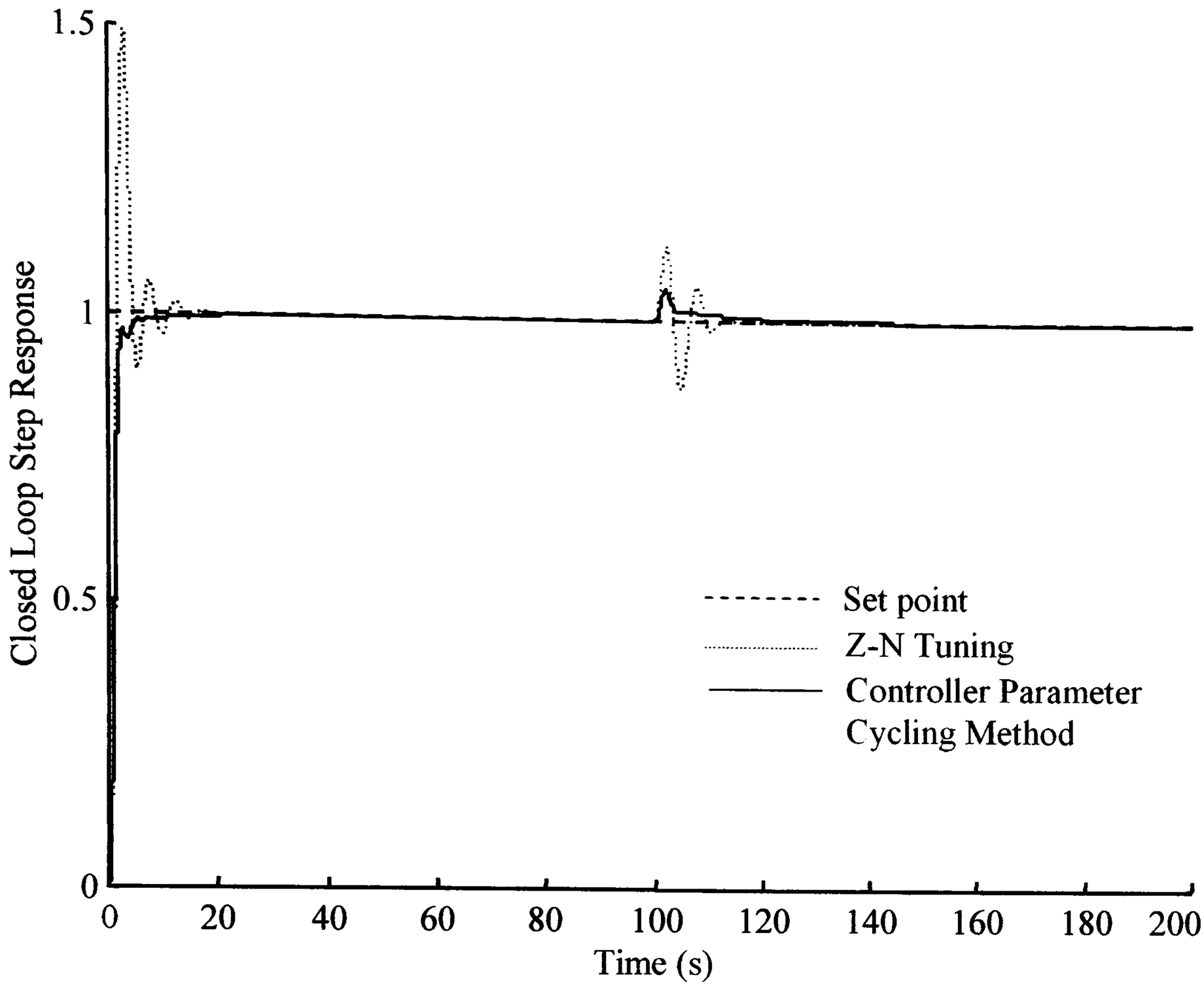


Figure 5.14: Closed Loop Response for Loop One.

When the step input is applied to loop two the effect on loop one is much reduced when the CPC tuned controllers are used compared with the response when the Z-N tuned controllers are used. The Z-N tuned controllers give a percentage overshoot of 13% and a percentage undershoot of 12%, this can be compared with a percentage overshoot of 5% and no undershoot for the CPC tuned controllers. The settling time for the Z-N tuned controllers is approximately 11(s) compared with 3.5(s) for the CPC tuned controllers. The settling time was based on a ±2% criterion. The ISE figure for Loop One when a step input is applied to Loop Two is 0.05 for the Z-N tuned controllers compared with a value of 0.0072 when the CPC tuned controllers are used. Figure 5.15 shows the step response for the closed loop output of Loop Two when a step input is applied to the reference input of Loop One at time t = 0(s) and a step input is applied to the reference input of Loop two at time t = 100(s). From Figure 5.15 it can be seen that when a step input is applied to the reference input of Loop One that the interaction into Loop Two has a peak overshoot of 35.7% and a

peak undershoot of 33.5% for the Z-N tuned controllers compared with a peak percentage overshoot of 24% for the CPC tuned controllers.

