SIMULATION OF ROBOT MANIPULATOR CONTROL STRATEGIES.

PHILIP ANDREW ROUND

Doctor of Philosophy

THE UNIVERSITY OF ASTON IN BIRMINGHAM

December 1988

The University of Aston in Birmingham

Simulation of Robot Manipulator Control Strategies

Philip Andrew Round

Doctor of Philosophy 1988

Summary

The high capital cost of robots prohibit their economic application. One method of making their application more economic is to increase their operating speed. This can be done in a number of ways e.g. redesign of robot geometry, improving actuators and improving control system design. In this thesis the control system design is considered.

It is identified in the literature review that two aspects in relation to robot control system design have not been addressed in any great detail by previous researchers. These are : how significant are the coupling terms in the dynamic equations of the robot and what is the effect of the coupling terms on the performance of a number of typical independent axis control schemes?. The work in this thesis addresses these two questions in detail.

A program was designed to automatically calculate the path and trajectory and to calculate the significance of the coupling terms in an example application of a robot manipulator tracking a part on a moving conveyor. The inertial and velocity coupling terms have been shown to be of significance when the manipulator was considered to be directly driven. A simulation of the robot manipulator following the planned trajectory has been established in order to assess the performance of the independent axis control strategies. The inertial coupling was shown to reinforce the control torque at the corner points of the trajectory, where there was an abrupt demand in acceleration in each axis but of opposite sign. This reduced the tracking error however, this effect was not controllable. A second effect was due to the velocity coupling terms. At high trajectory speeds it was shown, by means of a root locus analysis, that the velocity coupling terms caused the system to become unstable.

Key Words: **Robot Manipulator, Control, Simulation, Manufacturing, Path and Trajectory Planning.**

ACKNOWLEDGEMENTS

I would like to thank the following:

Professor K. Foster who supervised the project and provided many words of encouragement and advice throughout the research.

Mr. G.A. Jones who assisted in the supervision during the final year of the research when Professor Foster moved to Birmingham University, and provided encouragement and advice in preparing the thesis for submission.

I would like to take this opportunity to express my thanks to many members of the staff at the University of Aston who have freely given their time to assist me in this project.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1:Introduction

Chapter 1-Introduction

In the age of automation very many tasks that were previously performed by human workers are now performed by automated machinery. One of the most flexible of these is the robot manipulator. It has a wide variety of applications in processing, assembly, inspection and can operate in very hazardous environments otherwise unsuitable for the human worker. The robot manipulator manages to replace the human worker by emulating some of the characteristics of the human worker. For instance the manipulator resembles the human arm having shoulder, elbow, wrist and hand. It has six degrees of freedom that enables the manipulator to place its hand in any desired position and orientation within its reach capacity. Its highly sophisticated computer control system allows it to perform complicated tasks and motions and it can be readily reprogrammed to perform new tasks. With the introduction of sensory abilities including touch(i.e force feedback), vision and artificial intelligence it can be employed in even more difficult tasks. The origins of the robot manipulator however, lie in two pieces of different equipment. These are the teleoperator and the numerically controlled machine tool. It is interesting to consider how the technology of these machines resulted in the current robot technology.

The teleoperator was developed from the need to handle radio active waste that would be harmful for the human to handle. It was a device which was a substitute for the operator's hand and consisted of a pair of tongs(referred to as the slave) on the inside of the work area and two handles(referred to as the master) that operated the tongs from the outside of the work area. The tongs and handles were connected together by a six degree of freedom mechanism to provide the master and slave with freedom of position and orientation. The mechanism replicated in the slave the movements made by the master. The first servo powered teleoperator was developed in 1947.

The other development was the numerically operated machine tool. This was developed from the need to machine aircraft parts designed to be machined from solid rather than riveted together from separate parts. A research program initiated in 1949 by MIT lead to the development of a numerically controlled milling machine. This combined the sophisticated servo system technology and the newly developed digital computer technology. The part to be machined could be stored in

digital form on a punched tape. This was then fed to the machine and converted into machining operations by the the computer control system.

The independent development of these two different technologies led to the idea of combining them to produce the first industrial robot. This was demonstrated by George Devol in the 1960's. Early robots could perform very simple tasks, for instance they would have to be taught a sequence of movements by taking them through a sequence of task positions. These task positions were stored in digital memory, and then they could be played back as many times as was required. The early robot has evolved a great deal since then and has become a very sophisticated piece of automated machinery. It now appears in a variety of configurations including, Cartesian, polar, cylindrical, jointed(the one most resembling the human arm) and SCARA. It can be programmed to emulate the intricate movements of a skilled paint sprayer or follow complex trajectories specified by the user that are automatically computed by the inbuilt trajectory programmer. It can operate at high speeds and with a high degree of accuracy and repeatibility compared to the early robots.

The underlying characteristic of tasks that robots now perform instead of human workers is that of repetitive motions. This means that human workers can be redeployed in more interesting tasks that requires their intelligence and their abilities. Typical repetitive tasks where robots can be employed are parts assembly and parts transfer etc. The benefits of employing robots include reduced production costs and increased productivity. A robot can operate 98% of its time without rest and with only the minimum of maintenance while human workers require rest periods and they can suffer from fatigue. Robots can produce a higher percentage of good assemblies than human workers. This is because they repeat the same procedure exactly the same and do not make mistakes due fatigue. They can perform tasks at a higher speed than humans. For instance a robot arc welder can maintain an average rate of 750 mm/min along a straight line. A skilled welder doing the same job at best would average 250mm/sec. Another example of increased productivity is in the paint spraying of automobiles. Two paint spraying robots in an automobile assembly line take 90 seconds to paint a complete car body inside and out with two coats of paint using the wet on wet principle. A skilled paint sprayer cannot achieve this rate or quality of work and would take between 15-30 mins to complete the same job. Thus it can be seen the robot can be beneficially be employed in a range of jobs that were previously performed by

human workers and can achieve better results.

This is not the case when the economics of replacing dedicated automated machinery with robots is considered. For instance robots are used in the handling of parts in press working operations. In this application the robot loads flat blanks of material onto the press for a stamping operation. One of the limiting factors in using industrial robots for press loading is the cycle time of the press. Cycle times of less than a second are not uncommon in press working. These cycle times are far too fast for currently available commercial robots. There is a direct relationship between part size and the press cycle time required to make the part. Normally larger presses are required to stamp larger parts and so they are a lot slower. Robots can only really be deployed on the slower machines.

Another instance where the limitations of the robot speed limit its economical application is in assembly line work. In an assembly line e.g in the automobile industry, robots may be placed along side the moving production line. In a continuously moving production line the parts move relative to the robot along a conveyor at a constant speed. The robot must be able to maintain correct programmed postion, orientation of the end effector and motion velocity in relation to the moving part in order that a successful assembly operation is carried out on the part. The cycle time for the assembly operation must be set to within the robot's capability hence this determines the production line speed. Existing robots cannot operate at very high speeds with sufficient accuracy to make this application an economic one.

The application of industrial robots to parts transfer and assembly operations is limited by their high cost. A rating scale used by the manufacturing industry to assess economic viability of robots is the ratio of the number of operations they can perform per unit time to their cost. The more cycles the robot can perform per unit time the more rapidly it will payback the capital investment. There is a threshhold for the rating value when the application of the robots to a wide range of new tasks would become economically viable. The increase in variety of applications would mean that robots could be mass produced in large quantities and that would result in a considerable drop in their unit cost. In order to increase the rating scale the cycle time must be dropped. This means increasing the speed of operation of the robot manipulator.

The robot manipulator can be represented by a set of second order or higher, non linear, coupled equations. Designing a suitable control system for the robot is not easy since the well developed control design methods were designed for linear systems. Various control strategies have been postulated by researchers, one of these is independent axis design. In this design method each joint of the robot is modelled as an effective inertia and coupling between joints is ignored. The classical design methods are then applied to the simplified model of the manipulator. Current industrial robots using this design method, such as the Puma robot described by Fu[7], cannot operate at high speeds without exibiting significance degradation in their performance. Various other approaches include the Computed torque method by Paul[8] and Bejczy[9], the near minimum time controller by Kahn[12], the variable structure approach by Young[13] and the model reference adaptive method applied by Dubowsky[23], just to mention a few.

It is identified in the literature survey that two aspects in relation to robot control system design have not been addressed in any great detail by previous researchers. These are : How significant are the coupling terms in the dynamic equations of the robot manipulator, and what is the effect of the coupling terms on the performance of a number of typical independent axis control schemes?. The work in this thesis thus addresses these two questions in detail in order to obtain a greater understanding of the control system design problem for robot manipulators.

Chapter 2:Kinematics

Chapter 2-Robot Kinematics

An understanding of robot manipulator kinematics is essential in order to plan a suitable trajectory for the manipulator to follow. The trajectory information is an integral part of the control system, and provides the reference joint angles, velocities, accelerations and possibly rate of change of acceleration, as demand signals for each joint axis actuator to follow. In this chapter the kinematics of a general six degree of freedom manipulator is considered. The kinematics of the manipulator may be divided into two distinct sections, the Forward kinematics and the Inverse kinematics. The inverse kinematic problem is the more difficult to obtain for a manipulator and a number of possible approaches are discussed. The particular solution for the RTX manipulator(ref.4) is obtained including the Jacobian, the derivative of the Jacobian and their corresponding inverses. The limitations of the solution is considered in terms of workspace and singularities in order for implementaion into the trajectory planning program of Chapter 5.

Forward Kinematics

The Forward kinematic problem is obtaining the position of the end effector in Cartesian coordinates given the respective joint angles i.e.

$$(X,Y,Z,\phi_x,\phi_y,\phi_z)=\text{function}(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6)$$

where $\theta_1 \ldots\ldots \theta_6$ are the joint angles(for a prismatic joint this is a linear position variable)

X,Y,Z are the Cartesian coordinates

ϕ_x,ϕ_y,ϕ_z are the orientation angles

Inverse Kinematics

The inverse kinematic problem is; given a desired position and orientation of the end effector, what are the manipulator joint angles ?.

$$(\theta_1 \ldots\ldots \theta_6)=\text{function}(X,Y,Z,\phi_x,\phi_y,\phi_z)$$

2.1 Forward Kinematics

For simple manipulators the Forward kinematic relationship can be obtained quite simply. However, for manipulators having many degrees of freedom and complicated geometry the kinematic equations are not so easily deduced by inspection. A methodology for obtaining the kinematic equations, using homogeneous transformations was developed by Denavit[1] to describe linkages. This was later employed to describe manipulators by Pieper[2] and Paul[3], the method will be described.

It is necessary to first consider the homogeneous matrix, this can be used to describe the position and orientation of an object in space. The basic structure of the homogeneous matrix is:
N.B. The notation for this transform is defined in appendix 1.

$$T = \begin{bmatrix} n_x & s_x & a_x & p_x \\ n_y & s_y & a_y & p_y \\ n_z & s_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The 3 by 3 left hand row column sub-matrix represents the orientation and the 3 by 1 right hand column matrix represents the position. The 1 by 4 row matrix at the base can be used for perspective drawing but is not needed for our purposes. The homogeneous matrix or transform can be interpreted as describing the position of a second coordinate frame, where the first three columns describe the direction of the X,Y,Z unit vectors of the second coordinate frame respectively and the end fourth column describes the position of the origin of the second coordinate frame.

With this interpretation of the homogeneous matrix in mind, as describing a second coordinate frame, in the methodology for obtaining the forward kinematic equations a coordinate frame is assigned to the end of each link of the manipulator. For an n degree of freedom manipulator there will be n links and n joints. The joints and links are numbered outwardly from the base, the base of the manipulator is link zero and is not considered as one of the links. Link 1 is connected to the base link by joint 1. There is no joint at the end of the final link. The relationship between successive frames is established according to the

following scheme of rotations and translations between frame n-1, n:

1.Rotation about Z_{n-1} an angle θ_n ---this is the variable for a revolute joint.

2.Translation along the Z_{n-1} a distance d_n

3.Translate along the rotated X_{n-1} a distance a_n ---this typically link length L_n

4.Rotation about the X_{n-1} the twist angle α_n

The relationship between successive frames may be expressed as the product of four homogeneous transformations relating the coordinate frame of link n-1 to the coordinate frame of link n. This produces what is referred to as the A matrix:
N.B. The noation for these transforms is defined in appendix 1.

$$^{n-1}A_n = Rot(Z,\theta)\ Trans(0,0,d)\ Trans(a,0,0)\ Rot(X,\alpha)$$

$$^{n-1}A_n = \begin{bmatrix} C\theta & -S\theta C\alpha & S\theta S\alpha & a\,C\theta \\ S\theta & C\theta C\alpha & -C\theta S\alpha & a\,S\theta \\ 0 & S\alpha & C\alpha & d \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

In a prismatic joint the distance d_n is the joint variable. The direction of the joint axis is the direction in which the joint moves. In the case of a prismatic joint the length a_n has no meaning and is set to zero. The Z axis of the prismatic link is aligned with the axis of joint n+1. The A matrix for the prismatic joint is shown below:

$$^{n-1}A_n = \begin{bmatrix} C\theta & -S\theta C\alpha & S\theta S\alpha & 0 \\ S\theta & C\theta C\alpha & -C\theta S\alpha & 0 \\ 0 & S\alpha & C\alpha & d \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Having assigned coordinate frames to each joint a parameter table may be drawn up that tabulates the complete kinematic characteristics of the manipulator. The various parameters for each link may then be substituted into the generalised A matrix and the relationship between the position of the end effector in base coordinates may be found from the following:

$${}^0A_n = {}^0A_1\,{}^0A_2\,{}^0A_3 \ldots\ldots\ldots\ldots {}^{n-1}A_n$$

The parameter table will typically consist of the following:

Link	Variable	α	a	d
1	θ_1	α_1	a_1	d_1
'	'	'	'	'
'	'	'	'	'
'	'	'	'	'
n	θ_n	α_n	a_n	d_n

As example a six degree of freedom robot manipulator will be considered that has the common jointed configuration.

2.1.1 Example

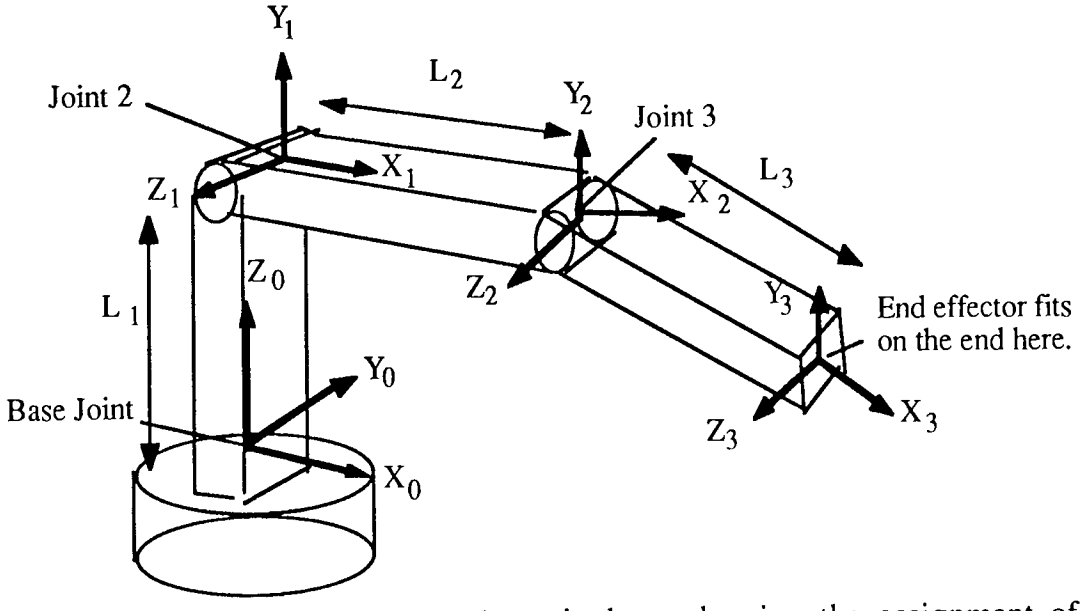


Figure 2.1. Drawing of a jointed manipulator showing the assignment of coordinate frames.

As shown in Fig.2.1 the first four coordinate frames are shown assigned to the respective joints, which provide the manipulator with the ability to move the end effector to the desired location in space. The end effector is not shown in this figure, normally the end effector will provide three degrees of freedom that will

allow it to be orientated in any arbitrary orientation. There is a desirability in terms of the inverse kinematic solution for the design engineer to ensure that the last three joint axes of the manipulator to intersect at a point. A general closed loop inverse kinematic solution for manipulators of this type may be found. The parameter table for the first three links is shown below:

Link	Variable	α	a	d
1	θ_1	+90	0	L_1
2	θ_3	0	L_2	0
3	θ_3	0	L_3	0
'	'	'	'	'
n	θ_n	α_n	a_n	d_n

The values of the first three A matrices may be found and are as follows:

$$A_1 = \begin{bmatrix} C\theta_1 & 0 & S\theta_1 & 0 \\ S\theta_1 & 0 & -C\theta_1 & 0 \\ 0 & 1 & 0 & L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} C\theta_2 & -S\theta_2 & 0 & L_2C\theta_2 \\ S\theta_2 & C\theta_2 & 0 & L_2S\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} C\theta_3 & -S\theta_3 & 0 & L_3C\theta_3 \\ S\theta_3 & C\theta_3 & 0 & L_3S\theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The overall transform is found by multiplying the three A matrices together:

${}^0T_w = {}^0T_3 = A_1 A_2 A_3$ where subscript w refers to the wrist .

The overall transform after simplification can shown to be the following:

$$
{}^{0}T_{w}=\begin{bmatrix} C_1C_{23} & -C_1S_{23} & S_1 & L_3C_1C_{23}+L_2C_1C_2 \\ S_1C_{23} & -S_1C_{23} & -C_1 & L_3S_1C_{23}+L_2S_1C_2 \\ S_{23} & C_{23} & 0 & L_3S_{23}+L_2S_2+L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

The wrist position is found by post multiplying the transform by a zero column vector, which defines the origin of the wrist coordinate frame with respect to this frame. The wrist position in terms of joint angles is as follows:

$$X_w=\text{Cos}\theta_1(L_3\text{Cos}(\theta_2+\theta_3)+L_2\text{Cos}\theta_2)$$

$$Y_w=\text{Sin}\theta_1(L_3\text{Cos}(\theta_2+\theta_3)+L_2\text{Cos}\theta_2)$$

$$Z_w=L_1+L_2\text{Sin}\theta_2+L_3\text{Sin}(\theta_2+\theta_3)$$

2.1.2 The end effector position vector.

If it is assumed that the last three joint axes have been designed to intersect at a point a line diagram of the jointed manipulator can now be drawn depicting this geometry as shown in Fig.2.2:

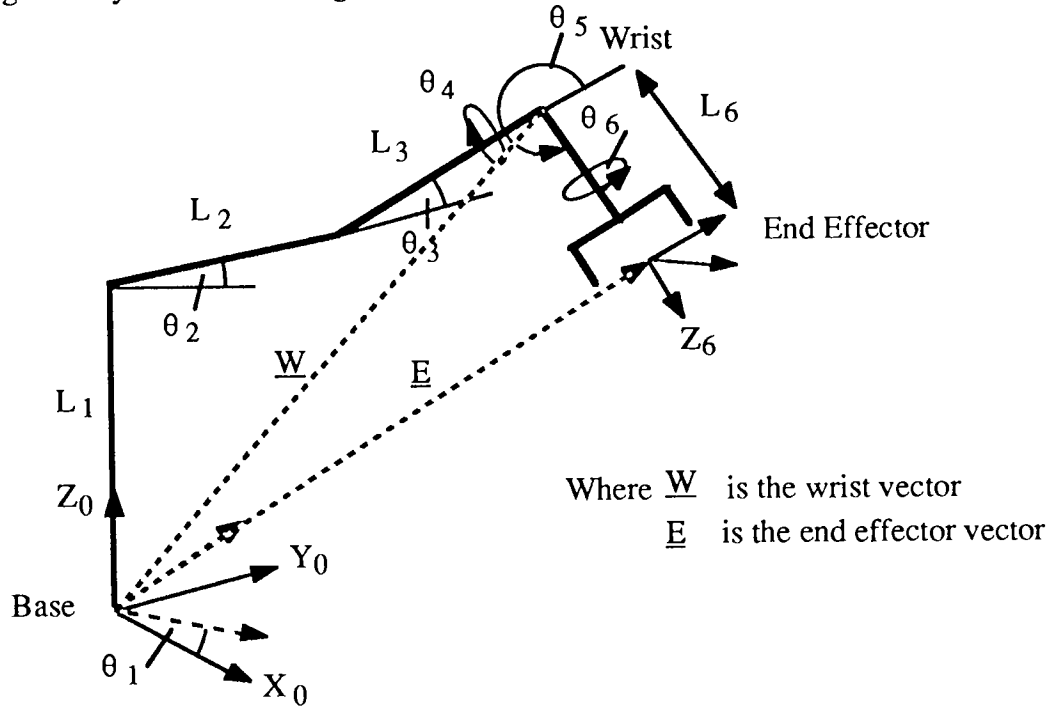


Figure 2.2. Schematic diagram of manipulator showing the joint axes.

As shown in Fig.2.2 the end effector has three degrees of freedom provided by three revolute joints that intersect at the wrist. The final coordinate frame by convention is positioned at the centre of the end effector jaws, with the Z axis unit vector pointing in the direction of approach. The distance from the wrist to this final coordinate frame is L_6. The desired orientation of the end effector would normally be known. The components of X,Y and Z that the end effector makes in base coordinates can be calculated. For instance, if the orientation of the end effector is defined using the Eulerian as follows:
N.B The notation for the Eulerian is defined in appendix 1.

$$R_{\phi,\theta,\psi}=\text{Rot}(Z,\phi)\ \text{Rot}(X,\theta)\ \text{Rot}(Z',\psi)$$

The transform for this is as follows:

$$R_{\phi,\theta,\psi}=\begin{bmatrix} C\phi\,C\psi-S\phi\,C\theta\,S\psi & -C\phi\,S\psi-S\phi\,C\theta\,C\psi & S\phi S\theta \\ S\phi\,C\psi-C\phi\,C\theta\,S\psi & -S\phi\,S\psi+C\phi\,C\theta\,C\psi & -C\phi S\theta \\ S\theta S\psi & S\theta C\psi & C\theta \end{bmatrix}$$

The component of the end effector in base coordinates is found by multiplying this transform by a column vector $[0,0,L_6]$, since the end effector position is defined in the direction of the Z axis of the final coordinate frame. The components of the wrist in base coordinates are as follows:

$$X_{L6}=L_6\sin\phi\sin\theta$$

$$Y_{L6}=-L_6\cos\phi\sin\theta$$

$$Z_{L6}=L_6\cos\theta$$

The final position of the end effector, defined by the vector $\underline{E}$ is thus calculated by adding the wrist vector $\underline{W}$ (as defined previously) to the three components of the end effector:

$X_{base}=X_w+X_{L6}$

$Y_{base}=Y_w+Y_{L6}$

$Z_{base}=Z_w+Z_{L6}$

2.1.3 The Jacobian

In trajectory planning it is necessary to know the relationship between the end effector velocities and the Cartesian velocities. If the position vector is defined as $\underline{P}$ as shown below:

$$\underline{P}=\begin{bmatrix} X \\ Y \\ Z \\ \phi_x \\ \phi_y \\ \phi_z \end{bmatrix}$$

The joint positions are also defined similarly as a vector of joint angles $\underline{P}_\theta$ as follows:

$$\underline{P}_\theta=\begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \\ \theta_6 \end{bmatrix}$$

$\underline{P}$ = function $\underline{P}_\theta$ defines the relationship between the variables in Cartesian space and joint space.

However, if differential displacements of the hand are considered, a vector of differential displacements can be defined as $\underline{D}$ as shown below:

$$\underline{D} = \begin{bmatrix} d_x \\ d_y \\ d_z \\ \delta_x \\ \delta_y \\ \delta_z \end{bmatrix}$$

where d_x,d_y and d_z are differential displacements along the X,Y and Z base axes.

δ_x,δ_y and δ_z are differential rotations about the X,Y and Z base axes.

Similarly, a six element vector of differential rotations of the six joint angles can be defined as $\underline{D}_\theta$ as shown below:

$$\underline{D}_\theta = \begin{bmatrix} d_{\theta 1} \\ d_{\theta 2} \\ d_{\theta 3} \\ d_{\theta 4} \\ d_{\theta 5} \\ d_{\theta 6} \end{bmatrix}$$

The relationship between the two differential vectors is as follows:

$\underline{D} = J\,\underline{D}_\theta$ where J defines the 6 by 6 Jacobian matrix.

If these differential motions occur over a finite time interval then the instantaneous velocities can be defined:

$$\dot{\underline{X}} = \left(\frac{1}{\Delta t}\right)\underline{D}$$

Where $\dot{\underline{X}}$ is the velocity vector expressed in Cartesian coordinates:

$$\dot{\underline{X}} = \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \\ \dot{\phi}_x \\ \dot{\phi}_y \\ \dot{\phi}_z \end{bmatrix}$$

Similarly, a vector of joint velocities can be defined:

$$\dot{\underline{\theta}} = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \\ \dot{\theta}_4 \\ \dot{\theta}_5 \\ \dot{\theta}_6 \end{bmatrix}$$

The relationship between Cartesian velocities and joint velocities can be written as follows:

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \\ \dot{\phi}_x \\ \dot{\phi}_y \\ \dot{\phi}_z \end{bmatrix} = \begin{bmatrix} J_{11} J_{12} J_{13} & J_{14} J_{15} J_{16} \\ J_{21} J_{22} J_{23} & J_{24} J_{25} J_{26} \\ J_{31} J_{32} J_{33} & J_{34} J_{35} J_{36} \\ J_{41} J_{42} J_{43} & J_{44} J_{45} J_{46} \\ J_{51} J_{52} J_{53} & J_{54} J_{55} J_{56} \\ J_{61} J_{62} J_{63} & J_{64} J_{65} J_{66} \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \\ \dot{\theta}_4 \\ \dot{\theta}_5 \\ \dot{\theta}_6 \end{bmatrix}$$

where for example $J_{11} = \frac{\partial x}{\partial \theta_1}$, $J_{21} = \frac{\partial y}{\partial \theta_1}$ etc.

For the special case of three axes intersecting at a point, the velocity of the wrist in terms of the first three joint velocities can be found from the equations defining the wrist position. These equations are partially differentiated with respect to the three joint angles as shown below:

$$\frac{\partial X_w}{\partial \theta_1} = -\sin\theta_1 (L_3 \cos(\theta_2 + \theta_3) + L_2 \cos\theta_2)$$

$$\frac{\partial X_w}{\partial \theta_2} = \cos\theta_1 (-L_3 \sin(\theta_2 + \theta_3) - L_2 \sin\theta_2)$$

$$\frac{\partial X_w}{\partial \theta_3} = \cos\theta_1 (-L_3 \sin(\theta_2 + \theta_3))$$

$$\frac{\partial Y_w}{\partial \theta_1} = \cos\theta_1 (L_3 \cos(\theta_2 + \theta_3) + L_2 \cos\theta_2)$$

$$\frac{\partial Y_w}{\partial \theta_2} = \sin\theta_1 (-L_3 \sin(\theta_2 + \theta_3) - L_2 \sin\theta_2)$$

$$\frac{\partial Y_w}{\partial \theta_3} = \sin\theta_1 (-L_3 \sin(\theta_2 + \theta_3))$$

$$\frac{\partial Z_w}{\partial \theta_1} = 0$$

$$\frac{\partial Z_w}{\partial \theta_2} = L_2 \cos\theta_2 + L_3 \cos(\theta_2 + \theta_3)$$

$$\frac{\partial Z_w}{\partial \theta_3} = L_3 \cos(\theta_2 + \theta_3)$$

Hence, a Jacobian relating wrist velocities to the first three joint velocities can be defined, where the elements of the Jacobian are as shown above.

$\dot{\underline{X}}_w = J_w \dot{\underline{\theta}}_{1\text{-}3}$ Where J_w is the Jacobian for the wrist

The elements of the Jacobian are configuration dependent as seen in the equation above.

The velocity contribution by the last three joints can be found by differentiating the components due to the end effector as shown below:

$$\dot{X}_{L6} = L_6 (\cos\phi \sin\theta\dot{\phi} + \cos\theta \sin\phi \dot{\theta})$$

$$\dot{Y}_{L6} = -L_6 (-\sin\phi \sin\theta\dot{\phi} + \cos\phi \cos\theta \dot{\theta})$$

$$\dot{Z}_{L6} = -L_6 (\sin\theta \dot{\theta})$$

The overall end effector velocities are found by adding the components due to the end effector to the wrist velocities.

The Jacobian can be found by using the homogeneous matrices as used by Paul[3] however, this method will not be considered.

2.1.4 Derivative of Jacobian

The derivative of the Jacobian relates the Cartesian accelerations to the joint accelerations. The Cartesian accelerations become a function of the joint displacements, velocities and accelerations.

$$\ddot{\underline{X}} = \text{function}(\underline{\theta}, \dot{\underline{\theta}}, \ddot{\underline{\theta}})$$

e.g. term $J_{11w}\dot{\theta}_1$ which contributes to the $\dot{X}$ velocity when differentiated becomes:

$$\begin{aligned}\frac{d(J_{11w}\dot{\theta}_1)}{dt} = &-\sin\theta_1 (L_3 \cos(\theta_2 + \theta_3) + L_2 \cos\theta_2) \ddot{\theta}_1 \\ &- \cos\theta_1 (L_3 \cos(\theta_2 + \theta_3) + L_2 \cos\theta_2) \dot{\theta}_1^2 \\ &+ \sin\theta_1 (L_3 \sin(\theta_2 + \theta_3) + L_2 \sin\theta_2) \dot{\theta}_1\dot{\theta}_2 \\ &+ \sin\theta_1 (L_3 \sin(\theta_2 + \theta_3)) \dot{\theta}_1\dot{\theta}_3\end{aligned}$$

The derivative of the Jacobian becomes a very complicated expression, as seen above for a single term.

2.2 Inverse Kinematics

The problem of finding the inverse kinematic equations is far harder than finding the forward kinematics equations. This is because the forward kinematic equations are non-linear and there is no general algorithm which may be used to solve the equations. The methods that exist can be broadly classified into two categories:

1) Closed form solutions

2) Numerical solutions

Numerical solutions are much slower than closed form solutions because of their iterative nature and they do not lend themselves readily to robot control systems. Closed form solutions can be divided into two different types:

1) Algebraic

2) Geometric

The distinction between the two methods is somewhat hazy due to the fact that the geometric methods are applied by algebraic expressions, the real difference lies in the approach.

It is difficult to find closed form solutions for general geometry manipulators and normally a solution is found that will be valid for that manipulator only. However, closed form solutions exist for manipulators of special geometry. As mentioned previously, manufacturers have realized this fact and design the manipulators with this type of geometry i.e. one type is a six degree of freedom manipulator with the end three joint axes intersecting at a point. With this type of geometry the problem of solving for the inverse kinematics can be divided into two sub-problems. The position of the wrist can be found given the required orientation and this wrist positon can be solved in terms of the first three joint angles. The remaining joint angles may then be solved given the required orientation and the solution of the first three joint angles.

2.2.1 Algebraic Approach

The algebraic approach by Paul[3] involves the use of the A matrices, this will be described. For a six degree of freedom manipulator the orientation and desired position of the manipulator would be known and can be specified as T_6. This is equal to the product of the six A matrices:

$$T_6=A_1(\theta_1)\, A_2(\theta_2)\, A_3(\theta_3)\, A_4(\theta_4)\, A_5(\theta_5)\, A_6(\theta_6)$$

The dependency of each A matrix on the joint variable is shown by the bracketed joint angle. Six matrix equations can be obtained by successively premultiplying the above equations by the A matrix inverses:

$$A_1(\theta_1)^{-1}\, {}^0T_6={}^1T_6$$

$$A_2(\theta_2)^{-1}\, A_1(\theta_1)^{-1}\, {}^0T_6={}^2T_6$$

$$A_3(\theta_3)^{-1}\, A_2(\theta_2)^{-1}\, A_1(\theta_1)^{-1}\, {}^0T_6={}^3T_6$$

$$A_4(\theta_4)^{-1}\, A_3(\theta_3)^{-1}\, A_2(\theta_2)^{-1}\, A_1(\theta_1)^{-1}\, {}^0T_6={}^4T_6$$

$$A_5(\theta_5)^{-1}\, A_4(\theta_4)^{-1}\, A_3(\theta_3)^{-1}\, A_2(\theta_2)^{-1}\, A_1(\theta_1)^{-1}\, {}^0T_6={}^5T_6$$

This technique of multiplying each side of the transform by an inverse is a way of separating out the variables into a solvable equation. The matrix elements on the right hand side of the equations are either zero, constants or functions of n to 6 joint variables(${}^nT_6$). On the left hand side of the equations are the elements of desired orientation and position specification transform together with functions of the ascending joint variables 1 to n. The object is to equate elements of each side of the matrix equations and solve for the joint variables in ascending order.

If the solution for the jointed manipulator is considered, a transform for the wrist position was obtained and can be equated to the position and orientation matrix :

$${}^0T_w=\begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

This can be equated to the transform previously obtained. However, the orientation values cannot be equated since the position of the wrist is only being considered. The position elements (1,4),(2,4) and (3,4) can be equated:

N.B The notation for the following equations is defined in appendix 1.

$$P_x = C_1(L_3C_{23}+L_2C_2) \qquad (2.1)$$

$$P_y = S_1(L_3C_{23}+L_2C_2) \qquad (2.2)$$

$$P_z = L_3S_{23}+L_2S_2+L_1 \qquad (2.3)$$

Dividing equation (2.2) by (2.1) angle θ_1 may be found:

$$\theta_1 = \tan^{-1}\left(\frac{p_y}{p_x}\right) \qquad (2.4)$$

The next step is to obtain $A_1^{-1}\,{}^0T_w = A_2A_3$, first A_1^{-1} is obtained :

where
$$A_1^{-1} = \begin{bmatrix} C_1 & S_1 & 0 & 0 \\ 0 & 0 & 1 & -L_1 \\ S_1 & -C_1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

and
$$A_2A_3 = \begin{bmatrix} C_{23} & -S_{23} & 0 & L_3C_{23}+L_2C_2 \\ S_{23} & C_{23} & 0 & L_3S_{23}+L_2S_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The terms in the position vectors are only considered, the right hand side gives:

$C_1 p_x + S_1 p_y$ element 1,4
$p_z - L_1$ element 2,4
$S_1 p_x - C_1 p_y$ element 3,4

Equating these elements with the right hand side (i.e. A_2A_3) gives the following equations:

$$C_1p_x+S_1p_y= L_3C_{23}+L_2C_2 \qquad (2.5)$$

$$p_z-L_1= L_3S_{23}+L_2S_2 \qquad (2.6)$$

The next step is to find $A_2^{-1}\, A_1^{-1}\, {}^0T_w=A_3$

where $$A_2^{-1} = \begin{bmatrix} C_2 & S_2 & 0 & -L_2 \\ -S_2 & C_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

This gives the following elements involving the position vectors:

$C_2 (C_1 p_x + S_1 p_y) + S_2 (p_z - L_1) - L_2$ element 1,4

$-S_2 (C_1 p_x + S_1 p_y) + C_2 (p_z - L_1)$ element 2,4

$S_1 p_x - C_1 p_y$ element 3,4

Equating to A_3 gives the two useful equations:

$$C_2(C_1p_x+S_1p_y) + S_2(p_z-L_1) - L_2= L_3C_3 \qquad (2.7)$$

$$-S_2(C_1p_x+S_1p_y) + C_2(p_z-L_1) =L_3S_3 \qquad (2.8)$$

From (2.5) and (2.6) let

$$R_x= L_3C_{23}+L_2C_2 \qquad (2.9)$$

$$R_y= L_3S_{23}+L_2S_2 \qquad (2.10)$$

Substituting for R_x and R_y in equations (2.7) and (2.8):

$$C_2R_x+S_2R_y-L_2=L_3C_3 \qquad (2.11)$$

$$-S_2R_x+C_2R_y=L_3S_3 \qquad (2.12)$$

Dividing (2.11) into (2.12) gives angle θ_3 however, angle θ_2 needs to be determined before this can be evaluated.

$$\theta_3 = \tan^{-1}\left(\frac{-S_2 R_x + C_2 R_y}{C_2 R_x + S_2 R_y - L_2}\right) \qquad (2.12a)$$

To find θ_2 using equations (2.9) and (2.10) and writing them as follows:

$$R_x - L_2 C_2 = L_3 C_{23} \qquad (2.13)$$

$$R_y - L_2 S_2 = L_3 S_{23} \qquad (2.14)$$

Squaring both sides of these equations and adding yields:

$$R_x^2 + R_y^2 + L_2^2 - L_3^2 = 2 L_2 (R_x C_2 + R_y S_2)$$

$$\text{letting } K = \frac{R_x^2 + R_y^2 + L_2^2 - L_3^2}{2 L_2}$$

$$K = R_x C_2 + R_y S_2 \qquad (2.15)$$

Equation (2.15) is of the form shown in Paul[3] and may be solved by letting:

$$R_x = r\cos\phi \quad \text{and} \quad R_y = r\sin\phi$$

$$\text{where } r = \sqrt{R_x^2 + R_y^2} \quad \text{and} \quad \phi = \tan^{-1}\left(\frac{R_y}{R_x}\right) \qquad (2.16)$$

Substituting for R_x and R_y in (2.15) gives:

$$K = r\cos\phi\, C_2 + r\sin\phi\, S_2 = r\cos(\phi - \theta_2) \qquad (2.16b)$$

Using the identity $\sin^2\theta + \cos^2\theta = 1$ to find $\sin(\phi - \theta_2)$:

$$\sin(\phi - \theta_2) = \frac{\sqrt{r^2 - K^2}}{r} \qquad (2.17)$$

Dividing (2.16b) into (2.17) gives the following:

$$\tan(\phi-\theta_2) = \frac{\sqrt{r^2 - K^2}}{K} \qquad (2.18)$$

Hence, using (2.16) and (2.18) θ_2 may be found:

$$\theta_2 = \tan^{-1}(\frac{R_y}{R_x}) - \tan^{-1}(\frac{\sqrt{r^2 - K^2}}{K}) \qquad (2.19)$$

Hence, all three angles may be determined if equations (2.4),(2.19) and (2.12a) are used in that sequence.

2.2.2 Geometric Approach

The problem with the algebraic approach is that it does not give any indication of how to choose from the many possible solutions that are obtained. With a geometric approach it is possble to gain a better understanding of the solutions obtained. This approach will be considered when solving for the RTX[4] manipulator later on in the chapter.

2.2.3 Inverse Jacobian

The forward kinematic equations relating Cartesian velocities to joint velocities was defined previously as:

$$\dot{\underline{X}} = J\,\dot{\underline{\theta}}$$

The inverse relationship requires the inverse J^{-1} to be obtained:

$$\dot{\underline{\theta}} = J^{-1}\,\dot{\underline{X}}$$

There are a number of methods that could be used to obtain the inverse Jacobian these include:

1. Symbolic inversion of matrix.
2. Numerical inverse at an instant in time(e.g. Gaussian elimination).

The symbolic inversion can be extremely complicated for a six degree of freedom manipulator. It will be seen that this method is used to obtain the inverse Jacobian for the RTX manipulator. Using numerical inversion at each step point, the new values of the Jacobian are obtained and the numerical Jacobian is then inverted. Using, say Gaussian elimination, the next point can be calculated at this point. The procedure is repeated step by step to obtain the desired joint angle rates.

2.2.4 Inverse Derivative Jacobian

The forward kinematic equation was defined previously as:

$\ddot{\underline{X}}$ = function($\underline{\theta}$,$\dot{\underline{\theta}}$,$\ddot{\underline{\theta}}$)

The inverse is defined as follows:

$\ddot{\underline{\theta}}$ = function($\underline{\theta}$,$\dot{\underline{\theta}}$,$\ddot{\underline{X}}$)

The inverse becomes extremely complex for a six degree of freedom system, the inverse is obtained for two degrees of freedom of the RTX manipulator.

2.3 Particular kinematic solution for RTX manipulator.

2.3.1 Forward Kinematics

The kinematic representation of the RTX shown with coordinated frames assigned is shown in Fig.2.3 below:

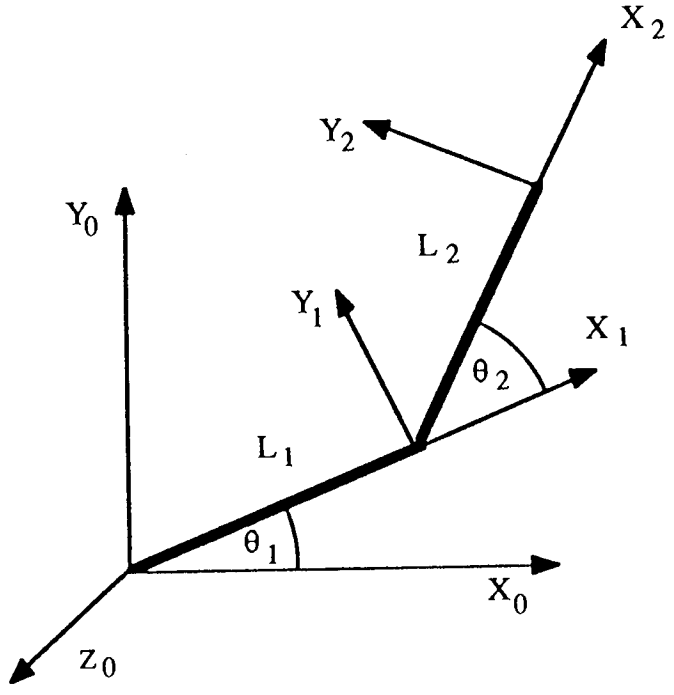


Figure 2.3. Drawing showing the kinematic representation of the RTX manipulator's two arm links.

