UNIVERSITY OF SOUTHAMPTON

FACULTY OF PHYSICAL AND APPLIED SCIENCES

Electronics and Computer Science

Design and experimental evaluation of iterative learning controllers on a multivariable test facility

by

Thanh Dinh Van

Thesis for the degree of Doctor of Philosophy

June 2013

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF PHYSICAL AND APPLIED SCIENCES
Electronics and Computer Science

Doctor of Philosophy

DESIGN AND EXPERIMENTAL EVALUATION OF ITERATIVE LEARNING CONTROLLERS ON A MULTIVARIABLE TEST FACILITY

by Thanh Dinh Van

Iterative learning control (ILC) algorithms are employed in many applications, especially these involving single-input and single-output plants undertaking repeated tasks with finite-time interval. ILC is applicable to systems executing a repeated trajectory tracking task, and uses data recorded over previous trials in the construction of the next control input. The objective is to sequentially improve tracking accuracy as the trial number increases. This method has been shown to operate well in the presence of significant modeling uncertainty and exogenous disturbances. However, for MIMO (multiple input -multiple output) systems, there exist far fewer applications reported in the literature, and minimal benchmarking and evaluation studies have been undertaken. To tackle this shortcoming, this thesis focuses on designing an electromechanical test-bed which can verify the weaknesses and the advantages of various ILC methods on a purpose-built platform. The system has two inputs and two outputs and enables variation of the interaction between inputs and outputs through simple and rapid parameter modification. This interaction variation permits the control problem to be modified, allowing stipulation over the challenge presented to the ILC controller. The system is made up of two back-to-back differential gearboxes with mass-spring-damper components to increase the system order and control difficulty. In its standard configuration, two motors provide torque to the two input ports and the two outputs are measured using encoders. This work enables a comparative summary of ILC approaches for MIMO systems, together with modifications for improved performance and robustness, and the development of new control schemes incorporating input and output constraints and point-to-point tracking capability. The system can also be configured in a variety of other arrangements, varying the number of inputs and outputs, and allowing noise to be injected using a dc motor. Models of the system are derived using a lumped parameter system representation, as well as purely from experimental input and output data. Simple structure controllers such as proportional-type ILC, derivative-type ILC and phase-lead ILC are then applied to test the combined performance of the controller and the MIMO system, and establish its efficacy as a benchmarking platform. Advanced controllers are then derived and applied and experimental data are used to confirm theoretical findings concerning the link between interaction and convergence rate, input norm and robustness.

Contents

List of Figures

List of Tables

Nomenclature

$\overline{\sigma}$	Maximum singular value
$\underline{\sigma}$	Minimum singular value
$\|.\|_\infty$	Infinity norm
β	Gain constant
$\|.\|$	Absolute value
$\|\|.\|\|_p$	p norm
$\|\|.\|\|_2$	2 norm
$\mathbb{R}$	Field of real numbers
s	Laplace transform variable
$\sigma_i(A)$	The i^{th} singular value of matrix A
θ	Angular value
G^T	Transpose of matrix G
I_n	$n \times n$ identity matrix
0_n	$n \times n$ zero matrix
$\lambda_i(A)$	The i^{th} eigenvalue of matrix A
$\rho(A)$	Spectral radius of matrix A, equivalent to $\max_i \|\lambda_i(A)\|$
$\mathbf{y}_d$	Tracking reference defined over whole trial
$\mathbf{y}_r$	Tracking reference for point-to-point movements
A^T	Transpose of matrix A
k	Iteration number
$\mathbf{y}_k$	Measured output on k^{th} trial
Q	Weighting matrix in gradient-based ILC
R	Weighting matrix in gradient-based ILC
N	Number of samples
T_s	Sampling time in seconds
$\mathbf{e}$	Control error
x	State vector
t	Time variable
γ	Constant matrix for basic ILC control update
L	Diagonal constant matrix
G	Transfer function matrix
δ	Integer time delay for phase-lead ILC

S	Shift operator matrix
D	Differential operator
m	Number of inputs
p	Number of outputs

Acknowledgements

Thanks to my family and my friends who support my studies. A grateful acknowledgment goes to my supervisors Dr Chris Freeman and Prof Paul Lewin for their help and guidance.

Chapter 1

Introduction

Within control theory, there are an exceptionally diverse range of control methodologies covering a wide variety of system types and applications. One such approach is iterative learning control (ILC) which has been developed for more than 25 years. ILC is suitable for systems which perform a repeated process defined over a finite time interval. ILC uses the previous error of a trial to modify the control signal with the aim of sequentially improving tracking accuracy. In practice, there does not exist a perfect model for a system, and the presence of unpredicted disturbance or noise on the system's signals means an exact representation of the system is impossible. However ILC has shown the ability to provide high performance without the need for an exact model, demonstrating the ability to cope with system uncertainty and exogenous disturbance (Bristow et al., 2006), (Ahn et al., 2007). In practical terms, this significant reduces the effort that an engineer must place on identifying a system model. This robustness to modeling uncertainty has led to its popularity in many industries, most notably in manufacturing (Kim and Kim, 1996), chemical process engineering (Tan and Tang, 2002), (Gao et al., 2001), robotics applications (Elci et al., 1994), (Norrlöf, 2002) and biomedical engineering (Huang et al., 2003). ILC has also been modified and integrated with other control techniques with the aim of further increasing their performance. For example ILC has been integrated with tuning of proportional plus integrated plus differential (PID) controllers in (Li, 2010), with fuzzy controllers in (Zheng et al., 2009), and with adaptive controllers in (French et al., 1999), (French and Rogers, 2000), (Tayebia and Islamb, 1984). There has been a large number of published works on the subject of ILC, many of which include experimental application. There is a distinct lack, however, of literature using multiple input, multiple output (MIMO) systems, which typically are much harder to control than single input, single output (SISO) systems. It is well known that designing controllers for MIMO systems can be extremely challenging as is illustrated in applications to, for example, an industrial distillation column (Tyréus, 1979), (Haurani et al., 2001), and this is certainly true for ILC. Additionally, no MIMO system exists in which the level of interaction level is adjustable in order to enable precise investigation

of the role played by input-output coupling on the subsequent performance of ILC. Such a platform is critical for examination and benchmarking of MIMO update algorithms.

Consequently, the main object of this research is focused on designing a test facility which can be used to implement different kinds of ILC algorithm and enable MIMO ILC approaches to be rigorously evaluated on a demanding multi-configurable experimental platform. The facility will also be used to investigate the effect of interaction and noise injection on the performance of different controllers. The MIMO testbed will be used to assess existing controllers as well as new varieties which track only a subset of outputs along the trial duration. Such point-to-point ILC updates will be compared in detail to their standard counter parts (Freeman and Tan, 2013), (Freeman and Tan, 2011), (Freeman, 2012). After deriving these algorithms, convergence rate and robustness analysis results are developed in order to examine role of interaction in detail. This leads to fundamental results on interaction, convergence and how it affects robustness and input norms. Moreover, it explicitly leads to a design framework for point-to-point ILC to overcome traditional control limitations.

The first task of the research is deriving transfer-functions for every configuration for the MIMO system so that an optimal design and configuration can be chosen. The resulting 2×2 MIMO system is controllable via a PC interface and have the facility for all inputs and outputs to be stored for further analysis. Basic ILC algorithms such as P-type ILC and D-type ILC will be used to control the system, as well as advanced ILC algorithms which will be studied and implemented using the same system. Advanced controllers are mainly based on a model of the system, and this is derived via frequency modeling. The method can reduce the order of the system to make computation shorter and more straightforward. The controllers studied in this thesis comprise gradient descent based ILC and norm optimal ILC . These two methods are chosen due to their popularity, their similarity in implementation and use of the same underlying cost function. For the first time, these controllers are analysed in depth with convergence rate and robustness results which illustrate the role of MIMO interaction and verify experimental findings.

In order to achieve these objectives, this thesis starts with a review of existing ILC algorithms in Chapter 2. The chapter also focus on application of ILC within industry. Additionally, leading MIMO controllers are also discussed and compared to ILC. In Chapter 3 Lagrange's equations are introduced and applied to derive transfer-functions for possible configurations of the MIMO system. The chapter also gives details of the principal components in the MIMO system and different arrangements are discussed in detail. In Chapter 4, each arrangement of the system is examined for a range of realistic parameters. Using extensive simulations, the optimum configuration is chosen based on the relative gain array method (Bristol, 1966). Components to realise disturbance/noise injection are then installed to complete the MIMO facility. The system is then tested in simulation with a variety of controllers. In Chapter 5, the practical realisation of the

MIMO system is undertaken through design in 3D Solidworks in order to aid manufacturing. The manufactured MIMO facility is then discussed and presented along with the electrical system. Before applying advanced ILC controllers, basic methods need to be applied and are given in Chapter 6. All the results are discussed and analysed in detail, and their shortcomings are highlighted. This provides motivation to apply advanced controllers to improve the results. Chapter 7, 8 introduce advanced controllers based on a model of the MIMO facility. All controllers in these chapters use the standard ILC framework and these results are then discussed and weaknesses analysed. Chapter 9 focusses on novel controller design with the aim of addressing limitations of the standard ILC framework. The class of algorithms is named point-to-point and the methodology can be extended to most ILC techniques. This chapter also provides gives analysis of the method in comparison with standard controllers. Performance is found to associate closely with the level of interaction, and this finding is supported by extensive experimental results. Finally Chapter 10 gives conclusions on the results and clearly highlights future work that should be conducted.

Chapter 2

Review of multivariable tracking and iterative learning control

Many methods have been derived to analyse and design controllers to stabilise a system and/or achieve performance objectives. However it still can be a very difficult task to satisfactorily control a multivariable system, these having more than one input and/or more than one output. Therefore in this chapter, multivariable systems will be reviewed and different control methods will be discussed and evaluated. Subsequently the most promising method will be tested and implemented on the experimental system to provide baseline performance. The literature review then summarises ILC approaches with specific focus on multivariable systems.

2.1 Multivariable systems

There are two types of system considered in this thesis; single input, single output (SISO) system and a multiple input, multiple output (MIMO) system. The difference between these systems lies in the presence of interaction such that any one output is influenced by more than one input, or any one input influences more than one output. In general, the more inputs and outputs, the more challenging it is to design suitable controllers when interaction is present. This chapter will focus on analysing and synthesising controllers to make the resulting MIMO system robust and achieve specific requirements.

Consider a MIMO plant with m inputs and p outputs. Thus, the basic transfer-function is $\mathbf{y}(s) = G(s)\mathbf{u}(s)$ where $\mathbf{y}$ is an $p \times 1$ vector, $\mathbf{u}$ is $m \times 1$ vector and $G(s)$ is $p \times m$ transfer-function matrix, alternatively

$$\begin{bmatrix} \mathbf{y}_1(s) \\ \vdots \\ \mathbf{y}_l(s) \end{bmatrix} = \begin{bmatrix} G_{11}(s) & \dots & G_{1m}(s) \\ \vdots & \ddots & \vdots \\ G_{l1}(s) & \dots & G_{lm}(s) \end{bmatrix} \begin{bmatrix} \mathbf{u}_1(s) \\ \vdots \\ \mathbf{u}_m(s) \end{bmatrix} \tag{2.1}$$

If there is any change in input component $\mathbf{u}_1$ all the outputs of the system will be affected in proportion to the interaction between inputs and outputs. In order to calculate the gain of a MIMO plant, the singular value decomposition (SVD) technique was introduced at around the end of the 19^{th} century by Eugenio Beltrami, Camille Jordan, James Joseph Sylvester, Erhard Schmidt and Hermann Weyl. Beltrami, Jordan and Sylvester came to the decomposition approach through a background in linear algebra while Schmidt and Weyl approached it from the route of integral equations. A full overview concerning the historical underpinnings of SVD can be found in Stewart (1993) with deeper theoretical detail given in Golub and Loan (1996).

The SVD method can be applied to analyse a MIMO system by decomposing the transfer-function matrix $G(s)$ in the form

$$G = U\Sigma V^T \tag{2.2}$$

where Σ is a $p \times m$ diagonal matrix with non-negative singular values, σ_i, arranged in descending order along its main diagonal. The matrix Σ is here represented as

$$\Sigma = \begin{pmatrix} \sigma_1 & 0 & \dots & 0 \\ 0 & \sigma_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \sigma_k \end{pmatrix}$$

where σ_1 is the maximum singular value and σ_k is the minimum singular value. U is an $p \times p$ unitary matrix of output singular vectors $\mathbf{u}_i$, and V is an $m \times m$ unitary matrix of input singular vectors, v_i which can be interpreted as input direction. V^T is the transpose of the matrix V. This method is still applicable for the case when the transfer-function matrix G is not a square matrix. The SVD is used to establish which input directions will not have any affect on the output, and conversely if any output direction cannot be controlled by an input. Additionally, the value of σ_i represents the gain of the subsystem associated with u_i and v_i. Therefore the ratio of the maximum singular value and minimum singular value is termed the conditional number which can be used to measure the input-output controllability. If the ratio is high (greater than ten), it indicates sensitivity to uncertainty and a difficulty to control. This means if there is any slight change in the input, there will be a far greater change in the output. If the conditional number is small, the system is easy to control and uncertainty is not likely to be serious. However, the SVD cannot show the magnitude of the interaction inside a MIMO system. In order to gain a general view of MIMO interaction, the relative gain array (RGA) was first introduced in Bristol (1966) and is defined as

$$RGA = \Lambda(G) = G \times (G^{-1})^T \tag{2.3}$$

where $\times$ denotes element by element multiplication. The value of RGA can be calculated

when s is equal to zero in the s-plane thus time is infinity and the frequency is zero. Therefore the system is analysed for the steady-state only. RGA is used to measure the interaction of a MIMO system so that the most important pairing can be found before starting to design controllers. Based on this theory, the method of decentralised control can choose the optimal coupling with the least interaction. However RGA still has a restriction since it is only able to show the interaction between each pair at steady-state, and hence the method has been extended to arbitrary frequencies in Bristol (1978). The use of the RGA was popularised by Shinskey (1988), who applied it to many applications including blending, energy conservation, and distillation. Later Kinnaert (1995) did a survey for dynamic extension of the RGA.

Further research shows that RGA can be modified to have a better index for stability and achievable performance, which is called performance RGA (Hovd and Skogestad, 1992). In (Grosdidier et al., 1985) it is shown that the method can be used to predict the closed-loop instability of multivariable systems. Following from this, RGA has been applied to choose the most desirable coupling for decentralised control (Albertos and Sala, 2004). Furthermore the method was used to approach interactions in the Takagi-Sugeno fuzzy model (Mollov et al., 2001).

An alternative way to approach the interaction problem is by using Participation Matrices which consider observability and controllability Gramians (Conley and Salgado, 2000). In a similar way Wittenmark and Salgado (2002) introduced the Hankel interaction index array. Both methods comprise appropriate approaches for decentralised and full multivariable controller structures. Another recent method has been proposed by Birk and Medvedev (2003) using the H_2 norm as a basis to measure the interaction effect. The thesis of Halvarsson (2010) provides a comparison between these methods using specific examples.

In this thesis, due to its simplicity and well established theoretical background, the RGA method is employed to find the best pairing for the MIMO system. The details of calculation are given in Section 4.5.

2.2 Basic controllers for MIMO systems

2.2.1 Decoupling and decentralise control

For a simple MIMO system with a feedback controller K, a simple approach involves the following two steps. Firstly a compensator $W(s)$ is designed to deal with interaction of the transfer-function matrix $G(s)$ so that the new transfer-function is more diagonal and hence easier to control. Then a diagonal controller is synthesised as in the case of a SISO system. Based on this procedure, the decoupling method (Skogestad and Postlethwaite, 1996) starts with designing a compensator $W(s) = G^{-1}(s)$ hence $G(s)W(s) = I$. This

method can be expressed using the block diagram of Figure 2.1. The designed controller is

Figure 2.1: Decoupling method block-diagram

$K(s) = l(s)G^{-1}(s)$ where $l(s)$ can be any transfer-function that realises the performance targets of the operator. In some cases the diagonal elements need to be kept the same so that the compensator matrix $W(s)$ can be set equal to $G^{-1}(s)(G_{\text{diag}})$. For example Zheng et al. (2004) used the decoupling method to design and control hard disk drives with dual actuators.

The advantage of this method is its simplicity and that all the control methods for SISO systems are available for application since, after the system is decoupled, it can be treated as parallel SISO systems with no interaction. However, the method has drawbacks which can make it unsuitable for practical implementation. The method involves the inverse of matrix $G(s)$, which leads to difficulty in some cases due to amplification of noise and lack of robustness with respect to the plant model. In addition, if the MIMO system is non-minimum phase, this method cannot be used since the controller will be unstable with right hand plane (RHP) poles. The method can use the RGA values, which are described in the previous subsection, to determine the appropriate pairs so that a diagonal controller can be designed separately. The purpose of RGA is to minimise the interaction of the system, detected by the off-diagonal terms. This method allows engineers to design a controller K independently, with each controller designed locally and then all the loops closed. However, the interaction from off-diagonal terms may cause the overall system to be unstable due to the presence of uncertainty and noise.

Another simple method is called decentralised control which ignores the interaction of the original plant. The method considers the MIMO system as parallel SISO plants. Therefore the controller $K(s)$ is designed independently for diagonal terms only. The method works well if the MIMO system is diagonal dominance otherwise the performance of the controller may be poor. The decentralised controller has been applied to control a hard disk drive system (Zheng et al., 2005) which has off-diagonal elements much smaller than the diagonal elements.

2.2.2 Centralised control

As opposed to the previously described control methods for MIMO plants, the centralised control approach synthesises a controller $K(s)$ to stabilise without transformation of the target controlled system into separate SISO systems. The method uses a state-space model description of the MIMO system as the starting point for analysis. The first

method relating to this category is the pole placement method which may be applied to both SISO and MIMO systems in state-space model form (Dorf and Bishop, 2005). This enables all poles of the system to be placed within the left hand plane so that the system is stable. The requirement of state observability clearly raises the issue of observer controller interaction and fortunately the separation principle holds for multi-variable systems (Albertos and Sala, 2004). Thus the observer and controller can be designed separately in order to reach a satisfactory output response.

A more complicated control method uses optimisation to decide on the state feedback matrix which best achieves a performance cost. Firstly, a constant J is introduced as the performance index for the targeted system. The cost J can be written as

$$J = \int_0^{t_f} g(\mathbf{x}, \mathbf{u}, t)dt \tag{2.4}$$

where t_f is the final time, $\mathbf{x}(t)$ is the state vector and $\mathbf{u}(t)$ equals the control vector. The main purpose of this algorithm is to try to find the optimum state feedback controller K so that the performance of the states and controller are optimum. The controller should be designed with a mixed cost because in reality the energy consumption is proportional to the square of the 2-norm of the input $\mathbf{u}(t)$ therefore the J index can strike a balance between system performance and energy consumption. Equation (2.4) can then be expressed as

$$J = \int_0^{\infty} (\mathbf{x}^T Q\mathbf{x} + \mathbf{u}^T R\mathbf{u})dt \tag{2.5}$$

The t_f parameter is now considered as infinity when the system reaches the steady-state, R and Q are symmetric and positive weighting matrices. The weighting matrices will be chosen so that the relative important of the state variable performance is contrasted with the importance of the expenditure of the system energy resource that is represented by $\mathbf{u}^T\mathbf{u}$. The general state-space model has the form

$$\dot{\mathbf{x}}(t) = A\mathbf{x}(t) + B\mathbf{u}(t) \tag{2.6}$$

$$\mathbf{y}(t) = C\mathbf{x}(t) \tag{2.7}$$

and the state feedback controller is $\mathbf{u}(t) = -K\mathbf{x}(t)$. The optimisation state feedback control method is straightforward to achieve with a low order system but in the case of higher order or MIMO systems the matrix A will be large and this method requires a high computational cost to find the controller K which achieves the optimum performance. Therefore this method must be solved computationally using a software environment such as Matlab. The controller K can be found from the equation

$$K = R^{-1}B^T P \tag{2.8}$$

where P is determined from the solution of the equation

$$A^T P + PA - PBR^{-1}B^T P + Q = 0 \tag{2.9}$$

which is termed a Riccati equation. This method will be applied to discrete iterative learning control and will be implemented later but with the error weighting instead of optimal state feedback.

Another method applicable to this section is mixed sensitivity design, which is discussed in, for example, Skogestad and Postlethwaite (1996) and Kwakernaak (2002). This approach has been applied to an aerospace case study Kwakernaak (2002). The purpose of the method is to design a feedback controller K which minimises the peak of the sensitivity function $S = \frac{I}{I+GK}$ where I is the identity matrix, and G is the transfer-function matrix. In the frequency domain, the sensitivity magnitude is then below the peak value for all frequencies. Therefore the effect of disturbance is limited and the system will be more robust with respect to external signal noise or plant uncertainty. The mixed sensitivity can be written as

$$\|N\|_\infty = \max_w \bar{\sigma}(N(jw)) < 1 \tag{2.10}$$

$$N = \begin{bmatrix} w_p S \\ w_T T \\ w_u K S \end{bmatrix} \tag{2.11}$$

The values of the weights w_p, w_T, w_u need to be chosen by the designer to address the particular requirements of the system. Here T is the complementary sensitivity $T = I - S$ and is associated with system performance. Therefore the H_∞ optimal solution can be defined as the minimum controller K which makes the infinity norm of $N(K)$ as small as possible

$$\min_k \|N(K)\|_\infty \tag{2.12}$$

Further examples and explanation can be found in Kwakernaak (1993). Maciejowski (1989) give further theory about MIMO control and design.

2.2.3 Direct Nyquist array, inverse Nyquist array method and Gershgorin band

The Nyquist method is very commonly used for SISO system analysis to investigate the stability of a closed-loop system. When the method is extended for application to MIMO systems it is known as the direct Nyquist array (DNA). In order to improve its convenience for analysing a system, the inverse Nyquist array (INA) has been derived based on the direct Nyquist array method. The method was firstly introduced by Rosenbrock (1969) and more details are described in (Rosenbrock, 1974). The approaches

aid engineers to synthesise a suitable feedback controller, $K(s)$ in order to stabilise the system. However, the method only works for diagonally dominant transfer-function matrices, therefore the controller $K(s)$ needs to be split into 2 different controllers; a diagonal controller K_d and a K_p pre-compensator. The controller K_p is used to ensure the transfer-function matrix $G(s)$ has diagonal dominance so that both Nyquist array methods can be used. The diagonal controller K_d can be designed individually for each loop to reach the specification requirements. The INA method is used more extensively for multi-variable systems compared to the direct Nyquist array because of its convenience and since it is possible to display results graphically. Full details relating to the INA method are given in (Munro, 1972). Considering the transfer-function matrix $G(s)$ and controller K_d, K_p, the open-loop transfer-function matrix $Q(s)$ can be expressed as

$$Q(s) = G(s)K(s) \tag{2.13}$$

so the close-loop transfer-function in Figure 2.2 is defined

$$H(s) = \big(I + Q(s)F\big)^{-1}Q(s) \tag{2.14}$$

where F is a diagonal gain $F = \text{diag}(f_i)$ with $i = 1, \ldots, m$. Therefore the inverse of

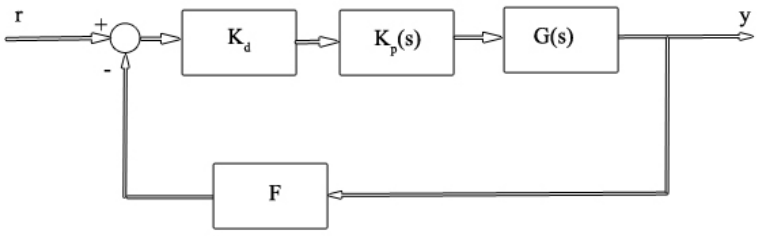


Figure 2.2: Feedback control loop.

the closed-loop transfer-function $H(s)$ is $\hat{H}(s) = F + \hat{Q}(s)$ where $\hat{H}(s) = H^{-1}(s)$ and $\hat{Q}(s) = Q^{-1}(s)$. The matrix $\hat{Q}(s)$ is a row diagonally dominant matrix if

$$|\hat{q}_{ii}(s)| > \sum_{\substack{j=1 \\ j\neq 1}}^{m} |\hat{q}_{ij}(s)| \qquad i = 1, 2, ..., m \tag{2.15}$$

or a column diagonal dominance matrix if

$$|\hat{q}_{ii}(s)| > \sum_{\substack{j=1 \\ j\neq 1}}^{m} |\hat{q}_{ji}(s)| \qquad i = 1, 2, ..., m \tag{2.16}$$

Here $\hat{q}_{ii}$ is a diagonal element of matrix $\hat{Q}(s)$, and m is the number of inputs/outputs. The diagonal dominance can be determined by graphical method called Gershgorin band

method. Circles are drawn with radius

$$d_i(s) = \sum_{\substack{j=1 \\ j \neq i}}^{m} |\hat{q}_{ij}(s)| \tag{2.17}$$

and centered on appropriate point of $\hat{q}_{ii}(s)$. If each the band produced excludes the origin, for $i = 1, \ldots, m$, the $\hat{Q}(s)$ is row dominant.

Therefore the stability of the system can be stated as follows. Let matrices $\hat{Q}(s)$ and $\hat{H}(s)$ be row or column dominant, let $\hat{Q}_{ii}(s)$ has contour $\hat{\Gamma}_{Qi}$ in the s-plane and let $\hat{H}_{ii}(s)$ has contour $\hat{\Gamma}_{Hi}$ in s-plane. Let $\hat{\Gamma}_{Qi}$ encircles the origin $\hat{N}_{Qi}$ times and $\hat{\Gamma}_{Hi}$ circles the origin $\hat{N}_{Hi}$ times clockwise respectively, then the closed-loop system is asymptotically stable if and only if

$$\sum_{i=1}^{m} \hat{N}_{Qi} - \sum_{i=1}^{m} \hat{N}_{Hi} = p_o \tag{2.18}$$

where p_o is the number of right-half s-plane poles of $H(s)$. Comparing the two methods, the inverse Nyquist direct method is slightly more complicated since it requires an inverse matrix, however, in practice the system must be analysed and a range if possible controllers will be established. The INA has greater advantage since all the necessary information is shown in the diagram as a function of frequency. At the same time, the Gershgorin band is defined as a tool to show whether the system is diagonally dominant. Follow this procedure, few applications give satisfactory results such as an automotive gas turbines, a chemical reactor, a multi stream heater (Munro, 1976), a turbogenerator (Ahson et al., 1979) and examples show how to use this method (Munro, 1990).

In general, by combining the Gorshgorin band and INA or DNA methods, an in-depth analysis framework is achieved for multivariable systems. This aids the engineer in choosing a suitable control method. However the method only applied for a square transfer-function matrix.

2.2.4 PI/PID controller design for MIMO systems

Proportional plus integral (PI) or proportional plus integral plus derivative (PID) are very common methods which are widely applied to SISO systems. Both are simple to calculate and can provide a high level of performance as well as robustness. For this reason, these methods are applied in many applications and throughout industry. For MIMO systems, the method has been modified so that it is dimensionally compatible and achieves the same main objectives as for a SISO system. The method has 2 types of control scheme: one is multi-loop control and the other one is decoupling control. The first approach treats the MIMO system as a whole and designs a controller matrix including off-diagonal and diagonal terms. The decoupling method seeks to design a matrix which can cancel out the interaction first, then a PI/PID controller is designed

separately for each loop to achieve performance objectives as in the SISO case. The decoupling control method was introduced in subsection 2.2 but now controller K is replaced by a PID controller matrix. There are many rules for tuning a PID controller and these are summarised in (O'Dwyer and Aidan, 2006). One common method used for MIMO systems is called biggest log modulus tuning (BLT) and this is explained fully in (Luyben, 1986). Before applying the tuning method it is assumed that the system has the most appropriate pairing with least interaction. Then the ultimate frequency (w_{ui}) is found when the phase angle is equal to -180 degrees and the ultimate gain (k_{ui}) is the reciprocal of the real part of G_{ii} which is the diagonal function, $i = 1, 2..., n$ where n is the number of inputs or outputs. The case in which systems have more inputs than outputs or vice versa is not addressed in (Luyben, 1986). Therefore the method can only be applied to a square matrix $G(s)$. Thus the equations used for calculating proportional gain and integral gain can be expressed as

$$K_{Pi} = \frac{K_{ui}}{F2.2} \tag{2.19}$$

$$\tau_{Ii} = \frac{F2\pi}{1.2\omega_{ui}} \tag{2.20}$$

where F is the detuning factor, which is normally chosen from 2 to 5 but in some cases to achieve stability, F can be outside this bound. In order to establish how well the method performs in reality, Luyben (1986) gives results for different MIMO systems such as the 2×2 MIMO system Tyreus stabiliser (Tyréus, 1979) or the 3×3 system appearing in (Ogunnaike and Ray, 1979). The performance of this method compares well with the Zeigler-Nichols (ZN) approach (Ziegler and Nichols, 1942).

Another method which is used to tune multiloop PI/PID controllers is based on the Gershgorin band (Chen and Seborg, 2001). It enables a controller to be found by calculating the ultimate point of the Gershgorin band and deriving the ultimate gain and ultimate frequency at that point. The method only applies to plants with a diagonally dominant matrix, otherwise the system needs to have a pre-compensator matrix (Hawkins, 1972), (Ford and Daly, 1979) before the PI/PID controller can be applied. The PI/PID action is tuned based on modified Ziegler-Nichols rules (Ziegler and Nichols, 1942). However this method needs to have a very accurate system model in order to guarantee that the PI/PID controller achieves satisfactory performance. Additionally, the method does not take into account the interaction, hence the result may be unpredictable. However the method appears to be an improvement on the ZN method and BLT based on the impressive results presented in the paper.

Another choice for tuning PI/PID for the MIMO system case is explained in (Xiong et al., 2007). The method is based on the effective energy transmission ratio array, effective relative gain array, relative frequency array, these terms all being defined in (Xiong et al., 2007), so that an equivalent transfer-function matrix for the closed-loop

system can be derived with stable pairs selected based on the information above. An off-diagonal controller is designed to achieved a satisfactory response. The method has been simulated and applied to control an industrial-scale polymerization reactor (Chien et al., 1999) and a binary ethanol-water system of a pilot-plant distillation column (Ogunnaike et al., 1983), and results show the method improves upon other PID control methods. The performance achieved is of a high level especially in terms of settling time and maximum overshoot.

The use of PI/PID control methods is very common for both SISO and MIMO systems because of their relative simplicity and robustness. However these methods often need an accurate system model especially in the MIMO case and performance degrades in the presence of any external disturbance and plant uncertainty, which always exist in reality. Thus performance similar to that found in simulation is hard to achieve.

2.3 Iterative learning control methods for SISO system

Iterative Learning Control (ILC) was formally conceived over 25 years ago, and has become an area of considerable research interest in both theoretical and application domains. ILC is suitable for systems which perform a repeated process defined over a finite time interval, termed a trial. It uses data recorded over previous trials to modify the control signal of the subsequent trial with the aim of sequentially improving tracking accuracy. It has been combined with many other methodologies including adaptive, model predictive, optimal, neural, fuzzy and robust control. In this subsection, ILC theory will be reviewed from basic to advanced approaches, with a focus on those approaches which have been implemented in practice.

2.3.1 Simple-structure ILC approaches

ILC is a method suitable for system which perform a tracking task over a finite, time interval. One finished, the system is reset to the same starting point and the task is repeated. The basic principle of ILC is to use the tracking error recorded over the previous trial to modify the control signal applied on the next so that the error of the system can be reduced ideally to zero as the number of trials increases. The concept of ILC can be found in a U.S patent (Garden, 1971) and later Cryer et al. (1976) proposed a discrete-time iterative controller. The first widely accepted publication on ILC is written in Japanese (Uchiyama, 1978) but it was not until 1984 that the method found significant research interest in the series of publications (Craig, 1984), (Arimoto et al., 1984).

Amongst the first papers to exploit the concept of ILC is (Arimoto et al., 1985) which presents an algorithm in which the update control input $\mathbf{u}_{k+1}(t)$ is equal to sum of the previous trial control input $\mathbf{u}_k(t)$ and a term dependent upon the previous error $\mathbf{e}_k(t)$.

The error of each trial is given by the difference between the reference signal or demand output $\mathbf{y}_d(t)$ and the real output of the system $\mathbf{y}_k(t)$. The update and error are given by

$$\mathbf{u}_{k+1}(t) = \mathbf{u}_k(t) + L\mathbf{e}_k(t) \tag{2.21}$$

$$\mathbf{e}_k(t) = \mathbf{y}_d(t) - \mathbf{y}_k(t) \tag{2.22}$$

Here k is the trial number and time $t \in [0, T]$. This algorithm is termed proportional ILC (P-type ILC) since the control input is related to the error via the constant L which affects the convergence speed of the algorithm. If the L constant is too big the system can be unstable, and if it is too small the rate of convergence is very slow. In accordance with the ILC paradigm, the system should have the same initial condition at each operation trial and in this case the dynamics are assumed time-invariant.

Another ILC algorithm is called derivative (D-type) ILC and is given as

$$\mathbf{u}_{k+1}(t) = \mathbf{u}_k(t) + L\dot{\mathbf{e}}_k(t) \tag{2.23}$$

Instead of the error, in this case, the rate of change of the error $\dot{\mathbf{e}}_k(t)$ is considered and is multiplied by a constant L before being adding to the previous control input $\mathbf{u}_k(t)$. As described in (Arimoto et al., 1984), (Arimoto et al., 1985) the condition for the error norm to converge to zero in the SISO case is

$$\mid I - CBL \mid < 1 \tag{2.24}$$

$$\mathbf{y}_d(0) = C\mathbf{x}^0 \tag{2.25}$$

Here the initial system state should be the same for each trial. C, B are continuous-time state-space model matrices and L is a scalar gain. Furthermore, $\mathbf{u}_0(t)$, $\mathbf{y}_d(t)$ must be continuously and differentiable on $[0, T]$. The value of CB should be different to zero so that the controller always exist a value L satisfy condition (2.24). When CB equals to zero a lifting method, as in Wijdeven and Bosgra (2008), is used so that model based ILC can force the system to track reference signals. However the method does not care about the performance during trials. If controlled system require a smooth operation during trials the method can give unsatisfied results. Another method is implemented by linear matrix inequalities Hladowski et al. (2010) allows the design of robust control laws in the presence of uncertainty in the dynamics produced along the trials. The method is also applied on a gantry robot performing a pick and place operation.

A further ILC method comprises the combination of the previous two methods, and is termed PD-type ILC. It is given by

$$\mathbf{u}_{k+1}(t) = \mathbf{u}_k(t) + (\Phi + L\frac{d}{dt})\mathbf{e}_k(t) \tag{2.26}$$

where Φ, L are scalar values which should be chosen carefully to ensure stability. These simple approaches exhibit satisfactory convergence properties for systems which have low relative degree (either 0 or 1). With high order systems the method often cannot achieve satisfactory performance since the convergence conditions are not met. Another issue is their use of the error derivative which is susceptible to high levels of noise. Clearly, practical conditions heavily impact on all algorithms. In order to fill the gap between theory and practical implementation, much effort has been expended on analysing different types of ILC algorithm in laboratory conditions. One such example is a non-minimum phase facility (Freeman et al., 2005) which has been used to test a large number of ILC approaches including D-type, P-type, phase-lead algorithm and more advanced model-based methods such as gradient ILC, norm-optimal ILC and Newton method based ILC. The papers (Bristow et al., 2006), (Ahn et al., 2007) provide a thorough review of ILC and its applications .

A simple extension to the P-type ILC is termed phase-lead ILC (Park et al., 1998). This method has been implemented on several applications and has led to a high level of performance provided the phase-lead value is chosen carefully. The phase-lead ILC can be written as

$$\mathbf{u}_{k+1}(t) = \mathbf{u}_k(t) + L\mathbf{e}_k(t+\delta) \tag{2.27}$$

$$\mathbf{e}_k = \mathbf{y}_d(t) - \mathbf{y}_k(t) \tag{2.28}$$

where δ is the phase-lead of the error signal which assumed to be the number of samples which, when shifting the demand forward, minimised the difference between the demand and the system output. When $\delta = 0$, phase-lead ILC equates to P-type ILC. One advantage of phase-lead method is that the optimum phase-lead does not change when the demand is altered. In order to use this and previous methods to control a given system, the continuous-time algorithms are converted into discrete updates. Therefore the discrete version of (2.27) can be written as

$$\mathbf{u}_{k+1}(i) = \mathbf{u}_k(i) + L\mathbf{e}_k(i+\delta) \tag{2.29}$$

$$\mathbf{e}_k(i) = \mathbf{y}_d(i) - \mathbf{y}_k(i) \qquad i = 0, 1, ...N \tag{2.30}$$

where i is the discrete-time instant and δ is now the integer phase-lead in samples. Similarly, N, is the trial length in sample. The discrete-time delay methods, P-type ILC, D-type ILC were implemented to control the same non-minimum phase system test facility (Freeman, 2004). The results show that phase-lead ILC outperforms P-type and D-type over 20 times in terms of normalized error. Similarly, the same method was applied on an industrial robot, SEIKO TT3000, in (Ye and Wang, 2003) and gave rise to improved results compared to P-type and D-type ILC. In addition, many types of filter were tested on the same system to provide a comparison between different techniques.

Generally, ILC method takes advantage of non-causal information available through previous trials to change controller signal so that the tracking ability can be better and better after a number of trials in finite time interval.

In the next sections, advanced ILC methods will be introduced in detail and performance comparisons will be drawn between ILC approaches.

2.3.2 Newton method based ILC

The proportional-type ILC law (2.21) may be regarded as an iterative scheme for minimisation of a cost involving the system error. It therefore follows that greater performance may be achieved through use of more advanced iterative optimisation techniques. The Newton method based ILC was proposed in (Avrachenkov, 1998) for continuous-time with convergence condition and later the same method was derived for discrete-time nonlinear systems (Lin et al., 2006), which contains full derivation and simulation results. In general, convergence of Newton method based ILC is very fast since its rate is quadratic rather than linear. The approach is based on the Newton method or Newton-Raphson (Kincaida and Cheney, 2002) method which can written as

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \tag{2.31}$$

where n is the trial number, x is the estimated root of $f(x)$ and $f(x)$ is a known function of x. The notation $f'(x)$ stands for the derivative of f with respect to x. The method is feasible if the function $f'(x)^{-1}$ exists. Another drawback of this method is that it may take a considerable effort to calculate the inverse of $f'(x)$.

To adapt equation (2.31) for the ILC framework, firstly, recall the basic proportional-type ILC equations (2.21) (2.22) and the error function $\mathbf{e}_k(t) = \mathbf{y}_d(t) - g(\mathbf{u}_k)$. Then introduce supervector notation as follows. Consider a SISO, linear time-invariant discrete plant with relative degree m and $t \in [0, 1, ..., N]$, and write in the form

$$\mathbf{y}(z) = H(z)\mathbf{u}(z) = (h_m z^{-m} + h_{m+1} z^{-(m+1)} + h_{m+2} z^{-(m+2)} + ...)\mathbf{u}(z). \tag{2.32}$$

Now introduce vectors of input, output and reference signals as follows

$$\mathbf{u}_k = [\mathbf{u}_k(0), \mathbf{u}_k(1), ..., \mathbf{u}_k(N-1)]^T \tag{2.33}$$

$$\mathbf{y}_k = [\mathbf{y}_k(m), \mathbf{y}_k(m+1), ..., \mathbf{y}_k(m+N-1)]^T \tag{2.34}$$

$$\mathbf{y}_d = [\mathbf{y}_d(m), \mathbf{y}_d(m+2), ..., \mathbf{y}_d(m+N-1)]^T \tag{2.35}$$

Then the SISO system can be described as $\mathbf{y}_k = H_p\mathbf{u}_k$, where H_p is defined as

$$H_p = \begin{bmatrix} h_m & 0 & 0 & \dots & 0 \\ h_{m+1} & h_m & 0 & \dots & 0 \\ h_{m+2} & h_{m+1} & h_m & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ h_{m+N-1} & h_{m+N-2} & h_{m+N-3} & \dots & h_m \end{bmatrix} \tag{2.36}$$

In (Lin et al., 2006) a general form of system description suitable for nonlinear plants is assumed, $g(\mathbf{u}_k) = \mathbf{y}_k$. Note that for the linear SISO above $g(\mathbf{u}_k) = H_P\mathbf{u}_k$. Suppose the P-type update has applied to the plant, hence substitute $g(\mathbf{u}_k)$ into (2.21) to give

$$\mathbf{u}_{k+1} = \mathbf{u}_k + L(\mathbf{y}_d - g(\mathbf{u}_k)) \tag{2.37}$$

If the scalar L is replaced by the operator $(\frac{d}{d\mathbf{u}_k}(\mathbf{y}_d - g(\mathbf{u}_k)))^{-1}$, the result equates to the Newton method where the minimised quantity is the error. The derivative of $\frac{d}{d\mathbf{u}_k}(\mathbf{y}_d - g(\mathbf{u}_k))$ is given by

$$\frac{d}{d\mathbf{u}_k}(\mathbf{y}_d - g(\mathbf{u}_k)) = -g'(\mathbf{u}_k) \tag{2.38}$$

Consequently the Newton method based ILC is expressed as

$$\mathbf{u}_{k+1} = \mathbf{u}_k + g'(\mathbf{u}_k)^{-1}\mathbf{e}_k \tag{2.39}$$

$$\mathbf{e}_k = \mathbf{y}_d - \mathbf{y}_k \tag{2.40}$$

This method inherits the fast convergence from the Newton method in a region sufficiently close to the real solution but it will be valid only when the inverse function $g'(\mathbf{u}_k)^{-1}$ exists. However this problem can be solved by writing the Newton method based ILC update in the form

$$\mathbf{u}_{k+1} = \mathbf{u}_k + w_{k+1} \tag{2.41}$$

and computing $w_{k+1} = g'(\mathbf{u}_k)^{-1}\mathbf{e}_k$ by solving the equation

$$g'(\mathbf{u}_k)w_{k+1} = \mathbf{e}_k \tag{2.42}$$

The term $g'(\mathbf{u}_k)$ is a linear time-varying system, and so this can be achieved via ILC. This also means calculation of the inverse is avoided. In (Lin et al., 2006) this method is implemented for a non-linear discrete system and there is a comparison between different choices of initial values. The same method has been applied on patients to help them recover after stroke (Davies et al., 2008) and (Cai et al., 2011b).

2.3.3 Gradient ILC

The gradient method has been proposed in the ILC literature for LTI systems such as (Kinosita et al., 2002) for continuous-time or (Owens et al., 2009) for discrete-time, but here it is expanded to the nonlinear case (Freeman, 2011) using the framework introduced previously for the Newton method-based approach. Let a non-linear discrete-time system be defined as

$$\begin{aligned} \mathbf{x}(i+1) &= f(\mathbf{x}(i), \mathbf{u}(i)) \\ \mathbf{y}(i) &= h(\mathbf{x}(i), \mathbf{u}(i)) \qquad \mathbf{x}(0) = \mathbf{x}_0 \end{aligned} \tag{2.43}$$

where $\mathbf{x}$ is the state vector and $f(.)$ and $h(.)$ are vector functions of $\mathbf{x}$ and $\mathbf{u}$ which are continuously differentiable. The system is defined at the sample times $i = 0, 1, 2, ...N-1$. The input and output vectors of the system are given as

$$\mathbf{u} = [\mathbf{u}(0), \mathbf{u}(1), \mathbf{u}(2), ..., \mathbf{u}(N-1)]^T \tag{2.44}$$

$$\mathbf{y} = [\mathbf{y}(0), \mathbf{y}(1), \mathbf{y}(2), ..., \mathbf{y}(N-1)]^T \tag{2.45}$$

The system (2.43) can be replaced by an algebraic function $g(\mathbf{u})$ such that $\mathbf{y} = g(\mathbf{u})$. Accordingly, the components of function $g(.)$ can be written as

$$g(.) = [g_0(.), g_1(.), g_2(.), ..., g_{N-1}(.)]^T \tag{2.46}$$

The desired reference signal is defined as $\mathbf{y}_d = [\mathbf{y}_d(0), \mathbf{y}_d(1), \mathbf{y}_d(2), ..., \mathbf{y}_d(N-1)]$. Therefore over each trial the relationship between the input output time-series is expressed as

$$\begin{aligned} \mathbf{y}(0) &= h(\mathbf{x}(0), \mathbf{u}(0)) = g_0(\mathbf{x}(0), \mathbf{u}(0)) \\ \mathbf{y}(1) &= h(\mathbf{x}(1), \mathbf{u}(1)) = h(f(\mathbf{x}(0), \mathbf{u}(0)), \mathbf{u}(1)) \\ &= g_1(\mathbf{x}(0), \mathbf{u}(0), \mathbf{u}(1)) \\ &\vdots \\ \mathbf{y}(N-1) &= h(\mathbf{x}(N-1), \mathbf{u}(N-1)) \\ &= h(f(\mathbf{x}(N-1), \mathbf{u}(N-2)), \mathbf{u}(N-1)) \\ &= g_{N-1}(\mathbf{x}(0), \mathbf{u}(0), \mathbf{u}(1), ..., \mathbf{u}(N-1)) \end{aligned}$$

Hence the error of the system can be written as

$$\|\mathbf{e}\|_2^2 = \|\mathbf{y}_d - g(\mathbf{u})\|_2^2 = f(\mathbf{u}) \tag{2.47}$$

The control signal $\mathbf{u}$ should be chosen such that the error function robustly converges to zero. An alternative to the Newton method is the gradient descent method to solve

$\min_{\mathbf{u}} \|\mathbf{y}_d - g(\mathbf{u})\|_2^2$. Applying this within the ILC framework yields

$$\begin{aligned}
\mathbf{u}_{k+1} &= \mathbf{u}_k - \beta \nabla f(\mathbf{u}_k) && (2.48)\\
&= \mathbf{u}_k + \beta \nabla (\mathbf{y}_d - g(\mathbf{u}_k)) && (2.49)\\
&= \mathbf{u}_k + \beta g'(\mathbf{u}_k)^T (\mathbf{y}_d - g(\mathbf{u}_k)) && (2.50)\\
&= \mathbf{u}_k + \beta g'(\mathbf{u}_k)^T \mathbf{e}_k && (2.51)
\end{aligned}$$

where β is a scalar gain. The derivative of the algebraic function $g(u)$ is given as

$$g'(\mathbf{u}) = \begin{bmatrix}
\frac{\partial g_0}{\partial \mathbf{u}(0)} & 0 & 0 & \dots & 0 \\
\frac{\partial g_0}{\partial \mathbf{u}(0)} & \frac{\partial g_1}{\partial \mathbf{u}(1)} & 0 & \dots & 0 \\
\frac{\partial g_0}{\partial \mathbf{u}(0)} & \frac{\partial g_1}{\partial \mathbf{u}(1)} & \frac{\partial g_2}{\partial \mathbf{u}(2)} & \dots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
\frac{\partial g_0}{\partial \mathbf{u}(0)} & \frac{\partial g_1}{\partial \mathbf{u}(1)} & \frac{\partial g_2}{\partial \mathbf{u}(2)} & \dots & \frac{\partial g_N}{\partial \mathbf{u}(N-1)}
\end{bmatrix} \tag{2.52}$$

Therefore $g'(\mathbf{u}_k)^T \mathbf{e}_k$ is the vector that determines the direction of the update vector. Previously, this update has only been considered for the LTI SISO case (Hätönen et al., 2003) in which case it becomes $\mathbf{u}_{k+1} = \mathbf{u}_k + \beta G^T \mathbf{e}_k$ with

$$G = \begin{bmatrix}
0 & 0 & 0 & \dots & 0 \\
CB & 0 & 0 & \dots & 0 \\
CAB & CB & 0 & \dots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
CA^{N-1}B & CA^{N-2}B & CA^{N-3}B & \dots & 0
\end{bmatrix} \tag{2.53}$$

The matrix G contains Markov parameters which can be obtained from the finite impulse response (FIR) of the plant. It can always be shown that there exists a scalar $\beta > 0$ which ensures convergence to zero for the nominal plant. In fact β can be designed to optimise the convergence rate. Using the trial varying value

$$\beta_{k+1} = \frac{\|G^T \mathbf{e}_k\|_2^2}{\|GG^T \mathbf{e}_k\|_2^2} \tag{2.54}$$

The proof that this gives the maximum convergence rate can be found in (Owens et al., 2009) but is for the SISO case only. The gradient ILC method has been implemented to control a gantry robot (Ratcliffe et al., 2006), hydraulic servo system (Hätönen and Owens, 2004) and a non-minimum phase test facility (Freeman et al., 2007c). The same method has been simulated with different models such as a flexible arm robot (Kinoshita et al., 2002) and a helicopter model (Ogoshi et al., 2002). The results of these papers all show a high rate of convergence of the norm error which is low after a small number of trials, although in some cases the method is slow. The papers (Dinh et al., 2012b,a) show that the gradient method can be extended for MIMO systems with an optimal constant β which enables the system to converge to zero norm as rapidly as possible.

2.3.4 Norm optimal ILC

Norm optimal ILC involves the minimisation of a performance index J in order to calculate the optimum control input $\mathbf{u}$ for each trial. Consider the discrete-time state-space model of a system

$$\mathbf{x}(i+1) = A\mathbf{x}(i) + B\mathbf{u}(i) \quad \mathbf{x}(0) = \mathbf{x}_0 \tag{2.55}$$

$$\mathbf{y}(i) = C\mathbf{x}(i) \quad \text{where} \quad x \in \mathbb{R}^n, \quad \mathbf{u} \in \mathbb{R}^m, \quad \mathbf{y} \in \mathbb{R}^p \tag{2.56}$$

where $i = 0, 1, \ldots N-1$. The state-space matrices A, B, C are assumed to be time-invariant for simplicity although the approach can also be applied to linear time-varying plants. The cost function following the k^{th} trial is given by

$$J_{k+1}(\mathbf{u}_{k+1}) = \|\mathbf{e}_{k+1}\|_Q^2 + \|\mathbf{u}_{k+1} - \mathbf{u}_k\|_R^2 \tag{2.57}$$

Where $\|.\|_w$ denotes the 2-norm with an inner product weighting w and $\mathbf{e}_{k+1} = \mathbf{y}_d - \mathbf{y}_{k+1}$. Using the more familiar formulation where the norms are written out as sums (Amann et al., 1996a), (2.57) becomes

$$J_{k+1} = \sum_{t=1}^{N} [\mathbf{y}_d(i) - \mathbf{y}_{k+1}(i)]^T Q(i) [\mathbf{y}_d(i) - \mathbf{y}_{k+1}(i)] + \sum_{t=0}^{N-1} [\mathbf{u}_{k+1}(i) - \mathbf{u}_k(i)]^T R(i) [\mathbf{u}_{k+1}(i) - \mathbf{u}_k(i)] \tag{2.58}$$

where $Q(i)$, $R(i)$ are weighting functions which are usually assumed static. The optimum control input $\mathbf{u}_{k+1}$ is found by setting the derivative respect to the control input to zero for each time instant.

$$\frac{\partial J_{k+1}}{\partial \mathbf{u}_{k+1}} = -G^T Q \mathbf{e}_{k+1} + R(\mathbf{u}_{k+1} - \mathbf{u}_k) = 0 \tag{2.59}$$

where the matrix $G \in \mathbb{R}^{(pN)\times(mN)}$ is given by

$$G = \begin{bmatrix} CB & 0 & \ldots & 0 \\ CAB & CB & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{N-1}B & CA^{N-2}B & \ldots & CB \end{bmatrix} \tag{2.60}$$

and Q, R are diagonal matrices with $Q(i)$, $R(i)$ on the leading diagonal. Therefore

$$\mathbf{u}_{k+1} = \mathbf{u}_k + R^{-1} G^T Q \mathbf{e}_{k+1} \tag{2.61}$$

For simplicity, denote $R^{-1}G^TQ$ as G^* so the final update is

$$\mathbf{u}_{k+1} = \mathbf{u}_k + G^*\mathbf{e}_{k+1} \tag{2.62}$$

The approach has been tested on an LTI (linear time-invariant) SISO system in (Ratcliffe et al., 2006), but it can be used for the general MIMO case (Amann et al., 1996b), (Amann et al., 1998). The non-causal term $G^*\mathbf{e}_{k+1}$ is not known at the start of trial $k+1$, but can be generated through use of feedback and feed-forward actions involving state estimation and the solution to a Ricatti equation (Polyanin and Zaitsev, 2005). This method was implemented on a non-minimum phase system and led to a high level of performance. Another application class, called molding processes (Gao et al., 2001), used an ILC method based on a similar optimal theory but with a slight difference; a new parameter was introduced $\rho = \frac{Q}{R}$ to ensure that the norm error converged to zero in any situation even when the initial position after the resetting process was not the same for each trial. Norm optimal ILC has also been applied to accelerator based free electron lasers in (Rogers et al., 2010), (Kichhoff et al., 2008). In this case NOILC did not reach the target performance for the application but the method used only 10 trials to reach a very small error value. Extensions have been proposed using a predictive mechanism Bristow and Alleyne (2003), constraints Chu and Owens (2010), projections Chu and Owens (2009), and to address point-to-point tracking Owens et al. (2012). If the application permits a large enough number of trials to be carried out, the method typically enables the system to reach the preset target.

2.4 ILC applications

Compared to other control methods, ILC has demonstrated clear advantage when applied to applications which involve repetitive tasks. When applied to systems with uncertainty and disturbance, ILC has shown the ability to give outstanding performance (Bristow et al., 2006), (Ahn et al., 2007). Due to these qualities, ILC has been used in many industrial areas. CNC machining (Kim and Kim, 1996) is one example, where ILC improved the tracking performance compared to a PID controller. In this case PID-type ILC was implemented in the discrete-time domain using a microprocessor. Under sufficient conditions the method guarantees convergence to zero error after a small number of trials. Similarly, the same technique has been applied to micro-scale robotics (Bristow and Alleyne, 2003) which are used for manufacturing photonic band gap (PBG) material. The application requires a highly accurate micro-scale (0.1μm-100μm) movement from the machine. The system not only needs an accurate controller but also all the servo and movement components need to be very precisely manufactured without any backlash. A feed-forward ILC controller has been successful applied to the system and shown to help improve the efficiency and be more effective than other methods such as lithography and laser-induced chemical vapour deposition (Lin et al., 1998), (Wanke et al., 1998).

Such alternatives not only cost much more than the micro-scale robot, but also have a long build time and place restrictions on size. Another application to which ILC has been employed is control of linear, permanent magnet, synchronous motors (LPMSM) (Butcher and Karimi, 2010) in which a new ILC method to cope with uncertainty of the system and the disturbance was used. The novelty of this ILC implementation for linear parameter-varying plants (Shamma and Athans, 1998) is illustrated by the algorithm still converging to the reference signal even when there is a disturbance in the reset procedure, meaning that the system is not reset to the same initial location after each trial. In general, standard ILC does not work well if there is a difference in initial position or an trial-varying disturbance. The ILC controller was designed, simulated and applied to LPMSM, and gave a better result compared to standard gradient ILC (Owens et al., 2009) and a PID controller. A similar problem in which the disturbance does not have exact initial reseting has been found in injection moulding processes (Gao et al., 2001), (Tan et al., 2003). The proposed method was used in polymer manufacturing techniques. It transformed polymer granules into various shapes and types of products, ranging from simple cups to precision lens and compact discs. In (Gao et al., 2001) the approach exploited an optimal norm controller to make the system track more accurately. The initial injection velocity response could not be repeated exactly, resulting in uncertainty of initialization error of the injection velocity control. Furthermore, there were disturbances during the moulding process from different sources, such as variation of the material or operating conditions. The method uses a modified norm optimal ILC method with a new variable, ρ, which has the following relationship with other components

$$R = \lambda I \tag{2.63}$$

$$Q = \mu I \tag{2.64}$$

$$\rho = \frac{\mu}{\lambda} \tag{2.65}$$

Here Q and R are the weighting matrices used in optimal control which must be selected carefully, I is the identity matrix and μ, λ is a positive design constant. These constants are used to ensure the controller can reject uncertain disturbances as well as track the desired reference with rapid convergence. Another modified ILC algorithm, called H_∞ ILC, was introduced in (Roover, 1996) and applies ILC in the H_∞ framework. The control input comprises the combination of current error feedback and past error feedforward components. The method has been successfully applied to control of a wafer stage (Roover and Bosgra, 2000) and the setup experiment is fully described in (Roover, 1997). As is usual, the application needs a very high accuracy to track the reference signal. As a result, the method gives outstanding performance within 5-10 iterations and the norm error can reach close to zero. Many other applications use ILC to cope with disturbance and plant uncertainty such as cold rolling mills (Garimella and Srinivasan, 1998), antilock braking of electric and hybrid vehicles (Mi et al., 2005) and underwater robot manipulators (Kawamura and Sakagami, 2002), (Sakagami et al., 2002), (Sakagami

and Kawamura, 2003) and acrobat robot (Yamakita et al., 2002), (Watabe et al., 2002).