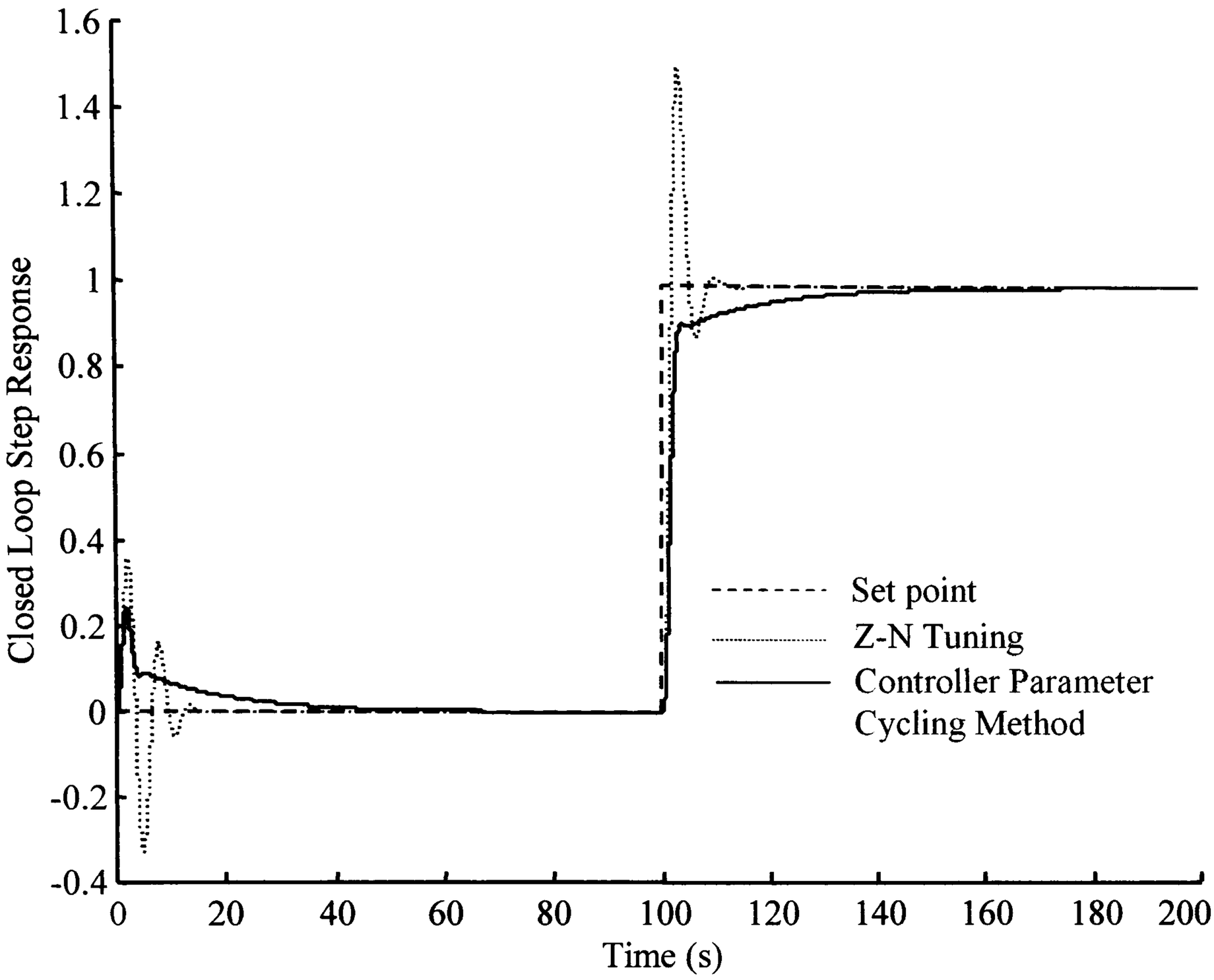


Figure 5.15: Closed Loop Response for Loop Two.

The settling time for the interaction disturbance is 12(s) for the Z-N tuned controllers and 28(s) for the CPC based controllers. The ISE term for the Z-N and CPC tuned controllers are respectively 0.3794 and 0.1663. Thus it can be seen that although the Z-N tuned controllers give a faster response than the CPC tuned controllers the disturbance rejection properties of the CPC tuned controllers are improved over the original Z-N tuned controllers. When the step input is applied to the reference input of Loop Two then from Figure 5.15 it can be seen that the peak percentage overshoot of the output of Loop Two using the Z-N tuned controller is approximately 51% compared with 24% for the corresponding figure using the CPC tuned controllers. The settling time for the closed loop system using the Z-N tuned Controllers is 5(s) compared with 30(s) for the system using the CPC tuned controllers. The settling

time was calculated using a ±2% criterion. The rise time of the closed loop output of Loop Two when the Z-N controllers are used is 2(s) compared with 3(s) for the system when the CPC tuned controllers are used. The rise time was based on the time taken for the output of Loop Two to go from 10% to 90% of its final value. The ISE for the step input applied at time t = 100(s) is 1.4686 for the system using the Z-N tuned controllers and 1.3447 for the system using the CPC based controllers. Thus although the Z-N tuned controllers give a faster response than is achieved using the CPC based controllers the CPC based controllers give an improved control performance as regards overshoot and ISE error reduction.

5.6 Summary Conclusions.

The method of Iterative Feedback Tuning has certain practical shortcomings; a particular one is that the generation of Hessian information is not a simple operation. In this chapter a new model-free procedure where the theory for generating the gradient and second order information arises from one unified theoretical framework was reported. The new method is termed Controller Parameter Cycling.

In this chapter the development of a first version of a numerical algorithm for the new procedure was reported. Whilst the procedure was successful in tuning a multivariable decentralised PID controller, certain issues remain to be investigated further:

(a) Experience with several examples has shown that many of the cost functions of fixed structure controllers are often very flat and it is necessary to ensure that sensible parameter updates are used in the routine. In the current version of the algorithm, the Levenberg-Marquardt procedure is used to try to ensure that successive parameter updates continue in the cost minimising direction; it would be useful to examine other procedures.