The parameter table for this manipulator is shown below:

Link	Variable	α	a	d
1	θ_1	0	L_1	0
2	θ_2	0	L_2	0

Table 2.1. Parameter table for two links of the RTX manipulator.

The A matrices for this manipulator are as follows:

$$A_1 = \begin{bmatrix} C_1 & -S_1 & 0 & L_1 C_1 \\ S_1 & C_1 & 0 & L_1 S_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad A_2 = \begin{bmatrix} C_2 & -S_2 & 0 & L_2 C_2 \\ S_2 & C_2 & 0 & L_2 S_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^0A_2 = A_1 A_2 = \begin{bmatrix} C_{12} & -S_{12} & 0 & L_1 C_1 + L_2 C_{12} \\ S_{12} & C_{12} & 0 & L_1 S_1 + L_2 S_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The forward kinematic equations are as follows:

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} L_1 \mathrm{Cos}\theta_1 + L_2 \mathrm{Cos}(\theta_1 + \theta_2) \\ L_1 \mathrm{Sin}\theta_1 + L_2 \sin(\theta_1 + \theta_2) \end{bmatrix} \qquad (2.20)$$

The relationship between joint velocities can be found by differentiating (2.20):

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \end{bmatrix} = \begin{bmatrix} (-L_1 \mathrm{Sin}\theta_1 - L_2 \mathrm{Sin}(\theta_1 + \theta_2)) & (-L_2 \mathrm{Sin}(\theta_1 + \theta_2)) \\ (L_1 \mathrm{Cos}\theta_1 + L_2 \mathrm{Cos}(\theta_1 + \theta_2)) & (L_2 \mathrm{Cos}(\theta_1 + \theta_2)) \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} \qquad (2.21)$$

The acceleration equations are most conveniently expressed in terms of the joint accelerations and the joint velocities.

$$\begin{bmatrix} \ddot{X} \\ \ddot{Y} \end{bmatrix} = \begin{bmatrix} -L_1 \sin\theta_1 & -L_2 \sin(\theta_1 + \theta_1) \\ L_1 \cos\theta_1 & L_2 \cos(\theta_1 + \theta_1) \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix}$$

$$- \begin{bmatrix} L_1 \cos\theta_1 & L_2 \cos(\theta_1 + \theta_1) \\ L_1 \sin\theta_1 & L_2 \sin(\theta_1 + \theta_1) \end{bmatrix} \begin{bmatrix} \dot{\theta}_1^2 \\ (\dot{\theta}_1 + \dot{\theta}_2)^2 \end{bmatrix} \qquad (2.22)$$

2.3.2 Inverse Kinematics

The inverse kinematic solution , using a geometric approach(see Fig.2.4), begins with the application of the cosine rule to find the joint angle θ_2.

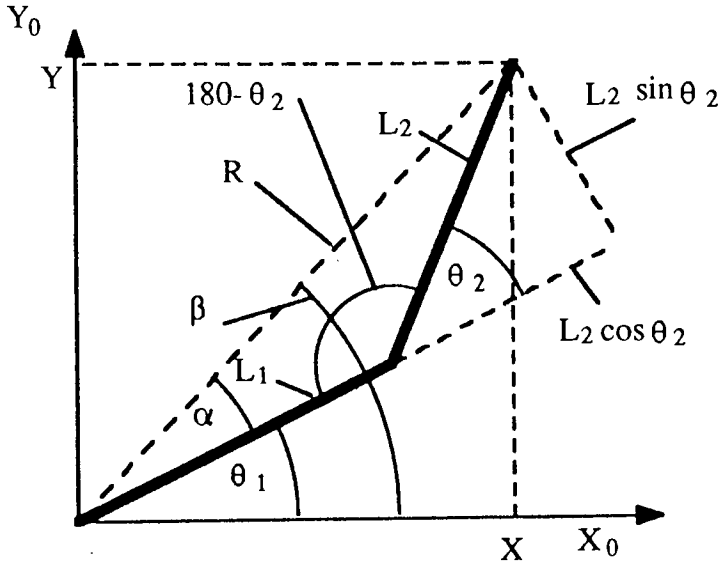


Figure 2.4. Diagram showing the geometric solution for the RTX manipulator.

$R^2=L_1^2+L_2^2 -2\, L_1\, L_2 \cos(180-\theta_2)$ where $\cos(180-\theta_2) =-\cos(\theta_2)$

and $R= \sqrt{X^2 + Y^2}$

Therefore joint angle 2 can be found to be equal to the following:

$$\theta_2 = \cos^{-1}\left(\frac{R^2 - L_1^2 - L_2^2}{2\, L_1\, L_2}\right) \qquad -(23)$$

The angle β is found from the coordinates of the end point:

$$\beta = \tan^{-1}\left(\frac{Y}{X}\right) \qquad (2.24)$$

Angle α is found to be equal to the following:

$$\alpha = \tan^{-1}\left(\frac{L_2 \sin\theta_2}{L_1 + L_2 \cos\theta_2}\right) \qquad (2.25)$$

Therefore joint angle 1 is found by subtracting (2.25) from (2.24):

$$\theta_1 = \tan^{-1}\left(\frac{Y}{X}\right) - \tan^{-1}\left(\frac{L_2 \sin\theta_2}{L_1 + L_2 \cos\theta_2}\right) \qquad (2.26)$$

2.3.3 Work Space

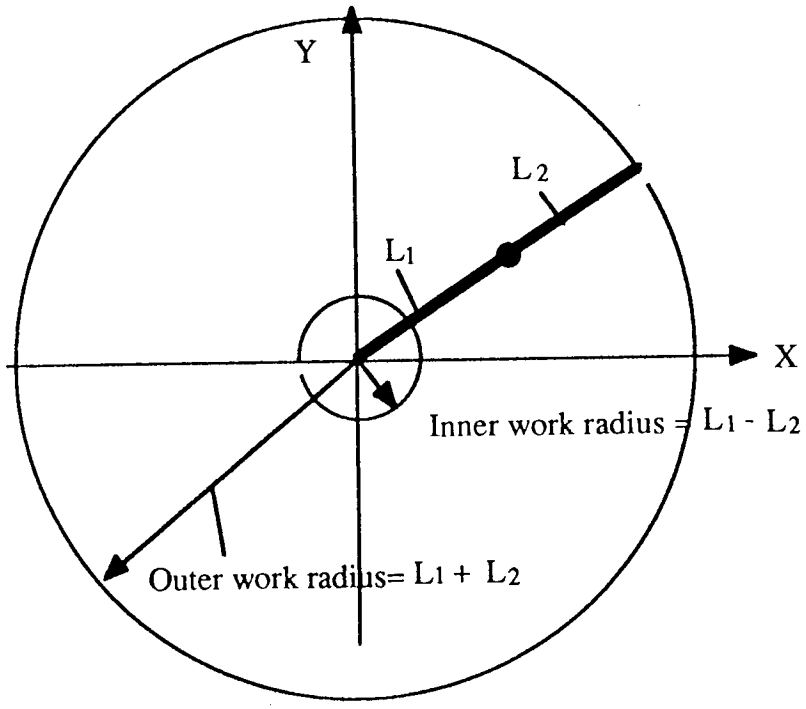


Figure 2.5. Diagram showing the work space for the RTX manipulator.

The boundaries of the workspace of the manipulator lie within an outer circle of radius $R_0=L_1+L_2$ and an inner circle of radius $R_i=L_1-L_2$. Whether it can achieve all of the positions within the workspace will depend upon the constraints of the joints angles. Program TRAJ PLAN (Trajectory planning program described in Chapter 5) contains a inverse kinematics procedure INVERSE for solving for the two joint angles, for a given set of input X and Y coordinates for the RTX manipulator. To check whether a set of X,Y coordinates are within the workspace, the equation for the workspace is the equation of a circle of radius R, if the manipulator only moves in a horizontal plane:

$$R^2=X^2+Y^2 \qquad (2.27)$$

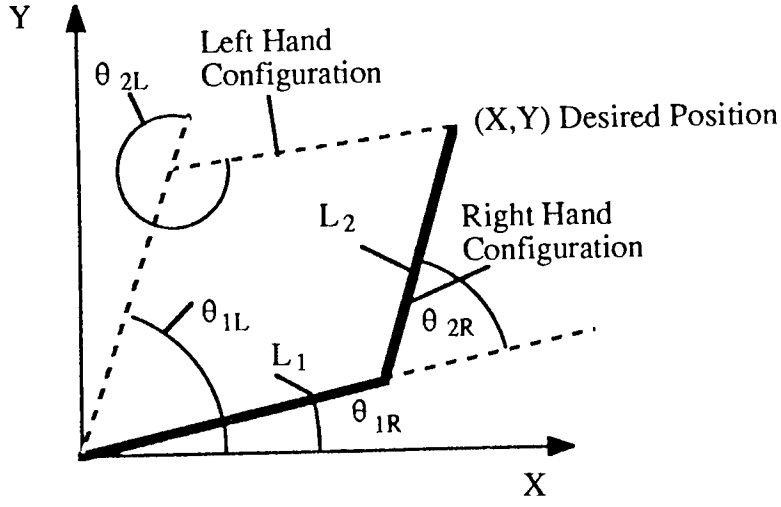
or $$R=\sqrt{X^2+Y^2}$$

i.e. $$L_1-L_2 < R < L_1+L_2 \qquad (2.28)$$

Equation (28) must be satisfied in order that the inverse kinematics algorithm can solve for the given set of X and Y coordinates.If this not satisfied, the input coordinates are rejected by the Trajectory planning program, since they are outside of the work space of the manipulator.

2.3.4 Number of possible solutions or configurations.

There are two solutions to each of the joint angles, this corresponds to two possible configurations of the manipulator as shown in Fig.2.6 below:



Where
θ_{1R} =Right Hand Solution to Joint 1
θ_{1L} =Left Hand Solution to Joint 1
θ_{2R} =Right Hand Solution to Joint 1
θ_{2L} =Left Hand Solution to Joint 1

Figure 2.6. Diagram showing the possible configurations of the manupulator.

The two possible configurations are referred to as Right Hand and Left Hand configurations. In equation (23) the argument of $\cos^{-1}$ can be either positive or

negative, i.e. in quadrants 1and 4 or 2 and 3 respectively as shown in Fig.2.7 below:

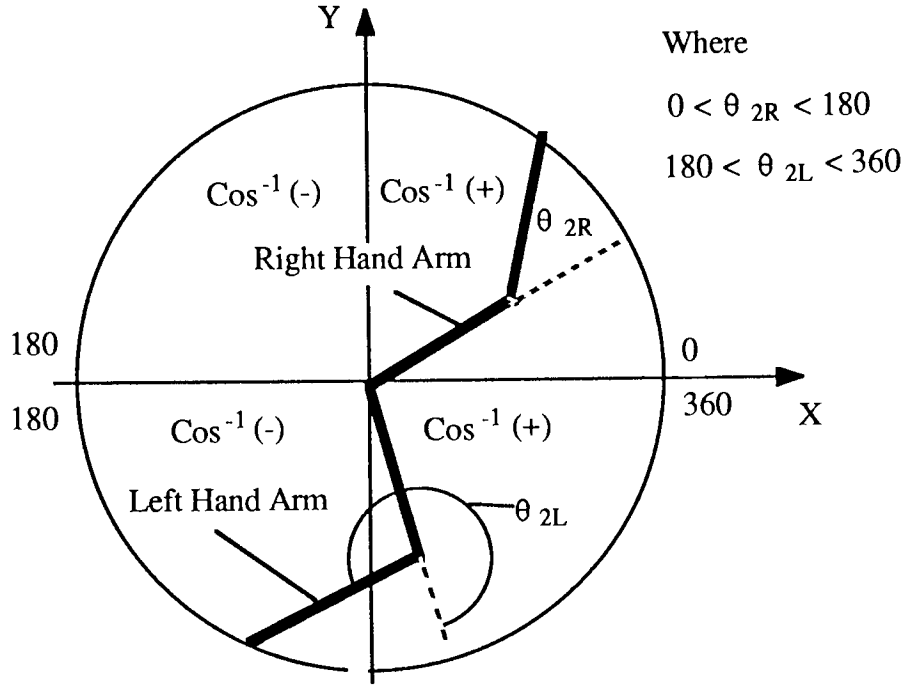


Figure 2.7. Diagram showing the left hand and right hand solutions possible.

In solving for Right Hand Arm	$0 < \theta_2 < 90$	for $\cos^{-1}(+)$
	$90 < \theta_2 < 180$	for $\cos^{-1}(-)$
In solving for Left Hand Arm	$180 < \theta_2 < 270$	for $\cos^{-1}(-)$
	$270 < \theta_2 < 360$	for $\cos^{-1}(+)$

The procedure INVERSE solves for the Left Hand Arm configuration.

When the $R^2=L_1^2+L_2^2$ the numerator of the argument of equation (23) becomes zero. This corresponds to an ambiguity if it is not treated specially. The $\cos^{-1}$(zero) occurs at 90^0 of 270^0 however, since the procedure INVERSE solves for the Left Hand Arm configuration where $180 < \theta_2 < 360$, then when this condition occurs θ_2 is set equal to 270^0.

The ambiguity of solving for α in equation (2.25) and β in equation (2.24) is overcome by using an implementation of an arc tangent function, atan2(y,x). This

is defined as:

$$\theta = \text{atan2}(y,x) = \begin{cases} 0^\circ < \theta < 90^\circ & \text{for } +x \text{ and } +y \\ 90^\circ < \theta < 180^\circ & \text{for } -x \text{ and } +y \\ -180^\circ < \theta < -90^\circ & \text{for } -x \text{ and } -y \\ -90^\circ < \theta < 0^\circ & \text{for } +x \text{ and } -y \end{cases}$$

When the manipulator is required to track along one of the principle axes the angle β cannot be found from equation (2.24) using the atan2(y,x). There are four possible solutions for angle β depending on the values of X and Y:

$$\beta = \begin{cases} \text{If } X=0 \text{ and } Y>0 \text{ then } \beta=90^0 \\ \text{If } X=0 \text{ and } Y<0 \text{ then } \beta=270^0 \\ \text{If } Y=0 \text{ and } X>0 \text{ then } \beta=0^0 \\ \text{If } Y=0 \text{ and } X<0 \text{ then } \beta=180^0 \end{cases}$$

2.3.5 Inverse Jacobian

For this manipulator the inverse Jacobian is found symbolically. The inverse is defined as:

$$J^{-1} = \frac{adj(J)}{|J|}$$ where adj(J)= Transpose of the matrix of the cofactors

$|J|$ = Determinant of the matrix J

Hence, after some manipulation the inverse Jacobian matrix equation can be shown to be :

$$\begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} = \frac{1}{L_1 L_2 \sin\theta_2} \begin{bmatrix} L_2\cos(\theta_1+\theta_2) & L_2\sin(\theta_1+\theta_2) \\ -L_1\cos\theta_1 - L_2\cos(\theta_1+\theta_2) & -L_1\sin\theta_1 - L_2\sin(\theta_1+\theta_2) \end{bmatrix} \begin{bmatrix} \dot{X} \\ \dot{Y} \end{bmatrix} \quad (2.29)$$

This matrix equation is implemented as procedure INVERSE JACOBIAN in the trajectory planning program.

2.3.6 Inverse Kinematic solution for Acceleration

The forward kinematic equation for accelerations of equation (2.22) is of the form:

$$\begin{bmatrix} \ddot{X} \\ \ddot{Y} \end{bmatrix} = M_1 \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_1 + \ddot{\theta}_2 \end{bmatrix} - M_2 \begin{bmatrix} \dot{\theta}_1^2 \\ (\dot{\theta}_1 + \dot{\theta}_2)^2 \end{bmatrix} \qquad (2.30)$$

Multiplying the equation throughout by M_1^{-1} and taking terms in joint accelerations to the left:

$$\begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_1 + \ddot{\theta}_2 \end{bmatrix} = M_1^{-1} \begin{bmatrix} \ddot{X} \\ \ddot{Y} \end{bmatrix} + M_1^{-1} M_2 \begin{bmatrix} \dot{\theta}_1^2 \\ (\dot{\theta}_1 + \dot{\theta}_2)^2 \end{bmatrix} \qquad (2.31)$$

The full equations are thus shown below:

$$\begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_1 + \ddot{\theta}_2 \end{bmatrix} = \frac{1}{L_1 L_2 \sin\theta_2} \begin{bmatrix} L_2\cos(\theta_1+\theta_2) & L_2\sin(\theta_1+\theta_2) \\ -L_1\cos\theta_1 & -L_1\sin\theta_1 \end{bmatrix} \begin{bmatrix} \ddot{X} \\ \ddot{Y} \end{bmatrix}$$

$$+ \frac{1}{L_1 L_2 \sin\theta_2} \begin{bmatrix} L_1 L_2\cos\theta_2 & L_2^2 \\ -L_1^2 & -L_1 L_2\cos\theta_2 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1^2 \\ (\dot{\theta}_1 + \dot{\theta}_2)^2 \end{bmatrix} \qquad (2.32)$$

This matrix equation as procedure INVERSE DERIV JACOBIAN in the Trajectory planning program.

Chapter 3:Dynamics

Chapter 3 - Robot Dynamics

In order to study the dynamic behaviour of robot manipulators it is necessary to formulate the dynamic equations that govern their motion. In chapter 2 it was seen that there are forward and inverse kinematics, in a similar manner there is the forward dynamics where the dynamic equations are formulated for the purpose of calculating the motion of the manipulator as a result of the input torque to each axis. This formulation is particularly suitable for the simulation of the robot manipulator. The inverse dynamic approach is to use the dynamic equations in conjunction with the trajectory data to calculate the required nominal torques to achieve the given trajectory, this approach is particularly useful in the control system design in order to compensate for the non-linear dynamics of the manipulator(as seen in Aada[5] and Craig[6]). In this chapter two methods of formulating the dynamic equations are considered, namely the Newton-Euler and the Lagrangian. The application of the Lagrangian to the general six degree of freedom manipulator of Fig.2.2 in chapter 2 is considered. The dynamic equations for the RTX robot are formulated using the two methods considered and the meaning of the terms discussed.

3.1 Newton -Euler Formulation

The Newton-Euler formulation involves the application of Newton's second law of motion i.e. 'The rate of change of momentum is proportional to the applied force'. The equations that are obtained include the forces and moments on the links of the manipulator. However, the equations are required in a form relating the joint torques to the joint motions. To obtain the equations in this form requires a great deal of algebraic manipulation in order to eliminate the forces.

Consider the general planar link manipulator with n number of links as shown in Fig.3.1. As seen in Fig.3.1 link n is driven by actuator n which is fixed to the end of link n-1. The angular position of link n is defined relative to link n-1 as θ_n. At the end of link n is the actuator driving link n+1, the position of link n+1 defined relative to link n as θ_{n+1}.

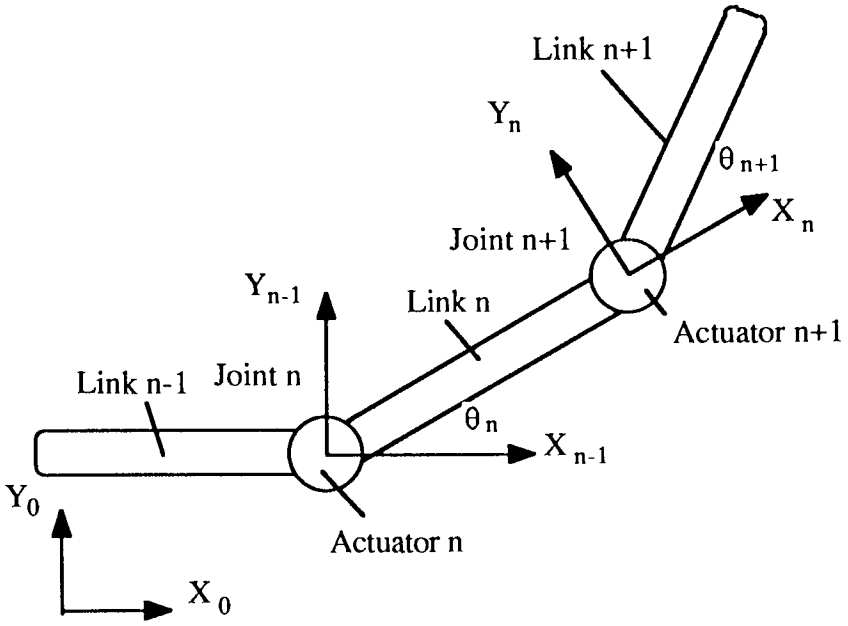


Figure 3.1. Diagram showing a general planar link robot manipulator.

If the link n is dislocated from between the two adjacent links and a free body diagram of the link n is draw showing the forces and moments acting on link n due to the adjacent links of n-1 and n+1 as seen in Fig.3.2:

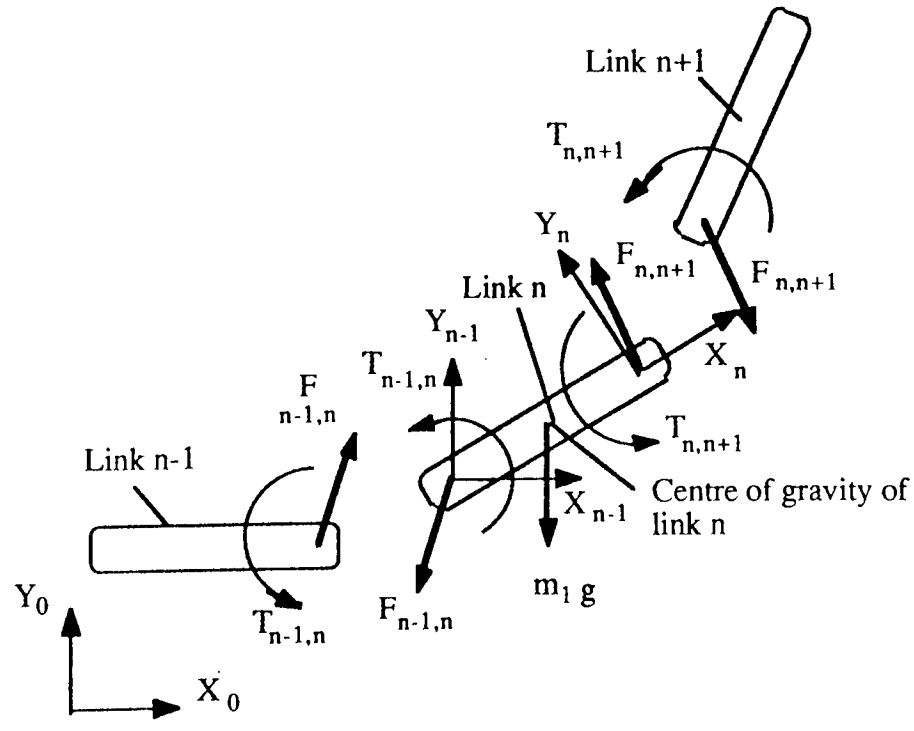


Fig.3.2. Showing the forces and torques acting between adjacent links.

N.B The notation for this is defined in appendix 1.

The linear force vector $F_{n-1,n}$ acts on link n and is exerted by link n-1. The linear force vector $F_{n,n+1}$ represents the force applied to link n+1 by link n. The other force acting on link n is that due to gravity acting in the negative Y direction at the link's centroid. The forces can then be equated to the inertial force, acting at the centroid, given by $m_n a_n$ where a_n is the vectorial acceleration of the link n. The equation if motion is:

$$F_{n,n+1} - F_{n-1,n} - m_n g - m_n a_n = 0 \qquad (3.1)$$

The next step is to consider the balance of moments about the centroid of link n. The moments acting about the centroid can now be equated to the rate of change of angular momentum for link n :

$$T_{n-1,n} - T_{n.n+1} - r_{n-1,n} * F_{n-1,n} + r_{n,n+1} * F_{n,n+1} - I_n \dot{\omega}_n - \omega_n * (I_n . \omega_n) = 0 \qquad (3.2)$$

where * indicates the cross product

$r_{n-1,n}$ is the vector distance of force $F_{n-1,n}$ from the centroid

$r_{n,n+1}$ is the vector distance of force $F_{n,n+1}$ from the centroid

Equations (3.1) and (3.2) are obtained for all links 1 to n. The equations obtained explicitly in the appropriate form in terms of input torques and output position variables are referred to as closed form dynamic equations. The closed form equations for an n degree of freedom manipulator are of the form:

$$T_i = \sum_{j=1}^{n} H_{ij} \ddot{q}_j + \sum_{j=1}^{n} \sum_{k=1}^{n} h_{ijk} \dot{q}_j \dot{q}_k + G_i \qquad i=1,.....,n \qquad (3.3)$$

where H_{ij},h_{ijk} and G_i are functions of joint displacements q_i,.....,q_n.

3.2 Lagrangian

The Lagrangian L is defined as the difference between the total kinetic and the total potential energy of the system.

$$L = K - P \qquad (3.4)$$

where K= total kinetic energy
P= total potential energy

The kinetic and potential energy of the system may be expressed in any convenient coordinate system. The dynamic equations, in terms of the coordinates used to express the kinetic and potential energy, are obtained from the following:

$$F_i = \frac{d}{dt}\frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} \qquad (3.5)$$

where

q_i the coordinate in which the kinetic and potential energy is expressed
$\dot{q}_i$ is the corresponding velocity
F_i is the corresponding force or torque

As an example consider the jointed manipulator in chapter 2 Fig.2.2. The first three links will be considered i.e. the torques at the first three joints due to the motion of the first three links. The representation of this jointed manipulator is shown in Fig.3.3. In order to simplify the sytem links 2 and 3 are considered to be represented by concentrated masses at the end. The position of the centre of gravity of these links thus corresponds to the position of the assigned coordinate frames. This makes the mathematics some what easier to handle for this example.

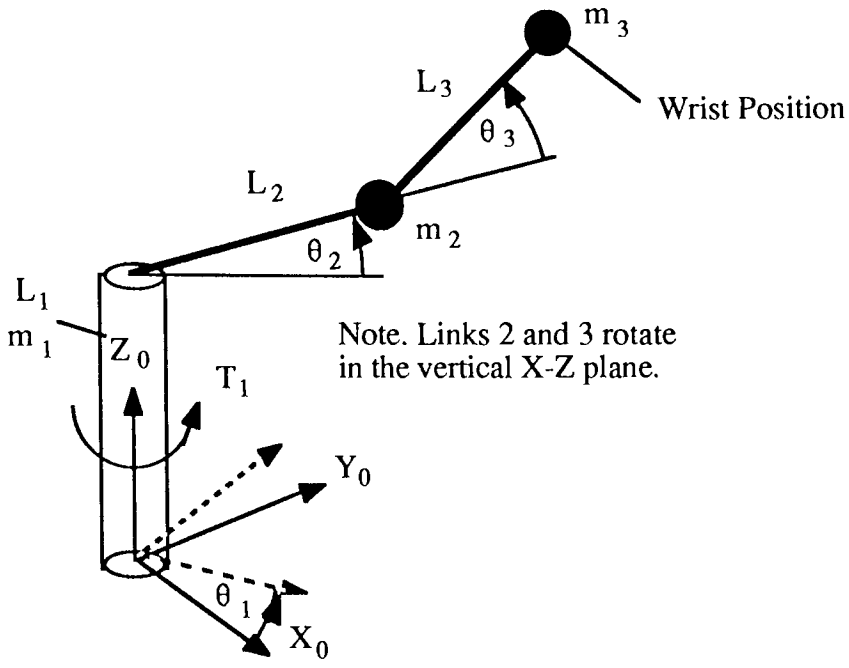


Figure 3.3. Diagram showing a jointed three degree of freedom robot.

As seen in Fig.3.3 link 1 is represented as a cylindrical bar of uniform cross section with a polar inertia about axis Z_0 of J_1. Links 1 and 2 are represented by concentrated masses at the end of their respective links.

The kinetic energy of link 1 is that due to its rotation motion about the Z_0 axis and is equal to:

$$Ke_1 = \frac{1}{2} J_1 \dot{\theta}_1^2 \qquad (3.6)$$

The kinetic energy of link 2 is found from the equation :

$$Ke_2 = \frac{1}{2} m_2 V_2^2 \qquad (3.7)$$

where V_2 is the resultant vector of velocity of m_2

The position of mass m_2 can be found from the A matrices of Fig.2.2 in chapter 2.

The position of the end of link 2 P_{21} is thus :

$$P_2 = A_1 A_2 [0,0,0,1]^T \qquad (3.8)$$

Evaluating the above gives the following:

$$\begin{aligned} X_2 &= L_2 \cos\theta_1 \cos\theta_2 \\ Y_2 &= L_2 \sin\theta_1 \cos\theta_2 \qquad (3.9) \\ Z_2 &= L_1 + L_2 \sin\theta_2 \end{aligned}$$

Differentiating equations (3.9) with respect to time gives the component velocities of m_2:

$$\begin{aligned} \dot{X}_2 &= L_2 (-\sin\theta_1 \cos\theta_2 \dot{\theta}_1 - \cos\theta_1 \sin\theta_2 \dot{\theta}_2) \\ \dot{Y}_2 &= L_2 (\cos\theta_1 \cos\theta_2 \dot{\theta}_1 - \sin\theta_1 \sin\theta_2 \dot{\theta}_2) \qquad (3.10) \\ \dot{Z}_2 &= L_2 \cos\theta_2 \dot{\theta}_2 \end{aligned}$$

Hence the resultant velocity V_2 is :

$$V_2^2 = \dot{X}_2^2 + \dot{Y}_2^2 + \dot{Z}_2^2$$

The position of the end link mass was determined in chapter 2 and the velocity components were also obtained for the Jacobian such that V_3^2 is :

$$\begin{aligned} V_3^2 = &(J_{11} \dot{\theta}_1 + J_{12} \dot{\theta}_2 + J_{13} \dot{\theta}_3)^2 + (J_{21} \dot{\theta}_1 + J_{22} \dot{\theta}_2 + J_{23} \dot{\theta}_3)^2 \\ &+ (J_{31} \dot{\theta}_1 + J_{32} \dot{\theta}_2 + J_{33} \dot{\theta}_3)^2 \end{aligned}$$

The potential energy P_E of the manipulator are determined from the Z component of the position of the centre of mass. Hence, for links 2 and 3 these are:

$$Pe_2 = m_2 g (L_1 + L_2 \sin\theta_2)$$

$$Pe_3 = m_3 g \left(L_1 + L_2 \sin\theta_2 + L_3 \sin(\theta_2 + \theta_3) \right)$$

Having obtained the total kinetic and potential energy of the system the Lagrangian is obtained by subtracting the potential energy from the kinetic energy. Determining the Lagrangian of the system is probably the most complex operation which then leaves the application of equation (3.5) to determine the torque applied at each joint. The advantage of the Lagrangian technique is that the constraint forces acting at the joints involved in the system are automatically eliminated.

3.3 Particular solution for the RTX 'SCARA' type robot manipulator

The SCARA (Selectively Compliant Arm for Robot Assembly) robot is a jointed manipulator with the arm mounted horizontally so that the arm's links do not have to move against gravity. The weight of the arm is taken by the vertical pedestal. A diagram of the manipulator under consideration is shown in Fig.3.4.

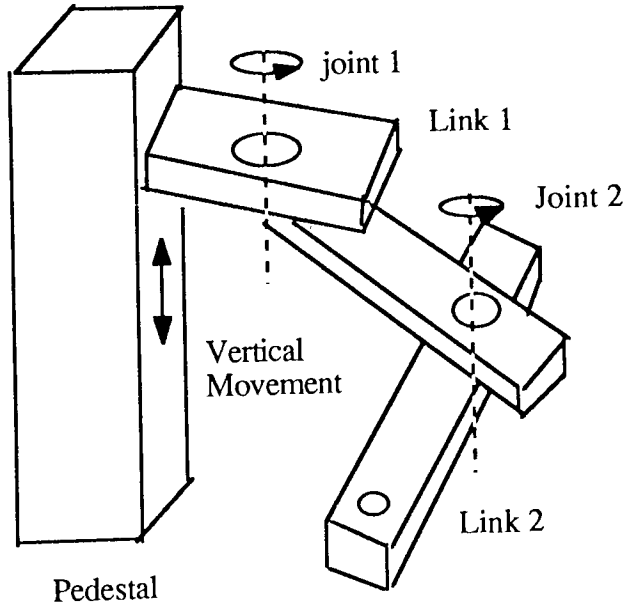


Figure 3.4. Diagram of a SCARA type manipulator.

In the SCARA type robot manipulators the two links of the horizontal arm must be driven by actuators at the revolute joints. The motors driving the links are normally positioned in the preceeding link i.e link n is driven by actuator n that is fixed in link n-1. The position of the motors in the RTX arm follow this configuration and are shown in Fig.3.5 .

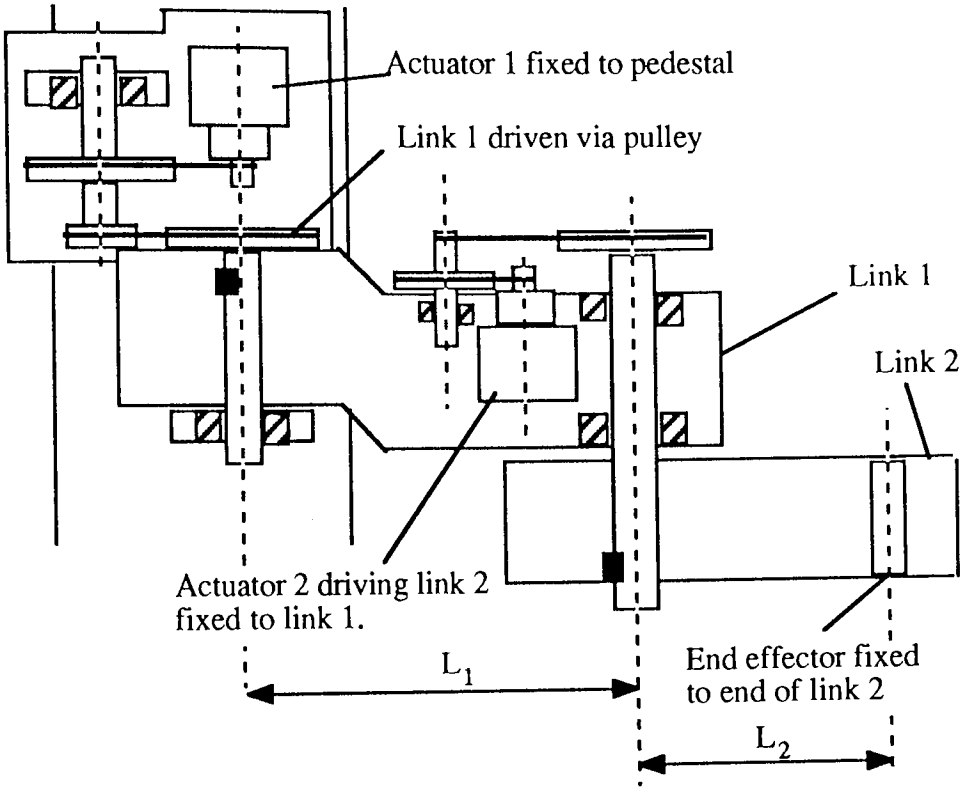


Figure 3.5. Cut away diagram of the RTX robot showing the position of the motors and pulleys driving the links.

The parameters for the RTX[4] are as follows:

L_1= 0.2535 m L_2= 0.2535 m

Drive Ratio 1 (ω_{01}/ω_{m1}) = 0.00114072

Drive Ratio 2 (ω_{02}/ω_{m2}) =0.00228145

where suffix $_0$ refers to output side and suffix $_m$ motor side

The inertial parameters of the links including masses m_1,m_2, inertia about the certre of gravity I_1,I_2 and the position of the centre of gravity from the rotation axis A_1,A_2 are as follows:

m_1= 2.64 Kg m_2= 3.31 Kg (13.31 Kg for a 10 Kg load)

I_1= 0.02392 Kgm^2 I_2= 0.033874 Kgm^2(and 0.0490233 Kgm^2 for 10 Kg load)

A_1= 0.117841 m A_2= 0.17545 m (and 0.234 m for 10 Kg load)

The actuator inertia's are 9.0 E-6 Kgm^2 (not referred)

The postion of the actuators for SCARA type manipulators are not always positioned as the in the RTX manipulator. For instance in the Adept[6] manipulator the motor driving link 2 is positioned at the pedestal along with the motor driving link 1 so that the reaction torque of motor 2 is not transmitted to link 1. The dynamic equations for this manipulator are thus different to that of the RTX type configuration manipulators. The dynamic equations will be derived for both type of manipulators using the Newton-Euler and Lagrangian techniques. The RTX manipulator type will be considered initially using Newtons second Law of motion.

In order to obtain a better understanding of the manipulator two corresponding free body diagrams have been draw in Fig.3.6, the first representing the Mass accelerations on the body and the second showing the forces causing the accelerations. The forces that occur at joint 2 are resolved in the tangential and radial directions.

From Fig.3.6 the moment balance for link 1 is as follows:

$$T_1 - T_2 + T_p L_1 = I_1 \ddot{\theta}_1 + m_1 A_1^2 \ddot{\theta}_1 \qquad (3.11)$$

For link 2 the moment balance is :

$$T_2 = I_2 (\ddot{\theta}_1 + \ddot{\theta}_2) + m_2 A_2^2 (\ddot{\theta}_1 + \ddot{\theta}_2) + m_2 L_1 A_2 \sin\theta_2 \dot{\theta}_1^2$$

$$+ m_2 L_1 A_2 \cos\theta_2 \ddot{\theta}_1 \qquad (3.12)$$

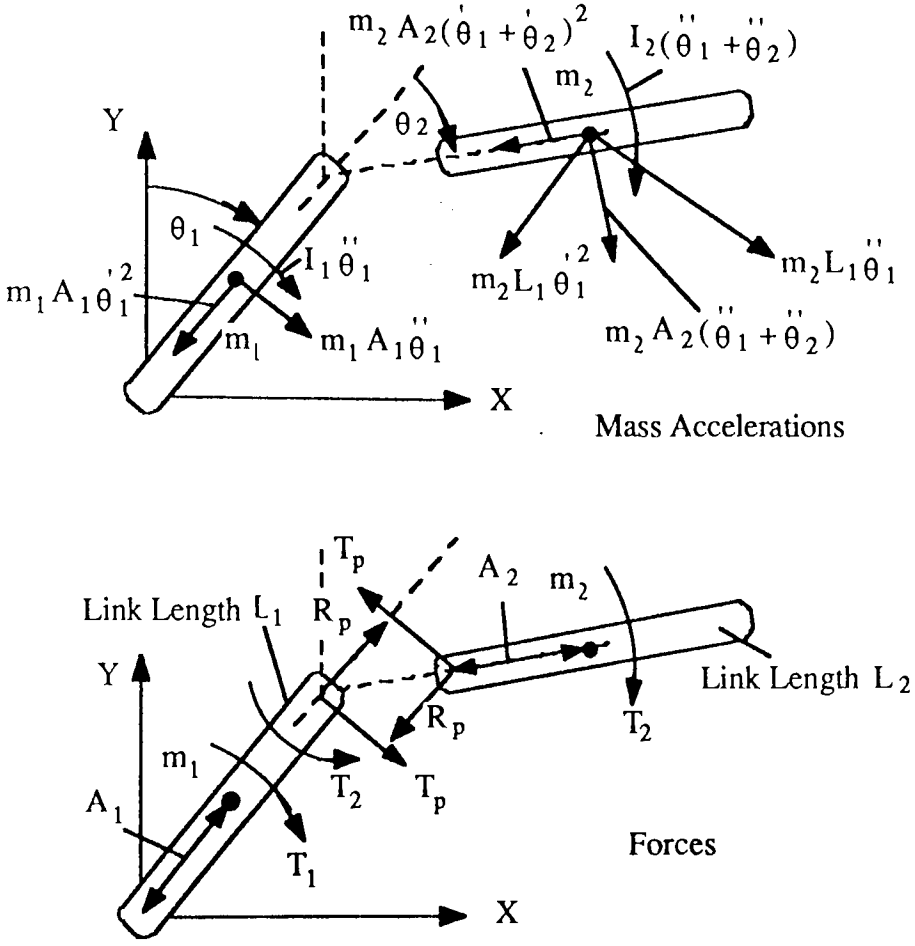


Figure 3.6. Diagram showing the mass accelerations and forces causing the mass accelerations for the two link RTX robot manipulator.

Resolving the forces in link 2 in the direction of T_p:

$$-T_p = m_2 L_1 \ddot{\theta}_1 + m_2 A_2 \cos\theta_2 (\ddot{\theta}_1 + \ddot{\theta}_2) - m_2 A_2 \sin\theta_2 (\dot{\theta}_1 + \dot{\theta}_2)^2 \quad (3.13)$$

Collecting terms in T_2

$$T_2 = (I_2 + m_2 A_2^2)\ddot{\theta}_2 + (m_2 L_1 A_2 \cos\theta_2 + m_2 A_2^2 + I_2)\ddot{\theta}_1$$

$$+ m_2 L_1 A_2 \sin\theta_2 \dot{\theta}_1^2 \quad (3.14)$$

Rearranging (3.11) in terms of T_1 and substituting for T_2 and T_p gives the

following:

$$T_1 = (I_1 + m_1 A_1^2 + I_2 + m_2 A_2^2 + m_2 L_1^2 + 2 m_2 L_1 A_2 \cos\theta_2) \ddot{\theta}_1$$

$$+ (I_2 + m_2 A_2^2 + m_2 L_1 A_2 \cos\theta_2) \ddot{\theta}_2 - 2 m_2 L_1 A_2 \sin\theta_2 \dot{\theta}_1 \dot{\theta}_2$$

$$- m_2 L_1 A_2 \sin\theta_2 \dot{\theta}_2^2 \quad (3.15)$$

Equations (3.14) and (3.15) will now be written in standard notation:

$$T_1 = D_{11} \ddot{\theta}_1 + D_{12} \ddot{\theta}_2 + D_{111} \dot{\theta}_1^2 + D_{122} \dot{\theta}_2^2 + D_{112} \dot{\theta}_1 \dot{\theta}_2 + D_1 \quad (3.16)$$

$$T_2 = D_{22} \ddot{\theta}_2 + D_{12} \ddot{\theta}_1 + D_{211} \dot{\theta}_1^2 + D_{222} \dot{\theta}_2^2 + D_{212} \dot{\theta}_1 \dot{\theta}_2 + D_2 \quad (3.17)$$

where

D_{ii} is known as the effective inertia at joint i, with an acceleration at joint i, causing a torque at joint i equal to $D_{ii} \ddot{\theta}_1$.