ILC algorithms are not only used to control mechanical applications but also can be used in controlling temperature, such as in a batch polymerization reactor (Lee et al., 1996) and extruder (Pandit and Buchheit, 1999). For the extruder application, researchers have used an optimal ILC which has a performance index Q defined by the following function of the input and output

$$Q[\mathbf{u}_{k+1}(t), \mathbf{y}_{k+1}(t)] = \int_0^T L(\mathbf{u}_{k+1}(t), \dot{\mathbf{u}}_{k+1}(t), ..., \mathbf{y}_{k+1}(t), \dot{\mathbf{y}}_{k+1}(t), ..., \mathbf{y}_d(t))dt \quad (2.66)$$

Here $\mathbf{u}, \mathbf{y}$ are the input and output of the system respectively, k is the trial number, t is the time variable and L is a function which is chosen to stipulate behavior of $\mathbf{u}$ and $\mathbf{y}$. Therefore in order to provide the optimum performance $\mathbf{u}, \mathbf{y}$ need to correspond to the minimum point of the function Q. The result of this method is improved with the aid of a non-causal filter and can reduce the extrusion time to about 10% compared to classical controllers. With the same control objective, the polymerization reactor application is a combination of a sequence of charge, heat-up, reaction and discharge phases. In particular, in the heat-up and reaction phases, the temperature variable requires a fast response time as well as a fast settling time but low overshoot magnitude. ILC was therefore applied to enable the reactor temperature to settle down to the set point in the minimum time while rejecting disturbances. The system uses a filter and feedback controller and a learning filter. If $H_1 = P^{-1}$, where H_1 is the learning filter and P is the model of plant, the control method is similar to the inverse method ILC which makes the system converge to the target reference quickly. Experimental results show very precise tracking even when it is designed based on a nominal process model. Despite the simplicity of the method, it has been applied to industrial batch reactors producing ABS polymers and has been in successful operation for more than 3 years.

Recently, ILC has been used to aid stroke patients rehabilitate their upper limbs Freeman et al. (2007b), Freeman et al. (2007a), Freeman et al. (2009c), Freeman et al. (2012). The experimental facility is under researching and develop in order to transfer the technology to all hospitals. The system is improving to have a cheap version for household using also. The facility uses functional electrical stimulation to help patients moving their upper limb to do specific tasks. Therefore they have motivation to practice everyday. The system was improved tracking ability by using ILC controller Hughes et al. (2009c), Hughes et al. (2009a) and adjoint, phase-lead were used in Freeman et al. (2008), Freeman et al. (2009a) and Freeman et al. (2011a) is for 3D stroke rehabilitation facility. Interview patients also gave positive comments and they would like to have further testing sessions Hughes et al. (2009b). Each patient has different muscle structure therefore in order to improve tracking ability of the facility a muscle model is identified such as upper extremity Freeman et al. (2009b), Le et al. (2010), Le et al. (2012), hand and wrist Soska et al. (2012). Based on derived models, more advance controller can be implemented

such as Newton method based nonlinear ILC Freeman et al. (2010) for robotic station and Cai et al. (2011c) for 3D stroke rehabilitation.

Whilst many such reported applications and cases of detailed experimental comparison and benchmarking exist for the SISO case, there are few instances involving MIMO systems Tyréus (1979); Haurani et al. (2001). These are generally more challenging due to interaction dynamics which typically increase controller demand as well as significantly complicating controller design and performance/robustness analysis. With mild interaction, one approach is to ignore the coupling and design multiple SISO controllers Wallén et al. (2008), however loss of performance is inevitable. In addition to a lack of MIMO application examples there exists no comparative benchmarking between algorithms, critical for thorough performance assessment prior to wider industrial implementation. To solve this problem, a multi-configurable experimental test facility is needed to enable MIMO ILC approaches to be rigorously evaluated.

In summary, ILC has been designed and implemented on many different types of control problem. Depending on each specific application, a ILC controller is chosen to achieve an appropriate objective. Experimental results confirm that ILC can yield satisfactory results despite significant modeling error. However it is clear that as accurate a model as possible is necessary to guarantee the best performance.

2.5 Summary

A review of MIMO control approaches has been conducted, centering on practical performance needs. In addition to feedback based techniques, a summary of ILC approaches has been presented, focusing on optimisation based methods, and those approaches which have been applied experimentally.

However, there is a little emphasis on MIMO system, in theoretical or application terms. In particular there is no analysis on the effect of interaction on performance. There is hence a pressing need for a MIMO benchmarking facility for evaluation and comparison of controllers. The MIMO facility must enable the interaction level to be adjusted, as well as provide disturbance/noise injection capability. Such a MIMO facility would give rise to a full robustness and performance characteristic of MIMO ILC.

Chapter 3

Test facility design and analysis

The review of applications to which ILC has been applied has exposed a lack of multi-variable system studies. In this chapter an electromechanical system will be developed for the purpose of benchmarking and evaluating controllers. To cover a wide range of characteristics it will encompass multiple inputs and outputs with variable interaction between these ports.

3.1 Differential gearbox

One approach to designing such a system is to use a differential gearbox which has 3 ports and therefore, if two devices are connected together, is intrinsically multivariable. Commonly the differential gear is used in both light duty and heavy duty vehicles. The component can help cars turn in sharp corners easily especially at high speed. The gear is constructed from a variety of bevel gears and it is therefore a challenging task to determine a transfer-function using Newton-based analysis of all the forces applied to the system. Consequently, Lagrange's method will instead be used to derive the dynamic representation for this system. Firstly, however, an overview of the structure and function of a gearbox will be described before Lagrange's method is applied to this particular system.

In the automotive industry, differential gearboxes are an important component of almost every vehicle. This device is used to transmit torque from the main engine to the rear wheels and allows the vehicle to turn easily at a bend in the road and avoid skidding. Basically, the gearbox has 1 input shaft, from the engine, and 2 outputs to the rear wheels. The speed of each rear wheel is dependent on their respective inertia. If the 2 sides are balanced on a straight road, the speeds of the 2 wheels are equal, but on a bend in the road, there will be different inertia associated with each wheel. As a result, the speeds are different so that the vehicle will change direction without loss of traction. A

schematic of a gearbox is given in Figure 3.1. Shaft A is the input gear from the engine to port A and the 2 rear wheels are called B, C. The angular speed of shaft B and shaft C can be different depending on the load applied to each.

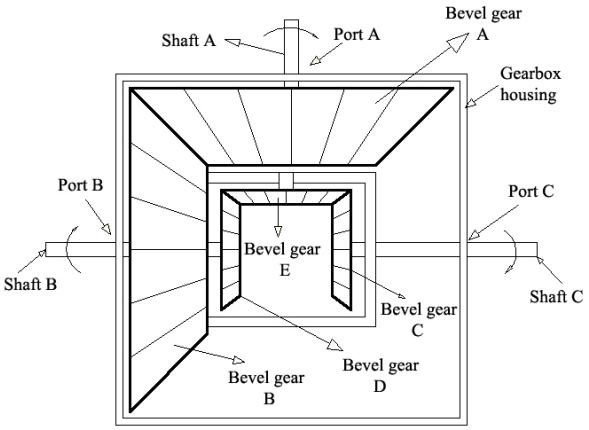


Figure 3.1: Differential gearbox

In some references the differential gearbox is presented as one of the special cases of an epicyclic gear. It can be seen that gear E is free to rotate around the housing when shaft B and shaft C have the same speed but their movements are in same directions. Otherwise, the gear E will be stationary with respect to gear D or gear C and rotate around itself if the angular speed of B, C are the same and in opposite directions.

A simple epicyclic train (Hannah and Stephens, 1963) has 3 components: the sun wheel (S), annulus (A) and planet (P) wheels which are connected together by an arm L. Details of the gearbox are shown in Figure 3.2. The planet gears (P) can rotate and are situated between the annulus and the sun wheel. Similarly, a differential gearbox has the same properties as the epicyclic gear (Holmes, 1977), (Morrison and Crossland, 1970). Comparing Figure 3.1 and Figure 3.2, the planet gears correspond to gear E in Figure 3.1, the sun wheel correspond to the gears connected to shaft B and C while the gear connected to shaft A corresponds to the annulus. The only difference is that in Figure 3.1, there is only one gear E and all gears are bevel gears and therefore they can be connected together at 90 degrees. The angular speed relationship between gears of the differential gearbox (Holmes, 1977) can be expressed as (3.1) where N_A is the speed of gear A and similarly for gear B and gear C. The structure also means that usually the number of teeth on gear B is equal to the number of teeth on gear C and the number of teeth on gear A is equal to the number of teeth on the bevel gear B. However this constraint equation can be different depending on the ratio between bevel gears.

$$2N_A = N_B + N_C \tag{3.1}$$

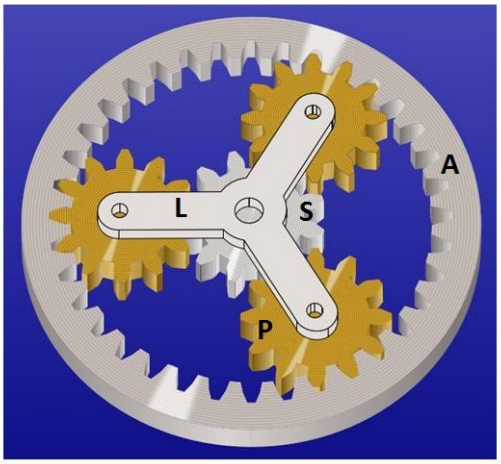


Figure 3.2: Epicyclic gear

3.2 Lagrange's equation

Lagrange's equation is one of the most prevalent methods used to derive the dynamics of a system and involves consideration of the total potential energy and kinetic energy. An alternative is the Newtonian method, which necessitates analysis of the reaction forces that are applied to a body together with a free body diagram (Close and Frederick, 1995) in order to derive an equation of motion. In contrast, Lagrange's method more concisely yields differential equations for a complex system that have many applied forces and reaction forces. The derivation and application of Lagrange's equation is fully explained in (Greenwood, 2003), (Palm, 2007), (Marion and Thornton, 1995). The method is encapsulated in the form of equation (3.2) which is called Lagrange's equation. The equations (3.2), (3.3) are applicable to holonomic systems with generalised independent coordinates $q_1, q_2 ... q_n$. The number of coordinates of a system should be chosen so that they can completely specify the state of the system. In general, there is no specific rule to choose coordinates in order to optimise the solution or simplify its derivation. Rather, their choice depends on the skill gained from experimentation and practice. The Lagrange function, is calculated as

$$L = \text{Kinetic energy} - \text{Potential energy}$$

$$\Rightarrow L = T - V$$

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = 0 \qquad (i = 1, 2...n) \tag{3.2}$$

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_i}\right) - \frac{\partial T}{\partial q_i} + \frac{\partial V}{\partial q_i} = 0 \qquad (i = 1, 2, ...n) \tag{3.3}$$

Here V denotes the total potential energy, T the total kinetic energy, q_i is the set of generalised coordinates, n is the total number of coordinates. The constraint coordinate function can be derived through differentiation of the constraint equations. If a portion of the applied generalised force is not obtained from a potential function, the equation above can be written as (3.4). This type of system is termed a nonholonomic system and has external forces applied to it.

$$\frac{d}{dt}\left(\frac{\partial T}{\partial q_i'}\right) - \frac{\partial T}{\partial q_i} + \frac{\partial V}{\partial q_i} = Q_i' \qquad (i = 1, 2, ...n) \tag{3.4}$$

Q_i' represents the applied forces on the system that are not derived from the potential energy. Here $Q_i = -\frac{\partial V}{\partial q_i} + Q_i'$. When considering nonholonomic systems, there may be more generalised coordinates than degrees of freedom. Therefore, the generalised coordinates are not completely independent and there will be generalised constraint forces C_i which are nonzero given by

$$C_i = \sum_{j=1}^{m} \lambda_j a_{ji} \qquad (i = 1, 2...n) \tag{3.5}$$

The Lagrange multiplier λ_j applies equally to all components a_{ji}. A single constraint force can be expressed as $C_j = \lambda_j a_j$. Assume that there are m holonomic constraint equations and n generated coordinates. The constraint equations then has the form

$$\phi_j(q, t) = 0 \quad (j = 1...m) \tag{3.6}$$

Therefore the coordinates a_{ji} can be calculated as

$$a_{ji}(q, t) = \frac{\partial \phi_j}{\partial q_i} \quad (i = 1, 2...n) \quad (j = 1...m) \tag{3.7}$$

To give

$$\frac{d}{dt}\left(\frac{\partial T}{\partial q_i'}\right) - \frac{\partial T}{\partial q_i} + \frac{\partial V}{\partial q_i} = \sum_{j=1}^{m} \lambda_j a_{ji} \qquad (i = 1, 2, ...n) \tag{3.8}$$

In a system which has dissipated energy, F, the final Lagrange's equation is given by

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_i}\right) - \frac{\partial T}{\partial q_i} + \frac{\partial V}{\partial q_i} + \frac{\partial F}{\partial \dot{q}_i} = Q_i' + \sum_{j=1}^{m} \lambda_j a_{ji} \qquad (i = 1, 2, ...n) \tag{3.9}$$

3.3 Gearboxes and transfer-functions

As described previously, Lagrange's equation can be used to derive dynamic equations for a mechanical system. The system shown in Figure 3.3 is now considered, and it will be assumed that there is no friction between gears. The variables θ_A, θ_B and θ_C are used to denote the output angular displacements. Each port of the gearbox has its own inertia,

damping and applied torque. Port A has inertia I_A, damping B_A and applied torque T_A. Port B has inertia I_B, damping B_B and applied torque T_B. Port C has inertia I_C, damping B_C and applied torque T_C. The differential gearbox has been discussed in the previous section. In this section, the transfer-function for this gearbox will be derived and used to find expressions for different arrangements of a composite system. These

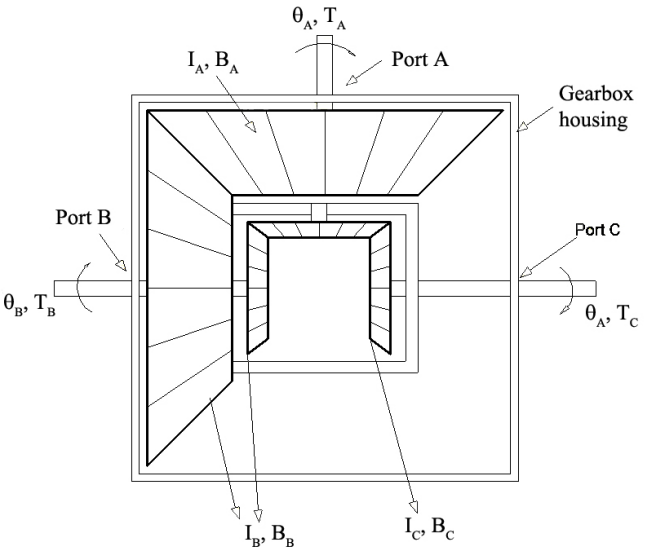


Figure 3.3: Differential gearbox

can then be analysed and simulated in Matlab.

To apply Lagrange's equation to the gearbox, Figure 3.3 requires kinetic and potential expressions, T and V. The former is given by

$$\text{Kinetic energy (T)} = \frac{1}{2}I_A\dot{\theta}_A^2 + \frac{1}{2}I_B\dot{\theta}_B^2 + \frac{1}{2}I_C\dot{\theta}_C^2$$

where I_A,I_B, I_C are the inertias of input A, output B and output C respectively. The potential energy for a rotational object can be considered to be zero. The terms θ_A, θ_B, θ_C denote the angular displacement of input A, output B and output C respectively and comprise the generalised system coordinates. As a result, $q_1 = \theta_A, q_2 = \theta_B, q_3 = \theta_C$. The system does not have any potential energy, so V can be considered to be zero. This yields

$$L = \frac{1}{2}I_A\dot{\theta}_A^2 + \frac{1}{2}I_B\dot{\theta}_B^2 + \frac{1}{2}I_C\dot{\theta}_C^2 \tag{3.10}$$

Applied partial differentiation of L gives

$$\frac{\partial L}{\partial \dot{\theta_A}} = I_A\dot{\theta}_A; \frac{\partial L}{\partial \dot{\theta}_B} = I_B\dot{\theta}_B; \frac{\partial L}{\partial \dot{\theta}_C} = I_C\dot{\theta}_C. \tag{3.11}$$

$$\Rightarrow \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_A}\right) = I_A\ddot{\theta}_A; \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_B}\right) = I_B\ddot{\theta}_B; \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_C}\right) = I_C\ddot{\theta}_C. \tag{3.12}$$

The dissipated energy can be calculated for the system:

$$\text{Dissipated energy} = F = \frac{1}{2}B_B\dot{\theta}_B^2 + \frac{1}{2}B_C\dot{\theta}_C^2 + \frac{1}{2}B_A\dot{\theta}_A^2 \tag{3.13}$$

Which yields

$$\frac{\partial F}{\partial \dot{\theta}_A} = B_A\dot{\theta}_A; \frac{\partial F}{\partial \dot{\theta}_B} = B_B\dot{\theta}_B; \frac{\partial F}{\partial \dot{\theta}_C} = B_C\dot{\theta}_C; \tag{3.14}$$

The constraint equation is given by

$$2\theta_A = \theta_B + \theta_C \tag{3.15}$$

Differentiate equation (3.15) with respect to time to give

$$2\dot{\theta}_A = \dot{\theta}_B + \dot{\theta}_C \tag{3.16}$$

$$\Rightarrow \dot{\theta}_B + \dot{\theta}_C - 2\dot{\theta}_A = 0$$

Therefore from equation (3.7) values of a_{ji} are given as $a_A = -2, a_B = 1, a_C = 1$. On each input or output of this system, torques may be applied and are denoted by T_A, T_B, T_C. These torques correspond to Q_i'. Therefore using Lagrange's equation (3.9) the following set of equations which describe the dynamics of the gearbox is obtained

$$I_A\ddot{\theta}_A + B_A\dot{\theta}_A - T_A = -2\lambda \tag{3.17}$$
$$I_B\ddot{\theta}_B + B_B\dot{\theta}_B - T_B = \lambda \tag{3.18}$$
$$I_C\ddot{\theta}_C + B_C\dot{\theta}_C - T_C = \lambda \tag{3.19}$$

Substitute λ from (3.19) into (3.17) and (3.18), in order to obtain the general dynamics for a differential gearbox given by

$$I_A\ddot{\theta}_A + B_A\dot{\theta}_A - T_A + 2I_C\ddot{\theta}_C + 2B_C\dot{\theta_C} - 2T_C = 0 \tag{3.20}$$
$$I_B\ddot{\theta}_B + B_B\dot{\theta}_B - T_B - I_C\ddot{\theta}_C - B_C\dot{\theta}_C + T_C = 0 \tag{3.21}$$
$$2\theta_A = \theta_B + \theta_C \qquad \text{(the constraint equation)} \tag{3.22}$$

These equations are next used to develop dynamic relationships for mechanical systems which incorporate gearboxes.

3.4 Case 1 arrangement and transfer-function

The first arrangement is shown in Figure 3.4 and features two motors attached to port 1 and port 2 of the differential gearbox 2. The output port A_1 of gearbox 2 is coupled with the port A_2 of the gearbox 1. The output ports B and C of gearbox 1 are connected to spring-mass-damper systems which are used to modify the system's response. Gearbox

2 has inputs T_1, T_2 and output θ_{A1}. The constraint equation for the upper gearbox is $2\theta_A = \theta_1 + \theta_2$ and the relationship between torque and angular displacement $T_A = G(s)\theta_A$ is now introduced where $G(s)$ is a continuous-time transfer-function. Using the general differential equation from the previous section, the new equations for gearbox 1 and gearbox 2 can be shown to be

Gearbox 2

$$I_{A1}\ddot{\theta}_A + B_{A1}\dot{\theta_A} - T_A + 2I_2\ddot{\theta}_2 + 2B_2\dot{\theta}_2 - 2T_2 = 0 \tag{3.23}$$

$$I_1\ddot{\theta}_1 + B_1\dot{\theta_1} - T_1 - I_2\ddot{\theta}_2 - B_2\dot{\theta}_2 + T_2 = 0 \tag{3.24}$$

$$2\theta_A = \theta_1 + \theta_2 \quad \text{(the constraint equation)} \tag{3.25}$$

Comparing these with the general differential gearbox equations, $I_A = I_{A1}$, $B_A = B_{A1}$, $I_C = I_2$, $B_C = B_2$, $I_B = I_1$, $B_B = B_1$

Gearbox 1

$$I_{A2}\ddot{\theta}_A + B_{A2}\dot{\theta_A} - T_A + 2I_{C1}\ddot{\theta}_C + 2B_{C1}\dot{\theta}_C - 2T_C = 0 \tag{3.26}$$

$$I_{B1}\ddot{\theta}_B + B_{B1}\dot{\theta}_B - T_B - I_{C1}\ddot{\theta}_C - B_{C1}\dot{\theta}_C + T_C = 0 \tag{3.27}$$

$$2\theta_A = \theta_B + \theta_C \quad \text{(the constraint equation)} \tag{3.28}$$

Again comparing these with the general case gives $I_A = I_{A2}$, $B_A = B_{A2}$, $I_C = I_{C1}$, $B_C = B_{C1}$, $I_B = I_{B1}$, $B_B = B_{B1}$. First substitute the constraint equation into equation

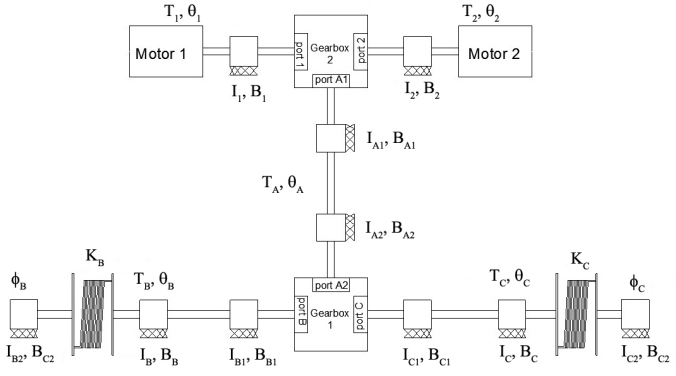


Figure 3.4: 2 differential gearboxes back to back.

(3.24). Let s be the Laplace transform variable so that $\dot{\theta} = \theta s$, and hence

$$(I_1 s^2 + B_1 s)(2\theta_A - \theta_2) - (I_2 s^2 + B_2 s)\theta_2 = T_1 - T_2 \tag{3.29}$$

$$\Rightarrow \theta_2 = \frac{T_2 - T_1 + 2(I_1 s^2 + B_1 s)\theta_A}{(I_1 + I_2)s^2 + (B_1 + B_2)s} \tag{3.30}$$

Substitution in (3.23) gives

$$(I_{A1}s^2 + B_{A1}s)\theta_A + \frac{2(I_2s^2 + B_2s)}{(I_1 + I_2)s^2 + (B_1 + B_2)s}(T_2 - T_1 + 2(I_1s^2 + B_1s)\theta_A) = 2T_2 + T_A \tag{3.31}$$

Now factorise by θ_A and substitute the value of $T_A = G(s)\theta_A$ into the equation above to give

$$\theta_A\left[(I_{A1}s^2 + B_{A1}s) + \frac{4(I_2s^2 + B_2s)(I_1s^2 + B_1s)}{(I_1 + I_2)s^2 + (B_1 + B_2)s} - G\right] =$$
$$T_1\frac{2(I_2s^2 + B_2s)}{(I_1 + I_2)s^2 + (B_1 + B_2)s} + T_2\frac{2[(I_1 + I_2)s^2 + (B_1 + B_2)s - (I_2s^2 + B_2s)]}{(I_1 + I_2)s^2 + (B_1 + B_2)s} \tag{3.32}$$

As shown in Figure 3.4, the outputs of the lower gearbox are connected to spring-mass-damper systems and therefore the transfer-functions for these loads needs to be derived first. These loads are shown in Figure 3.5 and Figure 3.6 for port B and port C respectively. Using Newton's second law and free-body diagrams to derive the transfer-function for the system in Figure 3.5, yields

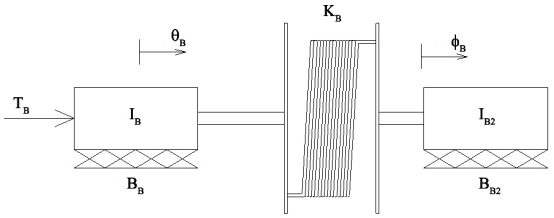


Figure 3.5: The load for output B.

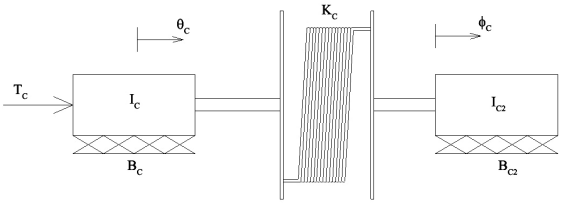


Figure 3.6: The load for output C.

$$I_B\ddot{\theta}_B + K_B(\theta_B - \phi_B) + B_B\dot{\theta}_B = T_B \tag{3.33}$$
$$I_{B2}\ddot{\phi}_B + B_{B2}\dot{\phi}_B = K_B(\theta_B - \phi_B) \tag{3.34}$$

From equation (3.34), the notation $s = \frac{d}{dt}$ is used to simplify the system and can be transfered back by inverse Laplace transform (assume zero initial condition). This yields:

$$\frac{\phi_B}{\theta_B} = \frac{K_B}{I_{B2}s^2 + B_{B2}s + K_B} \tag{3.35}$$

Substitute into equation (3.33) to get the transfer-function from T_B to θ_B

$$\frac{T_B}{\theta_B} = (I_B s^2 + B_B s + K_B) - \frac{K_B^2}{I_{B2} s^2 + B_{B2} s + K_B} \tag{3.36}$$

Similarly the transfer-functions for the system in Figure 3.6 can be derived as

$$\frac{\phi_C}{\theta_C} = \frac{K_C}{I_{C2} s^2 + B_{C2} s + K_C} \tag{3.37}$$

$$\frac{T_C}{\theta_C} = (I_C s^2 + B_C s + K_C) - \frac{K_C^2}{I_{C2} s^2 + B_{C2} s + K_C} \tag{3.38}$$

From equation (3.38) and (3.36), the value of torque that is applied on the input shaft of the gearbox should be negative and therefore:

$$T_B = \theta_B G_B(s) \tag{3.39}$$

$$T_C = \theta_C G_C(s) \tag{3.40}$$

where

$$G_B(s) := \frac{K_B^2}{I_{B2} s^2 + B_{B2} s + K_B} - (I_B s^2 + B_B s + K_B)$$

$$G_C(s) := \frac{K_C^2}{I_{C2} s^2 + B_{C2} s + K_C} - (I_C s^2 + B_C s + K_C)$$

Substitute into equation (3.27):

$$(I_{B1} s^2 + B_{B1} s - G_B)\theta_B = (I_{C1} s^2 + B_{C1} s - G_C)\theta_C \tag{3.41}$$

Using the constraint equation: $2\theta_A = \theta_B + \theta_C \Rightarrow \theta_B = 2\theta_A - \theta_C$ substitute into the equation above:

$$\theta_C \left(1 + \frac{I_{C1} s^2 + B_{C1} s - G_C}{I_{B1} s^2 + B_{B1} s - G_B}\right) = 2\theta_A \tag{3.42}$$

Substitute into equation (3.26) to get

$$\left[(I_{A2} s^2 + B_{A2} s) + \frac{4(I_{C1} s^2 + B_{C1} s - G_C)(I_{B1} s^2 + B_{B1} s - G_B)}{(I_{C1} + I_{B1}) s^2 + (B_{C1} + B_{B1}) s - (G_B + G_C)}\right] \theta_A = T_A \tag{3.43}$$

Similarly to equation (3.42), the transfer-function from θ_B to θ_A can be derived as

$$\theta_B \left(1 + \frac{I_{B1} s^2 + B_{B1} s - G_B}{I_{C1} s^2 + B_{C1} s - G_C}\right) = 2\theta_A \tag{3.44}$$

This equation gives $G(s)$ where $T_A = G(s)\theta_A$ therefore

$$\frac{\phi_B}{T_1(s)} = \frac{\theta_A}{T_1(s)}.\frac{\phi_B}{\theta_B}.\frac{\theta_B}{\theta_A} = G_{11}(s) \tag{3.45}$$

$$\frac{\phi_B}{T_2(s)} = \frac{\theta_A}{T_2(s)}.\frac{\phi_B}{\theta_B}.\frac{\theta_B}{\theta_A} = G_{12}(s) \tag{3.46}$$

$$\frac{\phi_C}{T_1(s)} = \frac{\theta_A}{T_1(s)}.\frac{\phi_C}{\theta_C}.\frac{\theta_C}{\theta_A} = G_{21}(s) \tag{3.47}$$

$$\frac{\phi_C}{T_2(s)} = \frac{\theta_A}{T_2(s)}.\frac{\phi_C}{\theta_C}.\frac{\theta_C}{\theta_A} = G_{22}(s) \tag{3.48}$$

Consequently the transfer-function for this MIMO system can be fully derived since $G_{11}(s), G_{12}(s), G_{21}(s), G_{22}(s)$ all can be calculated. Using notation $T_1(s) = \mathbf{u}_1(s)$, $T_2(s) = \mathbf{u}_2(s)$, $\phi_B(s) = \mathbf{y}_1(s)$, $\phi_C(s) = \mathbf{y}_2(s)$, therefore the transfer-function matrix for this configuration is given

$$\begin{bmatrix} \mathbf{y}_1(s) \\ \mathbf{y}_2(s) \end{bmatrix} = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \begin{bmatrix} \mathbf{u}_1(s) \\ \mathbf{u}_2(s) \end{bmatrix} \tag{3.49}$$

This matrix has been used in Matlab to simulate the system so that its response can be investigated with different parameters such as the damper constants, spring constants or inertias. As a result, the behavior of this system can be fully understood and the various impulse responses may be easily computed.

3.5 Case 2 arrangement and transfer-function

In this case, motors are placed at the side of each of the two gearboxes and deliver the torque input to the system. The mass-spring-damper components are placed at the other side of each of the gearboxes. The ports in the middle are coupled together so that they cause interaction between inputs. This arrangement is shown in Figure 3.7. The parameters I_1, B_1, T_1 respectively denote the inertia, damping and torque of port 1 of differential gearbox 2 . Similarly I_2, B_2, T_2 denotes the inertia, damper and the torque of port 2 of the differential gearbox 2 respectively. I_{A1} and B_{A2} are the inertia and damper values of the other port of gearbox 2. Similarly, I_{B1}, B_{B1}, T_B are the inertia, damper and torque of port B1 of differential gearbox 1. Likewise I_{C1}, B_{C1}, T_C are the inertia, damper, input torque of port C1 of differential gearbox 1. Finally, I_{A2}, B_{A2} are the inertia and damper values of the final port of gearbox 1. This type of arrangement is of great interest since the interaction (the dynamical connection between the 2 gearboxes) can be fully manipulated and the magnitude will depend on the value of inertia and damping I_{A1}, B_{A1}, I_{A2} and B_{A2}. Because of the new notation used, their dynamic equations, derived using Lagrange's analysis as before, are restated as

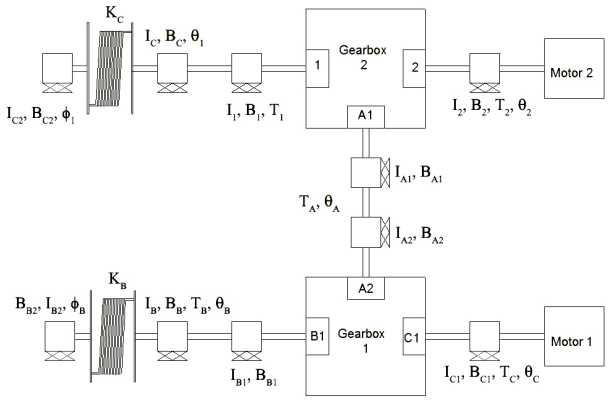


Figure 3.7: System arrangement 2.

Gear 1

$$I_{A2}\ddot{\theta_A} + B_{A2}\dot{\theta_A} - T_A + 2I_{C1}\ddot{\theta_C} + 2B_{C1}\dot{\theta_C} - 2T_C = 0 \tag{3.50}$$

$$I_{B1}\ddot{\theta_B} + B_{B1}\dot{\theta_B} - T_B - I_{C1}\ddot{\theta_C} - B_{C1}\dot{\theta_C} + T_C = 0 \tag{3.51}$$

$$2\theta_A = \theta_B + \theta_C \tag{3.52}$$

Gear 2

$$I_{A1}\ddot{\theta_A} + B_{A1}\dot{\theta_A} + T_A + 2I_2\ddot{\theta_2} + 2B_2\dot{\theta_2} - 2T_2 = 0 \tag{3.53}$$

$$I_1\ddot{\theta_1} + B_1\dot{\theta_1} - T_1 - I_2\ddot{\theta_2} - B_2\dot{\theta_2} + T_2 = 0 \tag{3.54}$$

$$2\theta_A = \theta_1 + \theta_2 \tag{3.55}$$

Note that for the upper gearbox the angular positions are defined in opposite directions to those of the lower to facilitate their connection. However the sign of T_A in equation (3.53) should be opposite to the sign of T_A in equation (3.50). Using the set of gearbox 1 equations, from equation (3.51):

$$\theta_B(I_{B1}s^2 + B_{B1}s - G_B) - \theta_C(I_{C1}s^2 + B_{C1}s) = -T_C \tag{3.56}$$

Since $T_B = G_B\theta_B$ and $\theta_C = 2\theta_A - \theta_B$, equation (3.56) becomes

$$\theta_B[(I_{B1} + I_{C1})s^2 + (B_{B1} + B_{C1})s - G_B] - 2\theta_A(I_{C1}s^2 + B_{C1}s) = -T_C \tag{3.57}$$

$$\text{To give } : \theta_A = \frac{\theta_B[(I_{B_1} + I_{C_1})s^2 + (B_{B_1} + B_{C1})s - G_B] + T_C}{2(I_{C1}s^2 + B_{C1}s)} \tag{3.58}$$

From equation (3.50)

$$\theta_A(I_{A2}s^2 + B_{A2}s) + 2\theta_C(I_{C1}s^2 + B_{C1}s) = T_A + 2T_C \tag{3.59}$$

Since $\theta_B = 2\theta_A - \theta_C$ substitute into equation (3.56) to result in

$$2\theta_A(I_{B1}s^2 + B_{B1}s - G_B) - \theta_C[(I_{C1} + I_{B1})s^2 + (B_{B1} + B_{C1})s - G_B] = -T_C \quad (3.60)$$

$$\Rightarrow \theta_C = \frac{2\theta_A(I_{B1}s^2 + B_{B1}s - G_B) + T_C}{(I_{C1} + I_{B1})s^2 + (B_{B1} + B_{C1})s - G_B} \quad (3.61)$$

Substitute equation (3.61) into equation (3.59) to obtain

$$\theta_A(I_{A2}s^2 + B_{A2}s) + \frac{4\theta_A(I_{B1}s^2 + B_{B1}s - G_B) + 2T_C}{(I_{C1} + I_{B1})s^2 + (B_{B1} + B_{C1})s - G_B}(I_{C1}s^2 + B_{C1}s) = T_A + 2T_C \quad (3.62)$$

to give

$$\theta_A\left(I_{A2}s^2 + B_{A2}s + \frac{4(I_{B1}s^2 + B_{B1}s - G_B)(I_{C1}s^2 + B_{C1}s)}{(I_{C1} + I_{B1})s^2 + (B_{B1} + B_{C1})s - G_B}\right) = T_A + 2T_C\left(1 - \frac{I_{C1}s^2 + B_{C1}s}{(I_{C1} + I_{B1})s^2 + (B_{B1} + B_{C1})s - G_B}\right) \quad (3.63)$$

The values of G_{A1} and G_C are then defined as

$$G_{A1} := (I_{A2}s^2 + B_{A2}s) + \frac{4(I_{B1}s^2 + B_{B1}s - G_B)(I_{C1}s^2 + B_{C1}s)}{(I_{C1} + I_{B1})s^2 + (B_{B1} + B_{C1})s - G_B} \quad (3.64)$$

$$G_C := 1 - \frac{I_{C1}s^2 + B_{C1}s}{(I_{C1} + I_{B1})s^2 + (B_{B1} + B_{C1})s - G_B} \quad (3.65)$$

Now considering the equations from gearbox 2, equation (3.54) can be written as

$$\theta_1(I_1s^2 + B_1s - G_1) - \theta_2(I_2s^2 + B_2s) + T_2 = 0 \quad (3.66)$$

Substituting $T_1 = G_1\theta_1$ and $\theta_1 = 2\theta_A - \theta_2$, equation (3.66) becomes

$$2\theta_A(I_1s^2 + B_1s - G_1) - \theta_2[(I_1 + I_2)s^2 + (B_1 + B_2)s - G_1] + T_2 = 0 \quad (3.67)$$

$$\Rightarrow \theta_2 = \frac{2\theta_A(I_1s^2 + B_1s - G_1) + T_2}{(I_1 + I_2)s^2 + (B_1 + B_2)s - G_1} \quad (3.68)$$

From equation (3.53)

$$\theta_A(I_{A1}s^2 + B_{A1}s) + T_A + 2\theta_2(I_2s^2 + B_2s) - 2T_2 = 0 \quad (3.69)$$

Substitute equation (3.68) into equation (3.69)

$$\theta_A\left(I_{A1}s^2 + B_{A1}s + \frac{4(I_1s^2 + B_1s - G_1)(I_2s^2 + B_2s)}{(I_1 + I_2)s^2 + (B_1 + B_2)s - G_1}\right) = -T_A + 2T_2\left(1 - \frac{I_2s^2 + B_2s}{(I_1 + I_2)s^2 + (B_1 + B_2)s - G_1}\right) \quad (3.70)$$

From this define:

$$G_{A2} := I_{A1}s^2 + B_{A1}s + \frac{4(I_1s^2 + B_1s - G_1)(I_2s^2 + B_2s)}{(I_1 + I_2)s^2 + (B_1 + B_2)s - G_1} \tag{3.71}$$

$$G_2 := 1 - \frac{I_2s^2 + B_2s}{(I_1 + I_2)s^2 + (B_1 + B_2)s - G_1} \tag{3.72}$$

Now substitute $\theta_2 = 2\theta_A + \theta_1$ into equation (3.66), giving

$$\theta_1[(I_1 + I_2)s^2 + (B_1 + B_2)s - G_1] - 2\theta_A(I_2s^2 + B_2s) + T_2 = 0 \tag{3.73}$$

$$\text{To give:} \qquad \theta_A = \frac{\theta_1[(I_1 + I_2)s^2 + (B_1 + B_2)s - G_1] + T_2}{2(I_2s^2 + B_2s)} \tag{3.74}$$

Since the spring-mass-damper system has the same behavior, as previously the derivations in Section 3.4 are reused to give

$$\phi_B = \frac{K_B}{I_{B2}s^2 + B_{B2}s + K_B}\theta_B \tag{3.75}$$

$$\phi_1 = \frac{K_C}{I_{C2}s^2 + B_{C2}s + K_C}\theta_1 \tag{3.76}$$

From this, define the transfer-functions

$$G_{BB} := \frac{K_B}{I_{B2}s^2 + B_{B2}s + K_B} \tag{3.77}$$

$$G_{AA} := \frac{K_C}{I_{C2}s^2 + B_{C2}s + K_C} \tag{3.78}$$

together with

$$G_B := \frac{K_B^2}{I_{B2}s^2 + B_{B2}s + K_B} - (I_Bs^2 + B_Bs + K_B) \tag{3.79}$$

$$G_1 := \frac{K_C^2}{I_{C2}s^2 + B_{C2}s + K_C} - (I_Cs^2 + B_Cs + K_C) \tag{3.80}$$

Here G_B, G_1 are the transfer-functions relating the torque applied at the input of the spring-mass-damper components and the angular position of the input. Adding equation (3.63) and equation (3.70) gives

$$\theta_A G_{A1} + \theta_A G_{A2} = T_A + 2T_C G_C - T_A + 2T_2 G_2 \tag{3.81}$$

which results in

$$\theta_A(G_{A1} + G_{A2}) = 2T_C G_C + 2T_2 G_2 \tag{3.82}$$

Substitute equation (3.58) into equation (3.82), to give

$$\frac{\theta_B[(I_{B1}+I_{C1})s^2+(B_{B1}+B_{C1})s-G_B](G_{A1}+G_{A2})}{2(I_{C1}s^2+B_{C1}s)} = 2T_2G_2+T_C\left(2G_C-\frac{G_{A1}+G_{A2}}{2(I_{C1}s^2+B_{C1}s)}\right) \tag{3.83}$$

Now define $G_{BA} := \frac{[(I_{B1}+I_{C1})s^2+(B_{B1}+B_{C1})s-G_B](G_{A1}+G_{A2})}{2(I_{C1}s^2+B_{C1}s)}$. Similarly substitute equation (3.74) into equation (3.82).

$$\frac{\theta_1[(I_1+I_2)s^2+(B_1+B_2)s-G_1](G_{A1}+G_{A2})}{2(I_2s^2+B_2s)} = 2T_CG_C+T_2\left(2G_2-\frac{G_{A1}+G_{A2}}{2(I_2s^2+B_2s)}\right) \tag{3.84}$$

Also define: $G_{1A} := \frac{[(I_1+I_2)s^2+(B_1+B_2)s-G_1](G_{A1}+G_{A2})}{2(I_2s^2+B_2s)}$. Considering $\phi_1(s)=\mathbf{y}_1(s)$, $\phi_B(s)=\mathbf{y}_2(s)$ and $T_2(s)=\mathbf{u}_1(s)$, $T_C(s)=\mathbf{u}_2(s)$, the transfer-function for this system can be written as

$$\begin{bmatrix}\mathbf{y}_1(s)\\ \mathbf{y}_2(s)\end{bmatrix} = \begin{bmatrix}G_{11}(s) & G_{12}(s)\\ G_{21}(s) & G_{22}(s)\end{bmatrix}\begin{bmatrix}\mathbf{u}_1(s)\\ \mathbf{u}_2(s)\end{bmatrix} \tag{3.85}$$

in which values of $G_{11}(s), G_{12}(s), G_{21}(s), G_{22}(s)$ are defined as

$$G_{11}(s) = \frac{\left(2G_2(s)-\frac{G_{A1}(s)+G_{A2}(s)}{2(I_2s^2+B_2s)}\right)}{G_{1A}(s)}G_{AA}(s) \tag{3.86}$$

$$G_{12}(s) = \frac{2G_C(s)}{G_{1A(s)}}G_{AA}(s) \tag{3.87}$$

$$G_{21}(s) = \frac{2G_2(s)}{G_{BA}(s)}G_{BB}(s) \tag{3.88}$$

$$G_{22}(s) = \frac{2G_C(s)-\frac{G_{A1}(s)+G_{A2}(s)}{2(I_{C1}s^2+B_{C1}s)}}{G_{BA}(s)}G_{BB}(s) \tag{3.89}$$

3.6 Case 3 arrangement and transfer-function

In this system the mass-spring-damper component is placed inbetween the 2 differential gearboxes. This component provides additional parametric freedom to control the interaction. This arrangement is given in Figure 3.8. From case 2, equation (3.63) and equation (3.70) can again be used. The only difference is that the values of θ_A and T_A for each gearbox are not the same. The value of θ_A splits into θ_{A_1} and θ_{A_2}. Likewise T_A splits into T_{A1} and T_{A2}.

$$\theta_{A_2}G_{A1} = T_{A2}+2T_CG_C \tag{3.90}$$

$$\theta_{A_1}G_{A2} = -T_{A1}+2T_2G_2 \tag{3.91}$$

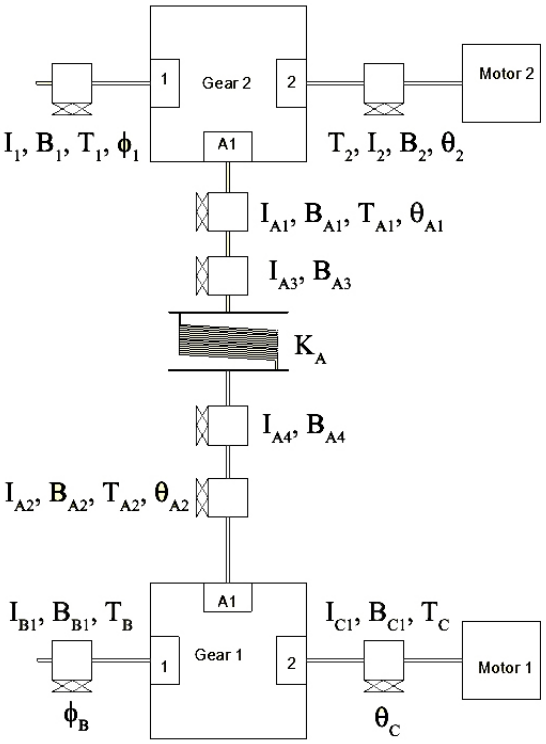


Figure 3.8: System arrangement 3.

Here T_{A1} and T_{A2} are the torques applied on port A_1 of gearbox 2 and port A_2 of gearbox 1 respectively. The location of each torque is shown in Figure 3.8. Using free-body diagrams, the relationship between components can be calculated as

$$I_{A3}\ddot{\theta}_{A1} + B_{A3}\dot{\theta}_{A1} + K_A(\theta_{A1} - \theta_{A2}) = T_{A1} \tag{3.92}$$

$$I_{A4}\ddot{\theta}_{A2} + B_{A4}\dot{\theta}_{A2} - K_A(\theta_{A1} - \theta_{A2}) = T_{A2} \tag{3.93}$$

where K_A is the spring constant. Using equation (3.92) and equation (3.93) substitute into equation (3.91) and equation (3.90) respectively, to give

$$\theta_{A2}G_{A1} = \ddot{\theta}_{A2}I_{A4} + B_{A4}\dot{\theta}_{A2} - K_A(\theta_{A1} - \theta_{A2}) + 2T_CG_C \tag{3.94}$$

$$\theta_{A1}G_{A2} = -\big(\ddot{\theta}_{A1}I_{A3} + B_{A3}\dot{\theta}_{A1} + K_A(\theta_{A1} - \theta_{A2})\big) + 2T_2G_2 \tag{3.95}$$

From equation (3.95)

$$\theta_{A1} = \frac{\theta_{A2}K_A + 2T_2G_2}{I_{A3}s^2 + B_{A3}s + K_A + G_{A2}} \tag{3.96}$$

Substitute into equation (3.94)

$$\theta_{A_2}(G_{A1} - I_{A4}s^2 - B_{A4}s - K_A + \frac{K_A^2}{I_{A3}s^2 + B_{A3}s + K_A - G_{A2}}) = \frac{-2T_2G_2K_A}{I_{A3}s^2 + B_{A3}s + K_A - G_{A2}} + 2T_CG_C \tag{3.97}$$

Similarly from equation (3.95) factorise θ_{A_2}

$$-K_A\theta_{A_2} = -\theta_{A1}(I_{A3}s^2 + B_{A3}s + K_A + G_{A2}) + 2T_2G_2 \tag{3.98}$$

From equation (3.94):

$$K_A\theta_{A1} = \theta_{A2}(I_{A4}s^2 + B_{A4}s + K_A - G_{A1}) + 2T_CG_C \tag{3.99}$$

Substitute equation (3.98) into equation (3.99) to give

$$\theta_{A1}[K_A - \frac{(I_{A3}s^2 + B_{A3}s + K_A + G_{A2})}{K_A}(I_{A4}s^2 + B_{A4}s + K_A - G_{A1})] = \frac{-2T_2G_2}{K_A}(I_{A4}s^2 + B_{A4}s + K_A - G_{A1}) + 2T_CG_C \tag{3.100}$$

The gearbox equation (3.58) and equation (3.74), using the current notation, are given by

$$\theta_{A1} = \frac{\theta_1[(I_1 + I_2)s^2 + (B_1 + B_2)s - G_1] + T_2}{2(I_2s^2 + B_2s)} \tag{3.101}$$

$$\theta_{A_2} = \frac{\theta_B[(I_{B1} + I_{C1})s^2 + (B_{B1} + B_{C1})s - G_B] + T_C}{2(I_{C1}s^2 + B_{C1}s)} \tag{3.102}$$

Substitute equation (3.101) into equation (3.100):

$$\theta_1\frac{[(I_1 + I_2)s^2 + (B_1 + B_2)s - G_1]}{2(I_2s^2 + B_2s)}G_{1A} = -T_2\left[2\frac{G_2}{K_A}(I_{A4}s^2 + B_{A4}s + K_A - G_{A1}) - \frac{G_{1A}}{2(I_2s^2 + B_2s)}\right] + 2T_CG_C \tag{3.103}$$

Define

$$G_{1A}(s) := K_A - \frac{(I_{A3}s^2 + B_{A3}s + K_A + G_{A2})(I_{A4}s^2 + B_{A4}s + K_A - G_{A1})}{K_A} \tag{3.104}$$

Substitute equation (3.102) into equation (3.97) :

$$\theta_B\frac{[(I_{B1} + I_{C1})s^2 + (B_{B1} + B_{C1})s - G_B]}{2(I_{C1}s^2 + B_{C1}s)}G_{1B} = \frac{2T_2G_2K_A}{I_{A3}s^2 + B_{A3}s + K_A - G_{A2}} + T_C\left(2G_2 - \frac{G_{1B}}{2(I_{C1}s^2 + B_{C1}s)}\right) \tag{3.105}$$

where $G_{1B}(s)$ can be defined as

$$G_{1B}(s) := G_{A1} - I_{A4}s^2 - B_{A4}s - K_A + \frac{K_A^2}{I_{A3}s^2 + B_{A3}s + K_A - G_{A2}(s)} \tag{3.106}$$

The other equations are now summarised as

$$G_{AA}(s) := \frac{[(I_1+I_2)s^2+(B_1+B_2)s-G_1(s)]}{2(I_2s^2+B_2s)} \tag{3.107}$$

$$G_{T2}(s) := -\big[2\frac{G_2(s)}{K_A}(I_{A4}s^2+B_{A4}s+K_A-G_{A1}(s))-\frac{G_{1A}(s)}{2(I_2s^2+B_2s)}\big] \tag{3.108}$$

$$G_{BB}(s) := \frac{[(I_{B1}+I_{C1})s^2+(B_{B1}+B_{C1})s-G_B(s)]}{2(I_{C1}s^2+B_{C1}s)} \tag{3.109}$$

$$G_{1B}(s) := G_{A1}(s)-I_{A4}s^2-B_{A4}s-K_A+\frac{K_A^2}{I_{A3}s^2+B_{A3}s+K_A-G_{A2}(s)} \tag{3.110}$$

$$G_{TC}(s) := 2G_2(s)-\frac{G_{1B}}{2(I_{C1}s^2+B_{C1}s)} \tag{3.111}$$

$$G_{A1}(s) := (I_{A2}s^2+B_{A2}s)+\frac{4(I_{B1}s^2+B_{B1}s-G_B(s))(I_{C1}s^2+B_{C1}s)}{(I_{C1}+I_{B1})s^2+(B_{B1}+B_{C1})s-G_B} \tag{3.112}$$

$$G_{A2}(s) := I_{A1}s^2+B_{A1}s+\frac{4(I_1s^2+B_1s-G_1(s))(I_2s^2+B_2s)}{(I_1+I_2)s^2+(B_1+B_2)s-G_1(s)} \tag{3.113}$$

$$G_2(s) := 1-\frac{I_2s^2+B_2s}{(I_1+I_2)s^2+(B_1+B_2)s-G_1(s)} \tag{3.114}$$

$$G_C(s) := 1-\frac{I_{C1}s^2+B_{C1}s}{(I_{C1}+I_{B1})s^2+(B_{B1}+B_{C1})s-G_B(s)} \tag{3.115}$$

If there are no additional components placed at port 1 of gearbox 2 and port B1 of gearbox 1, the transfer-functions $G_1(s)=G_B(s)$ are zero in which case $\theta_1=\phi_1$; $\theta_B=\phi_B$. Finally, the complete transfer-function matrix for this MIMO system is given by

$$\begin{bmatrix}\mathbf{y}_1(s)\\ \mathbf{y}_2(s)\end{bmatrix} = \begin{bmatrix}G_{11}(s) & G_{12}(s)\\ G_{21}(s) & G_{22}(s)\end{bmatrix}\begin{bmatrix}\mathbf{u}_1(s)\\ \mathbf{u}_2(s)\end{bmatrix} \tag{3.116}$$

where $\mathbf{y}_1=\phi_1(s)$, $\mathbf{y}_2=\phi_2(s)$ and $\mathbf{u}_1(s)=T_2(s)$, $\mathbf{u}_2(s)=T_C(s)$ in which each transfer-function is given as

$$G_{11}(s) = \frac{G_{T2}(s)}{G_{AA}(s)G_{1A}(s)} \tag{3.117}$$

$$G_{12}(s) = \frac{2G_C(s)}{G_{AA}(s)G_{1A}(s)} \tag{3.118}$$

$$G_{21}(s) = \frac{2G_2(s)K_A}{(I_{A3}s^2+B_{A3}s+K_A-G_{A2}(s))(G_{BB}(s)G_{1B}(s))} \tag{3.119}$$

$$G_{22}(s) = \frac{G_{TC}(s)}{G_{BB}(s)G_{1B}(s)} \tag{3.120}$$

3.7 Case 4 arrangement and transfer-function

This new arrangement is quite similar to the arrangement of case 2 but the spring-mass-damper components are between each motor and its associated gearbox. This new arrangement provides a different system behavior and therefore should be investigated

carefully in order to enable the most suitable choice of arrangement for future experimentation. The whole system is shown in Figure 3.9. There are no components connected to