The implementation of the gradient and Hessian extraction involves a heavy computational burden in terms of system runs and minimising iteration steps. Symmetry can be exploited to reduce the computational load and an intelligent strategy can be used to ensure the algorithm uses computed data to maximum effect in the sequence of iteration steps.

6 Conclusions

6.1 Identification Methods for Process Control Applications

A literature review was presented in Chapter 1 which showed that Astrom and Hagglund (1984) proposed a very simple and elegant closed loop experiment to compute the ultimate period method data for use in PID rule based design methods. In the Astrom and Hagglund method the PID controller is replaced by a relay, the period and magnitude of the resulting limit cycle is then used to calculate the parameters of a PID controller based on the rules supplied by Ziegler and Nichols. This represents an advance over the Ziegler and Nichols ultimate period method since the experiment is carried out in closed loop and a stable limit cycle is achieved for the majority of processes that are found in process industries. The Astrom and Hagglund relay experiment coupled with the availability of microprocessor based PID controllers effectively paved the way for the introduction of push-button auto-tuning.

The literature survey proceeded to show how the simple relay experiment had been modified to overcome accuracy and implementation problems. However, the new relay configurations that are required have begun to lose the original simplicity of the Astrom and Hagglund (1984) relay experiment. Furthermore, if the published literature were to show that there was a widespread industrial acceptance of the new more complex relay experiment methods then this would vindicate the case for the added complexity however this does not appear to be the situation.

It was this hiatus in the research associated with relay experiments that lead to the proposal of the Phase-Locked Loop (PLL) method of system identification The key features of the relay experiment are that it:

i) Is easy to implement

ii) Is carried out in closed loop

iii) Returns relatively accurate results, and

iv) Does not have extended identification times.

The initial goal of the research that produced the PLL method of system identification was *to carry out the relay experiment without using a relay.* Thus the intention was to identify the point on the frequency response curve of a process at which the phase angle was $-\pi$ (rad). This point is known as the phase crossover point and the frequency at which it occurs as the phase crossover frequency, $\omega_{-\pi}$. The result of the research was the PLL identifier (Crowe and Johnson, 1998; Johnson and Crowe, 1998; and Crowe 1998). The PLL method of system identification encompasses the key features of the relay experiment. A presentation of the basic Phase Locked Loop identifier theory was presented in Chapter 1.

Chapter 1 also gave a discussion of alternative versions of the PLL method of system identification. Balestrino *et al* (2000) use the relay experiment to obtain an estimate of the phase crossover frequency to initialise the Voltage Controlled Oscillator (VCO) frequency prior to carrying out the identification using the PLL Identifier. Clarke and Park (2003) have discussed a continuous time implementation of the PLL identification method. The contribution by Clarke and Park was to carry out a comparison between possible phase detection methods. The results of this comparison show that the Hilbert transform method of phase detection gives the best performance with regard to noise rejection over zero crossing and synchronous demodulation techniques. Some features of the PLL method were investigated in Chapter 1. In particular an extension to the basic PLL method was developed allowing the identification of type 1 systems in open loop. This was presented in Section 1.3.

Recently there has been a trend to use data driven techniques that do not require an explicit model of the process to be produced for a subsequent controller design. In subspace identification input and output data from the process is used to provide an implicit model of the process in terms of a state space representation that can be used directly to provide a controller design. This technique and the associated technical literature were described in section 1.4 of Chapter 1 as a transitional stage from explicit to implicit to model-free methods of controller design techniques.

A literature review of the model-free Iterative Feedback Tuning (IFT) method was presented in Section 1.5 since the thesis made a major contribution to this research through the controller parameter cycling method presented in Chapter 5.

6.2 Testing for the Existence of PID Controllers that can Achieve Specified Classical Robustness Measures.

In Chapter 2, the Phase-Locked Loop (PLL) method of system identification was used to identify a number of points on the frequency response curve of the process. The data found from the identification was then used to determine the viable gain and phase margin pairs that are achievable over a given range of gain and phase margin values. The results from the application of the method are then presented in a graphical form that shows the gain and phase margin pairings that can be achieved by a PI controller acting as the compensation element for a particular process. This method has two benefits:

i) The viable gain and phase margin pairings are shown in a form that is easily used, and

ii) A set of controller parameters relating to the gain and phase margin pairs is also generated.

Thus the method allows the control system designer the freedom to choose a viable gain and phase margin pair and gives the assurance that the design specification can be met. This was the starting point for further research into this type of PID existence problem. The main achievements presented in the Chapter can be listed under the following topics.

Graphical Design of PI Controllers: In Section 2.2.1 of the thesis, a method original to the author was discussed that allows the graphical representation of the viable gain and phase margin pairings that can be achieved by a PI controller acting on an unknown process. Additionally, the PI controller parameters to achieve the gain and phase margin pairings are given. A new Viable Gain Margin and Phase Margin Pairing algorithm was described in Section 2.2.1 and demonstrated successfully on two candidate processes;

one process having a right half plane zero and the other being high order, non-oscillatory.

Enumeration Method for PID Controller Design: A new enumeration method was described in Section 2.3.1 allowing a semi-graphical approach to be used to determine the parameters of a PID controller that meets a gain and phase margin design specification. A theorem underpinning the enumeration method was proposed and a proof of the theorem was presented in Section 2.3.1. The Enumeration Search Method algorithm developed in Section 2.3.1 was demonstrated successfully, in Section 2.3.2, on two candidate processes; one process representing a large class of processes to be found in process industries and the other being an oscillatory process.

Semi-Graphical Design of PID Controllers: In Section 2.3.3, a new semi-graphical technique was developed by the author to allow the PID controller parameter, k_p, to be related to a range of gain and phase crossover frequencies at which, using a set of derived equations, the remaining controller parameters can be calculated to achieve a design gain and phase margin. The new algorithm for the Graphical Design of PID Controllers developed in Section 2.3.3 was successfully demonstrated using the same candidate processes as were used in section 2.3.2.