D_{ij} is known as the coupling inertia between joints i and j, with an acceleration at joint i or j, causing a torque at joint i or j, $D_{ij} \ddot{\theta}_j$.

$D_{ijj} \dot{\theta}_j^2$ is the centripetal force acting at joint i, due to the velocity at joint j.

$D_{ijk} \dot{\theta}_j \dot{\theta}_k$ is known as the Coriolis force acting at joint i, due to velocities j and k.

D_i represents the gravity forces at joint i.

Comparing coefficients in equations (3.15),(3.14) to equations (3.16),(3.17):

$$D_{11} = (I_1 + m_1 A_1^2 + I_2 + m_2 A_2^2 + m_2 L_1^2 + 2 m_2 L_1 A_2 \cos\theta_2)$$
$$D_{12} = (I_2 + m_2 A_2^2 + m_2 L_1 A_2 \cos\theta_2)$$

$$D_{111} = 0$$

$$D_{122} = - m_2 L_1 A_2 \sin\theta_2$$

$$D_{211} = m_2 L_1 A_2 \sin\theta_2$$

$$D_{112} = - 2\, m_2 L_1 A_2 \sin\theta_2$$

$$D_1 = D_2 = 0$$

The two arm links of the RTX robot manipulator operate in the horizontal plane. However, it is likely that a SCARA type robot could be mounted in its work place in a vertical plane. In this situation the links of the arm will be under the force of gravity. In order to take this situation in to account the torques acting at the joints must be added to the dynamic equations. As shown in Fig.3.7 the gravity field is assumed to act in the Y direction. The force due to gravity acts at the centre of mass.

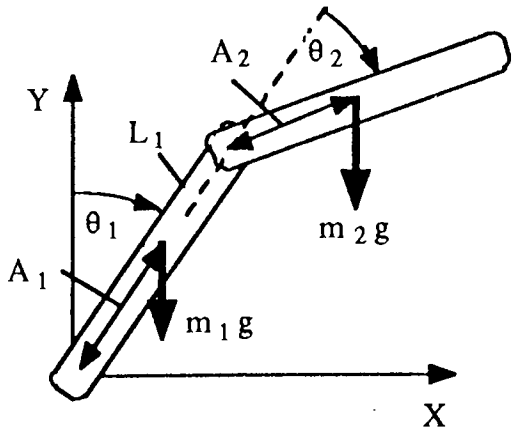


Figure 3.7. Diagram showing the gravity forces acting on the robot manipulator.

From Fig.3.7 the torque at joint 1 due to link 1 and 2 is :

$$D_1 = m_1 g A_1 \sin\theta_1 + m_2 g (L_1 \sin\theta_1 + A_2 \sin (\theta_1 + \theta_2)) \quad (3.18)$$

$$D_2 = m_2 g \; A_2 \sin (\theta_1 + \theta_2) \quad (3.19)$$

The dynamic equations can now be expressed in matrix form:

$$\mathbf{T} = \mathbf{M}(\theta)\,\ddot{\theta} + \mathbf{V}(\theta,\dot{\theta}) + \mathbf{G}(\theta) \qquad (3.20)$$

where

$\mathbf{M}(\theta)$ is an n*n inertia matrix
$\mathbf{V}(\theta,\dot{\theta})$ is an n*1 vector of centripetal and Coriolis terms
$G(\theta)$ is an n*1 vector of gravity terms

$$\mathbf{T} = \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} D_{122}\dot{\theta}_2^{\,2} + D_{112}\dot{\theta}_1\dot{\theta}_2 \\ D_{211}\dot{\theta}_1^{\,2} \end{bmatrix} + \begin{bmatrix} D_1 \\ D_2 \end{bmatrix} \qquad (3.21)$$

In order to obtain the dynamic equations in the forward dynamic format i.e. so that the motion of the manipulator can be observed for a given control input. The matrix equations of (3.23) are rearranged in terms of the joint accelerations:

$$\ddot{\theta} = \mathbf{M}^{-1}(\theta)\,[\,\mathbf{T} - \mathbf{V}(\theta,\dot{\theta}) - \mathbf{G}(\theta)\,] \qquad (3.22)$$

where $$\mathbf{M}^{-1}(\theta) = \frac{1}{D_{11}D_{22} - D_{12}D_{21}} \begin{bmatrix} D_{22} & -D_{12} \\ -D_{21} & D_{11} \end{bmatrix}$$

3.3.1 Lagrangian applied to the RTX

The Lagrangian is now applied to the manipulator of Fig.3.6. For link 1 the total kinetic energy is the rotational energy that it has about the Z_0 axis(at the origin of the X and Y axes out of the page). The inertia about axis 0 form the parallel axis is theorem is :

$I_{01} = I_1 + m_1 A_1^2$ where I_1 is the inertia about the centre of gravity

I_{01} is the inertia about the axis of rotation 0.

The kinetic energy of link 1 is thus:

$$K_{E1} = \frac{1}{2}(I_1 + m_1 A_1^{\,2})\,\dot{\theta}_1^{\,2} \qquad (3.23)$$

Link 2 has rotational K_{ER1} and translational K_{ET2} energy. The rotational energy about the centre of gravity is :

$$K_{ER2} = \frac{1}{2} I_2 (\dot{\theta}_1 + \dot{\theta}_2)^2 \qquad (3.24)$$

The translational energy is found be first differentiating the position vector to the centre of gravity of link 2 as follows:

$$X_2 = L_1 \sin\theta_1 + A_2 \sin(\theta_1 + \theta_2)$$

$$Y_2 = L_1 \cos\theta_1 + A_2 \cos(\theta_1 + \theta_2)$$

$$\dot{X}_2 = L_1 \cos\theta_1 \dot{\theta}_1 + A_2 \cos(\theta_1 + \theta_2)(\dot{\theta}_1 + \dot{\theta}_2)$$

$$\dot{Y}_2 = - L_1 \sin\theta_1 \dot{\theta}_1 - A_2 \sin(\theta_1 + \theta_2)(\dot{\theta}_1 + \dot{\theta}_2)$$

Squaring the velocity terms and summing gives:

$$\dot{X}_2^2 + \dot{Y}_2^2 = L_1^2 \dot{\theta}_1^2 + A_2^2 (\dot{\theta}_1^2 + \dot{\theta}_2^2 + 2\dot{\theta}_1\dot{\theta}_2) + 2 L_1 A_2 \cos\theta_2 (\dot{\theta}_1^2 + \dot{\theta}_1\dot{\theta}_2)$$

Hence the total kinematic energy of link 2 is:

$$K_{E2} = \frac{1}{2} I_2 (\dot{\theta}_1 + \dot{\theta}_2)^2 + \frac{1}{2} m_2 (L_1^2 \dot{\theta}_1^2 + A_2^2 (\dot{\theta}_1^2 + \dot{\theta}_2^2 + 2\dot{\theta}_1\dot{\theta}_2) + 2 L_1 A_2 \cos\theta_2 (\dot{\theta}_1^2 + \dot{\theta}_1\dot{\theta}_2))$$

Hence the Lagrangian is the total energy of the two links minus the potential energy. The potential energy is zero on the horizontal plane.

$$L = \frac{1}{2}(I_1 + m_1 A_1^2)\dot{\theta}_1^2 + \frac{1}{2} I_2 (\dot{\theta}_1^2 + \dot{\theta}_2^2 + 2\dot{\theta}_1\dot{\theta}_2) + \frac{1}{2} m_2 (L_1^2 \dot{\theta}_1^2 + A_2^2 (\dot{\theta}_1^2 + \dot{\theta}_2^2 + 2\dot{\theta}_1\dot{\theta}_2) + 2 L_1 A_2 \cos\theta_2 (\dot{\theta}_1^2 + \dot{\theta}_1\dot{\theta}_2)) \qquad (3.25)$$

The torque for each axis can now be found directly by application of equation (3.7):

$$\frac{\partial L}{\partial \dot{\theta}_1} = (I_1 + m_1 a_1^2)\dot{\theta}_1 + I_2(\dot{\theta}_1 + \dot{\theta}_2)$$
$$+ m_2(L_1^2\dot{\theta}_1 + A_2^2(\dot{\theta}_1 + \dot{\theta}_2) + 2L_1A_2\cos\theta_2(\dot{\theta}_1 + \frac{1}{2}\dot{\theta}_2)$$

$$\frac{d\partial L}{dt\,\partial \dot{\theta}_1} = (I_1 + m_1 a_1^2)\ddot{\theta}_1 + I_2(\ddot{\theta}_1 + \ddot{\theta}_2) + m_2(L_1^2\ddot{\theta}_1 + A_2^2(\ddot{\theta}_1 + \ddot{\theta}_2)$$
$$+ 2L_1A_2\cos\theta_2(\ddot{\theta}_1 + \frac{1}{2}\ddot{\theta}_2) - 2L_1A_2\sin\theta_2(\dot{\theta}_1\dot{\theta}_2 + \frac{1}{2}\dot{\theta}_2^2))$$

$$\frac{\partial L}{\partial \theta_1} = 0$$

$$T_1 = (I_1 + m_1A_1^2 + I_2 + m_2A_2^2 + m_2L_1^2 + 2m_2L_1A_2\cos\theta_2)\ddot{\theta}_1$$
$$+ (I_2 + m_2A_2^2 + m_2L_1A_2\cos\theta_2)\ddot{\theta}_2 - 2m_2L_1A_2\sin\theta_2\dot{\theta}_1\dot{\theta}_2$$
$$- m_2L_1A_2\sin\theta_2\dot{\theta}_2^2$$

The same result as found previously in as shown in equation (3.15).

Similarly T_2 can be found by application of equation (3.5):

$$\frac{\partial L}{\partial \dot{\theta}_2} = I_2(\dot{\theta}_1 + \dot{\theta}_2) + m_2(A_2^2(\dot{\theta}_1 + \dot{\theta}_2) + L_1A_2\cos\theta_2)$$

$$\frac{d\,\partial L}{dt\,\partial \dot{\theta}_2} = I_2(\ddot{\theta}_1 + \ddot{\theta}_2) + m_2(A_2^2(\ddot{\theta}_1 + \ddot{\theta}_2) - L_1A_2\sin\theta_2\dot{\theta}_1\dot{\theta}_2 + L_1A_2\cos\theta_2\ddot{\theta}_1)$$

$$\frac{\partial L}{\partial \theta_2} = -m_2L_1A_2\sin\theta_2(\dot{\theta}_1^2 + \dot{\theta}_1\dot{\theta}_2)$$

Hence, T_2 is as follows:

$$T_2 = (I_2 + m_2A_2^2)\ddot{\theta}_2 + (m_2L_1A_2\cos\theta_2 + m_2A_2^2 + I_2)\ddot{\theta}_1$$
$$+ m_2L_1A_2\sin\theta_2\dot{\theta}_1^2$$

i.e the same result previously as in equation (3.14)

The case where the joint actuators for the SCARA type manipulator are positioned both at the pedestal will now be solved using the Lagrangian technique. The generalised coordinate for link 2 is now chosen to be the absolute angle that link 2 makes with the Y axis rather than the relative angle to link 1 as shown in Fig.3.8.

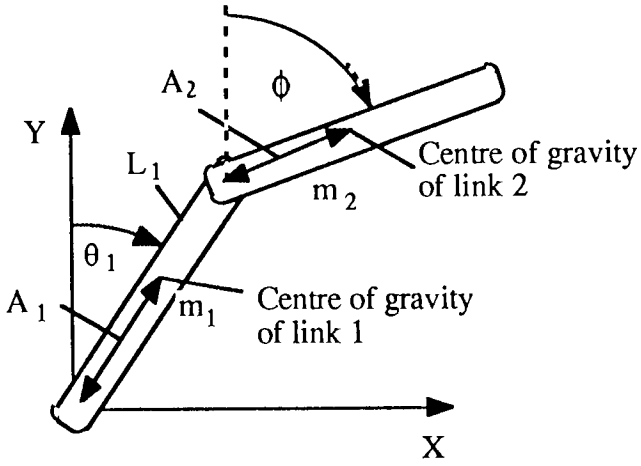


Figure 3.8. Diagram showing the generalised coordinates for the remotely driven SCARA type robot manipulator.

$$K_{E1} = \frac{1}{2}(I_1 + m_1 A_1^{\,2})\,\dot{\theta}_1^{\,2}$$

$$K_{e2} = \frac{1}{2} I_2 \dot{\phi}^2 + \frac{1}{2} m_2 (L_1^{\,2}\dot{\theta}_1^{\,2} + A_2^{\,2}\dot{\phi}^2 + 2 L_1 A_2 \cos(\phi-\theta_1)\,\dot{\theta}_1\dot{\phi})$$

$$L = \frac{1}{2}(I_1 + m_1 A_1^{\,2})\,\dot{\theta}_1^{\,2} + \frac{1}{2} I_2 \dot{\phi}^2 + \frac{1}{2} m_2 (L_1^{\,2}\dot{\theta}_1^{\,2} + A_2^{\,2}\dot{\phi}^2 + 2 L_1 A_2 \cos(\phi-\theta_1)\,\dot{\theta}_1\dot{\phi})$$

$$\frac{\partial L}{\partial \dot{\theta}_1} = (I_1 + m_1 A_1^{\,2})\,\dot{\theta}_1 + m_2 (L_1^{\,2}\dot{\theta}_1 + L_1 A_2 \cos(\phi-\theta_1)\,\dot{\phi})$$

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_1} = (I_1 + m_1 A_1^{\,2})\,\ddot{\theta}_1 + m_2 (L_1^{\,2}\ddot{\theta}_1 - L_1 A_2 \sin(\phi-\theta_1)\,(\dot{\phi}-\dot{\theta}_1)\,\dot{\phi} + L_1 A_2 \cos(\phi-\theta_1)\,\ddot{\phi})$$

$$\frac{\partial L}{\partial \theta_1} = m_2 L_1 A_2 \sin(\phi-\theta_1)\,\dot{\theta}_1\dot{\phi}$$

Hence T_1 is found to be :

$$T_1 = (I_1 + m_1 A_1^2 + m_2 L_1^2) \ddot{\theta}_1 + m_2 L_1 A_2 \cos(\phi-\theta_1) \ddot{\phi} - m_2 L_1 A_2 \sin(\phi-\theta_1) \dot{\phi}^2 \quad (3.26)$$

Since $\theta_1+ \theta_2=\phi$ then substituting in (3.26) gives:

$$T_1 = (I_1 + m_1 A_1^2 + m_2 L_1^2 + m_2 L_1 A_2 \cos\theta_2) \ddot{\theta}_1 + m_2 L_1 A_2 \cos\theta_2 \ddot{\theta}_2 - m_2 L_1 A_2 \sin\theta_2 (\dot{\theta}_1 +\dot{\theta}_2)^2 \quad (3.27)$$

For link 2 :

$$\frac{\partial L}{\partial \dot{\phi}} = I_2 \dot{\phi} + m_2 A_2^2 \dot{\phi} + m_2 L_1 A_2 \cos(\phi-\theta_1) \dot{\theta}_1$$

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\phi}} = I_2 \ddot{\phi} + m_2 A_2^2 \ddot{\phi} + m_2 L_1 A_2 \cos(\phi-\theta_1) \ddot{\theta}_1 - m_2 L_1 A_2 \sin(\phi-\theta_1)(\dot{\phi}-\dot{\theta}_1) \dot{\theta}_1$$

$$\frac{\partial L}{\partial \phi} = - m_2 L_1 A_2 \sin(\phi-\theta_1) \dot{\theta}_1 \dot{\phi}$$

$$T_2 = (I_2 + m_2 A_2^2) \ddot{\phi} + m_2 L_1 A_2 \cos(\phi-\theta_1) \ddot{\theta}_1 + m_2 L_1 A_2 \sin(\phi-\theta_1) \dot{\theta}_1^2 \quad (3.28)$$

Using the substitution $\theta_1+ \theta_2=\phi$ in equation (3.28) the result is the same as found previously in (3.14). When link 2 is driven from the pedestal the same torque is required to drive it as would be expected. However, the reaction torque is no longer reacted in link 1 so that if T_2 is not included in equation (3.13), in the Newton method, the torque equation for T_1 becomes the same as the Lagrangian result in equation (3.27).

So far two possible configurations of the motor attachment to the links have been considered. However, another possibility is if the motor driving link 2 is attached to the base but drives link 2 via a drive ratio(gear box or pulley mechanism) fixed to link 1. But since the intermediate drive is attached to link 2 according to Newtons law, 'for every action there is an equal and opposite reaction', hence the

torque T_2 driving link 2 will still be reacted in link 1 irrespective of the torque at the motor side of the drive.

3.3.2 Meaning of the terms in the dynamic equations

1) Effective inertia terms D_{11} and D_{22}

The effective inertia about joint 1, D_{11}= Inertia link 1 + Inertia link 2(about joint 1)

Inertia of link 1 about joint 1= $I_1+m_1 A_1^2$

Inertia of link 2 about joint 1= $I_2 +m_2 R^2$ where R is the distance of the centroid if link 2 to the joint 1 axis as seen in Fig.3.9 :

Applying the cosine rule to find distance R to Fig.3.9:

$$R^2 = L_1^2 + A_2^2 - 2 L_1 A_2 \cos(180-\theta_2)$$

but $\cos(180-\theta_2) = -\cos(\theta_2)$ therefore

$$R^2 = L_1^2 + A_2^2 + 2 L_1 A_2 \cos(\theta_2)$$

$$D_{11} = [(I_1 + m_1 A_1^2) + (I_2 + m_2 L_1^2 + m_2 A_2^2 + 2m_2 L_1 A_2 \cos(\theta_2))]$$

Link 1 Inertia Link 2 Inertia

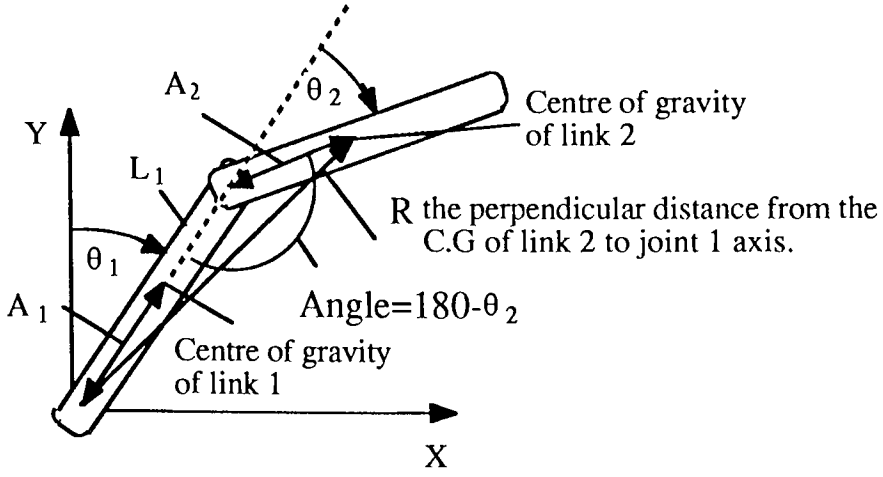


Figure 3.9. Diagram showing how varying inertia depends upon the arm configuration.

Inertia of link 2 about joint 2 using parallel axis is:

$$D_{22} = I_2 + m_2 A_2^2$$

2. Coupling terms D_{12}

If the case is considered when $\dot{\theta}_1 = \dot{\theta}_2 = \ddot{\theta}_1 = 0$ then $T_1 = D_{12}\ddot{\theta}_2$

For the RTX the motor driving link 2 is positioned in link 1 so that the torque T_2 is reacted in link 1 as seen in equation (3.11). As seen in this case the terms in equation (3.12) due to T_2 are :

$$(I_2 + m_2 A_2^{\,2})\ddot{\theta}_2$$

The other term in the coupling coefficient is due to the coupling force at the joint T_p as seen in equation (3.13). The tangential acceleration of link 2 resolved in the direction of T_p causing a torque :

$$m_2 L_1 A_2 \cos\theta_2 \ddot{\theta}_2$$

Similarly, when the case is considered $\dot{\theta}_1 = \dot{\theta}_2 = \ddot{\theta}_2 = 0$ then $T_2 = D_{12}\ddot{\theta}_1$ from equation (3.14).

3. Centrifugal terms

If the case is considered when $\dot{\theta}_2 = \ddot{\theta}_1 = \ddot{\theta}_2 = 0$

The motion of link 1 causes link 2 to be swung out in the direction of R_p as shown in Fig. 3.6 Mass Accelerations. The torque on link 2 due to this centrifugal force is thus :

$$T_{Cent} = m_2 L_1 A_2 \sin\theta_2 \dot{\theta}_1^2$$

Similarly, if the case is considered when $\dot{\theta}_1 = \ddot{\theta}_1 = \ddot{\theta}_2 = 0$

The centrifugal force of link 2 causes a force in link 1 as seen in the equation (3.13) for T_p. This torque is thus equal to:

$$T_{cent} = -m_2 L_1 A_2 \sin\theta_2 \dot{\theta}_2^2$$

4. Coriolis terms D_{112}

When a mass m moves at a velocity V_r relative to a moving coordinate frame rotating at an angular velocity ω, the mass is acted on by a Coriolis force given by:

$2 m (\omega * V_r)$ (3.29) where * refers to the cross product

If the case when the two joints rotate at velocities $\dot{\theta}_1$ and $\dot{\theta}_2$ at the same time is considered as shown in Fig.3.10:

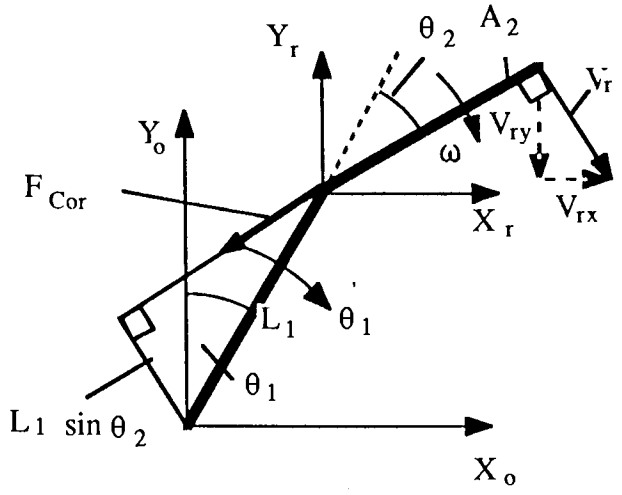


Figure 3.10. Diagram showing the Coriolis effect in the robot manipulator.

If a coordinate frame X_r-Y_r is attached to the tip of link 1,as shown in Fig.3.10 , it moves with the angular velocity of link 1 and at that instant is parallel to the base coordinate frame. The relative velocity components of link 2 to the relative coordinate frame are :

$$V_{rx} = A_2 \dot{\theta}_2 \cos(\theta_1 + \theta_2) \qquad V_{ry} = -A_2 \dot{\theta}_2 \sin(\theta_1 + \theta_2)$$

$$\omega = 0_x + 0_y - \dot{\theta}_{1z}$$

Using equation (3.29) to determine the force components :

$$F_x = +2 m_2 A_2 \sin(\theta_1 + \theta_2) \dot{\theta}_1 \dot{\theta}_2$$

$$F_y = -2 m_2 A_2 \cos(\theta_1 + \theta_2) \dot{\theta}_1 \dot{\theta}_2$$

The resultant force thus acts parallel with link 2 and causes a torque of T_{cor} on link 1 of magnitude:

$$T_{cor} = 2 m_2 L_1 A_2 \sin\theta_2 \dot{\theta}_1 \dot{\theta}_2$$

This is the value shown in equation (3.15) for T_1.

The equations (3.26),(3.28) for the SCARA type manipulator with the second link driven by a motor mounted on the pedestal will now be discussed.

The T_2 equation remains the same except the relative angular displacements and the derivatives are converted in terms of the θ_1 and ϕ. The effective inertia of link 2 remains the same while the coupling inertia is reduced to:

$$D_{12} = m_2 L_1 A_2 \cos(\phi - \theta_1)$$

The effective inertia of link 1 is now reduced to :

$$D_{11} = I_1 + m_1 A_1^2 + m_2 L_1^2$$

where since the torque T_2 is not reacted in link 1 , joint 1 only sees the inertia of link 1 about joint 1 , plus the mass of link 2 acting at the link 1 which is transmitted by the joint coupling forces(T_p).

There is no Coriolis force because now the choice of reference frame for link 2 is not rotating with respect to the fixed plane.

The centrifugal terms remain the same since these terms are still transmitted at joint 2 by the coupling forces.

To conclude this chapter in comparing the equations for the two types of SCARA motor configuration it is seen that for the remotely driven link 2 the non-linear terms have been reduced. The effective inertia in link 1 is not variable as in the RTX and there is no longer any Coriolis term in the torque equation for link 1.

Chapter 4:Control Strategies

Chapter 4 - Control Strategies

4.1 Introduction

The control of robot manipulators has attracted a great deal of interest in recent years and hence there has been an abundance of literature published on the subject. The design of control systems for robot manipulators requires an understanding of the dynamics of the manipulator in order to appreciate the challenge that it presents. As seen in chapter 3 the dynamics of the manipulator are represented by second order, highly coupled non-linear equations. This makes designing a control system for the manipulator extremely difficult since well established classical design methods were developed for linear systems. In this chapter the various approaches to robot manipulator control system design are reviewed and the limitations of the current knowledge highlighted.

The methods of approach to robot manipulator control system design can be broadly classified into three categories:

1. Joint Motion Control (JMC)
2. Resolved Motion Control (RMC)
3. Adaptive Control (AC)

4.1.1 Joint Motion Control

In JMC the error signal to the controller is obtained from the difference between the desired joint position and the actual joint position . The desired joint position is obtained from the conversion of the cartesian set points into joint set points using an inverse kinematics algorithm.

4.1.2 Resolved Motion Control

In RMC the error signal to the controller is obtained from the difference between the Cartesian set point and the actual cartesian position. The Cartesian position is obtained by converting the actual joint positions using the forward kinematics. The error signal in Cartesian space is then converted, using an inverse kinematics algorithm , into equivalent joint error space.

4.1.3 Adaptive Control (AC)

Adaptive control systems differ from the two previous categories in that they can adapt to changes in the dynamics of the system. They usually do not require the parameters of the system to be accurately known.

The three broad classes of control schemes can be subdivided into more specific categories. First the JMC schemes are considered:

4.2 Joint Motion Control schemes

a) Independent axis control
b) Computed torque
c) Minimum time control
d) Variable structure control
e) Non linear control
f) Cerebellar model articulation control
g) Fuzzy logic

The notation that will be used to represent the manipulator is that used previously in chapter 3 equation (3.22) representing the dynamics of the manipulator::

$$\mathbf{T} = \mathbf{M}(\theta)\,\ddot{\theta} + \mathbf{V}(\theta,\dot{\theta}) + \mathbf{G}(\theta) \qquad (4.1)$$

where

$\mathbf{M}(\theta)$ is an n*n inertia matrix
$\mathbf{V}(\theta,\dot{\theta})$ is an n*1 vector of centripetal and Coriolis terms
$G(\theta)$ is an n*1 vector of gravity terms

$$\mathbf{T} = \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} D_{122}\dot{\theta}_2^{\,2} + D_{112}\dot{\theta}_1\dot{\theta}_2 \\ D_{211}\dot{\theta}_1^{\,2} \end{bmatrix} + \begin{bmatrix} D_1 \\ D_2 \end{bmatrix} \qquad (4.2)$$

Independent axis and computed torque control will be considered together.

4.2.1 Independent axis and Computed torque

The basic control structure of these two methods can be represented by the following multivariable block diagram, Fig.4.1.

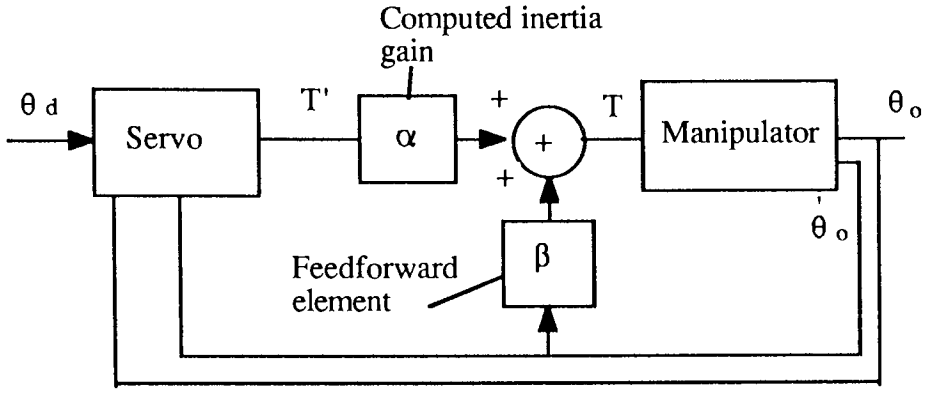


Figure 4.1. Showing the structure of the computed torque controller.

As seen in Fig.4.1 the structure of the control system involves a servo portion multiplied by a inertial gain term and a feedforward element. The ouput joint angles and their first derivatives are fed back to the servo and used in the calculation of the feedforward element.

In traditional control system design for robot manipulators each axis of the manipulator is considered as an effective inertia and all coupling terms between axes are ignored. This type of control system design method is referred to as independent axis design. This type of design is evident in many current industrial robots e.g. the Puma 560 series of robot arms as described in Fu[7]. The Puma 560 series use a PID closed control around each axis, the controller coefficients of are fixed gain.

In the notation used for two axes of the manipulator the control system is of the form:

Let $E = \theta_d - \theta_o$ and $\dot{E} = \dot{\theta}_d - \dot{\theta}_o$

$$T_1' = K_{v1}\dot{E}_1 + K_{p1}E + K_{I1}\int E\,dt$$

$$T_2' = K_{v2}\dot{E}_2 + K_{p2}E + K_{I2}\int E\,dt$$

$\alpha = I$

where I is a 2 by 2 identity matrix

$\beta = 0$

and K_{pi} is the proportional gain controller coefficient

K_{vi} is the derivative gain controller coefficient

K_{Ii} is the integral gain controller coefficient.

The justification for using this type of control system is that the coupling terms are neglible compared to the inertia torque. This is sometimes the case when the gearing between actuator and link axis is high. However, if the coupling terms are significant then they can substantially affect the performance of the manipulator, particularly at high speeds. In fact there is no literature to my knowledge on independent axis control systems that shows how the coupling terms affect the manipulator dynamic performance. This is a particular line of research that will be pursued by the author in this thesis.

The computed torque method is a model based control system design it is sometimes referred to feedforward control. It requires the parameters of the manipulator to be accurately determined. The method was first used by Paul[8] on the Stanford robot and also by Bejczy[9]. The design of such a controller is presented in the form of a tutorial by Luh[10].

The principle of the method is to make the non-linear manipulator appear to act as a unit inertia by feeding forward the computed values of the coupling terms. The servo portion of the controller is then added to the computed torque in order to steer the system in the presence of disturbances that cause errors.

The controller is of the form:

$$T = \alpha T' + \beta \qquad (4.3)$$

let $I(\theta)$ = Diagonal inertia matrix

$I_o(\theta)$ = Off diagonal inertial coupling matrix

and subscript c refers to the computed values of the parameters

$$\alpha = I_c(\theta)$$
$$\beta = I_{oc}(q) + V_c(\theta,\dot{\theta}) + G_c(q)$$

The servo portion of the controller can be a number of types, for instance consider a PD system with acceleration feedforward as follows:

$$T' = \ddot{\theta}_d + K_v(\dot{\theta}_d - \dot{\theta}_0) + K_p(\theta_d - \theta_o)$$

where

K_p is the proportional gain controller coefficient

K_v is the derivative gain controller coefficient

Hence, the computed torque to the manipulator's actuator is:

$$T = I_c(\theta)(\ddot{\theta}_d + K_v(\dot{\theta}_d - \dot{\theta}_0) + K_p(\theta_d - \theta_o)) + I_{oc}(\theta) + V_c(\theta) + G_c(\theta) \qquad (4.4)$$

This can then be equated to equation (4.1) for the actual system:

$$I(\theta)\ddot{\theta} + V(\theta,\dot{\theta}) + G(\theta) + I_o(\theta)\ddot{\theta} = I_c(\theta)(\ddot{\theta}_d + K_v(\dot{\theta}_d - \dot{\theta}_0) + K_p(\theta_d - \theta_o)) + I_{oc}(\theta)\ddot{\theta}_d + V_c(\theta,\dot{\theta}) + G_c(\theta) \qquad (4.5)$$

Providing $V_c(\theta,\dot{\theta}) = V(\theta,\dot{\theta})$, $G_c(\theta) = G(\theta)$ and $I_c(\theta) = I(\theta)$

Then the actual nonlinear parameters of the left hand side cancel with the corresponding computed values on the right hand side.
The transfer function for the system is as follows:

$$\theta_o = \frac{(S^2 + K_v S + K_p) I_c(\theta)\ \theta_d}{I(\theta) S^2 + I_c(\theta) K_v S + I_c(\theta) K_p} \qquad (4.6)$$

Let $I_c(\theta) / I(\theta) = C$ then equation(4.6) can be written as:

$$\theta_o = \frac{(S^2 + K_v S + K_p)\,\theta_d}{S^2/C + K_v S + K_p} \qquad (4.7)$$

For this particular controller, providing $I_c(\theta) = I(\theta)$ then the numerator cancels with the denominator and there should be perfect dynamic tracking. This can written alternatively as follows:

$$(\ddot{\theta}_d - \ddot{\theta}_o) + K_v(\dot{\theta}_d - \dot{\theta}_o) + K_p(\theta_d - \theta_o) = 0 \qquad (4.8)$$

Substituting for the difference between the desired and actual joint angle equation (4.8) becomes:

$$\ddot{E} + K_v \dot{E} + K_p E = 0 \qquad (4.9)$$

where $E = \theta_d - \theta_o$

This is the system equation in error space and has the form:

$$(S^2 + 2\zeta\omega_n S + \omega_n^2)\,E = 0$$

Hence, the controller coefficients can be chosen to give the desired natural frequency and damping ratio.

There are many possible combinations of servo control possible for implementation with the feedforward controller. It is not evident in the literature which of the combinations perform the best in dynamic tracking particularly when the feedforward component is not present. The servo controller must deal with the full force of the coupling terms in these circumstances. This is a particular area of investigation that the author feels is worth pursuing.

The literature to date on the computed torque has shown mainly simulation results however, An[11] in 1986 show experimental results for the MIT serial link direct drive arm. He tested a PD control system on each axis of the manipulator and then added to this an off-line computed torque method and an on-line computed torque method. A fifth order polynomial in joint space was used to generate the trajectory. The tracking ability of the PD system alone was significantly improved upon with the introduction of the off-line feedforward dynamics for the first two

axes. The third axis tracking ability was not improved upon due to the dominance of unmodelled dynamics in the motor and the small inertia of this axis. The on-line computed torque method did not show any significant performance over the off-line computed torque method.

4.2.2 Minimum time control

The objective of minimum time control is to minimize the time taken for a manipulator to move from an initial position to a final position. This is a time optimal problem that Kahn[12] investigated. The performance index that this to be minimized is thus defined is the task cycle time. Due to the fact that the equations of the manipulator are nonlinear only a numerical solution to the equations can be found to the time optimal problem that does not take into account the system disturbances. Additionally the solution would have to be computed for each manipulator motion which for real time tracking would be unsuitable. To over come these problems Kahn proposed an approximation to the time optimal problem by representing the nonlinear equations by their linearised versions. This method of approximation was referred to as near-minimum time control.

The method was shown to give reasonably close results to the time optimal solutions. The method is extremely complex for manipulators having more than four degrees of freedom. The effects of unknown disturbance loads is not taken into account in the solution.

4.2.3 Variable Structure Control

The strategy of variable structure controller design is totally different to conventional control system designs. Normally control systems have a fixed structure, in variable structure systems the structure of the controller changes as a function of the states, to achieve certain desired objectives. It is often referred to as sliding mode control and was a theory for designing switching feedback regulators. It was developed primarily in the Soviet Union as an outgrowth of earlier work on open loop minimum time 'bang bang' control. The control method can be explained by considering a second order system with position feedback in equation (4.10):

$$\ddot{x} + a\dot{x} + (b+K)x = 0 \qquad (4.10)$$

The structure of the system in the phase plane depends upon the value of K and the trajectories of the system can be a number of types. For example if a=0 and K takes the following values:

$K = \alpha - b$, $K = -\alpha - b$ for $\alpha > 1$

The corresponding trajectories that the system follows are either elliptical or saddles as shown in fig.4.2 below:

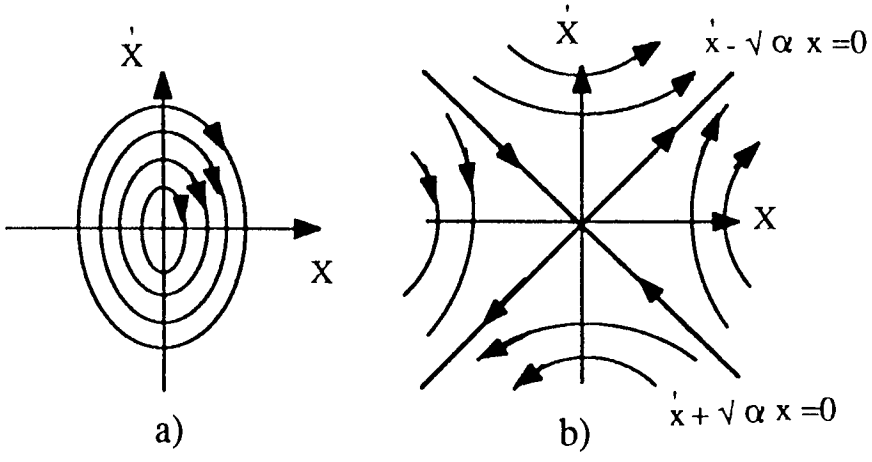


Figure 4.2. Showing the trajectory types of elliptical or saddle.

If the phase plane is divided into four regions Fig.4.3 a) and switching occurs at the boundaries of each region and the switching control is chosen to be:

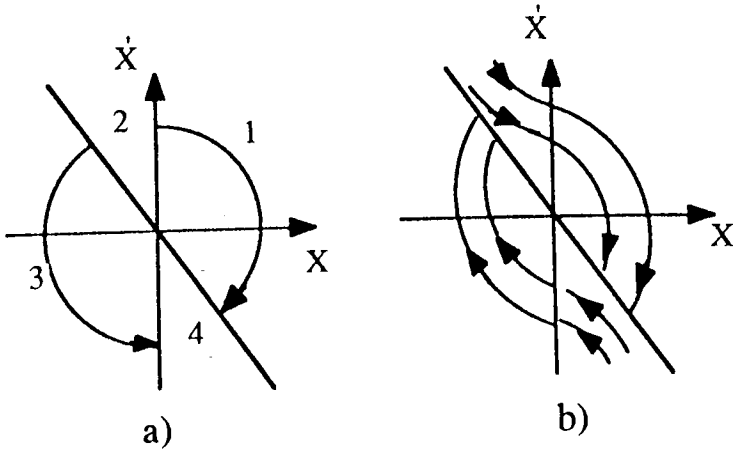


Figure 4.3. Showing the resultant trajectories when switching occurs.

$U(t) = (\alpha - b)\, x$ in regions 1 and 3

$U(t) = (-\alpha - b)\, x$ in regions 2 and 4

The resultant closed loop trajectory then takes the form shown in Fig.4.3 b). The resulting system is globally asymptotically stable even though the two feedback sytems from which it was formed were stable and unstable respectively.

The trajectories of the closed loop switching system were obtained from parts of the non-switching feedback systems hence, the trajectories resemble a non-switching trajectory at finite intervals. Trajectories can be obtained that do not resemble the non-switching trajectory. For instance instead of switching on the line shown in Fig.4.3 a) it is possible to switch on the line defined by say:

$\dot{x} + c\, x = 0$ where $0 < c < \sqrt{\alpha}$

If the sytem with elliptical trajectories is made to switch on line $L_1 = C_1$ the and the saddle system is made to switch on the line L_2 with $C = C_2$ ($C_1 > C_2$) and the initial conditions are:

$\dot{x}(0) > 0$ and $x(0) > 0$

Then the trajectory is an elipse until it hits L_1, at which point it switches to the saddle trajectory moving towards L_2. When switching occurs again on L_2 the trajectory is elliptical again until it reaches L_1. The resulting trajectory remains in the shaded area R and zigzags to the origin as seen in Fig.4.4. If C_1 and C_2 tend to C then the trajectory switches with infinite frequency on the line :

$\dot{x} + c\, x = 0$

resulting in a zigzag line. This rapid switching would not occur in a physical system since there are always delays. However, it is possible to idealise the situation and say that the trajectory is forced to remain on the switching line. The trajectory moves slowly down the switching line to the origin and is said to be in the sliding mode. The motion is no longer composed of the subtrajectories but is a

new motion altogether.