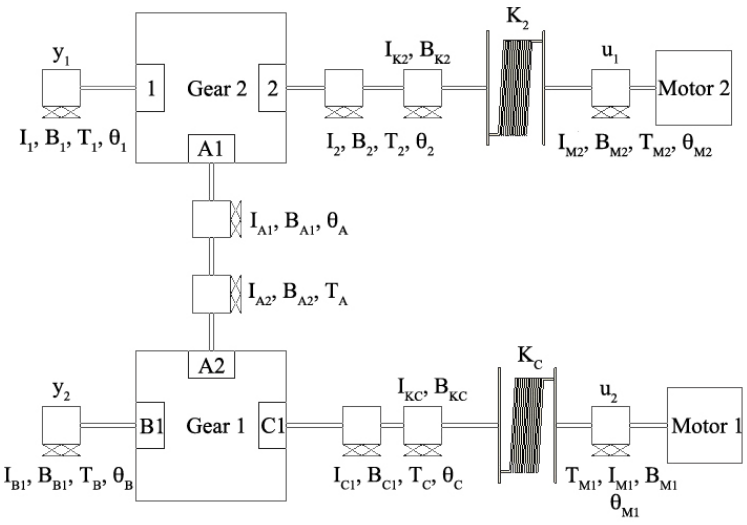


Figure 3.9: The arrangement 4

port 1 and B1 of the gearboxes leading to the transfer-function $G_B = G_1 = 0$ and externally applied torques $T_B = T_1 = 0$. Recall the earlier case of two differential gearboxes connected back-to-back (case 2). Here the set of equations for gearbox 1 is

$$I_{A2}\ddot{\theta}_A + B_{A2}\dot{\theta_A} - T_A + 2I_{C1}\ddot{\theta_C} + 2B_{C1}\dot{\theta_C} - 2T_C = 0 \tag{3.121}$$

$$I_{B1}\ddot{\theta}_B + B_{B1}\dot{\theta_B} - I_{C1}\ddot{\theta_C} - B_{C1}\dot{\theta_C} + T_C = 0 \tag{3.122}$$

$$2\theta_A = \theta_B + \theta_C \quad \text{(the constraint equation)} \tag{3.123}$$

and the set of equations for gearbox 2 is

$$I_{A1}\ddot{\theta}_A + B_{A1}\dot{\theta_A} + T_A + 2I_2\ddot{\theta_2} + 2B_2\dot{\theta_2} - 2T_2 = 0 \tag{3.124}$$

$$I_1\ddot{\theta}_1 + B_1\dot{\theta_1} - I_2\ddot{\theta_2} - B_2\dot{\theta_2} + T_2 = 0 \tag{3.125}$$

$$2\theta_A = \theta_1 + \theta_2 \quad \text{(the constraint equation)} \tag{3.126}$$

The spring-mass-damper system which is connected to gearbox 1 and 2 will be derived to obtain the transfer-functions and then these transfer-functions will be substituted into the sets of differential gearbox equations. Firstly, consider the spring-mass-damper between motor 2 and port 2 of gearbox 2. This system is shown in Figure 3.10. Here T_{M2} and T_2 are the torque of motor 2 and the applied torque on the gearbox respectively. Using the free-body diagram method, the set of equations that describe this system are

$$I_{M2}\ddot{\theta}_{M2} + B_{M2}\dot{\theta}_{M2} + K_2(\theta_{M2} - \theta_2) = T_{M2} \tag{3.127}$$

$$K_2(\theta_{M2} - \theta_2) = I_{K2}\ddot{\theta_2} + B_{K2}\dot{\theta_2} + T_2 \tag{3.128}$$

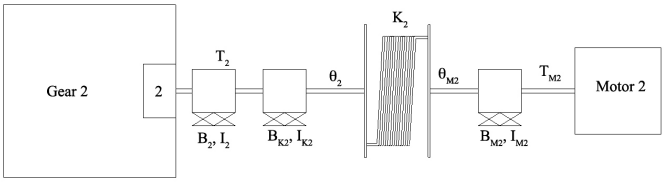


Figure 3.10: spring-mass-damper components

From equation (3.128)

$$T_2 = \theta_{M2}K_2 - \theta_2(K_2 + I_{K2}s^2 + B_{K2}s) \tag{3.129}$$

From equation (3.127)

$$\theta_{M2} = \frac{T_{M2} + K_2\theta_2}{I_{M2}s^2 + B_{M2}s + K_2} \tag{3.130}$$

Substitute into equation (3.129) to give the torque T_2 applied to gearbox 2, port 2, in terms of θ_2 and the torque applied by motor 2, T_{M2}

$$T_2 = \frac{K_2T_{M2}}{I_{M2}s^2 + B_{M2}s + K_2} + \theta_2\left[\frac{K_2^2}{I_{M2}s^2 + B_{M2}s + K_2} - (I_{K2}s^2 + B_{K2}s + K_2)\right] \tag{3.131}$$

Substitute equation (3.131) into equation (3.124) and equation (3.125) to give

$$\theta_A(I_{A1}s^2 + B_{A1}s) + T_A + 2\theta_2\left[I_2s^2 + B_2s - \left(\frac{K_2^2}{I_{M2}s^2 + B_{M2}s + K_2} - (I_{K2}s^2 + B_{K2}s + K_2)\right) - \frac{2T_{M2}K_2}{I_{M2}s^2 + B_{M2}s + K_2}\right] = 0 \tag{3.132}$$

$$\theta_1(I_1s^2 + B_1s) - \theta_2\left[I_2s^2 + B_2s - \left(\frac{K_2^2}{I_{M2}s^2 + B_{M2}s + K_2} - (I_{K2}s^2 + B_{K2}s + K_2)\right) - \frac{T_{M2}K_2}{I_{M2}s^2 + B_{M2}s + K_2}\right] = 0 \tag{3.133}$$

Now define

$$G_2 := I_2s^2 + B_2s - \left(\frac{K_2^2}{I_{M2}s^2 + B_{M2}s + K_2} - (I_{K2}s^2 + B_{K2}s + K_2)\right) \tag{3.134}$$

$$G_{M2} := \frac{K_2}{I_{M2}s^2 + B_{M2}s + K_2} \tag{3.135}$$

therefore

$$\theta_A(I_{A1}s^2 + B_{A1}s) + T_A + 2\theta_2G_2 - 2T_{M2}G_{M2} = 0 \tag{3.136}$$

$$\theta_1(I_1s^2 + B_1s) - \theta_2G_2 + G_{M2}T_{M2} = 0 \tag{3.137}$$

Substitute $\theta_1 = 2\theta_A - \theta_2$ into equation (3.137) to give

$$\theta_2 = \frac{2\theta_A(I_1s^2 + B_1s) + G_{M2}T_{M2}}{I_1s^2 + B_1s + G_2} \tag{3.138}$$

Substitute into equation (3.136)

$$\theta_A\left[I_{A1}s^2+B_{A1}s+\frac{4G_2(I_1s^2+B_1s)}{I_1s^2+B_1s+G_2}\right]+T_A-T_{M2}\left[2G_{M2}-\frac{2G_2G_{M2}}{I_1s^2+B_1s+G_2}\right]=0 \quad (3.139)$$

Define

$$G_{A2}(s) := I_{A1}s^2+B_{A1}s+\frac{4G_2(I_1s^2+B_1s)}{I_1s^2+B_1s+G_2} \quad (3.140)$$

Therefore equation (3.139) is equivalent to

$$G_{A2}(s)\theta_A=-T_A+T_{M2}\left[2G_{M2}(s)-\frac{2G_2G_{M2}(s)}{I_1s^2+B_1s+G_2(s)}\right] \quad (3.141)$$

Substitute equation $\theta_2=2\theta_A-\theta_1$ into equation (3.137) to yield

$$\theta_A=\frac{\theta_1(I_1s^2+B_1s+G_2(s))+G_{M2}(s)T_{M2}}{2G_2(s)} \quad (3.142)$$

Secondly, the lower half of the system will be investigated. Figure 3.11 shows all the parameters and the components of the spring-mass-damper system and how it is connected to the system. As with the first half, the transfer-function for this system can be derived

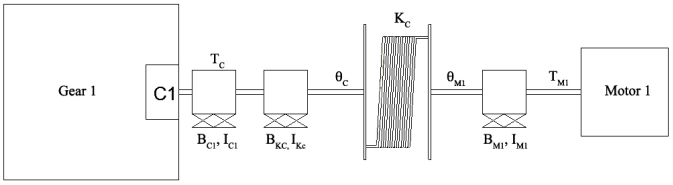


Figure 3.11: Lower half spring-mass-damper components

using free-body diagrams as

$$I_{M1}\ddot{\theta_{M1}}+B_{M1}\dot{\theta_{M1}}+K_C(\theta_{M1}-\theta_C)=T_{M1} \quad (3.143)$$

$$\ddot{\theta_C}I_{KC}+\dot{\theta_C}B_{KC}=K_C(\theta_{M1}-\theta_C)-T_C \quad (3.144)$$

From equation (3.143)

$$\theta_{M1}=\frac{T_{M1}+K_C\theta_C}{I_{M1}s^2+B_{M1}s+K_C} \quad (3.145)$$

Substitute into equation (3.144) to produce

$$T_C=\theta_C\left[\frac{K_C^2}{I_{M1}s^2+B_{M1}s+K_C}-(I_{KC}s^2+B_{KC}s+K_C)\right]+\frac{K_CT_{M1}}{I_{M1}s^2+B_{M1}s+K_C} \quad (3.146)$$

Substitute this equation into equation (3.121) and equation (3.122)

$$\theta_A(I_{A2}s^2 + B_{A2}s) - T_A + 2\theta_C\left[I_{C1}s^2 + B_{C1}s - \left(\frac{K_C^2}{I_{M1}s^2 + B_{M1}s + K_C} - (I_{KC}s^2 + B_{KC}s + K_C)\right)\right] - \frac{2K_C T_{M1}}{I_{M1}s^2 + B_{M1}s + K_C} = 0 \quad (3.147)$$

$$\theta_B(I_{B1}s^2 + B_{B1}s) - \theta_C\left[I_{C1}s^2 + B_{C1}s - \left(\frac{K_C^2}{I_{M1}s^2 + B_{M1}s + K_C} - (I_{KC}s^2 + B_{KC}s + K_C)\right)\right] + \frac{K_C T_{M1}}{I_{M1}s^2 + B_{M1}s + K_C} = 0 \quad (3.148)$$

Define

$$G_C(s) := I_{C1}s^2 + B_{Cs1}s - \left(\frac{K_C^2}{I_{M1}s^2 + B_{M1}s + K_C} - (I_{KC}s^2 + B_{KC}s + K_C)\right) \quad (3.149)$$

$$G_{M1}(s) := \frac{K_C}{I_{M1}s^2 + B_{M1}s + K_C} \quad (3.150)$$

So that equation (3.147) and equation (3.148) can be written as

$$\theta_A(I_{A2}s^2 + B_{A2}s) + 2\theta_C G_C(s) - 2T_{M1}G_{M1}(s) = T_A \quad (3.151)$$

$$\theta_B(I_{B1}s^2 + B_{B1}s) - \theta_C G_C(s) + T_{M1}G_{M1}(s) = 0 \quad (3.152)$$

Substitute $2\theta_B = 2\theta_A - \theta_C$ into equation (3.152)

$$\theta_C = \frac{2\theta_A(I_{B1}s^2 + B_{B1}s) + T_{M1}G_{M1}(s)}{I_{B1}s^2 + B_{B1}s + G_C(s)} \quad (3.153)$$

Substitute into equation (3.151)

$$\theta_A\left[I_{A2}s^2 + B_{A2}s + \frac{4G_C(s)(I_{B1}s^2 + B_{B1}s)}{I_{B1}s^2 + B_{B1}s + G_C(s)}\right] + T_{M1}\left(\frac{2G_{M1}G_C(s)}{I_{B1}s^2 + B_{B1}s + G_C(s)} - 2G_{M1}(s)\right) = T_A \quad (3.154)$$

Define

$$G_{A1}(s) := I_{A2}s^2 + B_{A2}s + \frac{4G_C(s)(I_{B1}s^2 + B_{B1}s)}{I_{B1}s^2 + B_{B1}s + G_C(s)} \quad (3.155)$$

and substitute into equation (3.154) to give

$$\theta_A G_{A1} = T_A + T_{M1}\left(2G_{M1} - \frac{2G_{M1}G_C}{I_{B1}s^2 + B_{B1}s + G_C(s)}\right) \quad (3.156)$$

From equation (3.152), substitute $\theta_C = 2\theta_A - \theta_B$ to give

$$\theta_A = \frac{T_{M1}G_{M1} + \theta_B(I_{B1}s^2 + B_{B1}s + G_C(s))}{2G_C(s)} \quad (3.157)$$

Now add equation (3.156) and equation (3.141) to give

$$\theta_A(G_{A1}(s)+G_{A2}(s)) = T_{M1}\left(2G_{M_1}(s) - \frac{2G_{M1}(s)G_C(s)}{I_{B1}s^2+B_{B1}s+G_C(s)}\right)+ T_{M2}\left(2G_{M2}(s) - \frac{2G_2(s)G_{M2}(s)}{I_1s^2+B_1s+G_2(s)}\right) \quad (3.158)$$

Since the value of torque T_A in gear 1 is opposite to the value of torque in gear 2 therefore they cancel out. Substitute equation (3.157) and equation (3.142) into equation (3.158):

$$\theta_1\frac{(I_1s^2+B_1s+G_2(s))(G_{A1}(s)+G_{A2}(s))}{2G_2(s)} = T_{M1}\left[2G_{M1} - \frac{2G_{M1}(s)G_C(s)}{I_{B1}s^2+B_{B1}s+G_C(s)}\right] +T_{M2}[2G_{M2}(s) - \frac{2G_2G_{M2}(s)}{I_1s^2+B_1s+G_2(s)} - \frac{G_{M2}(s)(G_{A1}(s)+G_{A2}(s))}{2G_2(s)} \quad (3.159)$$

$$\theta_B\frac{(I_{B1}s^2+B_{B1}s+G_C(s))(G_{A1}(s)+G_{A2}(s))}{2G_C(s)} = T_{M1}\left[2G_{M1}(s) - \frac{2G_{M1}(s)G_C(s)}{I_{B1}s^2+B_{B1}s+G_C(s)} - \frac{G_{M1}(s)(G_{A1}(s)+G_{A2}(s))}{2G_C(s)}\right] + T_{M2}\left[2G_{M2}(s) - \frac{2G_2(s)G_{M2}(s)}{I_1s^2+B_1s+G_2(s)}\right] \quad (3.160)$$

Finally, the transfer-functions for this arrangement can be written as

$$\begin{bmatrix}\mathbf{u}_1(s)\\ \mathbf{u}_2(s)\end{bmatrix}\begin{bmatrix}G_{11}(s) & G_{12}(s)\\ G_{21}(s) & G_{22}(s)\end{bmatrix}\begin{bmatrix}\mathbf{y}_1(s)\\ \mathbf{y}_2(s)\end{bmatrix} \quad (3.161)$$

where $T_{M2}(s) = \mathbf{y}_1(s)$, $T_{M1}(s) = \mathbf{y}_2(s)$ and $\mathbf{u}_1(s) = \theta_1(s) = \phi_1(s)$, $\mathbf{u}_2(s) = \theta_2(s) = \phi_2(s)$. The other transfer-functions are defined as

$$G_1(s) := \frac{(I_1s^2+B_1s+G_2(s))(G_{A1}(s)+G_{A2}(s))}{2G_2(s)} \quad (3.162)$$

$$G_B(s) := \frac{(I_{B1}s^2+B_{B1}s+G_C(s))(G_{A1}(s)+G_{A2}(s))}{2G_C(s)} \quad (3.163)$$

to give the transfer-functions components

$$G_{11}(s) := \frac{2G_{M2}(s) - \frac{2G_2(s)G_{M2}(s)}{I_1s^2+B_1s+G_2(s)} - \frac{G_{M2}(s)(G_{A1}(s)+G_{A2}(s))}{2G_2(s)}}{G_1(s)} \quad (3.164)$$

$$G_{12}(s) := \frac{2G_{M1}(s) - \frac{2G_{M1}(s)G_C(s)}{I_{B1}s^2+B_{B1}s+G_C(s)}}{G_1(s)} \quad (3.165)$$

$$G_{21}(s) := \frac{2G_{M2}(s) - \frac{2G_2(s)G_{M2}(s)}{I_1s^2+B_1s+G_2(s)}}{G_B(s)} \quad (3.166)$$

$$G_{22}(s) := \frac{2G_{M1}(s) - \frac{2G_{M1}(s)G_C(s)}{I_{B1}s^2+B_{B1}s+G_C(s)} - \frac{G_{M1}(s)(G_{A1}(s)+G_{A2}(s))}{2G_C(s)}}{G_B(s)} \quad (3.167)$$

Having established dynamic representations for a variety of system configurations, the next chapter will examine what parameter values are feasible in order to result in desirable dynamic characteristics for the final system.

3.8 Summary

This chapter investigated a variety of different 2×2 MIMO structure incorporating differential gearboxes and spring-mass-damper sections. Each combination was analysed and a matrix linking inputs and outputs was derived. In order to further examine each layout, simulation studies will be reported out in the next chapter, together with measures of interaction. A huge variety of feasible system parameters (damping, inertia and stiffness) will be analysed to arrive at an optimum selection. Addition of a disturbance/noise injection device will also be discussed in term of both location and characteristic dynamic.

Chapter 4

Parameter selection and system simulation

As described in the previous chapter, the transfer-functions of the dual differential gearboxes connected with spring-mass-damper sections in various arrangements have been successfully derived. In this chapter, these transfer-functions will be implemented in Matlab and simulated with realistic parameter values and hence the behavior of each arrangement can be analysed in detail. The end result will be the selection of a final design for manufacture. The design should provide a stable or marginally stable relationship between inputs and outputs. The interaction should be able to be modified through variation of a small number of parameters. Finally, the system needs to be controllable and observable to increase the range of controllers which may be applied.

4.1 Simulation of case 1

From the previous chapter, the transfer-functions for Case 1 in Section 3.4 have been derived and are given by (3.45) to (3.49). Many simulations has been carried out using Matlab with different parameter values. In general, the transfer-function has 1 pole at the origin and is therefore marginally stable. Figure 3.4 shows the system layout and associated parameters, which has been chosen with physically realisable values. Pole-zero cancellation has been carried out prior to calculation of the impulse response. As described in (Williamson, 1999), the impulse response values provide a measure of the input-output magnification of a system. Consider a digital signal $\mathbf{y}$ which is the solution of the causal n^{th} order linear time invariant difference equation

$$\mathbf{y}(k+n) = \sum_{p=0}^{n-1} a_p \mathbf{y}(k+p) + \sum_{p=0}^{n} b_p \mathbf{u}(k+p); \quad k \geq k_0 - n \tag{4.1}$$

where a_p, b_p for all integers p are real constants with $a_0 \neq 0$, k_0 is a fixed integer and k is a variable integer. The signal $\mathbf{y}$ is uniquely determined by the initial conditions $\mathbf{y}(k_0 - n), \mathbf{y}(k_0 - n + 1), ..., \mathbf{y}(k_0 - 1)$ and the input signal $\mathbf{u}$. Therefore the complete response $\mathbf{y}$ for $k \geq 0$ can be expressed in the form

$$\mathbf{y}(k) = \mathbf{y}_i(k) + \sum_{m=0}^{k} h(k-m)\mathbf{u}(m) \tag{4.2}$$

Here $\mathbf{y}_i$ is the zero input response (initial condition) and h is the unit impulse response. Consequently, an upper bound on the infinity norm of the complete response $\mathbf{y}$ of the linear time invariant difference equation (4.1) can be expressed in terms of norms on the unit impulse response h and the input signal u. This is given by

$$\|\mathbf{y}\|_\infty \leq \|\mathbf{y}_i\|_\infty + L \tag{4.3}$$

where $\mathbf{y}_i$ is the zero input response and L is defined as $L = \|h\|_1\|\mathbf{u}\|_\infty$. The impulse responses shown in the simulations which follow therefore provide information concerning the extent of interaction of the system. In all sections in this chapter, inertial, damping and stiffness-values K have units kgm^2, Nsm^{-2} and Nm^{-1} respectively. Simulation of

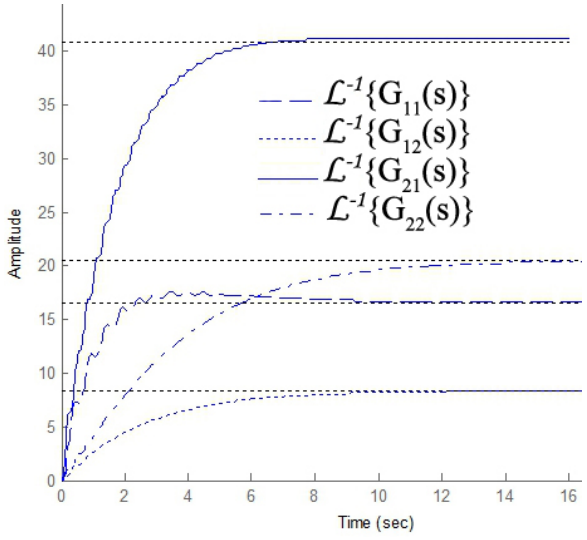


Figure 4.1: The impulse response of case 1 arrangement.

case one has been conducted using the parameters shown in Table 4.1.

Within this set of parameters I_{B1}, B_{B1}, I_{C1}, B_{C1}, I_{A2}, B_{A2}, I_{A1}, B_{A1}, I_1, B_1, I_2 and B_2 are the inertial and damper constants of the differential gearbox. These values are very small compared to the other parameters. In reality having selected a gearbox, these values cannot be changed. The impulse response of the system is shown in Figure 4.1. The steady-state outputs of the system can either assume a negative or a positive value depending on the defined positive direction between output angle displacement and input torque. The settling time is quite slow but it can be controlled by adjusting the damper

constants B_B and B_C. The spring constants K_B and K_C produce more fluctuation in the response of the system. From Figure 4.1 a significant interaction is present. No matter what parameter values are chosen, the degree of interaction cannot be fully controlled (as specified by the magnitude of G_{11} and G_{22} with respect to G_{12} and G_{21}).

The controllability and the observability of the transfer-functions can be established by application of theorems in (Dorf and Bishop, 2005), (Srivastava et al., 2009). For a system described by the state-space model

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{4.4}$$

$$y(t) = Cx(t) \tag{4.5}$$

controllability is determined by examining the algebraic condition for the controllability matrix P_C

$$P_C = [B \quad AB \quad A^2B \cdots A^{n-1}B] \tag{4.6}$$

$$\text{rank}(P_C) = n \tag{4.7}$$

The matrix A is an $n \times n$ matrix and B is an $n \times 1$ matrix. For multi-input systems, B can be $n \times m$ where m is the number of inputs. So if the determinant of matrix P_C is non-zero or the matrix P_C is full row rank, the system is controllable. In order to ascertain observability, the observability matrix P_O is constructed as

$$P_O^T = [C \quad CA \cdots CA^{n-1}] \tag{4.8}$$

$$\text{rank}(P_O^T) = n \tag{4.9}$$

where P_O^T is the transpose of the P_O matrix, C is an $m \times n$ matrix, and x is an $n \times 1$ column vector. The system is observable if the P_O^T matrix has full row rank. Applying equations (4.6) and (4.8) to the system transfer-function to investigate the controllability and observability of the system shows that the matrix P_C and P_O are each full row rank so this transfer-function with realistic parameters is controllable and observable. However through inherent coupling existing between the transfer-functions, there is also limited scope to independently specify any one transfer-function in the matrix. As a result, this configuration will not be considered for the test facility.

Table 4.1: Chosen parameter values

System parameter values							
I_1	0.0001 $(kg.m^2)$	I_2	0.0006 $(kg.m^2)$	I_{A1}	0.0009 $(kg.m^2)$	I_{A2}	0.0008 $(kg.m^2)$
I_{B1}	0.0004 $(kg.m^2)$	I_{C1}	0.0004 $(kg.m^2)$	B_{A1}	0.0001 $(\frac{N.m.s}{rad})$	B_{A2}	0.0009 $(\frac{N.m.s}{rad})$
B_1	0.0001 $(\frac{N.m.s}{rad})$	B_2	0.0002 $(\frac{N.m.s}{rad})$	B_{B1}	0.0002 $(\frac{N.m.s}{rad})$	B_{C1}	0.0002 $(\frac{N.m.s}{rad})$
I_B	0.04 $(kg.m^2)$	I_C	0.03 $(kg.m^2)$	I_{B2}	0.008 $(kg.m^2)$	I_{C2}	0.005 $(kg.m^2)$
B_{B2}	0.01 $(\frac{N.m.s}{rad})$	B_{C2}	0.006 $(\frac{N.m.s}{rad})$	B_B	0.03 $(\frac{N.m.s}{rad})$	B_C	0.01 $(\frac{N.m.s}{rad})$
K_B	1 $(\frac{N.m}{rad})$	K_C	2 $(\frac{N.m}{rad})$				

4.2 Simulation of case 2

Matlab has also been used to simulate the response of case 2 with the results shown in Figure 4.2, for a wide variety of parameter values. The transfer-functions are given by equations from (3.85) to (3.89). G_{11} and G_{22} are the diagonal terms of the transfer-

Table 4.2: Chosen parameter values

System parameter values							
I_1	10^{-4} $(kg.m^2)$	I_2	10^{-4} $(kg.m^2)$	I_{A1}	10^{-4} $(kg.m^2)$	I_{A2}	0 $(kg.m^2)$
I_{B1}	10^{-4} $(kg.m^2)$	I_{C1}	10^{-4} $(kg.m^2)$	B_{A1}	0 $(\frac{N.m.s}{rad})$	B_{A2}	0 $(\frac{N.m.s}{rad})$
B_1	0.0002 $(\frac{N.m.s}{rad})$	B_2	0.0002 $(\frac{N.m.s}{rad})$	B_{B1}	0.0002 $(\frac{N.m.s}{rad})$	B_{C1}	0.0002 $(\frac{N.m.s}{rad})$
I_B	0.01 $(kg.m^2)$	B_B	0.013 $(\frac{N.m.s}{rad})$	I_{C2}	0.011 $(kg.m^2)$	B_{C2}	0.014 $(\frac{N.m.s}{rad})$
I_C	0.03 $(kg.m^2)$	B_C	0.012 $(\frac{N.m.s}{rad})$	I_{B2}	0.02 $(kg.m^2)$	B_{B2}	0.008 $(\frac{N.m.s}{rad})$
K_B	1.5 $(\frac{N.m}{rad})$	K_C	1 $(\frac{N.m}{rad})$				

function matrix and G_{12} and G_{21} are off-diagonal entities which specify interaction. In this system the steady-state of each transfer-function can be changed easily by adjusting specific parameters. For example, the steady-state values of G_{11} and G_{22} can be changed by varying the values of B_{C2} and B_{B2}. The steady-state value of the coupling transfer-functions G_{12}, G_{21} can be changed by adjusting the value of B_{A1} and B_{A2}. The settling time can be changed by varying the damper coefficients B_B and B_1. In particular, this set-up is more convenient than the previous arrangement because the interaction can be canceled out completely if there is a very large value of damper B_{A1} or B_{A2} or both. The value of I_{A1} and I_{A2} corresponds to the inner inertia of the differential gearbox 1 and 2 but these values can be increased by connecting an external damper component which will augment the value of damping in the connection shaft. Figure 4.2 shows the system response with very little inertia in the coupling, therefore the interaction is very high. G_{11}, G_{22} are approximately equal to G_{12} and G_{21} respectively. In this

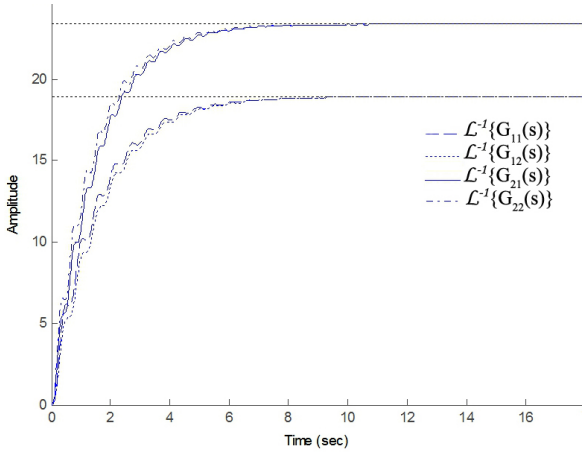


Figure 4.2: The impulse response of case 2 arrangement.

arrangement, the fluctuation of each transfer-function can be minimised by increasing

the spring constant. If the spring constants are very stiff (K tends to infinity) the output will not have any fluctuation at all but if the stiffness constant is small, there will exist significant fluctuation. Applying equations (4.6) and (4.8) to check the controllability and observability of the system using Matlab yields results which show that the system is controllable and observable since matrices P_O^T and P_C are full row rank. Therefore, this system is ideal for the intended purpose of controller validation and benchmarking since the response of this system can be fully adjusted.

4.3 Simulation of case 3

The arrangement corresponding to this case is shown in Figure 3.8. The transfer-functions for this system are given by equations (3.116) to (3.120). The system has only one spring-mass-damper section located in the coupling connection. These components will determine the interaction effect and the performance of the system. Using realistic parameters values, the impulse response is shown in Figure 4.3. The set of parameters used is given in Table 4.3.

Table 4.3: Chosen parameter values

System parameter values							
I_1	10^{-4} $(kg.m^2)$	B_1	10^{-4} $(\frac{N.m.s}{rad})$	I_2	10^{-4} $(kg.m^2)$	B_2	10^{-4} $(\frac{N.m.s}{rad})$
I_{A1}	10^{-4} $(kg.m^2)$	I_{A2}	0.0002 $(kg.m^2)$	B_{A1}	0.0002 $(\frac{N.m.s}{rad})$	B_{A2}	10^{-4} $(\frac{N.m.s}{rad})$
I_{B1}	10^{-4} $(kg.m^2)$	I_{C1}	10^{-4} $(kg.m^2)$	B_{B1}	0.0002 $(\frac{N.m.s}{rad})$	B_{C1}	10^{-4} $(\frac{N.m.s}{rad})$
I_{A3}	0.009 $(kg.m^2)$	B_{A3}	0.009 $(\frac{N.m.s}{rad})$	I_{A4}	0.013 $(kg.m^2)$	B_{A4}	0.003 $(\frac{N.m.s}{rad})$
I_B	0.03 $(kg.m^2)$	B_B	0.01 $(\frac{N.m.s}{rad})$	I_C	0.02 $(kg.m^2)$	B_C	0.008 $(\frac{N.m.s}{rad})$
K_A	1.5 $(\frac{N.m}{rad})$						

Similarly to the other 2 simulations, G_{11} and G_{22} are the 'direct' transfer-functions for the system and G_{12} and G_{21} determine the interaction. In this case, there is no damper or inertia at the output ports, therefore the settling time for each output is very long. However the I_B, B_B, I_C, B_C parameters can be used to change the settling time as well as the magnitude. These components are located at the output of each port. I_B, B_B is connected to I_{B1} and B_{B1}. I_C, B_C is connected to I_1 and B_1. With the spring-mass-damper situated between the gearboxes, these parameters can directly adjust the interaction of the system. If the dampers B_{A3} and B_{A4} are increased, the magnitude of interaction is smaller. If the inertial I_{A3} or I_{A4} parameters decrease, the response time for the system will be faster. The stiffness constant in this system can be used to adjust the fluctuation in the G_{11}, G_{12}, G_{21} and G_{22} responses. Therefore, this configuration needs the aid of extra output components in order to change the response of the system. Applying the controllability and observability tests given by (4.6) and (4.8) to this transfer-function, the values are all full row rank. it can thus be concluded

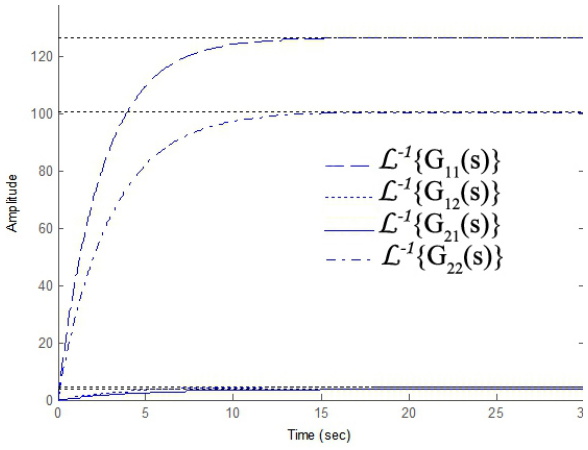


Figure 4.3: The impulse response of case 3 arrangement.

that this system is controllable and observable. As a result, this arrangement can be combined with other spring-mass-damper sections to make a more complicated system.

4.4 Simulation of case 4

The arrangement corresponding to case 4 can be seen in Figure 3.9. The chosen parameters of this system are given in Table 4.4. The derived transfer-functions are given in (3.161) to (3.167). The corresponding impulse responses for this system are shown in

Table 4.4: Chosen parameter values

System parameter values							
I_1	10^{-4} $(kg.m^2)$	B_1	10^{-4} $(\frac{N.m.s}{rad})$	I_{B1}	10^{-4} $(kg.m^2)$	B_{B1}	10^{-4} $(\frac{N.m.s}{rad})$
I_{A1}	10^{-4} $(kg.m^2)$	B_{A1}	10^{-4} $(\frac{N.m.s}{rad})$	I_{A2}	10^{-4} $(kg.m^2)$	B_{A2}	10^{-4} $(\frac{N.m.s}{rad})$
I_{C1}	10^{-4} $(kg.m^2)$	B_{C1}	10^{-4} $(\frac{N.m.s}{rad})$	I_{K2}	0.012 $(kg.m^2)$	B_{K2}	0.011 $(\frac{N.m.s}{rad})$
I_{M2}	0.02 $(kg.m^2)$	B_{M2}	0.01 $(\frac{N.m.s}{rad})$	I_2	10^{-4} $(kg.m^2)$	B_2	10^{-4} $(\frac{N.m.s}{rad})$
I_{KC}	0.001 $(kg.m^2)$	B_{KC}	0.011 $(\frac{N.m.s}{rad})$	I_{M1}	0.01 $(kg.m^2)$	B_{M1}	0.008 $(\frac{N.m.s}{rad})$
K_2	1 $(\frac{N.m}{rad})$	K_C	2 $(\frac{N.m}{rad})$				

Figure 4.4. As with other arrangements, the interaction can be controlled by increasing the value of damping in the connection between differential gearboxes, and the system will have less fluctuation if the spring constant is higher. The stabilised values of output terms G_{11} and G_{22} can be changed easily if there are external inertia and damper devices connected. The smaller the inertia, the faster the system response. As a result, the response of this system can be fully manipulated as required. Using equations (4.8) and (4.6) to obtain the controllability and observability matrices respectively, the results from Matlab confirm that these matrices are of full row rank, and therefore this system is both controllable and observable.

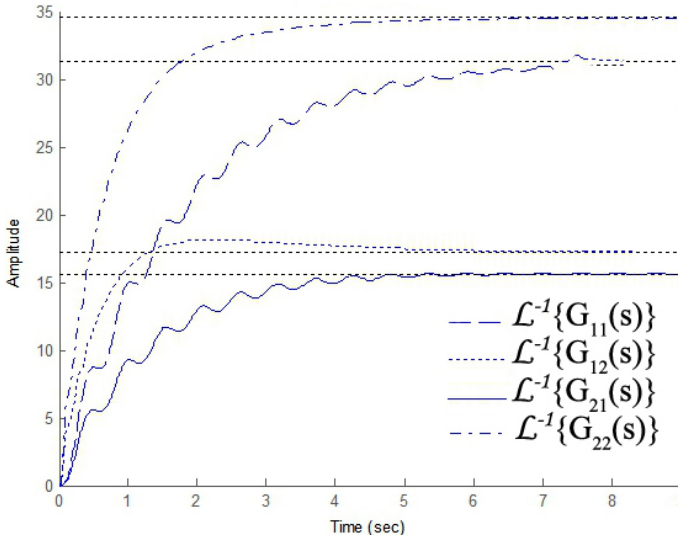


Figure 4.4: The impulse response of case 4 arrangement.

4.5 Choosing the optimum configuration for the MIMO facility

From the preceding four configurations and corresponding transfer-functions, the only arrangement of the MIMO system which cannot be used is arrangement 1 because the interaction factor cannot be canceled completely and the output of this system is very difficult to change since the interaction is substantial. Therefore there are three remaining configurations left. The remaining configurations will be analysed in order to choose the most suitable system to manufacture. Transfer-functions relating the inputs and outputs, taken as $\{\mathbf{u}_1, \mathbf{u}_2\}$ and $\{\mathbf{y}_1, \mathbf{y}_2\}$ respectively, have been calculated in each case, and are given later in this section. To estimate the magnitude of the interaction existing in the MIMO system, the relative gain array (RGA) Bristol (1966) is used and is written as

$$RGA \text{ number} = \|G(jw) \times (G^{-1}(jw))^T - I\|_{\text{sum}} \tag{4.10}$$

where $\times$ denotes element-by-element multiplication, G is the transfer-function matrix and w is the frequency ($rads^{-1}$). The sum norm is defined as $||A||_{\text{sum}} = \sum_{i,j} |a_{ij}|$. If the RGA number is close to zero, this indicates little interaction at that frequency, with values greater than one indicating substantial interaction and associated control difficulty.

For arrangement 3 in Figure 3.8, two differential gears are connected via a central spring-mass-damper. Varying the latter's mass, damping and spring constant has little effect on the outputs coupled via the gears but directly affects the interaction. The RGA number for this first arrangement with a selection of realisable parameter values is shown in Figure 4.5. For high interaction $B_{A3} = B_{A4} = 0.0001$, medium interaction $B_{A3} = B_{A4} = 0.0005$, low interaction $B_{A3} = B_{A4} = 0.002$. Based on the RGA number and numerous

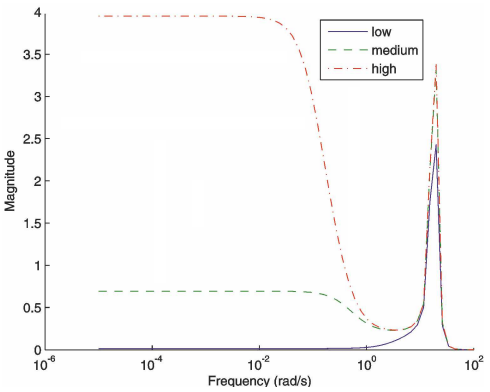


Figure 4.5: RGA number for arrangement 1.

simulations with a variety of parameter values, this arrangement is found to be unsuitable because significant interaction at 10 $rads^{-1}$ cannot be reduced as shown in the figure. Additionally it is not possible to influence the dynamics of the system through further variation of spring-mass-damper components as is desirable in practice. In arrangement 4, shown in Figure 3.9, the interaction can be stipulated by varying the damper and mass values located between the 2 differential gears. The dynamics of output 1 of gear 2 can be modified by varying $I_{M2}, B_{M1}, K_2, I_{K2}, B_{K2}$. Similarly, the dynamics of output B_1 of gear 1 can be modified by changing $I_{M1}, B_{M1}, K_C, I_{KC}, B_{KC}$, but this arrangement needs to have a very stiff spring in order to cope with the necessarily high torque demand from the induction motors which drive the inputs. For high interaction $B_{A1} = B_{A2} = 0.0001$, medium interaction $B_{A1} = B_{A2} = 0.0005$, low interaction $B_{A1} = B_{A2} = 0.002$. The RGA number is given in Figure 4.6 for typical parameter values and shows that

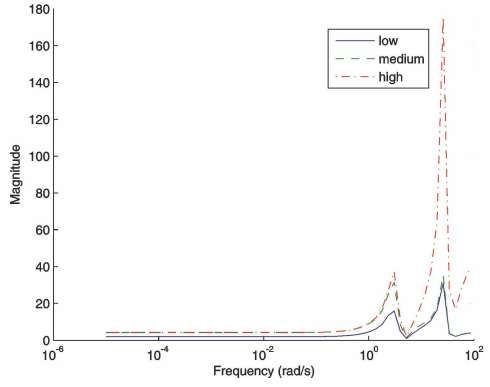


Figure 4.6: RGA number for arrangement 2.

the system has low interaction at low frequencies but at higher frequencies the system has much more interaction and will hence always present significant control difficulty. This has also been confirmed by numerous simulations using a wide range of parameter values. Therefore this arrangement is also deemed unsuitable. The final layout considered is arrangement 2 in Figure 3.7 which has a level of interaction which can be transparently manipulated using a small number of physical parameters. Additional variation in spring-mass-damper constants has a large effect on the outputs, and the RGA number can be

decreased within the operating frequency range for medium and low interaction cases, as shown in Figure 4.7. For high interaction $B_{A1} = B_{A2} = 0.0001$, medium interaction $B_{A1} = B_{A2} = 0.0005$, low interaction $B_{A1} = B_{A2} = 0.002$. Hence this arrangement is ideal for the MIMO test-bed facility.

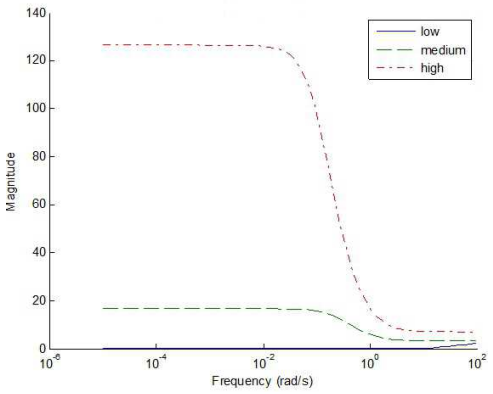


Figure 4.7: RGA number for arrangement 3.

4.6 PID tuning for the system in case 2

PID is one of the simplest methods to control a system. It is easy to implement and generally gives a reasonable performance, however it is rarely used for MIMO systems since the interaction can be a significant problem to deal with. In this case, the biggest log modulus tuning (BLT) method (Luyben, 1986) will be used to investigate how the interaction will cause difficulty in controlling the system when using an arrangement similar to Figure 3.7. The pairing is for $\mathbf{u}_1$ to $\mathbf{y}_1$ and $\mathbf{u}_2$ to $\mathbf{y}_2$ and all the parameters for high interaction are given in Table 4.2. The impulse response for this case is shown in Figure 4.2. The parameters chosen for low interaction are shown in Table 4.5. The only

Table 4.5: Low interaction table.

System parameter values							
I_1	10^{-4} $(kg.m^2)$	I_2	10^{-4} $(kg.m^2)$	I_{A1}	10^{-4} $(kg.m^2)$	I_{A2}	10^{-4} $(kg.m^2)$
I_{B1}	10^{-4} $(kg.m^2)$	I_{C1}	10^{-4} $(kg.m^2)$	B_{A1}	0.5 $(\frac{N.m.s}{rad})$	B_{A2}	0.5 $(\frac{N.m.s}{rad})$
B_1	0.0002 $(\frac{N.m.s}{rad})$	B_2	0.0002 $(\frac{N.m.s}{rad})$	B_{B1}	0.0002 $(\frac{N.m.s}{rad})$	B_{C1}	0.0002 $(\frac{N.m.s}{rad})$
I_B	0.01 $(kg.m^2)$	B_B	0.013 $(\frac{N.m.s}{rad})$	I_{C2}	0.011 $(kg.m^2)$	B_{C2}	0.04 $(\frac{N.m.s}{rad})$
I_C	0.03 $(kg.m^2)$	B_C	0.012 $(\frac{N.m.s}{rad})$	I_{B2}	0.02 $(kg.m^2)$	B_{B2}	0.05 $(\frac{N.m.s}{rad})$
K_B	1.5 $(\frac{N.m}{rad})$	K_C	1 $(\frac{N.m}{rad})$				

difference in this table compared with Table 4.2 is that the value of damper coefficients B_{A2}, B_{A1} is 0.5 instead of 0. These values enable full control over the interaction. Additionally I_{A2} is 0.0001, B_{B2} is 0.05 and B_{C2} is 0.04. The impulse response of this system is simulated in Matlab and the result is shown in Figure 4.8. Comparing results in Figure 4.2 and Figure 4.8, there is substantial difference in the interaction magnitude. The

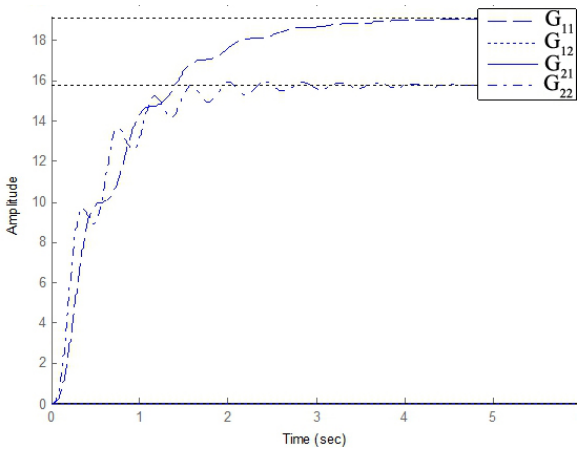


Figure 4.8: Low interaction impulse response

high interaction case has interaction functions close to the dominant transfer-functions whilst in the low interaction case they almost disappears. These two cases will now be controlled by the BLT method. Firstly, ultimate gain k_u and ultimate frequency w_u of each diagonal transfer-function G_{ii} are calculated as in the classical SISO manner. For the high interaction case, the controller for the first pair is $K_{P1} = 0.5214$, $\tau_{I1} = 1.419$ and the second pair is $K_{P2} = 0.688$, $\tau_{I2} = 1.0267$. Similarly, for the low interaction case, the first pairing $K_{P1} = 0.2742$, $\tau_{I1} = 1.5355$, the values for the second pair are $K_{P2} = 0.3692$, $\tau_{I2} = 1.0267$. The performance of each case is decided by the error norm. The result of this method is highly unsatisfactory since the output is forced to a different value leading to error norms for the high interaction case and the low interaction case of 1.4594×10^4, 3.6687×10^5 respectively. This method is not successful since the interaction is ignored. The same problem also happens to the ICD method (O'Reilley and Leithead, 1991) where the controller PID cannot be tuned to be stable in order to track a reference signal. Instead the Matlab tool-box for tuning a PID for a SISO system is used for the MIMO case to get suitable constants. As with other previous approaches, this method ignores all the interaction of the system. The constants are shown in Table 4.6 for the high interaction case and the total norm error is 46.85. The low interaction case has the PID constants given in Table 4.7 and the total norm error for this case is 29.7061. There

Table 4.6: PID value based on Matlab PID toolbox

PID tuning values for high interaction case			
P_1	0.0240 ($\frac{N.m}{rad}$)	P_2	0.0197 ($\frac{N.m}{rad}$)
I_1	8.3881×10^{-5} ($\frac{N.m.s}{rad}$)	I_2	6.8871×10^{-5} ($\frac{N.m.s}{rad}$)
D_1	0.0022 ($\frac{N.m}{rad.s}$)	D_2	-0.0026 ($\frac{N.m}{rad.s}$)

is a large total norm error difference between the low and high interaction cases since the low interaction case is much easier to control with the high interaction case requiring more effort to achieve a reasonable error value. The tracking results for high interaction and low interaction cases are shown in Figure 4.9, Figure 4.10 respectively.

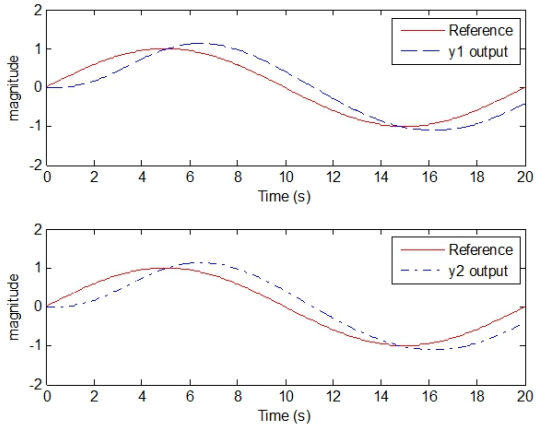


Figure 4.9: Tracking sine-wave for high interaction case

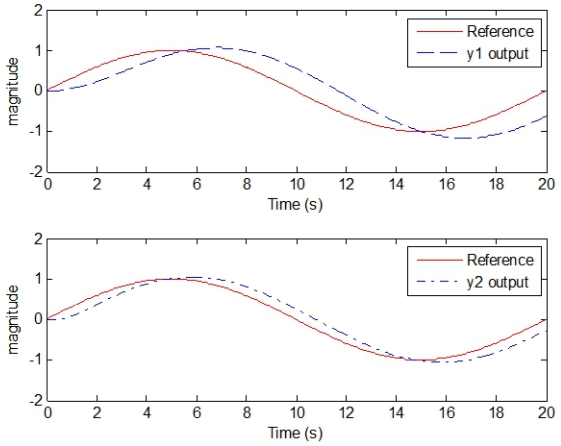


Figure 4.10: Tracking sine-wave for low interaction case

The results of this method are far superior to other two methods described in (O'Reilley and Leithead, 1991) and (Luyben, 1986) . It can enable the MIMO system to track two sine-wave reference signals but the accuracy is not high.

In the next section, the gradient ILC method will be used to control this MIMO system. The method does not need an accurate model description and it can automatically modify the input signal so that the output tacking can be improved significantly after a few trials of the tracking task.

Table 4.7: PID value based on Matlab PID toolbox

PID tuning values for low interaction case			
P_1	0.0810 ($\frac{N.m}{rad}$)	P_2	0.0887 ($\frac{N.m}{rad}$)
I_1	0.0124 ($\frac{N.m.s}{rad}$)	I_2	0.0055 ($\frac{N.m.s}{rad}$)
D_1	0.1066 ($\frac{N.m}{rad.s}$)	D_2	0.0104 ($\frac{N.m.s}{rad.s}$)

4.7 Simulations of gradient ILC using MIMO system model

All four configurations of the MIMO system have been simulated and compared. In Section 4.5, based on the relative gain array method the most suitable arrangement was chosen, and hence control design is conducted only for arrangement case 2, as in the previous section. In this section, a more advanced controller is used so that the system's tracking ability can be improved, hence confirming suitability and nominal performance prior to manufacture. The implemented controller is gradient ILC which was introduced in Section 2.3.3. The MIMO system has two outputs and requires two references . The first reference signal is a sine-wave and the second has a trapezium form. Both are given below

$$y_{d1} = \begin{cases} 0 & \text{if } 0 \leq t \leq 0.5 \\ sin(2\pi t) & \text{if } 0.5 < t \leq 5.5 \\ 0 & \text{if } 5.5 < t \leq 6 \end{cases} \tag{4.11}$$

$$y_{d2} = \begin{cases} 0 & \text{if } 0 \leq t \leq 0.5 \\ 0.05t & \text{if } 0.5 \leq t \leq 1.5 \\ 5 & \text{if } 1.5 < t \leq 4.11 \\ -0.05t & \text{if } 4.11 < t \leq 5.11 \\ 0 & \text{if } 5.11 < t \leq 6. \end{cases} \tag{4.12}$$

The results from Matlab for the case 2 system with low interaction are shown in Figure

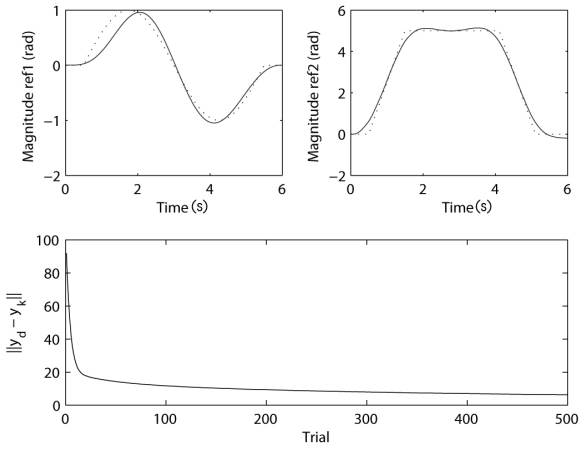


Figure 4.11: Results for the low interaction case.

4.11. The dotted lines are the desired output and the continuous line is the output from the model over the 500th trial. After trying many values of β the best value for this system was found to be 7.10^{-9}. The controller gives rise to a very impressive convergence rate over the first 20 trials, but then slows down to reach a final error of 6.2. Output 1 is more difficult to control since it responds to the input signal slower than the output 2. This explains why output 2 cannot track reference signal 1 in the first 2 seconds of each trial.

The medium interaction level has also been tested under the same conditions. It has the same reference signals of 6 second duration and a number of total trials 500. The results for this case is shown are shown in Figure 4.12. As expected, the interaction affects the performance, as it does with any controller hence the results are worse than the low interaction case. The error norm has the same convergence rate as before but after 500 trials the final value of error norm is 10, which is higher than when using the low interaction level. Similarly, the medium interaction case has the same difficulty in tracking reference 1 over the first few seconds of each trial.

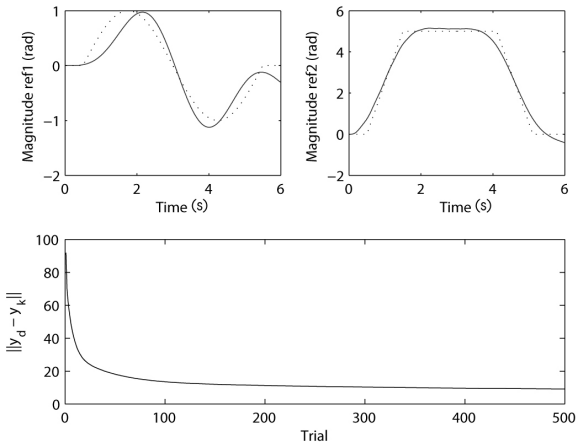


Figure 4.12: Results for medium interaction case.

The final investigated case is that of high interaction which is simulated similarly to the first two cases. Figure 4.13 shows that performance is far poorer than previously, and the convergence rate is much slower. This case needs around 100 trials to reach a value of 20 for error norm and after 500 trials the error norm is approximately 19. The interaction factor makes the MIMO system much harder to control.

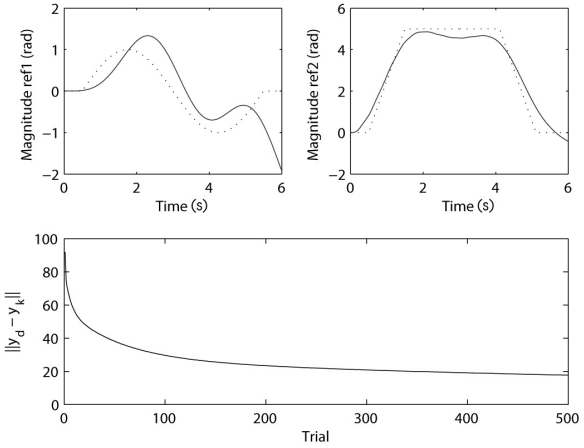


Figure 4.13: Results for high interaction case.

Over 500 trials the error norm of the three different levels of interaction was still high, therefore longer tests were performed to assess whether improvement would still occur.

The results are shown in Figure 4.14 with the x-axis taking a logarithm scale. The error norm results are shown to aid comparison. The high interaction case is still the most difficult to control, however the result for 5000 trials are still impressive clearly. It has a far superior error norm value over 500 trials. The error norm of the low interaction case is around 2 over the 5000th trial and 5 for the medium interaction case.

Compared to the PID controller tests in the previous section, the reference signal used for gradient ILC is much harder. The duration of each trial is much shorter and it has a really challenging velocity profile which the MIMO system finds difficult due to the high inputs needs. Therefore the error norm takes many more trials to reach a zero value. If the number of trials is large enough, the error norm will eventually reach a zero value.

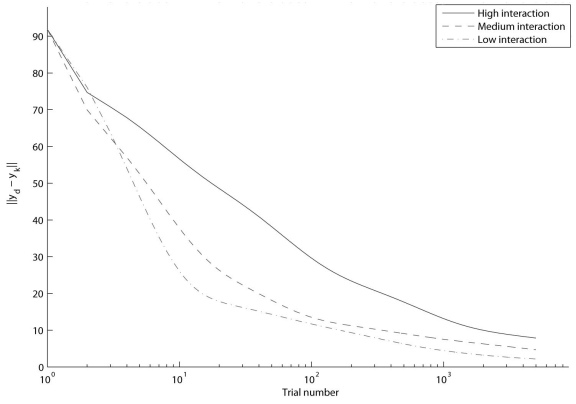


Figure 4.14: The error norm of 3 interaction levels.

Overall, the arrangement case 2 has been shown to be a suitable choice for manufacturing since it is easy to adjust the interaction level and the system is possible to control. The MIMO system also presents sufficient challenge for any implemented controller. Additionally, the simulations have shown the effect of interaction on the performance of the controller, which significantly elevates the control difficulty.