6.3 The design of PID Controllers to meet Classical Robustness Measures.

In Chapter 3 the problem of using the Phase Locked Loop identifier online to design PI controllers to meet specifications on (i) gain margin and phase margin or (ii) maximum sensitivity and phase margin was solved. The iterative method generated a sequence of PI controller parameters however, the original PI controller parameters are not updated. Use is made of the flexibility of the PLL identifier to identify the frequency response of the new controller in series with the existing process at the gain crossover and the phase crossover points. The data supplied from these identifications are supplied to a least squares algorithm that is used to calculate the next iterate of the PI controller parameters. The technique continues in this way until design requirements on gain margin and phase margin have been met. The method allows the control system

to remain in closed loop throughout the design process. The method also has the added advantage that the PI controller parameters are updated only at the end of the design. The second design method discussed in Chapter 3 is that of maximum sensitivity and phase margin design of PI controllers. The design algorithm that is employed for the maximum sensitivity and phase margin design is similar to that used for the gain margin and phase margin design method. In the maximum sensitivity and gain margin method the flexibility of the PLL method of system identification is utilised such that the points at which the frequency response curve of the compensated forward path, comprising the newly designed PI controller and the existing process intersects the $1/M_s$ circle are identified. The algorithm is constructed to ensure that the frequency response of the compensated forward path always intersects the $1/M_s$ circle at two points such that a bound is maintained on the tangency angle, θ_s. In addition to the maximum sensitivity identifications the PLL is used to identify the compensated forward path at the gain crossover point. The data supplied by these identifications is supplied to a least squares routine that calculates an updated value for the new PI controller parameters. The algorithm continues in this way until the PI controller parameters are such that the design maximum sensitivity and phase margin design specifications have been met. As with the gain margin and phase margin PI design method the control system is in closed loop throughout the design and the controller parameters are only updated at the end of the design. The major achievements presented in Chapter 3 are for the following topics.

Gain and Phase Margin Design: An automated PI controller design method was developed in Section 3.3 that allows the iterative determination of the parameters of a PI controller to achieve gain and phase margin specifications. The Gain Margin and Phase Margin: Automated PI Controller Design algorithm was demonstrated successfully in Section 3.3.2 on two candidate processes that are representative of those found in process industries. A theorem and its proof relating to the convergence of the algorithm were developed in Section 3.3.1.

Maximum Sensitivity and Phase Margin Design: In Section 3.4 an automated PI controller design method was developed that allows the iterative calculation of the parameters of a PI controller to achieve maximum sensitivity and phase margin

specifications. The Automated PI Controller Design for Desired Maximum Sensitivity and Phase Margin Specifications algorithm was demonstrated successfully in Section 3.4.2 on two candidate processes that are representative of those found in process industries. A theorem and its proof relating to the convergence of the algorithm were developed in Section 3.4.1.

6.4 Closed Loop Identification and Tuning of Cascade and Multi-Input Multi-Output Control Systems.

In process industries a common control paradigm is that of cascade control. A cascade control system is characterised as having an inner or slave control loop and an outer or master control loop. The relay method of Astrom and Hagglund (1984) has been applied to tuning of a cascade connected control system by Hang *et al* (1994). In Chapter 4 the Phase Locked Loop equivalence of the Hang *et al* relay procedure was investigated and reported upon.

The extension of the PLL method to the identification of multivariable systems was reported in Chapter 4. This was shown to be relatively simple to implement and to give accurate estimates of the frequency response of the process transfer function elements. Having shown that the identification of a multivariable process can be accomplished in closed loop by the PLL method of system identification, an extension of the Fung *et al* (1995) controller design method to multivariable systems was used to carry out a gain and phase margin design for the example system of Wood and Berry (1973).

The major achievements presented in Chapter 4 are for the following topics.

Cascade Connected Control Systems: In Section 4.3.1 the Phase-Locked Loop (PLL) method was extended to the closed loop identification of the processes connected in closed loop. The algorithm Closed Loop Identification of Cascade Connected Control Systems Using the PLL Method was developed and demonstrated successfully on a cascade connected control system that is representative of those found in process

industries. In Section 4.3.2 the Closed Loop Identification and Tuning of a Cascade Connected System algorithm was developed and successfully demonstrated. A stability test was included that confirms that the closed loop cascade system will still be stable when the inner controller parameters are updated.

Multivariable Control Systems: In Section 4.4.1 the extension of the Phase-Locked Loop system identification method to the closed loop identification of the individual transfer function elements of a multivariable process was discussed and developed. In Section 4.4.2 the Multivariable Process Closed Loop Identification algorithm was successfully demonstrated on the Wood and Berry (1973) column example. An extension of the exact gain and phase margin PI controller design method due to Fung et al (1998) was developed in Section 4.4.3. The Exact Gain and Phase Margin Design for a 2-Input 2-Output Process algorithm was developed and successfully demonstrated.

6.5 Continuous Parameter Cycling Method of Model-Free Controller Design.

In Chapter 5, the Iterative Feedback Tuning (IFT) method of Hjalmarsson *et al* (1994, 1998) was presented. The key features of the IFT method were found to be:

i) A system description involving a stochastic process output disturbance

ii) A two degrees of freedom control law

iii) The use of a stochastic optimisation approach, and

iv) A restricted structure control law.