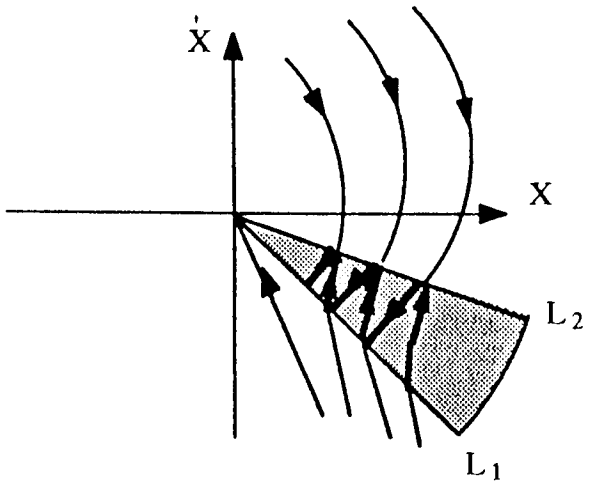


Figure 4.4. Showing the trajectory when switching occurs between two lines.

The motion along the switching line is defined by the equation :

$$\dot{x} + c\,x = 0$$

it is independent of the original system parameter b, providing the sliding mode continues to exist. This is in essence the approach to variable structure systems design.

In Young[13] a variable structure controller was designed for a two degree of freedom manipulator and tested in simulation for a step input demand in each axis. The control system was shown to perform well in idealised conditions. Others who have applied variable structure design approach to robot manipulators include Staszulonek[14], he inertially decoupled the axes of the manipulator and then applied the design method. The manipulator was found to perform 40% better than a PID control system in simulation. Also Harishima[15] applied the principle of variable structure systems to a manipulator control system design. However, a non-linear compensation term was also used in the design. The system was tested in simulation tracking a trajectory and was found to perform reasonably well. To my knowledge the variable structure system has not been tested experimentally.

The advantages of variable structure design is that it requires less prior knowledge of the parameters of the system. Providing a sliding mode is obtained the system

remains insensitive to parameter variations and disturbances and hence, nonlinear interactions between joints can be eliminated. However, the variable structure control produces a discontinuous feedback control signal which changes sign rapidly which could excite structural resonant frequencies of the system.

4.2.4 Nonlinear control

In nonlinear control the objective is to determine control laws that decouple the nonlinear equations of the system and allow the poles of the system to be arbitrarily assigned. Freund[16] applied a nonlinear decoupled control to robot manipulators. The result was a nonlinear statefeedback controller that required full state feedback of all joint angles and velocities. The controller is very similar to Luh's[17] resolved acceleration control system(this is considered later in the chapter). The method was tested in simulation and was shown to perform far better than an equivalent linearised control system.

4.2.5 Cerebellar Model Articulation Controller(CMAC)

The CMAC is claimed to be an adaptive controller however, it appears to be better classified as a learning controller since it does not adapt in real time but changes controller gains according to previous knowledge of other paths. The CMAC was devised by Albus[18,19] and is based upon neurophysiological theory. The method does not require modelling of the dynamics of the manipulator. It requires the manipulator to be taught a motion, while it observes and records the joint variables during the motion. The internal parameters of the system are then adjusted in direction that is calculated to improve upon the tracking ability. A large number of discrete paths must be stored from which interpolated paths can be generated. The method requires the necessity to control a large amount of computer memory. The accuracy of the interpolated path may not be within the tracking requirements. This method has not been tested experimentally by Albus and has not been researched by any other authors.

4.2.6 Fuzzy Logic Controller

Fuzzy logic controllers were initially developed over the last two decades for the control of cement kilns in Spain and a number of industrial applications in Denmark. The derivation of transfer functions for the control design of cement

kilns was found to be extremely difficult if not impossible. So the control of the kilns was characterised by a set of Fuzzy logic rules. These were then used as means of establishing a control strategy. The Fuzzy logic rules do not change during the course of execution of the controller. The idea of Fuzzy logic has recently been applied to the control of robot manipulators by Scharf[20,21]. However, Scharf developed what is termed a Self Organizing Controller(SOC) which acts as a learning controller.

In Scharf's controller is a fixed fuzzy rule based controller which defines the control strategy and an additional SOC capability which enables it to create and modify the rules in the rule base. The controller receives the joint error and change in joint error signals. The manipulator used was of a jointed type and was tested in tracking a vertical square trajectory. The fuzzy logic controller was shown to exhibit less deviation from the required path than a fixed term PID controller that was also tested on the same trajectory. The speed that is was required to track at was not mentioned and thus it should be questioned how stable the system is at high tracking speeds and also whether the SOC could cause instability in certain circumstances.

4.3. Resolved Motion Control schemes

These can be divided into two sub-classes:

a) Resolved motion control
b) Resolved motion acceleration control

4.3.1 Resolved motion control

The idea of resolved motion control came about by the need to control human prostheses in coordinated motion.In this type of application the desired motion rates are specified in world coordinates rather than at joint level. In order to coordinate a jointed arm in world coordinates Whitney[22] made use of the Jacobian that relates differential motions in cartesian space to differential motion in joint space, as discussed in chapter 2. Hence the basic relationship between joint velocities and cartesian velocities is :

$$\dot{\underline{X}} = J\,\dot{\underline{\theta}} \qquad (4.11)$$

where J is the Jacobian

In order to specify the cartesian velocities the inverse Jacobian is found and thus we have :

$\dot{\underline{\theta}} = J^{-1}\,\dot{\underline{X}}$ where J^{-1} is the inverse Jacobian

The multivariable control system block daigram is of the form shown in Fig.4.5:

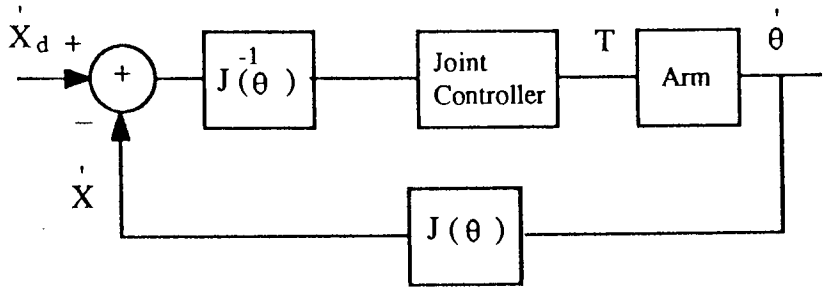


Figure 4.5. Showing the multivariable block diagram for resolved motion control.

As seen in Fig.4.5 the inverse Jacobian is used to convert the Cartesian velocitiy error into equivalent joint velocity error.The joint velocites which are fed back are converted into Cartesian velocities using the Jacobian. There can be a number of types e.g PD, PID etc. This is basically independent axis control since the dynamic coupling between joint axes is not taken into account.

4.3.2 Resolved Motion Acceleration Control

This is an extension of the concept of resolved motion rate control by Luh[17] to include acceleration control. Again the desired position and orientation of the hand are assumed to be known in advance including the accelerations. If equation (4.11) is differentiated with respect to time to find the acceleration relationship:

$$\ddot{\underline{X}} = J\,\ddot{\underline{\theta}} + \dot{J}\,\dot{\underline{\theta}} \qquad (4.12)$$

where $\dot{J}$ is the derivative of the Jacobian.

The controller in Cartesian space is of the form :

$$\ddot{X}_d + K_v (\dot{X}_d - \dot{X}) + K_p (X_d - X) = \ddot{X} \qquad (4.13)$$

Where the actual cartesian positions are found by using the forward kinematic and Jacobian algorithms. The required joint acceleration is found by substituting equation (4.13) into (4.12):

$$\ddot{\theta} = - K_v \dot{\theta} - J^{-1} [\ddot{X}_d + K_v (\dot{X}_d - \dot{X}) + K_p (X_d - X) - \dot{J} \dot{\theta}] \qquad (4.15)$$

The control torques to the manipulator can then be calculated, (with the knowledge of the required joint accelerations of equation (4.15) together with the joint velocities and positions fedback from joint transducers), from equation (4.1).

A simulation of this method was tested by Luh[17] on the Stanford six degree of freedom manipulator when following a required trajectory. The method was shown to work reasonably well in this simulated test.

The resolved motion acceleration control method requires extensive computation associated with calculating the Jacobian in real time and there are problems if the manipulator passes through a singularity. The method has not been implemented practically to my knowlege.

4.4 Adaptive Control

Many of the control methods considered so far rely on the parameters of the manipulator to be accurately known. However, when the parameters cannot be determined accurately this will result in tracking errors occuring. Also the manipulator may be required to pick up a pay load of unknown mass which could drastically affect the performance of the control system. In order to make the control system more robust i.e. insensitive to parameter variations or unmodelled dynamics, considerable research has been carried out in developing adaptive control algorithms. The adaptive algorithms can be divided into three sub-categories :

a) Model Reference Adaptive Control (MRAC)
b) Self Tuning Control (STC)

c) Adaptive Perturbation Control (APC)

4.4.1 Model Reference Adaptive Control

In MRAC the manipulator is modelled by a set of differential equations. The controller gains of the feedback control system are adjusted by an adaptation algorithm that is driven by the error between the actual manipulator response and the response of the model. The basic block diagram for this type of control system is shown in Fig.4.6 below:

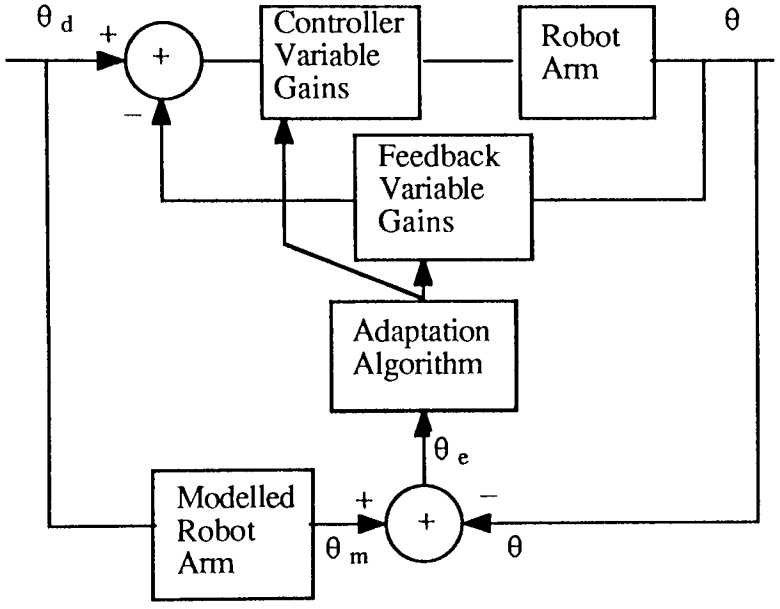


Figure 4.6. Block diagram showing the structure of Model Reference Adaptive Control(MRAC).

One of notable researchers to have applied the method of MRAC to robot manipulators was Dubowsky[23]. He applied the method to a six degree of freedom jointed manipulator in which the payload was combined with the third link since the hand was considered to be neglible distance from the wrist axis. The first three axes of the manipulator were modelled by second order time invarient differential equations. The coupling between the three axes was not considered in the analysis. The structure of the controller was proportional with velocity feedback. The adaptation algorithm was based upon the steepest desent method that minimises a quadratic function of error between modelled response and actual

response. The MRAC was tested in simulation in which the manipulator was required to perform a number of pick and place type operations. It was seen to perform far better that the fixed gain equivalent controller.

The advantages of MRAC are that does not require complex modelling of the manipulator dynamics. However, stability considerations are critical and Dubowsky used a linearised model for this purpose. They tested the control system at relatively slow speeds in which the dynamic coupling between joints may be justifiably small but at high speed the control system may not perform so well or could go unstable. There is still a lot of work to be done on this type of controller.

4.4.2 Self Tuning Control (STC)

The characteristics of STC are a identification algorithm, a control sysnthesis algorithm and the controller implementation. A block diagram of the STC structure is shown in Fig.4.7 below:

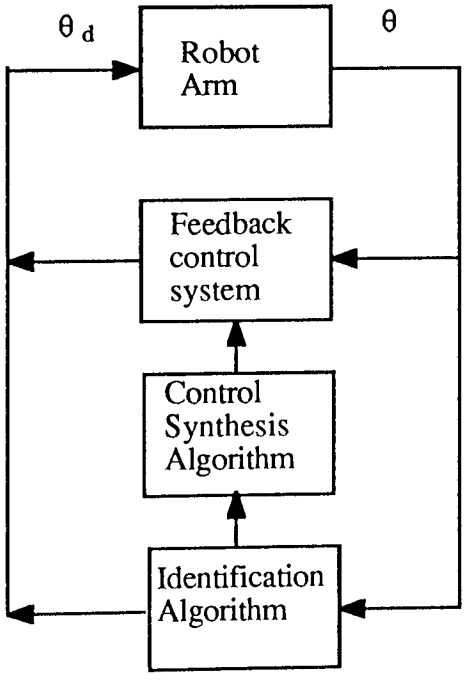


Figure 4.7. Showing the basic structure of a Self Tuning Control system.

This method of control was applied by Koivo[24] to the control of a manipulator. The control algorithm was designed assuming interaction between joints was

neglible. An autoregressive model was used to identify the parameters of the manipulator and an optimal controller synthesis algorithm. Simulation results were presented which showed resonable tracking ability. The advantage of this type of method is that the parameters of the manipulator do not have to be accurately determined. However, it is questionable with this controller design whether it could identify system parameters at a fast enough speed to benefit the controller implementation and whether it would be stable at high speeds.

4.4.3 Adaptive Perturbation Control (APC)

This method was used by Lee[25], it differs from the other adaptive control methods considered so far in that the coupling between joint axes is taken into account. This is done by computing the nominal torques required to achieve the desired trajectory. The adaptive part of the method is based on the linearised perturbation equations in the vacinity of the trajectory. The manipulator equations are linearised about the planned trajectory to obtain the linearised perturbation equations. The feedback controller computes the perturbation torques which will reduce the position and the velocity errors along the nominal trajectory to zero. The perturbation linearised equation coefficients are identified using a recursive least squares identification algorithm. The perturbation torques are obtained from an optimal control system. This type of adaptive controller reduces the nonlinear control problem to controlling the linear system about a nominal trajectory. The block diagram for this control strategy is shown in Fig.4.8 as follows:

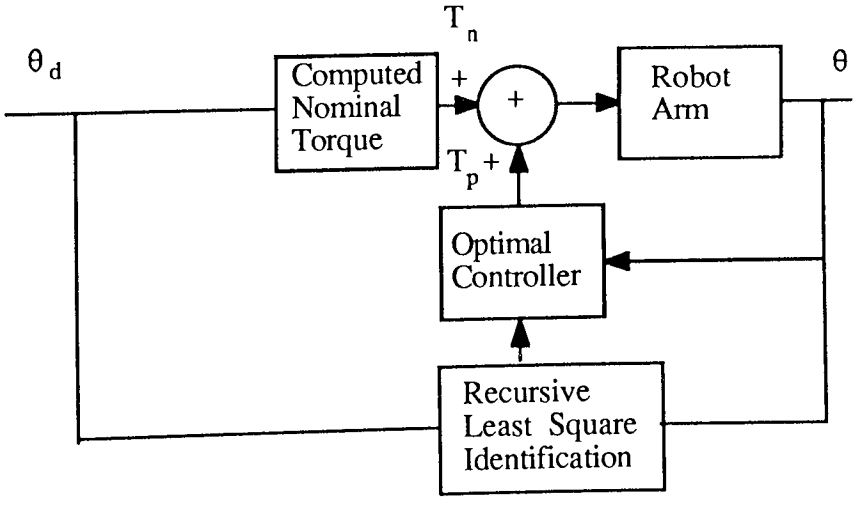


Figure 4.8. Showing the structure of Adaptive Perturbation Control.

The method was tested in simulation and compared to an equivalent PD controller

and was found to exhibit better tracking ability under various degrees of modelled inertia uncertainty.

4.5 Concluding remarks

The control of a manipulator has seen to be particularly difficult due to its nonlinear dynamics. The solutions that have been postulated by previous researchers try to achieve suitable control of the manipulator by compensating for the nonlinear dynamics of the manipulator. It is apparent from the literature that many of the control methods have not been tested at high speeds of tracking and it is questionable whether they would retain stability at high speeds. The question of how the nonlinear dynamic coupling terms affect the dynamic tracking ability has not been satisfactorily proven to the author's knowledge. The question of how relevant the dynamic coupling terms are particularly those due to the Coriolis effect and Centrifugal effects has not been satisfactorily established. In order to clearly answer these questions the author proposes in this research the following:

1.To set up realistic trajectory tracking situation such as a manipulator required to track a part on a conveyor.
2.As an example, a Scara type manipulator will be used in the tracking problem, this exhibits the characteristic dynamic coupling effects in its two horizontal jointed links.
3.To establish the significance of the coupling terms by computing the torques required for the manipulator to follow the trajectory.
4.To establish the relevance the coupling terms when a number of typical independent axis control schemes are applied to the manipulator, which are the basis for the computed torque technique.

Chapter 5:Off-line Path and Trajectory Trajectory Planning

Chapter 5 : Off-line Path and Trajectory Planning

5.1 Introduction

In this chapter, as an example, a study is made of a two degree of freedom robot manipulator that is required to carry out an operation on a part on a moving conveyor(as described in Groover[26]). The path is planned as a straight line between points. In planning the trajectory a linear function with parabolic blends via points is used. The trajectory is planned in Cartesian coordinates, these points are then converted into joint angles using an inverse kinematics algorithm. The Cartesian velocities and accelerations are converted into joint space equivalents using the inverse Jacobian and its derivative.

The dynamic equations of the manipulator are used together with the planned trajectory to calculate the expected load torques. The load torques are then referred back to the motor in order to calculate the expected motor torques. The motor characteristics are then used to assess whether the motor can achieve this particular path.

The significance of the terms in the load torque dynamic equations are examined. Each load torque term is expressed as a percentage of the total torque and the maximum percentage that each term takes is used as a measure of its significance for a particular path and trajectory.

5.2 Path and Trajectory Planning

In planning the path and trajectory the user is usually required to specify the start and finishing positions in Cartesian coordinates. The robot manipulator may need to pass through a number of via points in order to avoid collisions, these will be specified as well. The time duration between points will also be specified. From this information the path and trajectory planning algorithm will work out the complicated details of the trajectory.

The path and trajectory can be planned in either Cartesian or joint space. In planning the path in joint space, each path point(or corner point) is specified in Cartesian coordinates. Each corner point is then converted into a set of desired joint angles using the inverse kinematics algorithm. A smooth function is then

chosen which will pass through these desired joint positions. The time between points is the same for each joint so that each joint reaches the desired joint position at the same time so the correct Cartesian position is obtained at each corner point. The shape of the spatial path taken by the end effector is not a straight line through space but a complicated shape depending on the kinematics of the manipulator.

In planning the path in Cartesian space the spatial shape of the path that the end effector takes is chosen. The most common path specification is a straight line path. This type of path specification is called Cartesian straight line motion. Having planned the path and trajectory in Cartesian coordinates it is necessary to convert these points into equivalent joint angles, velocities and accelerations using the inverse kinematics algorithm. However, with Cartesian planning there are a number of problems relating to workspace and singularities. It is possible to specify a start and end position which is well within the manipulator's workspace, but the intermediate positions may not all lie within the workspace. Also there are locations in the manipulator's workspace where it is impossible to choose finite joint rates that do not yield the desired velocity of the end effector. This occurs at the boundary of the robot manipulator's work space. The planned path and trajectory must thus lie within the manipulator's work space.

The choice of the method of planning depends upon the application. There are a number of problems associated with Cartesian schemes so it is beneficial to use joint space schemes as a default. A typical application where a joint space scheme would be used is in a pick and place operation. Here the path is specified as a number of corner points which the end effector is required to pass through. The spatial path of the end effector between points is unimportant, provided it avoids collisions, and is not usually specified. However, in the example that will be considered later in the chapter , of a manipulator required to track a part on a conveyor, it is necessary for the manipulator's end effector to travel along a particular spatial path. It is evident that a Cartesian scheme is required in this particular application.

To achieve smooth motion between points it is necessary to choose a smoothing function that gives, as a bare minimum, continuous position and velocity. It may also be necessary to have continuous acceleration and possibly derivative of acceleration or 'Jerk'. This can be achieved in a number of ways either in joint space or Cartesian space. The smoothing function chosen is applicable to both

joint and Cartesian schemes. One method is to fit a polynomial that will satisfy the end constraints of each pair of corner points.

5.2.1 Polynomials

The order of the polynomial will depend upon the number of end constraints that are specified. For example with a cubic polynomial four constraints may be specified:

Initial position, x_0
Final position, x_f
Initial velocity, $\dot{x}_0$
Final velocity, $\dot{x}_f$

Where the cubic polynomial is of the form:

$$x(t)= a_0 + a_1 t + a_2 t^2 + a_3 t^3 \qquad (5.1)$$

$$\dot{x}(t)= a_1 + 2 a_2 t + 3 a_3 t^2 \qquad (5.2)$$

$$\ddot{x}(t)= 2 a_2 t + 6 a_3 t^2 \qquad (5.3)$$

The coefficients of the polynomial may be solved in terms of the end constraints:

$$a_0 = x_0$$

$$a_1 = \dot{x}_0$$

$$a_2 = \frac{3}{t_f^2}(x_f - x_0) - \frac{2}{t_f}\dot{x}_0 - \frac{1}{t_f}\dot{x}_f$$

$$a_3 = \frac{2}{t_f^3}(x_f - x_0) + \frac{1}{t_f^2}(\dot{x}_f + \dot{x}_0)$$

Hence, with a cubic polynomial it is possible to specify the initial and final position and velocities together with the total time of the motion t_f. The plots for a cubic polynomial where the velocity end constraints are zero are shown in Fig.5.1.

Higher order polynomials are sometimes used when it is necessary to specify a greater number of end constraints. For example, a quintic polynomial is required in order to be able to specify position, velocity and acceleration. If the quintic

polynomial is represented by the following equation:

$$x(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5 \qquad -(5.4)$$

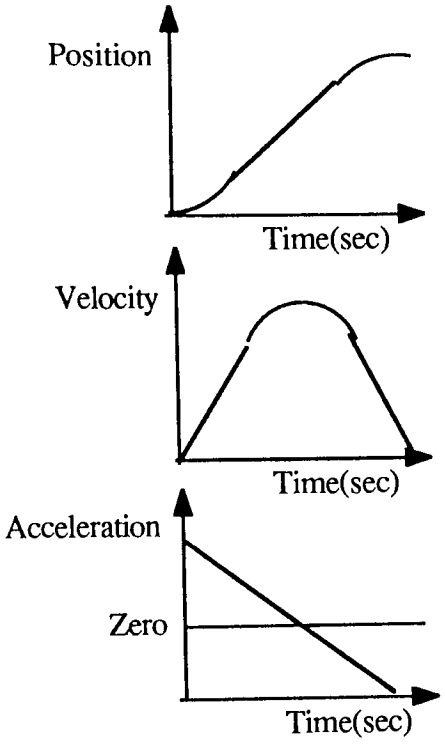


Figure 5.1. Diagram showing the position, velocity and acceleration plots for the planned trajectory using a 3rd order polynomial with four end constraints.

The polynomial coefficients as a function of the end constraints are then found to be:

$$a_0 = x_0$$

$$a_1 = \dot{x}_0$$

$$a_2 = \frac{\ddot{x}_0}{2}$$

$$a_3 = \frac{20\,x_f - 20\,x_0 - (8\,\dot{x}_f + 12\,\dot{x}_0)\,t_f - (3\,\ddot{x}_0 - \ddot{x}_f)\,t_f^2}{2\,t_f^3}$$

$$a_4 = \frac{30\,x_0 - 30\,x_f + (14\,\dot{x}_f + 16\,\dot{x}_0)\,t_f - (3\,\ddot{x}_0 - 2\,\ddot{x}_f)\,t_f^2}{2\,t_f^4}$$

$$a_5 = \frac{12\,x_f - 12\,x_0 + (6\,\dot{x}_f + 6\,\dot{x}_0)\,t_f - (\ddot{x}_0 - \ddot{x}_f)\,t_f^2}{2\,t_f^5}$$

The order of the required polynomial is related to the number of end constraints that are to be specified:

Order of Polynomial =Number of end constraints -1

The polynomial can be used to compute a smooth trajectory through points for both joint space and Cartesian schemes. However, it may be necessary to specify the spatial path between points. For instance, in the example that is to be considered the manipulator's end effector is required to follow the path of the part on the conveyor in a straight line. Thus, it is necessary to not only compute a smooth trajectory between points but to compute this trajectory so the spatial path is along a straight. The method that will be adopted for this purpose is referred to as Cartesian Straight-line motion. The particular application that will serve as an example throughout this text will now be considered.

5.2.2 Example Application

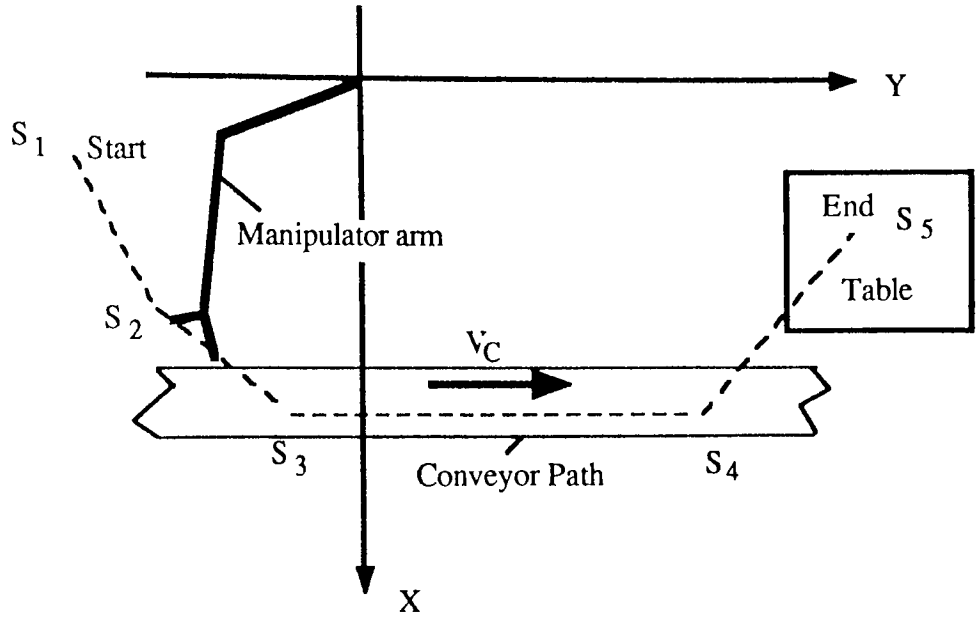


Fig.5.2. Diagram showing the example application of a manipulator required to track a part on a conveyor.

In the example application shown in Fig.5.2 the robot manipulator is required to carry out an operation on a part on a moving conveyor. The conveyor belt moves with a velocity of V_c mm/s . The spatial path of the manipulator needs to be along the length of the conveyor in a straight line at the same velocity as the part it is tracking which travels at the same speed as the conveyor. In order to achieve this motion the path must be planned in Cartesian space. A method to achieve this path will now be considered.

5.2.3 Cartesian Straight line Motion

In Cartesian straight line motion the spatial path is a straight line between points. To create a smooth trajectory with continuous position and velocity a parabolic blend period is computed in the region of each corner point. During the blend region a constant acceleration is applied to change the velocity smoothly. The linear portion of the path is computed using straight line interpolation and this is at constant velocity.

The solution for a multi-segment linear path with blends will now be considered.

The multi-segment path is shown in Fig.5.3. Here the corner points j,k and l are neighbouring path points. The time period of the segment jk is td_{jk}. The linear portion of segment jk occupies a time of t_{jk}. The velocity of the linear section of segment jk is X_{jk}' . The acceleration of the blend at j is X_j''. The time duration for this blend region is t_j.

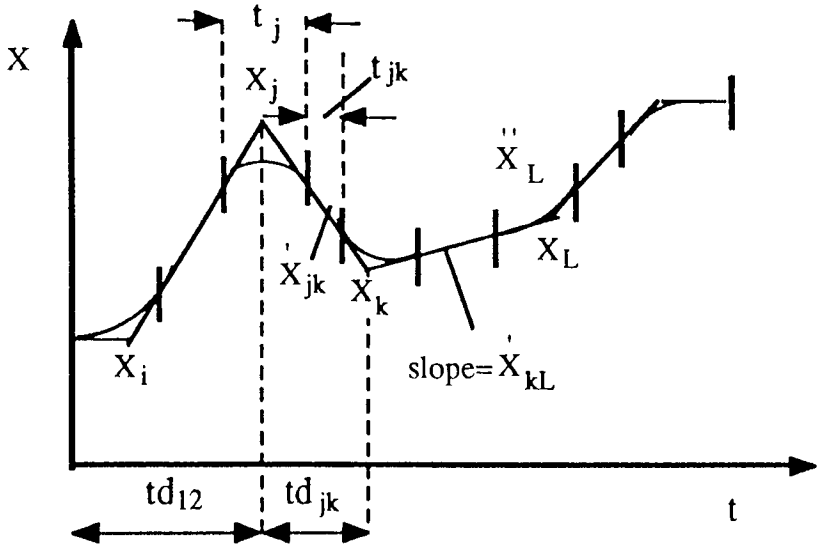


Figure 5.3. Diagram showing a multi-segment path with blend regions at the corner points.

For this particular application the problem is restricted to a single plane i.e. the X-Y plane. The X and Y coordinates are then converted into corresponding joint angles. For a multi-degree of freedom system the blend times for each degree of freedom must be the same. This ensures that the resultant motion of all the degrees of freedom will be a straight line in space during the linear region. The blend regions for each degree of freedom are the same, however,the acceleration used during the blend for each degree of freedom will be different.
For interior path points the equations for the velocities and times are as follows:

$$\dot{X}_{jk} = \frac{X_k - X_j}{td_{jk}}$$

where k and j are the specified corner points

td_{jk} the specified time for the duration of the segment

$$\ddot{X}_k = \frac{\dot{X}_{kl} - \dot{X}_{jk}}{t_k}$$

where t_k is the specified blend time at corner point k

$$t_{jk} = td_{jk} - \frac{1}{2} t_j - \frac{1}{2} t_k$$

The first and last segments are handled slightly differently since an entire blend region at one end of the segment is included in the total segment's time duration. For the first segment the equations are as follows:

$$\dot{X}_{12} = \frac{X_2 - X_1}{td_{12} - \frac{1}{2} t_1}$$

$$\ddot{X}_1 = \frac{\dot{X}_{12}}{t_1}$$

$$t_{12} = td_{12} - t_1 - \frac{1}{2} t_2$$

Similarly, for the last segment connecting corner points n-1 and n the equations are:

$$\dot{X}_{(n-1)n} = \frac{X_n - X_{n-1}}{td_{(n-1)\,n} - \frac{1}{2} t_n}$$

$$\ddot{X}_n = - \frac{\dot{X}_{(n-1)n}}{t_n}$$

$$t_{(n-1)\,n} = td_{(n-1)\,n} - t_n - \frac{1}{2} t_{n-1}$$

For the specific robot manipulator that will be used a path has been planned that is well within the work space of the manipulator. The path corner points and durations are listed below.The initial blend times used are also listed in this table 5.1.

Corner Point	P_X (mm)	P_Y (mm)	Blend Time t_i (Sec)	Duration td_{jk} (sec)
S1	80	-480	0.2	0.8
S2	280	-400	0.15	1.0
S3	400	-200	0.1	1.5
S4	400	160	0.15	0.7
S5	280	400	0.2	

Table 5.1. Showing planned path corner point positions and path times.

In the main program TRAJPLAN, the procedure INPUTPATH allows the path information listed in table 5.1 to be input into the program. In the robot system there are a number of constraints which include the maximum Cartesian velocity in each degree of freedom and the maximum acceleration in each degree of freedom. First the setting of the maximum velocity will be considered.

5.2.4 Maximum Velocity

As shown previously, for interior points the velocity is determined from the following equation:

$$\dot{X}_{jk} = \frac{X_k - X_j}{td_{jk}}$$

The velocity can be adjusted by changing the duration for the segment. For example, in the conveyor tracking problem the velocity of the conveyor is V_c. The positions S_3 and S_4 are known. In order for the manipulator to track the conveyor at a velocity V_c the time duration for segment 3 must be specified as $(S_4-S_3)/V_c$. Infact all time durations could be calculated in this way based upon the required resultant Cartesian velocity(see later).

5.2.5 Maximum Acceleration

The acceleration for interior points is determined from the equation:

$$\ddot{X}_k = \frac{\dot{X}_{kl} - \dot{X}_{jk}}{t_k}$$

The acceleration can be adjusted by changing the blend times. However, since the blend time must be the same for each degree of freedom it will give a different acceleration for each one. The blend time can be chosen so that the maximum acceleration in either X or Y is not exceeded.

The procedure INPUTPATH calculates the path velocities and accelerations based upon the specified corner points and time durations. These times can be iteratively adjusted to fulfill a required upper bound for the velocity and acceleration. The results for the initial unput times of Table 5.1 are shown in Table 5.2.

Corner Point S_i	XV_{jk} (i) (mm/sec)	XA(i) (mm/sec^2)	YV_{jk}(i) (mm/sec)	YA(i) (mm/sec^2)	TXY_{jk} (i) (secs)	TXY(i) (secs)
S1	285.7	1428.5	114.3	571.4	0.525	0.20
S2	120.0	-1104.7	200.0	571.4	0.875	0.15
S3	0.0	-1200.0	240.0	400.0	1.375	0.10
S4	200.0	-1333.3	400.0	1066.7	0.425	0.15
S5		1000.0		-2000.0		0.20

Table 5.2. Showing the planned path segment velocities and accelerations.

Note. For table 5.2 and subsequent tables the following notation is used for the velocities,accelerations and path times in reference to Fig.5.3:

$XV_{jk}(i) = \dot{X}_{jk}$(in Fig.5.3) the velocity in the X axis between corner points j and k

$YV_{jk}(i) = \dot{X}_{jk}$(in Fig.5.3) the velocity in the Y axis between corner points j and k

$XA(i) = \ddot{X}_i$(in Fig.5.3) the acceleration in the X axis at corner points i

$YA(i) = \ddot{X}_i$(in Fig.5.3) the acceleration in the Y axis at corner points i

$TXY_{jk}(i)=t_{jk}$ the time of the path at constant velocity between points j and k

$TXY(i)=t_i$ the time of the blend period at corner point i.

For the path values of Table 5.1, the maximum acceleration is seen to be:

$|YA(5)| = 2000$ mm/sec^2 in the Y axis.

If an artificial constraint of say $|YA|=1500$ mm/sec^2 is required then the blend time for segment 4 must be increased. In the results of table 5.3 the value of the blend time TXY(5) has been increased to 0.3 sec. This results in an acceleration for $|YA(5)|=1454$ mm/sec^2 which is within the required constraint of 1500 mm/sec^2. However, changing the blend time does not only affect the value of XA(5) and YA(5) but also the value of Cartesian velocities in that segment and the accelerations XA(4) and YA(4). The equation for computing the velocity of the final segment is:

$$\dot{X}_{(n-1)n} = \frac{X_n - X_{n-1}}{td_{(n-1)\,n} - \frac{1}{2}t_n}$$

If the value of t_n is increased it can be seen that the velocity will correspondingly increase. The equation for computing the acceleration at the final corner point is as follows:

$$\ddot{X}_n = -\frac{\dot{X}_{(n-1)n}}{t_n}$$

It can be deduced from the above equation that if the value of the final segment velocity is increased then the value of the end corner point acceleration will increase. The results for the change of blend period of TXY(5) from 0.2 to 0.3 sec is show in Table 5.3 below:

Corner Point S_i	XV_{jk} (i) (mm/sec)	XA(i) (mm/sec^2)	YV_{jk} (i) (mm/sec)	YA(i) (mm/sec^2)	TXY_{jk}(i) (secs)	TXY(i) (secs)
S1	285.7	1428.5	114.3	571.4	0.525	0.20
S2	120.0	-1104.7	200.0	571.4	0.875	0.15
S3	0.0	-1200.0	240.0	400.0	1.375	0.10
S4	-218.2	-1454.6	436.4	1309.1	0.325	0.15
S5		727.2		-1454.6		0.30

Table.5.3. Showing the new velocities and accelerations for a change in the blend period of TXY(5) from 0.2 to 0.3 secs.

It has been seen how the blend times may be adjusted in order that the accelerations are within an upper limit (later in the chapter it will be seen how this is done automatically by computer). In a like manner the segment durations may be adjusted so that all segment velocities are within an upper velocity limit. For instance, if the required velocity over segment 3 is 300mm/sec then the time duration for that segment must be changed:

$$\frac{S_4 - S_3}{V_c} = \frac{160 - (-200)}{300} = 1.2 \text{ sec}$$

Obviously, the velocity change will affect the acceleration at points either end of the segment namely XV_{jk} (3)+(4) and YV_{jk} (3) + (4) since these are calculated from the following equation:

$$\ddot{X}_k = \frac{\dot{X}_{kl} - \dot{X}_{jk}}{t_k}$$

The blend times t_k can be adjusted to compensate if the recalculated accelerations exceed the the upper limit. The results for the new time period td_{34}=1.2 sec are shown in Table 5.4 below:

Corner Point S_i	XV_{jk} (i) (mm/sec)	XA(i) (mm/sec^2)	YV_{jk} (i) (mm/sec)	YA(i) (mm/sec^2)	TXY_{jk}(i) (secs)	TXY(i) (secs)
S1	285.7	1428.5	114.3	571.4	0.525	0.20
S2	120.0	-1104.7	200.0	571.4	0.875	0.15
S3	0.0	-1200.0	300.0	1000.0	1.075	0.10
S4	-218.2	-1454.6	436.4	909.1	0.325	0.15
S5		727.2		-1454.6		0.30

Table 5.4. Showing the new velocities and accelerations for a change in the time period td_{34}=1.2 secs.

As shown in Table 5.4, the accelerations remain within the upper limit of 1500 mm,sec^2. It is not necessary to adjust the blend times. It has been seen how individual segment times and blend times can be adjusted to fulfill the velocity and acceleration limits for a particular segment. An alternative approach is to calculate the segment times and blend times for the complete path based upon a default maximum achievable resultant velocity and acceleration. This approach will now be considered.

5.2.6 Calculation of Path times based upon default resultant velocity and acceleration

To calculate the segment blend and segment times a default resultant Cartesian velocity and acceleration is specified. The equations previously used are recompiled in terms of the resultant:

For corner points 1 and 2

$$td_{12} - \frac{1}{2}t_1 = \frac{\sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2}}{V_R}$$

Where V_R = Resultant Velocity

The velocity in the first segment is calculated for the X and Y directions:

$$\dot{X}_{12} = \frac{X_2 - X_1}{td_{12} - \frac{1}{2}t_1} \quad \text{and} \quad \dot{Y}_{12} = \frac{Y_2 - Y_1}{td_{12} - \frac{1}{2}t_1}$$

The time period for the first blend period is calculated from the resultant acceleration A_R:

$$t_1 = \frac{\sqrt{\dot{X}_{12}^{\,2} + \dot{Y}_{12}^{\,2}}}{A_R}$$ where A_R is the resultant Acceleration

The acceleration in the first segment is calculated for the X and Y directions:

$$\ddot{X}_1 = \frac{\dot{X}_{12}}{t_1} \quad \text{and} \quad \ddot{Y}_1 = \frac{\dot{Y}_{12}}{t_1}$$

For the final segment, between corner points n-1 and n:

$$td_{(n-1)\,n} - \frac{1}{2}t_n = \frac{\sqrt{(X_n - X_{n-1})^2 + (Y_n - Y_{n-1})^2}}{V_R}$$

The velocity in the X and Y directions is calculated as follows:

$$\dot{X}_{(n-1)n} = \frac{X_n - X_{n-1}}{td_{(n-1)\,n} - \frac{1}{2}t_n} \quad \text{and} \quad \dot{Y}_{(n-1)n} = \frac{Y_n - Y_{n-1}}{td_{(n-1)\,n} - \frac{1}{2}t_n}$$

The time period for the final blend period is calculated from the resultant acceleration :

$$t_n = \frac{\sqrt{\dot{X}_{(n-1)n}^{\,2} + \dot{Y}_{(n-1)n}^{\,2}}}{A_R}$$

The acceleration in the final segment is calculated in the X and Y direction:

$$\ddot{X}_n = -\frac{\dot{X}_{(n-1)n}}{t_n} \quad \text{and} \quad \ddot{Y}_n = -\frac{\dot{Y}_{(n-1)n}}{t_n}$$

For the interior points j and k :

$$td_{jk} = \frac{\sqrt{(X_k - X_j)^2 + (Y_k - Y_j)^2}}{V_R}$$

The velocity in the X and Y direction are calculated:

$$\dot{X}_{jk} = \frac{X_k - X_j}{td_{jk}} \text{ and } \dot{Y}_{jk} = \frac{Y_k - Y_j}{td_{jk}}$$

The time period for the blend at corner point j is calculated:

$$t_j = \frac{\sqrt{(\dot{X}_{jk} - \dot{X}_{ij})^2 + (\dot{Y}_{jk} - \dot{Y}_{ij})^2}}{A_R}$$

The acceleration in the X and Y direction is calculated as follows:

$$\ddot{X}_j = \frac{\dot{X}_{jk}}{t_j} \text{ and } \ddot{Y}_j = \frac{\dot{Y}_{jk}}{t_j}$$

As an example of using this technique the conveyor tracking problem will be considered again. The conveyor velocity between points S_3 and S_4 is 300 mm/sec so the manipulator is required to move at this velocity. Since the conveyor is parallel to the Y axis there is no component of velocity in the X direction. The resultant velocity specified is hence the velocity in the Y direction. Previously a limit of 2000 mm/sec^2 was imposed on the acceleration. This will be used in this example and will be specified as the resultant acceleration to ensure that all Cartesian accelerations throughout the path are below this value. The results for this example are shown in Table 5.5. As can be seen in this table all of the velocities are below the specified resultant velocity and during the segment 3 the required conveyor velocity of 300 mm/sec is obtained.

