Lakehead University

Knowledge Commons,http://knowledgecommons.lakeheadu.ca

Electronic Theses and Dissertations Retrospective theses

2004

Iterative learning control for robot manipulators

Abdul, Sajan

http://knowledgecommons.lakeheadu.ca/handle/2453/4012

Downloaded from Lakehead University, KnowledgeCommons

ITERATIVE LEARNING CONTROL FOR ROBOT MANIPULATORS

SAJAN ABDUL

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS OF MScEng DEGREE IN CONTROL ENGINEERING
FACULTY OF ENGINEERING
LAKEHEAD UNIVERSITY
THUNDER BAY, ONTARIO
P7B 5E1

DECEMBER 23, 2003

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Library and
Archives Canada

Published Heritage
Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et
Archives Canada

Direction du
Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 0-612-96990-8
Our file Notre référence
ISBN: 0-612-96990-8

The author has granted a non-exclusive license allowing the Library and Archives Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou aturement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Acknowledgements

I would like to deliver my sincere thanks to my supervisor Dr. Abdelhamid Tayebi for his invaluable guidance, advice and encouragement at every stage of this thesis work. And I extend my gratitude towards my co-supervisor Prof. K Natarajan for his valuable suggestions and support. I am thankful for the help provided by Mr Kailash Bhatia in setting up the mechanical experimental platform.

All my friends and fellow graduate students at Lakehead University are gratefully acknowledged, for their support, inspiration and for those innumerable discussions we had.

Finally, I would like to thank my family and friends whose help and support made this entire effort possible.

Sajan Abdul

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Contents

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

List of Figures

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

List of Acronyms and Notations

AILC	Adaptive Iterative Learning Control
ARX	AutoRegressive with eXternal input
BSPA	Balanced Singular Perturbation Approximation
D-H	Denavit-Hartenberg
ILC	Iterative Learning Control
LTI	Linear Time Invariant
PD	Proportional Derivative
PI	Proportional plus Integral
PID	Proportional plus Integral plus Derivative
RILC	Robust Iterative Learning Control
RMS	Root Mean Square
SISO	Single Input Single Output
A^{-1}	inverse of matrix A
A^T	transpose of matrix A
$\triangleq$	the left side is defined by the right side
$\hat{G}$	estimate of function G
$u(t)$	input variable at time t

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Chapter 1

Introduction

When a system is performing the same task repeatedly it is, from an engineering perspective, advantageous to use the knowledge from the previous iterations of the same task in order to reduce the error on successive trials.

In control systems, the aim is to force the system output to follow a desired trajectory as closely as possible. Specific norms and measures of optimality are used to determine how close the output is to the desired trajectory. Although control theory provides many different possible solutions for such problem, it is not always possible to achieve a desired set of performance requirements. This may be due to the presence of unmodeled dynamics or parametric uncertainties exhibited during the system operation, or due to the lack of suitable design techniques for particular class of systems. Iterative learning control (ILC) is a relatively new addition to these techniques that, for a particular class of problems, can be used to overcome some of the difficulties associated with performance design of control systems.

The iterative learning control problem considers that the control task is to perform a specific tracking command many times. Between each command application, the system is returned to the same initial condition. The development of ILC originates

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

from the area of robotics where this sort of repetitive motions shows up in many applications. The formulation used consists of a feedback controller, and a learning law which adjusts the command to the system from one iteration to the next, in order to minimize the tracking error. ILC differs from most existing control methods in the sense that, it exploits every possibility to incorporate past control information, such as tracking errors and control input signals, into the construction of the present control action. There are two phases in ILC: initially memory components are used to store past control information, then the stored information is fused in a certain manner so as to ensure that the system meets control specifications such as convergence and robustness. It is worth to note that the control specifications may not be easily satisfied by other control methods as they require more prior knowledge of the process in the stages of controller design. ILC requires much less information of the system parameters to yield desired dynamic behaviour. ILC has received considerable attention from researchers owing to its simplicity and effectiveness.

1.1 Motivation and Objectives

ILC approach is motivated by the observation that, if the controller used is fixed and the system's initial operating conditions are the same for each trial, then any error at the output will be repeated for each trial. These errors, if stored during the system operation, can be used to modify the succeeding inputs. The refinements are applied to the input until the desired performance criteria are attained. Implementation of two new approaches to ILC are undertaken as part of the thesis work.

The objective of the thesis work is to test the practical feasibility of the ILC schemes discussed in [26], [27] and [28]. The control strategies are implemented on a 2-DOF planar manipulator prototype.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

1.2 Literature Review

Historically the term learning control appears in the context of adaptive control and cybernetics. One of the earliest articles appeared in 1970, and some works on learning and adaptation was done during the first half of 1970's. But in all these works, the learning control referred to an online controller tuning method. The learning controller in its present form *i.e.*, offline scheme, was proposed by Uchiyama in his article published in 1978. The idea wasn't noticed by the community until after a couple of years, since the paper was published in Japanese and it took a few more years to become an active area of research. During this time works [1], [6] were published which discussed about a method that could iteratively compensate for model errors and disturbances. Some of the ILC algorithms already proposed in the literature includes [2], [11], [16], [17]. A survey on ILC literature upto 1991 could be found in [19].

Although the resetting condition is one of the requirement in many existing ILC algorithms, works have been done in proving the convergence for non-zero initial errors, as discussed in [5], [14], [20].

Different approaches are followed in ILC for ensuring the convergence of the tracking error, including robust and adaptive ILC methods. The ILC scheme proposed by [1] is an open loop configuration, purely feedforward action depending on the output at the previous operation. This cannot be used in real systems because of the lack of robustness with respect to non repeating disturbances along iterations. In addition to this, the tracking error may increase to a large value before it eventually converges. Thus, in real systems a feedback scheme is employed along with iterative learning control for enhancing system robustness and performance. The closed loop stability and disturbance rejection are guaranteed by the feedback controller while the iterative learning controller provides improved tracking performance over iterations. With

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

the presence of plant uncertainty, the iterative control problem could be transformed into a robust control problem [18]. This provides one with the robust control design procedures for the solution of iterative learning control problem [25], [26], [27].

Employing a PD or PID controller is a commonly used approach in robotic applications, mainly due to simplicity of its implementation. As long as the evaluation of the instantaneous gravity forces is possible, global asymptotic stabilization of the manipulator is possible using a PD controller with gravity compensation [23], [24]. But since this condition cannot be always guaranteed in practical situations, the PD controller leads to a steady state error. This steady state error can be reduced using high gain feedback. However, the high proportional and derivative gain may result in actuator saturation and may excite high frequency modes. Using the PID controllers local asymptotic stability was proven, but under some relatively complex conditions. The introduction of the passivity property for manipulators, allowed the design of globally asymptotically stabilizing PID controllers without gravity compensation [2]. Considering the fact that the robot parameters appear linearly in the Lagrange equation, it has been shown in [22] that a PD controller with an additional adaptive term is able to globally asymptotically stabilize rigid robot manipulators.

Since robot manipulators are used in repetitive operations one could take advantage of this fact for improving the performance in succeeding operations. The ILC techniques are used in order to enhance the tracking performance of such applications where this sort of repetitive motion shows up. In this regard one could mention, to name a few, the works of [2, 17, 24] for the existing ILC schemes for robot manipulators. Most of these ILC schemes are based on the contraction-mapping theory and requires some prior knowledge of the system dynamics [19].

Recently another type of ILC algorithms were developed based on Lyapunov and Lyapunov-like energy functions. In [9], a standard Lyapunov design is used for the solution of ILC problems. The idea consists of using a standard adaptive design and

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

setting the initial parameter estimates with the final parameter values of the previous iteration. In the nonlinear ILC scheme proposed by [29], for the position tracking problem of uncertain manipulators, the convergence of the iterative process is proven using a Lyapunov-like energy function. But all these control laws require a certain *a priori* knowledge of the system parameters.

The adaptive ILC schemes proposed in [28] do not require any *a priori* knowledge of system parameters and the proof of convergence is based on a Lyapunov-like energy function. The control algorithm involves the use of an iteratively updated term designed to take care of the unknown parameters and disturbances, together with a PD controller. In contrast to classical ILC schemes where one needs the same number of iterative variables as the number of control inputs, here we require only two iterative variables for controller implementation.

1.3 ILC problem Formulation

The basic idea of ILC is shown in figure 1.1. Let, at k^{th} trial, $u_k(t)$ be the input and $y_k(t)$ be the output. Based on the error ($e_k(t) = y_d(t) - y_k(t)$, where $y_d(t)$ is the desired output) that is observed through the trial, the ILC algorithm computes a modified input signal, $u_{k+1}(t)$, to be stored in memory until the next trial. This input is designed such that it will produce a smaller error than the previous input. Given a system described by $y = f_s(u)$, and desired output response $y_d(t)$, the ILC problem is to find an optimal input $u^*(t)$ that satisfies

$$\min_{u(t)} \|y_d(t) - f_s(u(t))\| = \|y_d(t) - f_s(u^*(t))\| \tag{1.1}$$

In this context, ILC is an iterative technique for finding $u^*(t)$ over the finite time interval [0,T]. The ILC approach is to generate a sequence of inputs $u_k(t)$ such that

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

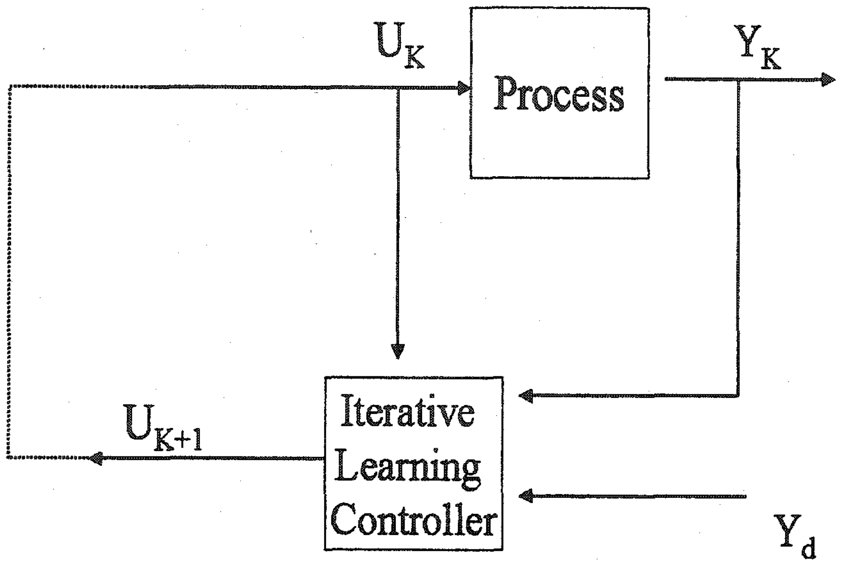


Figure 1.1: Concept of iterative learning control

the sequence converges to $u^*(t)$ such that

$$\lim_{k\to\infty} u_k(t) = u^*(t) \tag{1.2}$$

in the interval [0,T]. Based on the above mentioned properties and methodology on ILC, the following remarks could be made :

1. In a successful ILC algorithm the succeeding input will be computed so that the performance error will be reduced on the next trial. This is usually measured in terms of some norm, and the error should converge, with the condition that the convergence property must not depend on the desired response.

2. It is assumed that the initial conditions, during each trial, are always the same.

3. It is also assumed that the trial length,*i.e.* time span of operation is fixed.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

1.3.1 D type ILC

The simplest and earliest of ILC algorithms [1] is the derivative type given by,

$$u_{k+1}(t) = u_k(t) + \Gamma \dot{e}_k(t) \tag{1.3}$$

where Γ is the gain factor. For a SISO LTI system A, B, C, D with $CB \neq 0$, if $\|1 - CB\Gamma\| < 1$, and if some initial conditions are met, then on successive iterations $\lim_{k\to\infty} y_k(t) = y_d(t)$ in the sense of λ-norm defined by

$$\|x(t)\|_\lambda = \sup_{0\leq t\leq T} \left\{e^{-\lambda t}\|x(t)\|\right\}$$

1.3.2 PD control law

Consider the following control law [20],

$$u_{k+1}(t) = u_k(t) + \Gamma(\dot{e}_k(t) - Re_k(t)) \tag{1.4}$$

with

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) \end{aligned} \tag{1.5}$$

$$\|I - \Gamma CB\| \leq \rho_m \leq 1 \tag{1.6}$$

and suppose that (1.6) holds and the update law (1.4) is applied to (1.5). If the initial condition at each iteration remains the same, *i.e.* $x_k(0) = x_0, k = 0, 1, 2..$ then

$$\lim_{t\to\infty} y_k(t) = y_d(t) + e^{Rt}Cx_0$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

1.3.3 PID law

It is shown in [20] that when the ILC algorithm (1.7) is applied to system (1.5), the output trajectory converges to the form in (1.8).

$$u_{k+1}(t) = u_k(t) + \Gamma(\dot{e}_k(t) + Q_0 e_k(t) + Q_1 \int_0^t e_k(\tau)d\tau) \tag{1.7}$$

$$\begin{array}{l} lim \quad y_k(t) = y_d(t) + C_R e^{A_R t}\xi_0 \\ k \rightarrow \infty \end{array} \tag{1.8}$$

Here,

$$\begin{aligned} A_R &= \begin{bmatrix} 0 & I \\ -Q_1 & -Q_0 \end{bmatrix} \\ C_R &= \begin{bmatrix} I & 0 \end{bmatrix} \\ \xi_0 &= \begin{bmatrix} I \\ -Q_0 \end{bmatrix} C(x_0 - x_d(0)) \end{aligned}$$

From equation (1.8) it is clear that by introducing an integral term one can control the output trajectory in numerous ways.

1.4 ILC with Non-zero Initial Error

One restriction of ILC is that the initial condition at each iteration should be equal to the initial condition of the desired trajectory. The works on non-zero initial state error for error convergence has been done in [14]. Consider the linear time invariant system

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) \end{aligned} \tag{1.9}$$

where $x \in R^n$, $u \in R^r$, $y \in R^m$ denote state, input and output respectively. A, B and C are matrices with appropriate dimensions and CB is assumed to be non-singular.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Let the desired trajectory x_d be continously differentiable on [0,T] and it is assumed that

$$x_d(0) \equiv 0$$

Then the control considered by [1], for the system (1.9), is described as

$$u_{k+1}(t) = u_k + \Gamma \delta \dot{y}_k(t) \tag{1.10}$$

where,

$$\begin{aligned} \dot{x}_k(t) &= Ax_k(t) + Bu_k(t) \\ y_k(t) &= Cx_k(t) \\ y_d(t) &= Cx_d(t) \end{aligned}$$

and

$$\delta y_k(t) = y_d(t) - y_k(t)$$

The ILC structure is described in figure 1.2.

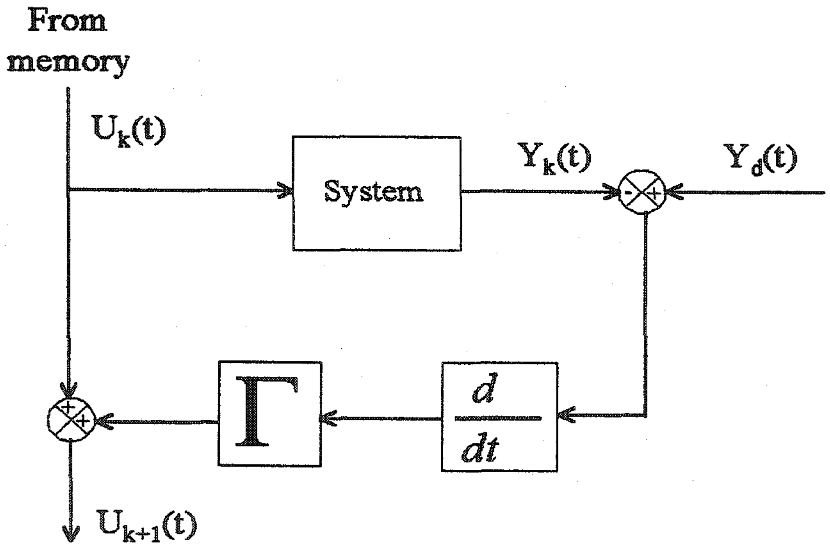


Figure 1.2: A simple D-type iterative learning control

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

And if conditions

$$\|I_r - \Gamma CB\| \leq \rho_m \leq 1 \tag{1.11}$$

$$y_k(0) = y_d(0) \equiv 0 \quad k = 0, 1, 2.. \tag{1.12}$$

are satisfied, then using the control law (1.10), the error between $y_k(t)$ and $y_d(t)$ approaches zero as $k \rightarrow \infty$.

It can be shown that if (1.11) is satisfied and the initial state error is bounded, that is, $\|x_k(0) - x_d(0)\| \leq b_{x0}$, then the error between $y_k(t)$ and $y_d(t)$ is bounded as $k \rightarrow \infty$ and this bound depends continously on the bound of the initial state error. Therefore as the bound on initial state error b_{x0} tends to zero, the bound on the output error also tends to zero. This shows that the learning law (1.10) is stable against any initial state error, if the error is bounded, even if the condition (1.12) does not hold. If the learning law (1.10) is applied to the system (1.9) under the assumption that $x_k(0) = x_0, k = 0, 1, 2..$, and if the condition (1.11) is satisfied, then

$$\lim_{k \to \infty} y_k(t) = y_d(t) + Cx_0$$

1.5 Overview of Thesis

The mathematical modeling, derivation of inverse kinematics, and dynamical equations are discussed in Chapter 2.

Chapter 3 gives a brief discussion on the experimental setup. The mechanical system, electronic circuitry and the data acquisition setup will be covered in this section.

Chapter 4 deals system identification, the experimental setup for identification procedure and system validations and model reduction.

Chapter 5 discusses the robust iterative learning control, the theory of robust control, μ-synthesis and experimental results.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Adaptive ILC is discussed in Chapter 6, with simulation and experimental results. In the final chapter we present the conclusions and some directions for future study.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Chapter 2

Modelling and Dynamics

2.1 Mathematical Model

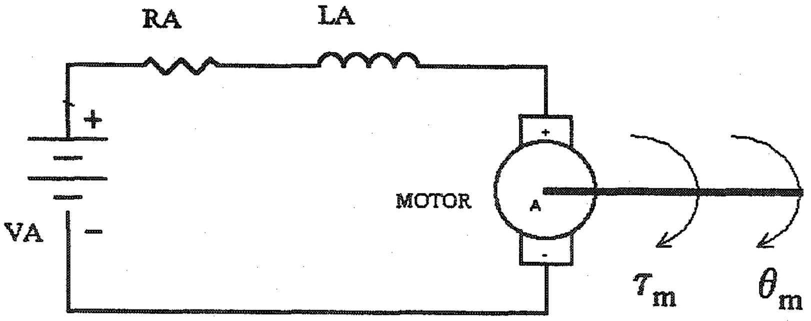


Figure 2.1: Circuit diagram for armature controlled DC motor

From [23] the schematic diagram of a DC motor is given in Figure 2.1.