4.8 Disturbance injection

After having selected the system and parameters and confirmed the ability of ILC to enforce tracking, the additional capability to inject disturbance is now examined. The noise/disturbance will be externally generated via a DC motor therefore simulation in the presence of injected noise and disturbance can be synthesised easily via Simulink and Matlab. The DC motor is initially installed at the output of gear 1. The new arrangement is shown in Figure 4.15. The transfer-function is modified based on the analysis in Section 3.5. From gear 1 there is an additional input torque which is termed

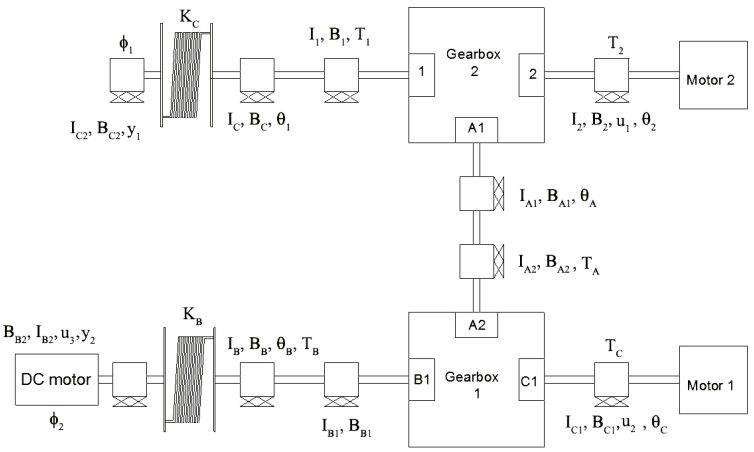


Figure 4.15: MIMO facility with disturbance/noise injection.

$\mathbf{u}_3$. The spring-mass-damper component has the new equation.

$$I_B\ddot{\theta}_B + B_B\dot{\theta}_B + K_B(\theta_B - \phi_B) = T_B \tag{4.13}$$

$$I_{B2}\ddot{\phi}_B + B_{B2}\dot{\phi}_B = K_B(\theta_B - \phi_B) + \mathbf{u}_3 \tag{4.14}$$

From (4.14)

$$I_{B2}\phi_B s^2 + B_{B2}\phi_B s = K_B\theta_B - K_B\phi_B + \mathbf{u}_3 \tag{4.15}$$

$$\Rightarrow \phi_B(I_{B2}s^2 + B_{B2}s + K_B) = K_B\theta_B + \mathbf{u}_3 \tag{4.16}$$

$$\Rightarrow \phi_B = \frac{K_B\theta_B + \mathbf{u}_3}{I_{B2}s^2 + B_{B2}s + K_B} \tag{4.17}$$

Substitute (4.17) into (4.13) to give

$$I_B\theta_B s^2 + B_B\theta_B s + K_B(\theta_B - \frac{K_B\theta_B + u_3}{I_{B2}s^2 + B_{B2}s + K_B}) = T_B \tag{4.18}$$

$$\Rightarrow \theta_B(I_B s^2 + B_B s + K_B - \frac{K_B^2}{I_{B2}s^2 + B_{B2}s + K_B}) = T_B + \frac{K_B u_3}{I_{B2}s^2 + B_{B2}s + K_B} \tag{4.19}$$

The transfer-function G_B is defined as

$$G_B = I_B s^2 + B_B s + K_B - \frac{K_B^2}{I_{B2}s^2 + B_{B2}s + K_B} \tag{4.20}$$

$$\Rightarrow T_B = \theta_B G_B - \frac{K_B u_3}{I_{B2}s^2 + B_{B2}s + K_B} \tag{4.21}$$

From gear 1, substitute (4.21) into (3.51) to give

$$\theta_B(I_{B1}s^2 + B_{B1}s - G_B) - \theta_C(I_{C1}s^2 + B_{C1}s) = -\frac{K_B u_3}{I_{B2}s^2 + B_{B2}s + K_B} - T_C. \tag{4.22}$$

Since $\theta_C = 2\theta_A - \theta_B$, substitute into (4.22) to give the relationship between θ_A and θ_B as

$$\theta_B\big[(I_{B1}+I_{C1})s^2+(B_{B1}+B_{C1})s-G_B\big]-2\theta_A(I_{C1}s^2+B_{C1}s)= \frac{-K_Bu_3}{I_{B2}s^2+B_{B2}s+K_B}-T_C$$

$$\Rightarrow \theta_A=\frac{\theta_B\big[(I_{B1}+I_{C1})s^2+(B_{B1}+B_{C1})s-G_B)\big]+T_C+\frac{K_Bu_3}{I_{B2}s^2+B_{B2}s+K_B}}{2(I_{C1}s^2+B_{C1}s)}. \tag{4.23}$$

Similarly substitute $\theta_B = 2\theta_A - \theta_C$ into (4.22) to give

$$2\theta_A(I_{B1}s^2+B_{B1}s-G_B)-\theta_C\big[(I_{C1}+I_{B1})s^2+(B_{C1}+B_{B1})s-G_B\big]= -\frac{K_Bu_3}{I_{B2}s^2+B_{B2}s+K_B}-T_C \tag{4.24}$$

$$\Rightarrow \theta_C=\frac{2\theta_A(I_{B1}s^2+B_{B1}s-G_B)+\frac{K_Bu_3}{I_{B2}s^2+B_{B2}s+K_B}+T_C}{(I_{C1}+I_{B1})s^2+(B_{C1}+B_{B1})s-G_B} \tag{4.25}$$

From (3.50)

$$\theta_A(I_{A2}s^2+B_{A2}s)+2\theta_C(I_{C1}s^2+B_{C1}s)=2T_C+T_A \tag{4.26}$$

Substitute (4.25) into (4.26) to give

$$\theta_A(I_{A2}s^2+B_{A2}s)+\frac{4\theta_A(I_{B1}s^2+B_{B1}s-G_B)+2T_C+\frac{2K_Bu_3}{I_{B2}s^2+B_{B2}s+K_B}}{(I_{C1}+I_{B1})s^2+(B_{C1}+B_{B1})s-G_B}(I_{C1}s^2+B_{C1}s) = 2T_C+T_A$$

$$\Rightarrow \theta_A\Big[I_{A2}s^2+B_{A2}s+\frac{4(I_{B1}s^2+B_{B1}s-G_B)(I_{C1}s^2+B_{C1}s)}{(I_{C1}+I_{B1})s^2+(B_{C1}+B_{B1})s-G_B}\Big]= 2T_C\Big(1-\frac{I_{C1}s^2+B_{C1}s}{(I_{C1}+I_{B1})s^2+(B_{C1}+B_{B1})s-G_B}\Big)+T_A- \frac{2K_Bu_3(I_{C1}s^2+B_{C1}s)}{(I_{B2}s^2+B_{B2}s+K_B)\big[(I_{C1}+I_{B1})s^2+(B_{C1}+B_{B1})s-G_B\big]}. \tag{4.27}$$

Therefore the transfer-functions G_{A1}, G_C are defined as

$$G_{A1} := I_{A2}s^2+B_{A2}s+\frac{4(I_{B1}s^2+B_{B1}s-G_B)(I_{C1}s^2+B_{C1}s)}{(I_{C1}+I_{B1})s^2+(B_{C1}+B_{B1})s-G_B} \tag{4.28}$$

$$G_C := 1-\frac{I_{C1}s^2+B_{C1}s}{(I_{C1}+I_{B1})s^2+(B_{C1}+B_{B1})s-G_B}. \tag{4.29}$$

Consequently (4.27) is equivalent to

$$\theta_AG_{A1}=2T_CG_C+T_A-\frac{2K_B(1-G_C)}{I_{B2}s^2+B_{B2}s+K_B} \tag{4.30}$$

Derivation for gear 2 is similar to that undertaken in Section 3.5 so the relationship between θ_A and the three inputs torques is

$$\theta_A(G_{A1}+G_{A2}) = 2G_C T_C - \frac{2K_B(1-G_C)}{I_{B2}s^2+B_{B2}s+K_B}u_3 + 2G_2T_2. \tag{4.31}$$

Substitute (4.23) into (4.31) to give

$$\theta_B\frac{(I_{B1}+I_{C1})s^2+(B_{B1}+B_{C1})s-G_B}{2(I_{C1}s^2+B_{C1}s)}(G_{A1}+G_{A2}) = T_C\big[2G_C - \frac{G_{A1}+G_{A2}}{2(I_{C1}s^2+B_{C1}s)}\big] + 2T_2G_2 - \frac{K_B}{I_{B2}s^2+B_{B2}s+K_B}\big[2(1-G_C)+\frac{G_{A1}+G_{A2}}{2(I_{C1}s^2+B_{C1}s)}\big]u_3. \tag{4.32}$$

In order to express the transfer-functions more concisely, G_{BA} is defined as

$$G_{BA} := \frac{(I_{B1}+I_{C1})s^2+(B_{B1}+B_{C1})s-G_B}{2(I_{C1}s^2+B_{C1}s)}(G_{A1}+G_{A2}) \tag{4.33}$$

Considering gear 2, from (3.74) substitute into (4.31) to give

$$\theta_1\frac{\big[(I_1+I_2)s^2+(B_1+B_2)s-G_1\big](G_{A1}+G_{A2})}{2(I_2s^2+B_2s)} = 2G_CT_C + \big[2G_2 - \frac{G_{A1}+G_{A2}}{2(I_2s^2+B_2s)}\big]T_2 - \frac{2K_B(1-G_C)}{I_{B2}s^2+B_{B2}s+K_B}u_3. \tag{4.34}$$

Therefore G_{1A} is defined as

$$G_{1A} := \frac{\big[(I_1+I_2)s^2+(B_1+B_2)s-G_1\big](G_{A1}+G_{A2})}{2(I_2s^2+B_2s)} \tag{4.35}$$

The overall transfer-function matrix for this arrangement is given by

$$\begin{bmatrix}\mathbf{y}_1(s)\\ \mathbf{y}_2(s)\end{bmatrix} = \begin{bmatrix}\underbrace{\begin{matrix}G_{11}(s) & G_{12}(s)\\ G_{21}(s) & G_{22}(s)\end{matrix}}_{G(s)} & \begin{matrix}G_{13}(s)\\ G_{23}(s)\end{matrix}\end{bmatrix}\begin{bmatrix}\mathbf{u}_1(s)\\ \mathbf{u}_2(s)\\ \mathbf{u}_3(s)\end{bmatrix} \tag{4.36}$$

where $\mathbf{u}_1 = T_2;\ \mathbf{u}_2 = T_C$. Each matrix component is written as

$$G_{11} := \frac{2G_2 - \frac{G_{A1}+G_{A2}}{2(I_2 s^2 + B_2 s)}}{G_{1A}} G_{AA} \tag{4.37}$$

$$G_{12} := \frac{2G_C}{G_{1A}} G_{AA} \tag{4.38}$$

$$G_{13} := \frac{2K_B(1-G_C)}{G_{1A}(I_{B2}s^2 + B_{B2}s + K_B)} G_{AA} \tag{4.39}$$

$$G_{21} := \frac{2G_2}{G_{BA}} G_{BB} \tag{4.40}$$

$$G_{22} := \frac{\left[2G_C - \frac{G_{A1}+G_{A2}}{2(I_{C1}s^2 + B_{C1}s)}\right]}{G_{BA}} G_{BB} \tag{4.41}$$

$$G_{23} := \frac{-K_B}{(I_{B2}s^2 + B_{B2}s + K_B)G_{BA}} \left[2(1-G_C) + \frac{G_{A1}+G_{A2}}{2(I_{C1}s^2 + B_{C1}s)}\right] G_{BB} \tag{4.42}$$

4.9 Simulation of case 2 with disturbance injection

Having selected a suitable configuration, injection of noise and disturbance is considered through addition of an extra input $\mathbf{u}_3$. The parameters are chosen as in Table 4.8. The impulse response for each interaction level is shown in Figure 4.16. The interaction level depends mainly on the coupling damper and mass inertia values therefore for the high interaction case, the values of B_{A1}, B_{A2}, I_{A1} and I_{A2} are very small. The impulse response for this case is shown in Figure 4.16(a). Here the steady-state value of G_{12} is close to the steady-state value of G_{11}. For output 2, the interaction level with input 1 is also high, which is reflected by the steady-state value of G_{21} being close to the steady-state value of G_{22}. In order to change the interaction level, the values B_{A1}, B_{A2} are modified and in the medium interaction case are equal to 0.0008 and 0.0001 respectively. These are shown in Figure 4.16(b), which illustrate that the steady-state value of G_{12}, G_{21} reduces significantly. Similarly, if the value of both B_{A1}, B_{A2} parameters increases to 0.001, this yields impulse responses for G_{12}, G_{21} which are much smaller than the two previous cases, as shown for the low interaction case which is displayed in Figure 4.16(c).

Table 4.8: Parameter values for arrangement 3.

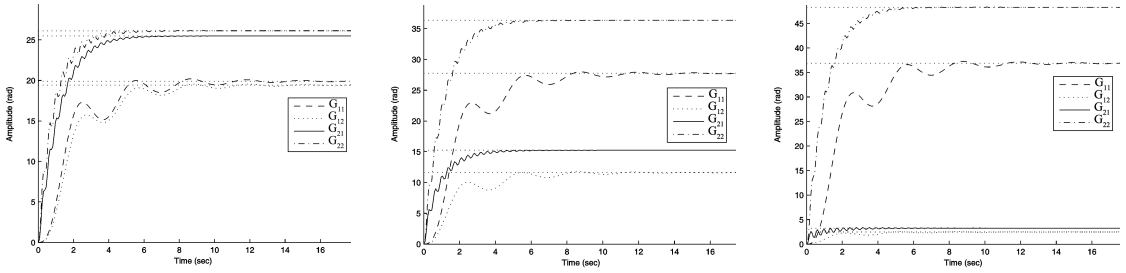
System parameter values							
I_1	10^{-4} $(kg.m^2)$	I_2	10^{-4} $(kg.m^2)$	I_{A1}	10^{-4} $(kg.m^2)$	I_{A2}	10^{-4} $(kg.m^2)$
I_{B1}	10^{-4} $(kg.m^2)$	I_{C1}	10^{-4} $(kg.m^2)$	B_{A1}	10^{-4} $(\frac{N.m.s}{rad})$	B_{A2}	10^{-4} $(\frac{N.m.s}{rad})$
B_1	0.0002 $(\frac{N.m.s}{rad})$	B_2	0.0002 $(\frac{N.m.s}{rad})$	B_{B1}	0.0002 $(\frac{N.m.s}{rad})$	B_{C1}	0.0002 $(\frac{N.m.s}{rad})$
I_B	0.01 $(kg.m^2)$	B_B	0.013 $(\frac{N.m.s}{rad})$	I_{C2}	0.021 $(kg.m^2)$	B_{C2}	0.014 $(\frac{N.m.s}{rad})$
I_C	0.03 $(kg.m^2)$	B_C	0.011 $(\frac{N.m.s}{rad})$	I_{B2}	0.01 $(kg.m^2)$	B_{B2}	0.008 $(\frac{N.m.s}{rad})$
K_B	1.5 $(\frac{N.m}{rad})$	K_C	0.05 $(\frac{N.m}{rad})$				

Having designed the physical system configuration and associated parameters, controllability and observability of a corresponding discrete-time state-space representation is established to ensure feasibility of later ILC approaches. Having finalised the system design the extra transfer-functions for the disturbance injection are derived as

$$G_{13} := \frac{2K_B(1-G_C)G_{AA}}{(I_{B2}s^2+B_{B2}s+K_B)G_{1A}} \tag{4.43}$$

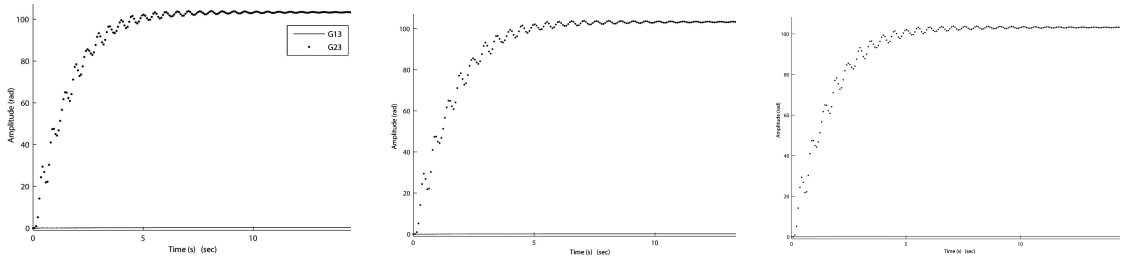
$$G_{23} := \left[\frac{-2K_B(1-G_C)}{I_{B2}s^2+B_{B2}s+K_B} - \frac{G_{A1}+G_{A2}}{2(I_{C1}s^2+B_{C1}s)}\frac{K_B}{I_{B2}s^2+B_{B2}s+K_B}\right]\frac{G_{BB}}{G_{BA}} \tag{4.44}$$

For all three cases of interaction the transfer-function matrix G_{13} gives the output impulse response, and does not vary markedly in term of transient response or steady-state value. The magnitude of steady-state is also very small compare to G_{11}, G_{12}, G_{21}, G_{22} in all three levels of interaction. This is shown clearly in Figure 4.17 therefore it can be neglected. At the same time, there is little difference in the response of G_{23}, and hence for simplicity G_{23} will be treated as the same transfer-function for all interaction levels. The remaining term in (4.36) are identical to the noise-free case.



(a) High interaction. (b) Medium interaction. (c) Low interaction.

Figure 4.16: Impulse response for arrangement 3 with disturbance injection.



(a) High interaction. (b) Medium interaction. (c) Low interaction.

Figure 4.17: Impulse response for MIMO system with disturbance injection.

All the transfer-functions for this configuration with disturbance injection have been derived and analysed carefully. It has confirmed suitability and therefore is chosen for manufacturing, prior to being used as a platform for implementing different controllers.

4.10 Summary

In this chapter many different system configurations have been derived and tested using RGA to ensure the system may be controlled but is sufficiently challenging to test controllers. Arrangement 2 has been chosen for manufacturing based on these factors. It has been simulated with both a PID controller and the gradient ILC method. The MIMO system has been tested in 3 levels of interaction. The results illustrate the role played by interaction, and the control difficulty inherent at high levels.

Disturbance injection has also been examined and feasibility established. Arrangement 2 enables interaction to be easily varied, and is the basis for more complex, expanded, system. In the next section the physical MIMO testbed is designed and manufactured.

Chapter 5

Experimental design and frequency based modeling

In Chapter 4 suitable parameter values were assigned to the components appearing in the MIMO system configuration. In this chapter the physical implementation of the testbed is addressed, together with choice of motors, materials, drivetrains, electrical systems and ancillary hardware. Identification of a model based on experimental data is also performed and evaluated.

5.1 Mechanical system design and component selection

In order to convert the designs of Chapter 3 into a practical system, there are many problems and obstacles that must be considered. One such issue is the choice of drive trains, and this has been solved by linking the two gearboxes using belt drives. Another issue was searching for suitable gearboxes, which was not trivial since the standard units commercially available include additional gearing, awkward input/output shaft positions, and usually have an unsuitable size. There are many other issues related to size, reduction of whirling friction and other unwanted properties, in addition to design for multi-configuration. Also the selection and design of spring-mass-damper components used to adjust the complexity of the coupling dynamics must be considered. The whole system was drawn using SolidWorks software which allows examination of the system in 3D so that all the dimensions can be checked carefully to ensure compatibility. Figure 5.1 shows the completed mechanical parts of the experimental test facility and details of components are given in Table 5.1.

The induction motors and drives were chosen to be of a type widely employed in industry. The differential gears selected are custom-made by Westgarage Engineering Services, UK to eliminate gearing and reduce inertia and friction. The real differential gears are

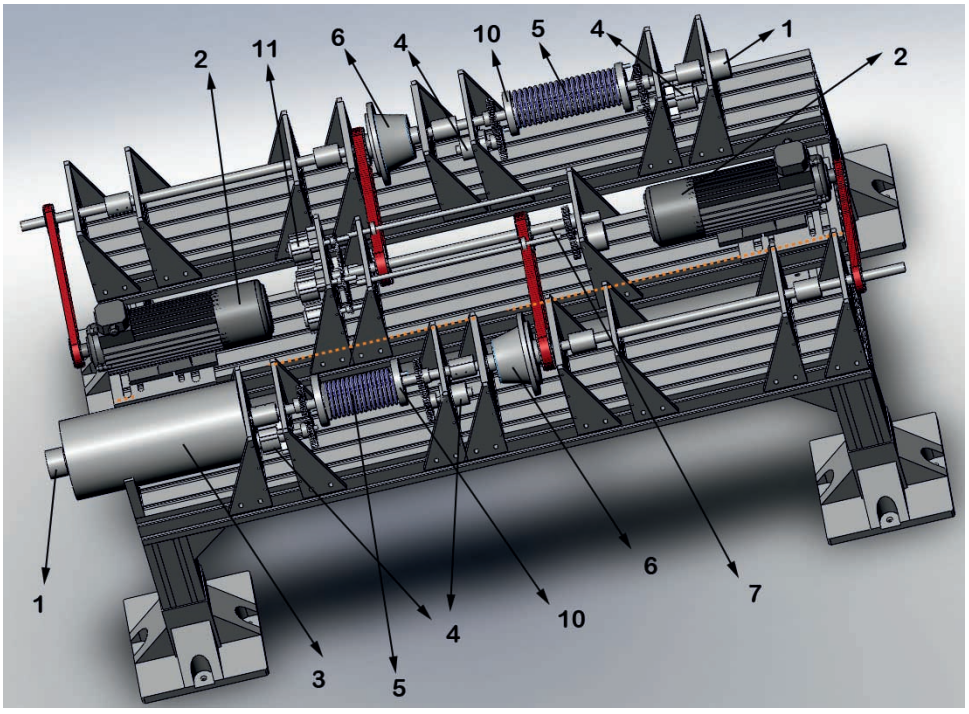


Figure 5.1: The system design in 3D.

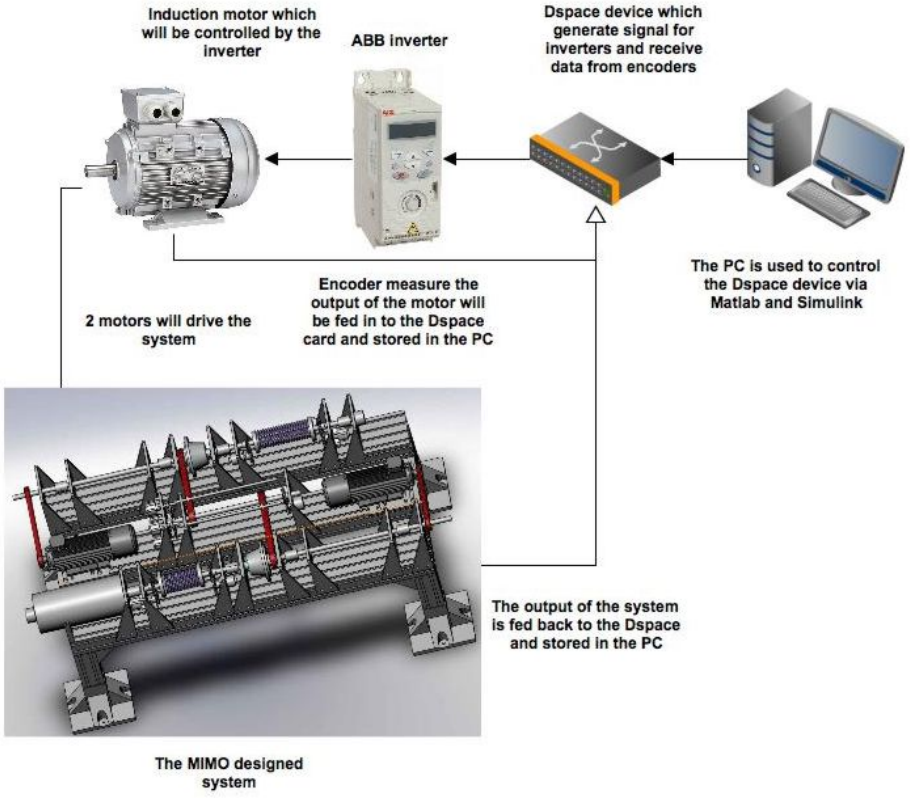


Figure 5.2: The completed system with control infrastructure.

different to the one in Figure 3.1. The real differential gears do not have housing but they are connected using belt drive therefore it still have all properties as discussed previously. Gearing between devices is implemented via belt drives to reduce the overall footprint and ensure low backlash. As described previously the interaction can be made easily adjustable in practice by combining B_{A1} and B_{A2} into a single parameter, which is realised by an adjustable number of dampers geared together to provide a suitable range of overall damping. Disturbance injection is realised mechanically using a DC motor which can be coupled to any drive shaft. The DC motor acts as a disturbance injector or alternatively plays the role of an extra load for the system. The speed/torque of DC motor is controlled via the dSpace card that is installed on the host PC, allowing injected noise profile to be adjusted in real-time. This maximises the scope of the system and the options available to the operator. The CAD drawings for individual components and those that were purchased can be seen in Appendix A.1. Figure 5.2 shows the electrical drive components and firmware. The final implementation of the two input, two output system is shown in Figure 5.3 with numbered components defined in Table 5.1. Note that this is only one possible configuration of inputs and outputs, since the position and number of motors can be easily altered whilst maintaining the same central structure.

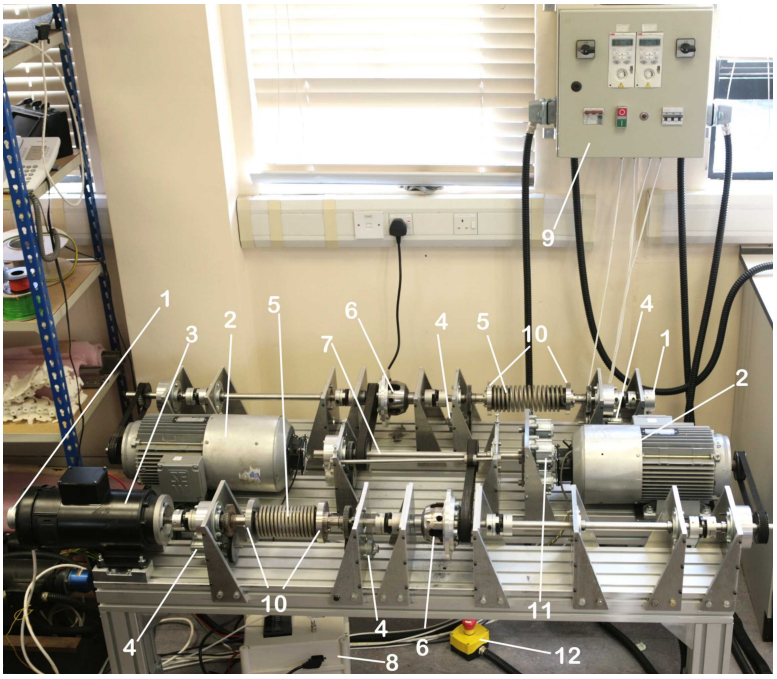


Figure 5.3: The complete MIMO system showing two induction motors and disturbance injection via a DC motor. Components labelled in Table 5.1.

Two induction motors drive the inputs and are equipped with integrated 2000 pulse per revolution encoders. The two remaining ports are initially designated system outputs and are fitted with 2500 pulses per revolution encoders. Each 1.1 kW AC induction motor is connected to an ABB inverter configured in torque mode, with an external demand signal supplied by real-time hardware comprising a ds1103 dSpace board which

interfaces directly with Matlab/Simulink to enable rapid controller development and implementation. The DC motor has been connected to one of the outputs but can be moved to inject variable noise and disturbance to any drive train. All the mechanical and electrical components have been fitted and tested. The electrical system incorporates the two inverters, four quadrant amplifier, and associated protection including emergency switches and circuit breakers.

The four adjustable dampers are shown in figure 5.3. If no interaction is required, there is a clamping device which will lock the coupling shaft (label 7). If the system is required to operate with extremely low interaction, all four dampers are geared to the coupling shaft which minimises the interaction between the two differential gearboxes. Similarly the low interaction case uses three dampers and the medium interaction case uses two dampers. The high interaction case has only one damper geared to the coupling shaft and the extremely high interaction case has no damper. Each level of interaction corresponds to a different transfer-function matrix which will be described later.

5.2 Frequency based modeling

In this section the frequency based modeling approach is applied, first to the inverter and motor system above, and then to the full MIMO system.

5.2.1 Inverter and induction motor model

Using a predetermined model structure is not only computationally intensive but also leads to difficulty in taking into account factors such as friction or other disturbances which are challenging to precisely model. Therefore the final representation often cannot describe a physical system accurately and sometimes can be over complicated and ill-suited to controller design. An alternative method that can be used instead is called frequency modeling. The method uses a harmonic function at a single frequency as the input signal which is applied to the system. The output of the system is measured and plotted on a Bode plot. This is then repeated for other frequencies of interest. A linear

Table 5.1: Components of the MIMO system

System components			
1	Encoder	2	Induction motor
3	DC motor	4	Damper
5	Spring	6	Differential gearbox
7	Coupling shaft	8	DC motor controller
9	Inverters and circuit breakers	10	Mass
11	Adjusting interaction dampers	12	Emergency button

model is then fitted to match the experimental results, using the Least Mean Squares (LMS) approach. The linear model is used to represent the experimental system since it has a similar frequency response and can be assumed to capture its steady-state behavior. This method will be applied to all the components to find approximate transfer-functions. Clearly this method is only suitable for systems which can be adequately modeled by linear dynamics.

An inverter is a device which is used to control the speed or torque of an induction motor. It achieves this by varying the frequency of the main power supply fed to the motor. A typical full electronic circuit is given in (Rashid, 1993) and is in wide spread use. The output of the system is measured using encoders which provide the position of the motors. Where to derive a transfer-function which describes how the input signal (voltage) and the output signal (position of the motor) are related, different sine-wave frequencies are fed into the system, then the position of the motor is recorded over a fixed time length. The graph of magnitude (dB) against frequency (Hz) is then plotted with each measurement the average of three separate tests. A transfer-function is then fitted based on standard Bode plotting rules together with LMS optimisation. This method does not require accurate knowledge regarding the actuator, drives circuit or mechanical system. The only requirement is to record an accurate output signal from the system. The method has been applied successfully in (Cai, 2009), (Barton, 1999) and is now applied to each combined ABB 2.2 kW inverter and induction motor without the additional mechanical components attached. The latter includes an integrated British encoder. This allows the technique to be refined prior to application on the full system. In order to subsequently derive a suitable transfer-function which has a frequency response similar to the experimental results, the identity function method can be used (Dutton et al., 1997). In order to increase accuracy, the SISO tool-box function in Matlab is used to tune the frequency response and display results without the need for further calculation. Unfortunately it has been found that there is a problem with the output of the motor when the injected frequency is high. The output drifts from the zero value, and fluctuates both up and down, an example being given in Appendix C.1. This problem can be solved by using the average value of crests and trough values, therefore with each frequency the phase shift and the magnitude can still be calculated. Finally, the result of the fitting process is shown in Figure 5.4 which shows that the fitted transfer-function is very close to the experimental results. The blue line is the fitted transfer-function, the red line is the average of 3 repeated measurements of the output. Each output is displayed on the same graph so that the accuracy of the model can be observed. At frequency around 38 rad/s, there is a sudden change in magnitude since this is due to the amplitude of the reference signal being changed from 10V to 1V which changes the operating point of the identification. This has been done because of the limitation of the motor. The input current for the motor is about 3A but with a high frequency (larger than 3Hz) and amplitude of 10V, the inverter will generate more than the allowed maximum current. Consequently the inverter will stop automatically and report an over-current fault. As a

result, the amplitude is set to 1V so that the inverter can work within the limited current range. All the data from the encoder and inverter can be seen in Appendix B.1. The

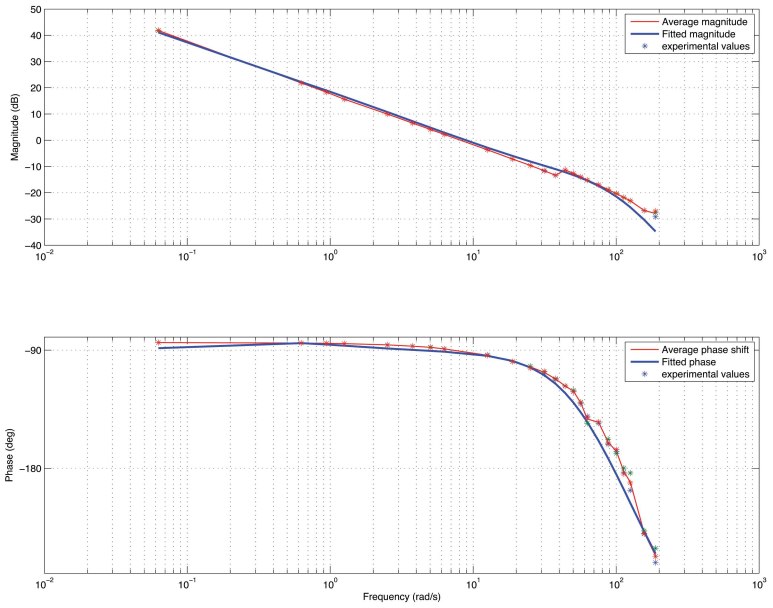


Figure 5.4: The fitting transfer-function compare to the experimental results.

transfer-function obtained for this system is

$$G_{\text{Induction motor}} = \frac{43466875.8037(s+32.43)(s+0.4212)}{s(s+0.515)(s+85.14)(s+104.1)(s+122.3)(s+149.5)} \tag{5.1}$$

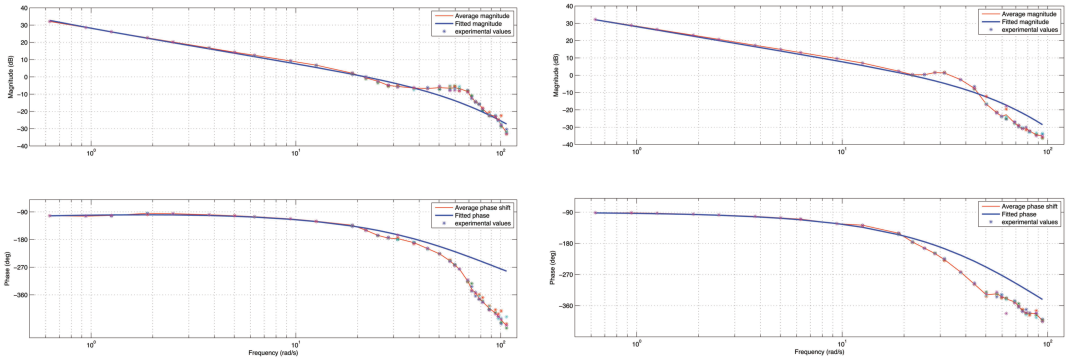
Similarly, when the full system is ready to operate the same technique is used to obtain a new transfer-function which has a lower order than that derived through the analysis of Chapter 3. This transfer-function will replace those models in subsequent model-based control.

5.2.2 Mechanical system modeling and validation

The frequency modeling method has been applied to the MIMO mechanical system. To increase accuracy, the DC motor amplifier is turned on throughout these tests, but receives a zero demand signal. It hence will not inject any noise or disturbance into the system. The testbed facility has different levels of interaction therefore each interaction level will be tested and will lead to a unique model. Additionally, the system has 2 inputs and 2 outputs so each side of the system necessitates separate tests involving injecting a sine-wave in one input applying to the other a zero value. The 2 outputs are then measured and recorded. The experiment collects 5 times and it is averaged for the final value. The interaction level is defined as c, which is varied in the range [0 0.2 0.4 0.6 0.8 1]. Here $c = 0$ denotes no interaction at all and corresponds to two parallel SISO systems operating independently. Conversely $c = 1$ refers to the maximum

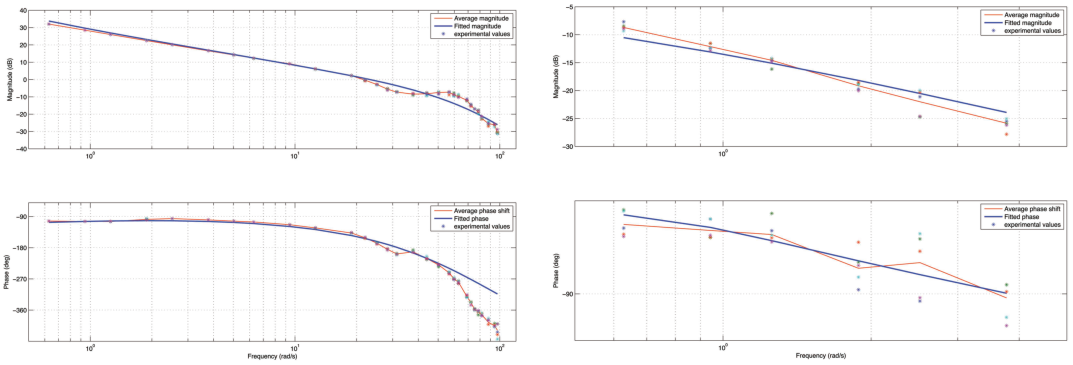
level of interaction. The input signal fed in has the form $\mathbf{u}(t) = A\sin(w2\pi t)$ with the sampling time is $T_s = \frac{1}{300}$ where A is the amplitude and w is the frequency in Hz. In this experiment A is 2 and the frequency is in the range from 0.1 to 17 (Hz).

Firstly, coupling shaft is clamped so that the interaction is eliminated completely. The input signal is fed in as described previously. The fitting for this case is shown in Figure 5.5 and the transfer-function is

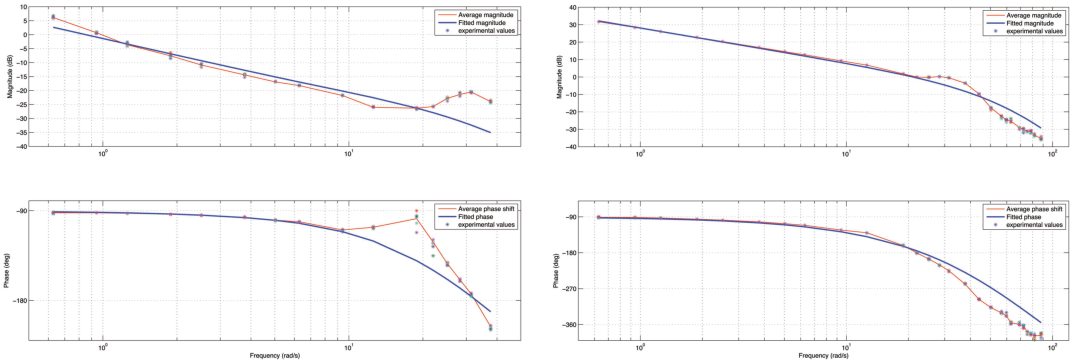


(a) G_{11} (b) G_{22}

Figure 5.5: Bode plot showing fitting for no interaction case, $c = 0$.

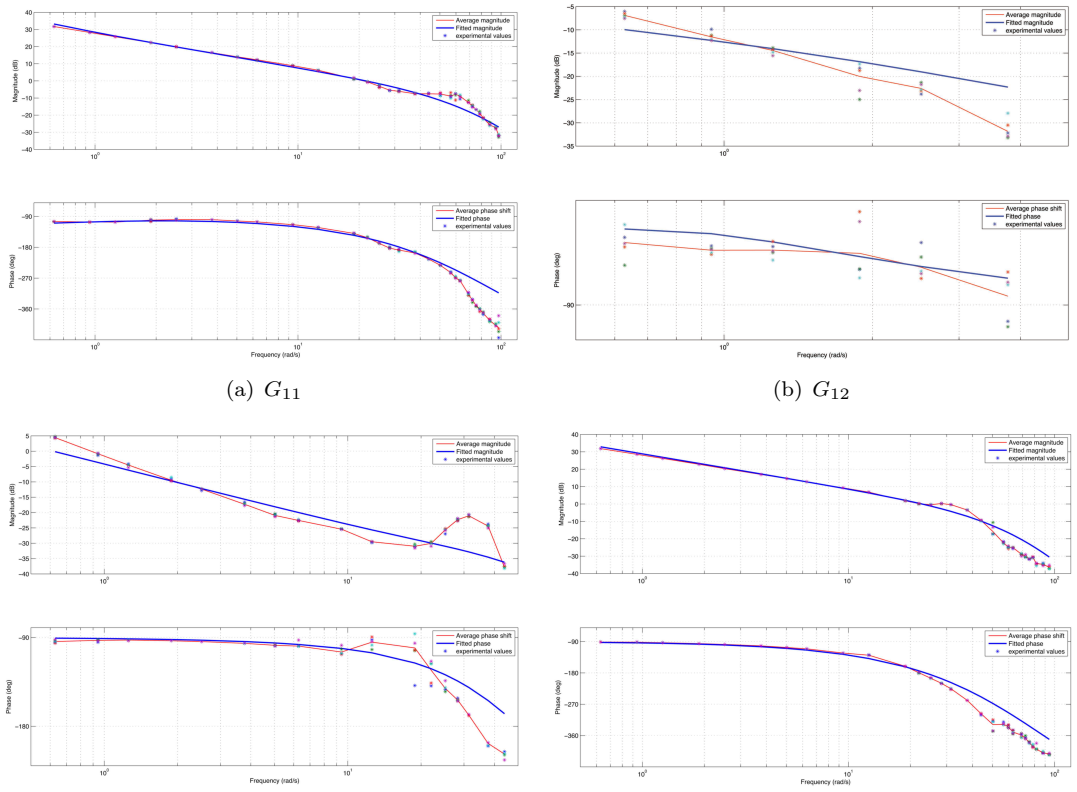


(a) G_{11} (b) G_{12}



(c) G_{21} (d) G_{22}

Figure 5.6: Bode plot fit for extreme low interaction case, $c = 0.2$.



(a) G_{11} (b) G_{12}

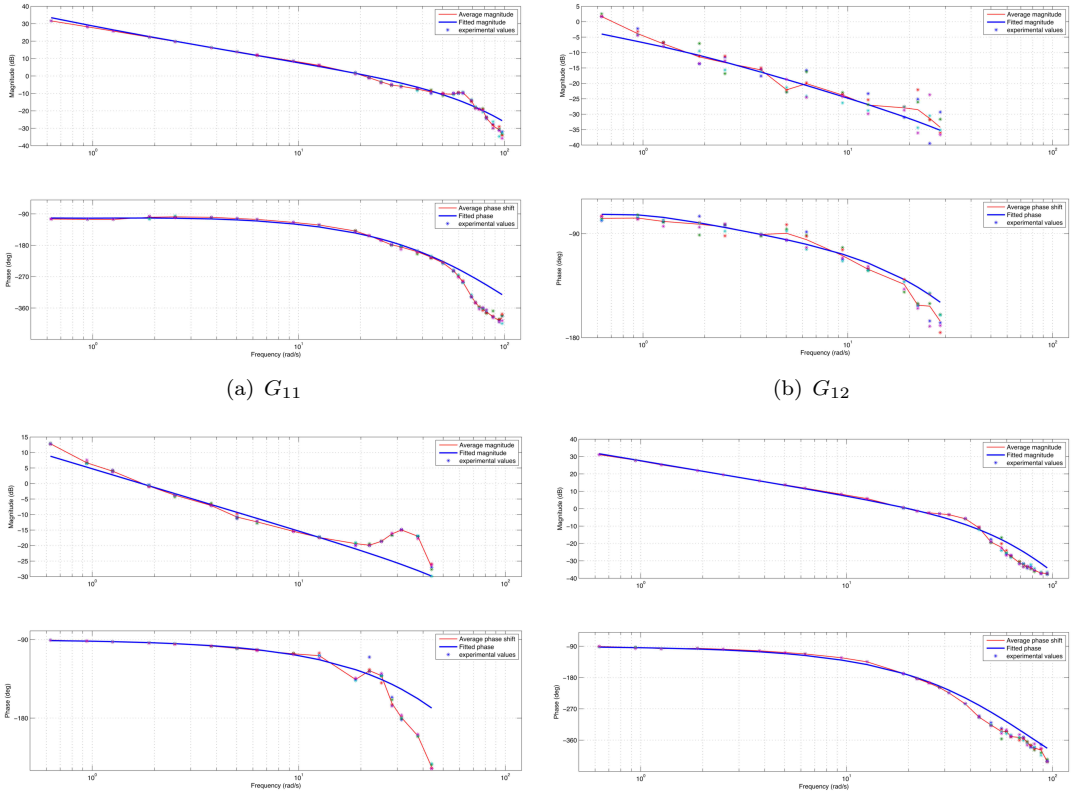
(c) G_{21} (d) G_{22}

Figure 5.7: Bode plot fit for low interaction case, $c = 0.4$.

$$
\begin{aligned}
G_{11}^{0}(s) &= \frac{1031255286(s+0.4308)}{s(s+0.272)(s+81.93)^2(s+92.56)(s+133.5)} \\
G_{12}^{0}(s) &= 0 \\
G_{21}^{0}(s) &= 0 \\
G_{22}^{0}(s) &= \frac{17801570474710}{s(s+59.75)(s+75.81)(s+88.13)(s+139.8)(s+153.3)(s+159.4)}.
\end{aligned} \tag{5.2}
$$

The method is then repeated for the extremely low interaction level which uses all four dampers to minimise the the extremely low interaction case ($c = 0.2$). Therefore the transfer-functions are

$$
\begin{aligned}
G_{11}^{0.2}(s) &= \frac{111826099400(s+0.4652)}{s(s+0.2488)(s+98.55)^5} \\
G_{12}^{0.2}(s) &= \frac{9(s+0.3822)}{s(s+0.7257)(s+43.36)} \\
G_{21}^{0.2}(s) &= \frac{11103002065(s+11.43)}{s(s+18.22)(s+27.94)(s+104.7)(s+137.5)(s+165.7)(s+244.4)} \\
G_{22}^{0.2}(s) &= \frac{7155482699531}{s(s+55.28)(s+75.43)(s+78.42)(s+102.9)(s+125)(s+130)},
\end{aligned} \tag{5.3}
$$



(a) G_{11} (b) G_{12}

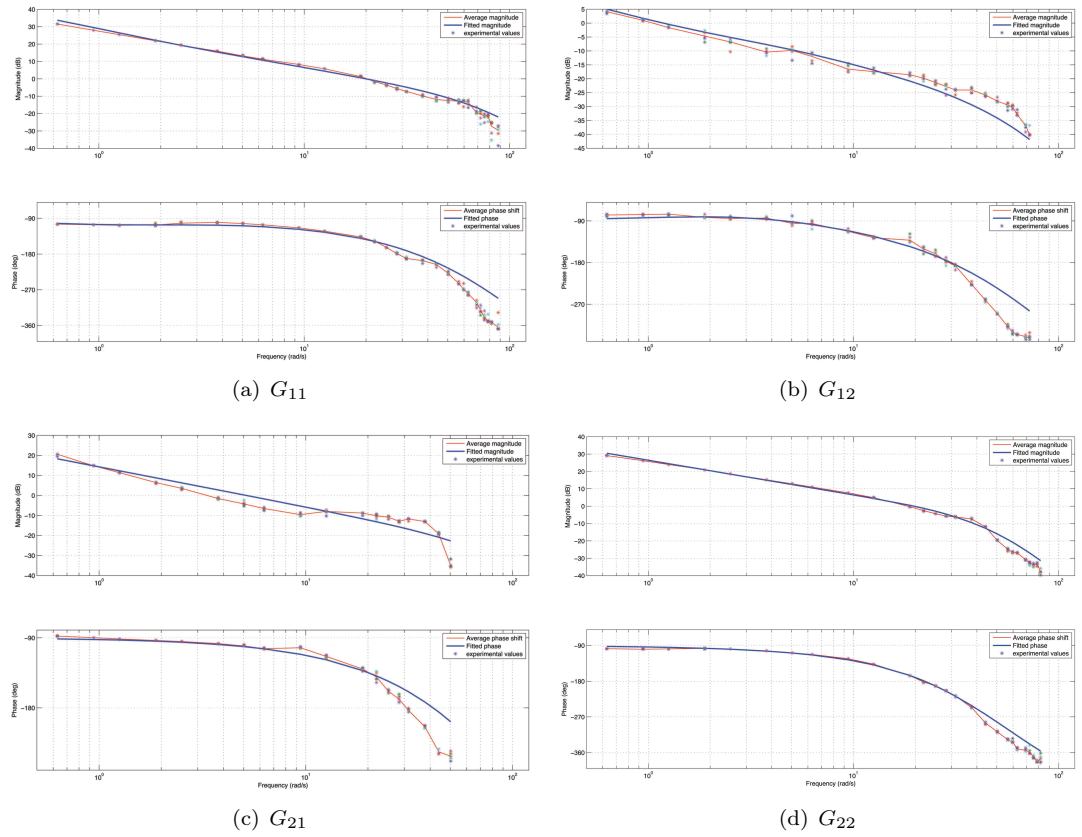
(c) G_{21} (d) G_{22}

Figure 5.8: Bode plot fit for medium interaction case, $c = 0.6$.

which has the Bode plot shown in Figure 5.6. The low interaction case ($c = 0.4$) operates with 3 dampers and the resulting transfer-function is

$$\begin{aligned} G_{11}^{0.4}(s) &= \frac{108961369833(s+0.15)}{s(s+0.2284)(s+83.96)(s+91.63)(s+97.13)(s+109.1)(s+115.7)} \\ G_{12}^{0.4}(s) &= \frac{2488(s+0.6273)}{s(s+1.168)(s+133.8)^2} \\ G_{21}^{0.4}(s) &= \frac{112194440(s+24.33)}{s(s+70.49)(s+73.28)(s+76.18)(s+92.52)(s+121.4)} \\ G_{22}^{0.4}(s) &= \frac{6806671977274}{s(s+80.3)^2(s+87.63)(s+92.89)^2(s+98.47)}, \end{aligned} \tag{5.4}$$

which is corresponding to the result in Figure 5.7. The extreme low interaction level and low interaction level do not have an accurate Bode plot fit since it is hard to detect the interaction level in high frequency due to the spring and dampers. Therefore the transfer-function is not accurate as expected. However it will not affect system performance too much because the interaction is not big and ILC controllers can learn and minimise it. The medium interaction case ($c = 0.6$) is then tested with the same method and the



(a) G_{11} (b) G_{12}

(c) G_{21} (d) G_{22}

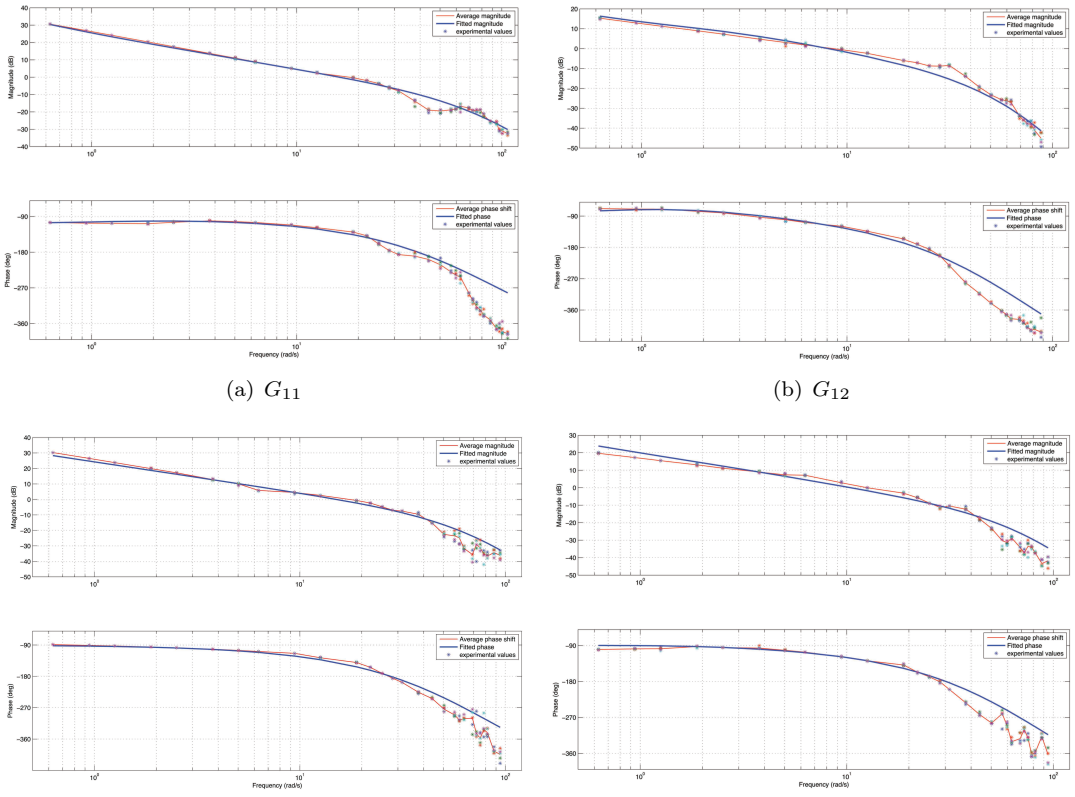
Figure 5.9: Bode plot fit for high interaction case, $c = 0.8$.

transfer-function is

$$\begin{aligned} G_{11}^{0.6}(s) &= \frac{40024701578405(s+0.1143)}{s(s+0.5739)(s+90.3)(s+93.34)(s+107.6)(s+121.3)(s+159.4)(s+201.3)} \\ G_{12}^{0.6}(s) &= \frac{160480(s+1.6)}{s(s+1.103)(s+64.95)(s+75.87)(s+95.79)} \\ G_{21}^{0.6}(s) &= \frac{122499582151}{s(s+115.7)(s+131.3)(s+171.5)(s+186.2)(s+217.5)} \\ G_{22}^{0.6}(s) &= \frac{4354268187193}{s(s+64.58)(s+73.64)(s+84.78)(s+87.71)(s+88.13)(s+98.55)}, \end{aligned} \tag{5.5}$$

with the Bode plot in Figure 5.9. The high interaction case ($c = 0.8$) and the extreme high interaction level ($c = 1$) are tested using the same technic therefore the transfer-functions are

$$\begin{aligned} G_{11}^{0.8}(s) &= \frac{61629241234845(s+0.2)}{s(s+0.9133)(s+116.7)(s+117.7)^2(s+123.9)(s+160.1)(s+177.4)} \\ G_{12}^{0.8}(s) &= \frac{3067892780938(s+0.2)}{s(s+3.935)(s+51.15)(s+103)(s+122.6)(s+130)(s+154.8)(s+184.4)} \\ G_{21}^{0.8}(s) &= \frac{33637744568707}{s(s+90.3)(s+146.1)(s+148.1)(s+151.9)(s+195.5)(s+345.1)} \\ G_{22}^{0.8}(s) &= \frac{15626746219(s+20.13)}{s(s+42.32)(s+49.2)(s+53.18)(s+55.02)^2(s+64.58)} \end{aligned} \tag{5.6}$$



(a) G_{11} (b) G_{12}

(c) G_{21} (d) G_{22}

Figure 5.10: Bode plot shows fitting for extreme high interaction case, $c = 1$.

for the high interaction level and

$$\begin{aligned} G^1_{11}(s) &= \frac{883232998(s+0.09)}{s(s+0.3641)(s+76.93)(s+86.44)(s+109.1)(s+115.7)} \\ G^1_{12}(s) &= \frac{1168992866700(s+1.0)}{s(s+2.374)(s+51.15)(s+62.12)(s+72.57)(s+103)^2(s+120.3)} \\ G^1_{21}(s) &= \frac{60812424193}{s(s+63.76)(s+71.65)(s+87.86)(s+110.9)^2} \\ G^1_{22}(s) &= \frac{40744427000(s+1.649)}{s(s+1.852)(s+69.8)(s+78.44)(s+81.54)(s+122.6)(s+146.1)}. \end{aligned} \tag{5.7}$$

for the extremely high interaction level. The Bode plots for these cases are shown in Figure 5.9 and Figure 5.10 respectively. The greater the interaction level the more data can be collected so that the Bode plot fitting method can achieve a better accuracy since the graph can show more information of the facility. The system only operates over frequency ranges up to 40 $rads^{-1}$ so that the transfer-functions for all interaction levels fit well up to this frequency. Hence transfer-functions have a smaller order compared to those of Chapter 3, leading to simpler controller implementation.