Corner Point Si	XV_{jk} (i) (mm/sec)	XA(i) (mm/sec^2)	YV_{jk}(i) (mm/sec)	YA(i) (mm/sec^2)	TXY_{jk}(i) (secs)	TXY(i) (secs)
S1	278.5	1856.9	111.4	742.8	0.5951	0.1500
S2	154.4	-1296.7	257.3	1522.7	0.6895	0.0958
S3	0.0	-1927.4	300.0	533.9	1.1255	0.0801
S4	-134.2	-1946.5	268.3	-459.5	0.7850	0.0689
S5		894.4		-1788.9		0.1500

Table 5.5. Showing the velocities, accelerations and path times based upon the specified resultant values.

5.2.7 Scaling of path

When the joint velocities and accelerations are exceeded in a path it is possible to scale down the path Cartesian velocities and accelerations to meet the joint capabilities. Alternatively the planned trajectory may not be making full use of the motor capabilties, in that case it is necessary to scale up the path velocities and accelerations. For instance in the previous path a velocity to acceleration ratio of 300/2000 was specified. If this path is to be scaled up by 5 and retaining the same ratio gives 1500/10000. These values are thus specified as the default resultant Cartesian values. The results for this specification is shown in Table 5.6.

Corner Point S_i	XV_{jk} (i) (mm/sec)	XA(i) (mm/sec^2)	YV_{jk} (i) (mm/sec)	YA(i) (mm/sec^2)	TXY_{jk}(i) (secs)	TXY(i) (secs)
S1	1392.7	9284.8	557.1	3713.4	0.0207	0.1500
S2	771.7	-6483.7	1286.2	7613.3	0.0676	0.0958
S3	0.0	-9637.3	1500.0	2669.3	0.1655	0.0801
S4	-670.8	-9732.5	1341.6	-2297.5	0.0694	0.0689
S5		4472.1		-8944.3		0.1500

Table 5.6. Showing the velocities, accelerations and path times for a specification of 1500 mm/sec resultant velocity and 10000 mm/sec^2 resultant acceleration.

As seen in Table 5.6 the Cartesian velocities and accelerations are all multiplied by the scaling factor of 5. However, the time period for which the path is linear is decreased considerably as shown by the column TXY_{jk} values. This means that the spatial path of the manipulator will not be following a straight line for large portion of the segment. In fact the majority of the segment is taken by the blend period. It is important that the proportion of the path that is linear is the same, this will now be considered.

5.2.8 Scaling of path to retain the same linear proportion

The ratio of TXY_{jk}/Td_{jk} expresses the proportion of each segment that the path is linear. This is calculated for path of table 5.5(refered to as p300/2000) and the path of table 5.6(p1500/10000) for each segment and expressed as a percentage in table 5.7.

Segment i	P300/2000 TXY_{jk} %(i)	Segment i	P1500/10000 TXY_{jk} %(i)
1	75.1	1	9.5
2	88.7	2	43.5
3	93.8	3	68.9
4	80.9	4	27.3

Table 5.7. Showing the percentages for each segment that the path is at a constant velocity for paths p300/2000 and p1500/10000.

In order to obtain the same ratio it is necessary to go back to the equations and find out what the relationship is beween the scaling factor and the linear ratio.

For interior points:

The corner point positions are fixed so expressions involving these terms are fixed:

Let $\sqrt{(X_k - X_j)^2 + (Y_k - Y_j)^2} = K$ a constant

Hence expressions for the segment times are:

$$td_{jk} = \frac{K_1}{V_R} \quad \text{and} \quad td_{ij} = \frac{K_2}{V_R}$$

The blend time at corner point j is:

$$t_j = \frac{\sqrt{\left(\frac{(X_k - X_j)}{td_{jk}} - \frac{(X_j - X_i)}{td_{ij}}\right)^2 + \left(\frac{(Y_k - Y_j)}{td_{jk}} - \frac{(Y_j - Y_i)}{td_{ij}}\right)^2}}{A_R}$$

Therefore substituting for td_{jk} and td_{ij} in the above equation:

$$t_j = \frac{V_R}{A_R} K_3 \quad \text{similarly } t_k = \frac{V_R}{A_R} K_3$$

The time period for the linear section of the segment is as follows:

$$t_{jk} = td_{jk} - \frac{1}{2} t_j - \frac{1}{2} t_k$$

The ratio t_{jk}/td_{jk} is hence obtained:

$$\frac{t_{jk}}{td_{jk}} = \frac{\frac{K_1}{V_R} - \frac{V_R K_3}{2 A_R} - \frac{V_R K_4}{2 A_R}}{\frac{K_1}{V_R}}$$

This simplifies to the following expression:

$$\frac{t_{jk}}{td_{jk}} = 1 - \frac{V_R^2}{A_R} K_5 \quad \text{where } K_5 \text{ is a function the the other constants}$$

This means that if the velocity is scaled by a factor of say N, in order to retain the same ratio the acceleration must be multiplyed by a factor of $M=N^2$. This expression can also be shown to be true for the initial and end segments. As an example the path P300/2000 will have the velocity scaled by 5, the acceleration must be scaled by 5^2 in order to retain the same linear ratio. The path specification becomes P1500/50000. The results for this path are shown in Table 5.8.

Corner Point S_i	XV_{jk} (i) (mm/sec)	XA(i) (mm/sec^2)	YV_{jk} (i) (mm/sec)	YA(i) (mm/sec^2)	TXY_{jk}(i) (secs)	TXY(i) (secs)
S1	1392.7	46423.8	557.1	18569.5	0.1190	0.0300
S2	771.7	-32418.5	1286.2	38066.3	0.1379	0.0192
S3	0.0	-48185.7	1500.0	13346.7	0.2251	0.0160
S4	-670.8	-48662.4	1341.6	-11487.7	0.1570	0.0138
S5		22360.7		-44721.4		0.0300

Table 5.8. Showing the calculated results for path p1500/50000.

5.3 Calculation of the Trajectory

The next step in the trajectory planning is to calculate the trajectory points for the X and Y axes at a chosen interval of time e.g Tsamp(the sample period). In the program TRAJPLAN the procedure CALTRAJECTORY calculates the the trajectory points from the path times(This program is implemented in Light Speed Pascal[27] on the Macintosh Computer[28]).

Each segment of the path is referred to by the segment variable, segnum e.g for segment 1, segnum=1. The position total at each end of the linear region is referred to by the variables $X_{t1}(1)$ and $X_{t2}(2)$ respectively. This is shown in Fig.5.4 for a path that has five corner points, hence four segments.

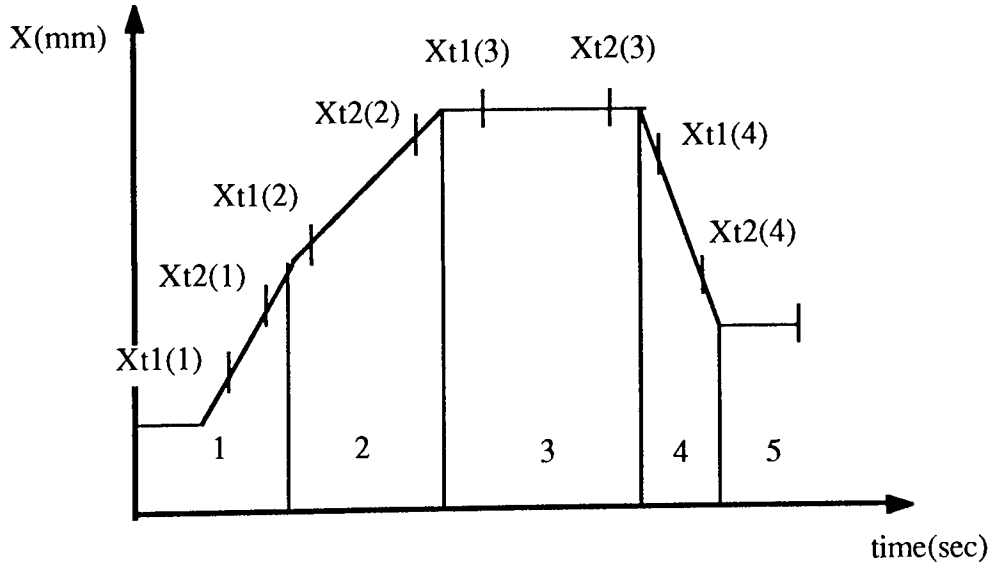


Figure 5.4. Diagram showing how the computer calculates the trajectory.

In the linear section the equation for the displacement is as follows:

$$X_{\text{Linear Region}} = Xt_1(\text{segnum}) + \dot{X}_{(\text{segnum})(\text{segnum}+1)} * (\text{Time} - \text{Tdtotal(segnum)})$$

where Time=the current cumulative time from zero
Tdtotal=is the time at Xt1(segnum)

When the program detects that it is in a blend region i.e when

$\text{Time} > \text{Tdtotal(segnum)} + Txy_{jk}(\text{segnum})$

then the equation for displacement becomes as follows:

$$X_{\text{blend Region}} = Xt_2(\text{segnum}) + \dot{X}_{(\text{segnum})(\text{segnum}+1)} * (\text{Time} - \text{Tdtotal}(\text{segnum}) - Txy_{jk}(\text{segnum}))$$

$$+ \frac{1}{2} Xa(\text{segnum}+1) * (\text{Time} - \text{Tdtotal}(\text{segnum}) - Txy_{jk}(\text{segnum}))^2$$

The velocity in the linear region is constant and equal to the calculated value of Xv_{jk}(segnum). The velocity in a blend region is calculated as follows:

$$\dot{X}_{\text{blend}} = Xv_{jk}(\text{segnum}) + Xv_{jk}(\text{segnum}) (\text{Time} - \text{Tdtotal}(\text{segnum}) - Txy_{jk}(\text{segnum}))$$

The acceleration during the linear section is zero and the acceleration during the blend period is equal to Xa(segnum+1).

The complete path for the planned path P300/2000 is shown in fig.5.5.

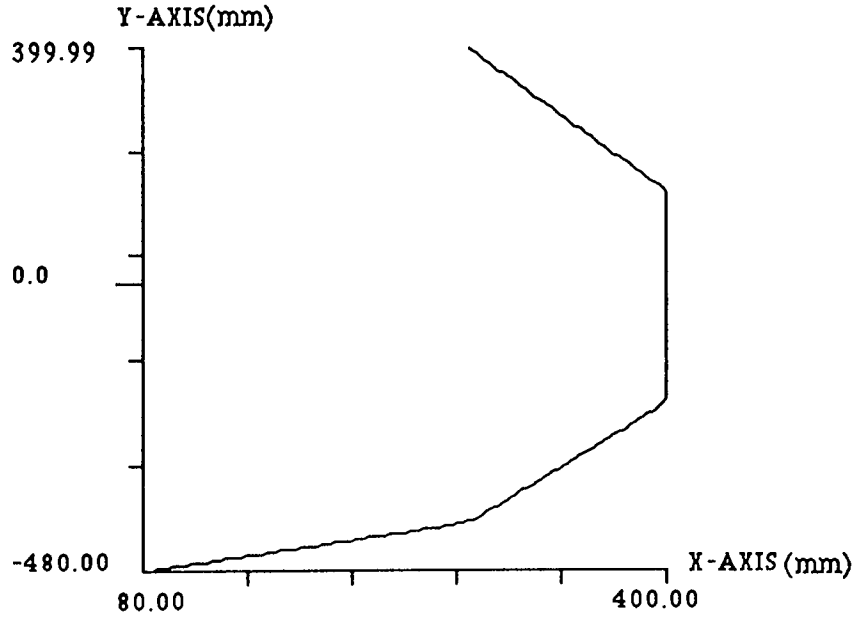


Figure 5.5. Plot of the X axis displacement versus the Y axis displacement for the complete path (p300/2000).

The X axis path plotted against time is shown in Fig.5.6 below:

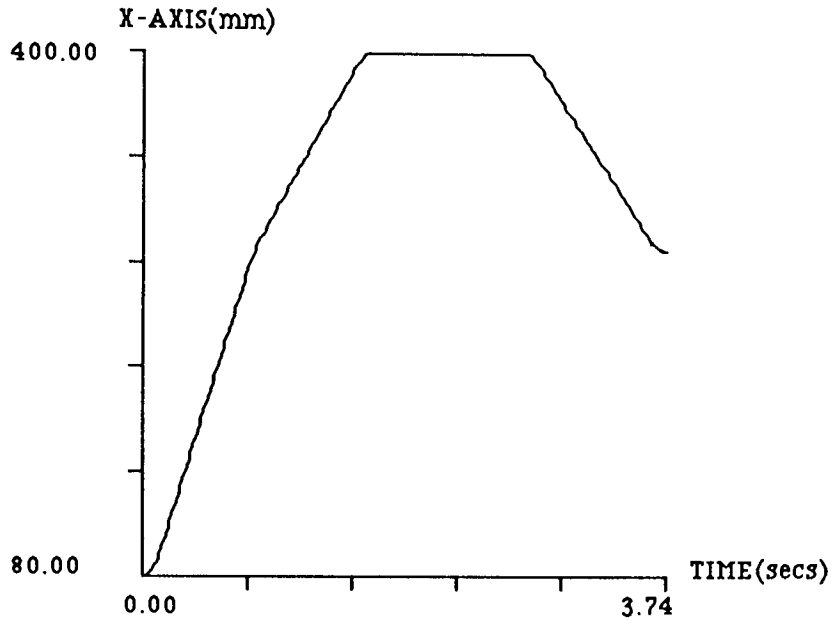


Figure 5.6. Plot of the X axis displacement versus time.

The X axis velocity is shown plotted against time in Fig.5.7 below:

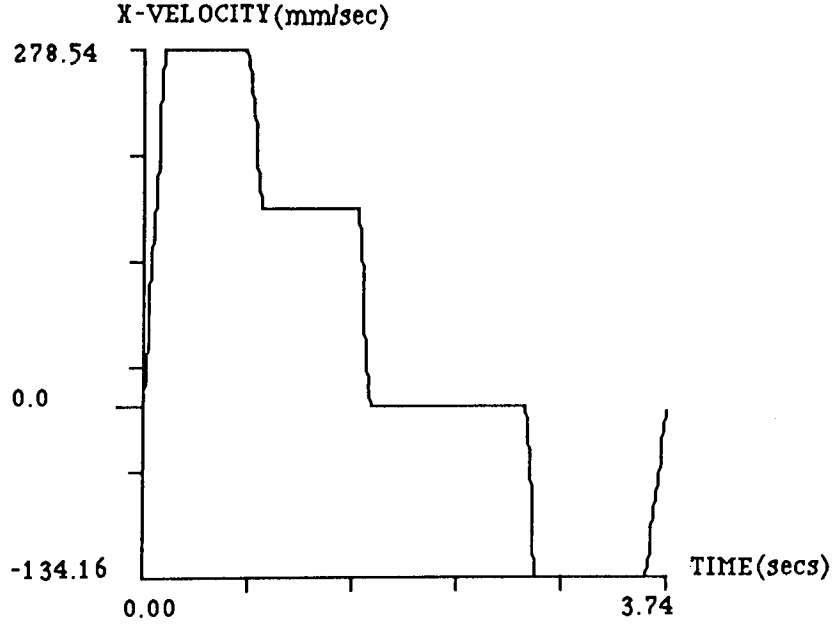


Figure 5.7. Plot of the X axis velocity versus time.

The acceleration in the X axis is shown plotted against time in Fig.5.8. The acceleration is seen to be discontinuous in that there is a step period of acceleration at each corner point.

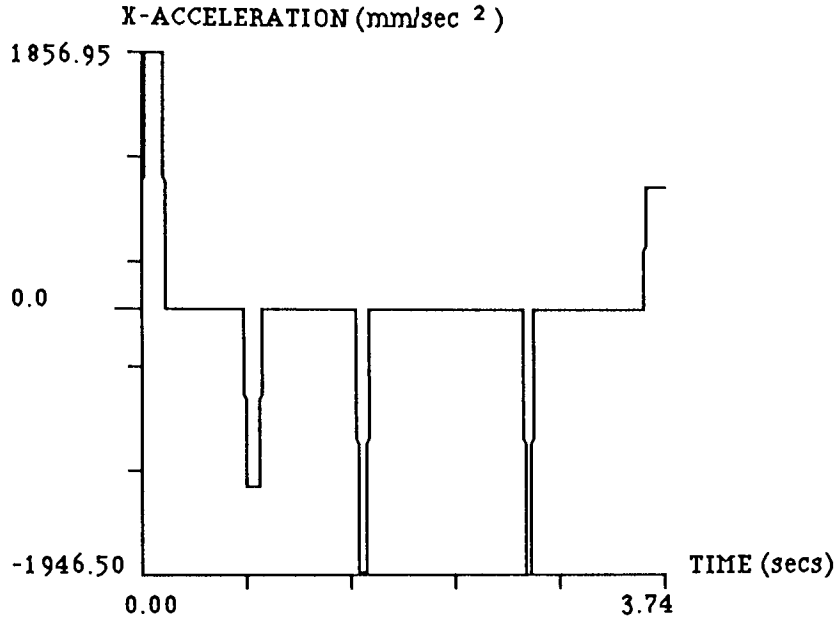


Figure 5.8. Plot of the X axis acceleration versus time.

The next step is to convert the cartesian X-Y trajectory values into equivalent joint space values,this is considered next.

5.3.1 Cartesian Space to Joint Space

The trajectory values signify the time related positions of the motion of the arm in Cartesian coordinates. The positions are then converted into joint space coordinates using the procedure INVERSE. The inverse kinematics procedure solves the kinematics of the arm for a left hand configuration. If the path has been planned correctly the range of values of θ_1 and θ_2 should be within the manipulator's work space. The plot of θ_1 and θ_2 against time are shown in Fig.5.9 and Fig.5.10 respectively.

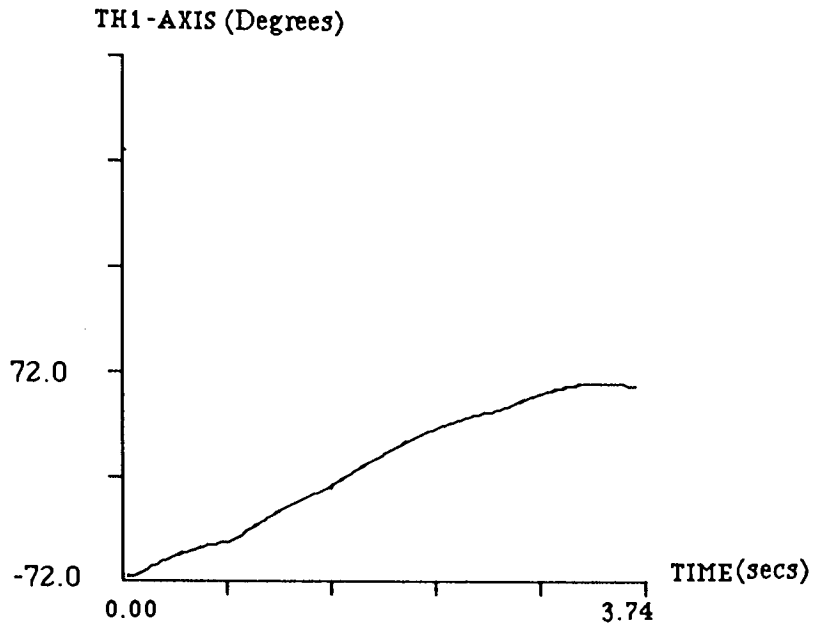


Figure 5.9. Plot of joint angle 1 displacement versus time.

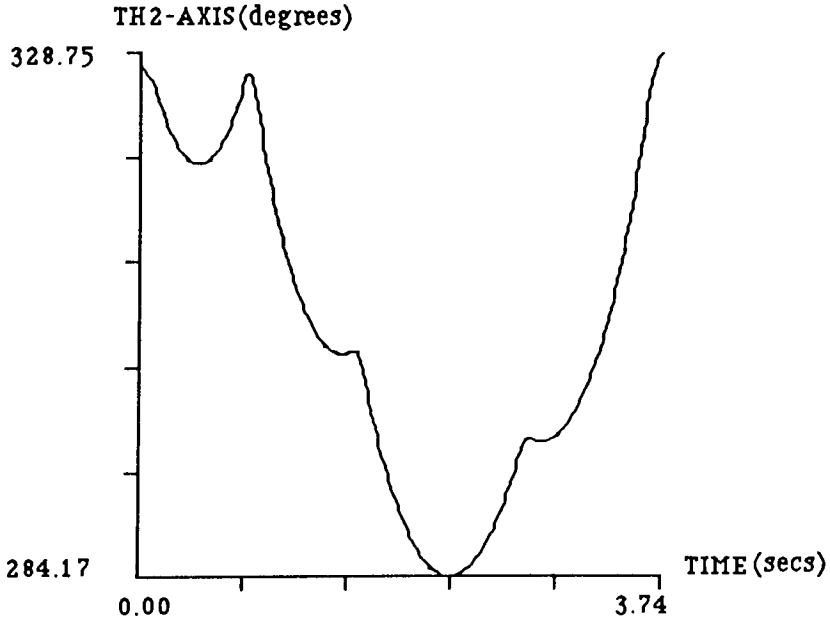


Figure 5.10. Plot of joint angle 2 displacement versus time.

The next step in the trajectory generation is to convert the Cartesian velocities into equivalent joint velocities. This is achieved using the procedure INVERSEJACOBIAN. The angular joint velocities are shown plotted against time in Fig.5.11 and Fig.5.12 respectively.

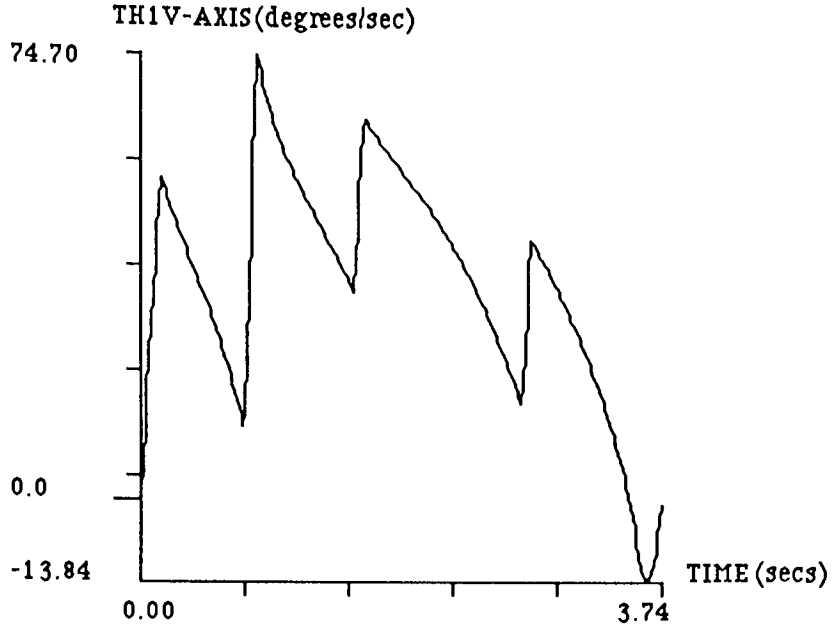


Figure 5.11. Plot of joint 1 angular velocity versus time.

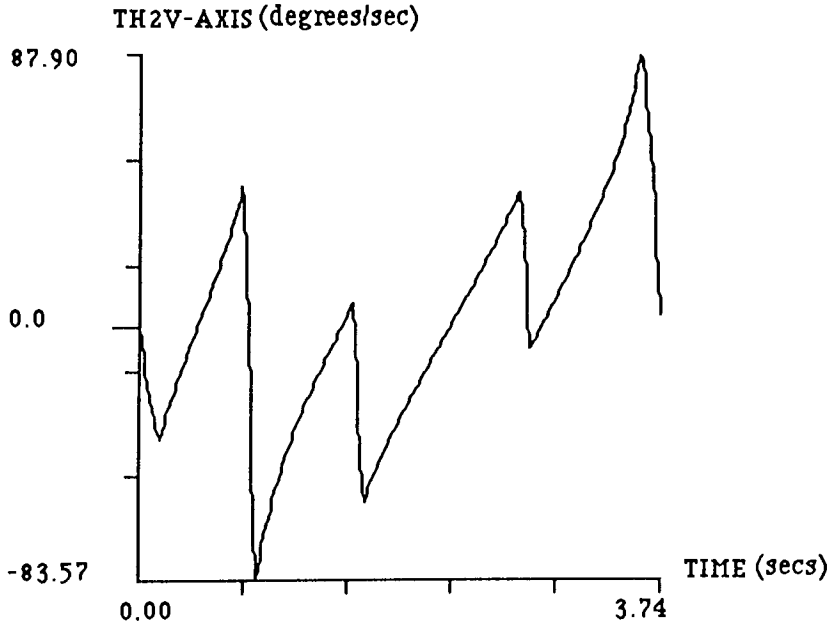


Figure 5.12. Plot of joint 2 angular velocity versus time

As seen in the joint velocity plots, there is an abrupt change of velocity in the region that the corner points occur.

The next step in the trajectory generation is the conversion of the Cartesian accelerations into equivalent joint accelerations. This is carried out by the procedure INVERSEDERIVJACOBIAN. The joint accelerations are shown plotted against time for joint 1 and joint 2 in Fig.5.15 and Fig.5.16 respectively.

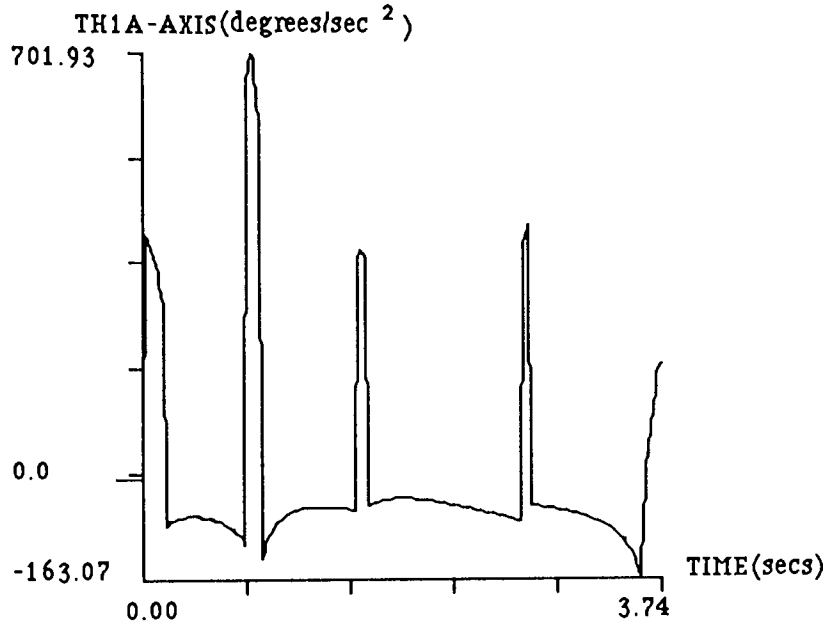


Figure 5.13. Plot of joint 1 angular acceleration versus time.

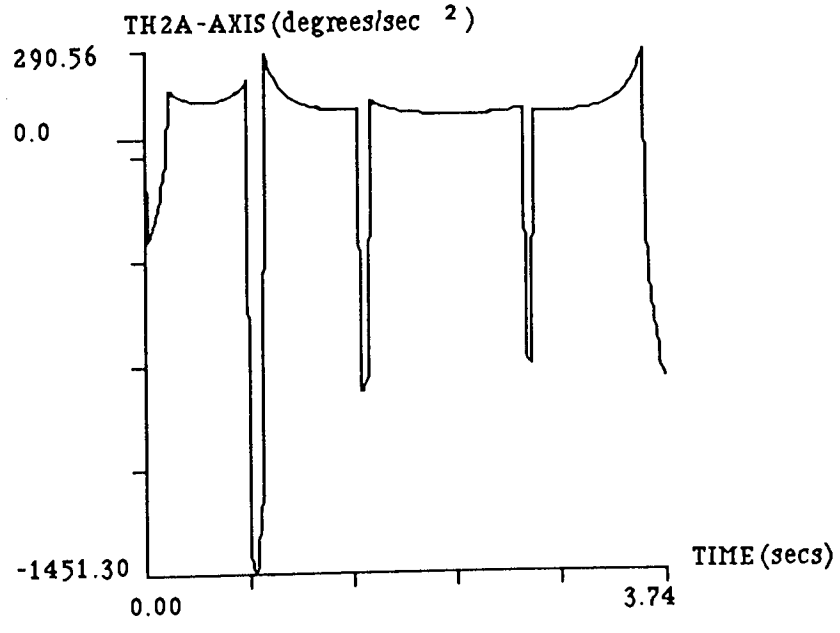


Figure 5.14. Plot of joint 2 angular acceleration versus time

As shown above in the acceleration plots above there is rapid joint acceleration

change that occurs at the corner points.

5.3.2 Calculation of Load and Motor Torques

The trajectory information can now be used to calculate the expected torque on load and motor sides. The trajectory data constitutes the joint angles, velocities and accelerations at each instant of time along the path. This is used to first calculate the non-linear coefficients of the dynamic equations e.g. D_{11},D_{12} etc. These can be graphically plotted in order to observe the degree of variation. For example the coefficient D_{11} is equal to the following:

$$D_{11}=(I_1+m_1a_1^2+I_2+m_2a_2^2+m_2l_1^2+2m_2l_1a_2\cos\theta_2)$$

This is shown plotted in Fig.5.15 for the planned path (p300/2000) below:

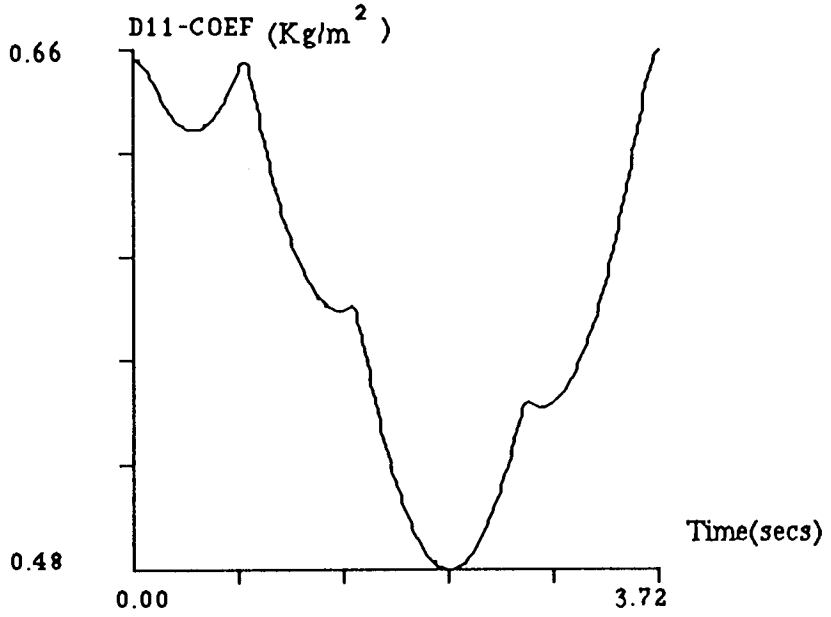


Figure 5.15. Plot of the dynamic coefficient D_{11} versus time for path p300/2000.

The dynamic terms of the load torque equations are then calculated for each joint. The load torque TL_1 and TL_2 are shown plotted in Fig.5.16 and Fig.5.17 respectively.

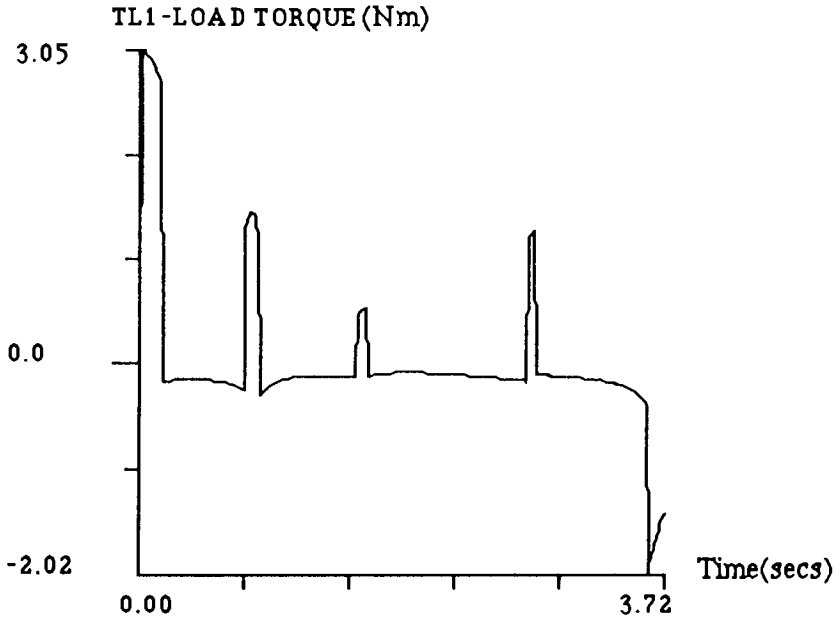


Figure 5.16. Plot of the load torque term TL_1 versus time.

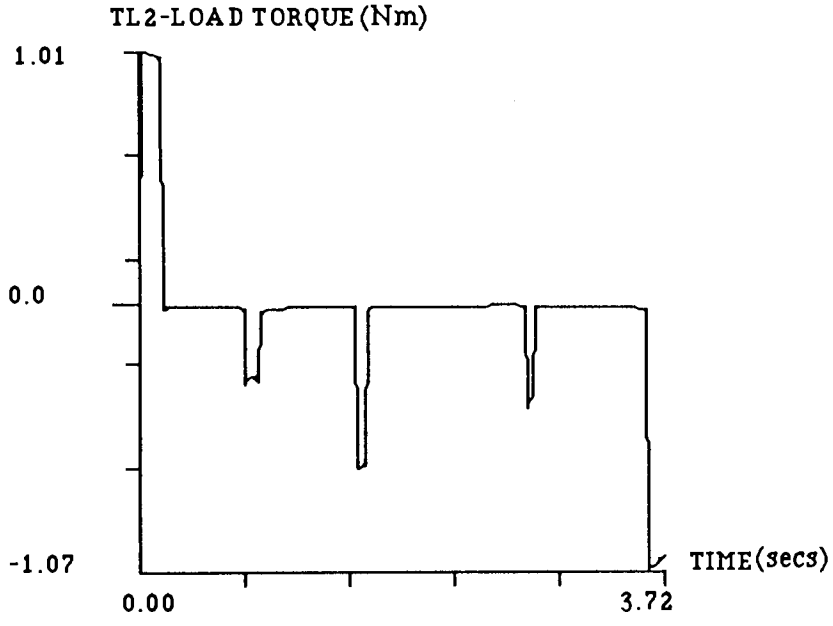


Figure 5.17. Plot of the load torque term TL_2 versus time.

Finally, the motor torque is calculated, i.e. the load torque reflected back to the motor. Each term in the motor torque equation may be plotted separately. The motor torques Tm_1 and Tm_2 are shown plotted in Fig.5.18 and Fig.5.19 respectively.

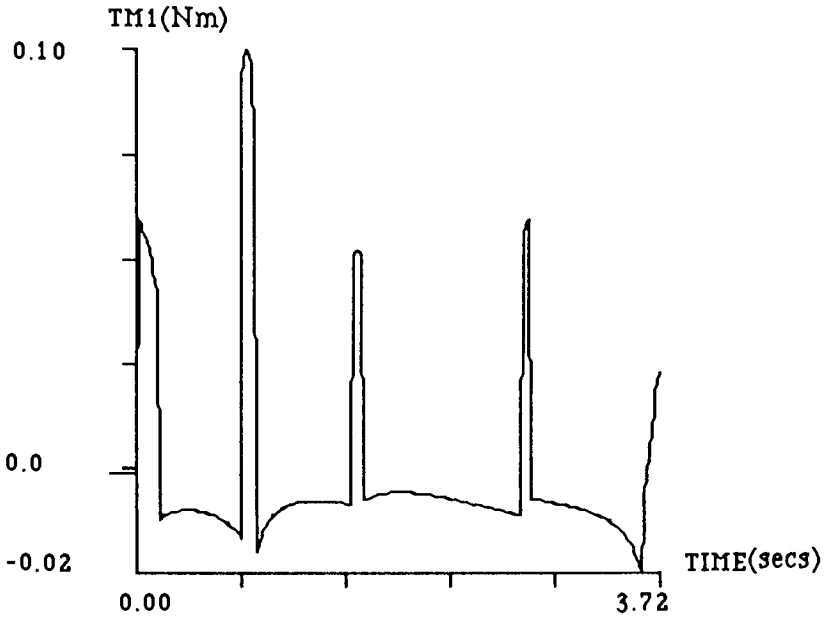


Figure 5.18. Plot of the motor torque term Tm_1 versus time.

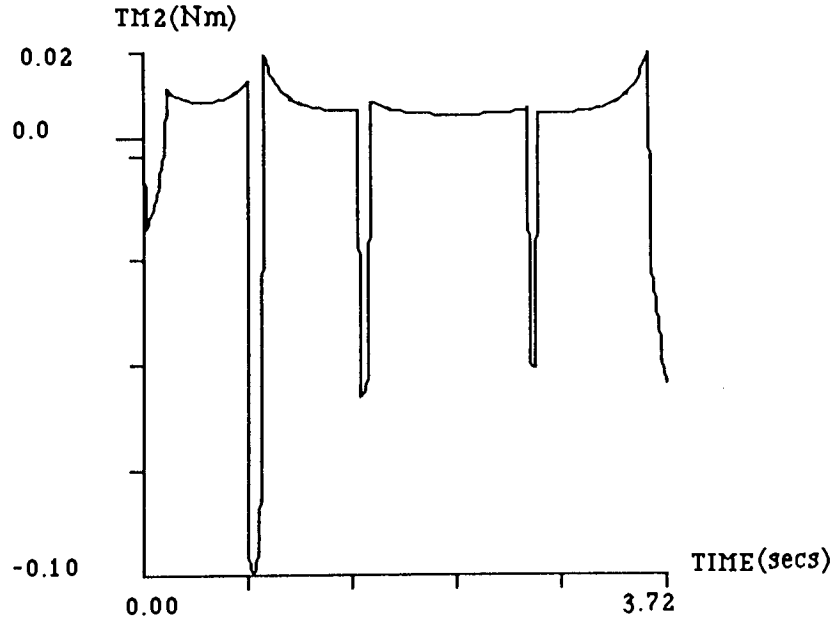


Figure 5.19. Plot of the motor torque term Tm_2 versus time.

So far, a path and trajectory have been planned in Cartesian space, this has then been converted into joint space using the inverse kinematics of the manipulator. From the dynamic equations of the manipulator and the trajectory data the load and motor torques have been calculated. The question arises, how does the expected torque requirements over the planned trajectory relate to the motor characteristics

?. The motor characteristics will now be considered.

5.4 Motor Characteristics

The characteristics of the motor are now considered in order to determine the upper bounds on the joint movements.

5.4.1 Motor Speed

The maximum speed of an armature controlled D.C Motor is a function of the back emf constant, the terminal voltage and the armature resistance. This relationship is shown in the following equation:

$$\text{No Load terminal speed} = \frac{\text{Terminal Voltage - no load current * armature resistance}}{\text{back emf constant}}$$

This limits the maximum joint velocities that can be attained. The path must be planned so that the joint velocities do not exceed the speed attainable by the motor. For the RTX robot, taking into account the drive ratios this limits the joint velocities as follows:

θ_{1max}=43 degrees/sec θ_{2max}=86 degrees/sec

As can be seen in the plots for the trajectory p300/2000 of joint velocities vs time (Fig.5.11 and Fig.5.12) these joint velocities exceed those bounded by the motor characteristics. Hence, it would be necessary to scale this path down to fulfill the motor speed capabilities.

The speed of the motor varies with load torque. The torque that the motor can supply at a given speed is a function of the power of the motor, this charcteristic will be considered next.

5.4.2 Power of Motor

The power of the motor can be represented by the following equation:

$P_{in}=P_{out}+P_{loss}$ where P_{in}=Power Input to motor= V*I

V=Terminal Voltage

I= Motor Current

P_{loss}=Power Losses in Motor=I^2R

R=armature resistance

P_{out}=Mechanical Power out

$P_{out}=T_m*\omega_m$

T_m=Motor Torque

ω_m=Motor rotational Velocity

For the RTX robot manipulator the motors used in the horizontal arm are rated as 3watts power however, this represents the continuous rating or RMS value. It is highly likely that under high accceleration demands this is exceeded. For instance, at the maximum error signal the motor sees the maximum terminal voltage, for the RTX this voltage is 24 volts. The maximum current is limited by the controller to 0.75 amps. It is possible to calculate the power loss as $I^2R=(0.75)^2*2.07=11.64$. Therefore,the instantaneous power is 18-11.64=6.36 watts as an absolute maximum. It will probably be less than this in order to protect the power amplifier.

From the planned trajectory and torque calculations it is possible to calculate the power required at each point along the trajectory and check that it does not exceed the instantaneous power of the motor. The RMS power can be calculated over the complete cycle to check that it falls within the rated value of 3 watts for the motor. The torque values are calculated by the program TORQUE and the torque program supplies the torque values to the program POWER which calculates the power required by the planned trajectory. The power requirements for the planned trajectory are as follows:

Motor Power 1 maximum=93.29 watts
Motor Power 2 maximum=51.81 watts
RMS Power 1= 7.72 watts
RMS Power 2= 3.51 watts

It is seen that the trajectory power requirements far exceed the motor capabilities.