V_A = armature voltage

L_A = armature inductance

R_A = armature resistance

V_b = back emf

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

b_m = viscous-friction coefficient

I_m = rotor inertia

i_A = armature current

θ_m = rotor position

θ = load position

τ_m = generated torque

τ = torque after gear system

η = gear ratio from motor shaft to load

The differential equation for the armature current is

$$V_A - V_b = L\frac{di_A}{dt} + R_A i_A, \tag{2.1}$$

where

$$V_b = K_b \dot{\theta}_m, \tag{2.2}$$

and K_b is the back emf constant. The torque developed by the motor is given by

$$\tau_m = K_m i_A, \tag{2.3}$$

$$\tau = \eta \tau_m, \tag{2.4}$$

$$\dot{\theta} = \frac{1}{\eta}\dot{\theta}_m. \tag{2.5}$$

From the equation of motion, we have

$$\tau_m = I_m \ddot{\theta}_m + b_m \dot{\theta}_m + \frac{1}{\eta}(I\ddot{\theta} + b\dot{\theta}), \tag{2.6}$$

which leads to $\tau = (I + \eta^2 I_m)\ddot{\theta} + (b + \eta^2 b_m)\dot{\theta}$. From the above expressions, we have

$$V_A = \frac{R_A}{K_m}\tau_m + K_b\dot{\theta}_m + \frac{L_A}{K_m}\frac{d\tau_m}{dt}, \tag{2.7}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

and based on the fact that the electrical time constant is much smaller compared to the mechanical time constant, the expression for V_A can be approximated to

$$V_A = \frac{R_A}{K_m}(I_m + \frac{I}{\eta^2})\eta\ddot{\theta} + [K_b + \frac{R_A}{K_m}(b_m + \frac{b}{\eta^2})]\eta\dot{\theta}. \tag{2.8}$$

This leads to a transfer function of the form

$$\frac{\theta(s)}{V_A(s)} = \frac{1}{s\ (sK_1 + K_2)}\ , \tag{2.9}$$

where $K_1 = \frac{R_A}{K_m}(I_m + \frac{I}{\eta^2})\eta$ and $K_2 = [K_b + \frac{R_A}{K_m}(b_m + \frac{b}{\eta^2})]\eta$.

2.2 Denavit-Hartenberg (D-H) Representation

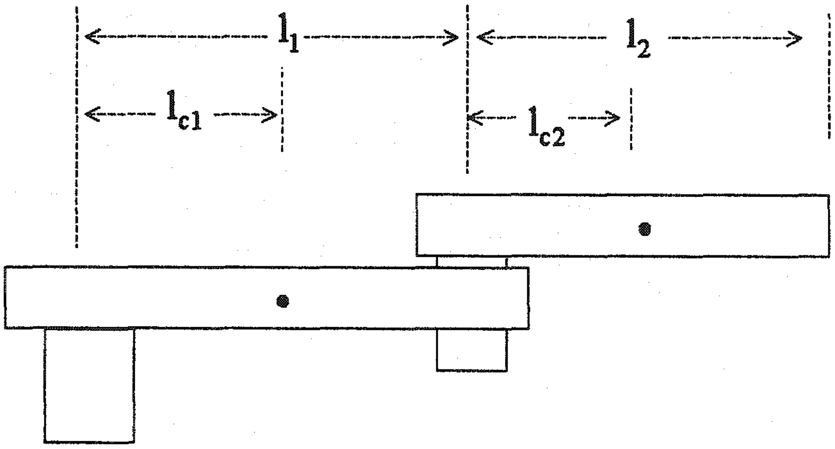


Figure 2.2: 2-DOF planar manipulator

The D-H representation (refer to Appendix B) is used for deriving forward kinematics of the manipulator under study (Figure 2.2). The resulting link table and the kinematic equations are shown,

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

DH Parameters for Robot				
Link	a_i	d_i	α_i	θ_i
1	a_1	0	0	$q_1{}^*$
2	a_2	0	0	$q_2{}^*$

$$A_1 = \begin{bmatrix} c_1 & -s_1 & 0 & a_1c_1 \\ s_1 & c_1 & 0 & a_1s_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad A_2 = \begin{bmatrix} c_2 & -s_2 & 0 & a_2c_2 \\ s_2 & c_2 & 0 & a_2s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_0{}^2 = A_1A_2 \quad \text{and} \quad T_0{}^1 = A_1$$

$$T_0{}^2 = \begin{bmatrix} c_{12} & -s_{12} & 0 & a_1c_1 + a_2c_{12} \\ s_{12} & c_{12} & 0 & a_1s_1 + a_2s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where,

$$\begin{aligned} c_1 &\triangleq \cos q_1, & s_2 &\triangleq \sin q_2 \\ c_2 &\triangleq \cos q_2, & c_{12} &\triangleq \cos(q_1 + q_2) \\ s_1 &\triangleq \sin q_1, & s_{12} &\triangleq \sin(q_1 + q_2) \end{aligned}$$

2.3 Inverse Kinematics

In the inverse kinematics problem, the solution for the joint parameters from the given position and orientation of the end effector, (x, y) and ϕ respectively are obtained.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

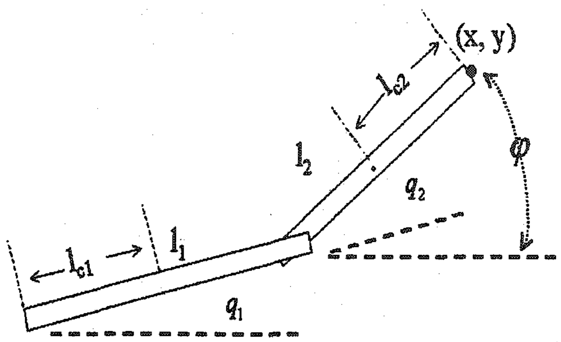


Figure 2.3: Two link revolute planar manipulator

Considering the x-y plane we have,

$$x = l_1 c_1 + l_2 c\phi \tag{2.10}$$

and

$$y = l_1 s_1 + l_2 s\phi \tag{2.11}$$

where $\cos\phi$ and $\sin\phi$ are denoted by $c\phi$ and $s\phi$ respectively. And l_1 and l_2 are the lengths of link1 and link2.

From the above equations, $c_1 = \frac{x - l_2 c\phi}{l_1}$ and $s_1 = \frac{y - l_2 s\phi}{l_1}$

$$\Rightarrow \quad q_1 = \arctan(c_1, s_1) \quad q_2 = \phi - q_1$$

2.4 Dynamical Equations

For describing the dynamics of rigid robot manipulators, one could use some analytical methods in mechanics, namely Euler-Lagrange and Newton-Euler method. While the former describes the mechanical system based on derivation of the Lagrangian, which is the difference of the kinetic and potential energy, the latter uses the balance of forces and torques for the dynamic model derivation.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Considering the 2-DOF robot manipulator shown in Figure 2.3, with the following notations :

l_1 : length of link 1

l_2 : length of link 2

m_1 : mass of link 1

m_2 : mass of link 2

l_{c_1} : distance to the centre of mass of the link 1

l_{c_2} : distance to the centre of mass of the link 2

I_1 : moment of inertia of link 1

I_2 : moment of inertia of link 2

$$\begin{aligned} x_1 &= l_{c_1} \cos q_1 \\ y_1 &= l_{c_1} \sin q_1 \\ x_2 &= l_{c_1} \cos q_1 + l_{c_2} \cos(q_1 + q_2) \\ y_2 &= l_{c_1} \sin q_1 + l_{c_2} \sin(q_1 + q_2) \end{aligned} \tag{2.12}$$

2.4.1 Euler-Lagrange method

In this method, a set of diffrential equations are derived under holonomic constraints, when the constraint forces satisfy the principle of virtual work. The derivation involves the computation of Langrangian given by the difference of the kinetic and potential energies. The potential energy, V of the system is given by,

$$V = m_2 g d, \tag{2.13}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

with d being the vertical distance between the two links.

And expression for Kinetic energy, K is,

$$\begin{aligned} K &= \frac{1}{2}m_1V_{c_1}^2 + \frac{1}{2}m_2V_{c_2}^2 + \frac{1}{2}I_1\dot{q}_1^2 + \frac{1}{2}I_2(\dot{q}_1 + \dot{q}_2)^2 \\ &= \frac{1}{2}m_1\dot{q}_1^2l_{c_1}^2 + \frac{1}{2}m_2l_1\dot{q}_1^2 + \frac{1}{2}m_2l_{c_2}^2(\dot{q}_1 + \dot{q}_2)^2 \\ &\quad +2m_2l_1l_{c_2}\dot{q}_1(\dot{q}_1 + \dot{q}_2)\cos q_2 + \frac{1}{2}I_1\dot{q}_1^2 + \frac{1}{2}(\dot{q}_1 + \dot{q}_2)^2 \qquad (2.14) \end{aligned}$$

The dynamic equations are given by,

$$\tau_1 = \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_1}\right) - \frac{\partial L}{\partial q_1} \qquad (2.15)$$

$$\tau_2 = \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_2}\right) - \frac{\partial L}{\partial q_2} \qquad (2.16)$$

where Lagrangian, L is given by

$$\begin{aligned} L &= K - V \\ &= \frac{1}{2}m_1\dot{q}_1^2l_{c_1}^2 + \frac{1}{2}m_2l_1^2\dot{q}_1^2 + \frac{1}{2}m_2l_{c_2}^2(\dot{q}_1 + \dot{q}_2)^2 - m_2gd \\ &\quad +m_2l_1l_{c_2}\dot{q}_1(\dot{q}_1 + \dot{q}_2)\cos q_2 + \frac{1}{2}I_1\dot{q}_1^2 + \frac{1}{2}I_2(\dot{q}_1 + \dot{q}_2)^2 \qquad (2.17) \end{aligned}$$

and

$$\begin{aligned} \frac{\partial L}{\partial \dot{q}_1} &= m_2l_1^2\dot{q}_1 + m_1l_{c_1}^2\dot{q}_1 + m_2l_{c_2}^2(\dot{q}_1 + \dot{q}_2) + I_1\dot{q}_1 \\ &\quad +m_2l_1l_{c_2}(2\dot{q}_1 + \dot{q}_2)\cos q_2 + I_2(\dot{q}_1 + \dot{q}_2) \qquad (2.18) \\ \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_1}\right) &= m_2l_1^2\ddot{q}_1 + m_1l_{c_1}^2\ddot{q}_1 + m_2l_{c_2}^2(\ddot{q}_1 + \ddot{q}_2) + I_1\ddot{q}_1 + I_2(\ddot{q}_1 + \ddot{q}_2) \\ &\quad +m_2l_1l_{c_2}(2\ddot{q}_1 + \ddot{q}_2)\cos q_2 - m_2l_1l_{c_2}(2\dot{q}_1 + \dot{q}_2)\dot{q}_2\sin q_2 \qquad (2.19) \\ \frac{\partial L}{\partial q_1} &= 0 \qquad \frac{\partial L}{\partial q_2} = -m_2l_1l_{c_2}\dot{q}_1(\dot{q}_1 + \dot{q}_2)\sin q_2 \qquad (2.20) \\ \frac{\partial L}{\partial \dot{q}_2} &= m_2l_{c_2}^2(\dot{q}_1 + \dot{q}_2) + I_2(\dot{q}_1 + \dot{q}_2) + m_2l_1l_{c_2}\dot{q}_1\cos q_2 \qquad (2.21) \\ \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_2}\right) &= m_2l_{c_2}^2(\ddot{q}_1 + \ddot{q}_2) + I_2(\ddot{q}_1 + \ddot{q}_2) \\ &\quad +m_2l_1l_{c_2}\ddot{q}_1\cos q_2 - m_2l_1l_{c_2}\dot{q}_1\dot{q}_2\sin q_2 \qquad (2.22) \end{aligned}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

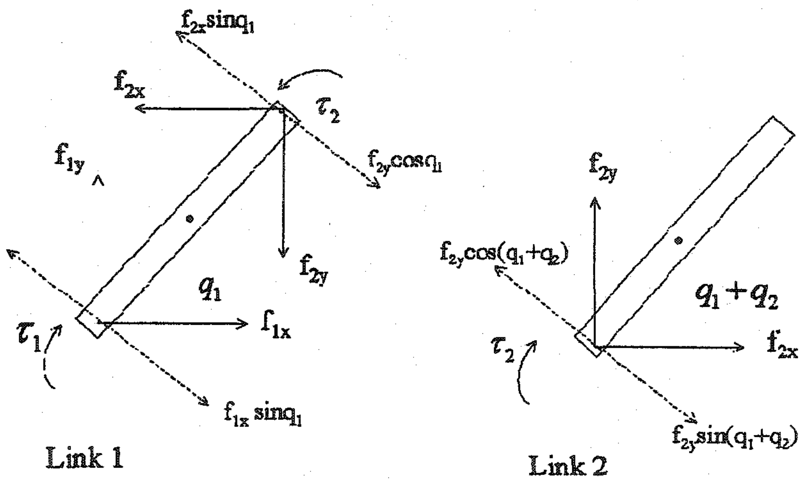


Figure 2.4: Force and moments on links

From equations (2.12) - (2.22), the dynamic model could be derived as,

$$\begin{aligned}
\tau_1 &= (m_1 l_{c1}^2 + m_2 l_1^2 + m_2 l_{c_2}^2 + I_1 + I_2)\ddot{q}_1 + (m_2 l_{c_2}^2 + I_2)\ddot{q}_2 \\
&\quad + m_2 l_1 l_{c_2}(2\ddot{q}_1 + \ddot{q}_2)\cos q_2 - m_2 l_1 l_{c_2}(2\dot{q}_1 + \dot{q}_2)\dot{q}_2 \sin q_2 \qquad (2.23) \\
\tau_2 &= I_2(\ddot{q}_1 + \ddot{q}_2) + m_2 l_{c_2}^2(\ddot{q}_1 + \ddot{q}_2) + m_2 l_{c_2} l_1 \dot{q}_1^2 \sin q_2 \\
&\quad + m_2 l_{c_2} l_1 \ddot{q}_1 \cos q_2 \qquad (2.24)
\end{aligned}$$

2.4.2 Newton-Euler method

This method is based on the balance of forces and torques using free-body diagrams as shown in Figure 2.4.

Based on the linear motion, of link 2, we have

$$\begin{aligned}
f_{2x} &= m_2 \ddot{x}_2, \\
f_{2y} &= m_2 \ddot{y}_2. \qquad (2.25)
\end{aligned}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Balance of torques for link 2 leads to,

$$\tau_2 + l_{c_2} f_{2y} \cos(q_1 + q_2) - l_{c_2} f_{2x} \sin(q_1 + q_2) = I_2(\ddot{q}_1 + \ddot{q}_2). \tag{2.26}$$

From (2.12), (2.25) and (2.26),

$$\tau_2 = I_2(\ddot{q}_1 + \ddot{q}_2) + m_2 l_{c_2}^2(\ddot{q}_1 + \ddot{q}_2) + m_2 l_{c_2} l_1 \dot{q}_1^2 \sin q_2 + m_2 l_{c_2} l_1 \ddot{q}_1 \cos q_2 \tag{2.27}$$

From the linear motion of link 1, we have

$$\begin{aligned} f_{1y} &= m_1 \ddot{y}_1 + f_{2y}, \\ f_{1x} &= m_1 \ddot{x}_1 + f_{2x}. \end{aligned} \tag{2.28}$$

Balance of torques for link 1, leads to

$$I_1 \ddot{q}_1 = -l_{c_1} f_{1y} \cos q_1 - (l_1 - l_{c_1}) f_{2y} \cos q_1 - \tau_2 + \tau_1 \tag{2.29}$$

$$+ l_{c_1} f_{1x} \sin q_1 + (l_1 - l_{c_1}) f_{2x} \sin q_1. \tag{2.30}$$

From (2.12), (2.28) and (2.30),

$$\begin{aligned} \tau_1 = {} & (m_1 l_{c1}^2 + m_2 l_1^2 + m_2 l_{c_2}^2 + I_1 + I_2)\ddot{q}_1 + (m_2 l_{c_2}^2 + I_2)\ddot{q}_2 \\ & + m_2 l_1 l_{c_2}(2\ddot{q}_1 + \ddot{q}_2)\cos q_2 - m_2 l_1 l_{c_2}(2\dot{q}_1 + \dot{q}_2)\dot{q}_2 \sin q_2. \end{aligned} \tag{2.31}$$

Now, combining (2.31) and (2.27), and rearranging the expression into a matrix form, we obtain

$$\begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \tag{2.32}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

with,

$$\begin{aligned}
m_{11} &= m_1 l_{c1}^2 + m_2 l_1^2 + m_2 l_{c_2}^2 + 2m_2 l_1 l_{c_2} \cos q_2 + I_1 + I_2 \\
m_{12} &= m_{21} \;=\; m_2 l_{c_2}^2 + m_2 l_1 l_{c_2} \cos q_2 + I_2 \\
m_{22} &= m_2 l_{c_2}^2 + I_2 \\
c_{11} &= -m_2 l_1 l_{c_2} \dot{q}_2 \sin q_2 \\
c_{12} &= -m_2 l_1 l_{c_2} (\dot{q}_1 + \dot{q}_2) \sin q_2 \qquad (2.33) \\
c_{21} &= m_2 l_1 l_{c_2} \dot{q}_1 \sin q_2, \qquad c_{22} \;=\; 0
\end{aligned}$$

The dynamical equation of the robot manipulator is given by,

$$M(q_k)\ddot{q}_k + C(q_k, \dot{q}_k)\dot{q}_k + G(q_k) = \tau_k \tag{2.34}$$

where, $M(q_k)$ is inertia matrix, $C(q_k, \dot{q}_k)\dot{q}_k$ is a vector resulting from coriolis and centrifugal forces. $G(q_k)$ is a vector resulting from gravitational forces. τ_k is the control input vector containing torques and forces to be applied at each joint.

2.5 Summary

- The derivation of the mathematical model of the system, helps us to have an idea on the order and of the system transfer function. And in this case, from (2.9) it is evident that the system has an integrator.
- The derivation of the inverse kinematic expressions provides one with the desired joint variables.
- In this chapter the dynamical equation of the manipulator was derived using both the Euler-Langrange and Newton-Euler methods. The knowledge of the system dynamical equation helps one in the design and implementaion of the control strategies.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Chapter 3

Experimental Setup

The experimental setup consists of the mechanical assembly, the electronic circuitry, and the data acquisition system.

3.1 Mechanical Assembly

The mechanical setup for the 2-DOF manipulator consists of two links made of aluminum connected through gear trains. The dimensions of the links being 74cm×5cm ×5cm and 47.5cm×5cm×5cm. The joints are actuated by two DC motors, Pittman GM9434 H187, via gear trains with optical encoders providing motor position measurements. The motor has a built-in gearbox with a transmission ratio of 1:5.9 and with a no load speed of 6151rpm. The mechanical assembly has a further gear ratio of 30:86. Counter weights are added to the links to reduce gravity effects on the system. The motors are driven by the associated circuitry.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

3.2 Circuitry

To drive the motors, which act as the actuators for the two links, the necessary current is provided by the electronic circuitry. The overall circuitry shown in Figure A.1 consists of motor power supply module, bridge module, power supply module for the various chips, current control module and signal separation module.