However, a known problem with IFT is that extraction of the Hessian is difficult to perform using similar experiments to those used to extract the gradient. This is the problem that was addressed in Chapter 5. A totally different approach to obtaining the gradient and Hessian information was given. This was reported in Chapter 5 as the Continuous Parameter Cycling (CPC) method. (Crowe *et al*, 2003). CPC is a model-free approach to the design of restricted structure controllers.

The CPC method employs a time varying controller gain perturbation. The time varying gain perturbation gives rise to a time varying cost function. By using sinusoidal controller gain perturbations the sinusoidal orthogonality can be employed to extract

both the gradient and the Hessian information from the time varying cost function. By employing a symmetry argument it was also possible to reduce the time by half over which the gain perturbation is applied. The CPC method of controller tuning was shown to have a unified theoretical basis and was demonstrated in Chapter 5 on a non-trivial example.

The major achievements presented in Chapter 5 were as follows:

i) To motivate the research, the method of Iterative Feedback Tuning (IFT) was reviewed in Section 5.2. A failing of the IFT method is that second order Hessian data cannot be easily generated by the method.

ii) A new and original method, Controller Parameter Cycling, was proposed. The theory underpinning the new method was presented in Section 5.3 and implementation considerations were given in Section 5.4.

iii) In Section 5.5 the new CPC algorithm was demonstrated successfully on the non-trivial example of the design of a decentralised controller for a 2-input, 2-output multivariable process.

6.6 Future Research.

In previous chapters a number of tools are described to allow:

i) The identification of a process in closed loop, whether it is a single, cascade or multivariable system.

ii) Determining viable gain and phase margin PI controller designs.

iii) Determining PID controller designs for particular gain and phase margin pairs.

iv) Iterative design of PI controllers to meet gain and phase margin designs.

v) Iterative design of PI controllers to meet maximum sensitivity and phase margin designs.

vi) Model-free design of PID controllers.

The above tools, when coupled with a suitable identifier, can be used to produce a controller design that will give an acceptable level of control system performance. However, a great deal of input would be required by a control engineering practitioner to carry out the required design steps. The key benefit of auto-tuners based on the relay

experiment is that they are effectively *plug and play* in their operation. The tools described above if appropriately organised could be used to implement an auto-tuner with a high degree of autonomy, however, before such an auto-tuner could be produced there are a number of areas that require further research.

Improving the Phase Locked Loop Identifier.

The Phase-Locked Loop (PLL) method of nonparametric system identification has been shown to be capable of performing closed loop identification of the frequency response of a process connected in a single-input single-output, cascade connected or multi-input multi-output control system. Two areas for further research are as follows.

Guidelines for Excitation Signal Size: The PLL method gives a high degree of measurement accuracy even when process disturbances or measurement noise is present. The excitation magnitude can be set to a low value such that the impact of the identification on the process can be kept to an acceptable level. Guidelines for selecting the appropriate level are required.

Initialisation for Reduced Identification Time: To perform PLL identifications the only input required by an operator is the phase or gain reference value, the excitation magnitude, the initial frequency for the voltage controlled oscillator and a gain value for the integrator. In practice it is found that an integrator gain value of 0.2 can be used for a wide range of processes. However an effective method of obtaining a set of initialising values that lead to a reduction in the identification time should be a topic for future research.

Improving the PID Controller Design Methods.

An Algorithm for Viable Maximum Sensitivity and Phase Margin Pairs: In Chapter 2 graphical methods were discussed that allow the design of PI controllers such that all of the viable gain and phase margin pairings within a prescribed range are given. It is possible to extend this PI controller design method such that all of the viable maximum

sensitivity and phase margin pairings within a specified range can be determined. However, the tangency angle must be specified *a priori*. The effect of having a fixed tangency angle on the resulting maximum sensitivity and phase margin pairings requires further research. Likewise the effect that a fixed tangency angle has on the resulting PI controller parameters and the effect this has on the closed loop system time response requires to be investigated.

Improved Selection for Controller Specification: Methods are required to aid the selection of a particular gain and phase margin or maximum sensitivity and phase margin pairing such that not only are the robustness measures met but also an acceptable time domain response is obtained for the closed loop system. Extensions to the semi-graphical gain margin and phase margin design of PID controllers are required such that selection methods are available to aid the choice of design frequencies relating to a particular phase margin and gain margin pair. The desirable outcome of such a selection method would be that the closed loop system has an acceptable time domain response in addition to having the desired robustness properties. These topics require further research.

Faster Convergence For Iterative PID Design Methods: Iterative design methods that allow the design of PI controllers that give a specific gain and phase margin or maximum sensitivity and gain margin are discussed in Chapter 3. Extensions to these methods are required such that the number of iterations that are required to achieve the required design specification can be reduced.

Faster Convergence for Iterative Maximum Sensitivity and Phase Margin Design Methods: The maximum sensitivity and phase margin iterative design method should be further researched to determine the effect of specifying a fixed tangency angle. This would certainly reduce the number of identifications required, however the effect this would have on the final PI controller design and on the closed loop system time response requires to be further investigated.

In Chapter 4, the use of the Phase-Locked Loop (PLL) method of system identification was applied to the identification of cascade and multivariable systems connected in closed loop. There are several areas for further research in this Chapter.

Phase-Locked Loop (PLL) identification of cascade systems: The use of the Phase-Locked Loop (PLL) method of system identification was applied to the identification of cascade systems connected in closed loop. Gain and phase margin design of the inner and outer PI controllers of a cascade system was performed. If gain and phase margin or maximum sensitivity and phase margin are to be used as the design specification for a cascade control system, then a method is required such that the inner loop controller can be designed to mitigate the effects of a process disturbance. Similarly, the method should address the design of the outer loop controller to give an acceptable time domain response. These ideas should be further researched to give a new and original version of the controller design algorithm.