5.4.3 Peak Motor Torque

The Peak Motor Torque is a function of the maximum supply current and the motor constant as follows:

Peak Motor Torque=Maximum Supply current * Motor constant

The maximum supply current is usually dictated by the controller, for the RTX the current is limited to a maximum of 0.75 amps. The motor constant is numerically equal to the back emf constant, for the RTX motors this is K_m=0.034Nm/amp. This gives a peak torque capacity of 0.026 Nm. The maximum torque required by the planned trajectory, calculated by the torque program is as follows:

Maximum Torque 1= 0.098 Nm
Maximum Torque 2= 0.022 Nm

The peak torque requirement for axis 1 is clearly exceeded. This means that the manipulator will not achieve the required acceleration at joint 1 for the total trajectory.

5.4.4 Peak Motor Acceleration

The peak acceleration can be calculated from the following equation:

$$\alpha = \frac{T}{J_{eff}}$$

Where T = Peak torque

J_{eff}=Effective Inertia

α=Angular acceleration

If the maximum effective inertia is used this gives a conservative estimate of the motor's acceleration capabilities. This has been calculated for each axis and is shown below:

Axis 1 maximum acceleration=171 degrees/sec^2
Axis 2 maximum acceleration=356 degrees/sec^2

As seen in the planned trajectory plots of joint accelerations(Fig.5.13 and Fig.5.14) the acceleration requirements exceed the motor capabilities. However, this is based upon no load conditions, so the acceleration will be considerable less for load conditions. Also, not all of the acceleration torque is available for accelerating the inertia, the dynamic terms such as coupling torque, caused by the Corriolis and centripetal accelerations, will take a proportion of the torque hence, reducing the acceleration capability of the motor.

To summarise, a planned trajectory should be within the following motor limitations:
1) Speed of Motor
2) Power of Motor
3) Peak Torque
4) Peak acceleration

If any of these motor characteristics are exceeded it is likely that the manipulator will not follow the planned trajectory.

The joint velocity and acceleration bounds are as follows:

Joint velocity axis 1 = 0.74 rads/sec or Motor velocity axis 1 = 654.5 rads/sec
Joint velocity axis 2 = 1.49 rads/sec or Motor velocity axis 2 = 654.5 rads/sec
Joint acceleration axis 1=2.98 rad/sec^2 or Motor acceleration axis 1=2615 rad/sec^2
Joint acceleration axis 2=6.21 rad/sec^2 or Motor acceleration axis 2=2723 rad/sec^2

Power and Torque bounds are as follows:
Power of Motor=3 watts(Continuous rating)

Peak Motor torque = 0.026 Nm

5.5 Bounds upon the dynamic coefficients based on the motor characteristics.

Having considered the motor limitations it is now possible to establish bounds on the dynamic coefficients, load and motor torque terms based upon the the motor limitations. First the dynamic coefficients are considered and their variation is tabulated in table 5.9.

Dynamic Coefficient	Maximum	Minimum
D_{11}	0.70	0.11
D_{12}	0.28	-0.01
D_{122}	0.15	-0.15
D_{112}	0.29	-0.29
D_{22}	0.14	0.14
D_{211}	0.15	-0.15

Table 5.9. Showing the maximum and minimum values of the dynamic coefficients based upon the motor characteristics.

The variation of the dynamic coefficients is dependent upon the configuration of link 2 i.e. a function of $\sin\theta_2$ and $\cos\theta_2$ which can take a range of values between +1 and -1. The range of the dynamic coefficients have been calculated for the RTX manipulator and using the bounds set by its actuators as shown in Table 5.9.

5.5.1 Variation in Load torque dynamic terms

Each term in the load torque dynamic equations is calculated based upon the previously tabulated values of table 5.9 and the upper bounds for the motor velocity and acceleration bounds. Each term is expressed as a percentage of the total torque.These values are tabulated in Table 5.10.

Load Torque Dynamic Term	Maximum (Nm)	% of Total	Minimum (Nm)
D11 $\ddot{\theta}_1$	2.10	46.5	0.08
D12 $\ddot{\theta}_2$	1.76	39.0	0.07
D122 $\dot{\theta}_2^2$	0.33	7.3	-0.32
D112 $\dot{\theta}_1 \dot{\theta}_2$	0.33	7.3	-0.29
Total=	4.51		
D22 $\ddot{\theta}_2$	0.84	47.6	0.84
D12 $\ddot{\theta}_1$	0.84	47.6	-0.03
D211 $\dot{\theta}_1^2$	0.08	4.6	-0.08
Total=	1.77		

Table 5.10. Showing the maximum and minimum that the load torque terms are as a percentage of the total torque.

As can be seen in table 5.10 above, the inertia torque($D_{11}\ddot{\theta}_1$) is a significant percentage of the total torque for axis 1(46%). The coupling torque is also a significant proportion of the torque (39%). The centripetal and Coriolis terms are a small percentage(7.3%) compared to the other terms. In table 5.10 the maximum torque totals for each axis were estimated to occur when each term in the dynamic equations(of eq. 3.15,3.14 of chapt.3) are at a maximum. It must be pointed out that it is assumed that the maximum for each dynamic term occur at the same time in order to calculate the percentage values. In practice this will not be the case and the individual dynamic terms could take larger percentages of the total torque. However, looking at the terms for say axis 1, the inertia and inertial coupling terms are a function of $\cos\theta_2$ and must be a maximum at $\theta_2=0,180,360^0$. However, the other coupling terms are negative and are a function of $\sin\theta_2$ and would be zero when $\theta_2=0,180,360^0$, hence, T_1 would not be at a maximum. Realistically, a maximum T_1 could be envisaged when $\sin\theta_2$ is negative and $\cos\theta_2$ is positive e.g when θ_2 is 315^0 i.e $\cos\theta_2=0.7$ and $\sin\theta_2=-0.7$. This would probably give a more reasonable estimate of the maximum torque T_1.

5.5.2 Variation in Inertia

The effective inertia as seen by the motor is calculated for no load conditions. The inertia of link 1 referred back to the motor through the gear box is found to be only 9% of the total inertia at the motor. The inertia as seen at motor 1 varies with the configuration of link 2 .The inertia of link 2 does not vary with configuration for this simplified manipulator and is 2% of the inertia as seen at motor 2. These results are listed below:

Inertia at motor 1 Maximum=9.9 E-6 Kgm^2 Minimum=9.15 E-6 Kgm^2
Inertia at motor 2 9.7 E-6 Kgm^2

5.5.3 Variation in Motor Torque dynamic terms

The torque taken up by the effective inertia of axis 1 is 90% of the total drive torque. The coupling inertia is still significant at 7%,while the Coriolis and centrifugal terms are insignificant at 1.5 %. These percentages are similar for axis 2. The computed values are shown in Table 5.11 below:

Motor Torque Dynamic Terms	Maximum (Nm)	% of total	Minimum (Nm)
Jeff1 $\ddot{\theta}_{m1}$	0.026	90.0	0.024
n1 D12 $\ddot{\theta}_2$	0.002	7.0	-8.1 E-5
n1 D122 $\dot{\theta}_2^2$	3.7 E-4	1.5	-3.7 E-4
n1 D112 $\dot{\theta}_1 \dot{\theta}_2$	3.7 E-4	1.5	-3.7 E-4
Jeff2 $\ddot{\theta}_{m2}$	0.026	92.0	0.026
n1 D12 $\ddot{\theta}_1$	0.0019	6.7	-7.8 E-5
n1 D211 $\dot{\theta}_1^2$	3.4 E-4	1.3	-3.4 E-4

Table 5.11. Showing the maximum and minimum that the motor torque terms are as a percentage of the total torque.

5.5.4 Power

The power requirements are based upon assuming that all maximum torque terms occur at the same time. These are calculated for each motor based the values in Table 5.11:

Motor Power 1=18.8 Watts
Motor Power 2=18.8 Watts

The RTX motors are only rated at 3 watts , however, this is a continuous rating.The instantaneous power rating was calculated previously to be 6.36 Watts. This means that the motor power is somewhat under rated for the extreme bounds of the torque equations. This could be over come by having a larger current amplifier.

5.6 Selection of Motor speed and acceleration characteristics based on Kinematics

It has been seen how the joint positions, velocities and accelerations can be computed for a planned trajectory in Cartesian space using the kinematics of the manipulator. With the characteristics of the drive motor given, it is possible to assess whether the motor can achieve these planned joint demands. If the planned joint demands are above those achievable by the motor the planned trajectory can be scaled down accordingly. However, from a design point of view the motor speed and acceleration characteristics would not be known and have to be chosen. What would be known is the required Cartesian velocity and the reach or working volume of the manipulator. The problem then is to determine what are the required motor speed and acceleration specifications. This problem will now be considered.

As seen in the chapter on robot kinematics, the relationship between Cartesian velocities and joint angles(i.e the inverse Jacobian) is not a simple one and depends upon the joint positions as well as the specified Cartesian velocities for each axis. However, at a singularity the determinant of the Jacobian becomes zero, where the determinant is:

Determinant=$L_1 L_2 \sin\theta_2 = 0$ at a singularity.

This occurs for θ_2=0 or θ_2=180 degrees

Near a singularity the angular velocity of the joints become very large. A singularity occurs at the boundary of the work space of the manipulator, for this manipulator this is near the perimeter of the circle and the centre of the circle that the links make with respect to joint 1. From the inverse Jacobian equation in the chapter on kinematics it is seen that the denominator of this equation is a function of $\sin\theta_2$:

$$\begin{bmatrix}\dot{\theta}_1\\ \dot{\theta}_2\end{bmatrix}=\frac{1}{L_1 L_2 \sin\theta_2}\begin{bmatrix} L_2\cos(\theta_1+\theta_2) & L_2\sin(\theta_1+\theta_2)\\ -L_1\cos\theta_1 - L_2\cos(\theta_1+\theta_2) & -L_1\sin\theta_1 - L_2\sin(\theta_1+\theta_2)\end{bmatrix}\begin{bmatrix}\dot{X}\\ \dot{Y}\end{bmatrix}$$

At a singularity the value of the denominator approaches zero and hence the value of the joint angle velocities becomes large. However, if the value of θ_2 is restricted the values of the maximum joint velocities are restricted. The value of θ_2 can be restricted by reducing the working volume that the manipulator can work in i.e. decreasing the outer and inner working radius. The working radius is a function of the link lengths and joint angle θ_2. From the inverse kinematic equations for position joint angle θ_2 is solved from the following equation:

$$\theta_2 = \cos^{-1}\left(\frac{R^2 - L_1^2 - L_2^2}{2 L_1 L_2}\right)$$

Where R is the radius of the end point to the center of joint 1.

If R is chosen as an inner and outer radius, this then restricts the value of θ_2 and hence the maximum that the joint angles can take near the singularity at the chosen inner and outer radius. The minimum value of $\sin\theta_2$ for the two solutions from the two chosen radii then used in determining the maximum joint velocities. Differentiating joint velocitiy 1 with respect to θ_1 and equating this resultant expression equal to zero to find a maximum:

$$\frac{\partial\dot{\theta}_1}{\partial\theta_2}=\frac{-L_2\sin(\theta_1+\theta_2)\,\dot{X}+L_2\cos(\theta_1+\theta_2)\,\dot{Y}}{L_1 L_2\sin\theta_2}=0$$

Assumuing L_1 and L_2 are the same and that the maximum values that the Cartesian velocities take is when they equal the specified resultant velocity, the above expression simplifies to the following:

$Sin(\theta_1+\theta_2)=cos(\theta_1+\theta_2)$

hence, solving for $\theta_1+\theta_2$ the solution is obtained from $Tan^{-1}(1)= \theta_1+\theta_2=45^0$ or 225^0.

5.6.1 Example

Consider the RTX manipulator with equal link lengths of 0.2535 m, if the inner radius is restricted to a tenth of the combined link measurement and the outer radius equal to the combined link measurement minus a tenth of the combined link measurement, what is the expected maximum joint 1 velocity for a specified Cartesian resultant velocity of 0.3 m/sec?.

R_{min}=0.0507 m and R_{max}=0.4563 m

This gives $cos\theta_2$=-0.98 or $cos\theta_2$=0.62 respectively.

This means that θ_2 is restricted between 51.67 to 168.54 degrees
and between 191.46 to 308.32 degrees.

Sin θ_2 is a minimum for θ_2=168.54 or 191.46 degrees, this will be used to calculate the maximum joint 1 velocity. Hence, using the inverse Jacobian expression the maximum joint 1 velocity can be found. This is calculated to be 482 degrees/sec.

In order test out this method a path will be considered that just cuts the inner circle at a tangent as shown in the following diagram(Fig.5.20).

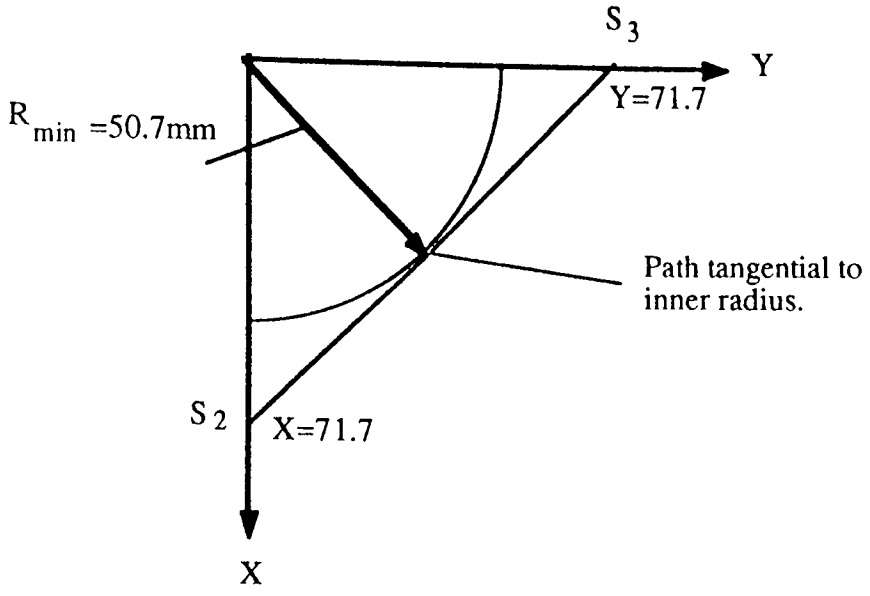


Figure 5.20. Diagram showing the example path.

The path that corresponds to the following corner points shown in Table 5.12.

Corner Point	P_X (mm)	P_Y (mm)
S1	400	0
S2	71.7	0
S3	0	71.7
S4	0	400

Table 5.12 Showing the path points for the example path.

The joint velocities calculated for this path are found to be:
Joint 1 velocity maximum=339 degrees/sec
Joint 2 velocity maximum=101 degrees/sec

These are well within our estimated value for joint 1 velocity of 482 degrees/sec. However, if instead of inputting the resultant velocity as 0.3m/sec as the required Cartesian velocity but input the resultant of an X and Y Cartesian velocity 0.3m/sec i.e a value of 0.424 m/sec. The maximum value of joint 1 velocity becomes the value predicted previously of 482 degrees/sec as would be expected.

In terms of the angular accelerations, the inverse derivative of the Jacobian is used and joint acceleration 1 is partially differentiated with respect to θ_1.This gives the same result as previously i.e a max occurs at $\theta_1 + \theta_2 = 45$ degree. Computing the maximum joint acceleration using the previous example values and assuming that the sum of the maximum joint velocities is not more than the single joint velocity, the estimated joint 1 acceleration is equal to 4546 degrees/sec^2. This compares with the maximum for the test path of Fig.5.12, for a specified Cartesian acceleration of 2 m/sec^2, $|\theta_1''| = 3041$ degrees/sec^2.

The method that has been outlined gives an esimate of joint 1 maximum velocity and acceleration. This estimate could be used for joint 2 maximum velocity and acceleration.With these estimates a suitable motor could be chosen that fulfills these joint requirements by consulting the motor characteristics from the relevant catalogues.

5.6 Selection of Motor Power,Torque and drive ratio based upon the Dynamic Equations

Having selected the speed and acceleration requirements of the motor purely based upon the kinematics, the next step is to consider the torque and power characteristics based upon the dynamic equations.

The configuration dependent dynamic coefficients boundaries have already been established. The maximum value of these is used together with the joint velocities and accelerations which were established at a singularity. The load torque expected maximum is thus calculated assumung all terms occur as a maximum at the same time as shown below:

$$T_{L1max} = D_{11max}\ddot{\theta}_{1max} + D_{12max}\ddot{\theta}_{2max} + D_{122max}\dot{\theta}_{2max}^2 + D_{112max}\dot{\theta}_{1max}\dot{\theta}_{2max}$$

$$T_{L2max} = D_{22max}\ddot{\theta}_{2max} + D_{12max}\ddot{\theta}_{1max} + D_{211}\dot{\theta}_{1max}^2$$

These equations give the maximum possible torque on the load side that the motor would be required to deliver. This would have to be matched to the peak torque characteristic. At this point none of the motor characteristics have been used in the

calculation of the expected maximum torque. Obviously when calculating the torque referred to the motor side the effective inertia term includes the motor inertia. Also the dynamic terms referred back to the motor will include the gearbox ratio. Its at this point that it would be necessary to choose a suitable motor that fulfills the speed and acceleration requirements and the torque requirements.

The power of the motor is an important characteristic that will have to be considered in the selection of the motor. This can be calculated as follows:

$$P_{1max} = T_{L1max}\,\dot{\theta}_{1max} \qquad \text{and } P_{2max} = T_{L2max}\,\dot{\theta}_{2max}$$

A direct drive motor could be considered if it can fulfill the joint velocity and acceleration requirements together with the torque and power requirements. However, if these cannnot be obtained by a direct drive motor a gearbox would have to be included. The ratio of the gear box will be a compromise between the kinematic requirements(maximum cartesian speeds required) and the torque requirements (load handling capacity).

For a mainly inertial system such as the manipulator an optimum gear ratio can be found. Consider the following system(Fig.5.21):

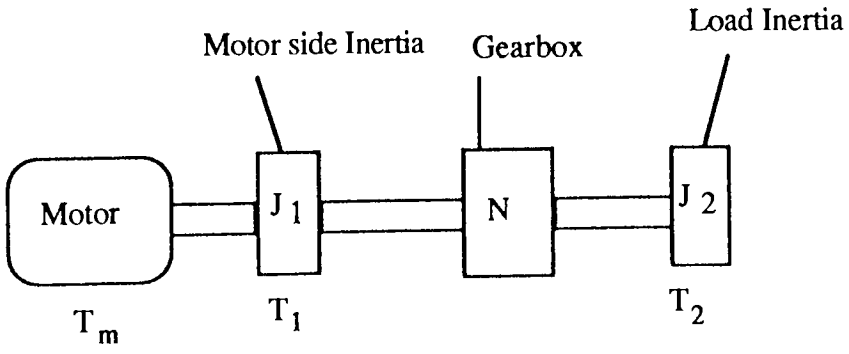


Figure 5.21 Diagram showing drive chain for the load inertia.

The torque at load J_2 side is as follows:

$$T_2 + n\,T_1 = J_2\,\dot{\omega}_2 + J_1\,\dot{\omega}_1 \quad \text{where } \frac{\dot{\omega}_1}{\dot{\omega}_2} = n$$

This can now be written as follows:

$$T_2 + n\,T_1 = (J_2 + n^2 J_1)\,\dot{\omega}_2$$

An optimum gear ratio that maximizes load shaft acceleration for a given input torque exists(Doebelin[29]). Since the torque effect on acceleration increases with n, where as the inertial effect decreases acceleration as n^2, an optimum n should exist. Rearranging the above expression in terms of the acceleration and differentiating the acceleration with respect to n:

$$\frac{\partial \dot{\omega}_2}{\partial n} = \frac{(J_2 + n^2 J_1)\,T_1 - (2\,n\,J_1)\,n\,T_1}{(J_2 + n^2 J_1)^2} = 0 \quad \text{for a max or min}$$

This gives an optimum ratio :

$$n_{opt} = \sqrt{\frac{J_2}{J_1}}$$

This optimum ratio reflects back the load inertia as equal to the drive inertia. For the RTX the ratio's for each axis are very large:

Axis 1 =877 and Axis 2=438

This results in the actuator inertia being far larger than the refected link inertia. For example :

Link 1 Inertia=0.66 kg m^2 Actuator 1 Inertia=9.0 E-6 kg m^2

Refected Link 1 Inertia=8.6 E-7 kg m^2

The RTX uses fairly small motors of power 3 Watts and speed of 2800 RPM. The peak torque is 0.026 Nm. They have used a very high ratio in order to fulfill the torque requirements, however they have had to compromise the Cartesian speed capabilities in doing so.

Having selected all of the motor characteristics based upon the extremes that would be demanded of the motor at any time in the path, it would be expected that

the motor should be capable of delivering the demands of the controller. How well the manipulator follows the path will depend upon the control systen design. This aspect will be considered in the next chapter.

5.7 Significance of the terms in the dynamic equations for a specific planned trajectory

The significance of the Load Torque terms in the dynamic equations will depend upon the path and the velocity to acceleration ratio chosen for that path. In order to assess the significance of each of the terms, the absolute total torque is calculated for each axis. The absolute value of each term is then expressed as a percentage of the total absolute torque. The maximum percentage of the total each term takes in the path, is then used as a measure of the significance of that term. The significance of the terms have been evaluated for path p/300/2000. These are shown tabulated in table 5.13. Also the results are shown for the same path but doubling the velocity(P/600/2000) and doubling the acceleration(P/300/4000). Also doubling the velocity and acceleration(P/600/4000) and retaining the same linear proportion(P/600/8000).

Path	TL1 A1 (%)	TL1 A3 (%)	TL1 A4 (%)	TL1 A5 (%)	TL1 max (Nm)	TL2 A1 (%)	TL2 A3 (%)	TL2 A4 (%)	TL2 max (Nm)
P/300/2000/	86.6	84.3	9.8	22.4	3.05	94.8	82.7	29.4	-1.08
P/600/2000/	92.2	90.4	10.6	22.1	3.05	99.5	99.4	29.4	-1.11
P/300/4000/	79.4	81.5	9.9	22.4	6.09	89.9	74.5	29.4	-2.10
P/600/8000/	86.6	84.3	9.8	22.4	12.19	94.8	82.7	29.4	-4.31

Table 5.13. Showing the significance of the load torque terms for a number of example trajectories.

As seen in table 5.13 the coupling inertia term for axis 1 is significant at 84.3%.

The Coriolis term is the next most significant as 22.4%. The least significant term is the centripetal term as 9.8%. For axis 2 the again the coupling inertia is of the same order as 82.7%. The centripetal term is more significant for axis 2 that axis 1 as 29.4%. There are no great changes in the significance of the terms for changes in the velocity and acceleration. The significance of the terms remains the same for the same linear ratio for path p/600/8000/. This is a very interesting result since not only has path scaling occurred but also dynamic scaling. The dynamic scaling can be deduced intuitively if the dynamic equations of the manipulator are observed. For instance consider the dynamic equation for axis 1.

For axis 1 the dynamic equation is

$$T_1 = D_{11}\ddot{\theta}_1 + D_{12}\ddot{\theta}_2 + D_{112}\dot{\theta}_1\dot{\theta}_2 + D_{122}\dot{\theta}_2^{\,2}$$

If the velocity is scaled by 2 and the acceleration by 4 then the equation above becomes:

$$T_{1S} = D_{11}\,4\,\ddot{\theta}_1 + D_{12}\,4\,\ddot{\theta}_2 + D_{112}\,2\,\dot{\theta}_1\,2\,\dot{\theta}_2 + D_{122}\,4\,\dot{\theta}_2^{\,2}$$

The value of 4 can be taken out as a common factor hence the above equation becomes:

$$T_{1S} = 4\,(\,D_{11}\,\ddot{\theta}_1 + D_{12}\,\ddot{\theta}_2 + D_{112}\,\dot{\theta}_1\,2\,\dot{\theta}_2 + D_{122}\,\dot{\theta}_2^{\,2}\,) = 4\,T_1$$

If we consider say the percentage that $D_{112}\dot{\theta}_1\dot{\theta}_2$ takes this is $\dfrac{D_{112}\dot{\theta}_1\dot{\theta}_2}{T_1}$

The scaled percentage becomes $\dfrac{4\,D_{112}\dot{\theta}_1\dot{\theta}_2}{4\,T_1}$

Hence, the ratio obtained for the scaled version is the same as the previous non scaled version provided that it is scaled according to the established relationship.

The figures calculated in table 5.13 show the order of the significance of each of the terms for a particular path and a particular velocity to acceleration ratio. However, if the path is near a singularity it would be expected that the significance of the velocity related terms would be greater. The path shown in fig.5.12 goes

near a singularity and the significance of the terms has been evaluated for the path for the same velocity to acceleration ratio as before. The results are shown in table 5.14.

Path	TL1 A1 (%)	TL1 A3 (%)	TL1 A4 (%)	TL1 A5 (%)	TL1 max (Nm)	TL2 A1 (%)	TL2 A3 (%)	TL2 A4 (%)	TL2 max (Nm)
P/300/2000/ Near a singularity	97.9	41.6	47.2	47.2	-6.4	86.1	38.7	51.9	1.02

Table 5.14. Showing the significance of the dynamic terms when the path goes near a singularity.

N.B The dynamic terms $T_{L1}A_1(\%)$ etc. are defined in appendix 1.

As seen in table 5.14 the Coriolis and centripetal terms become a far higher percentage of the total torque near a singularity, of the order of 50%.

Each of the dynamic terms do not necessarily reach their maximum value of significance at the same point in the trajectory. At a point along the trajectory when the total torque demand is high the percentage of the total that a particular dynamic term takes is more significant at that point(as the demand on the motor is high and may be exceeded) than when the torque demand is low. In order to take into account this significance the total absolute torque is expressed as a fraction(weighting value) of the maximum torque that it takes along the path. For a dynamic term a weighted percentage of the total absolute torque is calculated at each point along the trajectory by multiplying the percentage of the total absolute that the term takes by the weighting value at that point. The maximum value that the term takes is then a measure of the significance of that term. This has been calculated for the two paths of P/300/2000/ including the path that is close to a singularity as shown in Table 5.15.

Weighted load torque terms as a percentage of the total absolute torque.									
Path P/300/2000/	TL1 A1 (%)	TL1 A3 (%)	TL1 A4 (%)	TL1 A5 (%)	TL1 max (Nm)	TL2 A1 (%)	TL2 A3 (%)	TL2 A4 (%)	TL2 max (Nm)
Standard	55.0	44.9	1.51	2.23	14.6	51.7	47.8	2.4	6.6
Near a singularity	96.4	32.4	6.76	6.8	6.8	61.2	38.7	36.3	2.8

Table 5.15. Showing the weighted significance of the dynamic torque terms.

N.B The dynamic terms $T_{L1}A_1(\%)$ etc. are defined in appendix 1.

As seen in table 5.15 the weighted values are far lower than those in the previous table however, they are still shown to be of significance.

5.9 Generalisation of the dynamic equations

Consider a two link manipulator of uniform cross-sectional area(Fig.5.22). In the design of the manipulator the reach of the manipulator would be known (this sets the total of the sum of the link lengths) and the working volume(this sets the ratio of the two link lengths). The link ratio and the reach capacity in terms of the link lengths are defined as:

$\text{Ratio}=L_1/L_2$ and $\text{Reach}=L_1+L_2$

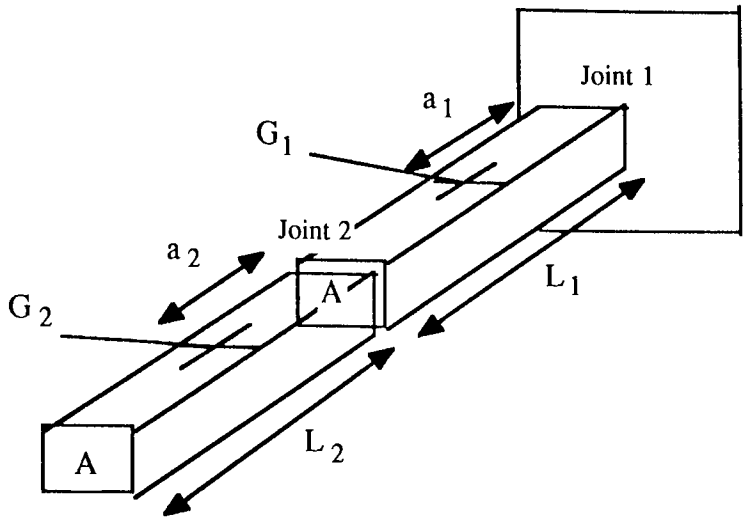


Fig.5.22. Diagram showing the generalisation of link parameters.

From the two equations above, expressions for the link lengths in terms of the Ratio and Reach can be found:

$$L_1 = \frac{\text{Reach*Ratio}}{(\text{Ratio}+1)} \qquad L_2 = \frac{\text{Reach}}{(\text{Ratio}+1)}$$

The dynamic equation coefficients can be related to the link lengths.The inertia of the links about the centre of gravity is:

$$I_{G1} = \frac{m_1 L_1^2}{12} \quad \text{and} \quad I_{G2} = \frac{m_2 L_2^2}{12}$$

The centre of gravity for each link is assumed to be situated at mid point along the link length.

$$a_1 = \frac{L_1}{2} \quad \text{and } a_2 = \frac{L_2}{2}$$

If it assumed that the two links will be designed to have the same cross-sectional area then $A_1=A_2$. The mass of the links can be related to the density of the material they are made from(ρ), the cross-sectional area(A) and the link lengths:

$$m_1=\rho\, A\, L_1 \quad \text{and} \quad m_2=\rho\, A\, L_2$$

Hence, the dynamic equation coefficients can be written in terms of the link lengths and divided through by the density and area as shown below:

$$\frac{D_{11}}{\rho A} = \left(\frac{L_1^3}{3} + \frac{L_2^3}{3} + L_2 L_1^2 + L_1 L_2^2 + L_1 L_2^2 \cos\theta_2\right)$$

$$\frac{D_{12}}{\rho A} = \left(\frac{L_2^3}{3} + \frac{L_1 L_2^2 \cos\theta_2}{2}\right)$$

$$\frac{D_{122}}{\rho A} = \left(-\frac{L_1 L_2^2 \sin\theta_2}{2}\right)$$

$$\frac{D_{112}}{\rho A} = \left(-L_1 L_2^2 \sin\theta_2\right)$$

$$\frac{D_{211}}{\rho A} = \left(\frac{L_1 L_2^2 \sin\theta_2}{2}\right)$$

$$\frac{D_{22}}{\rho A} = \left(\frac{L_2^3}{3}\right)$$

The maximum value that the coefficients take for a given reach can be calculated, a reach of 0.507m is used in the tabulated values shown in table 5.16.

Ratio	L_1 (mm)	L_2 (mm)	D_{11}	D_{12}	D_{122}	D_{112}	D_{211}	D_{22}
0.15	66.1	440.9	434	350	64	128	64	286
0.25	101.4	405.6	434	306	83	167	83	222
0.4285	152.1	354.9	434	245	96	192	96	149
0.66	202.9	304.2	434	188	94	188	94	94
1.0	253.5	253.5	434	136	81	163	81	54
1.5	304.2	202.8	434	90	63	125	63	28
2.33	354.7	152.3	434	53	41	82	41	12
4.0	405.6	101.4	434	24	21	42	21	3
7.0	443.6	63.4	434	10	9	18	9	0.8

Table 5.16. Showing the maximum values of the dynamic coefficients for a given

reach of 0.507m for a range of link ratios.

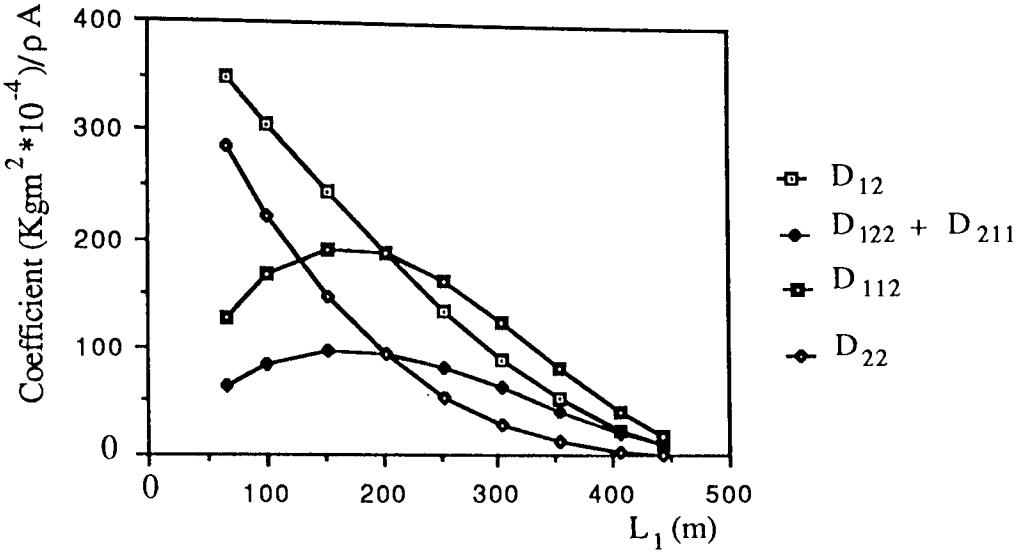


Fig.5.23. Plot of the results of table 5.16.

The tabulated values of table 5.16 are shown plotted in fig.5.23 against link length L_1. Coefficient D_{11} has not been plotted as this is constant as shown in the previous table. Equally the dynamic coefficients could be plotted against L_2.

The maximum values that the coefficients take for a given ratio are plotted in fig.5.23. In order to calculate the maximum expected torques, it is necessary to evaluate the maximum expected joint velocities and accelerations for a particular path and link ratio. Obviously the joint angles, velocities and accelerations vary dependent upon the link Ratio and Reach capacity.

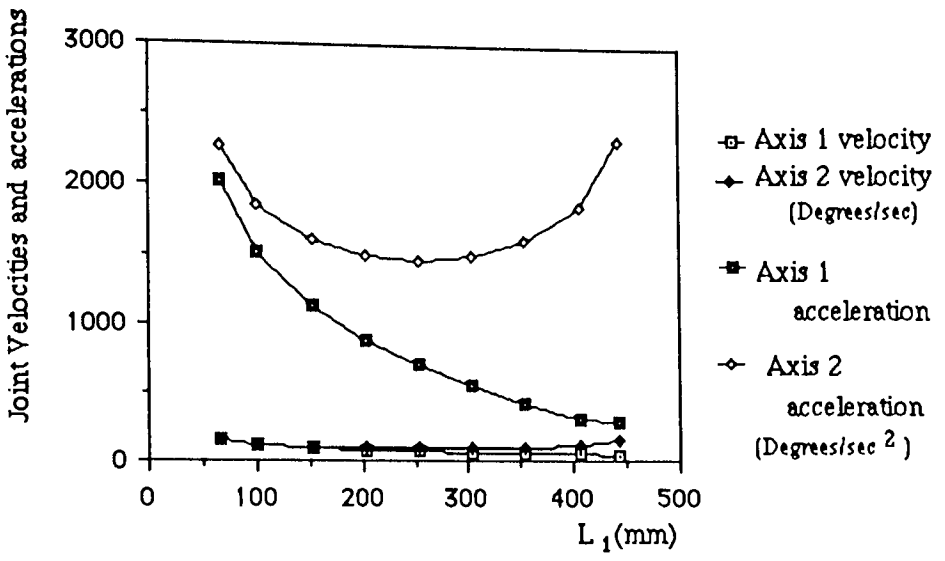


Figure 5.24. Plot of the link velocities and accelerations versus link length L_1.

The maximum joint velocities can be used in conjunction with the maximum coefficient values to calculate the expected torque values for axis 1 and axis 2.

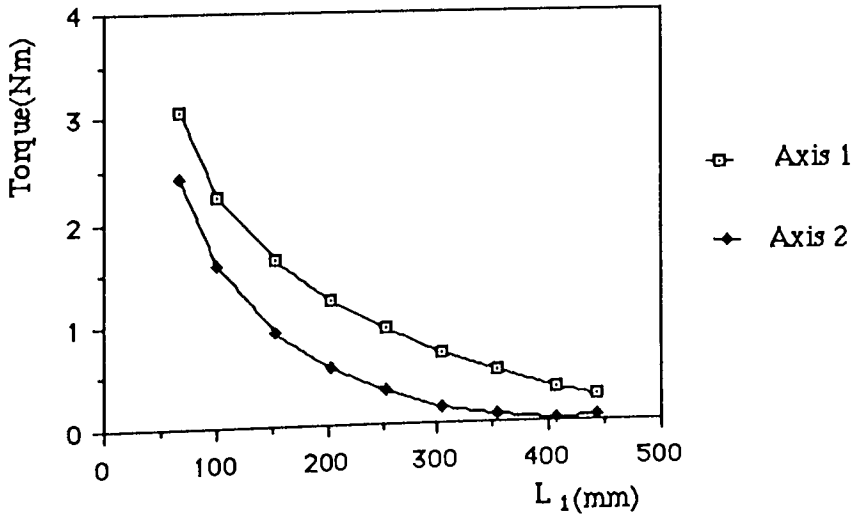


Figure 5.25. Plot of the expected maximum torque values versus link length L_1.

The expected maximum torques per unit area and density are shown in Fig.5.25 for a range of ratios.

In the next chapter the ability of the robot manipulator to follow the trajectories planned in this chapter are assessed.

Chapter 6:Simulation of Control Stategies

Chapter 6 - Simulation of Control Strategies

6.1 Introduction

In this chapter the main types of control strategies are considered including: Proportional plus Derivative(PD), Proportional+Derivative+Integral(PID) and Proportional with derivative feedback. Initially the design of suitable controller gains for these strategies is considered for the single axis case where the robot manipulator is modelled as a simple inertia. The actuator can be of a number of types, two types are considered that of a Direct Current(D.C) Motor a)Field controlled b)Armature controlled. The transfer function of the D.C. motor may be simplified under certain conditions, these conditions are considered. The control strategies are tested by using a full non-linear simulation of the two axis robot manipulator that is required to track along a preplanned path which was established in Chapter 5. The results of the simulation for the different controllers are compared later in this chapter.

In order to fully understand the performance of the various control strategies the non-linear equations of the manipulator are linearised about a particular operating point. These equations are then used to obtain a state space model of the manipulator together with the control systen under consideration. The linearised model is then verified against the full non-linear response. With the state space model, the response of the system is investigated in more detail by computing the eigenvalues and formulating the transfer function equations in symbolic form.

6.2 Independent Axis Design

6.2.1 Modelling of the manipulator

Assumptions in modelling the robot manipulator in simulation are as follows:

1. Each link of the robot is modelled as a rigid body.
2. Although each link is modelled as a rigid body, in order to set the gain of the controller the structural resonant frequency of each joint was experimentally determined with no load and a 10 Kg load. The controller gains are set to give a natural frequency of half the structural frequency.
3. The drive between the motor and the link of the manipulator is modelled as a drive ratio n. It is assumed that there is no compliance or backlash in the drive and it transmits the power with 100% efficiency. It is also assumed that there is no static or viscous friction in the drive system.
4. The dynamic model of the two link manipulator is defined by the derived equations 3.15 and 3.14 of chapter 3.
5. The torque is referred to the motor shaft and the motor is modelled as a simple inertia.

The equations of the manipulator on the load side, in the standard D notation ,as discussed in chapter 3, are as follows:

$$T_{L1} = D_{11}\ddot{\theta}_1 + D_{12}\ddot{\theta}_2 + D_{122}\dot{\theta}_2^2 + D_{112}\dot{\theta}_1\dot{\theta}_2 + D_1$$

$$T_{L2} = D_{22}\ddot{\theta}_2 + D_{12}\ddot{\theta}_1 + D_{211}\dot{\theta}_1^2 + D_2$$

Where suffix L refers to the torque at the load side of the gearbox.

Referring the torque to the motor side of the gearbox gives the following torque equations:

$$T_{M1} = (J_{A1} + J_{M1})\ddot{\theta}_{M1} + n_1 T_{L1}$$

$$T_{M2} = (J_{A2} + J_{M2})\ddot{\theta}_{M2} + n_2 T_{L2}$$

Where J_A and J_M refer to the actuator and motor fittings respectively.

and $n_1 = \omega_{L1}/\omega_{M1} < 1$ and similarly $n_2 = \omega_{L2}/\omega_{M2} < 1$

The equations can be written in terms of the effective inertia for each axis as follows:

$$T_{M1} = Jeff_1 \ddot{\theta}_{M1} + n_1 (D_{12} \ddot{\theta}_2 + D_{122} \dot{\theta}_2^2 + D_{112} \dot{\theta}_1 \dot{\theta}_2 + D_1)$$

$$T_{M2} = Jeff_2 \ddot{\theta}_{M2} + n_2 (D_{12} \ddot{\theta}_1 + D_{211} \dot{\theta}_1^2 + D_2)$$

Where $Jeff_1 = (J_{A1} + J_{M1} + n_1^2 D_{11})$ and $Jeff_2 = (J_{A2} + J_{M2} + n_2^2 D_{22})$

and $n_1^2 D_{11}$ is the reflected inertia of link1 at actuator 1 through gearbox 1

$n_2^2 D_{22}$ is the refected inertia of link2 at actuator 2 through gearbox 2.

If the coupling terms are ignored (this can often be justified if the coupling terms are small compared to the actuator torque), and the manipulator works in the horizontal plane (i.e. gravity terms are not present), the dynamic equations become:

$$T_{M1} = Jeff_1 \ddot{\theta}_{M1}$$

$$T_{M2} = Jeff_2 \ddot{\theta}_{M2}$$

The block diagram representation of the single axis system is shown in Fig.6.1.