3.2.1 Motor Power Supply Module

The schematic of the motor power supply module is shown in Figure A.7. The module provides power to the 2N6059 amplification circuit in bridge module, for the control of the current magnitude and direction.

3.2.2 Current Control Module

This circuit shown in Figure A.3, is used for controlling the magnitude of the motor current. The circuit is based on a proportional-integral (PI) controller. The output of the current control module is given by

$$u = \left(1 + \frac{R_{12}}{R_{11}}\right)\left(\frac{R_{10}R_7}{R_8R_5}e + \frac{R_{10}}{R_9R_6C_1}\int_0^t edt\right) \tag{3.1}$$

3.2.3 Bridge Module

The bridge circuit is constituted of four MOSFETs IRF540, as shown in Figure A.5. The MOSFETs Q_1, Q_2, Q_3 and Q_4 are used for switching the motor current directions. The current directions are determined by the MOSFET pairs Q_1, Q_3 and Q_2, Q_4 which moves the motor in clockwise and counter-clockwise directions. The desired current is obtained by applying the output of the current control module to the base of the 2N6059 transistor, the value of which could be measured as the voltage across the resistor R_9. To reduce high voltage buildup during the MOSFET switching,

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

between the drain and source MOSFETs resistors, capacitors and diodes are used. The resistors R_5, R_6, R_7 and R_8 used at the gates of each MOSFETs, are used to reduce the noise from the outputs of opto-isolation circuitry.

3.2.4 Opto-Isolation Module

This module provides an isolation between the power supplies and generates two 180-degree out of phase signals for the switching of MOSFET pairs. The schematic of this circuit is shown in Figure A.6. The module consists of DS0026CN clock driver chip, HP261A opto-isolation chip and associated circuitry. The output of phase signals guarantees that the MOSFET pairs Q_1, Q_3 and Q_2, Q_4 in the bridge module do not turn on and off at the same instant. This ensures the control on motor direction. The module has three opto-isolators for generating MOSFET driving signals. DS0026CN is a high speed two phase MOS clock driver containing the necessary interface circuit, with peak current of ±1.5A. The use of high speed opto-isolator and MOSFET drivers enables the circuit to operate at higher frequencies during data acquisition.

3.2.5 Signal-Separation Module

The schematic of this circuit is shown in Figure A.4. This module separates the command signal from data acquisition (DAQ) board into magnitude and direction signals. The direction output is a signal waveform with values of zero and

$$V_{dir} = \frac{R_4}{R_3 + R_4} V_+$$

where V_+ is the output of Op-amp LM1458.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

3.2.6 Power supply Module

It is used to provide ±12V and 5V supply voltages to the various integrated circuits, like opto-isolators, Op-amps, etc the power supply module as shown in Figure A.7 is incorporated. It is a full wave rectifier with required voltage regulators.

3.3 Data Acquisition Module

For the experimental setup dSPACE data acquisition board DS1102 is used with Matlab/simulink and a friendly graphical user interfaced (GUI) realtime workshop, Controldesk. Through the associated circuitry, the motor current serving as the system input is controlled through the DS1102 data acquisition board. DS1102 is a single board system, specifically designed for real-time control implementation. The DAQ board is based on Texas Instruments TMS320C31 third generation floating point digital signal processor, which builds the main processing unit. The DSP has been supplemented by a set of on-board peripherals frequently used in digital control systems. Analog to digital converters (ADC) and digital to analog converters (DAC), DSP micro controller based digital I/O subsystem and incremental sensor interfaces are included. The DAQ board has four ADC's, two 16-bit ADC's working at 250KHz sampling rate and two 12-bit ADC's working at 800KHz sampling rate. All ADC's have single ended bipolar inputs with ±10V input span. The board also includes four 12-bit DAC's working in ±10V range.

The voltage command signal from the analog output channel serves as the controller output. This signal is separated, by the signal separator module, into the magnitude and direction signal. These signals are used to control the magnitude and direction of the motor current.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Chapter 4

System Identification

System identification deals with the construction of mathematical models from observed data of the physical system. It hence becomes an integral part in the control of physical systems. The system identification problem can be divided into a number of subproblems:

1. experiment design,
2. data collection,
3. model structure selection,
4. model estimation, and
5. model validation.

For system identification two approaches are used : closed loop identification and open loop identification.

1. *Open Loop Identification*: The system model from the input to the output is identified in open loop configuration.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

2. *Closed Loop Identification*: The system model from the setpoint to the output (*indirect approach*) is identified, from the captured data. The system transfer function is obtained from the closed loop transfer function and the knowledge of the controller transfer function. Another approach followed in closed loop identification is the *direct approach*, where the model between the control input and the output of the closed loop system is identified.

4.1 Closed Loop Identification

During the closed loop identification the output is fed back by means of some feedback mechanism as shown in Figure 4.1. Here plant models are identified using data collected from closed loop experiments and the underlying processes are fully or partially under feedback control. It is sometimes necessary to perform identification experiments in a closed loop. The reason maybe that the plant is unstable or that it has to be controlled for production, economic or safety reasons. The closed loop

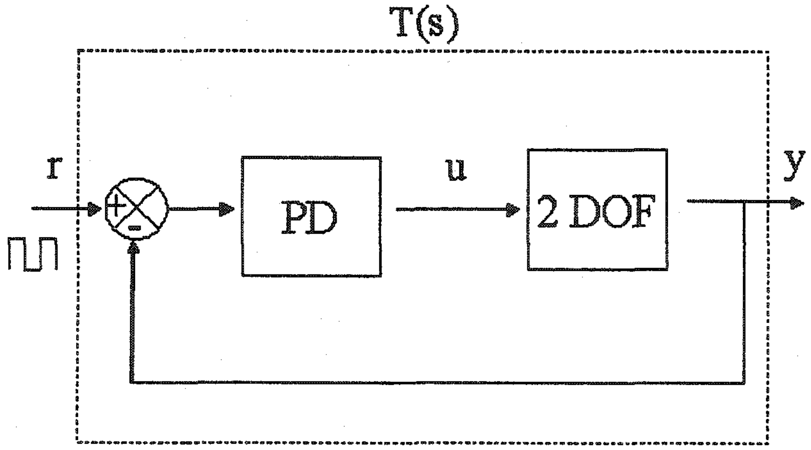


Figure 4.1: A closed loop system

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

identification is often a feasible approach offering a number of practical advantages [13]

1. validation of the designed controller and onsite re-tuning,
2. obtaining better models for controller design,
3. controller maintenance,
4. iterative identification in closed loop and controller re-design and
5. controller order reduction.

Closed loop identification could be done using the direct method or by the indirect method [7].

1. *Direct method* : The system output (y) and the input to the system (u) are used for determining the transfer function of the plant (Figure 4.1). Then

   $$\hat{G}(s) = \frac{Y(s)}{U(s)}$$

   will provide the transfer function with $Y(s)$ and $U(s)$ being the Laplace transforms of $y(t)$ and $u(t)$ respectively.

2. *Indirect method*: Here, the system output (y) and the setpoint (r) are used along with the controller transfer function to obtain the system transfer function.

   $$\hat{G}(s) = \frac{\hat{T}(s)}{1 - \hat{T}(s)} \frac{1}{C(s)} \tag{4.1}$$

   where, $\hat{T}(s)$ is the identified closed loop transfer function, C(s) is the controller transfer function.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

In this report the indirect closed loop identification method is used for the system identification because the system was not stable in open loop. The closed loop system is realized with a PD control, and the signals of relevance are collected. The collected signals are analyzed, and the ARX closed loop model is obtained using MATLAB [15]. Then the system transfer function is identified as per (4.1).

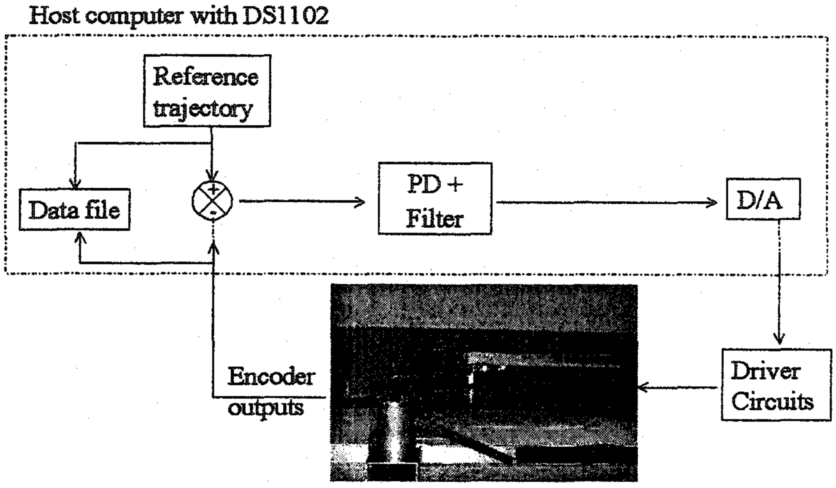


Figure 4.2: System setup for closed loop identification

4.2 Closed Loop Experiment

The system setup for the closed loop experiment is as shown in Figure 4.2. In this experiment the magnitude of excitation signal is chosen as the required maximum angular position. A PD controller of the form (4.2) was chosen as the controller.

$$C(s) = K_p + \frac{K_d s}{\frac{1}{f_c}s + 1} \tag{4.2}$$

where K_p is the proportional gain, K_d is the derivative gain and f_c is the cut off frequency (in rad/sec) of the lowpass filter. For link 1, K_p, K_d and f_c were chosen as

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

0.5, 0.1 and 20 rad/sec respectively. And for link 2, the parameters K_p, K_d and f_c were chosen to be 1.5, 0.1 and 30 rad/sec respectively.

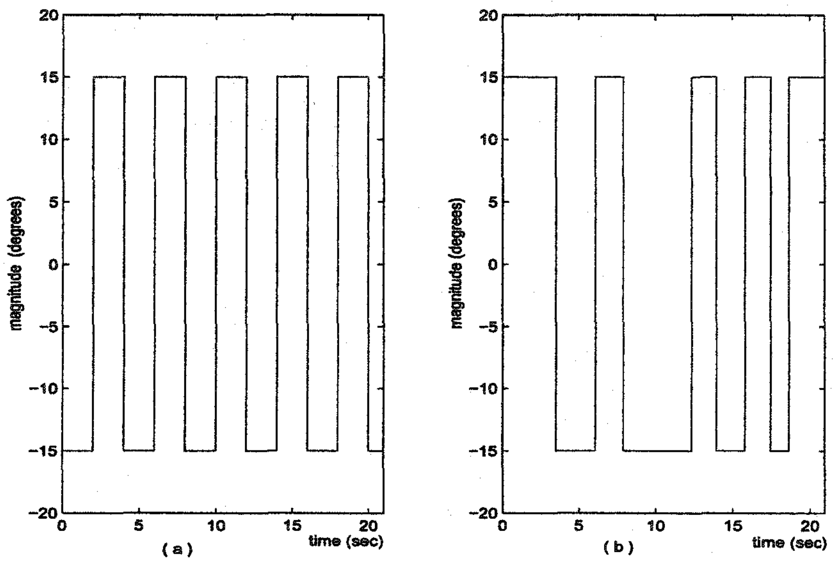


Figure 4.3: Input excitation signals (a) Square wave (b) Varying square waveform

For the closed loop identification procedure, two types of excitation signals (Figure 4.3) were used:

1. *Square waveform*: The position input for the manipulator is varied according to a square waveform of 0.25 Hz frequency and amplitude of 15 degrees.

2. *Varying square waveform*: The amplitude of input waveform is maintained a constant at 15 degrees and pulse width is altered. The width is selected such that the angular position has settled at the reference before the next setpoint change occurs.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

4.3 Model Validation

The identified models are validated to find the linear model closest to the system. In this thesis, we used the set-point following validation technique. In this validation method, comparison of the simulation outputs and measurement outputs is made under the same PD controller and position reference. The identified closed loop transfer functions are tabulated as in Tables 4.1 and 4.2. The percentage of the output variations that is reproduced by the model is calculated based on (4.3), to find the model of the best fit [15].

$$Fit = \left[1 - \frac{\|Y - \hat{Y}\|_2}{\|Y - \bar{Y}\|_2}\right] \times 100. \tag{4.3}$$

where Y is the measured output, $\hat{Y}$ is the simulated output and $\bar{Y}$ is the mean of the measured output. As per (4.3), 100% corresponds to a model of the best fit, *i.e*, when the measured output Y is equal to the simulated output $\hat{Y}$. The identified linear models obtained using the two excitation signals, *i.e.* square wave and varying square waveform, are tabulated in Table 4.1 and 4.2.

Based on the best fitting characteristics given by (4.3), the model with the best fit is chosen from all of the obtained linear models *i.e*, 8^{th} order closed loop transfer function for link 1 and 3^{rd} order closed loop transfer function for link 2. The open loop system transfer functions are then obtained as per (4.1). The output responses of the chosen linear models and the measured output for link 1 and link 2 are as shown in Figures 4.4 and 4.5.

The identified open loop transfer functions for link 1 is given by,

$$G_1(s) = \frac{N_1(s)}{D_1(s)} \tag{4.4}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table 4.1: Closed loop identification results for Link 1

Results for experiments on Link 1		
Excitation signal	System order	Fit
square wave	3	79.4717
	4	83.3664
	6	87.0752
	7	89.5196
	8	95.4855
	10	94.0327
	11	93.8297
varying square wave	3	79.1397
	4	81.3961
	5	82.7970
	9	90.4462
	10	90.6485
	14	90.9095
	15	90.8775

where

$$\begin{aligned} N_1(s) &= -3.308s^8 + 1837s^7 - 7.343 \times 10^5 s^6 + 8.876 \times 10^8 s^5 + 4.4811 \times 10^{10} s^4 \\ &\quad +5.422 \times 10^{13} s^3 + 2.144 \times 10^{15} s^2 + 6.067 \times 10^{16} s + 7.865 \times 10^{17} \end{aligned}$$

$$\begin{aligned} D_1(s) &= s^9 + 455.1s^8 + 7.633 \times 10^5 s^7 + 1.962 \times 10^8 s^6 + 1.439 \times 10^{11} s^5 + 1.331 \times \\ &\quad 10^{13} s^4 + 5.62 \times 10^{15} s^3 + 6.037 \times 10^{16} s^2 + 1.543 \times 10^{17} s + 7.82 \times 10^{15} \end{aligned}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table 4.2: Closed loop identification results for Link 2

Results for experiments on Link 2		
Excitation signal	System order	Fit
square wave	3	95.8855
	4	95.0112
	5	94.9981
	6	95.7260
	7	95.6995
	9	95.4980
	11	95.7556
varying square wave	3	93.6329
	4	93.9735
	5	93.8157
	7	93.8224
	9	94.0188
	10	94.0114
	15	94.4486

and for link 2

$$G_2(s) = \frac{N_2(s)}{D_2(s)} \tag{4.5}$$

where,

$$\begin{aligned} N_2(s) &= 0.125s^4 + 670.3s^3 + 2.85 \times 10^4 s^2 + 4.994 \times 10^7 s + 1.491 \times 10^9 \\ D_2(s) &= s^5 + 478.6s^4 + 6.871 \times 10^5 s^3 + 4.305 \times 10^7 s^2 \\ &\quad +3.789 \times 10^8 s + 2.724 \times 10^8 \end{aligned}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

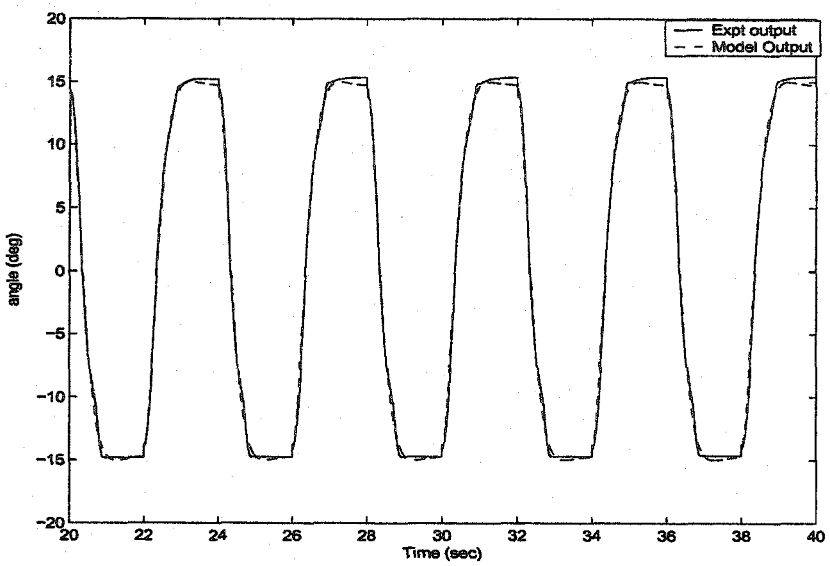


Figure 4.4: Outputs of the experimental system and the identified model for link1

4.3.1 Model Reduction

It is often desirable to approximate the state-space representation of a system with lower order state-space representation. This is for the reason that an identified model is often over parameterized, to capture the system dynamics. The need for an over parameterization is due to the non-linearity and the measurement noise. This procedure, referred to as model reduction, is carried out on the identified higher order transfer functions for the ease of controller implementation. Among the widely used model reduction methods one can mention the balanced truncation, balanced residualization, singular perturbation approximation and the optimal Hankel norm approximation [10], [21]. Truncation methods though simple, may result in unstable and non-minimal phase systems even if the original system is stable and minimal. In addition to this drawback, truncation methods result in low frequency steady state

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

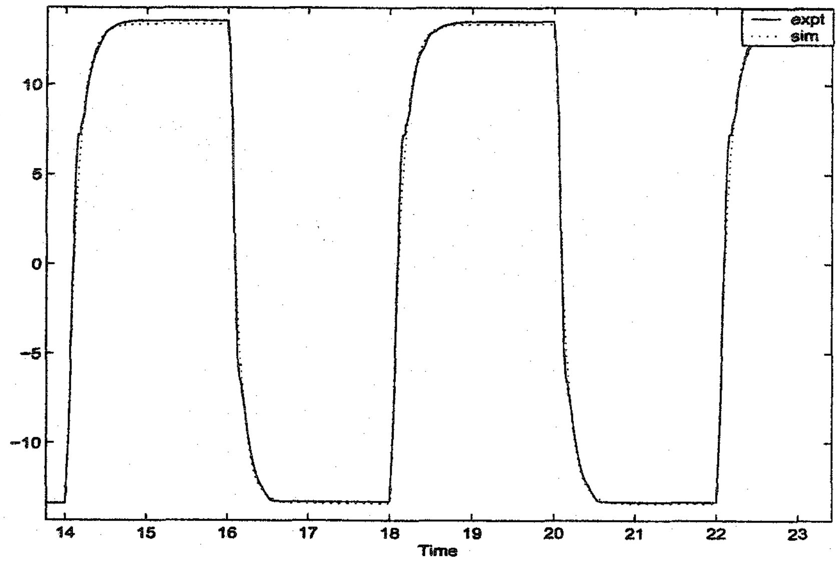


Figure 4.5: System and model outputs for link 2

error, while it offers zero error at infinite frequency.