Phase-Locked Loop (PLL) identification of multivariable systems: The use of the Phase-Locked Loop (PLL) method of system identification was applied to the identification of multivariable systems connected in closed loop. Gain and phase margin design of the PI controllers for a decentralised control system was performed. Further research should be conducted to apply a gain and phase margin or maximum sensitivity and phase margin as the design specification. The method should address the design of the loop controller to give an acceptable time domain response.

Reduction of loop interaction in decentralised control design: An extension to the exact gain and phase margin design method of Fung *et al* (1998) was used to design a decentralised controller for a multivariable system. The time domain response of the closed loop system shows that a large interaction between the control loops still exists. A method is required that allows the design of a full matrix controller such that gain and phase margin designs can still be specified with the interaction between control loops being significantly reduced; this requires further research.

In Chapter 5 the new and original Controller Parameter Cycling (CPC) method was devised and discussed. There are various issues arising from this chapter for further research.

Using Different Orthogonal Function Sets: The initial choice for the parameter perturbation was to use sinusoidal functions. This choice leads to a relatively large number of gain perturbations being implemented so that an accurate representation of the time varying cost function can be obtained. An investigation of other time functions that possess the required orthogonality properties is required to see if a reduction in the number of gain perturbations can be obtained and still achieve the required accuracy of representation of the time varying cost function.

Improving The Levenberg-Marquardt Routine: The choice of the parameter α for the Levenberg-Marquardt routine requires a certain skill on the part of the operator to ensure that both the cost function and the cost function gradient continue to reduce when the Hessian estimate is non-positive definite. Guidelines are required to ensure that changes made to α continue to reduce the cost function and the cost function gradient. Similarly when the Hessian estimate is positive definite guidelines are required to ensure that an appropriate step size is chosen. These guidelines require further research input.

Towards Autonomous PID Control – Developing a Phase Locked Loop Auto-tuner.

The identification and PID controller tuning tools discussed above have been implemented using either LabVIEW™ or Matlab™/Simulink™. If the tools are going to be used to form the basis of an auto-tuner then a common platform is required for their implementation. In addition, and depending on the platform choice, a range of Human Machine Interfaces (HMI) are required to be designed such that the operator is led through a simple procedure to allow single loop, cascade or multivariable controller design to be performed. The auto-tuner could be implemented in a Distributed Control System (DCS) as a function block that is called to commission new loops or to re-tune existing control loops or as a stand alone hardware unit that is connected into a control

system as required for identification or tuning duties. The further research needed to implement the above tools into an auto-tuner should consider the following practical issues:

i) The method of connection to the process (physical interfaces, serial links, instrument bus systems, etc.)

ii) An estimation of the processor loading when used in a DCS either as a module that is called for an individual loop or as a function block having many instantiations.

References

Abramowitz, M. and A. Stegun (Eds.), 1972, Handbook of Mathematical Functions, Dover Publications, ISBN 0-486-61272-4

Astrom, K. J. and T. Hagglund, 1984, Automatic Tuning of Simple Controllers with Specification on Phase and Amplitude Margins, Automatica, Vol. 20 No. 5, pp. 645-651

Astrom, K. J. and T. Hagglund, 1995, PID Controllers: Theory Design and Tuning, Instrument Society of America, ISBN 1-55617-516-7

Astrom, K. J., H. Panagopoulos and T. Hagglund, 1998, Design of PI Controllers based on non-convex optimisation, Automatica, Vol. 34, No. 5, pp. 585-601

Atherton, D. P., 1975, Non-Linear control engineering – describing function analysis and design, Van Nostrand Reinhold, London, U.K.

Balestrino, A., A. Landi and L. Scani, 2000, ATV techniques: troubles and remedies, ADCHEM 2000, IFAC International Symposium on Advanced Control of Chemical Processes, Pisa, Italy, 14th – 16th, June

Bi, Q., Q.-G. Wang and C.-C. Hang, 1997, Relay based estimation of multiple points on the process frequency response, Automatica, Vol. 33, No.4, pp. 1753-1757

Blevins, T. L., G. K. McMillan, W. K. Wojsznis and M. W. Brown, 2002, Advanced control unleashed – plant performance management for optimum benefit, Instrument Society of America, ISBN 1-55617-815-8

Bokensenbom, A. S. and R. Hood, 1949,General algebraic method applied to control analysis of complex engine types, National Advisory Committee for Aeronautics, Washington, D.C., U.S.A., Report NCA-TR-980

Clarke, D. W. and J. W. Park, 2003, Phase-locked loops for plant tuning and monitoring, IEE Proceedings Control Theory and Applications, Vol. 150, No. 1, pp 155-169, March

Crowe, J. and M. A. Johnson, 1998, A phase locked loop identifier and its application, Proceedings of the fifth IChemE. International Conference on Advances in Process Control (APC'5), (177-186), University of Wales, Swansea, U. K. ,2^{nd}-3^{rd}, September

Crowe, J. and M. A. Johnson, 1999, A new non-parametric identification procedure for online controller tuning, American Control Conference, pp. 3337-3341, San Diego, U.S.A. $2^{nd} - 4^{th}$, June

Crowe, J. and M. A. Johnson, 2000a, Automated PI controller tuning using a phase locked loop identifier module, IECON 2000, IEEE International Conference on Industrial Electronics, Control and Instrumentation, Nagoya, Japan, 22-28, October.