5.3 Disturbance injection

The DC motor is used as a disturbance or noise injector therefore in order to characterise the device it also needs to be identified. The same technique is applied to obtain such a model. The input signal here has magnitude $A = 1.5$ and a frequency range from 0.1 Hz to 7 Hz. The system is tested at the medium interaction level. The Bode plot fit is shown in Figure 5.11 and the transfer-function is

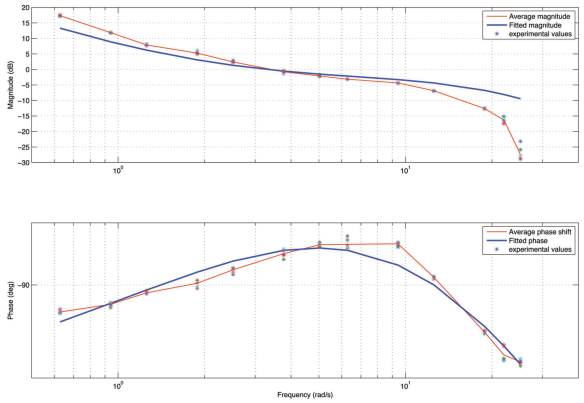


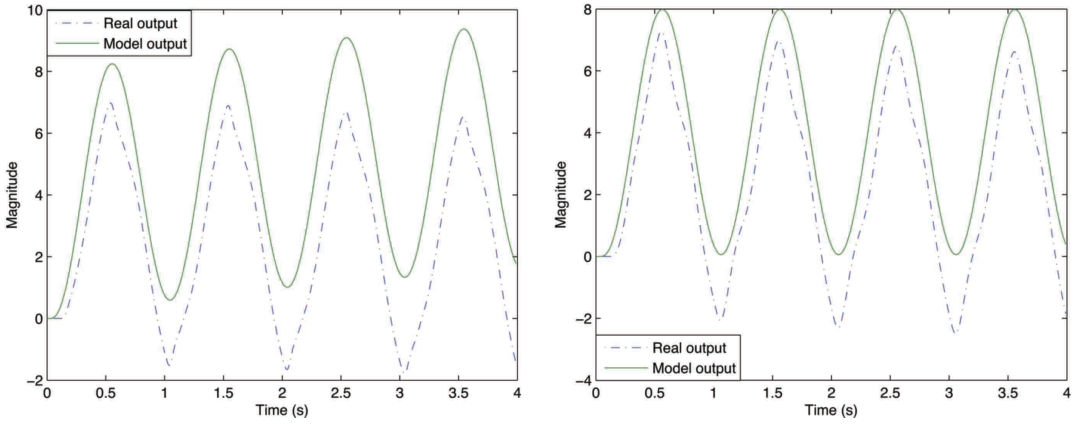
Figure 5.11: Bode plot fit for disturbance injector.

$$G_{23} = \frac{1848694.6893(s + 2.889)(s + 0.5334)}{s(s + 0.08998)(s + 25.35)(s + 28.6)(s + 46.34)(s + 72.85)}.$$

The Bode plot shows the DC motor acts as a form of a low pass filter so when noise is applied to the amplifier the range of frequencies is generated between 1 Hz to 4 Hz only. Higher than this frequency the system will not give any output. The transfer-function for the other output has not been derived because it has negligible amplitude. It will hence be assumed to be zero.

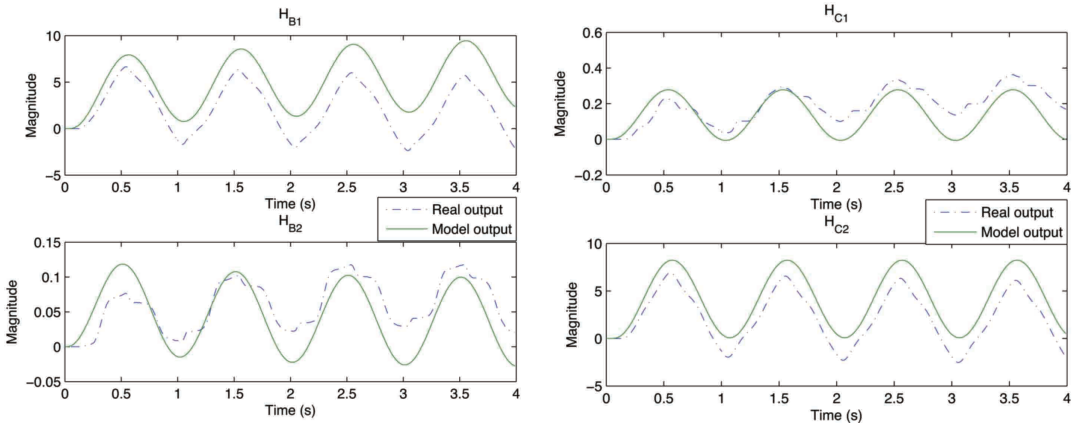
5.4 Model validation

Each transfer-function derived from the Bode plot fitting has been validated using a sine-wave signal with different values of frequency. The output of each model should approximate the real output of the system at that frequency. For simplicity a sine-wave with magnitude 2 and frequency 1 Hz is applied to all the identified transfer-functions of the MIMO system. Figure 5.12 shows the validation result for the zero interaction case. The model output is not exactly the same as the simulated output but it is acceptable since the drift effect is caused from the integrator. The other transfer-function for different levels of interactions have been validated similarly. Figure 5.13-5.17 shows the performance of these models compared to the real output. The output of each model is reasonably accurate. The final transfer-function that needs to be tested is that of



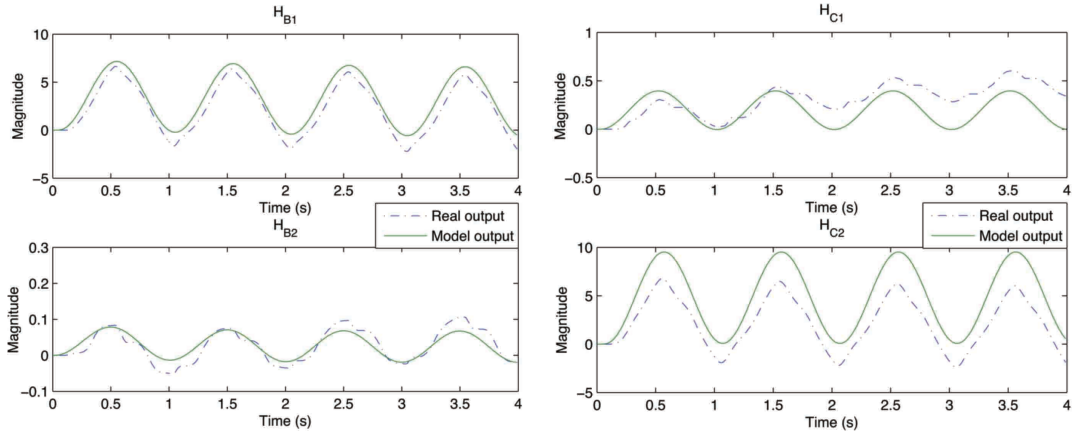
(a) G_{11} (b) G_{12}

Figure 5.12: Validation of the transfer-functions corresponding to no interaction case.



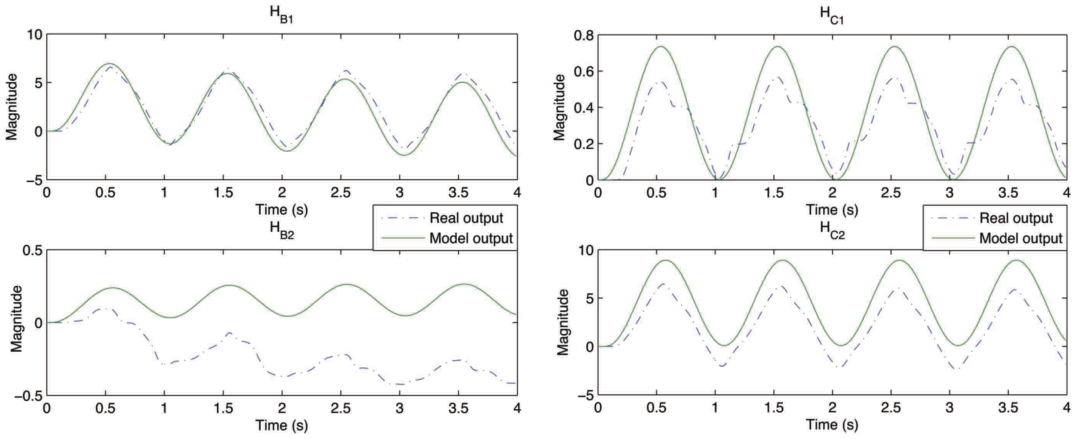
(a) G_{11} and G_{12} (b) G_{21} and G_{22}

Figure 5.13: Validation the transfer-functions of the extremely low interaction case.



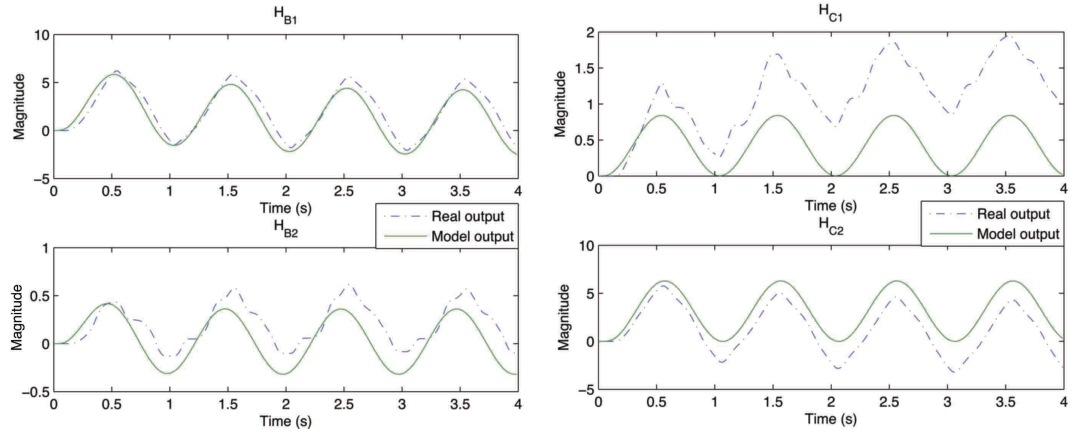
(a) G_{11} and G_{12} (b) G_{21} and G_{22}

Figure 5.14: Validation of the transfer-functions of the low interaction case.



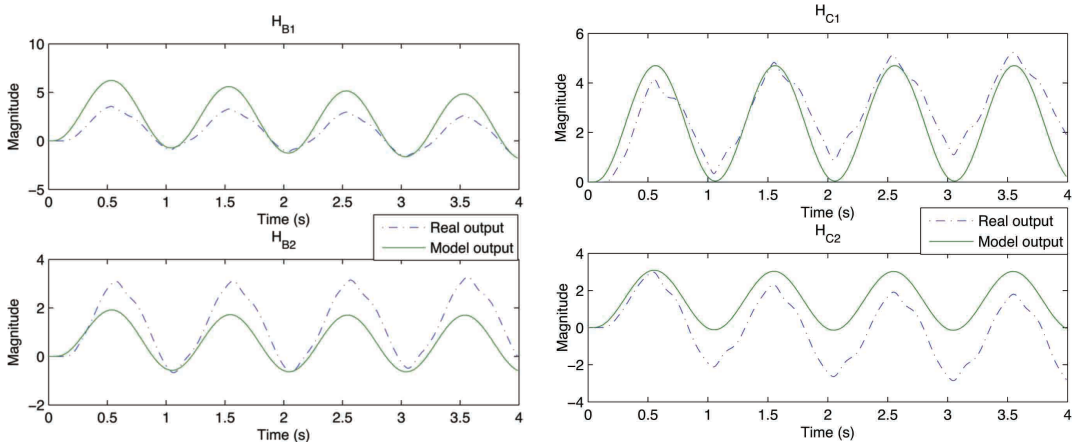
(a) G_{11} and G_{12} (b) G_{21} and G_{22}

Figure 5.15: Validation of the transfer-functions of the medium interaction case.



(a) G_{11} and G_{12} (b) G_{21} and G_{22}

Figure 5.16: Validation of the transfer-functions of the high interaction case.



(a) G_{11} and G_{12} (b) G_{21} and G_{22}

Figure 5.17: Validation of the transfer-functions of the extremely high interaction case.

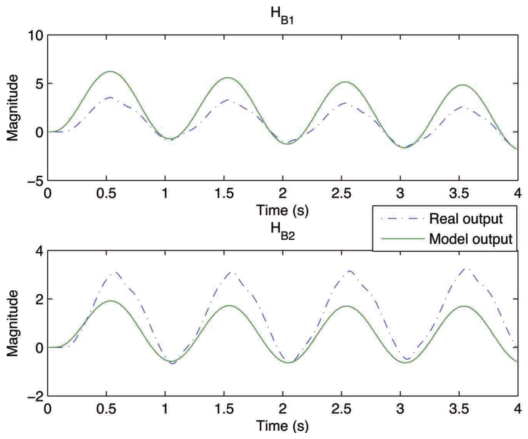


Figure 5.18: Validation the transfer-functions of the DC motor.

the DC motor and the result is shown in figure 5.18. From the medium interaction level and above, there is a drift phenomenon which model cannot incorporate. The use of a linear model hence represents a significant challenge in controlling the MIMO system. Especially in the highest interaction level, the interaction magnitude is approximately equal in size to the direct outputs. The lack of more accurate models mean the MIMO system needs to have robust controllers so that it can cope with uncertainty in modeling and the presence of nonlinear factors.

5.5 Summary

The system configurations of Chapter 3 and parameter selection results of Chapter 4 have been physically realised in the form of a practical system. This has been constructed, commissioned and tested, including layout, component selection, hardware, firmware and electrical system.

This chapter derived and validated models the MIMO system. For each level of interaction, different models were derived and will be used for model based controller design. The model accuracy can be judged by examining associated Bode plot fit to experimental data. Moreover the method yields a lower order system than those derived in Chapter 3 so enable quicker computation and easier implementation.

Chapter 6

Basic ILC controllers

6.1 Introduction

As described in the previous chapters, the MIMO facility has been completed, commissioned and tested. Before applying advanced controllers, basic ILC algorithms are applied to investigate baseline performance. Additionally, these experiments confirm the efficiency of the system. The simple controllers are termed P-type and D-type and were first proposed by Arimoto et al. (1985), Arimoto et al. (1984), Kawamura et al. (1985), Arimoto and Kawamura (1988). These methods have previously been applied to several SISO systems but there have been few experimental results performed on a MIMO system. Another simple ILC method, termed phase-lead ILC, has been applied to control a conveyor system and yielded impressive results (Barton et al., 2000). These basic controllers do not require a system model and have few parameters, which can be tuned heuristically by the operator. In this section these methods will be applied to the MIMO facility. This chapter also provides analysis and evaluation of each controller with reference to effects of parameter variation and a discussion of system performance.

6.2 Multivariable system description

The analysis which follows requires a general MIMO system description which is now introduced. Links are established between system norms that will be central to future performance measures.

Consider an m input, p output discrete-time system expressed by the transfer-function matrix

$$G(z) = \begin{bmatrix} G_{11}(z) & \dots & G_{1m}(z) \\ \vdots & \ddots & \vdots \\ G_{p1}(z) & \dots & G_{pm}(z) \end{bmatrix} \tag{6.1}$$

whose components are assumed to be stable, and, for simplicity, to have the same relative degree. The off-diagonal subsystems $G_{ij}(z)$, $i \neq j$, govern the interaction between input and outputs pairs, and their size is reflected in the H_2-norm of G, given by

$$\|G(z)\|_2 = \left(\frac{1}{2\pi} \int_0^{2\pi} \sum_{i,j} |G_{ij}(e^{j\theta})|^2 d\theta \right)^{1/2} \tag{6.2}$$

$$= \left(\sum_{i,j} ||G_{ij}(z)||_2^2 \right)^{1/2} = \left(\sum_{i,j} \|g_{ij}\|_2^2 \right)^{1/2} \tag{6.3}$$

with g_{ij} the impulse response of $G_{ij}(z)$. Hence the norm of $G(z)$ directly increases if the norm of any of the off-diagonal systems $G_{ij}(z)$, $i \neq j$ increases. Similarly the H_∞-norm of G, is given by

$$\|G(z)\|_\infty = \max_{\theta=[0,2\pi]} \bar{\sigma} \left(G(e^{j\theta}) \right) \tag{6.4}$$

and illustrates the effect of interaction terms in influencing the 'high-gain direction' which is associated with the maximum singular value $\bar{\sigma}\left(G(e^{j\theta})\right)$. Robustness, convergence and control effort properties are now examined with specific reference to the role of interaction dynamics. This requires the following lemma which relates the magnitude of off-diagonal system entries to the maximum singular value of GG^T.

To aid subsequent controller development and analysis, the system (6.1) is represented in state-space form as

$$\begin{array}{rcll} \mathbf{x}(i+1) & = & A\mathbf{x}(i) + B\mathbf{u}(i), & \mathbf{x}(0) = \mathbf{x}_0 \quad 0 \leq i \leq N \\ \mathbf{y}(i) & = & C\mathbf{x}(i), & \mathbf{x}(i) \in \mathbb{R}^n \quad \mathbf{u}(i) \in \mathbb{R}^m \quad \mathbf{y}(i) \in \mathbb{R}^p \end{array} \tag{6.5}$$

running over a trial length of N samples with state resetting between each trial. Hence $G(z) = C\left(zI - A\right)^{-1} B$ with $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$ with n the number of states. This leads to the impulse response components $g_{ij}(q) = \left(CA^{q-1}B\right)_{ij}$, $q = 1, 2, \ldots$. Since ILC runs over a finite interval, the input and output sequences of (6.5) can be expressed by the supervectors

$$\mathbf{u} = [\mathbf{u}(0)^T, \mathbf{u}(1)^T, \ldots, \mathbf{u}(N-1)^T]^T \quad \in \mathbb{R}^{mN}, \tag{6.6}$$

$$\mathbf{y} = [\mathbf{y}(n_r)^T, \mathbf{y}(n_r+1)^T, \ldots, \mathbf{y}(N+1-n_r)^T]^T \quad \in \mathbb{R}^{pN}. \tag{6.7}$$

where the relative degree of each subsystem, n_r, has been used to shift the output relative to the input to ensure that each output component can be influenced by the input. The

input/output time-series relationship can then be represented by $y = Gu + y_0$ where

$$G = \begin{bmatrix} CA^{n_r-1}B & 0 & 0 & \cdots & 0 \\ CA^{n_r}B & CA^{n_r-1}B & 0 & \cdots & 0 \\ CA^{n_r+1}B & CA^{n_r}B & CA^{n_r-1}B & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{N+2-n_r}B & CA^{N+1-n_r}B & CA^{N-n_r}B & \cdots & CA^{n_r-1}B \end{bmatrix} \in \mathbb{R}^{pN \times mN} \tag{6.8}$$

and

$$\mathbf{y}_0 = \begin{bmatrix} CA^{n_r} \\ CA^{n_r+1} \\ CA^{n_r+2} \\ \vdots \\ CA^{N+3-n_r} \end{bmatrix} \mathbf{x}_0 \tag{6.9}$$

Here $\mathbf{y}_0$ is the response to initial conditions whose effect can be absorbed into the reference trajectory, so that without loss of generality it is assumed $\mathbf{y}_0 = 0$, or equivalently $\mathbf{x}_0 = 0$.

Lemma 6.1. *Consider the supervector matrix G given by (6.8) corresponding to the state-space system (6.5). Assuming N exceeds the time taken for all impulse responses to approximately go to zero, the maximum singular value satisfies*

$$\bar{\sigma}\left(GG^T\right) \geq \max\left\{ \frac{1}{mN} \cdot \max_{1\leq i\leq p} \left(\sum_{j=1}^{m} \|g_{ij}\|_1 \right)^2, \frac{1}{pN} \cdot \max_{1\leq j\leq m} \left(\sum_{i=1}^{p} \|g_{ij}\|_1 \right)^2 \right\} \tag{6.10}$$

Proof. By applying standard matrix norms to G and exploiting the Toeplitz structure, it can be shown that

$$\begin{aligned} \|G\|_\infty = \max_{1\leq i\leq pN} \sum_{j=1}^{mN} |G_{ij}| &= \max_{\substack{1\leq i\leq p \\ 1\leq q\leq N}} \sum_{j=1}^{m}\sum_{n=1}^{q} |g_{ij}(n)| \\ &= \max_{1\leq i\leq p} \sum_{j=1}^{m}\sum_{n=1}^{N} |g_{ij}(n)| \\ &= \max_{1\leq i\leq p} \sum_{j=1}^{m} \|g_{ij}\|_1 , \\ \|G\|_1 = \max_{1\leq j\leq mN} \sum_{i=1}^{pN} |G_{ij}| &= \max_{\substack{1\leq j\leq m \\ 1\leq q\leq N}} \sum_{i=1}^{p}\sum_{n=1}^{q} |g_{ij}(n)| \\ &= \max_{1\leq j\leq m} \sum_{i=1}^{p}\sum_{n=1}^{N} |g_{ij}(n)| \\ &= \max_{1\leq j\leq m} \sum_{i=1}^{p} \|g_{ij}\|_1 \end{aligned} \tag{6.11}$$

Hence $\|G\|_2$ can be bounded using

$$\frac{1}{\sqrt{mN}}\|G\|_\infty \leq \|G\|_2 \leq \sqrt{pN}\,\|G\|_\infty, \qquad \frac{1}{\sqrt{pN}}\|G\|_1 \leq \|G\|_2 \leq \sqrt{mN}\,\|G\|_1 \quad (6.12)$$

to give

$$\|G\|_2 \geq \max\left\{\frac{1}{\sqrt{mN}}\cdot\max_{1\leq i\leq p}\sum_{j=1}^{m}\|g_{ij}\|_1, \frac{1}{\sqrt{pN}}\cdot\max_{1\leq j\leq m}\sum_{i=1}^{p}\|g_{ij}\|_1\right\} \quad (6.13)$$

which leads to (6.10) since $\sqrt{\bar{\sigma}\left(GG^T\right)} = \|G\|_2$. □

Since interaction terms directly contribute to the right-hand side of (6.10), greater interaction increases the lower bound of $\bar{\sigma}\left(GG^T\right)$.

The maximum singular values $\overline{\sigma}(GG^T)$ are 456.51, 457.06, 457.11, 459.01, 488.04, 565.01 for the zero interaction case, the extremely low interaction case, the low interaction case, the medium interaction case, the high interaction case and the extremely high interaction case respectively. These confirm that, as c increases, so too does $\overline{\sigma}(GG^T)$. This is also reflected in plots of the RGA number in Dinh et al. (2012c) which rises as c is increased. The connection between c and $\overline{\sigma}(GG^T)$ will form a crucial component of future analysis.

Remark 6.2. It can further be shown that

$$\|G(z)\|_\infty = \|G\|_2 = \bar{\sigma}\left(G\right) = \sqrt{\bar{\sigma}\left(GG^T\right)} \quad (6.14)$$

and hence the bound (6.10) links the magnitude of off-diagonal system components to the l_∞-norm of $G(z)$, given by (6.4).

6.3 Proportional type ILC

The P-type ILC algorithm from Arimoto et al. (1985) is given in discrete-time for multivariable systems by

$$\mathbf{u}_{k+1}(i) \;=\; \mathbf{u}_k(i) + \gamma\mathbf{e}_k(i) \quad (6.15)$$

where γ is a constant matrix defined by

$$\gamma = \begin{bmatrix} \gamma_{11} & \gamma_{12} & \cdots & \gamma_{1p} \\ \gamma_{21} & \gamma_{22} & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \gamma_{m1} & \cdots & \cdots & \gamma_{mp} \end{bmatrix} \quad (6.16)$$

the input, reference, output are given as

$$\begin{aligned}
\mathbf{u}_k &= [\mathbf{u}_k(0)^T, \mathbf{u}_k(1)^T, \ldots, \mathbf{u}_k(N-1)^T]^T \quad \in \mathbb{R}^{mN}, && (6.17)\\
\mathbf{y}_d &= [\mathbf{y}_d(n_r)^T, \mathbf{y}_d(n_r+1)^T, \ldots, \mathbf{y}_d(N+1-n_r)^T]^T \quad \in \mathbb{R}^{pN}, && (6.18)\\
\mathbf{y}_k &= [\mathbf{y}_k(n_r)^T, \mathbf{y}_k(n_r+1)^T, \ldots, \mathbf{y}_k(N+1-n_r)^T]^T \quad \in \mathbb{R}^{pN}. && (6.19)
\end{aligned}$$

and $\mathbf{e}_k = \mathbf{y}_d - \mathbf{y}_k$. The objective of ILC is to force the system to track the reference signal such that the error norm decreases and reaches zero in a finite number of trials. At the same time the controller signal approaches a fixed signal. This requirement is stated as

$$\lim_{k\to\infty} ||\mathbf{e}_k|| = 0, \qquad \lim_{k\to\infty} ||\mathbf{u}_k - \mathbf{u}_d|| = 0. \tag{6.20}$$

Over the k^{th} trial the relationship between input and output in discrete-time can be expressed by $\mathbf{y}_k = G\mathbf{u}_k$ where $\mathbf{u}_d$ is the ideal control input such that the output equals the reference $\mathbf{y}_d$.

The error progression is given by

$$\begin{aligned}
\mathbf{e}_{k+1} &= \mathbf{y}_d - \mathbf{y}_{k+1} && (6.21)\\
&= \mathbf{y}_d - G(\mathbf{u}_{k+1}) && (6.22)\\
&= \mathbf{y}_d - (G\mathbf{u}_k + GL\mathbf{e}_k) && (6.23)\\
&= \mathbf{e}_k - GL\mathbf{e}_k && (6.24)\\
&= (I - GL)\mathbf{e}_k && (6.25)
\end{aligned}$$

where $L = \text{diag}\{\gamma, \ldots, \gamma\}$. Therefore a necessary and sufficient condition for convergence of an arbitrary error signal is hence

$$\max_i |\lambda_i(I - GL)| < 1 \tag{6.26}$$

where λ_i is the i^{th} eigenvalue. The matrix γ is typically chosen experimentally and for safety reasons its elements should be chosen starting from a low value to avoid aggressive behavior of the MIMO facility. Note that the condition in (6.26) can only be satisfied if the system relative degree is zero and $p \leq m$. In particular, for LTI SISO systems, $\lambda_i = 1 - \gamma h(0)$ where $h(0)$ is the first Markov parameter of the system. The MIMO facility is a 2×2 system therefore $m, p = 2$ and

$$\gamma = \begin{bmatrix} \gamma_{11} & \gamma_{12} \\ \gamma_{21} & \gamma_{22} \end{bmatrix}. \tag{6.27}$$

The interaction has been adjusted to zero ($c = 0$) and increases to the highest interaction level (c = 1). The diagonal gain values $\gamma_{11} = \gamma_{22}$ and off-diagonal gain values $\gamma_{21} = \gamma_{12} = 0$ have been chosen after trying many different values which all cause the system to be

unstable. The results are shown in Figure 6.2 with tracking references shown in Figure 6.1. References are sine-waves with different magnitudes and the same period. Reference 1 corresponding to output 1 has a magnitude of 7 *rad* while the magnitude of reference 2 is 5 *rad*. Reference 2 is shifted around 60 degrees compared to reference 1.

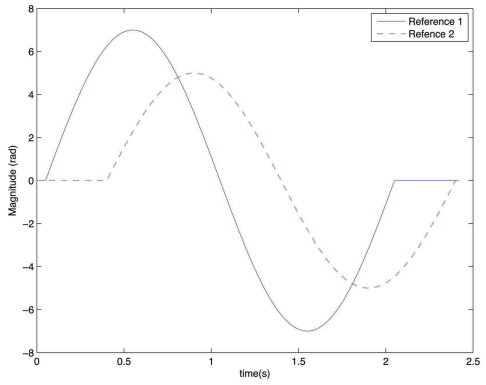


Figure 6.1: References for basic ILC controllers.

For all interaction levels, when γ_{11} is set equal to 0.01 and 0.02, the error norm keeps increasing and the MIMO system becomes unstable after the first few trials. A gain of $\gamma_{11} = -0.01$ is then used and the error norm is seen to reduce over the first 10 trials but after that the error norm increases with a higher rate than the other 2 values. The poor performance is not surprising since this method does not employ any model information.

P-type ILC is often implemented in combination with a PID controller or other feedback controller but to ensure a fair comparison here no feedback controller is used. The heuristic approach to tuning is time consuming and potentially damages the system. Hence an alternative scheme is necessary.

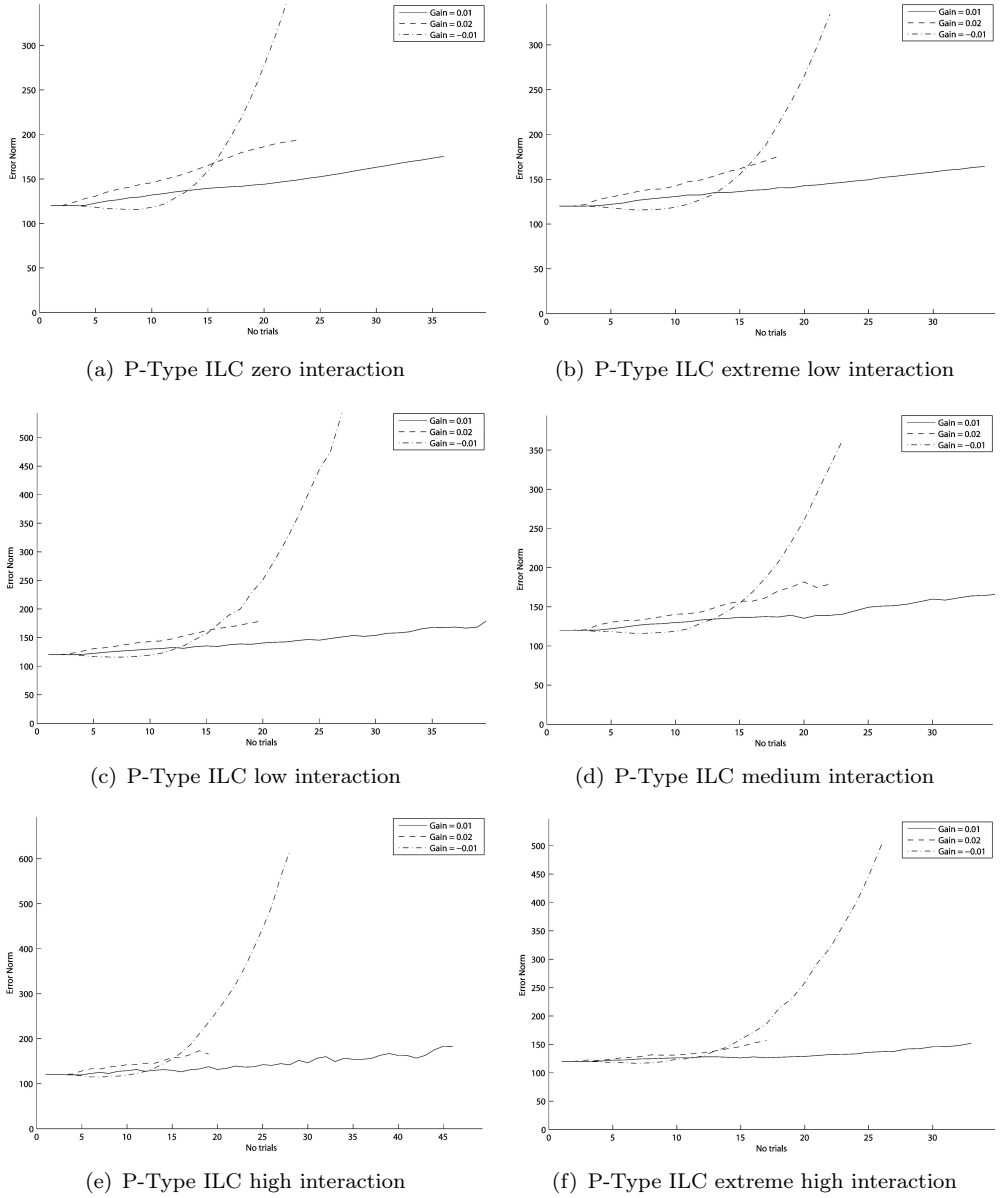
6.4 Derivative type ILC

Another approach to control this system is D-type ILC, which uses the derivative of the error to update the controller input. The controller update is given as (2.23) in continuous-time and is derived below for discrete-time. Similar to the discrete-time P-type ILC in the previous section, the control update is expressed as

$$\mathbf{u}_{k+1}(i) \quad = \quad \mathbf{u}_k(i) + \gamma \frac{\mathbf{e}_k(i+1) - \mathbf{e}_k(i)}{T_s} \tag{6.28}$$

where T_s denotes time sampling. In supervector form, this is given by

$$\mathbf{u}_{k+1} = \mathbf{u}_k + \gamma D \mathbf{e}_k \tag{6.29}$$



(a) P-Type ILC zero interaction

(b) P-Type ILC extreme low interaction

(c) P-Type ILC low interaction

(d) P-Type ILC medium interaction

(e) P-Type ILC high interaction

(f) P-Type ILC extreme high interaction

Figure 6.2: P-type ILC with varied gain.

where

$$D = \frac{1}{T_s}\begin{bmatrix} -1 & 1 & 0 & 0 & \dots & 0 \\ 0 & -1 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 1 \\ 0 & 0 & 0 & 0 & \dots & -1 \end{bmatrix} \in \mathbb{R}^{mN \times pN} \tag{6.30}$$

γ is defined in (6.16) and $L = \text{diag}\{\gamma, \dots, \gamma\}$. The necessary and sufficient condition for convergence is hence

$$\max_i |\lambda_i(I - GLD)| < 1 \tag{6.31}$$

and it can be shown that this can be satisfied if the system has relative degree 1 and $p \leq m$.

The method is next applied with different levels of interaction using the same references as in the previous section. For simplicity the gain assumes the form $\gamma_{11} = \gamma_{22}$ and $\gamma_{12} = \gamma_{21} = 0$. This effectively ignores all interaction and provides a fair comparison with P-type ILC. Additionally, it is difficult to find values for off-diagonal components since they can easily make the system unstable.

The results for this method are displayed in Figure 6.3. For the zero interaction case, shown in Figure 6.3(a), the convergence over the first few trials is reasonable for all chosen gain values. The larger the gain the faster the convergence. However the error norm diverges in a fewer number of trials. For the biggest gain, 0.02, the error norm diverges after 6 trials while for a 0.01 gain, it diverges after 15. Lower gain values, 0.005 and 0.002, experience divergence after 24 and 75 trials respectively. When increasing the interaction level, the same convergence rate trends occur but it is more difficult for the controller enforce force reference tracking. This is especially true in the high interaction case and the extremely high interaction case shown in Figure 6.4(e) and Figure 6.4(f) respectively. Here the minimum error norm achieves 75, which is higher than other interaction levels.

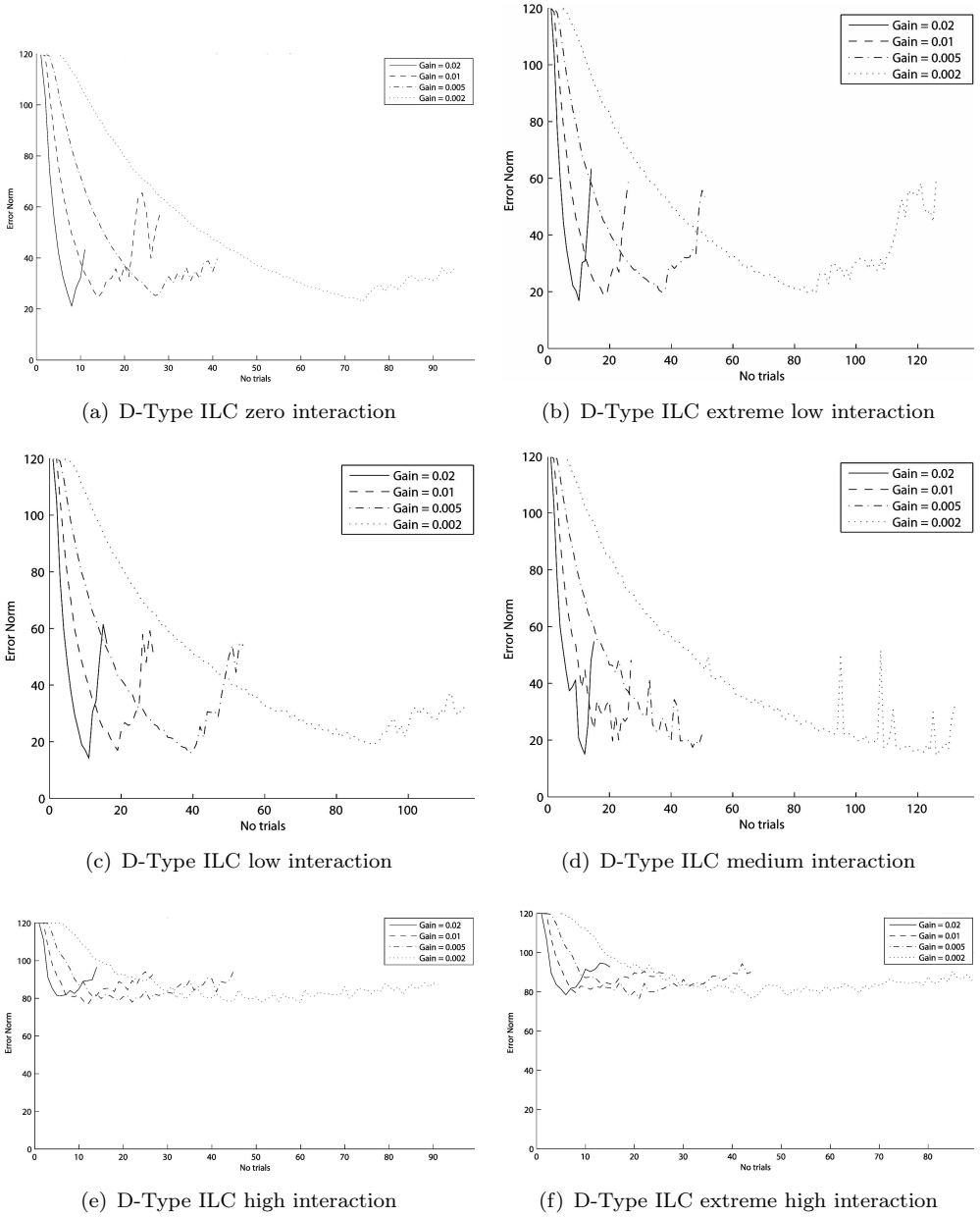
Over all interaction levels, D-type ILC cannot drive the error norm to a low level. The error norm diverges after a few trials, whose number depends on the chosen gain constant. Compared to P-type ILC this method is superior since it can force the outputs to track references and reduce the error norm after a few trials. However, the heuristic tuning method also requires much effort to find a suitable gain constant matrix for the controller and the operator cannot be sure that the chosen gain is the optimum value. As the number of inputs and outputs increase, it will take more time to find a suitable range of gains in matrix γ.

6.5 Phase-lead ILC

Phase-lead is a popular method that is used to control a practical system. For example, in Tutty et al. (2012), phase-lead ILC was used for controlling load on wind turbines, in Freeman et al. (2011b) it was used for stroke rehabilitation and in Wallén et al. (2008) it was applied to industrial robotics. The controller does not require a model and hence can be applied easily in practice. The controller is given by

$$\mathbf{u}_{k+1}(i) = \mathbf{u}_k(i) + \gamma \mathbf{e}_k(i+\delta) \tag{6.32}$$

$$\mathbf{e}_k(i) = \mathbf{y}_d(i) - \mathbf{y}_k(i), \tag{6.33}$$



(a) D-Type ILC zero interaction

(b) D-Type ILC extreme low interaction

(c) D-Type ILC low interaction

(d) D-Type ILC medium interaction

(e) D-Type ILC high interaction

(f) D-Type ILC extreme high interaction

Figure 6.3: D-type ILC with varied gain.

where δ is a positive integer time delay and γ is defined in (6.16). This can be written in supervector form as

$$\mathbf{e}_{k+1} = (I - G\gamma S)\mathbf{e}_k \tag{6.34}$$

where the shift operator

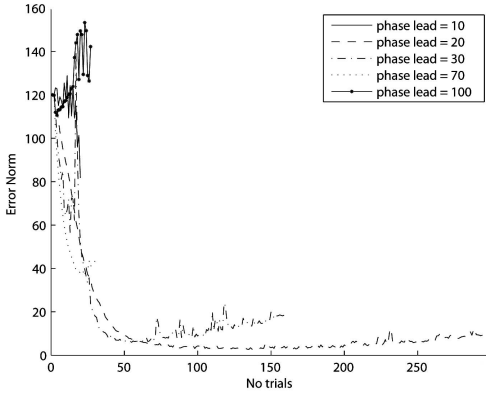
$$S = \begin{bmatrix} 0 & \dots & 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 1 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \dots & 0 \end{bmatrix} \in \mathbb{R}^{pN \times pN} \tag{6.35}$$

and $L = \text{diag}\{\gamma, \dots, \gamma\}$. Hence a necessary and sufficient condition for convergence to zero error is

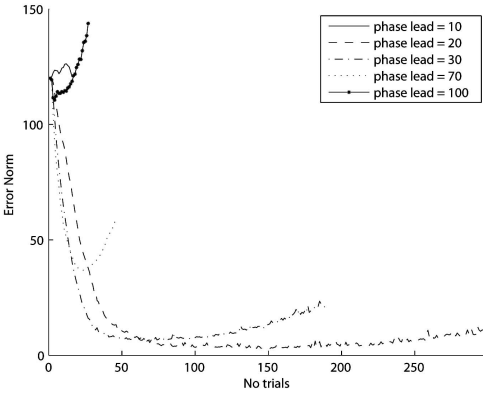
$$\max_i |\lambda_i(I - GLS)| < 1 \tag{6.36}$$

Phase-lead ILC has been applied to the MIMO facility with varying levels of interaction. The same reference signals as shown in Figure 6.1 have been used. After many tests, the best combination of gains comprised γ_1 and γ_2 as both 0.03 and δ as 10, 20, 30, 70 and 100. Results for the low interaction case are shown in Figure 6.4(a). Time-delays of 100, 70 and 10 give rise to error norm divergence after around 10 trials. The only values of time-delay that produce satisfactory results are 20 and 30. For a value of 20, the error norm converges dramatically and reaches a minimum value of 10 after about 50 trials. However after that the error norm diverges and the system needs to be turned off to protect mechanical parts since the control signal is too high. The delay time, $\delta = 30$, yields the greatest performance, providing a similar convergence rate but keeping the error norm at a reasonably low value over 300 trials. However it tends to give rise to divergence if the system is run over a large number of trials. The same observation holds for the extremely low interaction, low interaction, medium interaction and high interaction cases which are shown in Figure 6.4(b)-6.4(e) respectively. However, the result is different in the highest interaction case. All the delay time values converge after a few trials except $\delta = 10$, which diverges in the first few trials. The delay-time constant $\delta = 30$ still has the best result but the system must be shut down after 60 trials due to the high control input norm. In this case, the results are far less satisfactory than other interaction levels in terms of both the convergence rate and the error norm value.

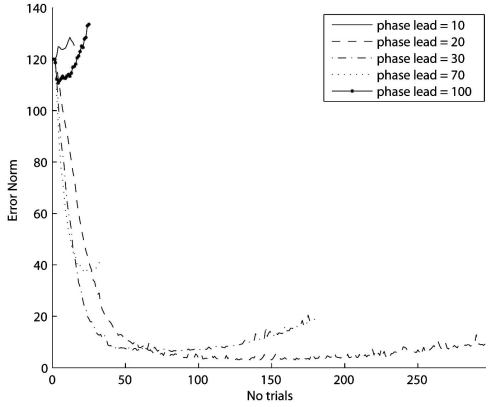
Generally, the phase-lead method yields better results compared to the other two controllers. Like them, it does not need any model within the control update algorithm. The output can satisfactorily track references in the first few trials but still diverges after a set number of trials depending on the tuning delay factor and the gain matrix γ. In general the method only works well with systems that behave like a pure time delay (Cai et al., 2007), (Cai et al., 2008b). Therefore for the MIMO facility, phase-lead ILC cannot be expected to give high performance. The MIMO facility has integrator hence models always exist uncertainty that increase when the interaction increases. Additionally, the



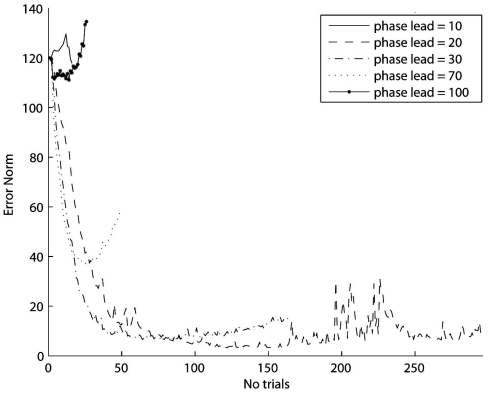
(a) Phase-lead type ILC with zero interaction



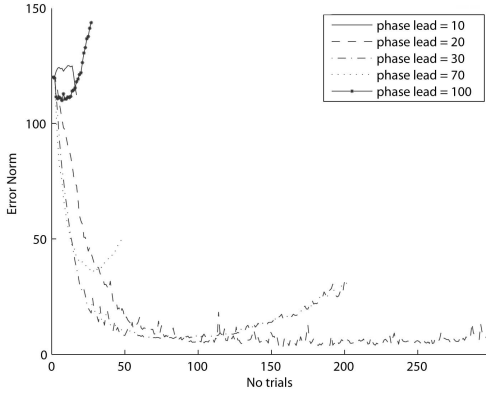
(b) Phase-lead type ILC with extremely low interaction



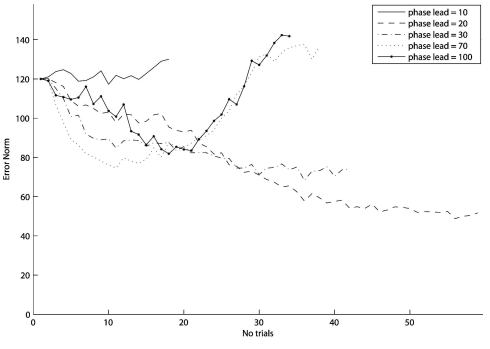
(c) Phase-lead type ILC with low interaction



(d) Phase-lead type ILC with medium interaction



(e) Phase-lead type ILC with high interaction



(f) Phase-lead type ILC with extremely high interaction

Figure 6.4: Phase-lead ILC with gain = 0.03.

heuristic tunning method is also time consuming, and the operator cannot be sure they have found the optimal parameters.

6.6 Summary

Three simple structure controllers for ILC have been presented in discrete-time and applied to the 2×2 MIMO facility. These controllers require a tuning gain matrix γ and in one case a delay-time constant, and do not require explicit model. This aids usability but necessarily degrades performance. Additionally, the heuristic tuning requires significant experiments to arrive at suitable values and hence is extremely time consuming. Consequently practical constraints mean it cannot yield impressive performance for a system with multiple inputs and outputs.

Basic ILC controllers also provide an overview of baseline performance achievable for the MIMO facility and reveal the effect of interaction levels on the system performance. Interaction is seen to cause increasing controller norm and error norm. Therefore it is frequently necessary to engage safety devices to protect the mechanical system.

In order to achieve better results and save time in finding a good combination of tunning parameters, a more advanced controller is required. The controller should utilise all information from the MIMO facility to aid the control update. This will help improve the response time to any reference signal and yield faster convergence rates. The next chapter will describe more advanced controllers which aim to improve the baseline results discussed in this chapter, using the same test conditions.

Chapter 7

Gradient ILC

A well established algorithmic framework has been developed, coupling analysis with practical performance demands, and this is especially true for linear optimal algorithms whose properties have been extensively studied Bristow (2008b); Ratcliffe et al. (2005); Donkers et al. (2008); Wang et al. (2010); Barton et al. (2008); Mishra and Tomizuka (2005); Davies et al. (2008); Butcher et al. (2008b). Gradient-based ILC algorithms have received significant attention in the literature due to their attractive theoretical properties.

Unlike basic ILC controllers, gradient-based ILC algorithms require a system model in the control update. In this chapter a popular type of ILC update termed gradient ILC is analysed in order to understand the performance characteristics for multivariable systems, robustness, convergence and control effort properties under a variety of conditions.

Whilst the role of interaction terms is well understood in multivariable control, it is not well explored in ILC, in either theoretical or practical domains. In order to tackle this, in this chapter, the effect on performance of off-diagonal sub-systems $G_{ij}(z)$, $i \neq j$, is explicitly considered.

7.1 Gradient ILC with variable optimal β

Gradient ILC has appeared in (Furuta and Yamakita, 1987), (Hätönen et al., 2004), (Owens et al., 2007), (Kinosita et al., 2002) for SISO systems and is now derived for MIMO systems.

7.1.1 Algorithm description

Consider again the objective of ILC, with $\mathbf{u}_k$ and $\mathbf{y}_k$ the input and output vectors respectively on the k^{th} trial, and the goal being to find a sequence of control inputs

satisfying

$$\lim_{k\to\infty} \|\mathbf{e}_k\| = 0, \qquad \lim_{k\to\infty} \|\mathbf{u}_k - \mathbf{u}_d\| = 0 \tag{7.1}$$

where $\mathbf{e}_k = \mathbf{y}_d - \mathbf{y}_k$ is the tracking error, and $\mathbf{u}_d$ is the unknown desired input sequence corresponding to reference signal $\mathbf{y}_d$. The vectors $\mathbf{u}_k, \mathbf{y}_d$ and $\mathbf{y}_k$ are given in (6.17 - 6.19) respectively. Over the k^{th} trial the relationship between input and output time-series can be expressed by $\mathbf{y}_k = G\mathbf{u}_k$.

To achieve this, on the $(k+1)^{th}$ trial gradient ILC minimises a cost function of the form

$$J(\mathbf{u}_{k+1}) = \left\{ [\mathbf{y}_d - \mathbf{y}_{k+1}]^T Q [\mathbf{y}_d - \mathbf{y}_{k+1}] + [\mathbf{u}_{k+1} - \mathbf{u}_k]^T R [\mathbf{u}_{k+1} - \mathbf{u}_k] \right\} \tag{7.2}$$

where

$$Q = \text{diag}\{Q(0), Q(1), \ldots, Q(N-1)\}, \qquad R = \text{diag}\{R(0), R(1), \ldots, R(N-1)\} \tag{7.3}$$

in which weighting matrices $Q(i)$ are $R(i)$ are symmetric and positive semi-definite for all i. Notational simplification occurs if the input and output spaces $\mathbf{U}$ and $\mathbf{Y}$ respectively, are adopted with inner products

$$\langle \mathbf{u}_1, \mathbf{u}_2 \rangle_{\mathbb{U}} = \mathbf{u}_1^T R \mathbf{u}_2 = \sum_{i=0}^{N-1} \mathbf{u}_1(i)^T R(i) \mathbf{u}_2(i) \tag{7.4}$$

$$\langle \mathbf{y}_1, \mathbf{y}_2 \rangle_{\mathbb{Y}} = \mathbf{y}_1^T Q \mathbf{y}_2 = \sum_{i=0}^{N-1} \mathbf{y}_1(i)^T Q(i) \mathbf{y}_2(i). \tag{7.5}$$

Using these (7.2) becomes

$$J(\mathbf{u}_{k+1}) = ||\mathbf{e}_{k+1}||^2_{\mathbb{Y}} + ||\mathbf{u}_{k+1} - \mathbf{u}_k||^2_{\mathbb{U}}. \tag{7.6}$$

The gradient optimisation method may be applied to minimise the cost function (7.6) by setting the required change in input proportional to the gradient of the error with respect to the input, to give the ILC update

$$\mathbf{u}_{k+1} = \mathbf{u}_k + \beta R^{-1} G^T Q \mathbf{e}_k \tag{7.7}$$

$$= \mathbf{u}_k + \beta G^* \mathbf{e}_k \tag{7.8}$$

since $R^{-1}G^TQ$ is equivalent to the adjoint operator G^* of G with respect to the weighted inner product equations (7.4) and (7.5). Substitution into (7.6) this gives

$$J(\mathbf{u}_{k+1}) = \| (I - \beta G G^*) \mathbf{e}_k \|^2_{\mathbb{Y}} + \| \beta G^* \mathbf{e}_k \|^2_{\mathbb{U}} \tag{7.9}$$

which is minimised with respect to the positive scalar β by setting β equal to

$$\hat{\beta} = \frac{\mathbf{e}_k^T QGG^*\mathbf{e}_k}{(GR^{-1}G^TQ\mathbf{e}_k)^TQ(GR^{-1}G^TQ\mathbf{e}_k)+(R^{-1}G^TQ\mathbf{e}_k)^TR(R^{-1}G^TQ\mathbf{e}_k)} \tag{7.10}$$

$$= \frac{\mathbf{e}_k^T QGG^*\mathbf{e}_k}{\| GG^*\mathbf{e}_k \|_{\mathbb{Y}}^2 + \| G^*\mathbf{e}_k \|_{\mathbb{U}}^2} \tag{7.11}$$

For simplicity now suppose $Q = qI$ and $R = I$. As q is increased from zero to ∞, the optimum scalar multiplier $\beta q = \hat{\beta}(q)q$ in (7.8) varies continuously and monotonically from zero to a maximum of

$$\frac{\| G^T\mathbf{e}_k \|^2}{\| GG^T\mathbf{e}_k \|^2} \tag{7.12}$$

which can be approximated by a static value of $\frac{1}{\|GG^T\|}$. Consider some initial arbitrary choice of $Q = qI$, $0 < q < \infty$, in update (7.8). Then within the approximate range $\left(0, \quad \frac{1}{q\|GG^T\|}\right)$ any scalar multiplier β may be selected in the update (7.8) such that the optimisation problem (7.2) is then solved for some particular choice of Q and R weighting matrices. This means β can be regarded as a tuning parameter within gradient ILC update (7.8).

Remark 7.1. It is shown in Butcher et al. (2008a) that the gradient algorithm can be implemented without an explicit model thus providing significant utility. Instead an additional experiment between each trial provides the term $G^*\mathbf{e_k}$.