The balanced singular perturbation approximation (BSPA) method offers improved low frequency model reduction characteristics [10]. This is a slight variation from existing balanced trunction algorithm, and offers exact matching at zero frequency thus leading to perfect steady state performance.

Due to the better approximation at low and medium frequencies the balanced singular perturbation approximation was used in our case.

Considering a linear, time-invariant system G(s) with the realization

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t) + Du(t) \end{aligned} \tag{4.6}$$

and partitioning the state vector x into components to be retained and those to be

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

discarded:

$$x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}. \tag{4.7}$$

The r-vector $x_1(t)$ contains the components to be retained, while $(n-r)$-vector $x_2(t)$ contains the components to be discarded. Partitioning the matrices A, B and C conformably with x, we obtain

$$\begin{aligned} A &= \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \\ B &= \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} \\ C &= \begin{bmatrix} C_1 & C_2 \end{bmatrix} \\ \Sigma &= \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} \end{aligned} \tag{4.8}$$

where $\Sigma_1 = diag(\sigma_1, \sigma_2, ..., \sigma_k)$ and $\Sigma_2 = diag(\sigma_{k+1}, \sigma_{k+2}, ..., \sigma_n)$ with $\sigma_k > \sigma_{k+1}$, where the ordered Hankel singular values of G(s) are denoted by σ_i.
The r^{th} order BSPA is given by $S_r(A,\ B,\ C,\ D) = (\hat{A}_{11},\ \hat{B}_1,\ \hat{C}_1,\ \hat{D})$, where

$$\begin{aligned} \hat{A}_{11} &= A_{11} - A_{12}A_{22}^{-1}A_{21}, \\ \hat{B}_1 &= B_1 - A_{12}A_{22}^{-1}B_2, \\ \hat{C}_1 &= C_1 - C_2A_{22}^{-1}A_{21}, \\ \hat{D} &= D - C_2A_{22}^{-1}B_2. \end{aligned} \tag{4.9}$$

The BSPA model reduction technique preserves the stability, controllability and observability grammians. The identified open loop transfer functions given by (4.4) and (4.5), are reduced to the third order transfer functions using the BSPA model

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

reduction technique. The transfer functions obtained after model reduction are given by:

$$G_1(s) = \frac{0.007543s^3 + 0.3964s^2 + 10.48s + 149.3}{s^3 + 11.26s^2 + 29.28s + 1.484} \tag{4.10}$$

$$G_2(s) = \frac{8.728 \times 10^{-4}s^3 - 5.83 \times 10^{-3}s^2 + 74.27s + 2260}{s^3 + 64.89s^2 + 574s + 412.9} \tag{4.11}$$

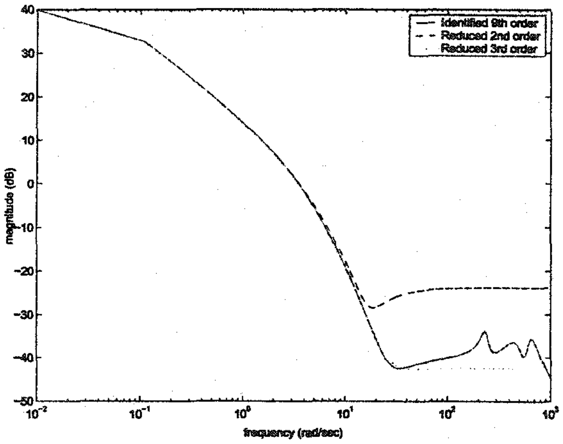


Figure 4.6: Magnitude plots of identified and (BSPA) reduced system transfer functions of link 1

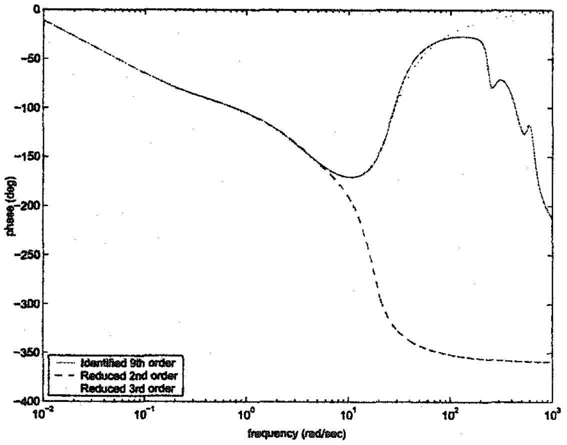


Figure 4.7: Phase plots of identified and (BSPA) reduced system transfer functions of link 1

The frequency response plots for the reduced models and the identified transfer functions are as shown in Figures 4.6-4.9.

Numerous data for each type of exciting signals were used for the identification. Based on the identification results, the following observations were made:

1. The use of the square waveform as excitation signal gave better results than the varying square waveform, since controlling the pulse width in varying square waveform was too critical.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

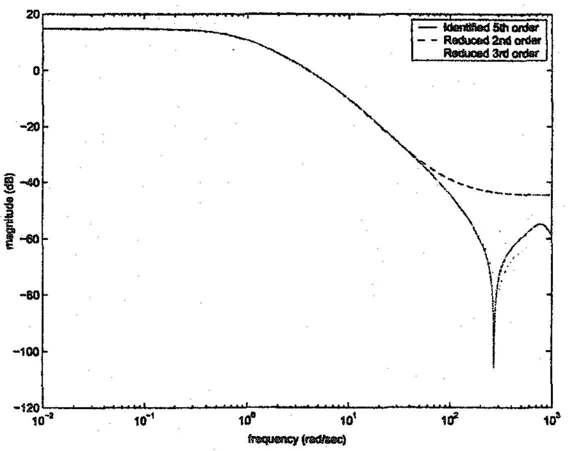


Figure 4.8: Magnitude plots of identified and (BSPA) reduced system transfer functions of link 2

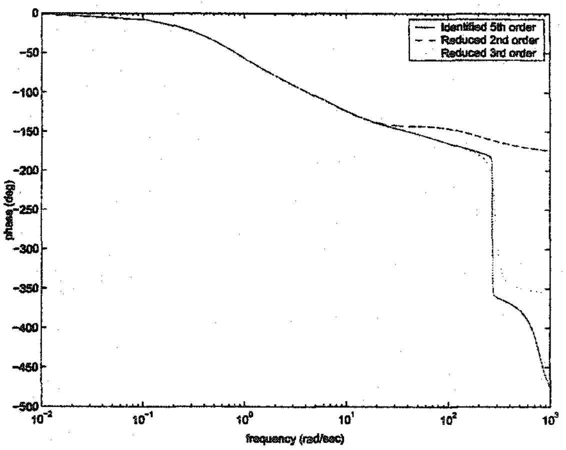


Figure 4.9: Phase plots of identified and (BSPA) reduced system transfer functions of link 2

2. For the varying square waveform the frequency of the signal variation is critical for the determination of accurate models.

3. Even though the identified discrete domain closed loop transfer functions were minimum phase, the conversion to continous domain introduced unstable zeros.

4. The identified models are of higher order than the mathematical model, due to the accommodated nonlinearities of the physical system. The identified models must be over parameterized to capture the system dynamics.

4.4 Summary

Based on the experimental results, the following conclusions could be drawn:

- The system under consideration has nonlinearities and a linear approximation of the transfer function was obtained from the identification process.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

- Both the identified transfer functions have a pole close to the imaginary axis, which could be justified as per the derived mathematical model in (2.9).

- Although more effective and accurate system identification methods can be used, the indirect closed-loop identification technique was used assuming that the designed robust controller would take care of the lost precision in the identified system.

- The identified transfer function is of higher order, since it accounts for the nonlinearities of the physical system. Model reduction techniques are carried out for the ease of controller design.

- The identified transfer functions (4.4), (4.5) are non-minimum phase, though the identified discrete domain system has no unstable zeros. This happens due the discrete to continuous domain conversion techniques.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Chapter 5

ROBUST ILC DESIGN VIA μ-SYNTHESIS

In [25], [27] a relation between iterative learning control problem and robust control problem is established. Thus, a design of a robust controller will in turn guarantee a solution for the iterative learning control problem. This allows the designer to use tools available in robust control, like loop shaping, model matching, H_∞ and μ-synthesis, for ILC design. The existence of a robust controller enables one to benefit from its performance in the first iteration when learning controller is not effective.

The design of robust controllers can cover parametric uncertainties (affecting low and medium frequencies) and unstructured model uncertainties (often located in the high frequency range). The $\mu-$synthesis problem consists of finding the controller that minimizes a given μ condition. A method to synthesize a $\mu-$optimal controller is the DK-iteration. This method combines H_∞ synthesis and $\mu-$analysis, and often yields good results. The starting point is the upper bound on the μ in terms of the scaled singular value,

$$\mu(N) \leq \min_{D \in \mathcal{D}} \bar{\sigma}(DND^{-1})$$

The aim is to find the controller that minimizes the peak value over frequency of this

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

upper bound, namely

$$\min_k(\min_{D\in\mathcal{D}} \|DN(K)D^{-1}\|_\infty)$$

by alternating between minimizing $\|DN(K)D^{-1}\|_\infty$ with respect to either K or D (while holding the other fixed). To start the iterations, one selects an initial stable rational transfer matrix D(s) with appropriate structure. The identity matrix is often a good initial choice for the D provided the system has reasonably scaled for performance. The D-K iteration then proceeds as follows:

1. *K-step*: Synthesize an H_∞ controller for the scaled problem $\min_k \|DN(K)D^{-1}\|_\infty$ with fixed D(s).

2. *D-step*: Find $D(j\omega)$ to minimize at each frequency $\bar{\sigma}(DND^{-1}(j\omega))$ with fixed N.

3. Fit the magnitude of each element of $D(j\omega)$ to a stable and minimum phase transfer function D(s) and go back to step 1. The iteration may continue until satisfactory performance is achieved $\|DN(K)D^{-1}\|_\infty < 1$ or until the H_∞ norm no longer decreases.

5.1 Robust Controller

Considering the structure as shown in Figure 5.1 where $\Delta(s) \in \mathcal{RH}_\infty$ with $\|\Delta(s)\|_\infty < \frac{1}{\beta}$ and $\beta > 0$.

If the generalized plant $M \in \mathcal{C}^{(q_1+q_2)\times(p_1+p_2)}$ is a matrix partitioned as follows:

$$M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}$$

Then, the lower linear fractional transformation (LFT) with respect to Δ could be defined as

$$\mathcal{F}_l(M,\Delta) = M_{11} + M_{12}\Delta(I - M_{22}\Delta)^{-1}M_{21},$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

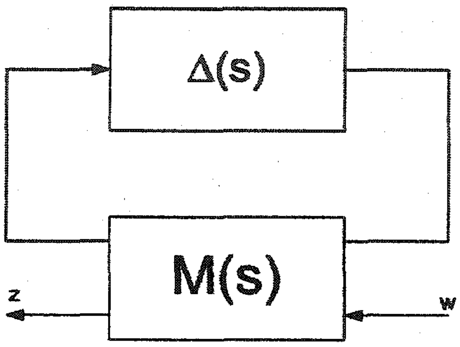


Figure 5.1: M-delta structure

where $\Delta \in \mathcal{C}^{q_1 \times p_1}$ and is well posed if $(I - M_{22}\Delta)^{-1}$ exists. And the upper LFT with respect to Δ is defined as

$$\mathcal{F}_u(M, \Delta) = M_{22} + M_{21}\Delta(I - M_{11}\Delta)^{-1}M_{12},$$

with the well posedness being guaranteed by the existence of $(I - M_{11}\Delta)^{-1}$.

The robust ILC scheme, proposed in [27], is as shown in the Figure 5.2, with the iterative rule

$$V_{k+1}(s) = W_1(s)(V_k(s) + U_k(s))$$

with $V_1(s) = 0$. The system could be described in the multiplicative uncertain form as

$$G = (1 + \Delta W_2)G_n,$$

with G_n being the nominal system, W_2 a known stable transfer function and Δ is an unknown stable transfer function satisfying $\|\Delta\|_\infty \leq 1$. It is considered that $y_d(0) = y_k(0) = 0$. The aim is to design a controller satisfying the robust performance condition

$$\||W_1 S| + |W_2 T|\|_\infty < 1, \tag{5.1}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

with $S = \frac{1}{1+CG_n}$ is the sensitivity function, and $T = 1 - S$ the complimentary sensitivity function. Then the tracking error uniformly converges to

$$e_\infty(t) = \lim_{k\to\infty} e_k(t) = \mathcal{L}^{-1}\left(\frac{1-W_1}{1-W_1+CG_n(1+\Delta W_2)}Y_d\right) \tag{5.2}$$

which implies the error convergence to zero when $W_1(s) = 1$.

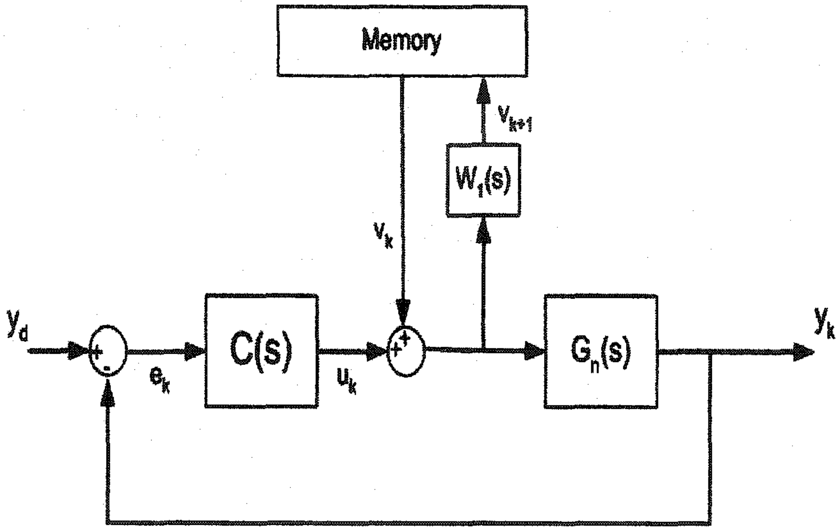


Figure 5.2: Robust ILC scheme

5.2 Uncertainties

System uncertainties can be classified as repeatable or non-repeatable from a learning control point of view. Repeatable uncertainties are those which are invariant over iterations and may be structured or unstructured. Similarly, non-repeatable uncertainties are those which are variant over iterations and may be structured or unstructured. Learning control schemes cannot handle non-repeatable uncertainties.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

5.3 Design Steps

As shown in [26], the Youla parameterization

$$C(s) = \frac{X(s) + N_2(s)Q(s)}{Y(s) - N_1(s)Q(s)}, \tag{5.3}$$

is used, where $\frac{N_1(s)}{N_2(s)}$ is the coprime factorization of $G_n(s)$, with $Q(s)$, $N_1(s)$ and $N_2(s)$ being stable rational transfer functions. The stable rational functions $X(s)$ and $Y(s)$ are solutions of the Bezout identity

$$N_1(s)X(s) + N_2(s)Y(s) = 1,$$

For a given nominal system $G_n(s)$, the transfer functions $N_1(s)$, $N_2(s)$, $X(s)$, and $Y(s)$, can be obtained using the following procedure [8], [30]:

- STEP 1. Obtain a state space realization A_G, B_G, C_G, D_G of $G_n(s)$, *i.e.*,

$$G_n(s) = D_G + C_G(sI - A_G)^{-1}B_G \triangleq \left[\begin{array}{c|c} A_G & B_G \\ \hline C_G & D_G \end{array}\right]$$

- STEP 2. Obtain F such that $A_G + B_GF$ is stable. Transfer functions $N_1(s)$ and $N_2(s)$ are given by

$$N_1(s) \triangleq \left[\begin{array}{c|c} A_G + B_GF & B_G \\ \hline C_G + D_GF & D_G \end{array}\right], \quad N_2(s) \triangleq \left[\begin{array}{c|c} A_G + B_GF & B_G \\ \hline F & 1 \end{array}\right]$$

- STEP 3. Obtain H such that $A_G + HC_G$ is stable. Transfer functions $X(s)$ and $Y(s)$ are given by

$$X(s) \triangleq \left[\begin{array}{c|c} A_G + HC_G & H \\ \hline F & 0 \end{array}\right], \quad Y(s) \triangleq \left[\begin{array}{c|c} A_G + HC_G & -B_G - HD_G \\ \hline F & 1 \end{array}\right]$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

In the case of stable systems, the above expressions could be taken as, $N_1 = G_n, N_2 = 1, X = 0, Y = 1$ which would lead to internal model parameterization, $C(s) = \frac{Q(s)}{1-G_n(s)Q(s)}$, which is a particular case of Youla parameterization. The robust performance condition (5.1) is equivalent to the following condition [8]:

$$\left\| \frac{W_1 S}{1+\Delta W_2 T} \right\|_\infty < 1 \ and \ \|W_2 T\|_\infty < 1.$$

and since the sensitivity and complimentary sensitivity functions with the Youla Parameterization are given by $S = N_2(Y - N_1 Q)$ and $T = 1 - S = N_1(X + N_2 Q)$, then the condition

$$\left\| \frac{W_1 N_2(Y - N_1 Q)}{1+\Delta W_2 N_1(X + N_2 Q)} \right\|_\infty < 1 \ and \ \|W_2 N_1(X + N_2 Q\|_\infty < 1 \tag{5.4}$$

if satisfied will guarantee the boundedness of the tracking error in the ILC scheme in Figure 5.2, for all $k \in N$, and its uniform convergence to the value given in (5.2) when $k \to \infty$.

Considering the generalized matrix given by

$$M_1 = \begin{bmatrix} -W_2 N_1(X + N_2 Q) & W_2 N_1(X + N_2 Q) \\ -W_1 N_2(Y - N_1 Q) & W_1 N_2(Y - N_1 Q) \end{bmatrix} \tag{5.5}$$

with the following upper LFT:

$$\mathcal{F}_u(M_1, \Delta) = \frac{W_1 N_2(Y - N_1 Q)}{1 + \Delta W_2 N_1(X + N_2 Q)}, \tag{5.6}$$

which is well posed if $\|W_2 N_1(X + N_2 Q)\| < 1$. Thus as per [26] and [27], and control scheme as in Figure 5.2 with C parameterized as in (5.3), then the existence of Q satisfying

$$\sup_{\omega \in \mathcal{R}} \mu_\Delta(M_1(j\omega)) \ < \ 1 \tag{5.7}$$

will guarantee error convergence to (5.2).