Crowe, J. and M. A. Johnson, 2000b, Open and closed loop process identification by a phase locked loop identifier module, ADCHEM 2000, IFAC International Symposium on Advanced Control of Chemical Processes, Pisa, Italy, $14^{th} - 16^{th}$, June

Crowe, J. and M. A. Johnson, 2000c, Process identifier and its application to industrial control, IEE Proceedings Control Theory and Applications, Vol. 147, No. 2, pp 196 – 204, March

Crowe, J. and M. A. Johnson, 2001a, Automated PI control tuning to meet classical performance specifications using a phase locked loop identifier, 2001 American Control Conference, Arlington, VA, USA, 25-27, June.

Crowe, J. and M.A. Johnson, 2001b, PID Tuning for Classical Robustness Specifications by Enumeration Methods, IECON 2001, IEEE International Conference on Industrial Electronics, Control and Instrumentation, Denver, USA, 29 Nov.– 02 Dec.

Crowe, J. and M.A. Johnson, 2002a, Automated Maximum Sensitivity and Phase Margin Specification Attainment in PI Control, Asian Journal of Control, Vol. 4, No. 4, December.

Crowe, J. and M.A. Johnson, 2002b, Toward Autonomous PI Control Satisfying Classical Robustness Specifications, IEE Proceedings: Control Theory and Applications, Vol. 149, No. 1, January 2002.

Crowe, J., 1998, Identification methods for process control applications, M.Phil. Thesis, Department of Electrical and Electronic Engineering, University of Strathclyde, Glasgow, U.K.

Crowe, J., M. A. Johnson and J. Wilkie, 2001, Recent developments in PID control for process control applications, CPACT conference in Advances in Process Analytics and Control, Glasgow, UK, 3-4, April.

Crowe, J., M. A. Johnson and M. J. Grimble, 2003a, On the Closed Loop Identification of Systems within Cascade Connected Control Strategies, European Control Conference, University of Cambridge, UK, 1 – 4 September.

Crowe, J., M. A. Johnson and M. J. Grimble, 2003b, PID Parameter Cycling to Tune Industrial Controllers – A new model-free approach, SYSID, 13th IFAC Symposium on System Identification, Rotterdam, The Netherlands, 27 – 29 August

de Arruda, G.H.M. and P. R. Barros, 2003, Relay-based closed loop transfer function frequency point estimation, Automatica, Vol. 39, pp. 309-315

Eker. I., and M. A. Johnson, 1996, New aspects of cascade and ,ulti-loop process control, Trans. I. Chem. E., Vol. 74, part A, January

Favoreel, W., B. De Moor and P. Van Overschee, 1999, Model-free subspace-based LQG-design, Proceedings American Control Conference, San Diego, California, June

Favoreel, W., B. De Moor, M. Gevers and P. Van Overschee, 1998, Model-free subspace-based LQG-design, Katholieke Universiteit Leuven, Department Electrotechniek, Report No. ESAT-SISTA/TR 1998-34.

Fung. H. W., Q.-G. Wang and T. H. Lee, 1998, PI Tuning in Terms of Gain and Phase Margin, Automatica, Vol. 34, No. 9, pp. 1145 - 1149

Grimble, M. J. and M. A. Johnson, 1988, Optimal control and stochastic estimation: Theory and applications, Vol. 1, John Wiley and Sons Ltd., Chichester, UK, ISBN 0-471-90593-3

Hang, C.-C., A. P. Loh and V. U. Vasnani, 1994, Relay feedback auto-tuning of cascade controllers, IEEE Trans. on Cont. Sys. Tech., Vol. 2, No. 1, pp. 42-45, March

Hang, C.-C., K. J. Astrom and W.-K. Ho, 1991, Refinements of the Ziegler-Nichols tuning Formula, IEE Proceedings, part D, Vol. 138, No. 2, March

Harris, T. J., 1989, Assessment of control loop performance, Canadian Journal of Chemical Engineering, Vol. 67, pp. 856-861

Hersh, M. A. and M. A. Johnson, 1997, A Study of Advanced Control Systems in the Workplace, Control Eng. Practice, Vol. 5, N0. 6, pp. 771 – 778

Hjalmarsson, H., S. Gunnarsson and M. Gevers, 1994, A convergent iterative restricted complexity control design scheme, Proc. CDC, Florida, USA, December

Hjalmarsson, H., S. Gunnarsson and M. Gevers, 1998, Iterative feedback tuning: theory and applications, IEEE CSS magazine, August, pp. 26-41

Ho, W. K., T. H. Lee and O. P. Gan, 1995, Tuning of Multiloop PID controllers based on gain and phase margin specifications,

Ho, W.-K., C.-C. Hang and L.-S. Cao, 1995, Tuning of PID controllers based on gain and phase margin specifications, Automatica, Vol. 31, No. 3, pp. 497-503

Johnson, M. A. and J. Crowe, 1998, New approaches to non-parametric identification for control applications, Preprints IFAC Workshop on Adaptive Systems in Control and Signal Processing, (309-314), Glasgow, Scotland U. K., 26th-28th, August.

Kadali, R., B. Huang and A. Rossiter, 2003, A data driven subspace approach to predictive controller design, Control Engineering Practice, Vol. 11, pp. 261-278

Kammer, L. C., R. R. Bitmead and P. L. Bartlett, 2000, Direct iterative tuning via spectral analysis, Automatica, Vol. 36, pp. 1301-1307
Kaya and Atherton, 2001

Khalil. H. K., 1992, Nonlinear systems, MacMillan, New York, ISBN 0-02-363541-X

Lee, T. H., Q. G. Wang and K. K. Tan, 1995, A modified relay based technique for improved critical point estimation in process control, IEEE Trans. on Control Systems Technology, Vol. 3, No. 3, pp. 330-337

Ljung, L., 1987, System Identification: theory for the user, Prentice Hall, Englewood Cliffs, USA, ISBN 0-13-881640-9

Loh, A. P., X. Cai and W. W. Tan, 2001, Novel use of relays in identifying frequency response points, Proc. ACC, Arlington, Virginia, 25–27 June.