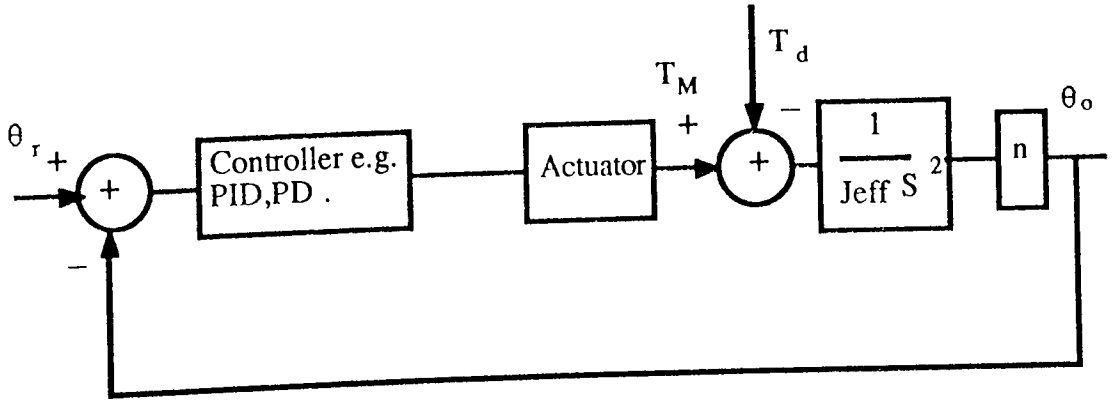


Figure 6.1. Block diagram representation of the independent axis control system for a single axis of the robot manipulator.

6.2.2 Modelling of actuator

A schematic diagram of a typical D.C. motor is shown in Fig.6.2 as follows:

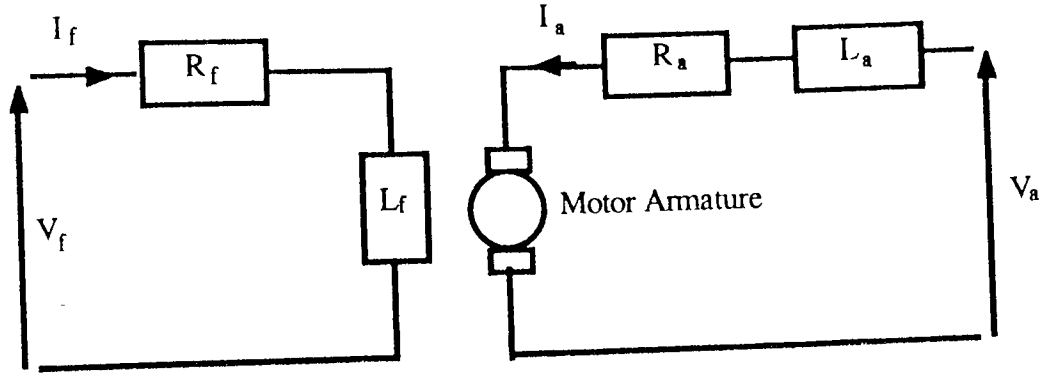


Figure 6.2. Schematic diagram of a typical Direct Current(D.C) motor.

6.2.3 Field Controlled D.C. Motor

The torque produced by the motor is proportional to the product of the armature current I_a and the air gap flux ψ,which is proportional to the field current

i.e

$$\psi=K_f I_f \qquad (6.1)$$

where K_f is a constant.

The torque T produced by the motor can be written as follows:

$$T=K_f I_f K_a I_a \qquad (6.2)$$

where K_a is a constant.

In the Field-Controlled D.C. Motor the armature voltage V_a is held constant, hence the armature current I_a is kept constant. With a constant armature voltage the developed torque can be taken to be proportional to the field current:

$$T=K_t I_f \qquad (6.3)$$

where K_t = motor torque constant

The equation for the field current is as follows:

$$I_f = \frac{V_f}{L_f S + R_f} \qquad (6.4)$$

The motor can be considered as a inertia J with a viscous damping of B. The torque produced by the motor is thus:

$$T = (J S^2 + B S)\,\theta \qquad (6.5)$$

Substituting (6.4) into (6.3) and equating to (6.5) the overall transfer function may be obtained:

$$\frac{\theta}{V_f} = \frac{K_t}{(L_f S + R_f)(J S^2 + B S)} \qquad (6.6)$$

Dividing numerator and denominator of (6.6) by R_f and B the following transfer function is obtained:

$$\frac{\theta}{V_f} = \frac{K_t / R_f B}{S(1 + T_m S)(1 + T_f S)} \qquad (6.7)$$

where $T_m = J / B$ = motor time constant

$T_f = L_f / R_f$ = field time constant

The representation of the motor transfer function can be simplified. For instance, consider the case when $T_f << T_m$, also if the system has very little damping then the damping factor B is set to zero. The transfer function for the motor can thus be modelled as a gain term, for the purpose of the simulation model this gain term will be taken as unity.

6.2.4 Armature Controlled D.C. Motor

In the Armature-controlled D.C. motor the field current is held constant. For a constant field current the flux becomes constant and so the torque produced by the motor is directly proportional to the armature current:

$$T = K_t I_a \qquad (6.8)$$

When the armature is rotating a voltage is induced in the armature that is proportional to the product of the flux and the angular velocity. For a constant flux, the induced voltage V_b is directly proportional to the angular velocity ω:

$$V_b = K_b \omega \qquad (6.9)$$

where K_b is the back emf constant.

The voltage applied to the armature V_a equals the sum of the voltages across the armature inductance and resistance plus the induced voltage V_b:

$$V_a = R_a I_a + L_a I_a S + V_b \qquad (6.10)$$

Representing the transfer function of the armature controlled D.C. Motor in block diagram form as shown in Fig.6.3:

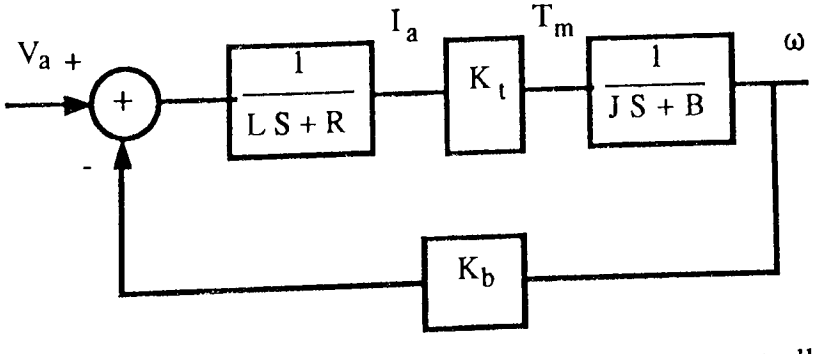


Figure 6.3. Block diagram representation of an armature controlled D.C. Motor.

The transfer function for the armature controlled D.C. Motor from fig.6.3 is as follows:

$$\frac{\omega}{V_a} = \frac{1/K_b}{\frac{JL}{K_t K_b} S^2 + \frac{(BL + RJ)}{K_t K_b} S + (\frac{RB}{K_t K_b} + 1)} \qquad (6.11)$$

Factorising the denominator in the from of $(T_m S + \gamma)(T_a S + 1) + 1$:

$$T_m T_a S^2 + (\gamma T_a + T_m) S + (\gamma + 1) \qquad (6.12)$$

Equating the denominator of (6.11) to (6.12) gives the following coefficients:

$\gamma = R B / K_t K_b$ = Damping factor

$T_m = R J / K_t K_b$ = motor time constant

$T_a = L / R$ = armature time constant

In order to simplify the transfer function the case will be considered when the viscous damping coefficient B is zero and $T_a << T_m$.The transfer function simplifies to the following:

$$\frac{\omega}{V_a} = \frac{1/K_b}{\frac{RJ}{K_t K_b} S + 1} \qquad (6.13)$$

The simplified transfer function for the field controlled D.C. Motor will be used in considering the various independent axis control strategies. The general block diagram for this case is a modified version of Fig.6.1 and is redrawn in Fig.6.4 below:

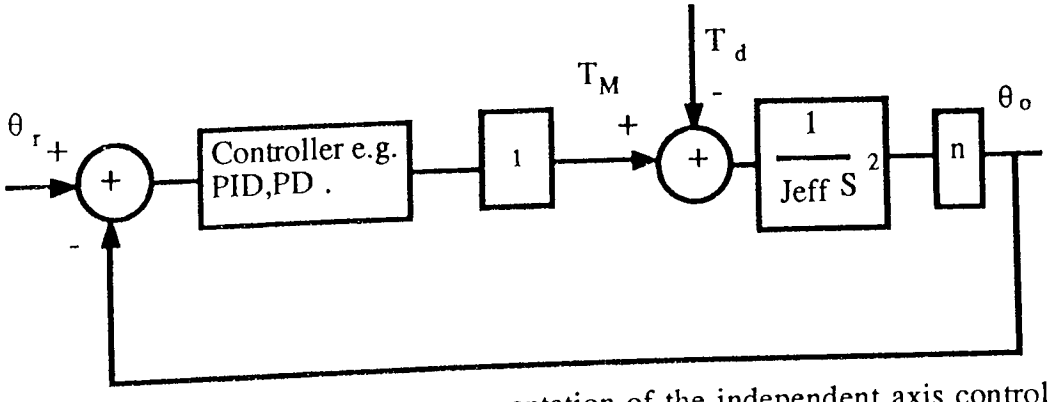


Fig.6.4. General block diagram representation of the independent axis control system for a single axis of the robot manipulator.

6.3 Independent Axis Control Strategies

The control strategies that are considered are listed below:

i) PD control
ii) Proportional + derivative feedback
iii) PID control
iv) PI + derivative feedback
v) PD + acceleration feedforward
vi) PID with acceleration feedforward

The characteristics of the first four control systems are summarised in the table below in terms of a) Characteristic equation order and numerator order b) Type number c)Error to a test input d) Error to a disturbance input:

Type of Control	Order of Num.	Order of Den.	Type No.	Test Input	Error to test input	Test Distur-bance	Error to test distur-bance
PD	1	2	2	$\frac{\theta_r}{S^3}$	$\frac{\theta_r J}{K_p N}$	$\frac{T_d}{S}$	$\frac{T_d}{K_p}$
P + deri-vative feedback	0	2	1	$\frac{\theta_r}{S^2}$	$\frac{\theta_r K_d}{K_p}$	$\frac{T_d}{S}$	$\frac{T_d}{K_p}$
PID	2	3	3	$\frac{\theta_r}{S^4}$	$\frac{\theta_r J}{K_I N}$	$\frac{T_d}{S^2}$	$\frac{T_d}{K_I}$
PI + deri-vative feedback	1	3	2	$\frac{\theta_r}{S^3}$	$\frac{\theta_r K_d}{K_I}$	$\frac{T_d}{S^2}$	$\frac{T_d}{K_I}$

Table 6.1. Showing the characterisation of the control stategies considered.

The latter two control system strategies are not included in table 6.1 as they are slightly different from the other four strategies, they will now be considered:

vi) PD + acceleration feedforward

The block diagram for this type of controller is shown in Fig.6.5:

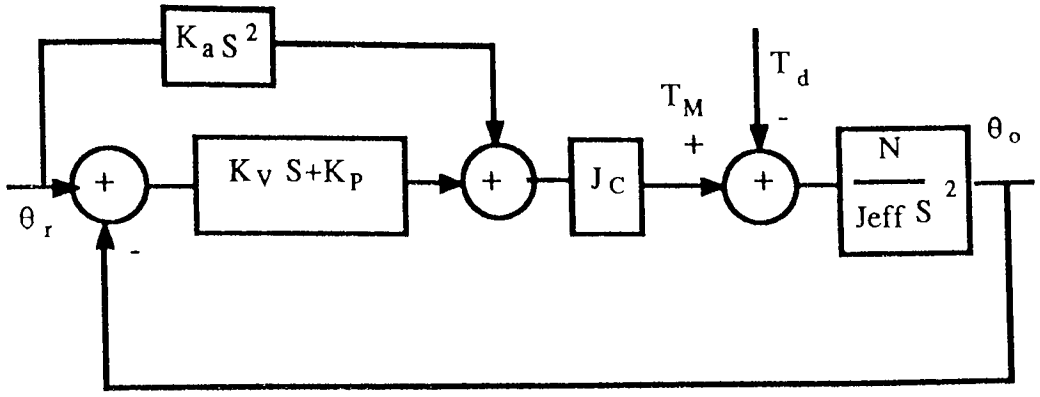


Figure. 6.5. Block diagram representation of the PD + acceleration feedforward independent axis control system for a single axis of the robot manipulator.

The transfer function for this system can be shown to be as follows:

$$\theta_o = \frac{(K_a S^2 + K_v S + K_p) J_c n \theta_r - T_d}{J S^2 + J_c n K_v S + J_c n K_p} \qquad (6.14)$$

If the transfer function is divided by J the actual inertia and letting $C=J_c/J$ then the transfer function becomes :

$$\theta_o = \frac{(K_a S^2 + K_v S + K_p) C n \theta_r - T_d n / J}{S^2 + C n K_v S + C n K_p} \qquad (6.15)$$

It is seen that if the value of J equals the Computed value J_c and $K_a=1/n$, the numerator cancels the denominator and there is no error to what ever type of input except that due to disturbance loads. However, this is not likely to occur since there will be inaccuracies in the estimate of the inertia. The inertia of the manipulator will vary and is configuration dependent also, it will change if the manipulator picks up an object. Thus, unless the computed inertia is continually computed, based on these changes, the numerator and denominator of the transfer function will not cancel exactly.

vi) PID with acceleration feedforward

The block diagram for this control system is very much the same as v) except there is the addition of the integral control term. The transfer function can be shown to be as follows:

$$\theta_o = \frac{(K_a S^3 + K_v S^2 + K_p S + K_I) C n \theta_r - T_d n S / J}{S^3 + C n K_v S^2 + C n K_p S + C n K_I} \qquad (6.16)$$

Again the ratio of the Computed inertia to actual inertia is represented by C. Full cancellation of the numerator with the denominator only occurs when this ratio is equal to 1. A summary of the characteristics of the two control system types of v) and vi) is shown in the table 6.2.

Type of Control	Order of Num.	Order of Den.	Type No with $C \neq 1$.	Test Input	Error to test input	Test Distur-bance	Error to test distur-bance
PD with acceleration feedforward	2	2	2	$\frac{\theta r}{S^3}$	$\frac{\theta r (1-C)}{CN K_p}$	$\frac{T_d}{S}$	$\frac{T_d}{K_p J C}$
PID with acceleration feedforward	3	3	3	$\frac{\theta r}{S^4}$	$\frac{\theta r (1-C)}{C K_I N}$	$\frac{T_d}{S^2}$	$\frac{T_d}{K_I J C}$

Table 6.2. Showing a summary of the characteristics of the two controller systems with acceleration feedforward.

6.3.1 Choice of Controller Coefficients

The object in choosing the controller coefficients is to obtain a suitable damping ratio and undamped natural frequency of response from the system. It is of benefit to increase the control system gain and hence the undamped natural frequency, since this minimizes the errors due to disturbance loads and tracking errors. However, the undamped natural frequency cannot be arbitrarily chosen since if it is set too high it may excite the structural resonant frequencies of the joints. The limitation on the setting of the undamped natural frequency will now be considered.

6.3.2 Structural frequency

The joint of a manipulator may be represented as an effective inertia J_{eff} mounted on a torsional shaft of stiffness K_{eff} as shown in Fig.6.6. The equation for this system is as follows:

$$- K_{eff}\theta = J_{eff}\ddot{\theta} \qquad (6.17)$$

The solution to this equation gives the frequency of oscillation which is :

$$\omega_s = \sqrt{\frac{K_{eff}}{J_{eff}}} \qquad (6.18)$$

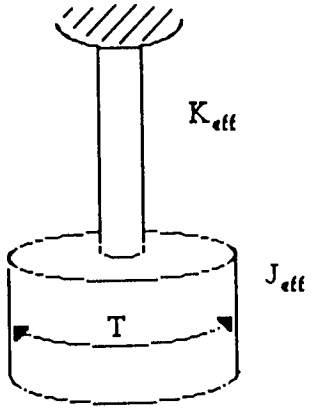


Figure 6.6. Diagram showing the stuctural representation of a joint of the robot manipulator.

It was shown by Book[30] that for conservative design a safety factor of 200% should be used. The undamped natural frequency ω_n of the closed loop control system must be set at no more than half the structural resonant frequency ω_s, for conservative design. It is obvious from equation (6.18) that with high inertial loads the natural resonant frequency goes down, which means that control gains must consequently be decreased to maintain the same performance.

6.3.3 Controller Gains

For control systems i) and ii) the characteristic equation is the standard second

order, of the form:

$$S^2 + 2\zeta\omega_n S + \omega_n^2 = 0 \qquad (6.19)$$

For instance if the choice of controller gains for the PD system is considered then then the characteristic equation of this system is :

$$S^2 + \frac{n K_v}{J} S + \frac{n K_p}{J} = 0 \qquad (6.20)$$

Comparing coefficients of equations (6.19) and (6.20) the undamped natural frequency can be equated to the structural frequency as:

$$\omega_n = \sqrt{\frac{n K_p}{J}} = \frac{\omega_s}{2} = \frac{\sqrt{\frac{K_{eff}}{J}}}{2} \qquad (6.21)$$

Rearranging in terms of the controller gain K_p gives the following expression:

$$K_p = \frac{K_{eff}}{4n} \qquad (6.21b)$$

An expression for the damping ratio can be found and can be shown to be:

$$\zeta = \frac{K_v}{2\sqrt{\frac{K_p J_{max}}{n}}} \qquad (6.22)$$

As shown in equation (6.22) the maximum inertia is used since this ensures that the damping ratio does not go below the set amount and increases if the inertia goes below the set amount. Rearranging (6.22) an expression for K_v is obtained:

$$K_v = 2\zeta\sqrt{\frac{K_p J_{max}}{n}} \qquad (6.23)$$

For control system v) this has the same order characteristic equation as i) and ii) but the value of the computed inertia J_c must be included in the calulation of the controller gain:

$$K_p = \frac{K_{eff}}{4 J_c n} \qquad (6.24)$$

where J_c is the computed inertia gain term.

An expression for the damping ratio can similarly be obtained:

$$\zeta_{min} = \frac{K_v}{2} \sqrt{\frac{J_{cmin} n}{J_{max} K_p}} \qquad (6.24b)$$

In order to ensure that the damping ratio does not become lower than the required for a fixed setting of K_v and J_c it is necessary to calculate K_v for when (6.24b) is a minimum. This occurs when J_c is a minimum and J is a maximum, hence for fixed value gains $J_c = J_{min}$ and K_v is :

$$K_v = \frac{2 \zeta_{min}}{\sqrt{\frac{J_{c\,min} n}{J_{max} K_p}}} \qquad (6.25)$$

For systems iii) and vi) the characteristic equation is third order, and system iii) is type 3. Since each term of the characteristic equation has a an adjustable controller coefficient in it (except the highest order term), it is possible to position the poles to an 'optimum' position. In Graham[31] they thoroughly investigated various criteria for optimum transient response. The clear winner was the Integral of Time by Absolute Error (ITAE) index. The application of this criterion results in the selction of standard forms for the characteristic equation coefficients. For a third order type 3 system the standard form is as follows, in terms of the numerator the minimum number of terms to satisfy the requirements of the type number are included in transfer function of (6.26):

$$\frac{\theta_o}{\theta_r} = \frac{2.97\,\omega_o S^2 + 4.94\,\omega_o^2 S + \omega_o^3}{S^3 + 2.97\,\omega_o S^2 + 4.94\,\omega_o^2 S + \omega_o^3} \qquad (6.26)$$

For system iii) the transfer function can be written in the same form (6.26) :

$$\frac{\theta_o}{\theta_r} = \frac{\frac{n K_v}{J} S^2 + \frac{n K_p}{J} S + \frac{n K_I}{J}}{S^3 + \frac{n K_v}{J} S^2 + \frac{n K_p}{J} S + \frac{n K_I}{J}} \qquad (6.27)$$

The coefficients of the two transfer functions can now be equated and rearranged in terms of the controller coefficients to give the following set of equations :

$$K_v = \frac{2.97\ \omega_0 J}{n} \qquad (6.28)$$

$$K_p = \frac{4.94\ \omega_0^2 J}{n} \qquad (6.29)$$

$$K_I = \frac{\omega_0^3 J}{n} \qquad (6.30)$$

However, the question arises how is the value of ω_o selected ?. The criterion for choosing the natural frequency of the system is the same as for the second order system i.e.:

$$\omega_n = \frac{\omega_s}{2} = \frac{\sqrt{\frac{K_{eff}}{J_{max}}}}{2} \qquad (6.31)$$

The relationship between ω_o and ω_n can be found by writing the ITAE index coefficients in the normalised form by letting

$$\lambda = \frac{S}{\omega_o}$$

This give the following normalised form of the ITAE index:

$$\lambda^3 + 2.97\lambda^2 + 4.94\lambda + 1 = 0 \qquad (6.32)$$

The roots for this equation are thus:

$r_1 = -1.3688 + j\,1.5589$
$r_2 = -1.3688 - j\,1.5589$
$r_3 = -0.2328$

This give a damping ratio of $\zeta = 0.66$ and a natural frequency $\omega_{nn}=2.075$ rad/sec(subscript $_{nn}$ refers to the normalised natural frequency). Since this is the natural frequency scaled by a factor of ω_o then the requred natural frequency is:

$$\omega_{n\ requred} = \frac{\omega_s}{2} = \omega_o\,\omega_{nn} \qquad (6.33)$$

Rearranging in terms of ω_o the required expression is obtained:

$$\omega_o = \frac{\sqrt{\frac{K_{eff}}{J_{max}}}}{2\,\omega_n} \qquad (6.34)$$

The value of ω_o may now be substituted into equations (6.28) to (6.30) to obtain the required controller coefficients.

For system vi) the characteristic equation is of the form:

$$(S^3 + C\,n\,K_v\,S^2 + C\,n\,K_p\,S + C\,n\,K_I)\,\theta_o = 0 \qquad (6.35)$$

Equating this to the ITAE standard form to find the controller coefficients:

$$K_v = \frac{2.97\,\omega_o}{C\,n} \qquad (6.36)$$

$$K_p = \frac{4.94\,\omega_o^{\,2}}{C\,n} \qquad (6.37)$$

$$K_I = \frac{\omega_o^{\,3}}{C\,n} \qquad (6.38)$$

Where $C = \frac{J_c}{J_{design}}$

If equations (6.36) to (6.38) are resubstituted into (6.35) the following equation is

obtained:

$$S^3 + \frac{J_c}{J_{act}}\left(\frac{2.97\,\omega_o J}{J_c}\right) S^2 + \frac{J_c}{J_{act}}\left(\frac{4.94\,\omega_o^2 J}{J_c}\right) S + \frac{J_c}{J_{act}}\left(\frac{\omega_o^3 J}{J_c}\right) \qquad (6.39)$$

The J_c's cancel throughout and the ω_o may be removed by letting $\lambda=S/\omega_o$. A root locus for the variation of $E=J/J_{act}$, for a range of values from 0.5 to 1.5 is shown in Fig.6.7.

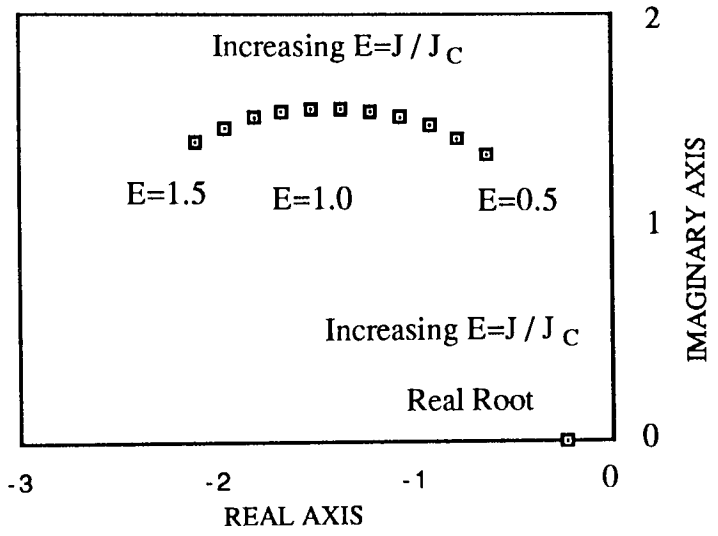


Fig.6.7. The root locus for the variation of $E=J/J_c$.

As seen in Fig.6.7 as the ratio E increases the poles go further away from the imaginary axis, therefore to ensure stability E must be chosen to be greater than or equal to one. If the design value of J is chosen to be J_{max}, this condition is satisfied and E is always greater than 1. Also $J_c=J_{max}$ to ensure the largest gain which makes $C=J_c/J_{act}=1$ at the design condition.

The structural resonances for the RTX manipulator joints were determined experimentally. For the path considered these correspond to the following:

Load(Kg)	Axis 1(rad/s)	Axis 2(rads/s)
0	9.48	11.65
10	8.26	10.00

The controller gains were determined based upon the different regimes outlined previously in the chapter.The table of controller gains for the six control strategies considered are shown in table 6.3 :

Control Load(kg)		K_{p1}	K_{p2}	K_{v1}	K_{v2}	K_{I1}	K_{I2}	J_{1c}	J_{2c}
1.	0	0.194	0.144	0.057	0.035				
	10	0.194	0.144	0.066	0.040				
2.	0	0.194	0.144	0.057	0.035				
	10	0.194	0.144	0.066	0.040				
3.	0	0.233	0.166	0.059	0.035	0.103	0.094		
	10	0.233	0.166	0.067	0.041	0.080	0.081		
4.	0	0.230	0.171	0.043	0.026	0.202	0.185		
	10	0.230	0.171	0.050	0.031	0.176	0.159		
5.	0	20152	15071	5955	3598			9.63E-6	9.71E-6
	10	16570	11063	5614	3083			11.71E-6	13.05E-6
6.	0	22574	17064	5945	3655	10433	9697	9.86E-6	9.71E-6
	10	17172	12693	5185	3152	6922	6221	12.96E-6	13.05E-6

Note. Control Types in column 1 refer to the following:

1. PD controller
2. P + derivative feedback
3. PID controller
4. PI + derivative feedback
5. PD + acceleration feedforward
6. PID + acceleration feedforward

Table 6.3. Showing the controller gains for the six independent axis control strategies.

6.3.4 Simulation results

The results of the simulation are shown in table 6.4(these were simulated using the simulation language ACSL[32]). The absolute angular error is tabulated in radians. Two design conditions are considered, designing for a zero load and designing for a 10 Kg load. The results for no load and 10Kg load are tabulated for each of these design conditions.

Controller	Design Load(Kg)	Applied Load(Kg)	Absolute Error(Rads) Axis 1	Axis 2
1. PD Control	0	0	0.0786	0.1040
	0	10	0.0869	0.1102
	10	10	0.0805	0.0986
	10	0	0.0725	0.0943
2. "Pd" Control.	0	0	0.2877	0.2901
	0	10	0.2921	0.2907
	10	10	0.3229	0.3066
	10	0	0.3244	0.3066
3. PID Control	0	0	0.0742	0.1104
	0	10	0.0819	0.1261
	10	10	0.0769	0.1090
	10	0	0.0694	0.0972
4. PI + derivative feedback.	0	0	0.1738	0.2046
	0	10	0.1810	0.2233
	10	10	0.1866	0.2260
	10	0	0.1795	0.2196
5. PD + acceleration feedforward	0	0	0.0028	0.0088
	0	10	0.0173	0.0301
	10	10	0.0141	0.0416
	10	0	0.0156	0.0398
6. PID + acceleration feedforward.	0	0	0.0041	0.0086
	0	10	0.0150	0.0293
	10	10	0.0197	0.0414
	10	0	0.0241	0.0402

Note. "Pd" is Proportional control with derivative feedback

Table 6.4. Showing the simulation results for the six independent axis control schemes.

The absolute error in radians is defined as the difference between the planned trajectory in joint space and the actual trajectory followed by the simulated robot manipulator system in joint space. The absolute error as tabulated in table 6.4 for

joints 1 and 2 show the maximum absolute error that occurs along the complete trajectory. This generally occurs following one of the corner points when the manipulator is required to accelerate rapidly to reach the next segment velocity.

In reference to table 6.4 of the simulation results it is seen that controller no. 2 (Pd control) performs extremely badly. This is expected since it is a type 1 system hence, it has an error to a velocity and acceleration input and an error to a disturbance input.

Controller 1. PD and 3. PID have comparable performance, this is considerably improved by the introduction of acceleration feedforward.

Generally when the fixed controller is designed about the higher load and that load is applied its performance is worse than when it is designed about zero load and a zero load is applied. This is because the controller gains are constrained by the reduction in the structural resonant frequency caused by the increase in inertial load.

The coupling terms in the dynamic equations are small when referred to the motor for the RTX since the gear ratios are so high. In order to compare controller types when the coupling terms are of more significance consider the system when it has optimum and direct drive ratios. However, it will be assumed that the structural resonant frequencies for each axis are the same for the RTX under a 10 Kg load. Of course this is not truly realistic since for a direct drive robot the structural stiffness is increased considerably. However, it is the effect of the significance of the coupling terms that are to be studied so all other factors are kept constant. In order to subtract initially the effect of the coupling terms the robot is modelled as an effective inertia only for each arm. For axis 1 the inertia will vary according to the position of axis 2 while the inertia of axis 2 is constant. The results of the simulation are shown for the reduced model of the robot and for the full model in table 6.5.

Controller	Model Simple (S) / Full (F)	Optimum Ratio Absolute Error(Rads) Axis 1	Optimum Ratio Absolute Error(Rads) Axis 2	Direct Drive Absolute Error(Rads) Axis 1	Direct Drive Absolute Error(Rads) Axis 2
1.PD Control	S	0.0896	0.1091	0.0895	0.1091
	F	0.0570	0.0812	0.0562	0.0828
2. PD + Acceleration Feedforward.	S	0.0245	0.0	0.0275	0.0
	F	0.0523.	0.138	0.0732	0.1880
3. PID Control	S	0.0861	0.1109	0.0860	0.1109.
	F	0.0540	0.0768	0.0532	0.0776
4.PID + Acceleration Feedforward	S	0.0099	0.0	0.0119	0.0
	F	0.0718	0.1470	0.0954	0.2018

Table 6.5. Showing the results for the four independent axis control stategies when the robot manipulator is directely driven.

As seen in table 6.5 comparing the results for the different controllers for the direct drive condition it is seen that the feedforward acceleration controllers perform a good deal worse than their corresponding counterparts which was contrary to the previous result for the RTX. The reason for this can be seen by comparing the results with those obtained for the simple model of an effective inertia. For the simple inertia case where the robot is modelled without the coupling terms the results predict that the feedforward acceleration controllers should perform better. This leads to the conclusion that when the non-linear terms in the dynamic model of the manipulator are significant i.e. the gear ratio is small, the feedforward acceleration controllers are severely affected by the coupling terms i.e.having low disturbance rejection properties. In fact it is seen that the straight PID control system performs marginally better than the PD control system. This can probably be accredited to the fact that the PID controller has greater disturbance rejection properties.

An unexpected observation from table 6.5 is that PD and PID control systems perform better for the coupled system than they they do for the simplified model

where there is no coupling present. This will be considered in detail why this occurs later in the chapter.

6.4 Linearisation of manipulator equations.

In order to understand in more detail the various control strategies the non-linear equations of the manipulator are linearised about a particular operating point. The dynamic equations written in state space format are as follows:

$$T = M(\theta)\,\ddot{\theta} + V(\theta,\dot{\theta}) + G(\theta) \qquad (6.40)$$

where $M(\theta)$ is the n*n inertia matrix
$V(\theta,\dot{\theta})$ is the n*1 vector of centrifugal and Coriolis terms
$G(\theta)$ is a n*1 vector of gravity terms

By writing the $V(\theta,\dot{\theta})$ in a different form the equations can be written in configuration space form:

$$T = M(\theta)\,\ddot{\theta} + B(\theta)\,[\dot{\theta}\dot{\theta}] + C(\theta)\,[\dot{\theta}^2] + G(\theta) \qquad (6.41)$$

Where $B(\theta)$ is a matrix of dimensions n*n(n-1)/2 of Coriolis coefficients.

$[\dot{\theta}\dot{\theta}]$ is an n[n-1]/2 *1 vector of joint velocity products given by :

$$[\dot{\theta}\dot{\theta}] = [\,\dot{\theta}_1\dot{\theta}_2, \dot{\theta}_1\dot{\theta}_3, \text{----------}, \dot{\theta}_{n-1}\dot{\theta}_n\,]^T$$

$C(\theta)$ is a matrix of centrifugal coefficients and $[\dot{\theta}^2]$ is a n*1 vector given by

$$[\,\dot{\theta}_1^2, \dot{\theta}_2^2, \text{----------}, \dot{\theta}_n^2\,]^T$$

In the configuration space format all of the coefficient matrices are a function of the manipulator position. This is a suitable form in which to linearise the equations since they will be linearised at a given position. For the manipulator under consideration the equations in configuration form are :

$$T = \begin{bmatrix} Jeff_1/n_1 & n_1 D_{12} \\ n_2 D_{12} & Jeff_2/n_2 \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} n_1 D_{112} \\ 0 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \dot{\theta}_2 \end{bmatrix}$$

$$+ \begin{bmatrix} 0 & n_1 D_{122} \\ n_2 D_{211} & 0 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1^2 \\ \dot{\theta}_2^2 \end{bmatrix} \quad (6.42)$$

At a particular configuration the non-linear terms are the Centrifugal and Coriolis terms, these are linearised as follows:

$$\frac{\partial D_{112} \dot{\theta}_1 \dot{\theta}_2}{\partial \theta_1} \Big| \dot{\theta}_{2i \text{ const.}} = D_{112} \dot{\theta}_{2i} D \quad (6.43)$$

$$\frac{\partial D_{112} \dot{\theta}_1 \dot{\theta}_2}{\partial \theta_2} \Big| \dot{\theta}_{1i \text{ const.}} = D_{112} \dot{\theta}_{1i} D \quad (6.44)$$

$$\frac{\partial D_{122} \dot{\theta}_2^2}{\partial \theta_2} \Big|_{i \text{ const.}} = 2 D_{122} \dot{\theta}_{2i} D \quad (6.45)$$

$$\frac{\partial D_{211} \dot{\theta}_1^2}{\partial \theta_1} \Big|_{i \text{ const.}} = 2 D_{211} \dot{\theta}_{1i} D \quad (6.46)$$

Letting

$$K_1 = D_{112} \dot{\theta}_{2i}$$
$$K_2 = D_{112} \dot{\theta}_{1i}$$
$$K_3 = 2 D_{122} \dot{\theta}_{2i}$$
$$K_4 = 2 D_{211} \dot{\theta}_{1i}$$

Writing the linearised equations in configuration space as follows:

$$\begin{bmatrix} T_{m1} \\ T_{m2} \end{bmatrix} = \begin{bmatrix} Jeff_1/n_1 & n_1 D_{12} \\ n_2 D_{12} & Jeff_2/n_2 \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} +$$

$$\begin{bmatrix} n_1 K_1 & n_1 (K_2 + K_3) \\ n_2 K_4 & 0 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} \quad (6.47)$$

Where T_{m1} and T_{m2} are the torques for joint 1 and 2 respectively referred to the motor side. The effective inertia at motor side for links 1 and 2 respectively are $Jeff_1$ and $Jeff_2$ and are equal to the following:

$Jeff_i = (Ja_i + Jm_i + n_i^2 D_{ii})$ where $Ja_i + Jm_i$ is the inertia of motor and fittings.

$n_i^2 D_{ii}$ is the referred link inertia.

Consider PD control where the multivariable block diagram is of the form:

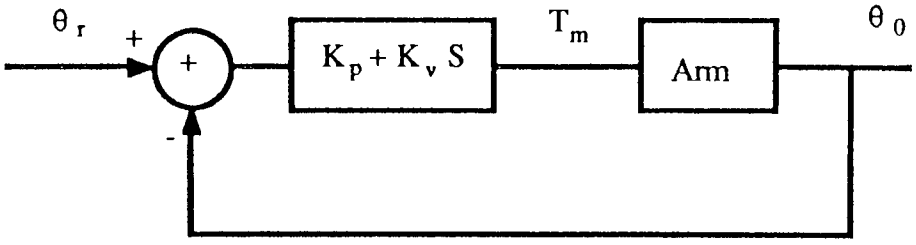


Figure 6.8. Block diagram for the PD control system.

The transfer function equations for PD control for both axes can be shown to be:

$$\theta_{r1} (K_{p1} + K_{v1} S) = \theta_1 (Jeff_1 / n_1 S^2 + (n_1 K_1 + K_{v1}) S + K_{p1})$$

$$+ \theta_2 (n_1 D_{12} S^2 + n_1 (K_2 + K_3) S) \qquad (6.48)$$

$$\theta_{r2} (K_{p2} + K_{v2} S) = \theta_2 (Jeff_2 / n_2 S^2 + K_{v2} S + K_{p2})$$

$$+ \theta_1 (n_2 D_{12} S^2 + n_2 K_4 S) \qquad (6.49)$$

Similarly the transfer function equations can be found for PID control and are shown to be :

$$\theta_{r1}(K_{v1}S^2 + K_{p1}S + K_{I1}) = \theta_1(Jeff_1/n_1S^3 + (n_1K_1 + K_{v1})S^2 + K_{p1}S + K_{I1})$$

$$+ \theta_2(n_1D_{12}S^3 + n_1(K_2 + K_3)S^2) \qquad (6.50)$$

$$\theta_{r2}(K_{v2}S^2 + K_{p2}S + K_{I2}) = \theta_2(Jeff_2/n_2S^3 + K_{v2}S^2 + K_{p2}S + K_{I2})$$

$$+ \theta_1(n_2D_{12}S^3 + n_1K_4S^2) \qquad (6.51)$$

The initial objective is to verify the linearised model of the robot manipulator. The model of the manipulator with PD controller will be considered. The model of this system is represented in state-space format in the control design package CTRL 'C'[33]. Initially the analogue block diagram for the linked PD system is drawn up as follows:

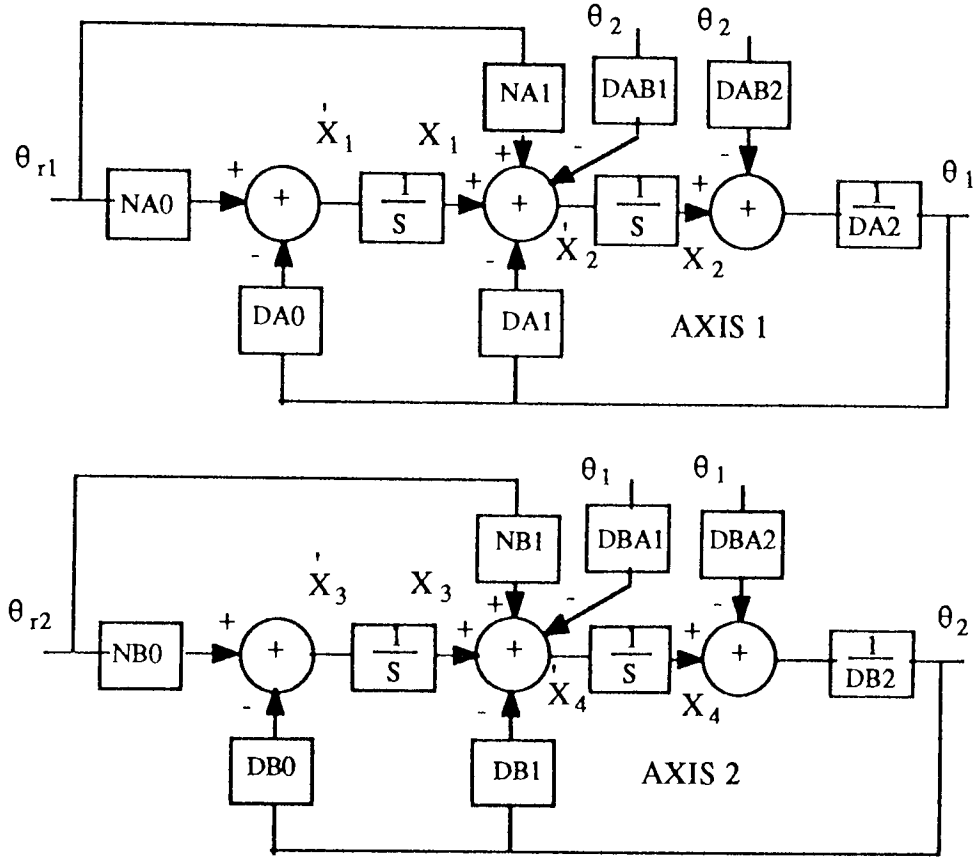


Figure 6.9. Analogue block diagram representing the dynamically coupled robot manipulator with PD control.

For axis 1 the coefficients are:

Numerator coefficients

$NA1 = K_{v1}$

$NA0 = K_{p1}$

Denominator coefficients

$DA2 = Jeff_1 / n_1$

$DA1 = n_1 K_1 + K_{v1}$

$DA0 = K_{p1}$

$DAB2 = n_1 D_{12}$

$DAB1 = n_1 (K_2 + K_3)$

For axis 2 the coefficients are:

Numerator coefficients

$NB1 = K_{v2}$

$NB0 = K_{p2}$

Denominator coefficients

$DB2 = Jeff_2 / n_2$

$DB1 = K_{v2}$

$DB0 = K_{p2}$

$DBA2 = n_2 D_{12}$

$DAB1 = n_2 K_4$

$X_1, X_2, X_3, X_4, X_5, X_6$ are arbitrary states as shown in the block diagram.