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

For a given W_1, W_2, X, Y and G_n, the function Q(s) satisfying (5.7) could be calculated using $\mu-$synthesis. To this end, the matrix M_Q is introduced such that $M_1 = \mathcal{F}_l(M_Q, Q)$.

$$M_Q = \left[\begin{array}{cc|c} -W_2N_1X & W_2N_1X & W_2N_2 \\ -W_1N_2Y & W_1N_2Y & -W_1N_2 \\ \hline -N_1 & N_1 & 0 \end{array}\right] \tag{5.8}$$

Realization of Q(s) using $\mu-$synthesis could be done by the D-K iteration procedure by minimizing

$$\|D\mathcal{F}_l(M_Q, Q)D^{-1}\|_\infty \tag{5.9}$$

by iteratively solving for D and Q.

The existance of Q satisfying (5.7) implies that $\|W_2N_1(X + N_2Q)\|_\infty < 1$ and $\|\mathcal{F}_u(M_1, \Delta)\|_\infty < 1$, which guarantees the convergence of the iterative scheme and the robust performance of the feedback system at the first iteration.

5.4 Choice of the Weighting Functions

The knowledge of the weighting functions is required for the design of a robust controller for the identified transfer function. The weighting functions W_1 and W_2 are selected based on certain criteria [21]. The fact that W_1 close to 1 ensures zero error convergence, determines choice of this weighting function. And W_2 is selected from a rough approximation of the relative uncertainty at the steady state, and the approximate frequency at which the relative uncertainty reaches 100%.

$$W_2(s) = \frac{\tau s + r_0}{(\tau / r_\infty)s + 1} \tag{5.10}$$

where $1/\tau$ is the approximate frequency at which the relative uncertainty reaches 100%, r_0 is the relative uncertainty at steady state and r_∞ is the magnitude of the

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

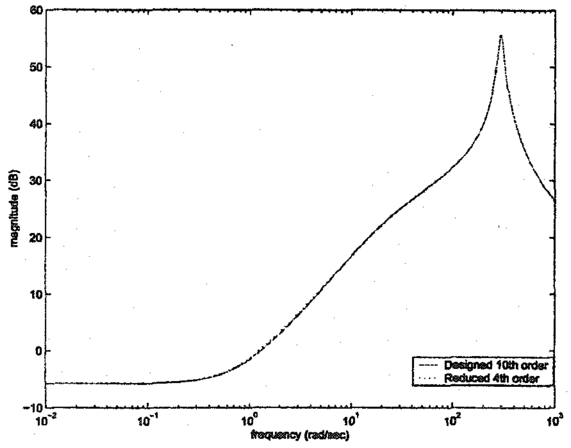


Figure 5.3: Magnitude plots of designed and (BSPA) reduced controller transfer functions

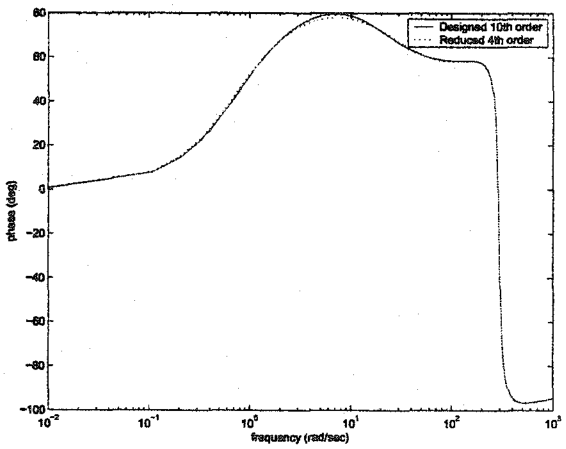


Figure 5.4: Phase plots of designed and (BSPA) reduced controller transfer functions

weight at high frequency, typically a value greater than 2. In this experimental setup τ was selected as 0.01, r_∞ as 2 and r_0 as 0.5.

5.5 Simulation Results

For the design of controller, parameterization was chosen as in (5.3), with $N_1 = G_n$, $N_2 = 1$, $X = 0$ and $Y = 1$. Using the μ-synthesis toolbox in MATLAB, the robust controller is designed using the known weighting function and the identified nominal system [3]. During the experiment model reduction techniques were carried out on the obtained transfer functions. Since better approximations were obtained for low and medium frequency ranges, the singular perturbation approximation method was preferred.

The transfer function obtained after model reduction is given by:

$$G_n(s) = \frac{0.0008728s^3 - 0.00583s^2 + 74.27s + 2260}{s^3 + 64.89s^2 + 574s + 412.9}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

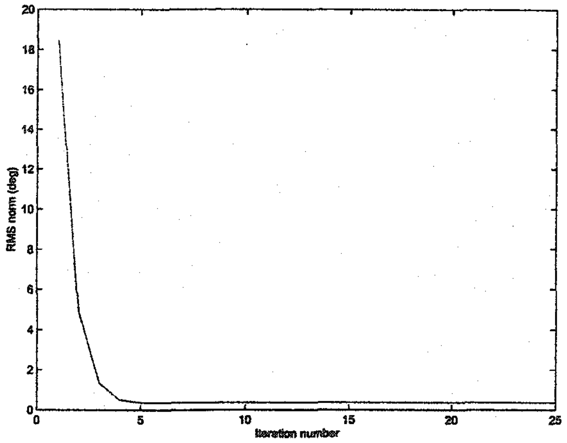


Figure 5.5: RMS norm of tracking error versus iteration number (simulated for $W_1(s) = \frac{1}{0.09s+1}$)

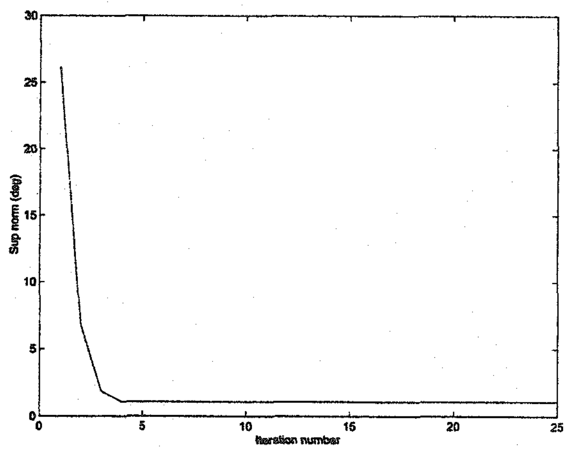


Figure 5.6: Sup-norm of tracking error versus iteration number (simulated for $W_1(s) = \frac{1}{0.09s+1}$)

and with (5.8) and $N_1 = G_n$, $N_2 = 1$, $X = 0$, $Y = 1$ and

$$W_2(s) = \frac{0.01s + 0.5}{0.005s + 1} \qquad W_1(s) = \frac{1}{0.09s + 1}$$

and the μ–synthesis method [3],we obtain, after model reduction, the following controller

$$C(s) = \frac{0.2422s^4 + 18620s^3 + 6.15 \times 10^6 s^2 + 4.155 \times 10^8 s + 3.102 \times 10^8}{s^4 + 239.7s^3 + 101700s^2 + 1.809 \times 10^7 s + 6.029 \times 10^8}$$

leading to $\| |W_1 S| + |W_2 T| \| \ = \ 0.6312$ as shown in Figure 5.18.

The RMS norm given by

$$\|e_k\|_{rms} = \sqrt{\frac{1}{N}\sum_{n=1}^{N} e_k^2(nT)}\ ,$$

where k is iteration number, T is the sampling period and N is the number of samples, was calculated on error signal at each iteration.

The reference trajectory was taken as $y_d = 100\sin(0.1571t)$, $t \in [0, 20s]$ with $y_d(0) = y_k(0) = 0$ satisfied on all iterations.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

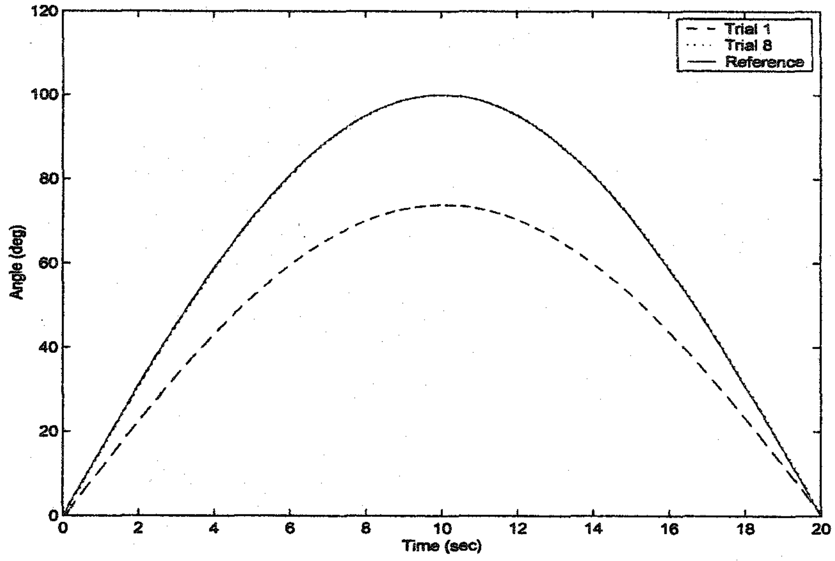


Figure 5.7: Trajectories of outputs (simulated for $W_1(s) = \frac{1}{0.09s+1}$)

The designed higher order controller is reduced using BSPA technique, the frequency response of the original and reduced controller transfer functions are as shown in Figures 5.3 and 5.4.

The learning controller was realized in SIMULINK environment with $\Delta = 0.5$ and the results are as shown in Figures 5.5 to 5.13 and the results agrees with (5.2). The choice of $W_1 = 1$ gives an error convergence to zero, and a value of $W_1 = \frac{4}{0.1s+1}$ gives convergence to a higher non-zero value. A higher value of W_1 guarantees better performance at the first iteration but not a satisfactory convergence property.

5.6 Experimental Results

The experimental setup consists of a single link planar manipulator and the associated electronic circuits and was implemented in SIMULINK/dSpace environment with a sampling period of 0.004 *sec*. For the experimental implementation of robust learning

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

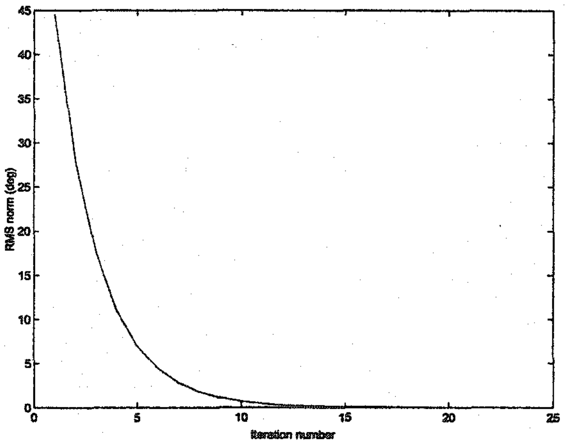


Figure 5.8: RMS norm of tracking error versus iteration number (simulated for W_1=1)

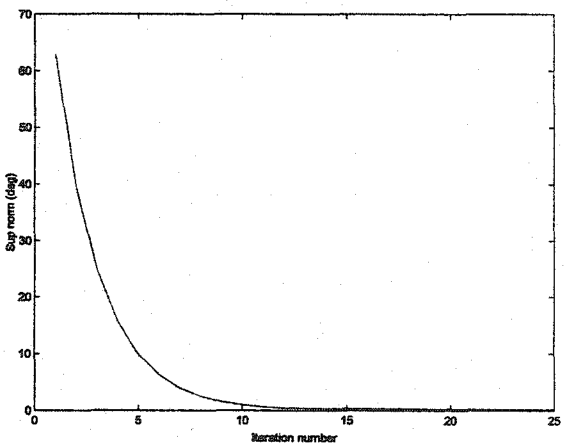


Figure 5.9: Sup-norm of tracking error versus iteration number (simulated for W_1=1)

controller (Figure 5.14), a first order low pass filter of 10 rad/sec cut-off frequency was used. The addition of filter leads to $\||W_1S| + |W_2T|\|_\infty = 1.0781$ which does not satisfy the performance criteria for a frequency range of about 14 rad/sec to 22 rad/sec (shown in Figure 5.19), but it satisfies the robust performance condition at the operating frequencies *i.e.* lower frequencies.

The experimental results for the implemented learning controller are shown in Figures 5.15, 5.16 and 5.17.

5.7 Summary

- The designed robust iterative learning controller offers good performance and error convergence.

- In this learning procedure, apart from the identified system model, the choice of weighting functions are critical. The choice of $W_1(s) = 1$ guarantees error

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

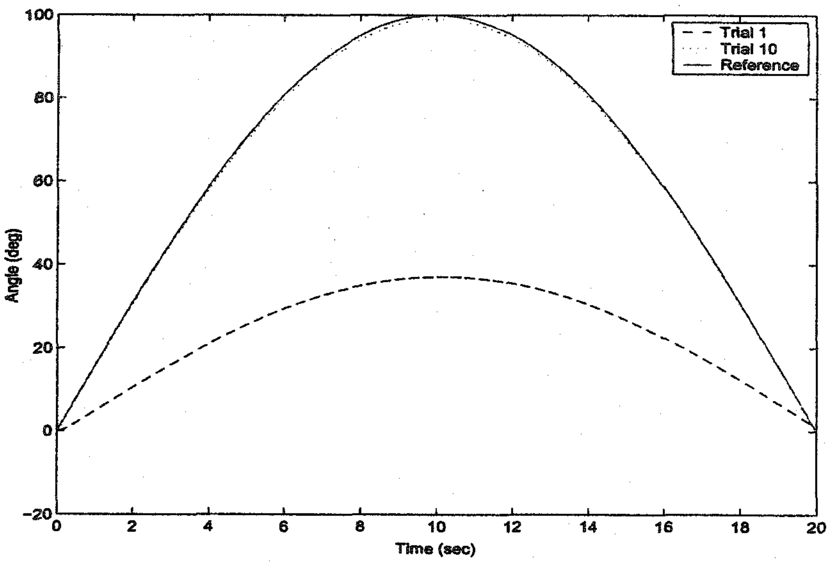


Figure 5.10: Trajectories of outputs (simulated for W_1=1)

convergence to zero but the closed loop performance for the first iteration may not be satisfactory.

- The main practical drawback of this technique is the increased design complexity and resulting higher order controllers. Hence the model reduction is necessary in this procedure.

The adaptive ILC schemes discussed later in the report, enables one to come around many of the above mentioned drawbacks.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

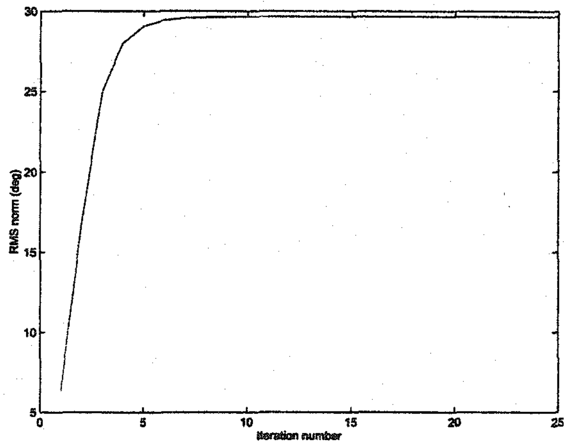


Figure 5.11: RMS norm of tracking error versus iteration number (simulated for $W_1(s) = \frac{4}{0.1s+1}$)

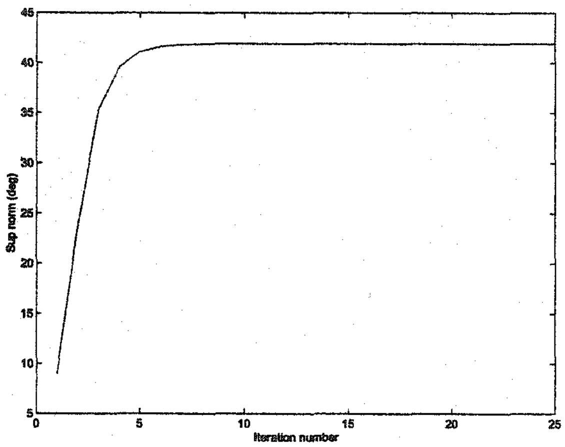


Figure 5.12: Sup-norm of tracking error versus iteration number (simulated for $W_1(s) = \frac{4}{0.1s+1}$)

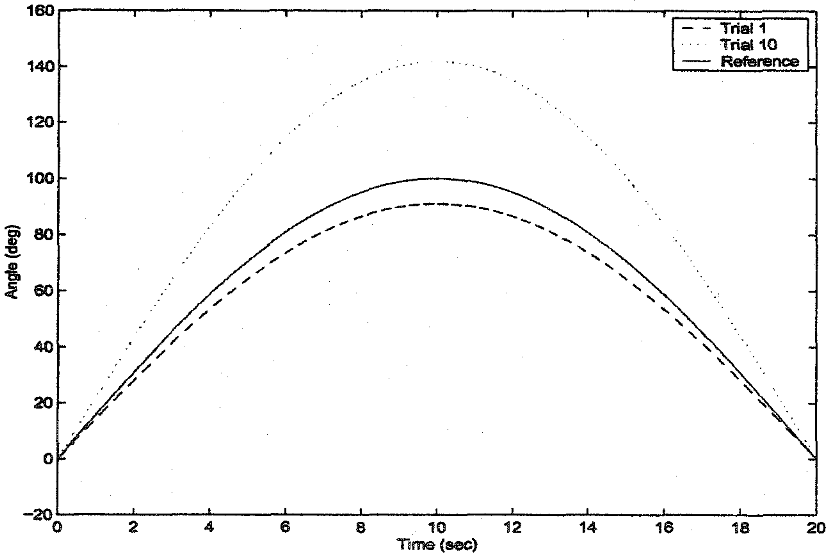


Figure 5.13: Trajectories of outputs (simulated for $W_1(s) = \frac{4}{0.1s+1}$)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

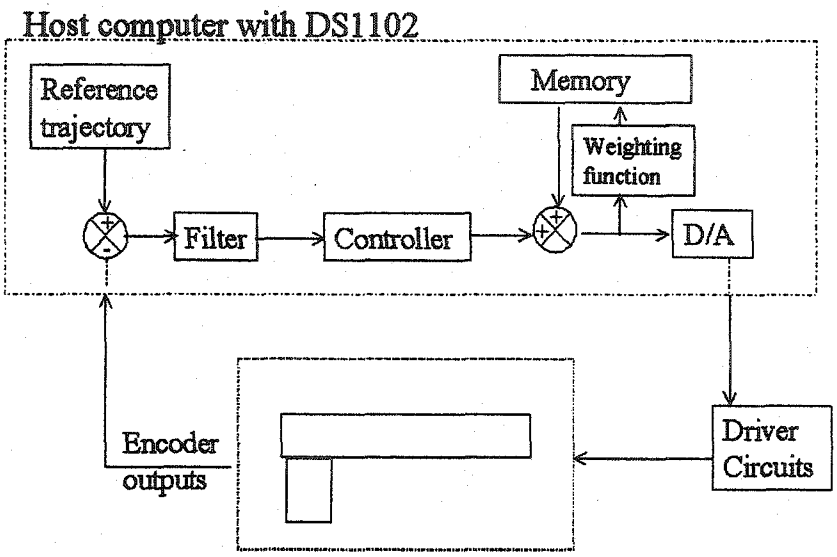


Figure 5.14: Experimental setup for ILC implementation

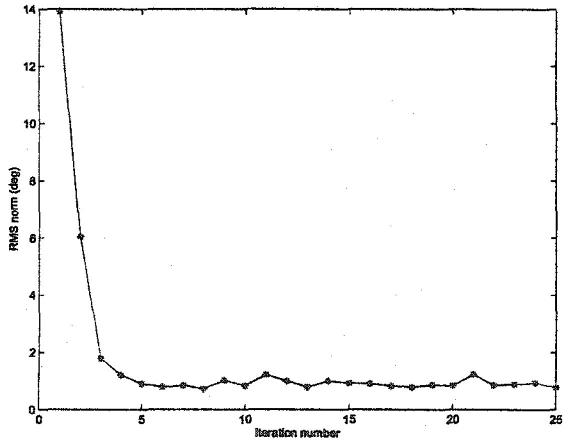


Figure 5.15: RMS norm of tracking error versus iteration number (filter of 10 rad/sec)