Mahathanakiet, K., M. A. Johnson, A. Sanchez and M. Wade, 2002, Iterative feedback tuning and an application to a wastewater treatment plant, Proc. Asian Cont. Conf., pp. 256-261, Singapore, September

McGrath, R. J., V. Rajaraman and V. C. Rideout, 1961, A Parameter Perturbation Adaptive Control System, IRE Transactions on Automatic Control, Vol. 6, pp. 154-161

Narendra, K. S. and L.E. McBribe, 1964, Multiparameter self-optimising systems using correlation techniques, IEEE Transactions on Automatic Control, Vol. 9, pp. 31-38

O'Dwyer, A., 1998a, PI and PID controller tuning rules for time delay processes: a summary. Part 1: PI Controller tuning rules

O'Dwyer, A., 1998b, PI and PID controller tuning rules for time delay processes: a summary. Part 2: PID Controller tuning rules

Ogata, K., 1997, Modern Control Engineering, Prentice-Hall International, ISBN 0-13-261389-1.

Palmor, Z. J., Y. Halevi and N. Kransky, 1995, Automatic tuning of decentralised PID controllers for TITO processes, Automatica, Vol. 31, No. 7, pp 1001-1010

Panagopoulos, H., K. J. Astrom and T. Hagglund, 1999, Design of PID controllers based on constrained optimisation, Proc. American Control Conference, San Diego, California, 2-4 June.

Panda, R. C. and C.-C. Yu, 2002, Analytical expressions for relay feedback responses, Journal of Process Control

Robbins, H., and S. Munro, 1951, A stochastic approximation method, Ann. Math. Stat., Vol. 22, pp. 400-407

Schei, T. S., 1994, Automatic tuning of PID controllers based on transfer function estimation, Automatica, Vol. 30, No. 12, pp. 1983-1989, December

Schwarz, H. R., 1989, Numerical analysis: A comprehensive introduction, John Wiley and Sons Ltd., Chichester, UK.

Seborg, D. E., T. F. Edgar, and D. A. Mellichamp, 1989, Process Dynamics and Control, Wiley, New York, ISBN 0-471-85933-8

Shen, S.-H. and C.-C. Yu, 1994, Use of relay-feedback test for automatic tuning of multivariable systems, AIChE Journal, Vol. 40, No. 4, pp. 627-645, April

Shen, S.-H., H.-D. Yu and C.-C. Yu, 1996b, Use of BIASED-relay feedback for system identification, AIChE Journal., Vol. 42, No.4, pp. 1174-1180, April

Shen, S.-H., J.-S. Wu and C.-C. Yu, 1996a, Use of saturation-relay feedback for autotune identification, Chem. Engrg. Sci., Vol. 51, No.8, pp. 1187-1198

Shin, C.-H., M.-H. Yoon and I.-S. Park, 1997, Automatic tuning algorithm of the PID controller using two Nyquist points identification, SICE'97, Tokushima, Japan, 29-31 July

Soderstrom, T. and P. Stoica, 1989, System Identification, Prentice Hall, Englewood Cliffs, USA, ISBN 0-13-127606-9

Talkin, A. I., 1961, Adaptive Servo Tracking, IRE Transactions on Automatic Control, Vol. 6, pp.167-172

Tan, K. K., Q.-G. Wang, C. C. Hang with T. J. Hagglund, Advances in PID Control, Advances in Industrial Control Series, Springer, London, UK, ISBN 1-852-33138-0

Thornhill, N. F., M. Oettinger, P. Fedenczuk, 1999, Refinery-wide control loop performance assessment, Journal of Process Control, Vol. 9, pp. 109-124

Wang, Q.-G., B. Zou, T.-H. Lee and Q. Bi, 1997b, Auto-tuning of multivariable PID controllers from decentralised relay feedback, Automatica, Vol. 33, No. 3, pp.319-330

Wang, Q.-G., C.-C. Hang and Q. Bi, 1997a, Process frequency response estimation from relay feedback, Control Engineering Practice, Vol. 5, No. 9, pp.1293-1302

Warick, K. and D.Rees, (Eds.), 1988, Industrial Digital Control Systems, IEE Control Engineering Series, No.37, Peter Peregrinus Ltd., London, UK, ISBN 0-86341-139-8

Wilkie, J., M. A. Johnson and M. R. Katebi, 2002, Control Engineering an introductory course, Palgrave, Basingstoke, UK, ISBN 0-333-77129-X

Wood, R. K. and M. W. Berry, 1973,Terminal Composition Control of a Binary Distillation Column, Chem. Eng. Sci., Vol. 36, 375

Woodley, B. R., J. P. How and R. L. Kosut, 2001, Model free subspace based H_∞ control, Proceedings American Control Conference, Arlington, Virginia, 25-27 June

Yu, C.-C., Autotuning of PID Controllers, Advances in Industrial Control Series, Springer, London, UK, ISBN 3-540-76250-7

Zhuang, M. and D. P. Atherton, 1991, Tuning PID controllers with integral performance criteria, IEE Proceedings on Control Theory and Applications, Vol. 332, No.1, pp. 481-486.

Zhuang, M. and D. P. Atherton, 1994, PID controller design for a TITO system, IEE Proceedings on Control Theory and Applications, Vol. 141, No.2, pp. 111-120, March

Ziegler, J. G. and N. B. Nichols, 1942, Optimum settings for automatic controllers, Trans. ASME, Vol. 64, pp. 759-768.