7.1.2 Convergence analysis

From (7.8) the error evolution is given by

$$\mathbf{e}_{k+1} = (I - \beta GG^*)\mathbf{e}_k. \tag{7.13}$$

A necessary and sufficient condition for convergence is

$$\rho(I - \beta GG^*) < 1. \tag{7.14}$$

Since $\sigma_i(GG^*) \geq 0$ this is equivalent to

$$|1 - \beta\sigma_i(GG^*)| < 1, \qquad \forall i \tag{7.15}$$

so that $0 < \beta\overline{\sigma}(GG^*) < 2$ so $\beta < \frac{2}{\|GG^*\|}$. Hence convergence to zero error is guaranteed if G has full rank, and either the trial-dependent gain (7.11) is employed with any choice of symmetric positive definite Q and R, or a fixed gain satisfying

$$0 < \beta < \frac{2}{q \| GG^T \|} \tag{7.16}$$

is used together with $R = I$ and any choice of $Q = qI$, $0 < q < \infty$. Let V be the eigenvector matrix of GG^*, with Λ a diagonal matrix of corresponding eigenvalues $\big(\sigma_i(GG^*)\big)^2$. Then (7.13) can be written as

$$\mathbf{e}_{k+1} = V(1 - \beta\Lambda)V^{-1}\mathbf{e}_k \tag{7.17}$$

so that

$$V^{-1}\mathbf{e}_k = (1 - \beta\Lambda)^k V^{-1}\mathbf{e}_0. \tag{7.18}$$

Since $V^{-1} = V^T$, the component of $\mathbf{e}_0$ projected onto the i^{th} eigenvector of GG^* hence evolves as $\Big(1-\beta\big(\sigma_i(GG^*)\big)^2\Big)^k$. If $\underline{\sigma}^2$ is the minimum eigenvalue then the reference with slowest convergence rate is a scalar multiple of its corresponding eigenvector. Similarly if $\overline{\sigma}^2$ is the maximum eigenvalue then the reference with the fastest convergence rate is a scalar multiple of its corresponding eigenvector. For an arbitrary reference and gain satisfying (7.16), the error norm sequence lies in the interval

$$\Big(1 - \beta\big(\overline{\sigma}(GG^*)\big)^2\Big)^k \leq \frac{\|\mathbf{e}_k\|_{\mathbb{Y}}}{\|\mathbf{e}_0\|_{\mathbb{Y}}} \leq \Big(1 - \beta\big(\underline{\sigma}(GG^*)\big)^2\Big)^k \tag{7.19}$$

For a given reference an approximation of the convergence rate is the weighted sum

$$\frac{\| V^T\mathbf{y}_d \|_p}{\| (I - \beta\Lambda)V^T\mathbf{y}_d \|_p} \tag{7.20}$$

where $\| \cdot \|_p$ denotes the p-norm.

7.1.3 Control effort

In practice the input norm required to enforce tracking plays an important role in determining robustness, due to actuator saturation and the range over which the plant must be assumed linear. The input norms as $k \to \infty$ are hence derived. Assuming the condition (7.14) holds so the algorithm is convergent, a application of the ILC update yields

$$\begin{aligned}
\mathbf{u}_{k+1} &= \mathbf{u}_k + \beta G^*(\mathbf{y}_d - G\mathbf{u}_k) && (7.21)\\
&= (I - \beta G^*G)\mathbf{u}_k + \beta G^*\mathbf{y}_d && (7.22)\\
&= (I - \beta G^*G^{k+1})\mathbf{u}_0 + \Bigg(\sum_{i=0}^{k}(I - \beta G^*G)^i\Bigg)(\beta G^*)\mathbf{y}_d && (7.23)\\
\Rightarrow \mathbf{u}_\infty &= \big(I - (I - \beta G^*G)\big)^{-1}\beta G^*\mathbf{y}_d && (7.24)\\
&= G^\dagger\mathbf{y}_d. && (7.25)
\end{aligned}$$

where $(\cdot)^\dagger$ denotes the pseudo-inverse. This input sequence hence satisfies the bound

$$\|\mathbf{u}_\infty\| = \left\|G^\dagger \mathbf{y}_d\right\| \leq \frac{\|\mathbf{y}_d\|_{\overline{\mathbb{Y}}}}{\underline{\sigma}(GG^*)} \tag{7.26}$$

and hence is governed by $\underline{\sigma}(GG^*)$. The correlation between interaction and $\underline{\sigma}(GG^T)$ means that the bound on control effort typically increases with higher interaction.

7.1.4 Robustness analysis

Assume that the true plant is now $\tilde{G}$ and includes multiplicative modeling uncertainty described by the equation

$$\tilde{G} = UG \tag{7.27}$$

where G is the nominal plant model used in the update law (7.8).

Theorem 1. Suppose U is a Hermitian matrix. Then the weighting selection $R = rI$, $Q = qI$, with

$$0 < \beta < \frac{2}{\max_i \left\{0, \lambda_i(U)\right\}} \frac{1}{\overline{\sigma}^2(GG^*)} \tag{7.28}$$

guarantees convergence of the true plant to zero tracking error using gradient ILC update law (7.8).

Proof. The error is given by

$$\begin{aligned} \mathbf{e}_{k+1} &= \mathbf{y}_d - \tilde{G}\mathbf{u}_{k+1} && (7.29)\\ &= \mathbf{y}_d - UG(\mathbf{u}_k + \beta G^* \mathbf{e}_k) && (7.30)\\ &= (I - \beta UGG^*)\mathbf{e}_k && (7.31) \end{aligned}$$

since $\lambda_i(GG^*) = \sigma_i^2(GG^*) > 0$

$$0 < \underline{\sigma}^2(GG^*) \leq \lambda_i(GG^*) \leq \overline{\sigma}_i^2(GG^*) < 1 \tag{7.32}$$

A necessary and sufficient condition for convergence to an arbitrary reference is

$$\begin{aligned} \max_i |\lambda_i(I - \beta UGG^*)| &< 1 && (7.33)\\ \Rightarrow \max_i |1 - \beta\lambda_i(UGG^*)| &< 1 && (7.34) \end{aligned}$$

which can be written as

$$\overline{\sigma}^2(I - UGG^*) = \max_i |1 - \lambda_i(UGG^*)| < 1, \ \forall i \tag{7.35}$$

The assumption that U is Hermitian guarantees that $\lambda_i(UGG^*)$ are real and satisfy

$$0 < \underline{\sigma}^2(GG^*) \min_j \lambda_j(U) \leq \lambda_i(UGG^*) \leq \overline{\sigma}^2(GG^*) \max_j \lambda_j(U) < \max_j \lambda_j(U) \tag{7.36}$$

It can hence be shown that a sufficient condition is

$$\lambda_i(U) < \frac{2}{\beta \overline{\sigma}^2(GG^*)}, \quad \forall i \tag{7.37}$$

Hence a sufficient condition for convergence to zero error is

$$0 < \beta < \frac{2}{\max_i\{0, \lambda_i(U)\}} \cdot \frac{1}{\overline{\sigma}^2(GG^*)}, \quad \forall i. \tag{7.38}$$

So for any known uncertainty as defined in (7.27) exist an optimal β results in convergence to zero tracking error □

Remark 7.2. For any uncertainty U, there exists a β which satisfies condition (7.28).

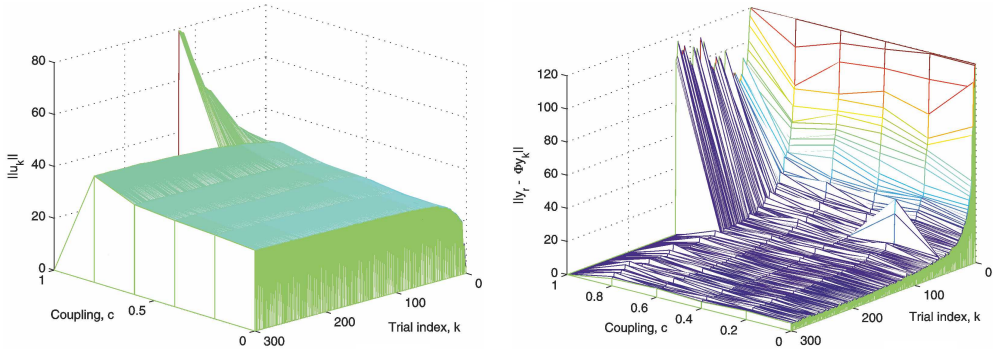
Remark 7.3. As discussed in Section 6.2, increasing interaction causes $\overline{\sigma}(GG^*)$ to increase, which hence leads to a reduction in robustness.

Remark 7.4. With no uncertainty, $U = I$ and (7.28) = (7.16).

7.1.5 Experimental results

In this section gradient ILC is applied to the MIMO testbed facility in order to investigate the performance degradation caused by MIMO interaction. Sinusoidal references are used with magnitudes of 7 and 5 for reference 1 and reference 2 respectively, as displayed in Figure 6.1. Both references have a total length $T = 2.4$ seconds. The reference is exactly the same as the version used in the previous chapter so that the performance of this method can be compared fairly.

The gradient ILC update (7.8) has been applied to track the reference and an optimal β is calculated using (7.11). After applying a wide range of weighting matrix values Q, R the best performance has been found to correspond to $R = I$ and $Q = 0.1I$ for the highest interaction case and $Q = 0.5I$ for the remaining interaction cases. Using the highest interaction case the controller norm becomes excessive and the system has been switched off to protect the facility after 109 trials. This is due to model uncertainty that is amplified by the interaction effect and leads to degradation in the controller's performance. The controller norm result for this case is shown in Figure 7.1(a). For the remaining interaction cases, controller norms reach steady values which correspond to accurate tracking. The corresponding errors norm are shown in Figure 7.1(b), and after 100 trials the zero interaction case attains an error norm of approximately 4 while the highest interaction case attains a value of 90 and then diverges. For the coupling level



(a) Controller norm.

(b) Error norm.

Figure 7.1: Standard gradient ILC

$c = 0.2$ there is a spike at trial number 70 due to an unpredictable disturbance generated within the differential gear but the controller quickly recovers accurate tracking.

Compared to basic ILC controllers gradient ILC provides a significant improvement in tracking performance. The controller keeps the error norm at a low value and maintains it over the test duration. The results do not show any sign of divergence for the zero interaction case to the high interaction case. The only case that gradient ILC fails to control adequately is the highest interaction level, which is associated with much uncertainty in the model. The performance could be improved if a more accurate model could be derived but this is highly time consuming.

The experimental results also confirm observations in Section 7.1.3 and 7.1.4 since increasing the interaction level leads to increased controller effort and reduced robustness. Therefore for the highest interaction level the error norm cannot converge to very low value.

7.2 Summary

The gradient ILC method has been derived for multivariable systems. The properties of the method have been discussed with regard to control effort, convergence rate and robustness. To investigate the theory, experimental results using the MIMO facility have been provided. The results support theoretical conclusions that interaction plays a significant role in influencing the error and the controller effort. If the interaction level is sufficient large it can make the controller become unstable. Consequently, an alternative controller is required to improve the tracking performance of the system. In the next chapter another advanced controller is hence introduced and analysed.

Chapter 8

Norm optimal iterative learning control

Along with gradient ILC, another prominent member of the class of linear optimal ILC algorithms is norm optimal ILC (NOILC) which selects the next control input to minimise a cost function involving the predicted error and the change in successive control inputs (Bristow et al., 2006; Ahn et al., 2007). NOILC can be implemented using a purely feed-forward structure (Buchheit et al., 1994), or through combination of state feedback and predictive feed-forward action (Amann et al., 1996a,b). The framework has been applied to a range of systems including gantry robots (Ratcliffe et al., 2006), industrial robotic systems (Barton and Alleyne, 2011), rehabilitation platforms (Rogers et al., 2010), free electron lasers (Rogers et al., 2010) and pneumatic muscle actuators (Schindele and Aschemann, 2011). Extensions have been proposed using a predictive mechanism (Bristow and Alleyne, 2003), constraints (Chu and Owens, 2010) and projections (Chu and Owens, 2009).

This chapter uses the NOILC framework to investigate the role of MIMO interaction on ILC performance, including convergence speed, control effort, and robustness to modeling uncertainty. In particular, a rigid connection between increased interaction and reduced robustness bounds is derived. Two implementations of NOILC are experimentally assessed on the MIMO test facility over a wide range of interaction and noise conditions, and results are compared with theoretical predictions.

8.1 Algorithm description

As previously described, the objective of ILC is to use experimental data over repeated trials to generate a signal $\mathbf{u}_k$ such that

$$\lim_{k\to\infty} ||\mathbf{e}_k|| = 0, \qquad \lim_{k\to\infty} ||\mathbf{u}_k - \mathbf{u}_d|| = 0 \tag{8.1}$$

where $\mathbf{e}_k = \mathbf{y}_d - \mathbf{u}_k$ is the system tracking error on trial k. NOILC achieves this objective by calculating the control input supervector, $\mathbf{u}_{k+1}$, for application to the plant (6.8) on the $k+1^{th}$ trial, such that a quadratic cost function of the form

$$J(\mathbf{u}_{k+1}) = \left\{ [\mathbf{y}_d - \mathbf{y}_{k+1}]^T Q [\mathbf{y}_d - \mathbf{y}_{k+1}] + [\mathbf{u}_{k+1} - \mathbf{u}_k]^T R [\mathbf{u}_{k+1} - \mathbf{u}_k] \right\} \tag{8.2}$$

is minimised, hence balancing tracking error reduction against change in successive control inputs. Here $\mathbf{y}_{k+1} = G\mathbf{u}_{k+1}$ is the predicted output on the $k+1^{th}$ trial, and

$$Q = \text{diag}\{Q(0), Q(1), \ldots, Q(N-1)\}, \qquad R = \text{diag}\{R(0), R(1), \ldots, R(N-1)\} \tag{8.3}$$

in which weighting matrices $Q(i)$ are $R(i)$ are symmetric and positive semi-definite for all i. Notational simplification occurs if the input and output spaces $\mathbb{U}$ and $\mathbb{Y}$ respectively, are adopted with inner products

$$\langle \mathbf{u}_1, \mathbf{u}_2 \rangle_{\mathbb{U}} = \mathbf{u}_1^T R \mathbf{u}_2 = \sum_{i=0}^{N-1} \mathbf{u}_1(i)^T R(i) \mathbf{u}_2(i) \tag{8.4}$$

$$\langle \mathbf{y}_1, \mathbf{y}_2 \rangle_{\mathbb{Y}} = \mathbf{y}_1^T Q \mathbf{y}_2 = \sum_{i=0}^{N-1} \mathbf{y}_1(i)^T Q(i) \mathbf{y}_2(i). \tag{8.5}$$

which allows (8.2) to be expressed as

$$J(\mathbf{u}_{k+1}) = ||\mathbf{e}_{k+1}||^2_{\mathbb{Y}} + ||\mathbf{u}_{k+1} - \mathbf{u}_k||^2_{\mathbb{U}}. \tag{8.6}$$

where $\mathbf{e}_{k+1} = \mathbf{y}_d - \mathbf{y}_{k+1}$. Hence the NOILC control input is the trial

$$\mathbf{u}_{k+1} = \arg\min_{\mathbf{u}_{k+1}} J(\mathbf{u}_{k+1}) \tag{8.7}$$

with the weights R and Q employed to set the balance between convergence and trial-to-trial input change. Full details appear in, for example, Amann et al. (1996a). The problem (8.7) can be directly solved and the resulting $\mathbf{u}_{k+1}$ applied to the plant in a feed-forward implementation, or the same solution can be expressed in terms of the plant states, leading to a combined feed-forward and state feedback realisation. Both options are now summarised, and are compared experimentally.

8.1.1 NOILC feed-forward implementation

The unique solution to (8.7) can be directly realised in the purely feed-forward structure given by

$$\begin{aligned} \mathbf{u}_{k+1} &= \mathbf{u}_k + R^{-1}G^TQ\mathbf{e}_{k+1} && (8.8)\\ &= \mathbf{u}_k + R^{-1}G^TQ(I + GR^{-1}G^TQ)^{-1}\mathbf{e}_k \\ &= \mathbf{u}_k + G^*(I + GG^*)^{-1}\mathbf{e}_k. && (8.9) \end{aligned}$$

in which $G^* = R^{-1}G^TQ$ is the plant adjoint operator.

8.1.2 NOILC feed-forward and state feedback implementation

To embed the possibility of increased robustness to noise and modeling uncertainty, the solution to (8.7) can also be realised using combined feed-forward and current trial state feedback action Amann et al. (1996a,b). In particular, the non-causal solution (8.8) is equivalent to the co-state system

$$\begin{aligned} \psi_{k+1}(i) &= A^T\psi_{k+1}(i+1) + C^TQ(i+1)\mathbf{e}_{k+1}(i+1) && (8.10)\\ \mathbf{u}_{k+1}(i) &= \mathbf{u}_k(i) + R^{-1}(i)B^T\psi_{k+1}(i) && (8.11) \end{aligned}$$

for $i = 0, 1, \ldots N-1$ with terminal condition $\psi_{k+1}(N) = 0$. Causal implementation is achieved by assuming full state knowledge and writing the co-state in the form

$$\psi_{k+1}(i) = -K(i)(I + BR^{-1}(i)B^TK(i))^{-1}A[x_{k+1}(i) - x_k(i)] + \xi_{k+1}(i). \qquad (8.12)$$

which combines state feedback, with an additional feed-forward, predictive term $\xi_{k+1}(i)$. Equating representations, it can be shown that the matrix gain $K(i)$ is the solution of the discrete-time matrix Riccati equation on the interval $i = 0, 1, \ldots N-1$, that is

$$\begin{aligned} K(i) = A^TK(i+1)A + C^TQ(i+1)C - \\ A^TK(i+1)B\big(B^TK(i+1)B + R(i+1)\big)^{-1}B^TK(i+1)A \end{aligned} \qquad (8.13)$$

with the terminal condition $K(N) = 0$. This equation does not depend on experimental data and is calculated only once before the test begins. The predictive term $\xi_{k+1}(i)$ is given by

$$\xi_{k+1}(i) = (I + K(i)BR^{-1}(i)B^T)^{-1}(A^T\xi_{k+1}(i+1) + C^TQ(i)\mathbf{e}_k(i)) \qquad (8.14)$$

with the terminal condition $\xi_{k+1}(N) = 0$, and must be calculated after each trial is performed. Hence inserting (8.12) in (8.11) and rearranging, the control input on sample

i is then given by

$$\mathbf{u}_{k+1}(i) = \mathbf{u}_k(i) - (B^T K(i) B + R(i))^{-1} B^T K(i) A[\mathbf{x}_{k+1}(i) - \mathbf{x}_k(i)] + R^{-1}(i) B^T \xi_{k+1}(i) \quad (8.15)$$

8.2 Convergence analysis

From (8.9) the trial-to-trial error evolution is governed by

$$\mathbf{e}_{k+1} = (I + GG^*)^{-1} \mathbf{e}_k. \quad (8.16)$$

and convergence to zero tracking error is guaranteed with any choice of symmetric positive definite Q and R since G has full rank. Let V and Λ be matrices containing the eigenvectors and corresponding eigenvalues, $\sigma_i^2(GG^*)$, of GG^*. Then (8.16) can be written in the form

$$\mathbf{e}_{k+1} = V(I + \Lambda)^{-1} V^{-1} \mathbf{e}_k \quad (8.17)$$

and repeated action results in

$$V^{-1} \mathbf{e}_k = (I + \Lambda)^{-k} V^{-1} \mathbf{e}_0 \quad (8.18)$$

The component of $\mathbf{e}_0$ projected onto the i^{th} eigenvector of GG^* hence evolves as $(1 + \sigma_i^2(GG^*))^{-k}$. Taking norms, tracking error on the k^{th} trial is bounded by

$$\left(1 + \overline{\sigma}^2(GG^*)\right)^{-k} \leq \frac{\|\mathbf{e}_k\|_{\mathbb{Y}}}{\|\mathbf{e}_0\|_{\mathbb{Y}}} \leq \left(1 + \underline{\sigma}^2(GG^*)\right)^{-k}. \quad (8.19)$$

The fastest convergence rate is hence dictated by $\overline{\sigma}^2(GG^*)$. Assuming for simplicity the form $R = rI$, $Q = qI$, this increases with the level of interaction since

$$\overline{\sigma}(GG^*) = \overline{\sigma}(GqG^T r^{-1}) = \frac{q}{r} \overline{\sigma}(GG^T) \quad (8.20)$$

The slowest convergence is governed by the minimum singular value $\underline{\sigma}(GG^*) = \frac{q}{r} \underline{\sigma}(GG^T)$. Whilst not directly linked to MIMO interaction, there is often a strong correlation between increased interaction and a reduction in $\underline{\sigma}(GG^T)$, which hence reduces the convergence rate of the slowest mode. The values of of $\underline{\sigma}(GG^T)$ for each interaction case are calculated in Table 9.2. It shows values close to zero but due to the model uncertainty the trend cannot be shown clearly.

8.3 Control effort

In practice the input norm required to enforce tracking plays an important role in determining robustness, due to actuator saturation and the range over which the plant may be assumed linear. The limiting input as $k \to \infty$ is found through repeated application of (8.9), which yields

$$\begin{aligned} \mathbf{u}_{k+1} &= \left(I - G^*(I+GG^*)^{-1}G\right)^{k+1}\mathbf{u}_0 + \\ &\quad \left(\sum_{i=0}^{k}(I - G^*(I+GG^*)^{-1}G)^i\right) G^*(I+GG^*)^{-1}\mathbf{y}_d \qquad (8.21) \\ \Rightarrow \mathbf{u}_\infty &= \left(G^*(I+GG^*)^{-1}G\right)^{-1}G^*(I+GG^*)^{-1}\mathbf{y}_d \qquad (8.22) \\ &= G^\dagger \mathbf{y}_d \qquad (8.23) \end{aligned}$$

where $(\cdot)^\dagger$ denotes the pseudo-inverse. This input sequence hence satisfies the bound

$$\|\mathbf{u}_\infty\| = \left\|G^\dagger \mathbf{y}_d\right\| \leq \frac{\|\mathbf{y}_d\|_{\mathbb{Y}}}{\underline{\sigma}(GG^*)} \qquad (8.24)$$

and hence is governed by $\underline{\sigma}(GG^*)$. The correlation between interaction and $\underline{\sigma}(GG^T)$ means that the bound on control effort typically increases with higher interaction. Values of $\underline{\sigma}(GG^T)$ are given for the MIMO test facility in Table 9.2.

8.4 Robustness analysis

Assume now that the true plant, $\tilde{G}$, includes multiplicative modeling uncertainty described by the equation

$$\tilde{G} = UG \qquad (8.25)$$

Here G is the nominal plant model used in the update law (8.9) or (8.13)-(8.15) to generate control input $\mathbf{u}_{k+1}$, whereas $\tilde{G}\mathbf{u}_{k+1}$ gives the output when subsequently applied in practice to the true plant (producing the experimental tracking error $\mathbf{e}_{k+1} = \mathbf{y}_d - \tilde{G}\mathbf{u}_{k+1}$).

Theorem 2. Suppose that U is a Hermitian matrix. Then the weighting selection $R = rI$, $Q = qI$, with

$$0 < \frac{q}{r} < \frac{2}{\max_i \{0, \lambda_i(U) - 2\}} \cdot \frac{1}{\overline{\sigma}^2(GG^T)} \qquad (8.26)$$

guarantees convergence of the true plant to zero tracking error using the feed-forward NOILC law (8.9).

Proof. Employing the real plant (8.25) in the NOILC update (8.9) gives rise to

$$\mathbf{e}_{k+1} = (I - UGG^*(I+GG^*)^{-1})\mathbf{e}_k \qquad (8.27)$$

Let the Hermitian matrix

$$W = GG^*(I + GG^*)^{-1} \tag{8.28}$$

so that

$$\mathbf{e}_{k+1} = (I - UW)\mathbf{e}_k \tag{8.29}$$

The eigenvalues of W are all real and satisfy

$$\lambda_i(W) = 1 - (1 + \lambda_i(GG^*))^{-1}, \qquad i = 1, \cdots N \tag{8.30}$$

and since $\lambda_i(GG^*) = \sigma_i^2(GG^*) > 0$

$$0 < \frac{\underline{\sigma}^2(GG^*)}{1 + \underline{\sigma}^2(GG^*)} \leq \lambda_i(W) \leq \frac{\overline{\sigma}^2(GG^*)}{1 + \overline{\sigma}^2(GG^*)} < 1 \tag{8.31}$$

A necessary and sufficient condition for the output to converge to an arbitrary reference is

$$\rho(I - UW) = \overline{\sigma}^2(I - UW) = \max_i |1 - \lambda_i(UW)| < 1 \quad \forall i \tag{8.32}$$

The assumption that U is Hermitian guarantees that $\lambda_i(UW)$ are real and satisfy

$$0 < \underline{\sigma}^2(W) \min_j \lambda_j(U) \leq \lambda_i(UW) \leq \overline{\sigma}^2(W) \max_j \lambda_j(U) < \max_j \lambda_j(U) \tag{8.33}$$

and using (8.28) a sufficient condition for convergence to zero error is

$$\lambda_i(U) < \frac{2}{\overline{\sigma}^2(W)}, \quad \forall i \tag{8.34}$$

The choice of weights $R = rI$ and $Q = qI$, $r, q > 0$, on the right-hand side of (8.34) gives

$$\frac{2(1 + \overline{\sigma}^2(GG^*))}{\overline{\sigma}^2(GG^*)} = \frac{2(1 + \frac{q}{r}\overline{\sigma}^2(GG^T))}{\frac{q}{r}\overline{\sigma}^2(GG^T)} = \frac{2(\frac{r}{q} + \overline{\sigma}^2(GG^T))}{\overline{\sigma}^2(GG^T)}. \tag{8.35}$$

This leads to the condition

$$\lambda_i(U) \quad < \quad \frac{2}{\frac{q}{r}\overline{\sigma}^2(GG^T)} + 2, \quad \forall i \tag{8.36}$$

So that for a known uncertainty, any weight such that (8.26) holds will result in convergence to zero tracking error. □

Remark 8.1. For the case of no model uncertainty, $U = I$, and all weighting values provide convergence to zero error. The bound on the admissible modeling uncertainty (8.36) strictly increases as $\frac{q}{r}$ reduces.

Remark 8.2. It is shown in Hätönen (2004) that U can be realised as a positive definite transfer-function $U(z)$. Therefore a sufficient condition for monotonic convergence to zero error is that $\arg\{U(e^{j\omega})\}$ lies in the open interval $(-\pi/2, \pi/2)$ $\forall$ i, with a gain chosen to satisfy (8.26).

From Theorem 2, robustness to multiplicative model uncertainty reduces as $\overline{\sigma}^2(GG^T)$ increases. From Lemma 6.1 this is also strongly correlated with increased interaction between input and output pairs, and hence the robustness bound (8.26) decreases as the level of interaction increases.

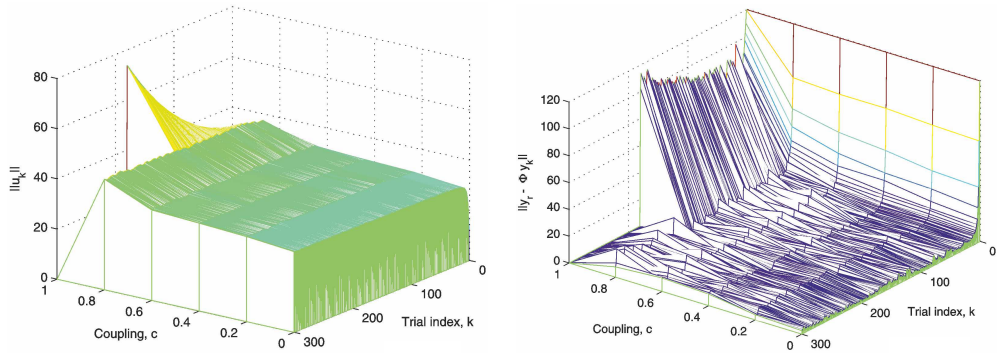
This section has illustrated that achievable robustness and performance bounds using NOILC typically degrade as MIMO interaction increases. The next section presents results when NOILC is applied to the MIMO test facility with varied levels of interaction.

8.5 Experimental results

In this section both implementations of NOILC algorithms are applied to the MIMO test-bed facility. Sinusoidal references are used with magnitudes of 7 and 5 for reference 1 and reference 2 respectively, and are shown in Figure 6.1. Both references have a total length $T = 2.4$ seconds. The transfer-function model for each interaction case was derived and validated in Chapter 5 and are the same as were used previously.

8.5.1 Feedforward NOILC

Experiments were undertaken using the NOILC update (8.9). Based on a large number of tests, the best performance has been achieved using $R = I$ and $Q = 0.05I$ for all interaction cases except the highest interaction level ($c = 1$), which uses $Q = 0.005I$. The error norm results in Figure 8.1(b) show rapid convergence rates for all coupling



(a) Controller norm.

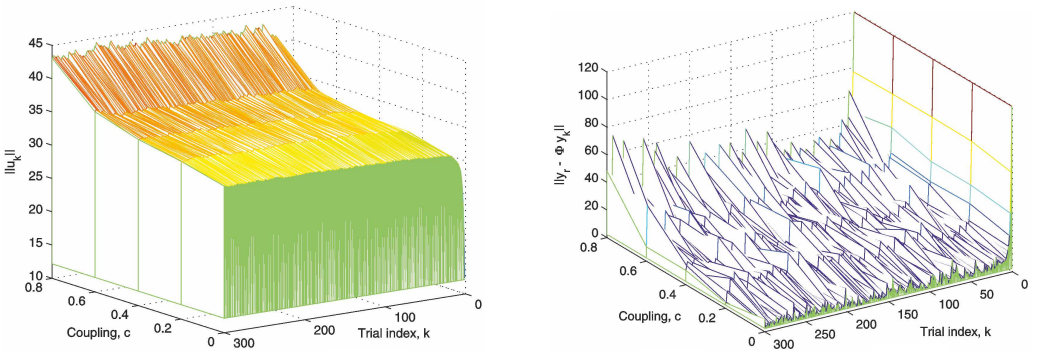
(b) Error norm.

Figure 8.1: Feedforward norm optimal ILC

values except the highest interaction case that requires a smaller value of Q to delay the onset of instability, which always occurs. In the case of $c = 0$, after 60 trials, the error norm almost attains its final value, which is around 1. However the final error norm increases with the level of interaction, as is most obvious in the cases of $c = 0.8, 1$. There is a significant difference in the error norm of the highest interaction case and the other

interaction levels. The controller cannot cope with the high model uncertainty magnified by the effect of substantial interaction and therefore the system diverges after around 180 trials. The corresponding controller norm is extremely high and still has not reached a steady value after 180 trials. The controller norm results are shown in Figure 8.1(a). Compared to gradient ILC, feed-forward NOILC displays superior convergence rates and reduced final error norm, but higher controller norms at low levels of interaction.

Noise is now injected into the system via the DC motor, which can be considered as a low pass filter as shown in Figure 5.11 with a cut-off frequency $\approx 14\ rads^{-1}$. The chosen noise amplitude is 0.5 and the weighting matrices used in the controller are $Q = 0.05I$ and $R = I$. The results of standard feed-forward NOILC are shown in Figure 8.2 and do not include the extremely high interaction case since none of the previous controllers were able to adequately control it. For the zero interaction case the error



(a) Controller norm.

(b) Error norm.

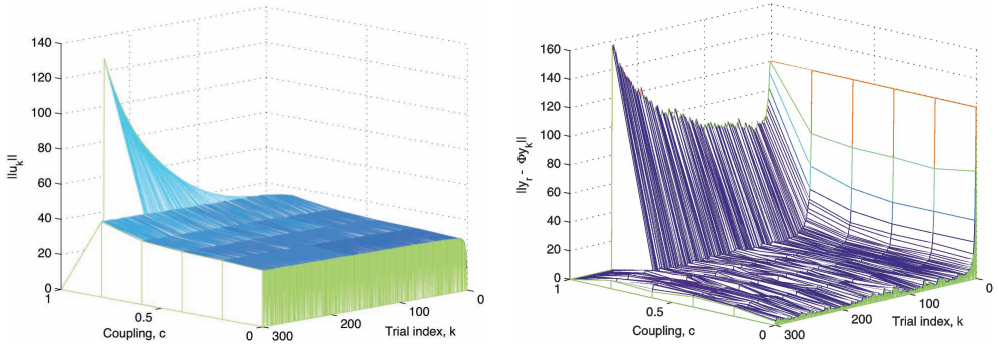
Figure 8.2: Feedforward NOILC with noise injection.

norm converges but does not reach zero due to the presence of unrepeatable noise. With higher interaction levels the noise effect increases, and in the high interaction case the error norm diverges after 200 trials. The controller norm corresponding to this error norm is higher compared to the case without noise, and all controller norms are seen to fluctuate due to the presence of noise.

Injecting noise leads to degradation in performance of the controller and can cause divergence as in the high interaction case. The effect is shown clearly in the high interaction case and the zero interaction case. In the high interaction case with noise the error norm starts to diverge after 250 trials while without noise the error norm remains at around 5. In the no interaction case with noise, the error norm fluctuates around 3 but without noise the error norm is more stable and stays at around 2. The highest interaction case with noise is not shown because of the aggressive behavior which results in the experiment being terminated. Additionally, feed-forward NOILC failed to control the highest interaction case without noise so no more information can be gleaned with the addition of noise.

8.5.2 Feedforward plus state feedback NOILC

Feedforward plus state feedback NOILC has been applied to control the system using controller update (8.15), and the same sinusoidal references as in Figure 6.1. For the extremely high interaction case the weight $Q = 0.005I$ is used, and the rest of the interaction cases use $Q = 0.05I$. The results for this experiment are shown in Figure 8.3. Similarly to the case of feed-forward NOILC, the highest interaction case presents a significant challenge for this controller. A high control effort is generated, and the system's automatic protection switches the system off after 250 trials. In all other cases the controller is successful in driving the error norm low and the control effort within a satisfactory operating range which does not increase over 300 trials.



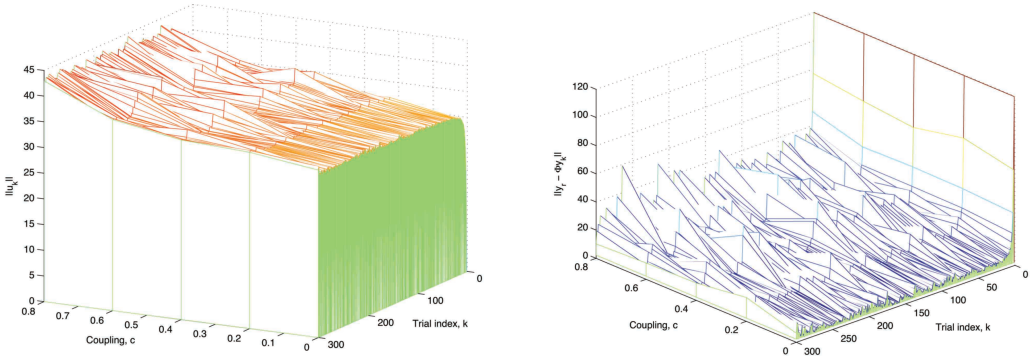
(a) Controller norm. (b) Error norm.

Figure 8.3: Standard state feedback NOILC.

Compared to feed-forward NOILC, this controller generates a similar control input norm but provides an improvement in error norm. Moreover it gives rise to a steadier control input norm value, as is clearly seen in the high interaction case in Figure 8.3(a).

With the same level of noise injection, feed-forward plus state feedback NOILC is again implemented. The resulting error norms fluctuate less with levels of interaction between $c = 0$ and 0.6 compared to the case of feed-forward NOILC with noise. For the high interaction case the result fluctuates less, the error norm generally stays under 40 and the trend does not diverge over 300 trials as shown in Figure 8.4. In general, state feedback NOILC shows more robust performance compared with feed-forward NOILC over the same conditions. It gives rise to similar controller norms but provides far superior error norms.

The experimental results for NOILC confirm that the control effort increases with the interaction level as in Section 8.3. The results also show reducing robustness when noise is injected which causes an increase in model uncertainty as stated in Theorem 2. Therefore q has been chosen to be smaller for the high interaction case.



(a) Controller norm.

(b) Error norm.

Figure 8.4: Standard state feedback NOILC with noise injection.

8.6 Summary

All the controllers implemented highlight the effect of interaction on the control effort and error norms. For coupling value c=0 to c = 0.8 the standard state feedback NOILC has lower error norm than standard feed-forward NOILC with the same controller norm while c = 1 both controller fail to keep the error norm low after 300 trials. In the injection noise, the standard state feedback NOILC still is still better than feed-forward NOILC in term of controller norm and error norm.

In general the interaction increases so too does the control effort, and this gives rise to an increased error norm due to magnified plant uncertainty and noise which are exacerbated by the larger operating region and more rapidly changing signals. Moreover, the experiments show that NOILC can provide a superior convergence rate but gives rise to higher controller norms, especially at low interaction levels. Both NOILC approaches cannot cope with the highest interaction case.

In the next chapter it is shown that both gradient ILC and NOILC can be improved by a new technique which enables them to cope with the highest interaction level.

Chapter 9

Point-to-point ILC

The ILC approaches seen thus far have displayed limited ability to deal with highly interacting systems. NOILC and gradient ILC are among the few approaches that have been applied to solve the point-to-point (or intermediate point) ILC problem where the need to track all reference points in $\mathbf{y}_d$ is relaxed. This will be shown to be an effective tool in addressing performance limitations associated with highly interacting MIMO systems. In Freeman (2012) gradient ILC is applied to this problem and in Owens et al. (2012) NOILC is employed in both feedback and feed-forward forms. In the next subsection results are summarised and expanded using the same notation.

9.1 Problem description

The point-to-point problem considers the case in which the component of the reference $\mathbf{y}_d$ corresponding to the j^{th} plant output is required to be tracked only at a fixed number, $M_j \leq N$, of sample instants given by $0 \leq n_{j,1} < n_{j,2} < \cdots < n_{j,M_j} < N$. Suppose the ILC algorithms considered in Chapter 8 and 7 were required to track these time instants. Both involve minimisation of the cost

$$J(\mathbf{u}_{k+1}) = \left\{ [\mathbf{y}_d - \mathbf{y}_{k+1}]^T Q [\mathbf{y}_d - \mathbf{y}_{k+1}] + [\mathbf{u}_{k+1} - \mathbf{u}_k]^T R [\mathbf{u}_{k+1} - \mathbf{u}_k] \right\} \tag{9.1}$$

with weighting matrices

$$Q = \text{diag}\{Q(0), Q(1), \ldots, Q(N-1)\}, \qquad R = \text{diag}\{R(0), R(1), \ldots, R(N-1)\} \tag{9.2}$$

Hence this simply involves the choice

$$Q(i) \begin{cases} > 0 & \text{if} \quad i = (n_{j,q} - 1)p + j, \quad q = 1, \ldots M_j, \quad j = 1, \ldots p, \\ = 0 & \text{otherwise} \end{cases} \tag{9.3}$$

in the cost (7.6) with the resulting updates (8.9), (8.15) and (7.8) unchanged for gradient ILC and NOILC respectively. Subsequent analysis however requires the cost to be rewritten to exclude points that are not tracked, becoming

$$J_{k+1}(\mathbf{u}_{k+1}) \quad = \quad ||\overline{\mathbf{e}}_{k+1}||^2_{\overline{\mathbb{Y}}} + ||\mathbf{u}_{k+1} - \mathbf{u}_k||^2_{\mathbb{U}} \tag{9.4}$$

with $\overline{Q} = \Phi Q \Phi^T$ where $\overline{\mathbf{e}}_{k+1} = \mathbf{y}_r - \Phi \mathbf{y}_{k+1}$ is the point-to-point tracking error and the inner product $< \overline{\mathbf{y}}_1, \overline{\mathbf{y}}_2 >_{\overline{\mathbb{Y}}} = \overline{\mathbf{y}}_1^T \overline{Q} \overline{\mathbf{y}}_2$. The vector $\mathbf{y}_r$ contains only those points required to be tracked, in the same order as they appear in $\mathbf{y}_d$. The matrix $\Phi \in \mathbb{R}^{M \times pN}$ is defined as follows:

Introduce $\boldsymbol{\psi} \in \mathbb{R}^{pN}$ with elements

$$\psi_i = \begin{cases} 1 & \text{if} \quad \lfloor (i-1)/p \rfloor \in \mathcal{S}_{i - \lfloor (i-1)/p \rfloor p}, \\ 0 & \text{otherwise} \end{cases} \tag{9.5}$$

where $\lfloor \cdot \rfloor$ denotes the 'floor' function and the set $\mathcal{S}_j = n_{j,1}, ..., n_{j,M_j}$. The former is a vector whose $(i \times p + j)^{th}$ point is 1 if the j^{th} output at time i is required to be tracked, and 0 otherwise. Φ is produced by splitting each non-zero element of $\boldsymbol{\psi}$ into a separate row, giving

$$\Phi_{i,j} = \begin{cases} 1 & \text{if} \quad \psi_j = 1, \sum_{q=1}^{j} \psi_q = i \\ 0 & \text{otherwise} \end{cases} \tag{9.6}$$

with $M = \sum_{j=1}^{p} M_j$. When any output vector is pre-multiplied by Φ, it extracts the components that correspond to prescribed point-to-point locations, whilst retaining their original order, hence $\mathbf{y}_r = \Phi \mathbf{y}_d$.

Consider the general ILC control update using the point-to-point algorithm

$$\mathbf{u}_{k+1} \quad = \quad \mathbf{u}_k + L(\mathbf{y}_r - \Phi \mathbf{y}_k) \tag{9.7}$$

$$\quad = \quad \mathbf{u}_k + L\overline{\mathbf{e}}_k \tag{9.8}$$

The convergence condition is given by

$$\rho(I_M - \Phi G L) < 1 \tag{9.9}$$

and will be shown to provide greater performance than standard ILC which corresponds to $\Phi = I_M,\ M = N$.

9.2 Point-to-point ILC motivation

Theorem 9.1. *Let d denote the rank deficiency of the plant matrix G (the number of linearly dependent rows). If $d > 0$ the ILC update (9.7) with $\Phi = I$ cannot force the plant to track an arbitrary reference trajectory $\mathbf{y}_d$. However the point-to-point update*

(9.7) with $\Phi \neq I$ can enforce tracking of an arbitrary reference $\mathbf{y}_r$ if and only if the tracked points are chosen such that

$$M \leq Np - \max\{d, N(p-m)\} \tag{9.10}$$

Proof. A necessary and sufficient condition for an operator L to exist satisfying (9.9) is that rank $(\Phi G) = M$. For the standard ILC case $\Phi = I$, $M = N$ and hence rank $(\Phi G) = N - d < M$, leading to $I_M - \Phi G L$ having d eigenvalues at unity. Now the i^{th} row of ΦG is the $(j|\Phi_{i,j} = 1)^{th}$ row of G, hence if $p \leq m$ and the point-to-point samples are chosen to correspond to any subset of linearly independent rows of G, the convergence condition (9.9) can be satisfied. If $p > m$ then the additional condition $M \leq Nm$ is imposed. □

The ability of point-to-point ILC to employ a modified standard reference to recover feasibility is extremely important, however many tasks are naturally defined only at a small number of points, and hence additional benefits may also be expected by not enforcing unnecessary tracking.

To show its ability to address the limitations caused by MIMO interaction, the following lemma is required.

Lemma 9.2. *Let the matrix Φ correspond to a point-to-point tracking task defined using (9.3), and suppose an arbitrary point $n_{i,j}$ is removed from the tracking specification. This corresponds to a new Φ matrix denoted by $\tilde{\Phi}$, which is equal to Φ but with the q^{th} row removed (where $q = i \times p + j$). Let the M eigenvalues of the matrix $A = (\Phi G)(\Phi G)^T$ be denoted $\lambda_M < \lambda_{M-1} \cdots < \lambda_2 < \lambda_1$, which also equal the singular values since A is Normal. Similarly, let the $M-1$ eigenvalues of the matrix $B = (\tilde{\Phi} G)(\tilde{\Phi} G)^T$ be denoted $\mu_M < \mu_{M-1} \cdots < \mu_3 < \mu_2$, which also equal the singular values since B is Normal. Then the following relationship holds*

$$\lambda_M < \mu_M < \lambda_{M-1} < \mu_{M-1} \cdots < \mu_3 < \lambda_2 < \mu_2 < \lambda_1. \tag{9.11}$$

and in particular

$$\overline{\sigma}(\tilde{\Phi} G(\tilde{\Phi} G)^T) < \overline{\sigma}(\Phi G(\Phi G)^T) \tag{9.12}$$

and

$$\underline{\sigma}(\tilde{\Phi} G(\tilde{\Phi} G)^T) > \underline{\sigma}(\Phi G(\Phi G)^T) \tag{9.13}$$

Proof. This is a special case of Cauchy's Interlace Theorem for eigenvalues of Hermitian matrices, with a full proof appearing in Freeman and Tan (2013). □

Lemma 9.2 shows how the maximum singular value, $\overline{\sigma}(\Phi G(\Phi G)^T)$, is monotonically reduced by removal of tracking points within the ILC task. At the same time the minimum singular value, $\underline{\sigma}(\Phi G(\Phi G)^T)$, is monotonically increased. The next sections illustrate the performance advantages it provides.

9.3 Point-to-point gradient ILC

9.3.1 Algorithm description

Application of gradient ILC using the cost function (9.4) yields the update

$$\mathbf{u}_{k+1} \quad = \quad \mathbf{u}_k + \beta(\Phi G)^* \overline{\mathbf{e}}_k \tag{9.14}$$

where $(\Phi G)^* = R^{-1}(\Phi G)^T \overline{Q}$. This is simply the general form (7.8) but with a redefined plant operator to account for the choice of $Q(i)$ components (9.3). Following previous comments, consider an arbitrary choice of $\bar{Q} = qI$, $0 < q < \infty$, used in input (9.14). It can then be shown that any choice of positive scalar β in the approximate range $\left(0, \quad \frac{1}{q\|\Phi G(\Phi G)^T\|}\right)$ minimises the cost function (9.4) for a particular choice of $\bar{Q}$ and R weights. In particular, (9.4) is minimised for the explicit choice of $\bar{Q}$ and R using a value of β given by

$$\hat{\beta} = \frac{\overline{\mathbf{e}}_k^T Q \Phi G(\Phi G)^* \overline{\mathbf{e}}_k}{\| \Phi G(\Phi G)^* \overline{\mathbf{e}}_k \|_{\mathbb{Y}}^2 + \| (\Phi G)^* \overline{\mathbf{e}}_k \|_{\mathbb{U}}^2} \tag{9.15}$$

9.3.2 Convergence analysis

From (9.14) the error evolution is given by

$$\overline{\mathbf{e}}_{k+1} \quad = \quad (I - \beta \Phi G(\Phi G)^*) \overline{\mathbf{e}}_k \tag{9.16}$$

A necessary and sufficient convergence condition is

$$\rho\big(I - \beta \Phi G(\Phi G)^*\big) < 1 \tag{9.17}$$

so that

$$0 < \beta \sigma_i\big(\Phi G(\Phi G)^*\big) < 2, \ \forall i \tag{9.18}$$

giving

$$\beta < \frac{2}{||\Phi G(\Phi G)^*||}. \tag{9.19}$$

Convergence to zero error is guaranteed if ΦG has full rank, and either the trial-dependent gain (9.15) is employed with any choice of symmetric positive definite $\bar{Q}$ and R, or a fixed gain satisfying

$$0 < \beta < \frac{2}{q \| \Phi G(\Phi G)^T \|} \tag{9.20}$$

is used together with $R = I$ and any choice of $\bar{Q} = qI$, $0 < q < \infty$.

Let V be the eigenvector matrix of $\Phi G(\Phi G)^*$, with Λ a diagonal matrix of corresponding eigenvalues $\left(\sigma_i(\Phi G(\Phi G)^*)\right)^2$. Then (9.16) can be written as

$$\overline{\mathbf{e}}_{k+1} = V(1-\beta\Lambda)V^{-1}\overline{\mathbf{e}}_k \tag{9.21}$$

so that

$$V^{-1}\overline{\mathbf{e}}_k = (1-\beta\Lambda)^k V^{-1}\overline{\mathbf{e}}_0. \tag{9.22}$$

Since $V^{-1} = V^T$, the component of $\Phi\mathbf{e}_0$ projected onto the i^{th} eigenvector of $\Phi G(\Phi G)^*$ hence evolves as $\left(1-\beta\left(\sigma_i(\Phi G(\Phi G)^*)\right)^2\right)^k$. If $\underline{\sigma}^2$ is the minimum eigenvalue then the point-to-point reference with slowest convergence rate is a scalar multiple of its corresponding eigenvector. Similarly if $\overline{\sigma}^2$ is the maximum eigenvalue then the point-to-point reference with the fastest convergence rate is a scalar multiple of its corresponding eigenvector. For an arbitrary reference, the error norm sequence lies in the interval

$$\left(1-\beta\left(\overline{\sigma}(\Phi G(\Phi G)^*)\right)^2\right)^k \leq \frac{\|\overline{\mathbf{e}}_k\|_{\overline{\mathbb{Y}}}}{\|\overline{\mathbf{e}}_0\|_{\overline{\mathbb{Y}}}} \leq \left(1-\beta\left(\underline{\sigma}(\Phi G(\Phi G)^*)\right)^2\right)^k \tag{9.23}$$

For a given reference an approximation of the convergence rate is the weighted sum

$$\frac{\| V^T\mathbf{y_r} \|_p}{\| (I-\beta\Lambda)V^T\mathbf{y_r} \|_p} \tag{9.24}$$

where $\|\cdot\|_p$ denotes the p-norm.

9.3.3 Control effort

Application of (9.14) for any β satisfying (9.15) or (9.20) yields

$$\begin{aligned}
\mathbf{u}_{k+1} &= \mathbf{u}_k + \beta(\Phi G)^*(\mathbf{y}_r - \Phi G\mathbf{u}_k) && (9.25)\\
&= (I-\beta(\Phi G)^*\Phi G)\mathbf{u}_k + \beta(\Phi G)^*\mathbf{y}_r && (9.26)\\
&= (I-\beta(\Phi G)^*\Phi G)^{k+1}\mathbf{u}_0 + \left(\sum_{i=0}^{k}(I-\beta(\Phi G)^*\Phi G)^i\right)(\beta(\Phi G)^*)\mathbf{y}_r && (9.27)\\
\Rightarrow \mathbf{u}_\infty &= (I-(I-\beta(\Phi G)^*\Phi G))^{-1}\beta(\Phi G)^*\mathbf{y}_r && (9.28)\\
&= (\Phi G)^\dagger\mathbf{y}_r. && (9.29)
\end{aligned}$$

Hence the equation (7.26) is replaced by

$$||\mathbf{u}_\infty|| = ||(\Phi G)^\dagger\mathbf{y}_r|| \leq \frac{||\mathbf{y}_r||}{\underline{\sigma}\left(\Phi G(\Phi G)^*\right)}. \tag{9.30}$$

9.3.4 Robustness analysis

In the point-to-point case the uncertainty model from (7.27) is replaced by

$$\Phi\tilde{G} = U\Phi G \tag{9.31}$$

and the condition (7.28) changes to

$$0 < \beta < \frac{2}{\max_i \left\{0, \frac{\Re\{\lambda_i(U)\}^2 + \Im\{\lambda_i(U)\}^2}{\Re\{\lambda_i(U)\}}\right\}} \frac{1}{\overline{\sigma}^2(\Phi G(\Phi G)^*)} \tag{9.32}$$

Applying Lemma 9.2 shows that the robustness bound increases as points are removed from the point-to point-tracking task. Appropriate use of point-to-point ILC can therefore mitigate the robustness degradation observed in Section 7.1.4.

9.3.5 Experimental results

Table 9.1 shows the maximum and minimum of singular values of GG^T and $\Phi G(\Phi G)^T$ for varying interaction levels. Note that the input bound is conservative and will only match observed results when $\overline{\mathbf{y}}_d$ is a multiple of the eigenvector corresponding to the slowest eigenvalue. For standard gradient ILC it is clear that:

1. convergence of the slowest mode is not possible in practice since $\underline{\sigma}(GG^*) = \frac{q}{r}\underline{\sigma}(GG^T)$ approaches 0 (from (7.19)),
2. the input bound required is also prohibitively large (from (7.26)),
3. as interaction increases, robustness bounds reduce since $\overline{\sigma}(GG^T)$ increases (from (7.28) and (6.10))

For point-to-point gradient ILC, the convergence, input norm and robustness properties all significantly improve. In particular

1. convergence of the slowest mode is far more rapid, but still reduces as interaction increases (from (9.23), (6.10) and (9.13)),
2. the input norm bound is far smaller, but still increases as interaction increases (from (9.30), (6.10) and (9.13)),
3. the robustness bounds are far larger, but still reduce as interaction increases (from (9.32), (9.12) and (6.10)).

The point-to-point gradient ILC update (9.14) has been applied with the optimal β calculated using (9.15). The highest performance found in practice corresponds to the

Table 9.1: The smallest, largest singular value and the bound of controller effort for varying interaction level.

Coupling	Standard ILC		Point-to-point gradient ILC		
				Robustness†	Input bound
c	$\underline{\sigma}(GG^T)$	$\overline{\sigma}(GG^T)$	$\underline{\sigma}(\Phi G(\Phi G)^T)$	$\overline{\sigma}(\Phi G(\Phi G)^T)$	$\frac{\|\|\mathbf{y}_r\|\|_{\overline{\mathbb{Y}}}}{\underline{\sigma}}$
0	$5.06\ 10^{-9}$	456.51	0.1891	5.5655	87.3829
0.2	$5.10\ 10^{-11}$	457.06	0.1802	5.5872	89.5615
0.4	$1.13\ 10^{-9}$	457.11	0.1686	6.5419	98.0077
0.6	$1.32\ 10^{-11}$	459.01	0.1616	6.5710	102.2531
0.8	$2.79\ 10^{-10}$	488.04	0.1482	7.0148	111.4987
1	$3.77\ 10^{-11}$	565.01	0.0549	7.5565	300.9854

†Robustness measure appearing in (9.32) : a smaller value indicates greater robustness.

weighting matrices $R = I$, $\overline{Q} = 0.1I$ for the highest interaction case and the remaining interaction cases employ $\overline{Q} = I$. These have been found after application of many alternative values. The reference signals are shown in Figure 9.1. Both references have a total length of $T = 2.4$ seconds. Six reference points are also defined for use in point-to-point ILC and are given by $\overline{y}_d = [6.9,\ 0.5,\ 3.2,\ 0.9,\ -6.9,\ 1.5]^T$ at the time samples $\mathcal{S}_1 = \{0.5,\ 0.9,\ 1.5\}$, $\mathcal{S}_2 = \{0.8,\ 1.4,\ 1.7\}$. The odd values of $\overline{y}_d$ relate to input 1 and the even values relate to input 2.

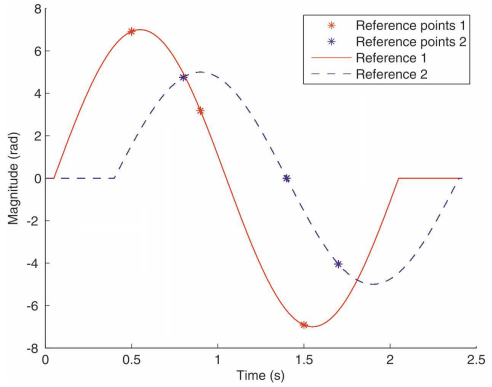
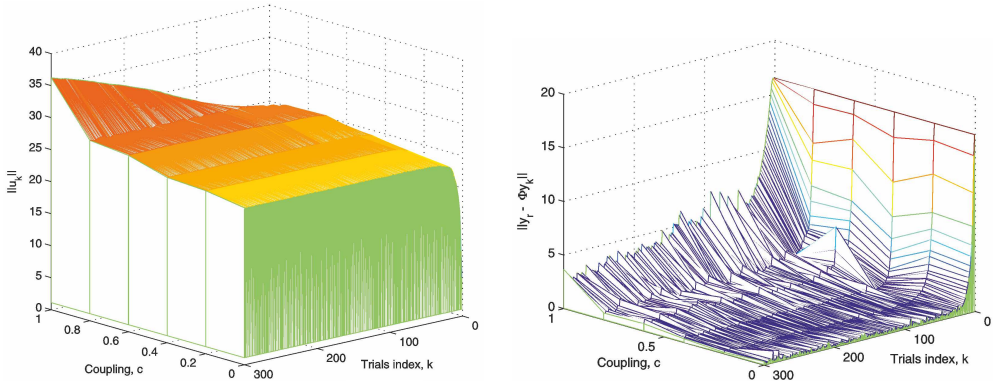


Figure 9.1: References for standard and point-to-point ILC controllers.

For values of $c = 0, 0.2, 0.4, 0.6, 0.8, 1$ the errors norm maintains low values of around 1, and takes about 10 trials to reach that value. The convergence rates and final errors are quite similar for these cases but they have a different saturated control norm since a higher interaction level generally requires a higher control effort to force the system to track the reference signals. Increasing the interaction level leads to increased error norms and also controller norms. A significant change in convergence rate is shown in the highest interaction case because it has a lower value of q in order to increase robustness to cope with high uncertainty and noise. Full results are shown in Figure 9.2 which confirms that a zero error norm is still hard to achieve in practice due to model uncertainty and noise.