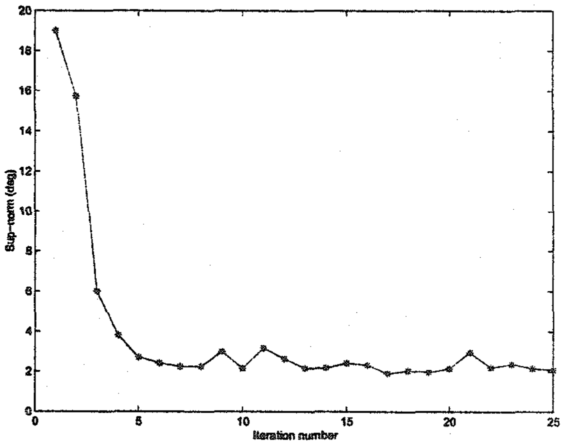


Figure 5.16: Sup-norm of tracking error versus iteration number (filter of 10 rad/sec)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

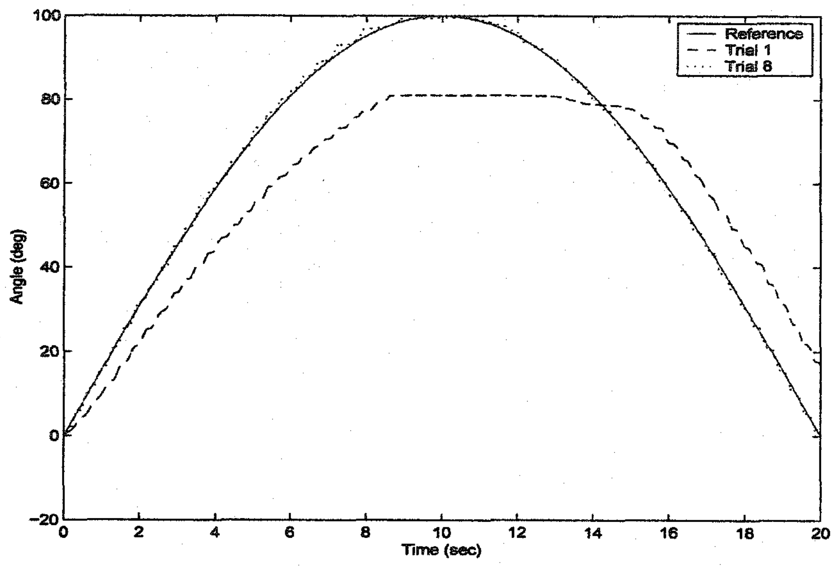


Figure 5.17: Trajectories of outputs

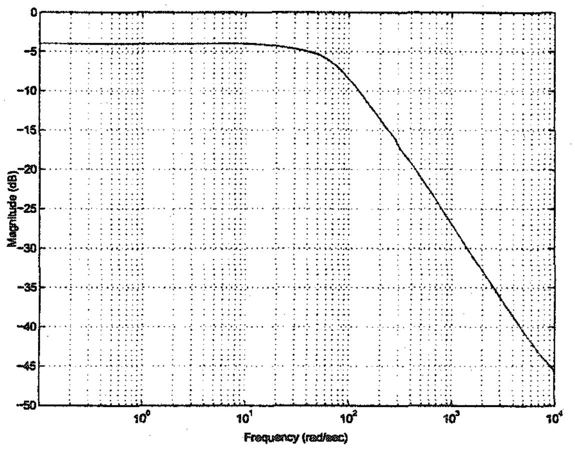


Figure 5.18: $|W_1S|+|W_2T|$ versus frequency (with designed controller)

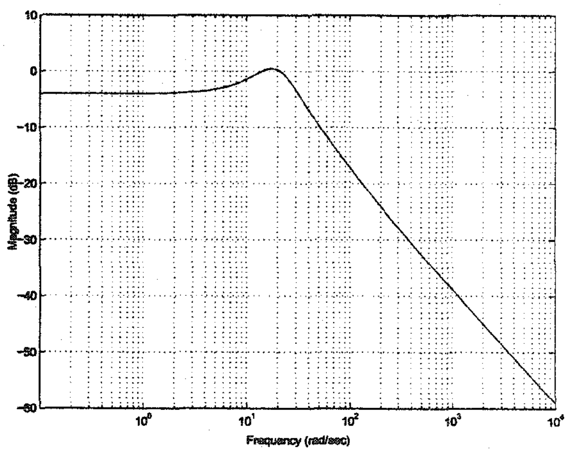


Figure 5.19: $|W_1S|+|W_2T|$ versus frequency (with controller and filter)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Chapter 6

Adaptive ILC for Robot Manipulators

The experimental results of adaptive ILC schemes proposed in [28] are discussed in this section. The control algorithm involves the use of an iteratively updated term designed to take care of the unknown parameters and disturbances, together with a PD controller. In contrast to classical ILC schemes where one needs the same number of iterative variables as the number of control inputs, here we require only two iterative variables for controller implementation.

6.1 Adaptive Scheme

Using the Lagrangian formulation, the equations of motion of a n degrees of freedom rigid manipulator may be expressed by

$$M(q_k)\ddot{q}_k + C(q_k, \dot{q}_k)\dot{q}_k + G(q_k) = \tau_k(t) + d_k(t) \tag{6.1}$$

where $t \in R$ denotes time, $k \in N$ denotes the iteration number. The signals $q_k \in R^n$, $\dot{q}_k \in R^n$, $\ddot{q}_k \in R^n$ are joint position, joint velocity and joint acceleration vectors

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

respectively at the iteration k. $M(q_k) \in R^{n\times n}$ is inertia matrix, $C(q_k, \dot{q}_k)\dot{q}_k \in R^n$ is a vector resulting from Coriolis and centrifugal forces. $G(q_k) \in R^n$ is a vector resulting from gravitational forces. $\tau_k \in R^n$ is the control input vector containing torques and forces to be applied at each joint. $d_k \in R^n$ are the vectors containing the unmodeled dynamics and other unknown external disturbances.

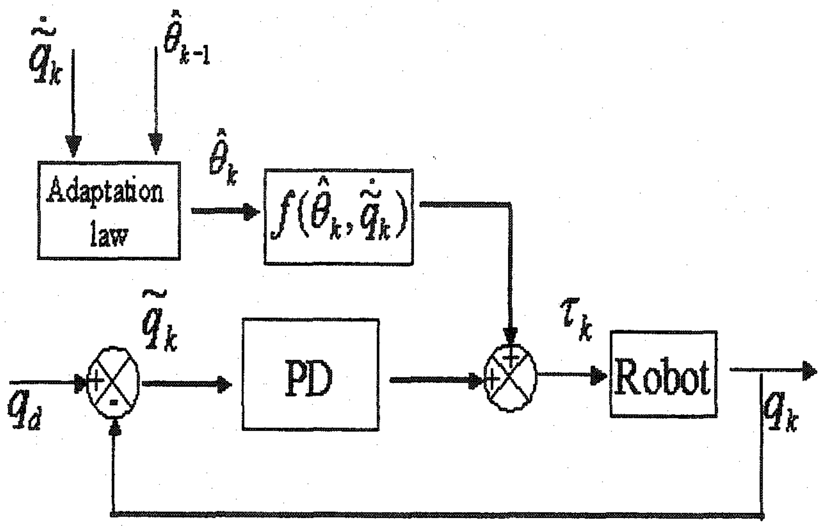


Figure 6.1: Adaptive ILC scheme

The ILC scheme used in this thesis is based on the following assumptions:

$A-1$ The reference trajectory, its first and second time-derivative, $q_d(t)$, $\dot{q}_d(t)$ and $\ddot{q}_d(t)$, and the disturbance $d_k(t)$ are bounded $\forall t \in [0, T]$ and $\forall k \in N$.

$A-2$ The resetting condition is satisfied, *i.e.*, $q_d(0)-q_k(0) = \dot{q}_d(0)-\dot{q}_k(0) = 0, \forall k \in N$.

and the following properties, characteristic to all robot manipulators, are taken into consideration:

i) $M(q_k) \in R^{n\times n}$ is symmetric, bounded and positive definite.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

ii) The matrix $\dot{M}(q_k)-2C(q_k,\dot{q}_k)$ is skew symmetric, hence $x^T(\dot{M}(q_k)-2C(q_k,\dot{q}_k))x = 0, \forall x \in R^n$.

iii) $\|C(q_k,\dot{q}_k)\| \leq k_c\|\dot{q}_k\|$ and $\|G(q_k)\| < k_g, \forall t \in [0,T]$ and $\forall k \in N$, where k_c and k_g are unknown positive parameters.

The adaptive ILC scheme is as explained.

6.1.1 Adaptive ILC Scheme

Considering the system given by (6.1), satisfying the properties i,ii,iii, and the control law [28]

$$\tau_k(t) = K_P\tilde{q}_k(t) + K_D\dot{\tilde{q}}_k(t) + \eta(\dot{\tilde{q}}_k)\hat{\theta}_k(t) \qquad (6.2)$$

with

$$\hat{\theta}_k(t) = \hat{\theta}_{k-1}(t) + \Gamma\eta^T(\dot{\tilde{q}}_k)\dot{\tilde{q}}_k(t), \qquad (6.3)$$

where $\hat{\theta}_{-1}(t) = 0$. The matrices $K_P \in R^{nxn}, K_D \in R^{nxn}$ and $\Gamma \in R^{2x2}$ are symmetric positive definite. The function $\eta(\dot{\tilde{q}}_k) = [\dot{\tilde{q}}_k \; sgn(\dot{\tilde{q}}_k)]$. If the assumptions (A1-A2) are satisfied, then $\tilde{q}_k(t)$, $\dot{\tilde{q}}_k(t)$ and $\tau_k(t)$ are bounded for all $t \in [0,T]$ and all $k \in N$ and $\lim_{k\to\infty} \tilde{q}_k(t) = \lim_{k\to\infty} \dot{\tilde{q}}_k(t) = 0$, $\forall t \in [0,T]$.

To prove the boundedness and the convergence to zero of the tracking error, we use the following composite energy function:

$$W_k(\dot{\tilde{q}}_k(t), \tilde{q}_k(t), \tilde{\theta}_k(t)) = V_k(\dot{\tilde{q}}_k(t), \tilde{q}_k(t)) + \frac{1}{2}\int_0^t \tilde{\theta}_k^T(\tau)\Gamma^{-1}\tilde{\theta}_k(\tau)d\tau \qquad (6.4)$$

with $\tilde{\theta}_k(t) = \theta(t) - \hat{\theta}_k(t)$, where $\theta(t) = [\alpha \;\; \delta]^T \in \mathbb{R}^2$ and $\hat{\theta}_k(t)$ is the estimated value of $\theta(t)$. The unknown parameters α and δ are obtained according to

$$\begin{aligned} \dot{\tilde{q}}_k^T(M(q_k)\ddot{q}_d + C(q_k,\dot{q}_k)\dot{q}_d + G(q_k) - d_k) &\leq \|\dot{\tilde{q}}_k\|(\beta + k_g + k_c\|\dot{q}_d\| \; \|\dot{q}_k\|) \\ &\leq \|\dot{\tilde{q}}_k\|(\beta + k_g + k_c\|\dot{q}_d\|^2 + k_c\|\dot{q}_d\| \; \|\dot{q}_k\|) \qquad (6.5) \end{aligned}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

where k_c, k_g and $\|M(q_k)\ddot{q}_d - d_k\| \leq \beta$ are defined as per the properties of manipulator $i)$, $ii)$ and $iii)$mentioned previously. And from the boundedness of $\dot{q}_d$, we have,

$$\begin{aligned} \dot{\tilde{q}}_k^T(M(q_k)\ddot{q}_d + C(q_k, \dot{q}_k)\dot{q}_d + G(q_k) - d_k) &\leq \dot{\tilde{q}}_k^T(\alpha\dot{\tilde{q}}_k + \delta sgn(\dot{\tilde{q}}_k)) \\ &\leq \dot{\tilde{q}}_k^T\eta(\dot{\tilde{q}}_k)\theta \qquad (6.6) \end{aligned}$$

where $\alpha = k_c Sup_{t\in[0,T]} \|\dot{q}_d\|$ and $\delta = \beta + k_g + k_c Sup_{t\in[0,T]} \|\dot{q}_d(t)\|^2$

This theorem is proved by showing that

$$\Delta W_k = W_k - W_{k-1}$$

is a bounded non-increasing sequence.

The difference is given by

$$\Delta W_k = V_k - V_{k-1} + \frac{1}{2}\int_0^T (\bar{\theta}_k^T\Gamma^{-1}\bar{\theta}_k - \tilde{\theta}_{k-1}^T\Gamma^{-1}\tilde{\theta}_{k-1})d\tau \qquad (6.7)$$

where $\bar{\theta}_k = \hat{\theta}_k - \hat{\theta}_{k-1}$

$$\Delta W_k \leq -V_{k-1} - \frac{1}{2}\int_0^t \dot{\tilde{q}}_k^T(\phi(q_k, \dot{q}_k, \dot{\tilde{q}}_k)\Gamma\phi^T(q_k, \dot{q}_k, \dot{\tilde{q}}_k) + 2K_D)\dot{\tilde{q}}_k d\tau \leq 0 \qquad (6.8)$$

which proves that W_k is a non-increasing sequence. Hence, W_k is bounded if W_0 is bounded.

The boundedness of W_k guarantees the boundedness of $\tilde{q}_k(t)$, $\dot{\tilde{q}}_k(t)$ and $\int_0^t \tilde{\theta}_k^T(\tau)\Gamma^{-1}\tilde{\theta}_k(\tau)d\tau$. The boundedness of W_0 is proved as follows :

$$\dot{W}_0 \leq \dot{\tilde{q}}_0^T(\phi(q_0, \dot{q}_0, \dot{\tilde{q}}_0)\tilde{\theta}_0 - K_D\dot{\tilde{q}}_0) + \frac{1}{2}\tilde{\theta}_0^T\Gamma^{-1}\tilde{\theta}_0 \qquad (6.9)$$

since $\hat{\theta}_{-1}(t) = 0$, we have, $\hat{\theta}_0(t) = \Gamma\phi^T(q_0, \dot{q}_0, \dot{\tilde{q}}_0)\dot{\tilde{q}}_0(t)$. Then,

$$\begin{aligned} \dot{W}_0 &\leq -\dot{\tilde{q}}_0^T K_D\dot{\tilde{q}}_0 + (\hat{\theta}_0^T + \frac{1}{2}\tilde{\theta}_0^T)\Gamma^{-1}\tilde{\theta}_0 \qquad (6.10) \\ &\leq -\dot{\tilde{q}}_0^T K_D\dot{\tilde{q}}_0 - \frac{1}{2}\tilde{\theta}_0^T\Gamma^{-1}\tilde{\theta}_0 + \theta^T\Gamma^{-1}\tilde{\theta}_0 \qquad (6.11) \end{aligned}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Using Young's inequality,

$$\theta^T \Gamma^{-1} \tilde{\theta}_0 \leq \mathcal{K} \|\Gamma^{-1}\tilde{\theta}_0\|_2^2 + \frac{1}{4\mathcal{K}}\|\theta\|_2^2,$$

for $\mathcal{K} > 0$.

$$\dot{W}_0 \leq -\rho_1 \|\dot{\tilde{q}}_0\|_2^2 - \rho_2 \|\tilde{\theta}_0\|_2^2 + \frac{1}{4\mathcal{K}}\|\theta\|_2^2, \tag{6.12}$$

which implies that,

$$\dot{W}_0 \leq \frac{1}{4\mathcal{K}\|\theta\|_2^2} \tag{6.13}$$

where $\rho_1 = \lambda_{min}(K_D)$, $\rho_2 = \frac{1}{2}\lambda_{min}(\Gamma^{-1}) - \mathcal{K}\lambda_{max}^2(\Gamma^{-1})$ with $0 < \mathcal{K} \leq \frac{\lambda_{min}(\Gamma^{-1})}{2\lambda_{max}^2(\Gamma^{-1})}$. The fact that θ is bounded implies the boundedness of W_0 in [0,T].

Now, it could be shown that,

$$W_k \leq W_0 - \sum_{j=1}^{k} V_{j-1} \tag{6.14}$$

$$\leq W_0 - \frac{1}{2}\sum_{j=1}^{k} \tilde{q}_{j-1}^T K_P \tilde{q}_{j-1} - \frac{1}{2}\sum_{j-1}^{k} \dot{\tilde{q}}_{j-1}^T M(q_{j-1})\dot{\tilde{q}}_{j-1}, \tag{6.15}$$

which implies

$$\frac{1}{2}\sum_{j=1}^{k} \tilde{q}_{j-1}^T K_P \tilde{q}_{j-1} - \frac{1}{2}\sum_{j-1}^{k} \dot{\tilde{q}}_{j-1}^T M(q_{j-1})\dot{\tilde{q}}_{j-1} \leq W_0 - W_k \leq W_0.$$

Hence $\tilde{q}_k(t)$ and $\dot{\tilde{q}}_k$ are bounded
$\forall\, t \in [0, T]$, $\forall\, k \in \mathcal{N}$ and
$\lim_{k\to\infty} \tilde{q}_k(t) \;=\; \lim_{k\to\infty} \dot{\tilde{q}}_k(t) \;=\; 0$, $\forall\, t \in [0, T]$.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

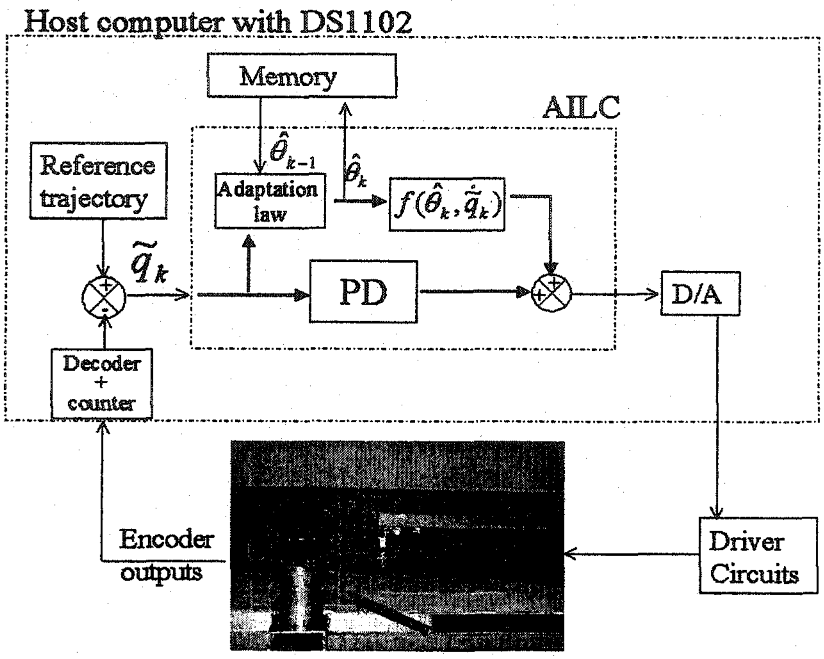


Figure 6.2: Experimental setup for adaptive ILC

6.2 Experimental Results

The adaptive ILC scheme discussed in (6.1.1) was applied to the 2-DOF planar manipulator. The experimental setup is as shown in Figure 6.2. In the experiment, low pass filters of cut off frequency 6 rad/sec and 4 rad/sec were used with the numeric differentiator for link1 and link2 respectively. The proportional gain K_P, the derivative gain K_D and the learning gain Γ were chosen as

$$K_P = \begin{bmatrix} 0.02 & 0 \\ 0 & 0.005 \end{bmatrix}, \quad K_D = \begin{bmatrix} 0.0002 & 0 \\ 0 & 0.0001 \end{bmatrix} \quad \Gamma = \begin{bmatrix} 0.0025 & 0 \\ 0 & 0.0025 \end{bmatrix}$$

The experiment was realised using SIMULINK/dSpace at a sampling period of $4msec$. A circular trajectory, centered at (0.5m, 0) with a radius of 0.3m, was chosen as a

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

reference for our experiment. That is,

$$x(t) = 0.5 + r\cos\omega t \qquad y(t) = r\sin\omega t,$$

where $\omega = 2\pi/T$ and r being the radius of the circle, ω the angular velocity and T the period of trial. In our experiment $\omega = 0.3142\ rad/sec$ and $T = 20\ sec$ were chosen. The setpoint to the actuators is obtained from inverse kinematics of the chosen reference trajectory and the resetting condition, $y_d(0) = y_k(0) = 0$ is satisfied on all iterations. The Figures 6.3, 6.4, 6.7 show the error convergence and output trajectories, with Figures 6.5 and 6.6 showing the joint positions of each of the two joints.

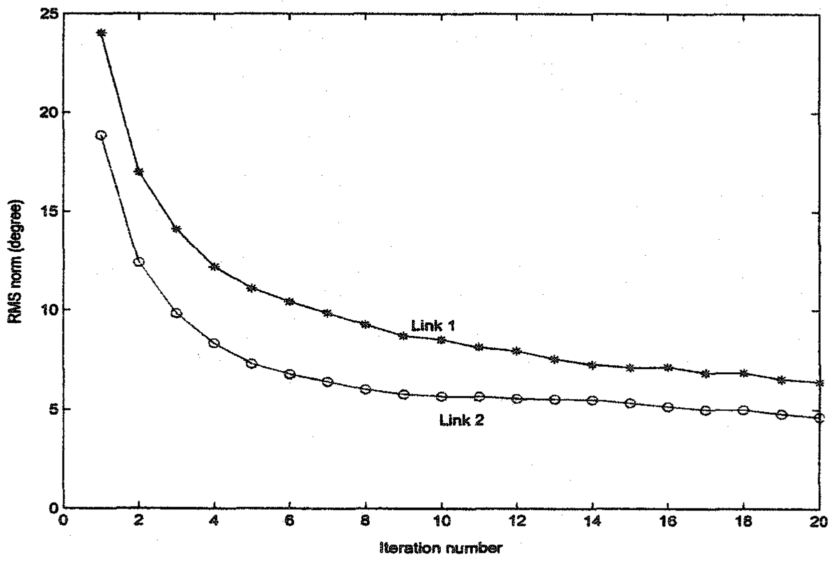


Figure 6.3: RMS norm of the tracking error versus the iteration number for link 1 and link 2

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

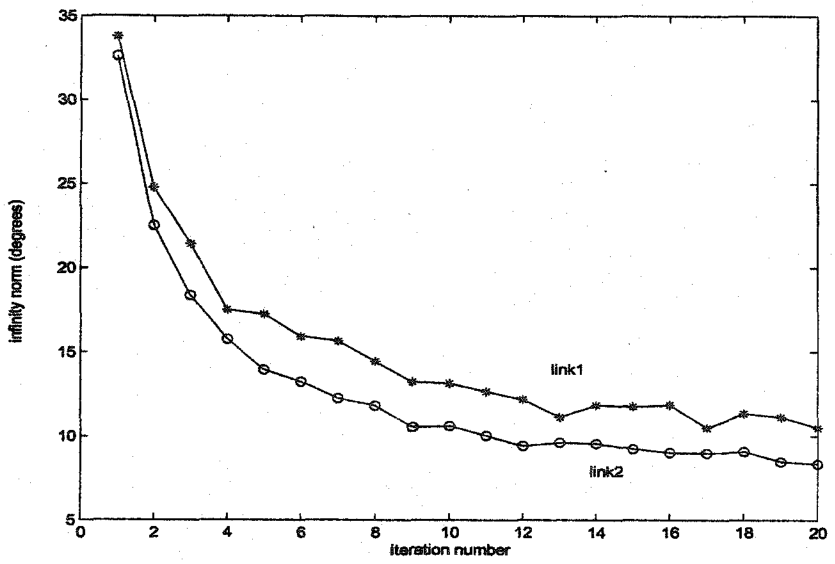


Figure 6.4: Infinity norm of the tracking error versus the iteration number for link 1 and link 2

6.3 Observations

Following observations are made based on the experiments performed on the planar manipulator:

1. The main advantage of this ILC scheme is that no knowledge of system parameters is required.

2. The convergence of the iterative process is guaranteed by the positive definiteness of K_P, K_D and Γ.

3. Another important feature of this ILC scheme is the presence of only two iterative parameters. The smaller number of iterative parameters helps in saving memory space during real-time implementation.

4. Acceleration measurements are not required for controller implementation.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

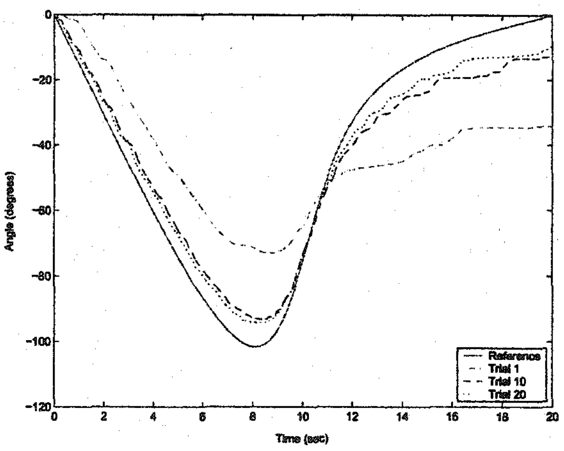


Figure 6.5: Joint positions for link1

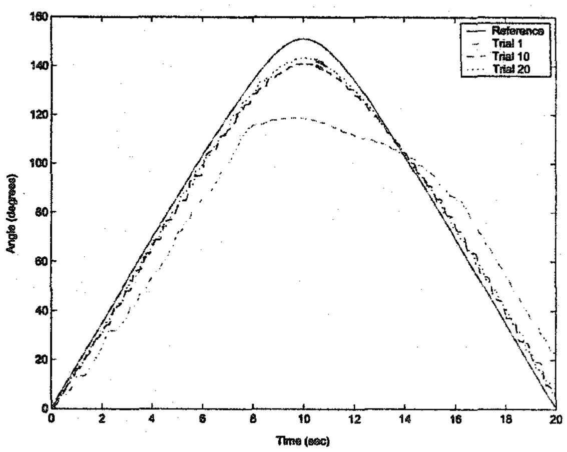


Figure 6.6: Joint positions for link2

5. The chattering occuring at higher iterations could be reduced by replacing signum function with a saturation block given by :

$$sat(x) = \begin{cases} 1 & for \quad x \geq \acute{x} \\ -1 & for \quad x \leq \acute{x} \\ x & for \quad |x| < \acute{x} \end{cases}$$

where x is the input and $\acute{x}$ is the limit of the saturation block.

6. A faster rate of convergence could be achieved by increasing the value of Γ, but this causes chattering due to amplification of noise present with the updating term.

7. At very low cut-off frequencies of low-pass filter, the system exhibits oscillatory behaviour.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

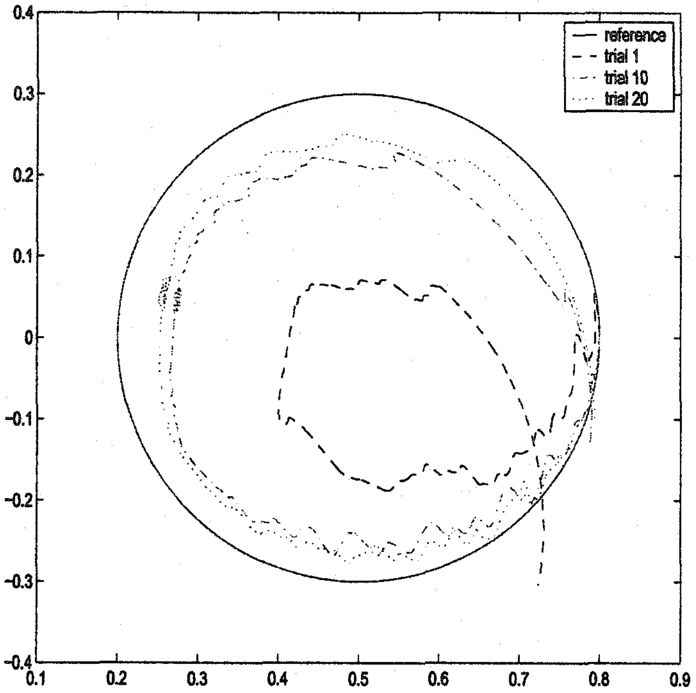


Figure 6.7: End-effector positions

6.4 Summary

- The adaptive ILC scheme approach works satisfactorily on the 2-DOF robot manipulator, though error convergence is slow. But increasing the Γ value for a faster convergence results in chattering as iteration progresses.

- The chattering of the system occurs at higher iteration numbers, and this could be minimized by reducing the cut-off frequency of the low pass filter.

- Another possible method of reducing the chattering is by using the saturation block instead of the signum function.

- Compared to classical ILC schemes where number of updating terms are equal

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

to the number of the control inputs, here we use just two iterative variables, irrespective of the degrees of freedom of manipulator.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

In this thesis the experimental implementation of ILC schemes were discussed and advantages in practical applications were studied. The following conclusions could be drawn based on this work:

- Expressing ILC problem as a robust control problem provides one with better options for controller design.

- In the robust ILC technique, the choice of weighting function W_1 close to one guarantees the error convergence to a small value.

- The adaptive control strategy discussed provides one with error convergence with minimum knowledge on the system parameters. This could be done with an additional advantage of lesser memory requirement than existing ILC techniques.

- In the implementation of the adaptive ILC, chattering occurs with increasing number of iterations due to noise amplification. This could be minimized to a

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

certain extent using a lowpass filter along with the numeric differentiator.

- Apart from the measurement noise, the signum function used in the adaptive ILC scheme, can cause chattering in the presence of noise. The use of a saturation block helps to reduce the chattering occuring due to the above mentioned reason.

- These learning procedures could provide better results with smaller sampling periods. But this depends greatly on the memory capacity of the data acquisition board. In this case one needs to compromise on the sampling period and the period of operation. In our case, implementation at smaller sampling periods could not be undertaken because of the memory constraint on the DAQ board.

7.2 Future Work

Based on the experimental work done, the following recommendations could be proposed for the future work:

- The learning algorithms could be implemented at lower sampling periods for obtaining better results.

- Future work could be done to develop an algorithm with only position feedback, thereby reducing the chances of noise entry into the system.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Bibliography

[1] S. Arimoto, S. Kawamura and F. Miyazaki, "Bettering operation of dynamic systems by learning". *Proceedings of 23rd Conference on Decision and Control, Las Vegas*, pp. 1064-1069, 1984.

[2] S. Arimoto, *Control theory of non-linear mechanical systems*, Oxford, 1996.

[3] G. J. Balas, J. C. Doyle, K. Glover, A. Pickard and R. Smith, *μ-Analysis and Synthesis Toolbox*, 2001.

[4] Z. Bien and J. X. Xu, *Iterative Learning Control: Analysis, Design, Integration and Applications*, Kluwer Academic Publishers, 1998.

[5] Y. Chen, C. Wen, Z. Gong and M. Sun, "An iterative learning controller with initial state learning". *IEEE Transactions on Automatic Control*, **44(2)**, pp. 371-376, 1999.

[6] J. J. Craig, "Adaptive Control of manipulators through repeated trials". *Proceedings of the American Control Conference*, San Diego, pp. 1566-1574, 1984.

[7] I. K. Craig and E. de Klerk, "An assignment to teach closed-loop system identification". *IFAC* 15*th* *Triennial World Congress*, Barcelona, Spain, 2002.

[8] J. C. Doyle and B. A. Francis, *Feedback Control Theory*, Macmillan Publishing Co. 1990.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

[9] M. French and E. Rogers, "Non-linear iterative learning by an adaptive Lyapunov technique". *International Journal of Control*, **73(10)**, pp. 840-850, 2000.

[10] M. Green and D. J. N. Limebeer, *Linear Robust Control*, Prentice-Hall Inc. 1995.

[11] R. Horowitz, "Learning control applications to mechatronics". *JSME International Journal*, **37(3)**, pp. 421-429, 1994.

[12] S. N. Huang, K. K. Tan and T. H. Lee, "Necessary and sufficient condition for convergence of iterative learning algorithm". *Automatica*, **38**, pp. 1257-1260, 2002.

[13] I. D. Landau, "Identification in closed loop: a powerful design tool (better design models, simpler controllers)". *Control Engineering practice*, Vol.9, pp. 51-65, 2001.

[14] H. S. Lee and Z. Bien, "Study on robustness of iterative learning control with non-zero initial error". *International Journal of Control*, **64(3)**, pp. 345-359, 1996.

[15] L. Ljung, *System Identification Toolbox*, 2001.

[16] R. Longman, "Iterative learning control and repetitive learning control for engineering practice". *International Journal of Control*, **73(10)**, pp. 930-954, 2000.

[17] J. H. Moon, T. Y. Doh and M. J. Chung,"An iterative learning control scheme for manipulators". *Proc. of International Conference on Intelligent Robots and Systems*, Grenoble, France, 1997.

[18] J. H. Moon, T. Y. Doh and M. J. Chung, "A Robust approach to iterative learning control design for uncertain systems". *Automatica*, **34(8)**, pp. 1001-1004, 1998.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

[19] K. L. Moore, M. Dahleh and S. P. Bhattacharyya, "Iterative learning control : a survey and new results". *Journal of Robotic Systems*, **9(5)**, pp. 563-594, 1991.

[20] K. H. Park and Z. Bien, "A generalized iterative learning controller against initial state error". *International Journal of Control*, **73(10)**, pp. 871-881, 2000.

[21] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control*, John Wiley & Sons Inc. 1996.

[22] J. J. Slotine and W. Li, *Applied Nonlinear Control*, Prentice Hall International, 1991.

[23] M. W. Spong and M. Vidyasagar, *Robot Dynamics and Control*, John Wiley & Sons Inc 1989.

[24] M. Takegaki and S. Arimoto, "A new feedback method for dynamic control of manipulators". *Journal of Dynamic Systems, Measurement and Control*, Vol. 103, pp. 119-125.

[25] A. Tayebi and M.B. Zaremba, "Internal Model-Based Robust Iterative Learning control for Uncertain LTI Systems". *In proc. of the 39th IEEE Conference on Decision and Control*, pp. 3439-3444, 2000.

[26] A. Tayebi and M. B. Zaremba, "Robust iterative learning control design for uncertain LTI systems via μ-synthesis", Technical report 2001.

[27] A. Tayebi and M.B. Zaremba, "Robust Iterative Learning Control Design is Straightforward for Uncertain LTI Systems Satisfying the Robust Performance Condition". *IEEE Transactions on Automatic Control*, **48(1)**, pp. 101-106, 2003.

[28] A. Tayebi. "Adaptive iterative learning control for robot manipulators". *Automatica*, vol. 40,No.7, 2004.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

[29] J. X. Xu, B. Vishwanathan and Z. Qu, "Robust learning control for robotic manipulators with an extension to a class of non-linear systems". *International Journal of Control*, **73(10)**, pp. 858-870, 2000.