Similarly to the previous cases, the final error depends on the coupling magnitude. However, now reasonable tracking is achieved in the highest interaction case, where, after 300 trials the error norm output is approximately 3. This confirms far superior performance compared to the standard ILC case. In general, point-to-point gradient ILC has a lower controller norm for all interaction cases compared with the standard gradient-based ILC, which has reduced the effect of uncertainty leading to convergence to a smaller error norm.



(a) Controller norm. (b) Error norm.

Figure 9.2: Point-to-point gradient ILC.

Point-to-point gradient ILC shows improved performance and robustness properties compared with standard gradient ILC and standard NOILC on this MIMO facility. Increased interaction is also shown to correspond to a higher input norm, higher error norm and slower convergence. These features magnify effects of model uncertainty, which are further exacerbated when the plant is forced to track additional points in the reference giving rise to learning transients and instability.

9.4 Point-to-point NOILC

In this section NOILC the performance properties in Chapter 8 are generalised for application to the point-to-point problem. This includes the 'standard' ILC problem discussed in Sections 8.1 and as a special case, through selection of $\Phi = I$.

9.4.1 Algorithm description

9.4.1.1 Point-to-point feed-forward NOILC

Employing the choice of Q components (9.3) for point-to-point tracking simply means substituting for the plant operator and error in (8.9). This yields the update

$$\mathbf{u}_{k+1} = \mathbf{u}_k + (\Phi G)^*(I + \Phi G(\Phi G)^*)^{-1}\overline{\mathbf{e}}_k \tag{9.33}$$

where $\overline{\mathbf{e}}_k$ is the experimentally recorded point-to-point tracking error.

9.4.1.2 Point-to-point feed-forward plus state feedback NOILC

The point-to-point implementation for state feedback NOILC is simply (8.13) - (8.15) using the weight selection (9.3).

9.4.2 Convergence analysis

The tracking error evolution (8.16) is exchanged for

$$\overline{\mathbf{e}}_{k+1} \quad = \quad (I + \Phi G(\Phi G)^*)^{-1}\overline{\mathbf{e}}_k. \tag{9.34}$$

and tracking error on the k^{th} trial is bounded by

$$\left(1 + \overline{\sigma}^2(\Phi G(\Phi G)^*)\right)^{-k} \leq \frac{\|\overline{\mathbf{e}}_k\|_{\overline{\mathbb{Y}}}}{\|\overline{\mathbf{e}}_0\|_{\overline{\mathbb{Y}}}} \leq \left(1 + \underline{\sigma}^2(\Phi G(\Phi G)^*)\right)^{-k}. \tag{9.35}$$

The slowest convergence rate is dictated by $\underline{\sigma}^2(\Phi G(\Phi G)^*)$ and, assuming for simplicity that $R = rI$, $Q = qI$, strictly increases as points are removed from the tracking task via Lemma 9.2. This hence mitigates reduction in convergence rates caused by MIMO interaction, as discussed in Section 8.2.

9.4.3 Control effort

The limiting input (8.23) is exchanged for

$$\mathbf{u}_{k+1} \quad = \quad (\Phi G)^\dagger \overline{\mathbf{y}}_d \tag{9.36}$$

which can be shown to correspond to the minimum input energy solution to the tracking task, given by

$$\arg\min_{\mathbf{u}} \left\{ \|\mathbf{u}\|^2 : \overline{y}_d = \Phi G \mathbf{u} \right\} \tag{9.37}$$

The input norm bound (8.24) is replaced by

$$\|\mathbf{u}_\infty\| = \left\| (\Phi G)^\dagger \overline{\mathbf{y}}_d \right\|_{\overline{\mathbb{Y}}} \leq \frac{\|\overline{\mathbf{y}}_d\|_{\overline{\mathbb{Y}}}}{\underline{\sigma}(\Phi G(\Phi G)^*)} \tag{9.38}$$

and strictly reduces as points are removed from the tracking task via Lemma 9.2. Hence suitably reducing the number of points tracked by ILC mitigates degradation of system performance caused by increased MIMO interaction.

9.4.4 Robustness analysis

In the point-to-point case, the specification of the true plant in Theorem 2 is replaced by

$$\Phi\tilde{G} = U\Phi G \tag{9.39}$$

and the condition (8.26) becomes

$$0 < \frac{q}{r} < \frac{2}{\max_i \left\{0, \frac{\Re\{\lambda_i(U)\}^2+\Im\{\lambda_i(U)\}^2}{\Re\{\lambda_i(U)\}} - 2\right\}} \cdot \frac{1}{\overline{\sigma}^2(\Phi G(\Phi G)^T)} \tag{9.40}$$

Applying Lemma 9.2 shows that the robustness bound increases as points are removed from the point-to point-tracking task. Appropriate use of point-to-point ILC can therefore mitigate the robustness degradation observed in Section 8.4.

9.4.5 Experimental results

Table 9.2 shows parameters governing the robustness, worse case convergence rate, and input norm for the six levels of interaction. The value of $\frac{q}{r}$ required to enforce a NOILC convergence rate of 0.5 is also shown. Note that the input bound is conservative and will only match observed results when $\overline{\mathbf{y}}_d$ is a multiple of the eigenvector corresponding to the slowest eigenvalue. For standard NOILC it is clear that:

1. convergence of the slowest mode is not possible in practice since $\underline{\sigma}(GG^*) = \frac{q}{r}\underline{\sigma}(GG^T)$ approaches 0 (from (8.19)),
2. the input bound required is also prohibitively large (from (8.24)),
3. as interaction increases, robustness bounds reduce since $\overline{\sigma}(GG^T)$ increases (from (8.26) and (6.10))

For point-to-point NOILC, the convergence, input norm and robustness properties all significantly improve. In particular

1. convergence of the slowest mode is far more rapid, but still reduces as interaction increases (from (9.35), (6.10) and (9.13)),
2. the input norm bound is far smaller, but still increases as interaction increases (from (9.38), (6.10) and (9.13)),
3. the robustness bounds are far larger, but still reduce as interaction increases (from (9.40), (6.10) and (9.12)).

Table 9.2: Robustness measures, input bounds and convergence rates of slowest mode for NOILC.

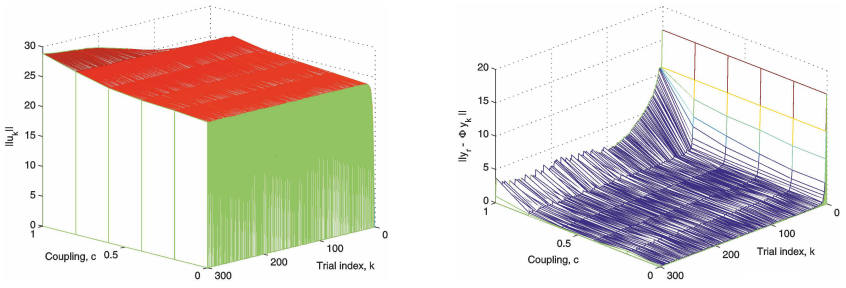
Coupling	Standard ILC		Point-to-point ILC				
c	$\underline{\sigma}(GG^T)$	$\overline{\sigma}(GG^T)$	$\underline{\sigma}(\Phi G(\Phi G)^T)$	Robustness† $\overline{\sigma}(\Phi G(\Phi G)^T)$	Convergence‡ $(1+\underline{\sigma}^2)^{-1}$	Convergence§ q	Input bound $\frac{\|\|\mathbf{y}_r\|\|_{\overline{Y}}}{\underline{\sigma}}$
0	$5.06\ 10^{-9}$	456.51	0.1891	5.5655	0.9655	27.9651	87.3829
0.2	$5.10\ 10^{-11}$	457.06	0.1802	5.5872	0.9671	29.3770	89.5615
0.4	$1.13\ 10^{-9}$	457.11	0.1686	6.5419	0.9724	35.1791	98.0077
0.6	$1.32\ 10^{-11}$	459.01	0.1616	6.5710	0.9746	38.2928	102.2531
0.8	$2.79\ 10^{-10}$	488.04	0.1482	7.0148	0.9785	45.5306	111.4987
1	$3.77\ 10^{-11}$	565.01	0.0549	7.5565	0.9970	331.7839	300.9854

†Robustness measure appearing in (9.40) : a smaller value indicates greater robustness.
‡Convergence rate of slowest mode, $q = 1$.
§Value of q required for the slowest mode convergence rate of 0.5.

Table 9.2 hence confirms results in Chapter 7 linking interaction with maximum and minimum singular values of GG^T. It also confirms the results in Section 9.2, 9.3, 9.4 linking point-to-point tracking with maximum and minimum singular values of $\Phi G(\Phi G)^T$.

9.4.5.1 Point-to-point feed-forward NOILC

The same references as used in the gradient ILC case of Section 9.3.5 are used and shown in Figure 9.1. Point-to-point feed-forward NOILC is implemented to track $\mathbf{y}_r$ using update equation (9.33) with $R = I$ and $\overline{Q} = I$ found to perform the best for all interaction cases except the highest interaction level which uses $\overline{Q} = 0.05I$.



(a) Controller norm. (b) Error norm.

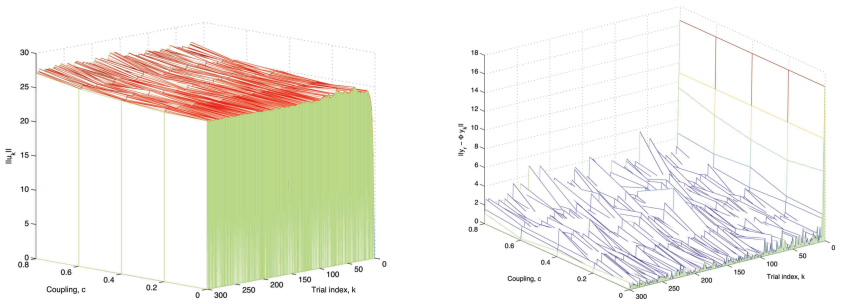
Figure 9.3: Point-to-point feed-forward NOILC

In Figure 9.3(b), the error norm shows a superior result to point-to-point gradient ILC. For levels of interaction between $c = 0$ and $c = 0.8$, the error norm is below 0.5 and is even smaller in the zero interaction case. The convergence is also fast and it takes only 3 trials to reach its final error norm. The most challenging case is still the highest interaction level, which has a slow convergence rate due to a lower value of q. It takes more than 100 trials to reach the final error value but this is maintained over 300 trials. The corresponding control effort norms are shown in Figure 9.3(a) and indicate that the controller norm increases when higher interaction is applied.

This method shows how point-to-point can mitigate the effect of uncertainty at the highest interaction level while the standard NOILC leads to instability. The point-to-point feed-forward NOILC yields favorable error and controller norm values. Compared to point-to-point gradient ILC this method shows an improvement in error norm, having a lower final value and a faster convergence rate. The controller effort has a lower value for the highest interaction case and reaches a steady-state value quicker.

The point-to-point method is again applied, but now noise is injected as described in Section 8.5.1. The controller update is still given by (9.33). Figure 9.4 shows both controller norms and error norms results. The controller norm and error norm increase when the interaction level rises. For interaction levels between $c = 0$ and $c = 0.8$ error norms fluctuate due to the injected noise. This leads to fluctuation in controller norm values also. The highest interaction case is not shown in this experiment because it responded aggressively to noise controller update displays instablity. For safety reasons and to protect the mechanical system the data for this interaction level has not been collected.

Compared to standard feed-forward NOILC with noise results shown in Figure 8.2, the error norm is significantly improved taking a lower value especially in the high interaction case. The point-to-point method aids the controller in maintaining a low error norm value around 3. Clearly, the controller norm also improves in term of steady value.



(a) Controller norm. (b) Error norm.

Figure 9.4: Point-to-point feed-forward NOILC with noise injection.

Overall the point-to-point algorithm helps standard NOILC to mitigate the effects of uncertainty and noise. Therefore in challenging situation like the high interaction case and the highest interaction case, divergence does not occur and the controller norm possesses a lower value.

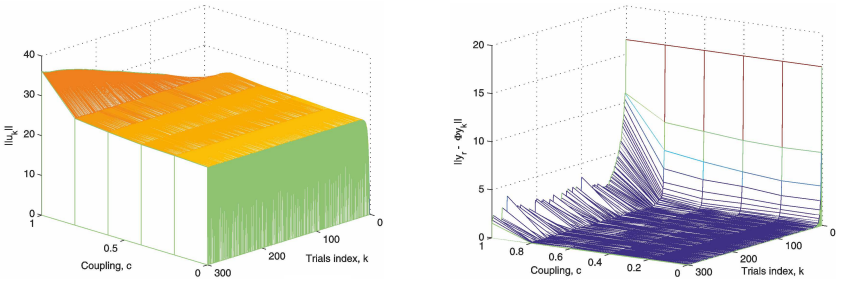
9.4.5.2 Point-to-point feed-forward plus state feedback NOILC results

The combined feed-forward and state feedback implementation of point-to-point NOILC is applied, using the controller update described in Section 9.4.1.2. The weights $\overline{Q} =$

$0.05I,\ R = I$ are employed for the highest interaction level, and the remaining cases use $\overline{Q} = I,\ R = I$. The reference is still the same as the previous case.

As shown in theory, the point-to-point method only tracks a subset of points and therefore leads to superior results in terms of both controller norm and error norm as confirmed in Figure 9.5. It yields low errors norm and almost reaches zero error after 4 trials in all interaction cases except for the highest. For interaction case $c = 1$, the convergence rate of the error norm is slower compare to other interaction levels. It takes 100 trials to reach an error norm of 3 and maintains that value until the end of the test. Corresponding to this result the input norm has a higher value than the feed-forward implementation of NOILC in order to force the system to track the reference. Hence it can be concluded that feed-forward NOILC is more robust.

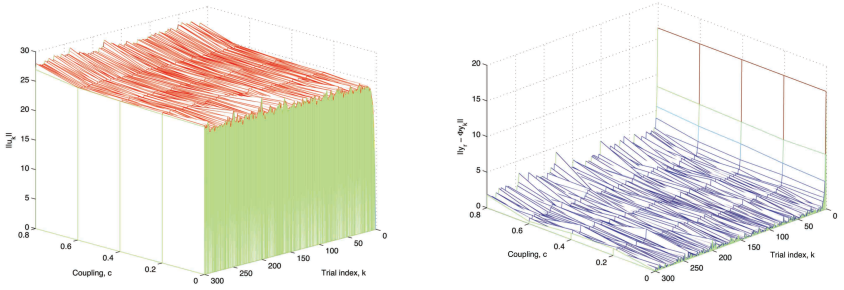
Compared to point-to-point feed-forward NOILC this method has a similar controller norm but a significantly lower error norm for all cases between $c = 0$ and $c = 0.8$. For interaction $c = 1$, the error norm has a similar value but the point-to-point feed-forward plus state feedback NOILC possesses a higher controller norm since the controller takes into account the estimated state error in the controller update. Therefore the performance is degraded by an increasing interaction level which causes a decline in robustness.



(a) Controller norm. (b) Error norm.

Figure 9.5: Point-to-point feed-forward plus state feedback NOILC.

Finally the point-to-point method is again implemented with the addition of injected noise. The error norm in Figure 9.6 confirms significant improvement compared with other controller methods. It does not fluctuate significantly and maintains a low value. At the same time the controller norm is slightly higher than the noise-free case and significantly lower than the standard NOILC case. For the high interaction level the error norm has a value of ≈ 2 which is smaller than that of point-to-point feed-forward NOILC. The control input norm is approximately the same but there is a significant improvement in the error norm. Due to protecting the mechanical system the highest interaction case is not implemented due to the aggressive response from the system.



(a) Controller norm. (b) Error norm.

Figure 9.6: Point-to-point state feedback NOILC with noise injection.

9.5 Summary

This chapter has introduced a new algorithm, termed point-to-point ILC, which can be integrated into gradient ILC and NOILC scheme such that its performance improves significantly. The theoretical properties of each previous controller have been extended to encompass this framework.

All the controllers implemented highlight the effect of interaction on the control effort and error norms. When the interaction increases so too does the control effort, and this gives rise to an increased error norm due to magnified plant uncertainty and noise which are exacerbated by the larger operating region and more rapidly changing signals. Therefore the error norm degrades, more oscillations are produced and the control effort norm is seen to increase in all cases. These confirm the theoretical predictions given in this chapter.

Both standard gradient ILC and NOILC approaches cannot cope with the highest interaction case, however experiments show that the point-to-point implementations of both algorithms restore satisfactory performance in terms of controller and final error norms. Hence the point-to-point method is a practical way of addressing the effect of a highly interacting systems if only a subset of points are required to be tracked.

Chapter 10

Conclusions and further work

A MIMO facility has been developed to meet the needs of ILC benchmarking, using theoretical analysis and extensive simulations. It has been successfully manufactured with the aid of 3D computer-aided design tools. The testbed is suitable for use with a wide range of ILC controllers which are straightforward to implement using the Matlab/Simulink environment. The facility also allows adjustment of interaction level by connecting/disconnecting the interaction dampers or clamping the coupling shaft. Additionally, disturbance/noise injection can be applied to generate a wide range of repetitive disturbances and/or random noise to make the system more challenging to control. The MIMO facility is hence an ideal tool with which to investigate a variety of controllers with respect to the level of interaction and effect of disturbance/noise injection.

With the aid of Matlab and Simulink, P-type, D-type and phase-lead ILC controllers have been applied to establish the efficacy of the MIMO system as a bench-marking platform. The controllers have shown the ability to force the MIMO system to track reference signals in the first few trials but after that the outputs diverge and cause an aggressive response. Hence advanced ILC controllers were applied in Chapter 7-9 to improve the performance. Before applying any advanced ILC method, a model for the MIMO facility was derived since the theoretical model was ill-suited due to being of a very high order and hence requiring excessive computational load. Therefore a reduced model was derived using a frequency modeling method, which yielded models suitable for implementable algorithms.

Standard gradient ILC and NOILC were introduced in Chapter 7, 8 and have been applied to control the MIMO system. These two controllers have the same cost function and have the same input norm when the number of trials tends to infinity. The two controllers also have the ability to mitigate effects of uncertainty and noise. The results show that the standard gradient ILC produces a higher controller norm than NOILC. In terms of convergence, NOILC gives improved results compared with standard gradient ILC as well as lower error norms. NOILC was shown to yield a lower error norm

and could maintain a small value during the tests. Overall, standard NOILC showed improvement over standard gradient ILC in terms of both controller norm and error norm. However, both controllers could not successfully control the MIMO facility with the highest interaction setting due to nonlinear factors and the model uncertainty that is associated with the highest interaction level. Hence alternative algorithms were required to minimise these factors.

The point-to-point algorithm was introduced in Chapter 9 to eliminate points within the demand reference, which were deemed not important to track. This method was shown to reduce the effect of uncertainty and disturbance which built up during trials. It was shown to maintain a low error norm during 300 trials and the controller norm was substantially smaller compared to that of both NOILC and gradient ILC. Therefore, in practice, this method provides superior results compared to standard controllers.

Under noise injection conditions, standard ILC controllers still produces satisfactory results up to the medium interaction level, however with higher interaction levels, the controllers were shown to yield unstable results since the error norm started to diverge after a few hundred trials. However, point-to-point gradient ILC and NOILC were able to cope with this condition and could keep the error norm at a low value.

Results in this thesis also confirmed the relationship between interaction level and the maximum eigenvalue of the plant model and showed that increasing interaction or noise/disturbance leads to a reduction in robustness for both NOILC and gradient ILC. With the aid of the point-to-point method, increasing the number of removed points leads to increased robustness for both controllers.

In summary, many controllers have been applied to investigate the effect of MIMO interaction and uncertainty. The facility that has been developed also shows how noise/disturbance injection degrades the performance of controllers. Basic ILC controllers which do not use a model have been shown to perform poorly while model based ILC provides superior results except in the case of the highest level of interaction. The point to- point method enhances the performance of advanced ILC even in the highest interaction or noise/disturbance conditions. Generally, the point-to-point method embeds robustness and mitigates the effects of uncertainty. The MIMO facility has hence been shown to be a valuable tool in investigating MIMO interaction and noise/disturbance conditions.

10.1 Future work

In order to improve the results achieved using model based ILC controllers, a more accurate model for the MIMO facility needs to be developed. The better the model, the better the results these controllers may be expected to give, especially at the highest interaction level.

The MIMO facility is the only benchmarking system that allows engineers to apply a broad range of ILC controllers. Therefore there is a scope for not only ILC algorithms to be investigated but for other control structures to be implemented and analysed. In general all approaches suffer from a lack of experimental analysis in the MIMO case and in the presence of noise/disturbance injection. Therefore the MIMO facility can play an important role in providing evaluation and assessment.

The MIMO system is still in phase 1, it can be modified to make a more complicated system with added spring-mass-damper components between induction motors and differential gearboxes. The devices will make the system more difficult to control and it is another challenge for ILC. The system has much more uncertainty and non-linear factors such as friction or backlash from dampers and coupling connection.

The thesis gives details of ILC performance for MIMO system under disturbance injection, varying interaction level but has not given any analysis under non-minimum phase effect for MIMO system. It is clearly that the inversed plant is unstable. Hence it is worth to investigate how NOILC, gradient ILC and point-to-point method perform with this kind of system.

In the industrial robot case the system contains non-linearities in the form of, for example, friction, backlash, and saturations. The ILC theory should be extended also to this kind of MIMO system. Especially point-to-point method is useful for robotic applications. It has been a favorite method used in rehabilitation system and gave promised results as discussed previously. Therefore it needs to investigate further for non-linear system since human arm or wrist are a non-linear model. The method can be tested and improved by using the MIMO rig before applying to try on controlling human parts.

Stochastic learning control algorithms are one class of controller that needs to be applied and evaluated using the MIMO system. These controllers were first introduced in (Saab, 2001a) for the discrete-time case. The controllers are based on P-type and D-type structures but the gain is designed using a state-space model of the system (Saab, 2001b), (Saab, 2003). The advantage of this method is that it takes into the account the disturbance model and noise, and therefore is suitable for the MIMO system which has one channel of disturbance injection. The method has also been applied to a gantry robot (Cai et al., 2008a) and compared to other controllers. However the gantry robot only has a small amount of interaction, and therefore the effect of noise and disturbance cannot be shown clearly. The MIMO facility will be a suitable platform with which to evaluate stochastic controllers, unlocking their full noise and disturbance capabilities. There is also an existing model of disturbance injection which can be used for immediate design.

Further testing can be carried out using the same technique but in a different arrangement. The method consists of two filters that need to be designed. One filter is applied to the measured error and another one to the controller update. The approach is derived and analysed in (Bristow, 2008a), (Bristow, 2010). It is based on minimising the error power

spectrum from trial to trial so that the fastest convergence can be achieved. The resulting filters also depend on a deterministic-to-stochastic ratio. Cai et al. (2011a) applied the approach and gave promising experimental results for the gantry robot. Therefore the MIMO facility provides a new challenge for the controller to address higher interaction levels and different disturbance powers.

The continuous-time NOILC has been applied to the non-minimum phase system and has yielded accurate results. For further investigation the continuous-time gradient ILC will be derived for the continuous-time case and applied on the same non-minimum phase system. The results will be evaluated and compared to those of NOILC.

In addition to these controllers there are many different approaches that are suitable for the MIMO system, for example adaptive iterative learning control or repetitive control methodologies.

Appendix A

System components

A.1 Computer aided designs

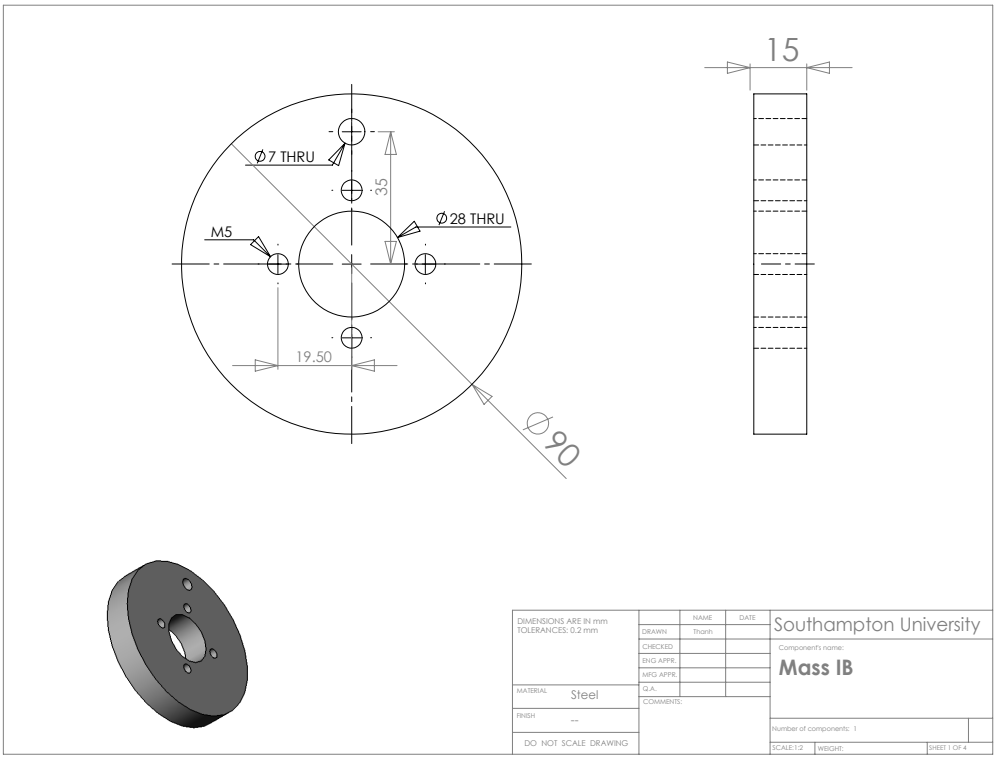


Figure A.1: Mass IB for the system

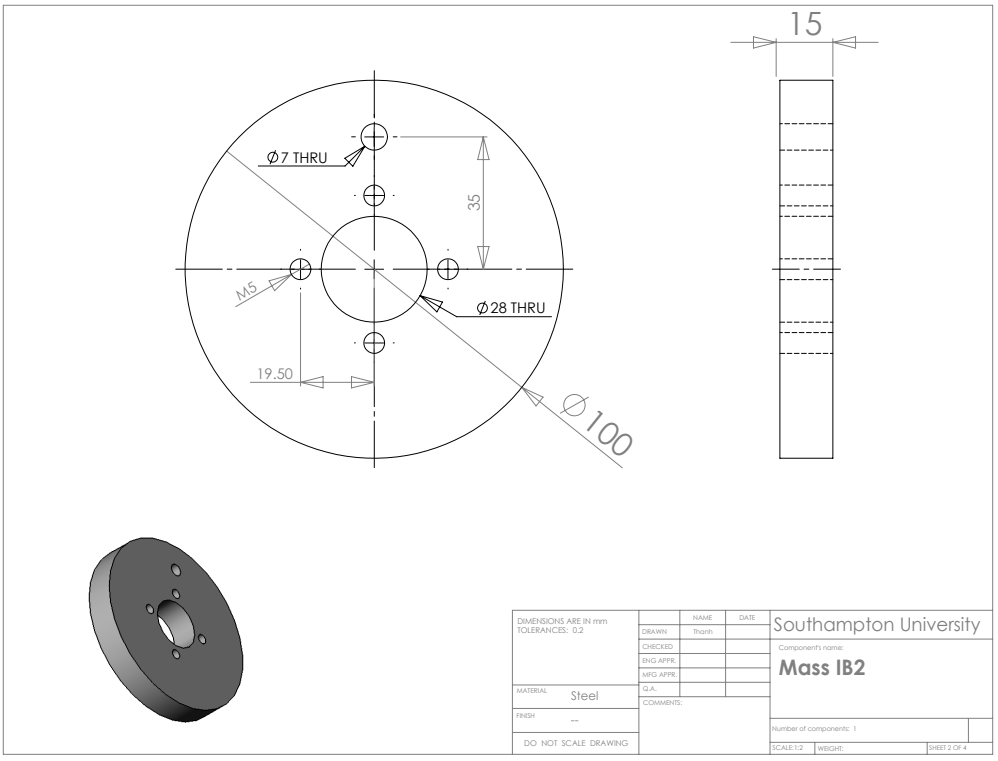


Figure A.2: Mass IB2 for the system

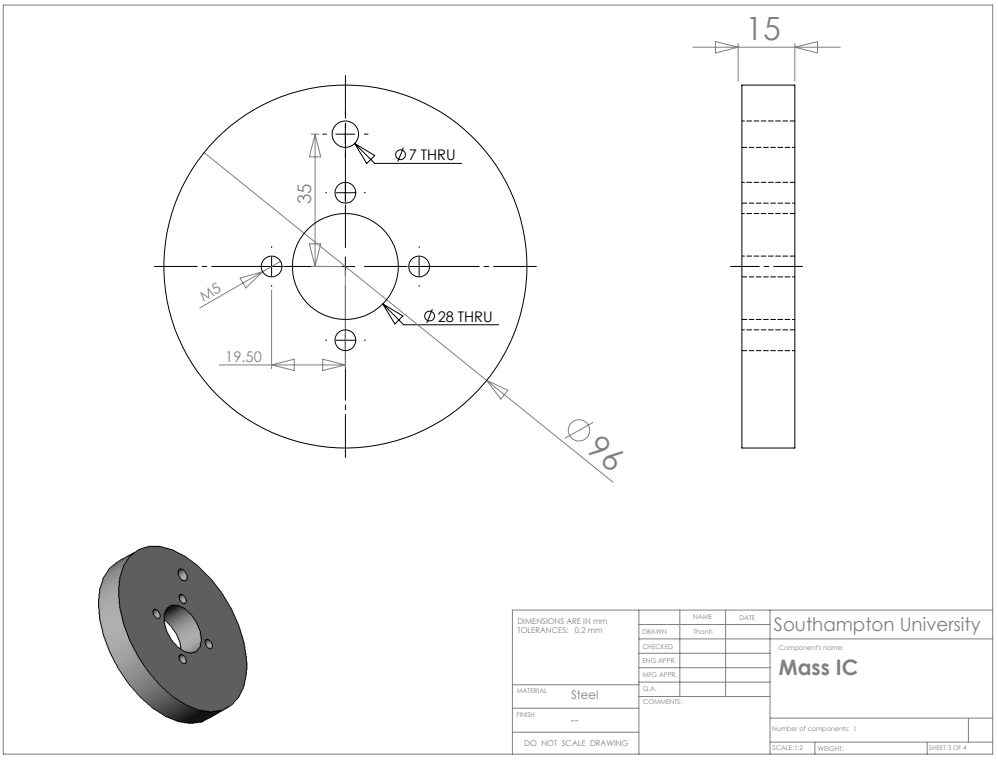


Figure A.3: Mass IC for the system

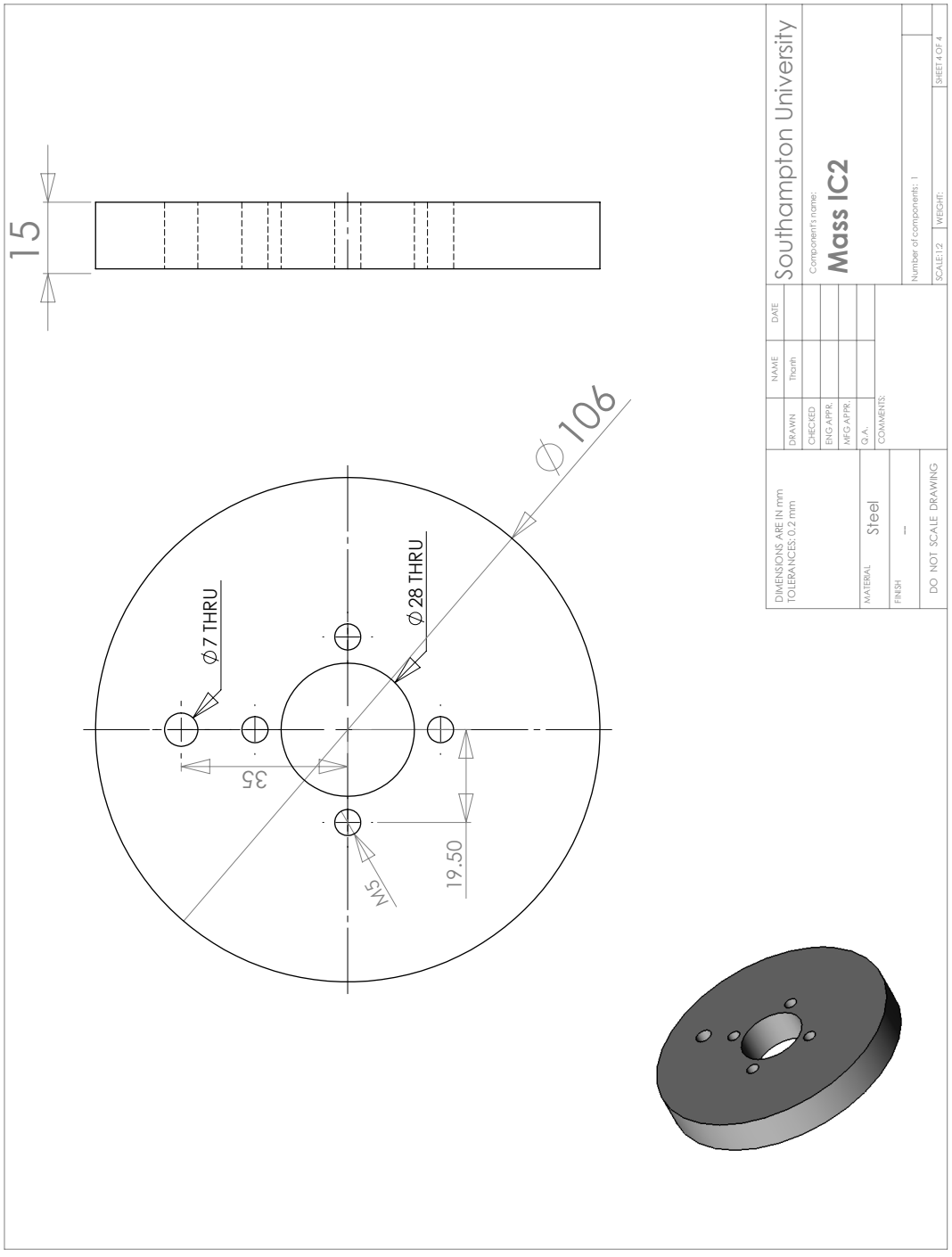


Figure A.4: Mass IC2 for the system

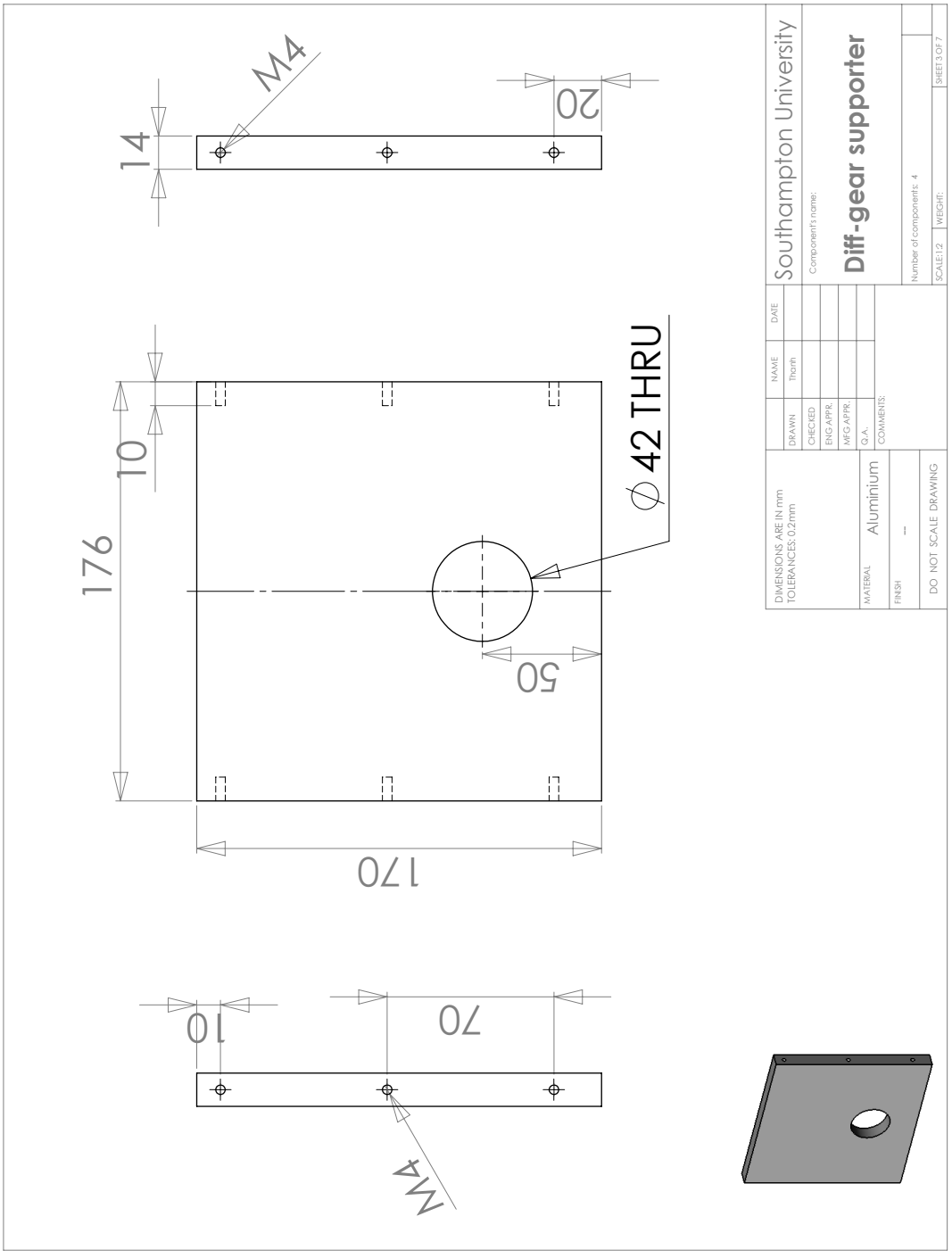


Figure A.5: Supporter for differential gear.

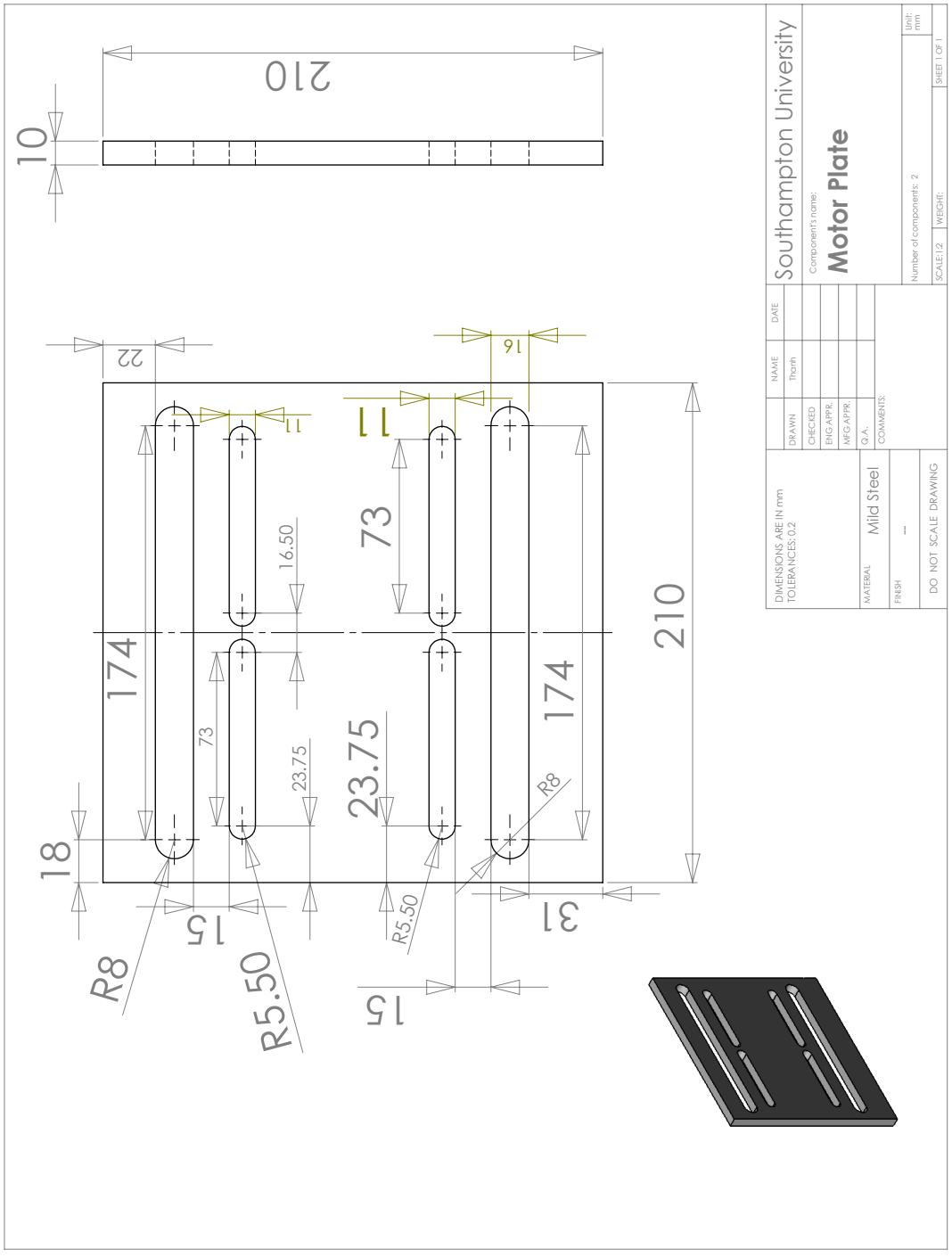


Figure A.6: Motor plate.

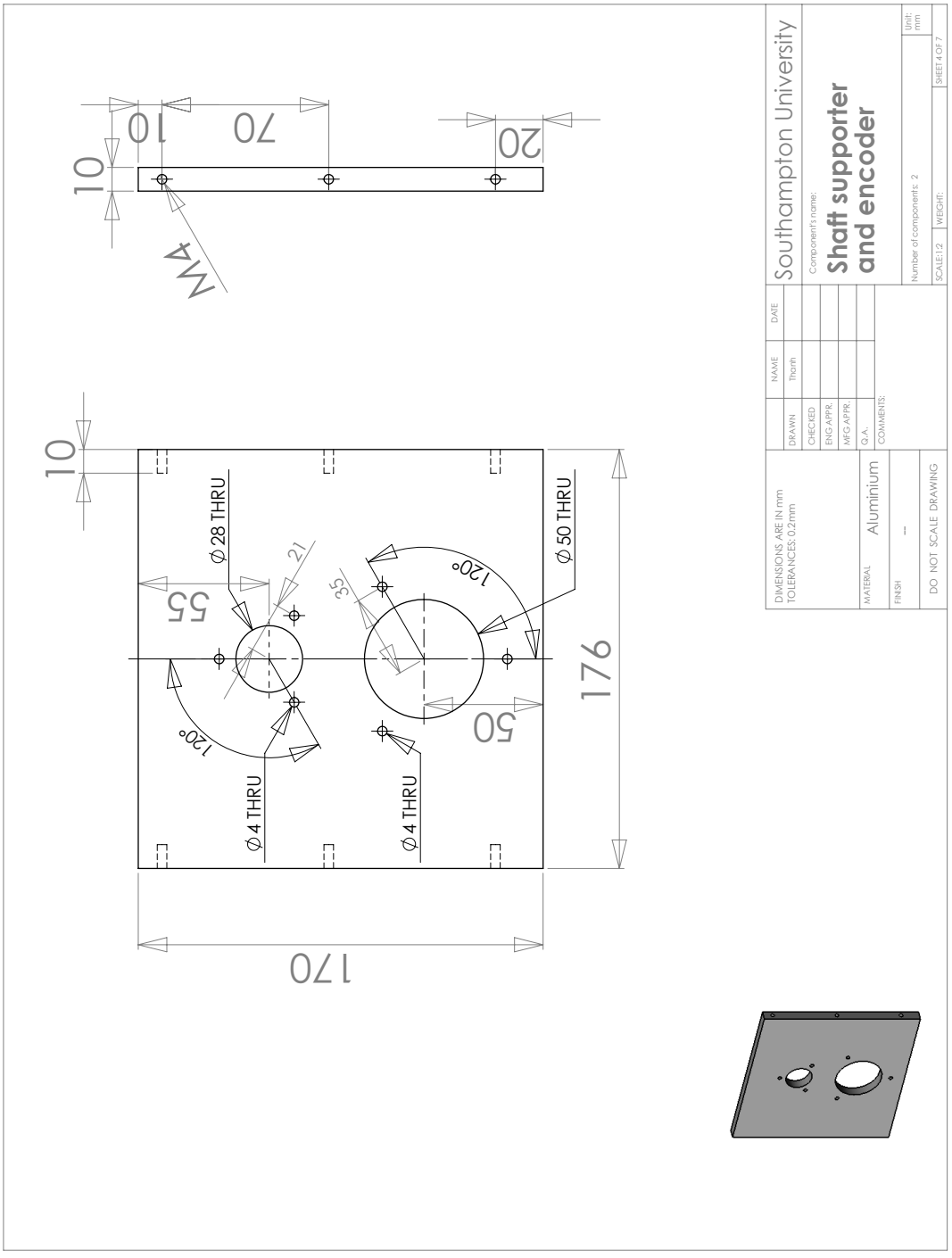


Figure A.7: Encoder and shaft supporter.

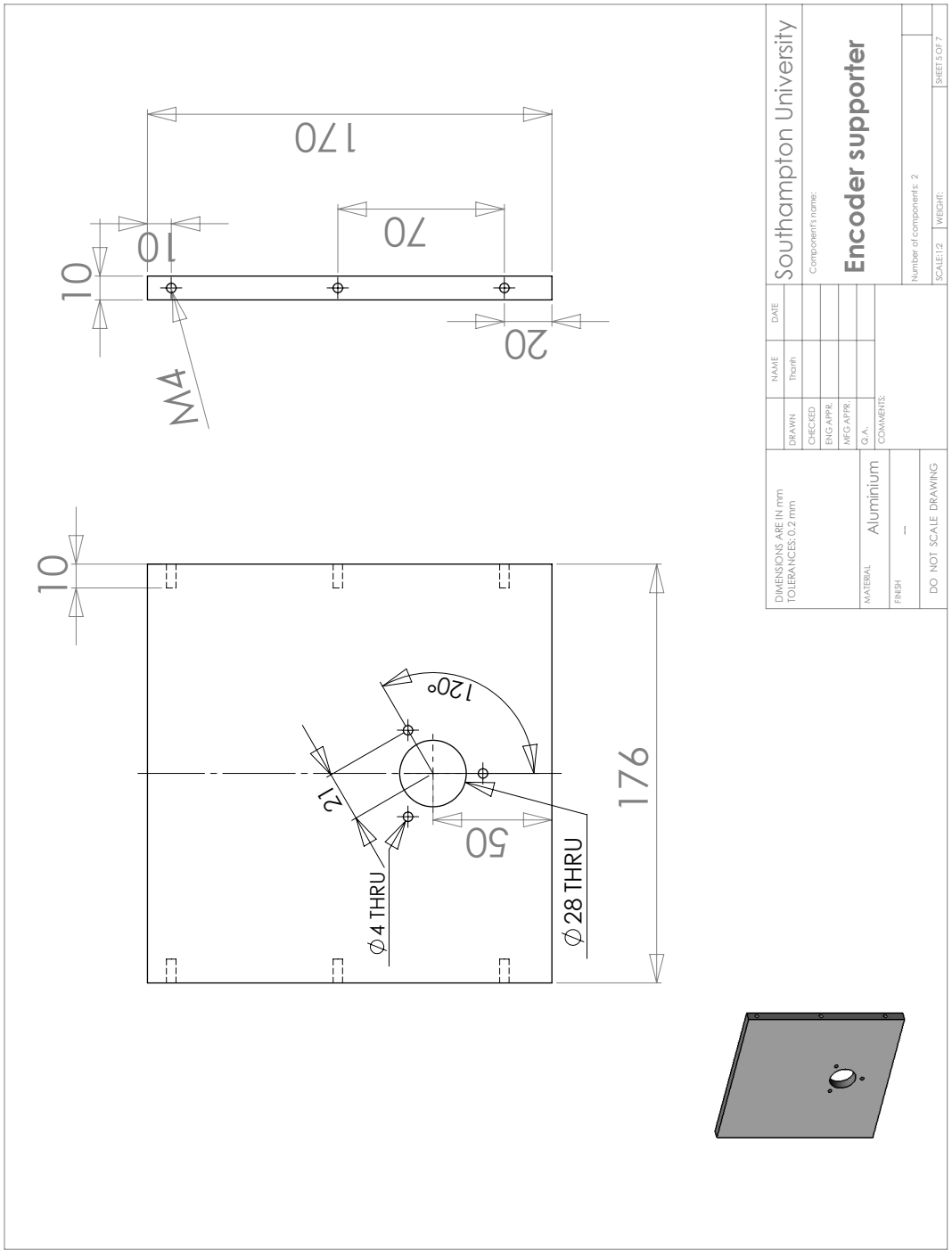


Figure A.8: Encoder supporter.

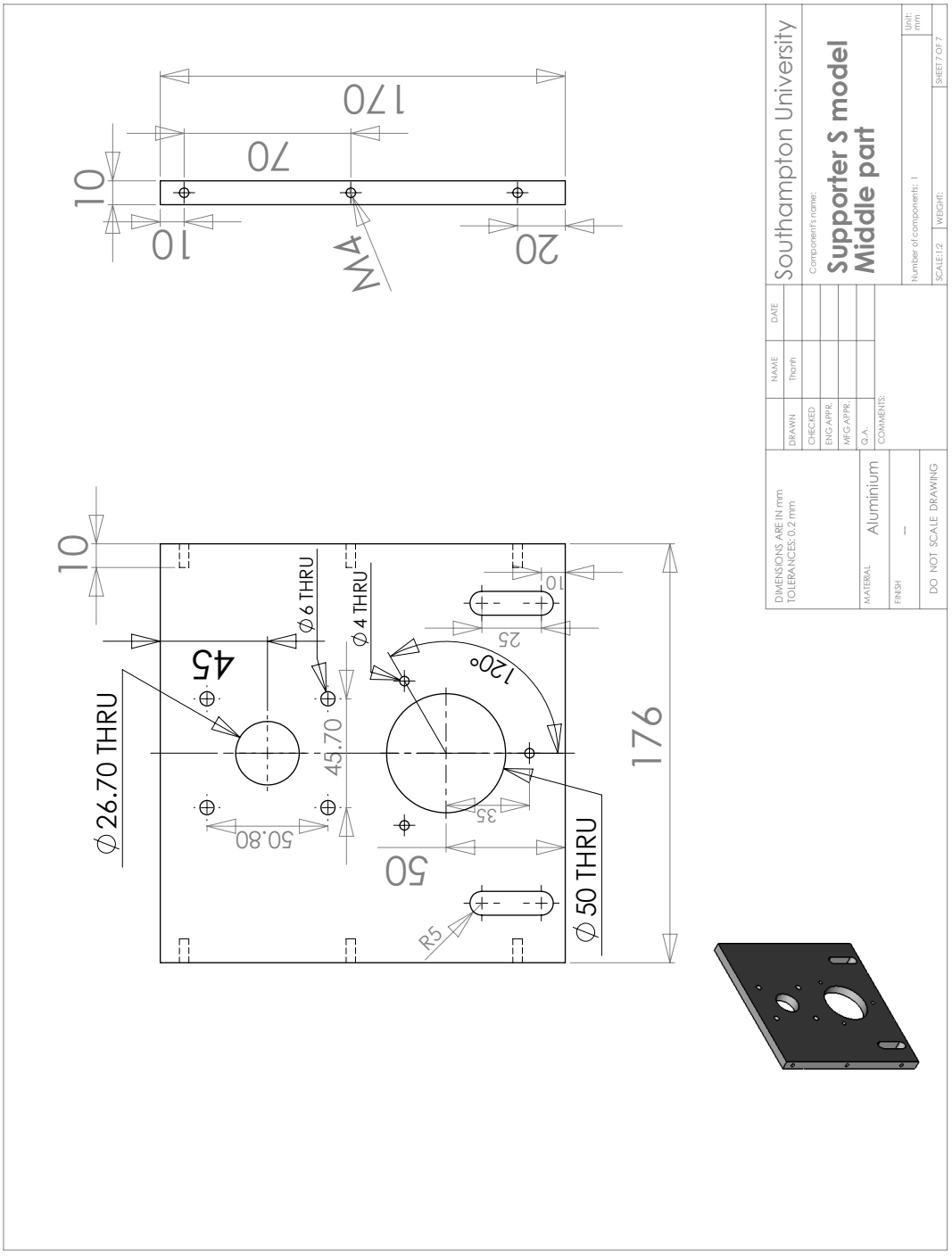


Figure A.9: Middle supporter for S dash-pot model.

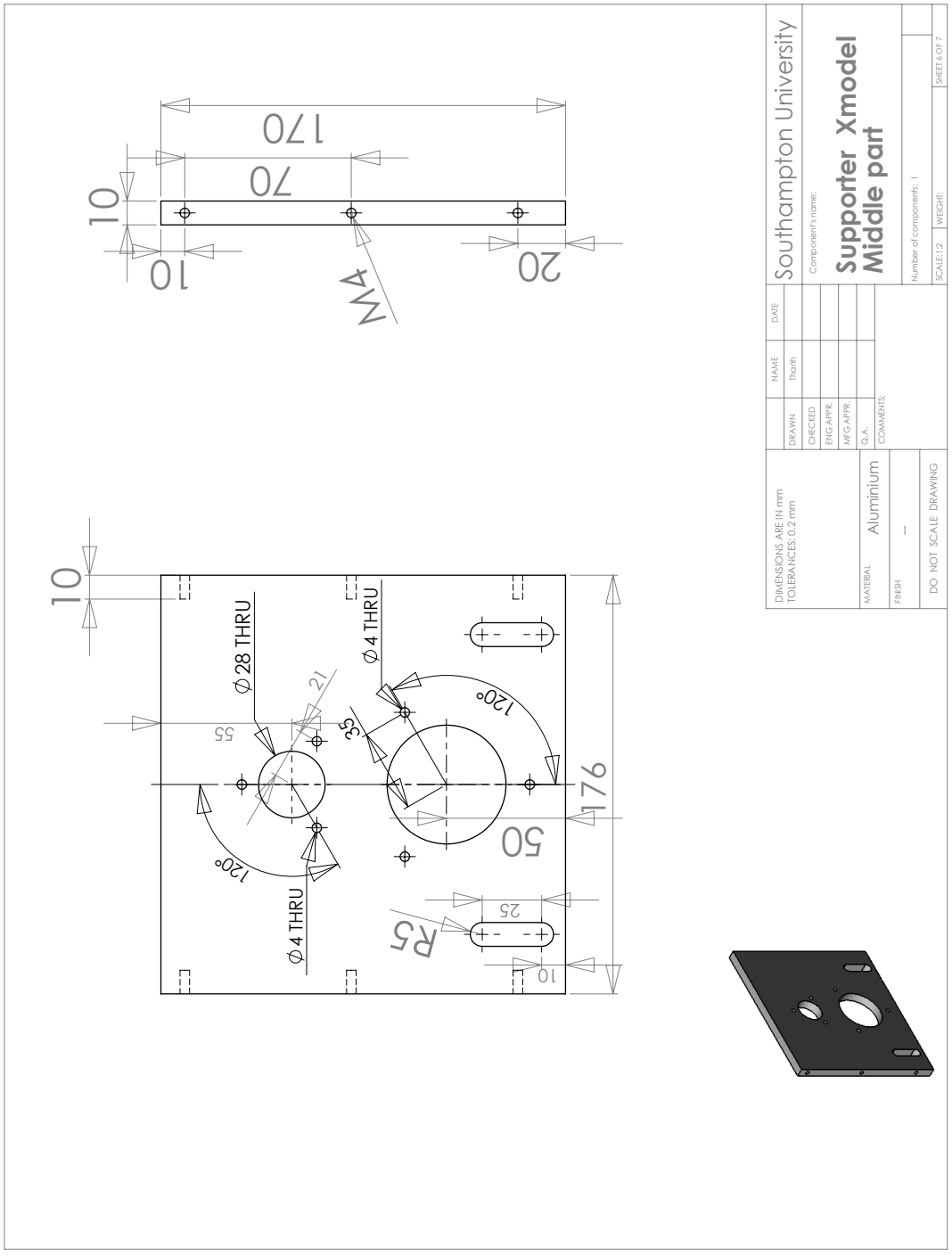


Figure A.10: Middle supporter for X dash-pot model.