[30] K. Zhou and J. C. Doyle, *Essentials of Robust Control* Prentice Hall, 1997.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Appendix A

Circuitry

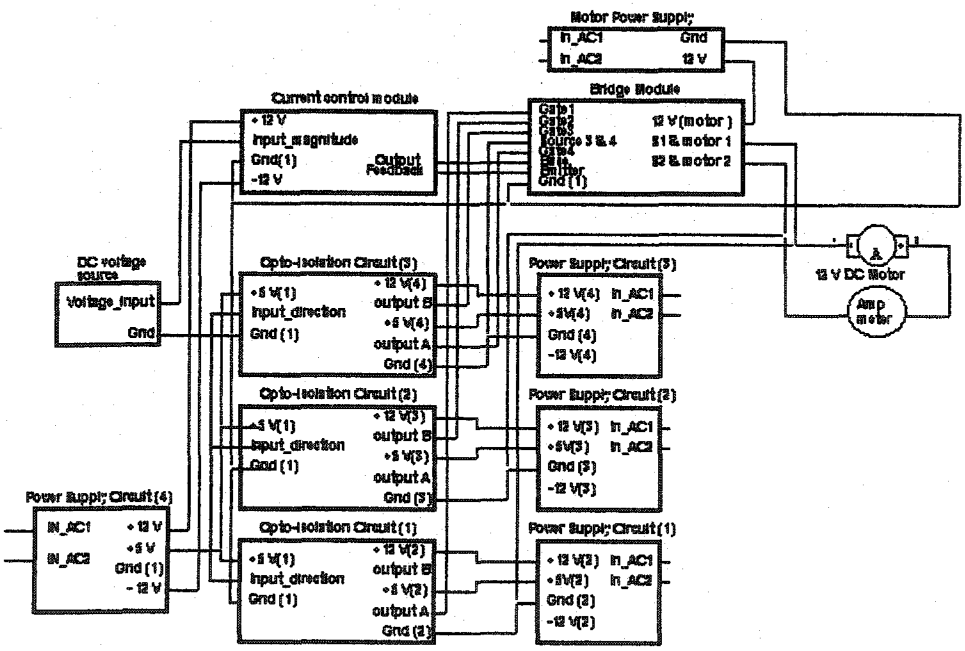


Figure A.1: Overall test circuitry

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

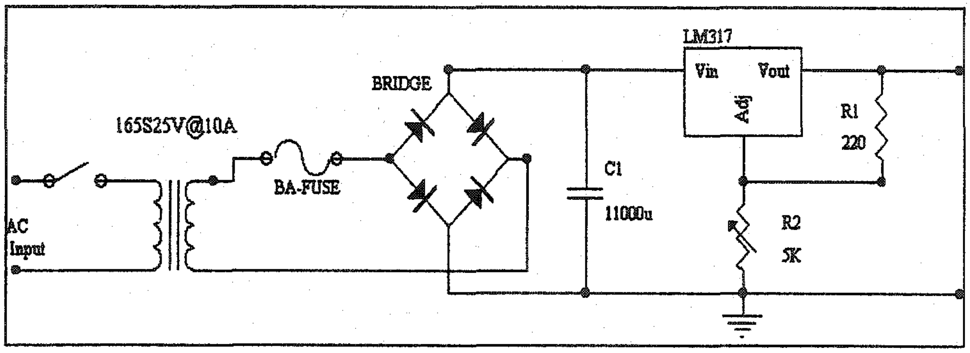
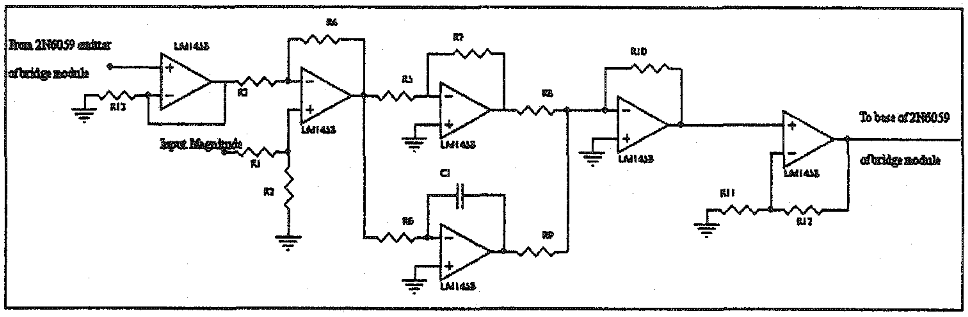


Figure A.2: Motor Power Supply Module

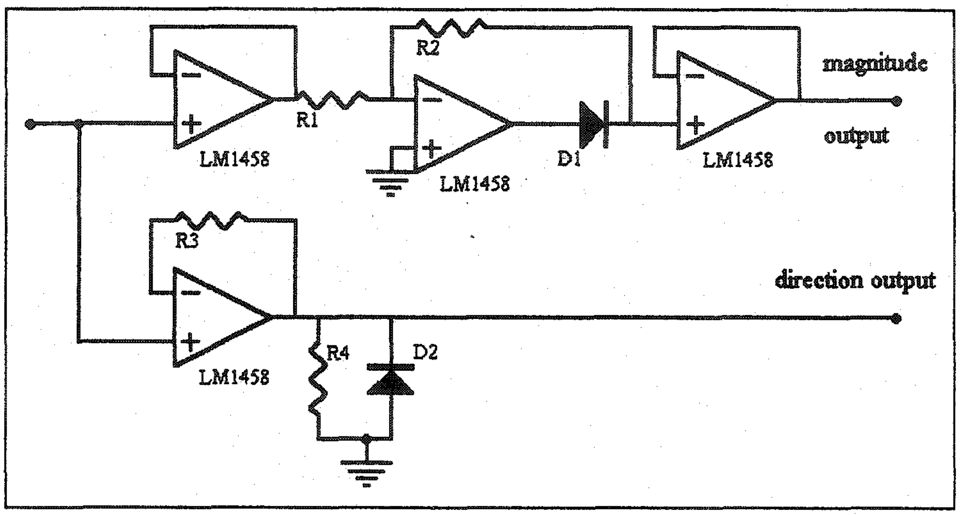


R1 = R2 = R3 = R4 = 2.2k   R5 = 180   R6 = R7 = 18k   C1 = 12nF

R8 = R9 = R10 = 1.2k   R11 = 82k   R12 = 5.6k   R13 = 1k

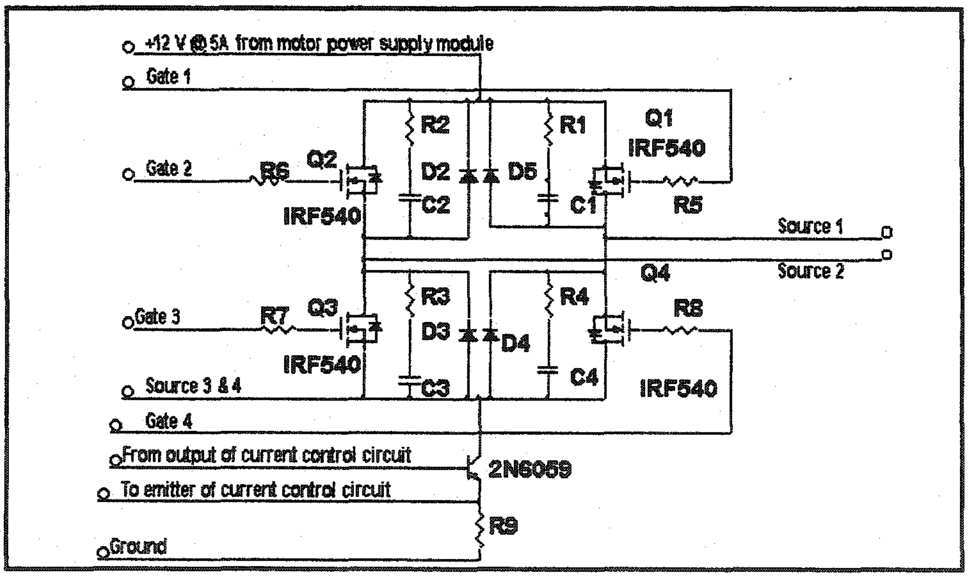
Figure A.3: Current Control Module

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.



R1 = R2 = 10k   R3 = 2.2k   R4 = 1k

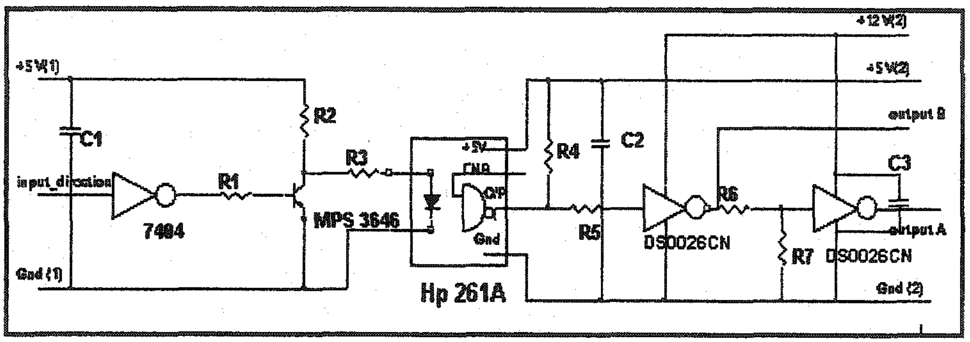
Figure A.4: Signal Separation Module



R1 = R2 = R3 = R4 = R9 = 1 ohm   R5 = R6 = R7 = R8 = 12 ohm   C1=C2=C3=C4=2.2 uF

Figure A.5: Bridge Circuit

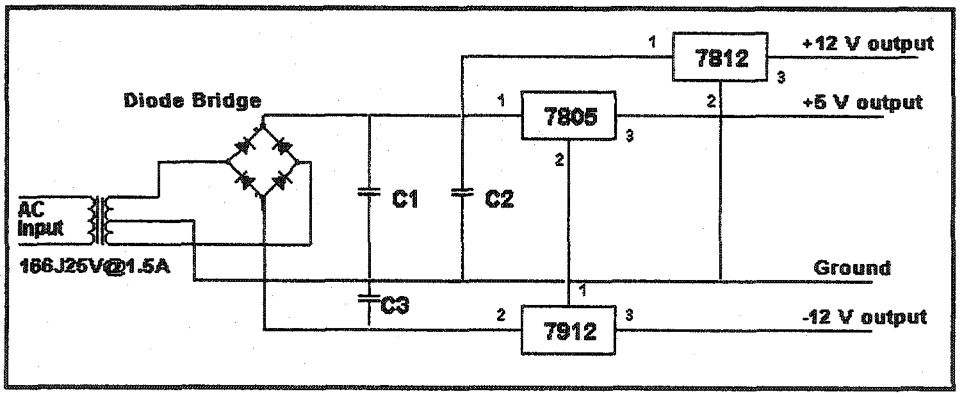
Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.



C1 = C2 = C3 = 0.1 uF R1 =4 .7 kohm R2 = 580 ohm R3 =160 ohm

R4 =1.0 kohm R5 = 270 ohm R6=10.0 kohm R7 =5.6 kohm

Figure A.6: Opto-isolator Module



C1 = 470 uF C2 = 2700 uF C3= 470 uF

Figure A.7: Power Supply Module

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Appendix B

Denavit-Hartenberg Representation

Denavit-Hartenberg or D-H convention is a commonly used method for selecting frames of reference in robots. In this convention, each homogeneous transformation A_i is represented as a product of four basic transformations

$$\begin{aligned} A_i &= Rot_{z,\theta_i}\, Trans_{z,d_i}\, Trans_{x,a_i}\, Rot_{x,\alpha_i} \\ &= \begin{bmatrix} c\theta_i & -s\theta_i c\alpha_i & s\theta_i s\alpha_i & a_i c\theta_i \\ s\theta_i & c\theta_i c\alpha_i & -c\theta_i s\alpha_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{B.1}$$

where θ_i, a_i, α_i and d_i are parameters of link i and joint i. These parameters are generally referred by the following names: a_i as the 'length', α_i as the 'twist', d_i is called the 'offset' and θ_i is called the 'angle'. Since the matrix A_i is a function of a single variable, the other three parameters are constant for a given link. The varying parameter, θ_i for a revolute joint and d_i for a prismatic joint is called the joint vari-

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

able.

In this procedure, a homogeneous transformation A_i is used to express the transformation of coordinates from one frame to another. In the usual representation, to express this transformation, six numbers are required *i.e.* three for displacement vector and three for the Euler angles corresponding to the rotation matrix. But in D-H representation, only four parameters are required for expressing the transformation. This is made possible by the following two features :

- The axis x_i is perpendicular to the axis z_{i-1}
- The axis x_i intersects the axis z_0.

For deriving the forward kinematics for any manipulator with n joints, using the D-H convention, one needs to follow some procedures, namely:

Step 1 : Locate and label the joint axes $z_0,...,z_{n-1}$.

Step 2 : Establish the base frame. Set the origin anywhere on the z_0-axis. The x_0 and y_0 axes are chosen to form a right-hand frame.
For $i = 1, \ldots, n-1$ Steps 3 to 5 are performed.

Step 3 : Locate the origin O_i where the common normal to z_i and z_{i-1} intersects z_i. If z_i intersects z_{i-1} locate O_i at this intersection. If z_i and z_{i-1} are parallel, O_i is located at joint i.

Step 4 : Establish x_i along the common normal between z_{i-1} and z_i through O_i, or in the direction normal to the $z_{i-1} - z_i$ plane if z_{i-1} and z_i intersect.

Step 5 : Establish y_i to complete a right-hand frame.

Step 6 : Establish the end-effector frame $O_n x_n y_n z_n$. Assuming the n^{th} joint is revolute, choose z_n along the direction of z_{n-1}. Establish the origin O_n conveniently along z_n. Set x_n and y_n to form the right-hand frame.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Step 7 : Create a table of link parameters a_i, d_i, α_i, θ_i.

a_i = distance along x_i from O_i to the intersecion of the x_i and z_{i-1} axis.

d_i = distance along z_{i-1} from O_{i-1} to the intersection of the x_i and z_{i-1} axes. d_i is variable if joint i is prismatic.

α_i = the angle between z_{i-1} and z_i measured about x_i.

θ_i = the angle between x_{i-1} and x_i measured about z_{i-1}.
θ_i is variable if joint i is revolute.

Step 8 : Form the homogeneous transformation matrix A_i using the above parameters.

Step 9 : Form $T_0{}^n = A_1A_2...A_n$. This gives the position and orientation of the tool frame expressed in base coordinates.

B.1 2-DOF Planar Robot Manipulator

Considering the two-link planar robot manipulator shown in Figure B.1. The joint axes z_0 and z_1 are normal to the page. Proceeding as per the steps for DH conventions, the base frame $O_0x_0y_0z_0$ is established. The origin O_0 is chosen at the intersection of z_0 axis with the plane and x_0 is chosen arbitrarily. Once the base frame is established, the frames $O_1x_1y_1z_1$ and $O_2x_2y_2z_2$ are chosen as per the DH conventions. The link parameters are then determined to express the homogeneous tranformation matrix as per (B.1).

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

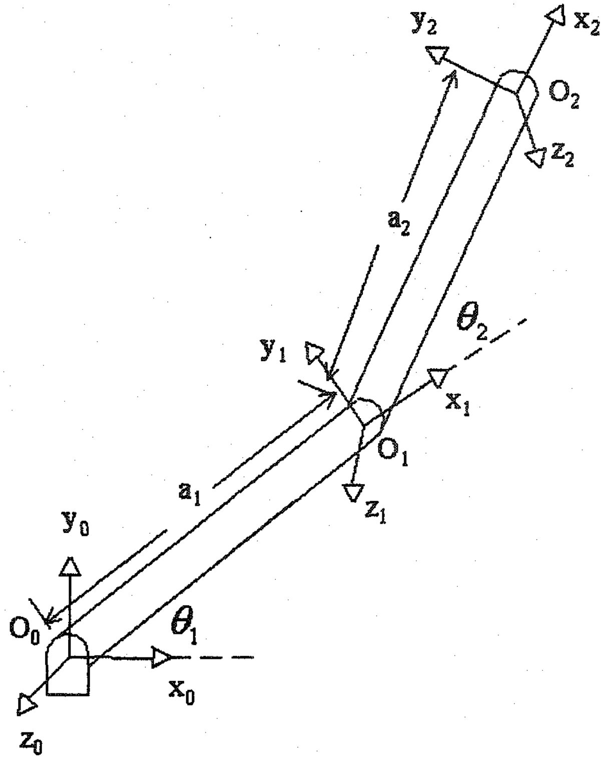


Figure B.1: Two-link planar robot manipulator

DH Parameters for Robot				
Link	a_i	d_i	α_i	θ_i
1	a_1	0	0	$\theta_1{}^*$
2	a_2	0	0	$\theta_2{}^*$

$$A_1 = \begin{bmatrix} c_1 & -s_1 & 0 & a_1c_1 \\ s_1 & c_1 & 0 & a_1s_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad A_2 = \begin{bmatrix} c_2 & -s_2 & 0 & a_2c_2 \\ s_2 & c_2 & 0 & a_2s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The transformation matrices ${T_0}^1 = A_1$ and ${T_0}^2$ are given by,

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

${T_0}^1 = A_1$ and ${T_0}^2 = A_1 A_2$

$$
{T_0}^2 = \begin{bmatrix} c_{12} & -s_{12} & 0 & a_1 c_1 + a_2 c_{12} \\ s_{12} & c_{12} & 0 & a_1 s_1 + a_2 s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

where,

$$
\begin{aligned}
c_1 &\triangleq \cos\theta_1, & s_2 &\triangleq \sin\theta_2 \\
c_2 &\triangleq \cos\theta_2, & c_{12} &\triangleq \cos(\theta_1 + \theta_2) \\
s_1 &\triangleq \sin\theta_1, & s_{12} &\triangleq \sin(\theta_1 + \theta_2)
\end{aligned}
$$

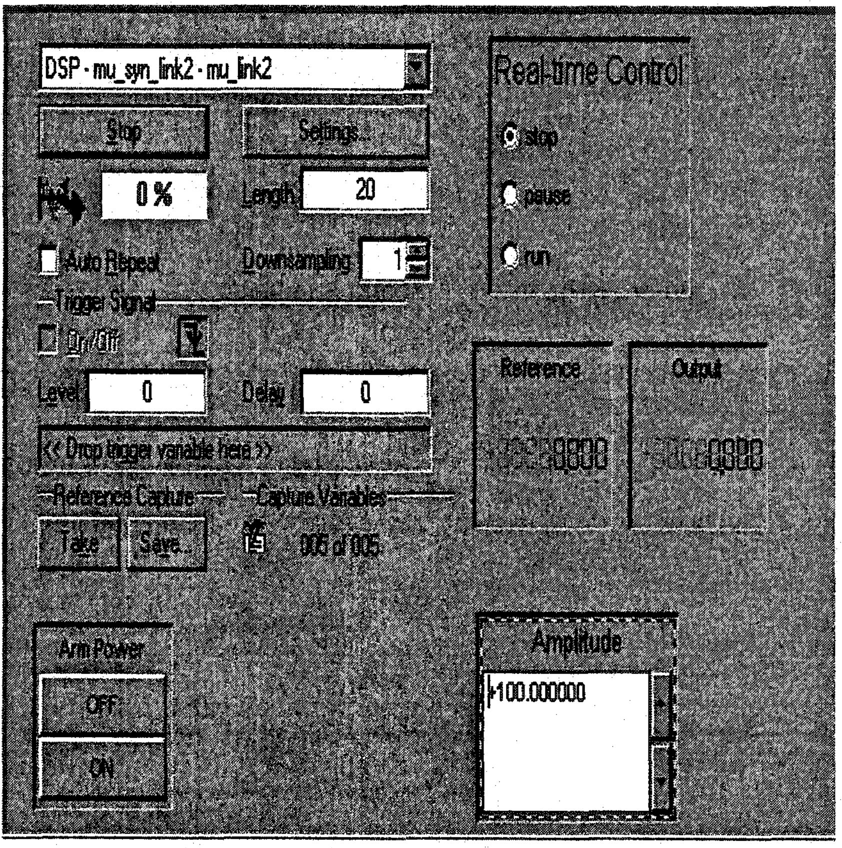
The rotational part of ${T_0}^2$ gives the orientation of frame $O_2 x_2 y_2 z_2$ with respect to the base frame and the first three elements of the fourth column of ${T_0}^2$ gives co-ordinates of end-effector.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Appendix C

GUI for Real-time Application

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.



DSP - mu_syn_link2 - mu_link2

0 %

Length: 20

□ Auto Repeat

Downsampling: 1

Trigger Signal

□ On/Off

Level: 0

Delay: 0

<< Drop trigger variable here >>

Reference Capture

Capture Variables

005 of 005

Real-time Control

● stop

○ pause

○ run

Amplitude

+100.000000

Figure C.1: Layout schematic for implementation in control desk

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.