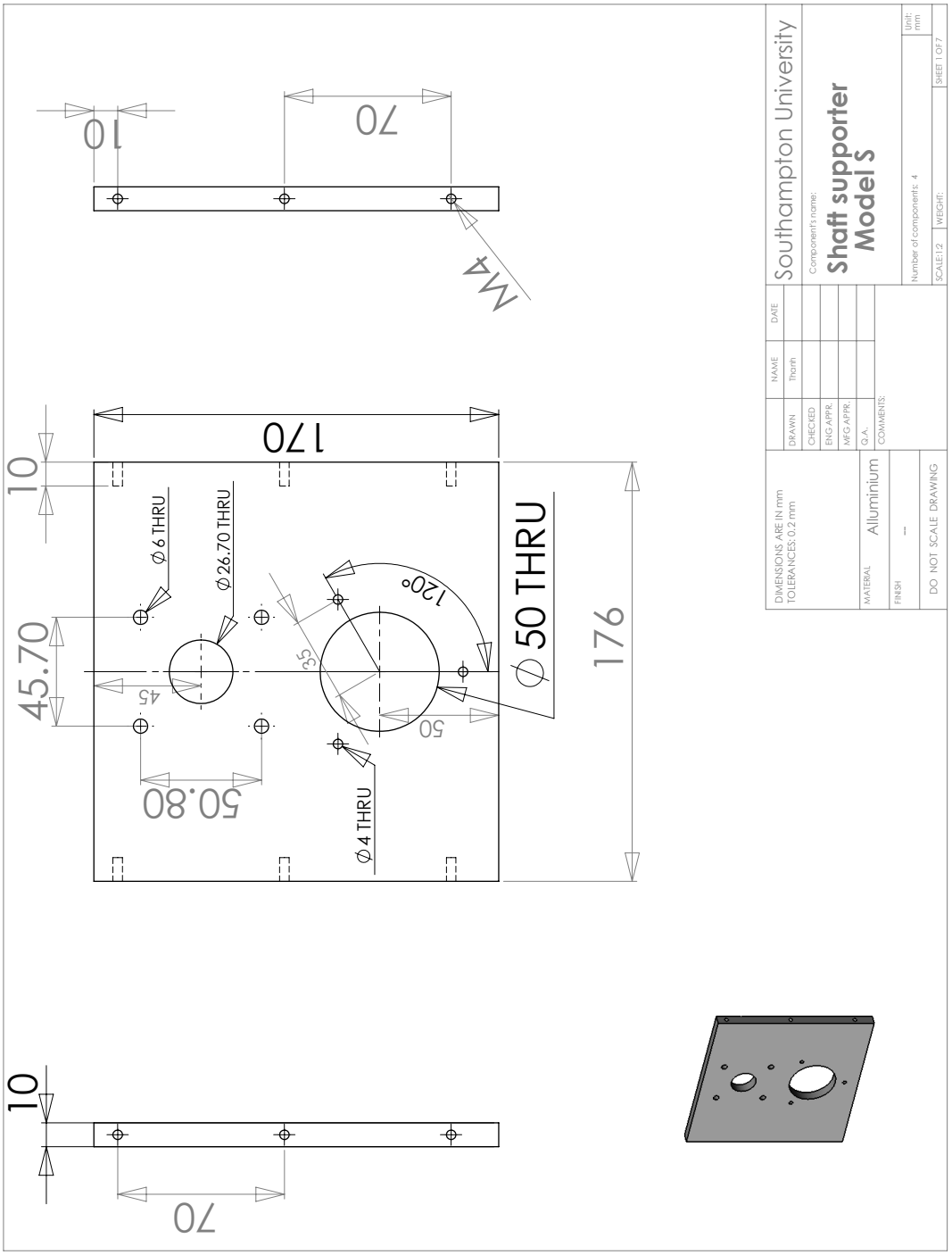


Figure A.11: Supporter for S dash-pot model.

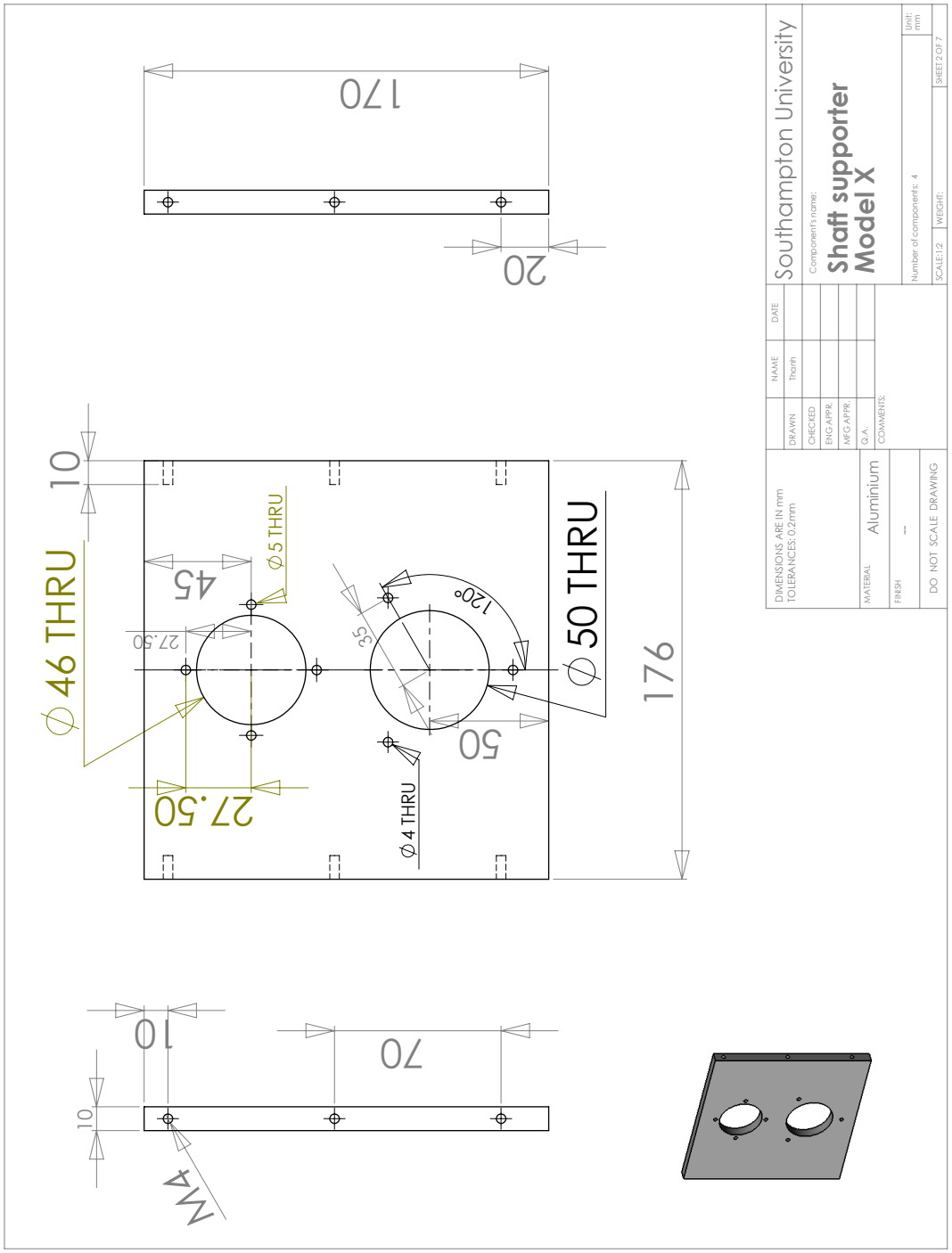


Figure A.12: Supporter for X dash-pot model.

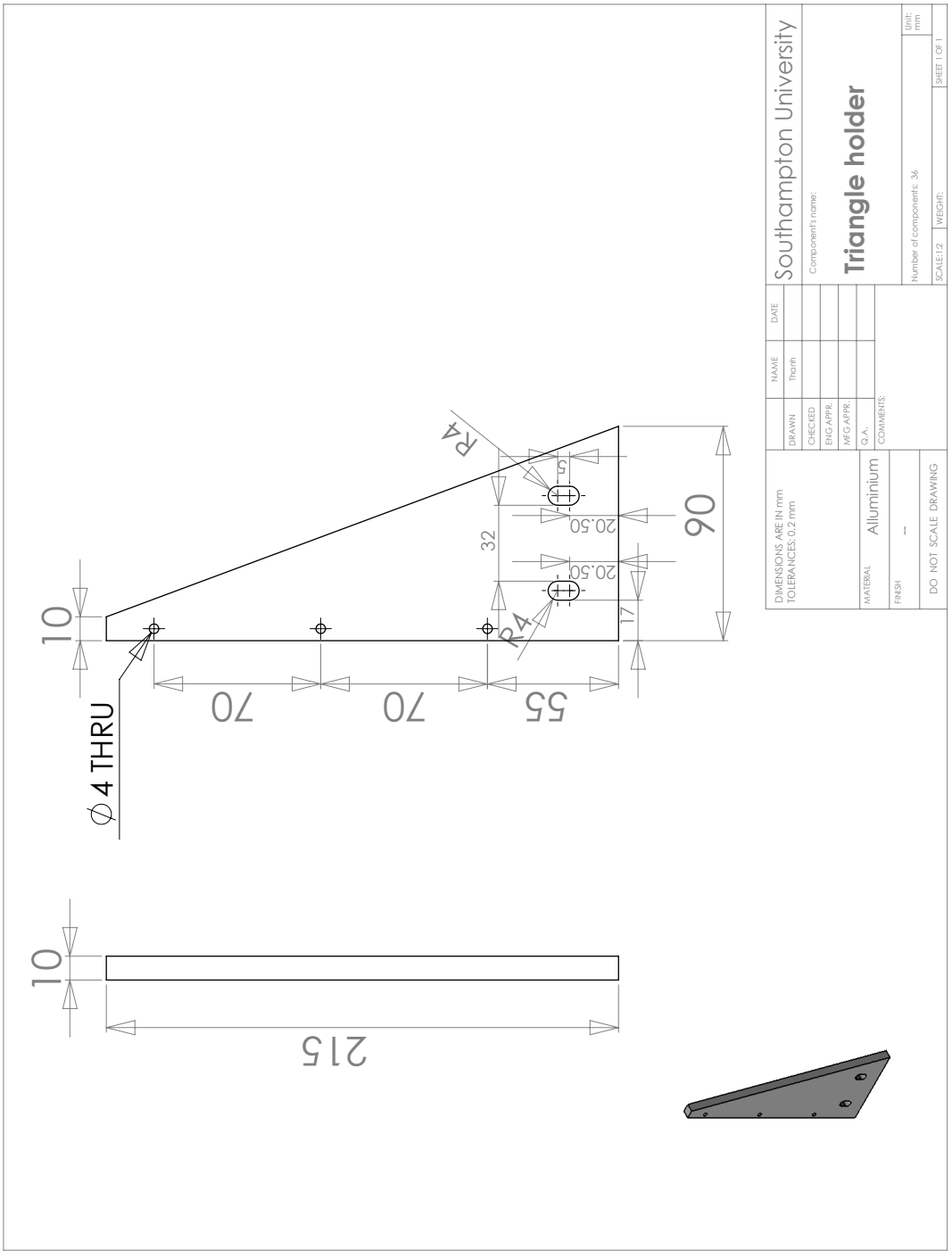


Figure A.13: Triangle holder.

N0	Name of components	Item reference	Dimension	Quantity	Price	Total	Company
1	Spring	Torsion spring	wire D 7mm, total length 140mm, turns: 15.5, average dameter: 70	1	46	46	Spring master
2	Spring	Torsion spring	wire D 7mm, total length 210mm, turns: 19.5, average dameter: 70	1	42	42	Spring master
3	Damper	Model S-1000cSt	damper const: 0.01	2	121.44	242.88	Kinetrol
4	Damper	Model X-1000cSt	damper const: 0125	2	42.62	85.24	Kinetrol
5	Flange bearing unit	755-334	bore size 20mm diameter	10	13.05	130.5	RS
6	Ball bearing for diff gear	409-0158	inner D: 20mm outer D: 42, width: 12	4	4.99	19.96	RS
7	Support beam	XCBM Lx88		4		97.64	Flexlink
8	Foot	XCFF 88 x 260		4		347.84	Flexlink
9	Base	XCBM L x 88 x 176		3		805.05	Flexlink
10	Support base beam	XCBM Lx88		2		112.11	Flexlink
11	Angle brackets	XCFA 88 A		12		98.52	Flexlink
12	taper bush for shaft (connected to diff gear)	184-599	20mm in diameter	2	6.42	12.84	RS
13	Belt drive for diff gear	475-0614	length 800mm width 16 pitch 5	2	8.2	16.4	RS
14	Pulley for diff gear and shaft	183-300	26mm width, max bore dia: 42mm	4	17.16	68.64	RS
15	Taper bush for diff gear	184-218	bore diameter: 35mm	2	6.44	12.88	RS
16	pulley for motor and shaft	183-580	max bore diameter 28 mm	4	19.81	79.24	RS
17	Taper bush for motor	183-007		2	4.45	8.9	RS
18	Taper bush for Shaft (Connect to motor)	183-760	Shaft diameter 20mm	2	4.45	8.9	RS
19	belt for motor	475-0490	length 900	2	7.57	15.14	RS
20	Encoder	291-4399	2500 pulse per revolution	3	202.38	607.14	RS
21	Spur gear for dashpot and shaft	AB1 1.25-60		8	68.18	545.44	HPC gear
22	Spur gear for encoder	AB1 1.25-52		4	57.72	230.88	HPC gear
23	Coupling for encoder	693-0812		2	15.67	31.34	RS
24	Coupling for shaft	693-0821	total length 56mm, bore size: 20mm OD: 51mm	2	15.67	31.34	RS
25	the hub plate	692-8128		2	28.1	56.2	RS
26	Belows coupling	693-2445		6	37.36	224.16	RS
27	Pre drilled shaft	489-1591	20mm diameter 1200mm length	0	111.77	0	RS
28	Ball bearing for tension belt drive	505-596	ID: 10mm, OD:30, width 14mm	2	20.8	41.6	RS

29	T-bolt	XLAT 17	90 each	2		0	Flexlink
30	Nuts	XLAN 8	90 each	2		0	Flexlink
31	Washer	BRB 8,4x16	90 each	2		0	Flexlink
32	Screw M4	158-3629	50 each	2	11	22	RS
33	Lead screw (For belt tension)	703-9383	Length 500mm, 10mm, 5mm lead	2	66.2	132.4	RS
34	Nuts for lead screw	275-490	50 screw	1	5.83	5.83	RS
35	Steel round shaft holders	CSH20-R	D=20mm	4	38.07	152.28	ondrives
36	DC motor drive	1220XRi	180V, 12.2A, 1.8Kw. Voltage supply 200-264V	1	450	450	Norwin
37	DC motor	DC motor	180V, 3000rpm , 1.1KW	1	482	482	Norwin
38	ABB inverter	ACS150		2	300	600	Inverter
39	block switch	2259377		1	£2.66	2.66	RS
40	block switch	2259383		1	£2.66	2.66	RS
41	start/stop swich	2259305		1	£11.45	11.45	RS
42	enclosure	4712599		1	£66.50	66.5	RS
43	chassis plate	4712072		1	£18.15	18.15	RS
44	contactor	4450982	7.5kW	1	£30.86	30.86	RS
45	6 way male conn	3003642		2	£5.24	10.48	RS
46	6 way hood	3003759		2	£7.32	14.64	RS
47	6 way female conn	3004207		2	£5.44	10.88	RS
48	6 way base	3003951		2	£6.92	13.84	RS
49	20mm conduit	603097		1	£81.50	81.5	RS
50	M20 type A fittings	603019		10	£2.12	21.2	RS
51	M20 lock nuts	2875531		10	£0.32	3.2	RS
52	DIN rail	467416		4	£3.03	12.12	RS
					Total	6163.43	

Figure A.15: Components of mechanical system

Appendix B

Device datasheets and measured data

B.1 Data collection for inverter and induction motor

Ts is the number of samples per second. TT is the trial time. Amp is the amplitude of the sine-wave applied to the system.

Ts = 1/1000	TT = 180	TT=40s	amp=5		TT=20s		TT=10s			TT=5s		TT=2s	
amp = 5													
Frequency	0.01	0.1	0.15	0.2	0.4	0.6	0.8	1	2	3	4	5	6
mag1	41.7814	21.8123	18.3086	15.6434	9.964	6.5405	4.1135	2.2233	-3.6066	-7.1418	-9.5977	-11.7089	-13.3966
phase1	-84.2	-84.6135	-84.8205	-85.044	-86.088	-86.94	-87.936	-89.1	-93.64	-98.8615	-102.24	-108.225	-112.536
mag2	41.7795	21.8129	18.3087	15.6436	9.965	6.5374	4.1185	2.2466	-3.6371	-7.1466	-9.6004	-11.5353	-13.3356
phase2	-84.211	-84.6225	-84.8295	-85.042	-85.968	-86.9832	-87.648	-88.975	-93.96	-98.8615	-102.24	-106.425	-111.672
mag3	41.891	21.8133	18.3089	15.6437	9.9654	6.5374	4.1133	2.232	-3.6422	-7.144	-9.6023	-11.5314	-13.3324
phase3	-84.231	-84.5865	-84.8205	-85.056	-85.968	-86.9832	-87.984	-89.01	-94.04	-98.6954	-103.68	-106.2	-111.888
mean mag	41.8173	21.81283	18.30873	15.64357	9.9648	6.538433	4.1151	2.233967	-3.62863	-7.14413	-9.60013	-11.5919	-13.3549
mean phase	-84.214	-84.6075	-84.8235	-85.0473	-86.008	-86.9688	-87.856	-89.0283	-93.88	-98.8061	-102.72	-106.95	-112.032

TT=2s,	amp = 1										
Frequency	7	8	9	10	12	14	16	18	20	25	30
mag1	-11.2854	-12.6452	-13.9802	-15.1418	-16.9687	-18.8609	-20.3001	-21.8467	-23.0887	-26.7147	-29.1563
phase1	-117.18	-121.577	-130.815	-140.4	-145.584	-161.7	-167.04	-183.06	-196.8	-230.087	-251.871
mag2	-11.2812	-12.6389	-13.944	-15.176	-16.9976	-18.7814	-20.2951	-21.7837	-23.1267	-26.7551	-27.563
phase2	-117.18	-120.549	-129.803	-145.8	-144.72	-157.92	-168.686	-179.82	-183.6	-227.739	-240.986
mag3	-11.2979	-12.6305	-13.9483	-15.137	-17.0024	-18.8858	-20.371	-21.8626	-23.0402	-26.7686	-26.987
phase3	-117.39	-121.783	-129.6	-141.3	-144.72	-160.86	-165.806	-184.275	-191.2	-229.696	-247.243
mean mag	-11.2882	-12.6382	-13.9575	-15.1516	-16.9896	-18.8427	-20.3221	-21.831	-23.0852	-26.7461	-27.9021
mean phase	-117.25	-121.303	-130.073	-142.5	-145.008	-160.16	-167.177	-182.385	-190.533	-229.174	-246.7

Figure B.1: Data collection from the inverter and the induction motor

B.2 Encoders

The encoder in the longer of the two induction motors is from the Muirhead Vatric company but this model is around 20 years old and this has been custom modified. Therefore there is no specific datasheet for this device. The only information obatianable is the wiring diagram shown in Table B.1. The encoder has 2000 PPR. The second

Table B.1: Muirhead Vatric encoder wiring diagram

No	Colour	Function
1	Green	$0V$
2	Blue	A
3	Yellow	B
4	Orange	Z
5	Brown	$\bar{A}$
6	white	$\bar{B}$
7	Purple	$\bar{Z}$
8	Red	$5V$
9	Case shield	$0V$

encoder is from the British Encoder company and this encoder has the same number of pulses (2000 PPR). The voltage applied is 5V. The wiring diagram is shown in Table B.2

Table B.2: British encoder wiring diagram

No	Colour	Function
1	Black	Com
2	white	$+VDC$
3	Brown	A
4	Yellow	$\bar{A}$
5	Red	B
6	Greem	$\bar{B}$
7	Orange	Z
8	Blue	$\bar{Z}$
9	Case shield	$0V$

B.3 Induction motors

The two induction motors are from a company called Eurodrive. All the parameters of the motors are from (Barton, 1999) and are given in the Table B.3

B.4 ABB inverter and setup parameters

The method for configuring the inverter is outlined in the manual document booklet. Table B.4 shows parameters which have been used configure the drive. The Hand/auto macro is used as 9902 are set at 5. Therefore the AI port is the frequency reference which corresponds to the speed reference since the speed of motor is proportional to the

Table B.3: Induction motor parameters

No	Name	Value	Unit
1	V_s	415	V
2	Frequency	50	Hz
3	Stator resistance	11	Ω
4	Rotor resistance	12	Ω
5	Stator self inductance	0.03	H
6	Rotor self inductance	0.0442	H
7	Mutual inductance	0.81	H
8	Number of poles	2	
9	Inertial	0.00427	kgm^2
10	Viscous friction	0.005	N_s/m
11	Dry friction	0.1	Nm

applied frequency. DI3, DI4,DI5 are not connected. DI1 is connected to +24V and DI2 is connected to a signal which stipulates the direction of the motor.

Table B.4: ABB inverter configuration in REM (remote) mode

No	Code	Value	Meaning
1	2202	0	Acceleration time
2	2203	0	Deceleration time
3	2008	$50Hz$	Maximum output frequency
4	1301	0	Minimum frequency refer to port AI
5	9905	$415V$	Motor nominal voltage
6	9906	$4.75A$	Motor Nominal current
7	9907	$50Hz$	Motor nominal frequency
8	9902	5	Using the mode 5 to control the inverter

Appendix C

Coding method

C.1 Frequency modeling

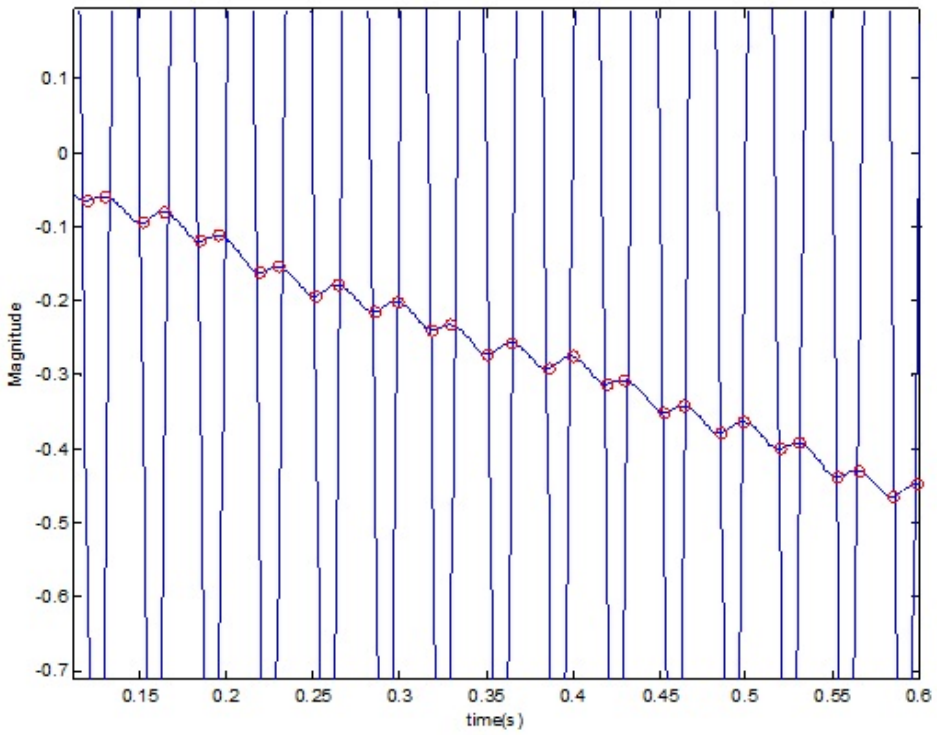


Figure C.1: The problem of drift in the frequency model.

References

Ahn, H. S., Chen, Y. Q., and Moore, K. L. (2007). Iterative learning control: brief survey and categorization. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 37:1099–1121.

Ahson, S. I., Hogg, B., and Pullman, R. (1979). Integrated control sysiem for turbogenerator desige by inverse nyquist array meod. In *IEEE Transactions on Power Apparatus and Systems*, volume PAS-98, pages 543–553.

Albertos, P. and Sala, A. (2004). *Multivariable control system.* Springer, first edition.

Amann, N., Owens, D. H., and Rogers, E. (1996a). Iterative learning control for discrete-time systems with exponential rate of convergence. *IEE Proceedings Control Theory Applications*, 143:217–224.

Amann, N., Owens, D. H., and Rogers, E. (1996b). Iterative learning control using optimal feedback and feedforward actions. *International Journal of Control*, 65:277–293.

Amann, N., Owens, D. H., and Rogers, E. (1998). Predictive optimal iterative learning control. *International Journal of Control*, 69:203–226.

Arimoto, S. and Kawamura, F. M. S. (1988). Realisation of robot motion based on a learning method. In *IEEE Transactions on System, Man, and Cybernetics*, volume 18, pages 126–133.

Arimoto, S., Kawamura, S., and Miyazaki, F. (1984). Bettering operation of dynamic system by learning: a new control theory for servomechanism or mechatronics systems. In *Proceedings of 23rd Conference on Decision and Control*, volume 23, pages 1064–1069.

Arimoto, S., Kawamura, S., Miyazaki, F., and Tamaki, S. (1985). Learning control theory for dynamical systems. In *Proceedings of 24rd Conference on Decision and Control*, volume 24, pages 1375–1380.

Avrachenkov, K. E. (1998). Iterative learning control based on quasi-newton methods. In *IEEE Conference on Decision and Control*, volume 1, pages 170–174.

Barton, A., Lewin, P. L., and Brown, D. (2000). Practical implementation of a real-time iterative learning position controller. *International Journal of Control*, 73(10):992–999.

Barton, A. D. (Jan,1999). Control of high speed chain conveyor systems. *PhD thesis, School of ECS, University of Southampton*.

Barton, K. L. and Alleyne, A. G. (2011). A norm optimal approach to time-varying ILC with application to a multi-axis robotic testbed. *IEEE Transactions on Control System Technology*, 19(1):166–180.

Barton, K. L., van de Wijdeven, J., Alleyne, A., Bosgra, O., and Steinbuch, M. (2008). Norm optimal cross-coupled iterative learning control. *47th IEEE Conference on Decision and Control*, pages 3020–3025.

Birk, W. and Medvedev, A. (2003). A note on gramian-based interaction measures. In *In Proceedings of European Control Conference*.

Bristol, E. H. (1966). On a new measure of interaction for multivariable process control. *IEEE Transactions on Automatic Control*, 11(1):133–134.

Bristol, E. H. (1978). Recent results on interactions for multivariable process control. *AIChE Anual Meeting, Chicago, IL*.

Bristow, D. A. (2008a). Frequency domain analysis and design of iterative learning control for systems with stochastic disturbances. In *American Control Conference*, pages 3901–3907.

Bristow, D. A. (2008b). Weighting matrix design for robust monotonic convergence in norm optimal iterative learning control. In *American Control Conference*, pages 4554–4560.

Bristow, D. A. (2010). Optimal iteration-varying iterative learning control for systems with stochastic disturbances. In *American Control Conference*, pages 1296–1301.

Bristow, D. A. and Alleyne, A. G. (2003). A manufacturing system for microscale robotic deposition. In *American Control Conference*, volume 3, pages 2620–2625.

Bristow, D. A., Tharayil, M., and Alleyne, A. G. (2006). A survey of iterative learning control a learning-based method for high-performance tracking control. *IEEE Control Systems Magazine 26*, pages 96–114.

Buchheit, K., Pandit, M., and Befort, M. (1994). Optimal iterative learning control of an extrusion plant. In *International Conference on Control*, volume 1, pages 652–657.

Butcher, M. and Karimi, A. (2010). Linear parameter-varying iterative learning control with application to a linear motor system. *IEEE/ASME Transactions on Mechantronics*, 15:412–420.

Butcher, M., Karimi, A., and Longchamp, R. (2008a). On the consistency of certain identification methods for linear parameter varying systems. In *Proceedings of the 17th IFAC World Congress*.

Butcher, M., Karimi, A., and Longchamp, R. (2008b). A statistical analysis of certain iterative learning control algorithms. *International Journal of Control*, 81(1):156–166.

Cai, Z. (Feb, 2009). Iterative learning control: Algorithm development and experimental benchmarking. *PhD thesis, School of ECS, University of Southampton*.

Cai, Z., Bristow, D. A., Rogers, E., and Freeman, C. T. (2011a). Experimental implementation of iterative learning control for processes with stochastic disturbances. In *IEEE International Symposium on Intelligent Control*, pages 406 – 411.

Cai, Z., Freeman, C. T., Lewin, P. L., and Rogers, E. (2008a). Experimental comparison of stochastic iterative learning control algorithms. In *American Control Conference*, pages 4548– 4553.

Cai, Z., Freeman, C. T., Lewin, P. L., and Rogers, E. (2008b). Iterative learning control for a non-minimum phase plant based on a reference shift algorithm. *Control Engineering Practice*, 16:633–643.

Cai, Z., Freeman, C. T., Rogers, E., and Lewin, P. L. (2007). Reference shift iterative learning control for a non-minimum phase plant. In *American Control Conference*, pages 558–563.

Cai, Z., Tong, D., Freeman, C. T., and Rogers, E. (2011b). Application of newton-method based ilc to 3D stroke rehabilitation. In *IFAC World Congress*, pages 4851–4856.

Cai, Z., Tong, D., Freeman, C. T., and Rogers, E. (2011c). Application of newton-method based ILC to 3d stroke rehabilitation. In *18th IFAC World Congress*.

Chen, D. and Seborg, D. E. (2001). Multiloop PI/PID controller design based on gershgorin bands. In *American Control Conference*, volume 5, pages 4122–4127.

Chien, I. L., Huang, H. P., and Yang, J. C. (1999). A simple multi-loop tuning method for pid controllers with no proportional kick. *Industrial and Engineering Chemistry Research*, 38:1456–1468.

Chu, B. and Owens, D. H. (2009). Accelerated norm-optimal iterative learning control algorithms using successive projection. *International Journal of Control*, 82(8):1469–1484.

Chu, B. and Owens, D. H. (2010). Iterative learning control for constrained linear systems. *International Journal of Control*, 83(7):1397–1413.

Close, C. M. and Frederick, D. K. (1995). *Modeling and analysis of dynamic systems*. Wiley, second edition.

Conley, A. and Salgado, M. E. (2000). Gramian based interaction measure. In *Proceedings of the 39th IEEE decision and Control*, volume 5, pages 5020–5022.

Craig, J. J. (1984). Adaptive control of manipulators through repeated trials. In *American Control Conference*, pages 1566–1573.

Cryer, B. W., Nawrocki, P. E., and Lund, R. A. (1976). A road simulation system for heavy duty vehicles. In *Society of Automative Engineers*.

Davies, I. L., Freeman, C. T., Lewin, P. L., Rogers, E., and Owens, D. H. (2008). Newton method based iterative learning control of the upper limb. In *Proceedings of the American Control Conference*, pages 3887–3892.

Dinh, T. V., Freeman, C. T., and Lewin, P. (2012a). Assessment of gradient-based point-to-point ILC for MIMO systems with varying interaction. In *IEEE Multi-Conference on Systems and Control*, pages 1220–1225.

Dinh, T. V., Freeman, C. T., and Lewin, P. (2012b). Convergence and robustness of a point-to-point iterative learning control algorithm. In *IEEE Conference on Decision and Control*, pages 4678–4683.

Dinh, T. V., Freeman, C. T., and Lewin, P. (2012c). Development of a multivariable test facility for the evaluation of iterative learning controllers. In *American Control Conference*, pages 621–626.

Donkers, T., van de Wijdeven, J., and Bosgra, O. (2008). Robustness against model uncertainties of norm optimal iterative learning control. In *American Control Conference*, pages 4561–4566.

Dorf, R. C. and Bishop, R. H. (2005). *Modern control systems*. Pearson Prentice Hall, tenth edition.

Dutton, K., Thompson, S., and Barraclough, B. (1997). *The art of control engineering*. Addison-Wesley, first edition.

Elci, H., Longman, R. W., Phan, M., Juang, J., and Ugoletti, R. (1994). Discrete frequency based learning control for precision motion control. In *IEEE International Conference on System, Man, and Cybernetics*, volume 3, pages 2767–2773.

Ford, M. P. and Daly, K. C. (1979). Dominance improvement by pseudo decoupling. *Proceedings of IEE*, 126:1316–1320.

Freeman, C. T. (2004). Experimental evaluation of iterative learning control performance for non-minimum phase plants. *PhD thesis, School of ECS, University of Southampton*.

Freeman, C. T. (2011). Constrained point-to-point iterative learning control. In *18th IFAC World Congress*, pages 3611–3616.

Freeman, C. T. (2012). Constrained point-to-point iterative learning control with experimental verification. *Control Engineering Practice*, 20(5):489–498.

Freeman, C. T., Davies, I., Lewin, P. L., and Rogers, E. (2008). Iterative learning control of upper limb reaching using functional electrical stimulation. In *The 17th IFAC World Congress*, pages 13444–13449.

Freeman, C. T., Hughes, A. M., Burridge, J. H., Chappel, P. H., Lewin, P. L., and Rogers, E. (2009a). Iterative learning control of FES applied to the upper extremity for rehabilitation. *Control Engineering Practice*, 13:368–381.

Freeman, C. T., Hughes, A. M., Burridge, J. H., Chappell, P. H., Lewin, P. L., and Rogers, E. (2007a). An experimental facility for the application of iterative learning control as an intervention aid to stroke rehabilitation. *Measurement and Control*, 40:20–23.

Freeman, C. T., Hughes, A. M., Burridge, J. H., Chappell, P. H., Lewin, P. L., and Rogers, E. (2007b). An experimental facility using functional electrical stimulation for stroke rehabilitation of the upper limb. In *IEEE International Conference on Rehabilitation Robotics*, pages 393–400.

Freeman, C. T., Hughes, A. M., Burridge, J. H., Chappell, P. H., Lewin, P. L., and Rogers, E. (2009b). A model of the upper extremity using fes for stroke rehabilitation. *Journal of Biomechanical Engineering*, 131:031011–12.

Freeman, C. T., Hughes, A. M., Burridge, J. H., Chappell, P. H., Lewin, P. L., and Rogers, E. (2009c). A robotic workstation for stroke rehabilitation of the upper extremity using fes. *Medical Engineering and Physics*, 31:364–373.

Freeman, C. T., Lewin, P., and Rogers, E. (2007c). Further results on the experimental evaluation of iterative learning control algorithms for non-minimum phase plants. *International Journal of Control*, 80:569–582.

Freeman, C. T., Lewin, P. L., E. Rogers, E., and Owens, D. H. (2010). Iterative learning control of the redundant upper limb for rehabilitation. In *American Control Conference*, pages 1278–1283.

Freeman, C. T., Lewin, P. L., and Rogers, E. (May, 2005). Experimental evaluation of iterative learning control algorithms for non-minimum phase plants. *International Journal of Control*, 78:826–846.

Freeman, C. T., Rogers, E., Hughes, A. M., Burridge, H. H., and Meadmore, K. L. (2012). Iterative learning control in healthcare electrical stimulation and robotic-assisted upper limb stroke rehabilitation. *IEEE Control Systems Magazine*, 32:18–43.

Freeman, C. T. and Tan, Y. (2011). Point-to-point iterative learning control with mixed constraints. In *American Control Conference*, pages 3657–3662.

Freeman, C. T. and Tan, Y. (2013). Iterative learning control with mixed constraints for point-to-point tracking. *IEEE Transactions on Control Systems Technology*, 21:604–616.

Freeman, C. T., Tong, D., Meadmore, K., Cai, Z., Rogers, E., Hughes, A. M., and Burridge, J. H. (2011a). Phase-lead iterative learning control algorithms for functional electrical stimulation based stroke rehabilitation. *Proceedings of the Institution of Mechanical Engineers - Part I: Journal of Systems and Control Engineering*, 225:850–589.

Freeman, C. T., Tong, D., Meadmore, K. L., Cai, Z., Rogers, E., Hughes, A. M., and Burridge, J. H. (2011b). Phase-lead iterative learning control algorithms for functional electrical stimulation based stroke rehabilitation. *Proceedings of The Institution of Mechanical Engineers - part I: Journal of Systems and Control Engineering*, 225(4):255–262.

French, M., Munde, G., Rogers, E., and Owens, D. H. (1999). Recent developments in adaptive iterative learning control. In *Proceedings of the 38th Conference on Decision and Control*, pages 264–26, Phoenix, Arizona, USA.

French, M. and Rogers, E. (2000). Nonlinear iterative learning by an adaptive lyapunov technique. *International Journal of Control*, 73:840–850.

Furuta, K. and Yamakita, M. (1987). The design of learning control systems for multivariable systems. *In Proceedings of the IEEE International Symposium on Intelligent Control*, pages 371–376.

Gao, F., Yang, Y., and Shao, C. (2001). Robust iterative learning control with applications to injection molding process. *Chemical Engineering Science*, 56:7025–7034.

Garden, M. (1971). Learning control of actuators in control systems. In *U.S*, page Patent 3555252.

Garimella, S. and Srinivasan, K. (1998). Application of iterative learning control to coil-to-coil control in rolling. *IEEE Transactions on Control Systems Technology*, 6:281–293.

Golub, G. and Loan, C. (1996). *Matrix computations*. John Hopings Press, third edition.

Greenwood, D. T. (2003). *Advanced dynamics*. Cambridge University Press, first edition.

Grosdidier, P., Morari, M., and Holt, B. R. (1985). Closed-loop properties from steady-state gain information. *Industrial and Engineering Chemistry, Fundamentals*, 24:221–235.

Halvarsson, B. (2010). *Interaction analysis in multivariable control systems: Application to bioreactor for nitrogen removal*. Phd thesis, Department of Information Technology, Uppsala University, Sweden.

Hannah, J. and Stephens, R. C. (1963). *Mechanics of machines.* Edward Arnold, first edition.

Hätönen, J., Harte, T., Owens, D., Ratcliffe, J., Lewin, P., and Rogers, E. (2003). A new robust iterative learning control algorithm for application on gantry robot. *Emerging Technologies and Factory Automation*, 2:305–312.

Hätönen, J. and Owens, D. H. (2004). Hydraulic servo system command shaping using iterative learning control. In *Proceedings of The UKACC Conference*, pages 117–121.

Hätönen, J., Owens, D. H., Lewin, P., and Rogers, E. (2004). Robustness analysis of a gradient-based repetitive algorithm for discrete-time systems. *IEEE Conference on Decision and Control*, 2:1301–1306.

Hätönen, J. J. (2004). *Issues of algebra and optimality in iterative learning control.* Phd thesis, Department of Process and Environmental Engineering, University of Oulu.

Haurani, A., Taha, O., Michalska, H., and Boulet, B. (2001). Multivariable control of a paper coloring process: a case study. In *American Control Conference*, volume 3, pages 2210–2215.

Hawkins, D. J. (1972). Pseudo diagonalisation and the inverse Nyquist array method. *Proceedings of IEE*, 119:337–342.

Hladowski, L., Galkowski, K., Cai, Z., Rogers, E., Freeman, C., and Lewin, P. (2010). Experimentally supported 2d systems based iterative learning control law design for error convergence and performance. *Control Engineering Practice*, 18:339–348.

Holmes, R. (1977). *The characteristics of mechanical engineering systems.* Pergamon Press, first edition.

Hovd, M. and Skogestad, S. (1992). Simple frequency-dependent tools for control system analysis, structure selection and design. *Automatica*, pages 989–996; No. 5.

Huang, Y. C., Chan, M., Hsin, Y. P., and Ko, C. C. (2003). Use of PID and iterative learning controls on improving intra-oral hydraulic loading system of dental implants. *IEEE International Symposium on Intelligent Control*, pages 63–68.

Hughes, A. M., Burridge, J. H., Freeman, C. T., Chappell, P. H., Lewin, P. L., Donovan-Hall, M., and Rogers, E. (2009a). Efficacy of iterative learning control for stroke rehabilitation. *Progress in Neurology and Psychiatry*, 13:16–20.

Hughes, A. M., Freeman, C. T., Burridge, J. H., Chappell, P., Lewin, P., and Rogers, E. (2009b). Clinical effectiveness and patient perceptions of an ilc mediated by ES system using a robotic workstation. In *European Stroke Conference.*

Hughes, A. M., Freeman, C. T., Burridge, J. H., Chappell, P. H., Lewin, P. L., and Rogers, E. (2009c). Feasibility of iterative learning control mediated by functional electrical stimulation for reaching after stroke. *Journal of Neurorehabilitation and Neural Repair*, 23:559–568.

Kawamura, S., Miyazaki, F., and Arimoto, S. (1985). Applications of learning method for dynamic control of robot manipulators. In *IEEE Conference on Decision and Control*, volume 24, pages 1381 – 1386.

Kawamura, S. and Sakagami, N. (2002). Analysis on dynamics of underwater robot manipulators based on iterative learning control and time-scale transformation. In *IEEE International Conference on Robotics and Automation*, volume 2, pages 1088–1094.

Kichhoff, S., Chmidt, S., Lichtenberg, G., and Werner, H. (2008). An iterative learning algorithm for control of an accelerator based free electron laser. *Proceedings of the 47th IEEE Conference on Decision and Control*, pages 3032–3037.

Kim, D. and Kim, S. (January, 1996). An iterative learning control method with application for CNC machine tools. *IEEE Transaction on Industry Applications*, 32:66–72.

Kincaida, D. and Cheney, W. (2002). *Numerical analysis: mathematics of scientific computing*. Thomson Learning, third edition.

Kinnaert, M. (1995). Interaction measures and pairing of controlled and manipulated variables for multiple-input multiple-output systems: A survey. *Journal A*, 4:15–23.

Kinoshita, K., Sogo, T., and Adachi, N. (2002). Adjoint-type iterative learning control for single-link flexible arm. *15th Triennial World Congress*.

Kinosita, K., Sogo, T., and Adachi, N. (2002). Iterative learning control using adjoint systems and stable inversion. *Asian Journal of Control*, 4:60–67.

Kwakernaak, H. (1993). Robust control and h_∞ optimization - tutorial paper. *Automatica*, 29:255–273.

Kwakernaak, H. (2002). Mixed sensitivity design an aerospace case study. *IFAC 15th Triennial World Congress*, 15.

Le, F., Markovsky, I., Freeman, C. T., and Rogers, E. (2010). Identification of electrically stimulated muscle models of stroke patients. *Control Engineering Practice*, 18:396–407.

Le, F., Markovsky, I., Freeman, C. T., and Rogers, E. (2012). Recursive identification of hammerstein systems with application to electrically stimulated muscle. *Control Engineering Practice*, 20:386–396.

Lee, K. S., Bang, S. H., Yi, S., Son, J. S., and Yoon, S. C. (1996). Iterative learning control of heat-up phase for a batch polymerization reactor. *Journal of Process Control*, 6:255–262.

Li, Y. H. . Z. X. . J. (2010). Missile PID controller parameter tuning based on iterative learning control. In *International Conference on Signal Processing Systems*, volume 3, pages 689–692.

Lin, S. Y., Fleming, J. G., Hetherington, D. L., Smith, B. K., Biswas, R., Ho, K. M., Sigalas, M. M., Zubruzycki, W., Kurth, S. R., and Bur, J. (1998). A three-dimensional photonic crystals operating at infrared wavelengths. *Nature*, 394:251–253.

Lin, T., Owens, D. H., and Hätönen, J. (2006). Newton method based iterative learning control for discrete non-linear systems. *International Journal of Control*, 79:1263–1276.

Luyben, W. L. (1986). Simple method for tuning SISO controllers in multivariable systems. *Industrial Engineering Chemistry Process Design and Development*, 25:654–660.

Maciejowski, J. M. (1989). *Multivariable feedback design*. Addison Wesley, first edition.

Marion, J. B. and Thornton, S. T. (1995). *Classical dynamics of particles and systems*. Harcourt Brace, fourth edition.

Mi, C., Lin, H., and Zhang, Y. (2005). Iterative learning control of antilock braking of electric and hybrid vehicles. *IEEE Transactions on Vehicular Technology*, 54:486–494.

Mishra, S. and Tomizuka, M. (2005). An optimization-based approach for design of iterative learning controllers with accelerated rates of convergence. In *44th IEEE Conference in Decision and Control*, pages 2427–2432.

Mollov, S., Babuška, R., and Verbruggen, H. B. (2001). Analysis of interactions in mimo takagi-sugeno fuzzy models. In *IEEE International Fuzzy Systems Conference*.

Morrison, J. and Crossland, B. (1970). *An introduction to the mechanics of machines*. Longman, s.i units edition.

Munro, N. (1976). Application of the inverse Nyquist array design method. In *IEEE Conference on Decision and Control*, volume 15, pages 348–353.

Munro, N. (1990). Multivariable control applications: turbine and chemical plant examples. In *IEE Colloquium on Successful Industrial Applications of Multivariable Analysis*, pages 1–4.

Munro, N. (October, 1972). Multivariable systems design using the inverse Nyquist array. *Computer-Aided Design*, 4:222–227.

Norrlöf, M. (2002). An adaptive iterative learning control algorithm with experiments on an industrial robot. *IEEE Transactions on Robotics and Automation*, 18(2):245–251.

O'Dwyer and Aidan (2006). *Handbook of PI and PID Controller Tuning Rules*. Imperial College Press, second edition.

Ogoshi, R., Sogo, T., and Adachi, N. (2002). Adjoint-type iterative learning control for nonlinear non-minimum phase system - application to planar model of a helicopter-. *Proceedings of the 41st SICE Annual Conference*, 3:1547–1550.

Ogunnaike, B. A., Lemaire, J. P., Morari, M., and Ray, W. H. (1983). Advanced multivariable control of a pilot-plant distillation column. *American Institute of Chemical Engineers Journal*, 29:632–640.

Ogunnaike, B. A. and Ray, W. H. (1979). Multivariable controller design for linear systems having multiple time delays. *American Institute of Chemical Engineers Journal*, 25:1043–1057.

O'Reilley, J. and Leithead, W. E. (Mar, 1991). Multivariable control by individual channel design. *International Journal of Control*, 54:1–46.

Owens, D. H., Freeman, C. T., and Dinh, T. V. (2012). Norm optimal iterative learning control with intermediate point weighting - Theory, algorithms and experimental evaluation. *IEEE Transactions on Control System Technology*, 21.

Owens, D. H., Hätönen, J., and Daley, S. (2007). Robust gradient-based iterative learning control. *IEEE International Conference on Networking, Sensing and Control*, pages 163–168.

Owens, D. H., Hätönen, J. J., and Daley, S. (2009). Robust monotone gradient-based discrete-time iterative learning control. *International Journal of Robust and Nonlinear Control*, 19(6):634–661.

Palm, W. J. (2007). *Mechanical vibration*. John Wiley and Sons Inc, first edition.

Pandit, M. and Buchheit, K. H. (1999). Optimizing iterative learning control of cyclic production processes with application to extruders. *IEEE Transactions on Control Systems Technology*, 7:382–390.

Park, K. H., Bien, Z., and Hwang, D. H. (1998). Design of an iterative learning controller for a class of linear dynamic systems with time delay. *IEE Proceedings Control Theory Applications*, 145:507–512.

Polyanin, A. D. and Zaitsev, V. F. (2005). *Hand book of exact solutions for ordinary differential equations*. CRC company, second edition.

Rashid, M. H. (1993). *Power electronics circuits, devices, and applications*. Prentice Hall, second edition.

Ratcliffe, J. D., Lewin, P. L., Rogers, E., Hätönen, J. J., and Owens, D. H. (2006). Norm-optimal iterative learning control applied to gantry robots for automation applications. *IEEE Transactions On Robotics*, 22(6):1303–1307.

Ratcliffe, J. D., van Duinkerken, L., Lewin, P. L., Rogers, E., and Hatonen, J. J. (2005). Fast norm-optimal iterative learning control for industrial applications. In *American Control Conference*, volume 3, pages 1951–1956.

Rogers, E., Owens, D. H., Werner, H., Freeman, C. T., Lewin, P. L., Kichhoff, S., Chmidt, S., and Lichtenberg, G. (2010). Norm optimal iterative learning control with application to problems in accelerator based free electron lasers and rehabilitation robotics. *European Journal of Control*, 16:497–524.

Roover, D. D. (1996). Synthesis of a robust iterative learning controller using an h_∞ approach. In *IEEE Conference on Decision and Control*, volume 3, pages 3044–3049.

Roover, D. D. (1997). Motion control of a wafer stage: A design approach for speeding up ic production. *PhD thesis; Delft University; The Netherlands.*

Roover, D. D. and Bosgra, O. H. (2000). Synthesis of robust multivariable iterative learning controllers with application to a wafer stage motion system. *International Journal of Control*, 73:968–979.

Rosenbrock, H. H. (1969). Design of multivariable control systems using the inverse Nyquist array. In *Proceedings of the Institution of Electrical Engineers*, volume 116, pages 1929–1936.

Rosenbrock, H. H. (1974). *Computer-aided control system design.* Academic Press INC, first edition.

Saab, S. S. (2001a). A discrete-time stochastic learning control algorithm. *IEEE Transactions On Automatic Control*, 46(6):877–887.

Saab, S. S. (2001b). On a discrete-time stochastic learning control algorithm. *IEEE Transactions On Automatic Control*, 46(8):1333–1335.

Saab, S. S. (2003). Stochastic p-type/d-type iterative learning control algorithms. *International Journal of Control*, 76(2):139–148.

Sakagami, N., Inoue, M., and Kawamura, S. (2002). Theoretical and experimental studies on iterative learning control for underwater robots. *Proceedings of the twelfth International Offshore and Polar Engineering Conference*, pages 296–300.

Sakagami, N. and Kawamura, S. (2003). Time optimal control for underwater robot manipulators based on iterative learning control and time-scale transformation. *Proceedings OCEANS*, 3:1180–1186.

Schindele, D. and Aschemann, H. (2011). ILC for a fast linear axis driven by pneumatic muscle actuators. In *IEEE International Conference on Mechatronics*, pages 967–972.

Shamma, J. and Athans, M. (1998). Guaranteed properties of gain scheduled control for linear parameter-varying plants. *Automatica*, 27:559–564.

Shinskey, F. G. (1988). *Process control systems*. MCGraw-Hill, third edition.

Skogestad, S. and Postlethwaite, I. (1996). *Multivariable feedback control analysis and design multivariable control systems*. Wiley, first edition.

Soska, A., Freeman, C. T., and Rogers, E. (2012). ILC for FES-based stroke rehabilitation of hand and wrist. In *IEEE Multiconference on Systems and Control*, pages 1267 – 1272.

Srivastava, M., Srivastava, M. C., and Bhatnagar, S. (2009). *Control system*. Tata McGraw-Hill, first edition.

Stewart, G. W. (1993). On the early history of the singular value decomposition. *SIAM review*, 35:551–566.

Tan, K. K., Huang, S. N., and Zhao, S. (2003). A novel predictive and iterative learning control algorithm. *Control and Intelligent System*, 31:1–9.

Tan, K. K. and Tang, J. C. (2002). Learning-enhanced PI control of ram velocity in injection molding machines. *Engineering Application of Artificial Intelligence*, 15:65–72.

Tayebia, A. and Islamb, S. (1984). Bettering operations of robots by learning. *Control Engineering Practice*, 14(7):843–851.

Tutty, O. R., Blackwell, M., Rogers, E., and Sandberg, R. (2012). Computational fluid dynamics based iterative learning control of peak loads in wind turbines. In *IEEE Conference on Decision and Control*, pages 3948–3953.

Tyréus, B. D. (1979). Multivariable control system design for an industrial distillation column. *Industrial and Engineering Chemistry Process Design and Development*, 18(1):1099–1121.

Uchiyama, M. (1978). Formation of high-speed motion pattern of a mechanical arm by trial. In *Society of Instrument and Control Engineers*, volume 14, pages 706–712.

Wallén, J., Norrlöf, M., and Gunnarsson, S. (2008). Arm-side evaluation of ILC applied to a six-degrees-of-freedom industrial robot. In *Proceedings of the 17th IFAC World Congress*, volume 17-1.

Wang, Y., Dassau, E., and Doyle, F. J. (2010). Closed-loop control of artificial pancreatic β-Cell in type 1 diabetes mellitus using model predictive iterative learning control. *IEEE Transactions on Biomedical Engineering*, 57(2):211–219.

Wanke, M. C., Lehmann, O., Muller, K., Wen, Q., and Stuke, M. (1998). Laser rapid prototyping of photonic band-gap microstructures. *Science*, 275:1284–1286.

Watabe, T., Yamakita, M., Mita, T., and Ohta, M. (2002). Output zeroing and iterative learning control for 3 link acrobat robot. *Proceedings of the 41st SICE Annual Conference*, 4:2579–2584.

Wijdeven, J. V. D. and Bosgra, O. (2008). Residual vibration suppression using hankel iterative learning control. *International Journal of Robust and Nonlinear Control*, 18:1034–1051.

Williamson, D. (1999). *Discrete-time signal processing*. Springer, first edition.

Wittenmark, B. and Salgado, M. E. (2002). Hankel-norm based interaction measure for input-output pairing. In *Proceeding of IFAC World Congress*, volume 15, pages 1623–1623.

Xiong, Q., Cai, W. J., and He, M. J. (2007). Equivalent transfer function method for PI/PID controller design of MIMO processes. *Journal of Process Control 17*, pages 665–673.

Yamakita, M., Yonemura, T., Michitsuji, Y., and Luo, Z. (2002). Stabilization of acrobat robot in upright position on a horizontal bar. *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3093–3098.

Ye, Y. and Wang, D. (2003). Better robot tracking accuracy with phase lead compensated ILC. *Proceedings of the IEEE International Conference on Robotics and Automation*, 3:4380–4385.

Zheng, J., Guo, G., and Wang, Y. (2004). Feedforward decoupling control design for dual-actuator system in hard disk drives. In *IEEE Transactions on Magnetics*, volume 40, pages 2080–2082.

Zheng, J., Guo, G., and Wang, Y. (2005). Identification and decentralized control of a dual-actuator hard disk drive system. In *IEEE Transactions on Magnetics*, volume 41, pages 2515–2521.

Zheng, J., Zhao, S., and Wei, S. (2009). Adaptively fuzzy iterative learning control for srm direct-drive volume control servo hydraulic press. In *International Conference on Sustainable Power Generation and Supply*, pages 1–6.

Ziegler, J. G. and Nichols, N. B. (1942). Optimum settings for automatic controllers. *Transaction of American Society of Mechnical Engineers*, 64:759–768.