The A,B,C and D state space matrices are found to be :

$$A = \begin{bmatrix} 0 & -\frac{DA0\ DB2}{DCOF} & 0 & \frac{DA0\ DAB2}{DCOF} \\ 1 & \frac{DAB1\ DBA2 - DA1\ DB2}{DCOF} & 0 & \frac{DA1\ DAB2 - DAB1\ DA2}{DCOF} \\ 0 & \frac{DB0\ DBA2}{DCOF} & 0 & -\frac{DB0\ DA2}{DCOF} \\ 0 & \frac{DB1\ DBA2 - DBA1\ DB2}{DCOF} & 1 & \frac{DBA1\ DAB2 - DB1\ DA2}{DCOF} \end{bmatrix}$$

$$B = \begin{bmatrix} NA0 & 0 \\ NA1 & 0 \\ 0 & NB0 \\ 0 & NB1 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & \frac{DB2}{DCOF} & 0 & -\frac{DAB2}{DCOF} \\ 0 & -\frac{DBA2}{DCOF} & 0 & \frac{DA2}{DCOF} \end{bmatrix}$$

$$D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

Where DCOF= $Jeff_1/n_1 * Jeff_2/n_2 - n_1 n_2 D_{12}^2$

6.4.1 Verification of Linearised Model

The model of the robot system under PD control is represented in state space form using CTRL'C'. The model is configuration dependent and the desired configuration can be chosen i.e. the postion of link two θ_2. The non-linear terms were linearised about the velocities of joint 1 and joint 2 these are chosen to suit the required conditions considered. A full non-linear model of the manipulator with PD control is represented in ACSL and used to compare with the linearised model.

Initially a step input to axis 1, of magnitude 0.5 was considered with zero input into axis 2. The actual velocities of axis 1 and 2 are plotted from the results for the full non-linear simulation as shown in Fig.6.10.

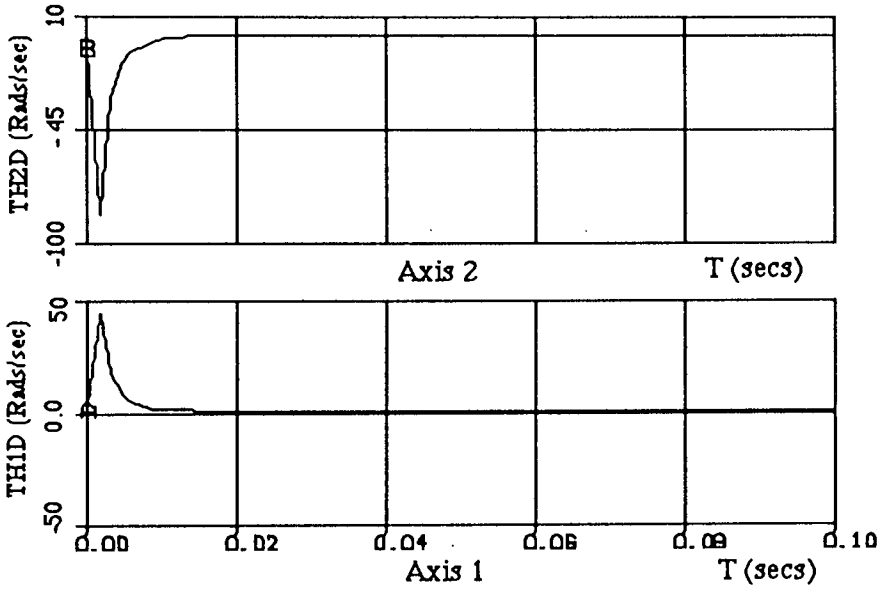


Figure 6.10. Plot of Joint velocities versus time for the PD control system.

A step demand in axis 1 means an instantaneous velocity change in axis 1. However, the final velocity is zero, so the linearisation velocity is chosen as zero for axis 1. The sudden torque demand in 1 causes link 2 to accelerate in the opposite direction due to the coupling effect. The equation of motion for axis 2, ignoring other terms, is as follows:

$$T_2 - D_{12}\ddot{\theta}_1 = -D_{12}\ddot{\theta}_1 = D_{22}\ddot{\theta}_2$$

where T_2 is initially zero

Hence, the torque accelerating axis 2 is that due to the motion of axis 1 and is negative which causes link 1 to accelerate in the opposite direction to axis 1. The final velocity of axis 2 is zero so the linearisation point is chosen as zero. The configuration operating point is zero being the final settling point of θ_2. The linearised response plotted with the full non-linear response is shown in fig.6.11.

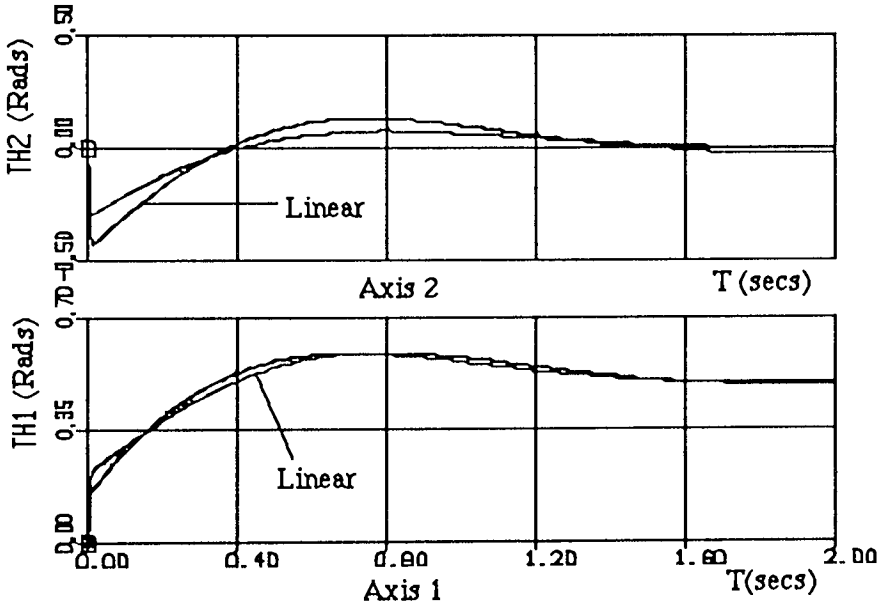


Figure 6.11. Plot of the linearised response with the full nonlinear response to a step input of 0.5 in axis 1.

As seen in Fig.6.11 the linearised response is shown to have an initial response that is greater than the non-linear response. The general differences between the two plots are small throughout the period of the simulation. The reason the differences occur is due to the choice of the operating points for the configuration and the velocities. In the graph for axis 2 the non-linear system appears to be more damped than the linearised system. At the initial position the instantaneous equation for axis 2 is:

$$T_{2c} - D_{12}\ddot{\theta}_1 + D_{211}\dot{\theta}_1^2 = D_{22}\ddot{\theta}_2$$

The initial torque available for acceleration of link 2 is caused by the coupling torque.This is reduced by the centrifugal torque produced by the motion of link 1 but acting in link 2 which is positive initially because:

$\sin(-\theta_2) = -\sin(\theta_2)$.

This is not taken into account in the linearised model since it was linearised about zero velocity for both axes. Similar discrepancies occur in axis 1 due to the choice of the operating points for the velocities. For axis 2 the instantaneous equation is of the form:

$$T_{1c} - D_{12}\ddot{\theta}_2 - D_{122}\dot{\theta}_1^2 + D_{112}\dot{\theta}_1\dot{\theta}_2 = D_{11}\ddot{\theta}_1$$

A similar analysis could be carried out for step input to axis 2 and zero input to axis 2. This can be shown to yield very similar close results in the comparison between the linearised model and the non linear model as shown in fig.6.12.

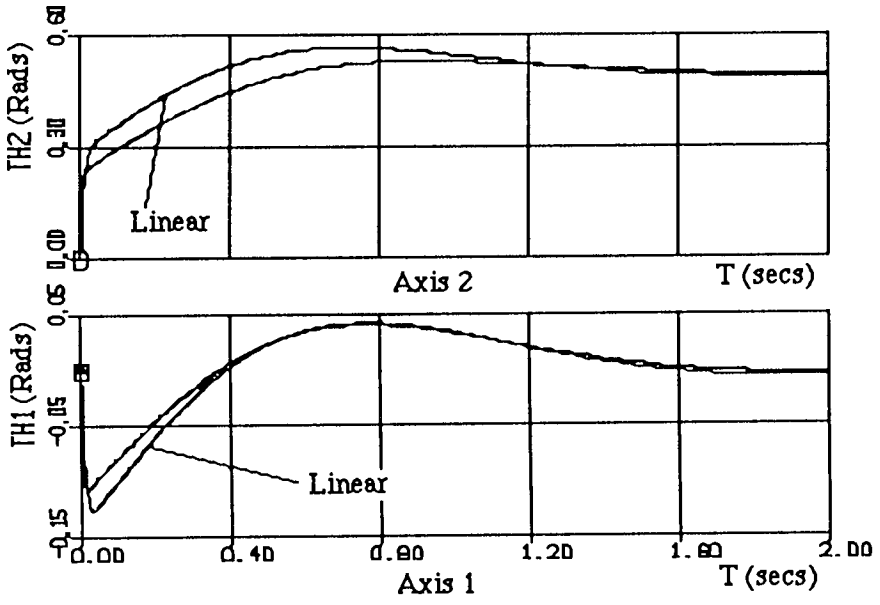


Figure 6.12. Plot of the linearised response with the full nonlinear response to a step input of 0.5 in axis 2.

Next a ramp input in axis 1 is considered with zero input in axis 2. First the non-linear velocity response for both axes is plotted as a guide for deciding the operating points for the linearised coefficients.The two joint velocities are shown plotted in fig.6.13:

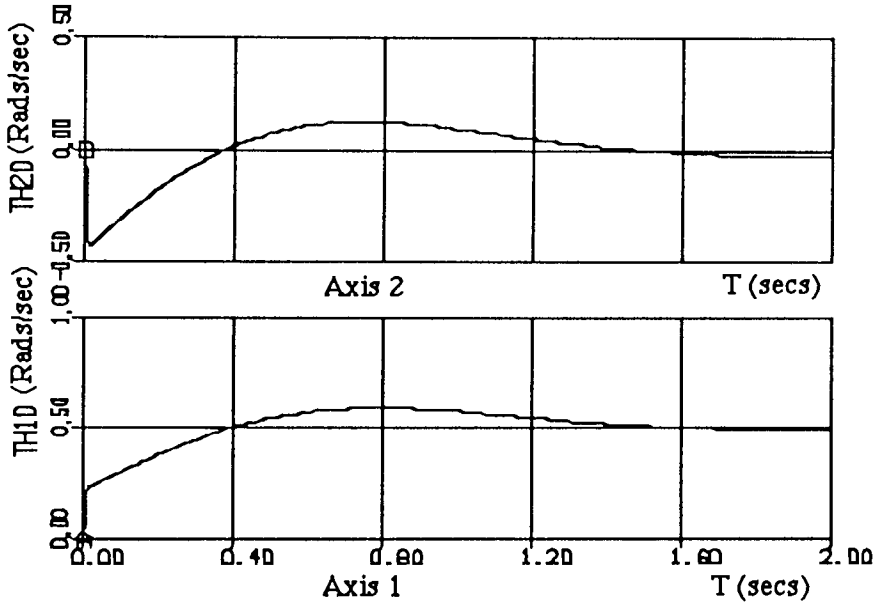


Figure 6.13. Plot of Joint velocities versus time for the full nonlinear system.

The choice of the velocity dependent linearisation coefficients is straight forward for axis 1. The magnitude of the ramp input for axis 1 is 0.5 rads/sec, so this is the chosen operating point. For axis 2 the response shown in Fig.6.13 is seen to to settle at zero, the chosen operating point is therefore zero. The configuration dependent operating point θ_2 is also chosen as zero. The position response for the linearised system, plotted for comparison with the non-linear system, showed no identifiable difference and is hence not plotted, however the response is shown plotted against the reference input in fig.6.14:

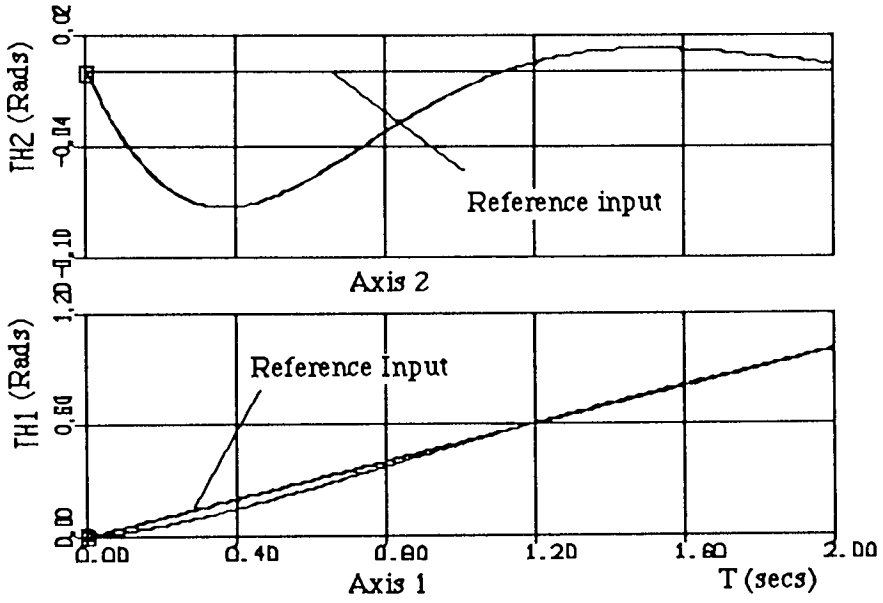


Figure 6.14. Plot of the full nonlinear response to a ramp input of 0.5 in axis 1.

Similar close matching of models for the case of a ramp input to axis 2 were obtained.

6.4.2 Transfer function and characteristic equation

The system transfer function defined in terms of the state-space matrices is :

$$f(s) = C^t [S\,I - A\,]^{-1} B\;U(S) \qquad (6.52)$$

In order to determine the transfer function symbolically the equations of the system were reformed for input into the Aston package which will determine the transfer function of the system symbolically. The Aston package is a computer aided control system design package (as described by Firoozian[34]). It allows the study of the stability, transient response, frequency response and steady state behaviour of the system to be obtained.

The initial design of the controller coefficients was based upon the assumption that the non-linear coupling terms of the dynamic equations of the manipulator could be ignored, so that each axis if the manipulator is represented by an effective inertia, with no interaction between the two axes. The characteristic equation for each axis is thus a second order equation for which the coefficients can be suitably chosen. In reality the characteristic equation is of fourth order and represents the

coupled system as shown in equation (6.53).

$$
\begin{aligned}
&(Jeff_1\ Jeff_2/n_1\ n_2 - n_1\ n_2\ D_{12}^2)\ S^4 \\
&+ (K_{v2}Jeff_1/n_1+Jeff_2/n_2(n_1K_1+ K_{v1})-n_1\ n_2\ D_{12}K_4 - n_1\ n_2(K_2+K_3)D_{12})S^3 \\
&+ (K_{p1}Jeff_2/n_2+K_{v2}\ (n_1K_1+ K_{v1})+ K_{p2}Jeff_1/n_1- n_1\ n_2(K_2+K_3)K_4)\ S^2 \\
&+ (K_{v2}\ K_{p1} + K_{p2}\ (n_1K_1+K_{v1}))\ S^1 \\
&+ (K_{p1}\ K_{p2})\ S^0 = 0 \qquad (6.53)
\end{aligned}
$$

The numerator coefficients can be expressed as an input-output matrix of the form:

$$
\begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} \begin{bmatrix} \theta_{r1} \\ \theta_{r2} \end{bmatrix} \qquad (6.54)
$$

Where the numerator coefficients are defined as :

$$
\begin{aligned}
G_{11} =[\ (\ K_{v1}\ Jeff_2/n_2)\ S^3 + (\ K_{v1}K_{v2} + Jeff_2/n_2\ K_{p1})\ S^2 \\
+ (\ K_{v2}K_{p1} + K_{v1}K_{p2}\ )\ S^1 + (\ K_{p1}\ K_{p2}\ )\ S^0\ ]
\end{aligned}
$$

$$
\begin{aligned}
G_{12}= -\ [\ (n_1D_{12}K_{v2})\ S^3 + (\ n_1D_{12}K_{p2}+ n_1(K_2+ K_3)\ K_{v2}\ )\ S^2 \\
+ (\ n_1(\ K_2+ K_3)\ K_{p2})\ S^1\ ]
\end{aligned}
$$

$$
\begin{aligned}
G_{21}= -\ [\ (\ n_2D_{12}K_{v1})\ S^3 + (\ K_{v1}\ n_2\ K_4 + K_{p1}\ n_2\ D_{12}\ )\ S^2 \\
+ (\ n_2K_{p1}K_4\ )\ S^1\ ]
\end{aligned}
$$

$$
\begin{aligned}
G_{22}= [\ (\ K_{v2}\ Jeff_1/n_1)\ S^3 + (\ K_{v2}(n_1K_1+ K_{v1}) + K_{p2}Jeff_1/n_1\ )\ S^2 \\
+ (\ K_{v2}K_{p1} + K_{p2}(\ n_1K_1 + K_{v1}))\ S^1 + (\ K_{p1}\ K_{p2}\ )\ S^0\ ]
\end{aligned}
$$

6.4.3 The effect of coupling on the dynamic response

The effect of inertial coupling and the other nonlinear coupling terms on the two axes of the manipulator can be seen by evaluating the Eigen values of the [S I - A] matrix. In fig.6.16 are plotted the roots of each axis in their design position where all coupling terms are ignored. However, if inertial coupling is taken into account with the terms arising from the centrifugal and Coriolis set to zero the actual position of the roots are quite far from this point as seen in fig.6.16. The roots consist of a pair of complex roots at -1.6 + j 2.8 and -1.6 -j 2.8 with a ω_n=3.22 rads/sec and a ζ=0.495 and two real roots at -3.37 and -111.45.

The effect of the other non-linear terms will now be considered.

1. Variation of K_3

Consider the case when joint velocity 1 is zero,thus K_1,K_2 and K_4 are equal to zero,and K_3 is :

$$K_3 = 2 D_{122} \dot{\theta}_{2i}$$

The D_{122} coefficient which is configuration dependent will be calculated at the design condition which corresponds to the maximum inertia condition.This occurs at θ_2=5.7 rads. A reasonable range of velocities for joint 2 that will be considered are + or - 25 rads/sec.

The root locus for the variation of K_3 can be drawn unscaled as shown in fig.6.14a. The roots consist of a pair of high frequency roots and low frequency roots as shown in fig.6.14a. The high frequency roots are real at K_3=–20 and become complex at K_3=+11. The high frequency roots move into the right hand side of the Argand diagram at K_3=+20 and are equal to +0.45 + or – j27.61. The correctly scaled root locus is shown in Fig.6.15. There also exists a set of low frequency roots as shown in Fig.6.14a and to scale in Fig.6.16. At K_3=–20 the low frequency roots are near the imaginary axis and are equal to –0.57 + or –j2.79. They would dominate the response since they are the nearest to the

imaginary axis. This corresponds to a large negative joint 2 velocity. However, as K_3 becomes large and positive the low frequency roots move towards the real axis and the high frequency roots now dominate as they are closer to the imaginary axis than the low frequency roots.

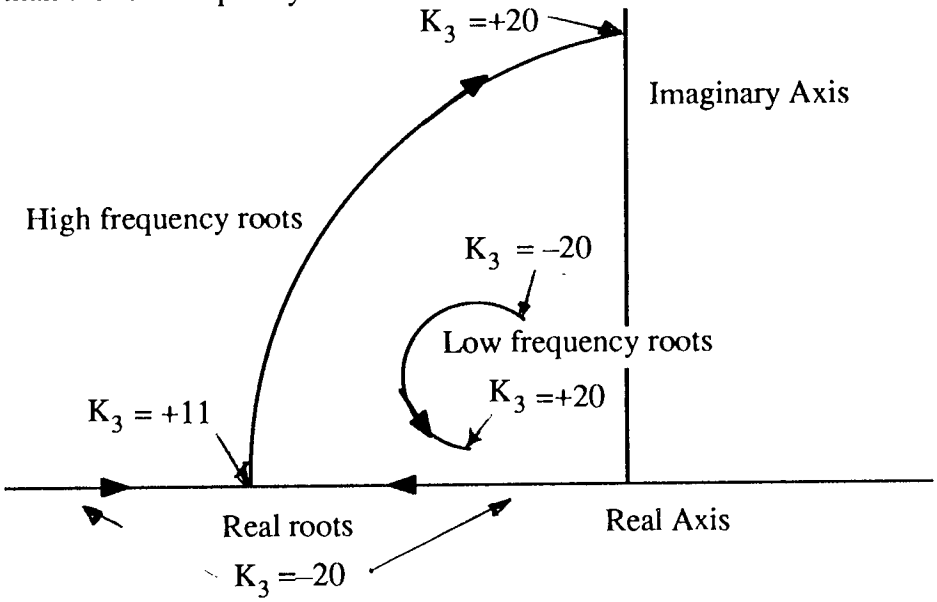


Fig.6.14a. Showing unscaled root locus for the variation of K_3 for the range K_3=–20 to +20.

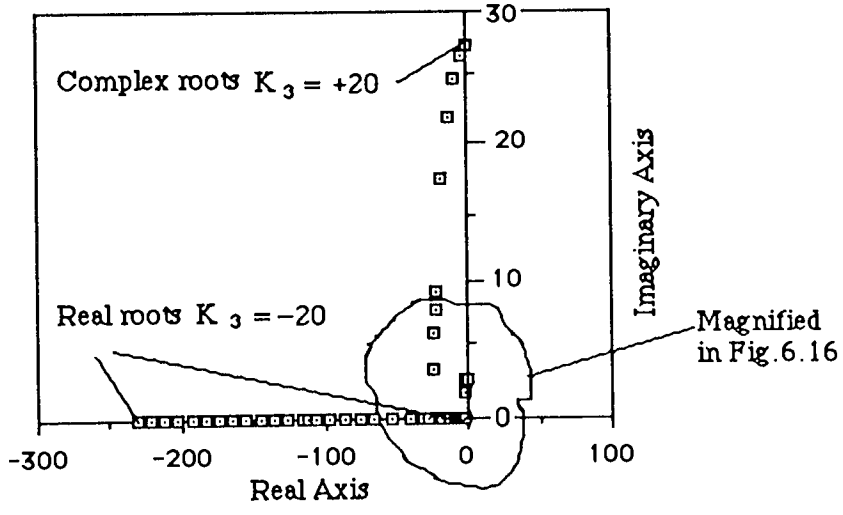


Figure 6.15. The root locus for the variation of K_3 for the range K_3=–20 to +20.

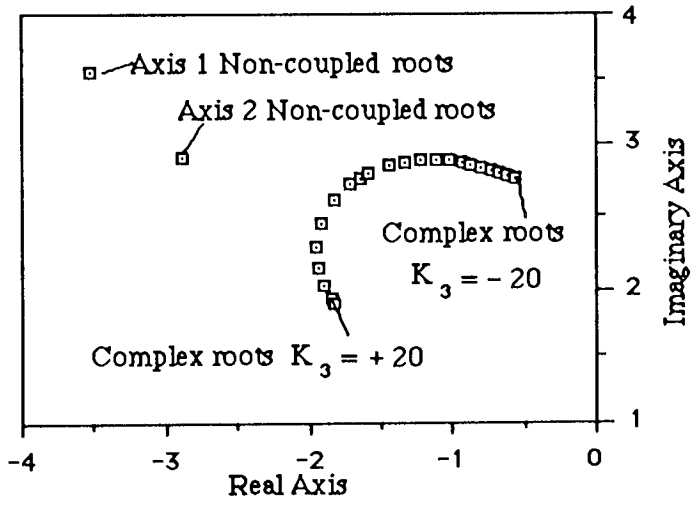


Fig.6.16. The root locus for the variation of K_3 for the range K_3=–20 to +20, showing the low frequency roots.

2. Variation of K_4

Consider the situation when joint 2 velocity is equal to zero, the coefficients K_1, K_2 and K_3 will equal zero. The value of K_4 is equal to the following:

$$K_4 = 2\, D_{211}\, \dot{\theta}_{1i}$$

However, referring to the C.E of (6.53) it is seen that the only terms involving K_4, are in the third order term of the C.E , when the other K's are set to zero, as follows:

$$(K_{v2}Jeff_1/N_1+Jeff_2/N_2(N_1K_1+ K_{v1})-N_1 N_2 D_{12}K_4 - N_1 N_2(K_2+K_3)D_{12})S^3$$

The specific term involving K_4 in the third order coefficient is $-N_1 N_2 D_{12}K_4$ and the specific term involving K_3 is $-N_1 N_2 D_{12}K_3$. They are both multiplied by the same terms however, the values of K_3 and K_4 are of different magnitude and are functions of θ_{2i} and θ_{1i} respectively.

$$K_3 = 2\, D_{122}\, \dot{\theta}_{2i}$$

For this robot manipulator $D_{211}=-D_{122}$, so if the range of θ_1 that we look at the root locus for the variation of K_4 is chosen to be the same range of θ_2 that we look at the root locus for the variation of K_3 they will be the same. The only difference occurs due to the sign of the dynamic coefficients being opposite. This can be seen in table 6.6, later in the chapter. Therefore the root locus for this situation need not be drawn.

3. Variation of K_1

K_1 arises from the linearisation of the Coriolis term in the dynamic equation for axis 1, as seen previously in equation (6.43). The value of K_1 is dependent upon the angular velocity of link 2 and the contribution to the torque T_{m1} depends in turn upon the angular velocity of link 1, as seen in equation equations (6.47). It is zero if either of the joint velocities are zero. The root locus for the variation of K_1 is seen in Fig.6.17 with the other K's set to zero, although this isn't a realistic situation that could occur.

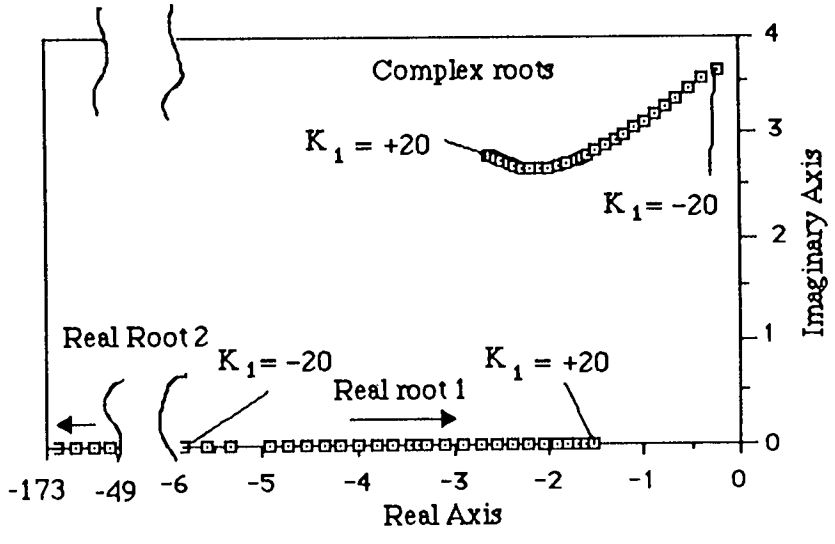


Figure 6.17. The root locus for the variation K_1 for a range of K_1=–20 to +20.

As seen in fig.6.17, the complex roots tend towards the right hand side of the argand diagram for K_1< -20 and the system would become unstable.The real roots move towards each other to -5.83 and -49.86 for K_1=–20.

The root locus for the system is more complex than the simplified cases considered so far, since the linearsed coefficients will all have finite values at a given instance of time in the trajectory, that are functions of the two joint velocities and the configuration of joint 2. If the configuration at the design point is considered (θ_2=5.7) i.e. $\sin\theta_2$ is negative then the sign of the linearised coefficients are as shown in table 6.6 for the different permutations of the joint velocities.

$\dot{\theta}_1$	$\dot{\theta}_2$	K_1	K_2	K_3	K_4
+	+	+	+	+	-
+	-	-	+	-	-
-	+	+	-	+	+
-	-	-	-	-	+
+	0	0	+	0	+
-	0	0	-	0	-
0	+	+	0	+	0
0	-	-	0	-	0

Table 6.6. Showing the sign of the linearised coefficients for the different permutations of the joint velocities.

The range of magnitude of the joint velocities take will depend upon the point considered on a particular trajectory. The root locus is now plotted for a range of joint velocities as seen in fig.6.18. Lines of constant axis 2 velocity are drawn in the vertical direction and curves of constant axis 1 velocity are drawn approximately horizontal. This root locus shows the high frequency roots which occur at fairly high magnitudes of joint velocities. For instance, with a joint 1 velocity of 20 rads/sec and a joint 2 velocity of 55 rads/sec the roots are in the right hand quadrant of the Argand diagram indicating that the system is unstable at these joint velocities. In fact this probably applies to at any axis 1 velocity.

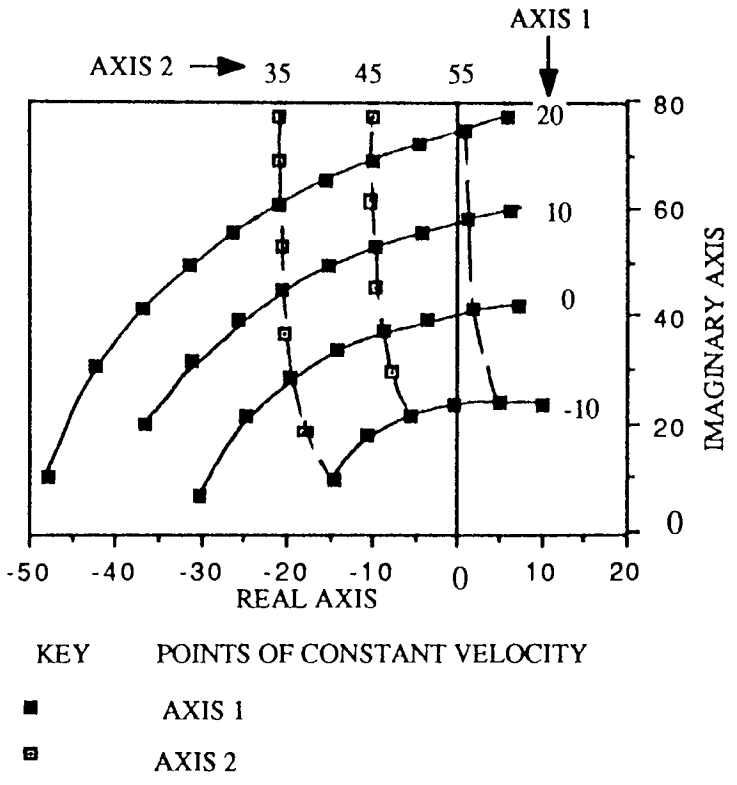
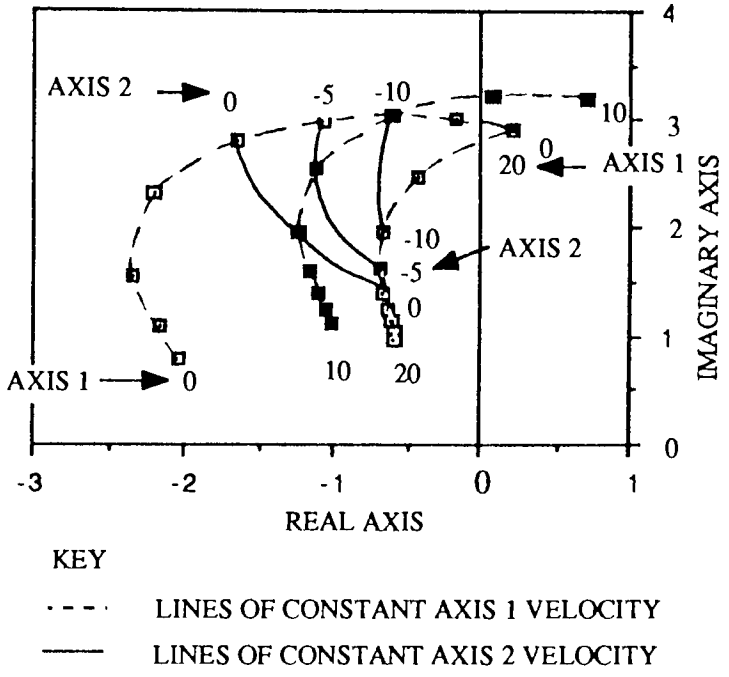


Figure 6.18. Root locus for the variation of the two joint velocities showing the high frequency roots.

It is also possible to draw the root locus for the variation of the joint velocities showing the lower frequency roots as seen in fig.6.19. Again, lines of constant axis velocity are drawn on the root locus plot. For instance, with a joint 1 velocity of 10 rads/sec and a joint 2 velocity of -15 rads/sec the roots are in the right hand quadrant of the Argand diagram indicating instability at these joint velocities. The question should be asked, are these velocities likely to occur in a trajectory ?. In the trajectory considered with the joint velocities shown plotted in fig.5.11 and fig.5.12 of chapter 5, for a maximum Cartesian velocity of 300 mm/sec and acceleration of 2000 mm/sec^2 the maximum joint velocities are 1.28 rads/sec and 1.55 rads/sec for joints 1 and 2 respectively. However, a more realistic Cartesian velocity of commercial robots is 6000mm/sec. In order to retain the scaling of the path the Cartesian acceleration must be increased to 800000 mm/sec^2. This is a twenty times increase in the Cartesian velocity and a four hundred times increase in the Cartesian acceleration. This causes approximately, a twenty times increase in the joint velocities to 26 rads/sec and 30 rads/sec for joints 1 and 2 respectively. So obviously the possibility of reaching a point on the argand diagram in or near the right hand quadrant is highly likely at those Cartesian velocities.



KEY

- - - - LINES OF CONSTANT AXIS 1 VELOCITY

—— LINES OF CONSTANT AXIS 2 VELOCITY

Figure 6.19. Root locus for the variation of the two joint velocities showing the low frequency roots.

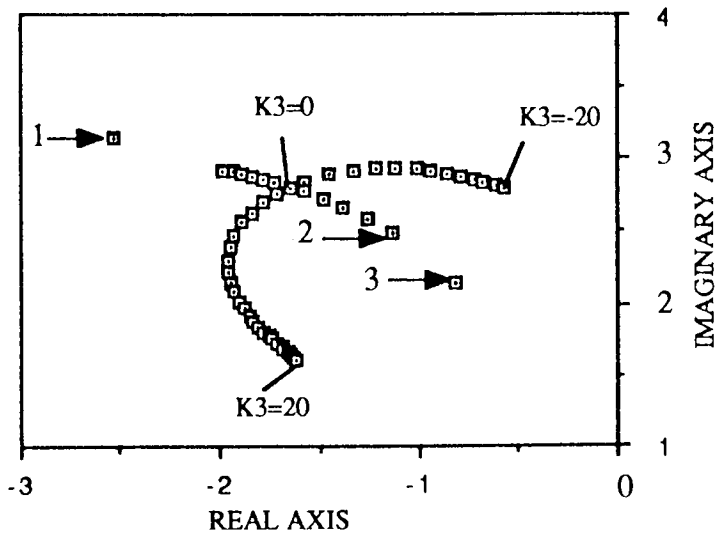
The root locus diagrams are drawn for the case where the joints are driven directly. However, if the joints are driven through a gear box the effect of higher joint velocities is reduced. This is seen by examining the characteristic equation of the system (equation 6.53) where the linearised coefficients are all multiplied by a combination of the gear box ratio. Having considered the PD control system, the effect of introducing acceleration feedforward to the system will be considered.

PD + Acceleration feedforward

Initially the characteristic equation of the control system is obtained in symbolic form as shown below:

$$
\begin{aligned}
&(Jeff_1 Jeff_2/n_1 n_2 - n_1 n_2 D_{12}^2) S^4 \\
&+ (J_{c1} K_{v2} Jeff_1/n_1 + Jeff_2/n_2 (n_1 K_1 + J_{c1} K_{v1}) \\
&\qquad - n_1 n_2 D_{12} K_4 - n_1 n_2 (K_2 + K_3) D_{12}) S^3 \\
&+ (J_{c1} K_{p1} Jeff_2/n_2 + J_{c2} K_{v2} (n_1 K_1 + J_{c1} K_{v1}) \\
&\qquad + J_{c2} K_{p2} Jeff_1/n_1 - n_1 n_2 (K_2 + K_3) K_4) S^2 \\
&+ (J_{c2} K_{v2} J_{c1} K_{p1} + J_{c2} K_{p2} (n_1 K_1 + J_{c1} K_{v1})) S^1 \\
&+ (J_{c1} K_{p1} J_{c2} K_{p2}) S^0 \qquad (6.55)
\end{aligned}
$$

As seen in equation (6.55) above, this characteristic equation is very similar to the simple PD system of equation (6.53) except that terms involving the controller coefficients K_{pi} and K_{vi} are multiplied by the computed inertia terms J_{c1} and J_{c2} for axis 1 and 2 respectively. How accurately the computed inertia terms reflect the actual inertia values will influence the stability of the system. This can be seen by superimposing the variation in the computed inertia on the root locus for the variation in K_3 root locus, fig.6.20.



Key to root positions:

1. J_{C1} and J_{C2} are 50% greater than the actual inertia.
2. J_{C2} is 50% less than actual inertia
3. J_{C1} and J_{C2} are 50% less than the actual inertia.

Figure 6.20. Root locus for the variation in computed inertia superimposed on the variation in K_3 for the PD + acceleration feedforward control system.

As seen in fig.6.20 the roots are plotted for J_{c2} plus and minus 50% of the actual inertia at K_3=0. When the computed inertia is less than actual inertia the roots are seen to move towards the right hand quadrant of the Argand diagram. This situation is equivalent to a large load being picked up by the manipulator that is far larger than expected. The roots are plotted for the cumulative effect of both J_{c1} and J_{c2}, e.g.point 3 in fig.6.20 is when J_{C1} and J_{C2} are 50% smaller than the actual inertia's. The roots are seen to move towards the right hand quadrant of the Argand diagram in this situation.

The numerator coefficients of the PD + acceleration feedforward control system, as defined previously by equation(6.54), are found to be as follows:

$$G_{11}= [\ (K_{a1}J_{c1}Jeff_2/n_2)\ S^4 + (J_{c2}K_{v2}Jeff_2/n_2+J_{c2}K_{v2}K_{a1}J_{c1})\ S^3$$
$$+ (\ J_{c1}K_{v1}J_{c2}K_{v2}+Jeff_2/n_2\ J_{c1}K_{p1}+\ J_{c2}K_{p2}K_{a1}J_{c1})\ S^2$$
$$+ (\ J_{c2}K_{v2}J_{c1}K_{p1}+\ J_{c1}K_{v1}J_{c2}K_{p2})\ S^1 + (\ J_{c1}K_{p1}J_{c2}K_{p2})S^0\ ]$$

$$G_{12}= -[(K_{a2}J_{c2}n_1D_{12}) S^4 + (n_1D_{12}J_{c2}K_{v2}+n_1(K_2+K_3)K_{a2}J_{c2}) S^3$$
$$+ (n_1D_{12}J_{c2}K_{p2}+n_1(K_2+K_3)J_{c2}K_{v2}) S^2$$
$$+ (n_1(K_2+K_3)J_{c2}K_{p2}) S^1$$

$$G_{21}= -[(n_2D_{12}K_{a1}J_{c1}) S^4 + (n_2D_{12}J_{c1}K_{v1}+K_{a1}J_{c1}n_2K_4) S^3$$
$$+ (J_{c1}K_{v1}n_2K_4+J_{c1}K_{p1}n_2D_{12}) S^2 + (n_2J_{c1}K_{p1}K_4) S^1]$$

$$G_{22}= [(K_{a2}J_{c2}Jeff_1/n_1) S^4 + (J_{c2}K_{v2}Jeff_1/n_1+K_{a2}J_{c2}(J_{c1}K_{v1}+n_1K_1) S^3$$
$$+ (J_{c2}K_{v2}(J_{c1}K_{v1}+n_1K_1) + J_{c2}K_{p2}Jeff_1/n_1 + K_{a2}J_{c2}J_{c1}K_{p1}) S^2$$
$$+ (J_{c2}K_{v2}J_{c1}K_{p1}+ J_{c2}K_{p2}(J_{c1}K_{v1}+n_1K_1) S^1$$
$$+ (J_{c1}K_{p1}J_{c2}K_{p2}) S^0] \qquad -(6.56)$$

Where $K_{a1}=1/n_1$ and $K_{a2}=1/n_2$ i.e. the acceleration gains.

As seen in equation(6.56) the numerator coefficients of the transfer function are the same order as the denominator characteristic equation however, there is no cancellation of the numerator with the denominator due to the presence of the coupling terms.

PID Control

For the PID closed loop system the characteristic equation in symbolic form is as follows:

$$(Jeff_1Jeff_2/n_1n_2 - n_1n_2D_{12}^2) S^6$$
$$+ (K_{v2}Jeff_1/n_1+Jeff_2/n_2(n_1K_1+K_{v1}) - n_1n_2D_{12}K_4 - n_1n_2(K_2+K_3)D_{12}) S^5$$
$$+ (K_{p1}Jeff_2/n_2+K_{v2}(n_1K_1+K_{v1})+K_{p2}Jeff_1/n_1- n_1n_2 (K_2+K_3)K_4) S^4$$
$$+ (K_{v2}K_{p1}+K_{p2}(n_1K_1+K_{v1})+K_{I1}Jeff_2/n_2+K_{I2}Jeff_1/n_1) S^3$$
$$+ (K_{p1}K_{p2}+K_{v2}K_{I1}+K_{I2}(n_1K_1+K_{v1}) S^2$$
$$+ (K_{I1}K_{p2}+K_{p1}K_{I2}) S^1 + (K_{I1}K_{I2}) S^0 \qquad (6.57)$$

If the characteristic equation is compared with that of the PD system of equation(6.53) it is seen that terms S^6, S^5 and S^4 are the same as coefficients S^4, S^3 and S^2 of the PD system. The other terms of the characteristic equation are those arising from the integral coefficient gains K_{I1} and K_{I2}.

If a similar root locus analysis is carried out for the PID control system as for the PD system, first the case when joint velocity 1 is zero is considered, hence K_1,K_2 and K_4 are zero. The root locus is drawn for the variation of K_3 as shown in Fig.6.21.

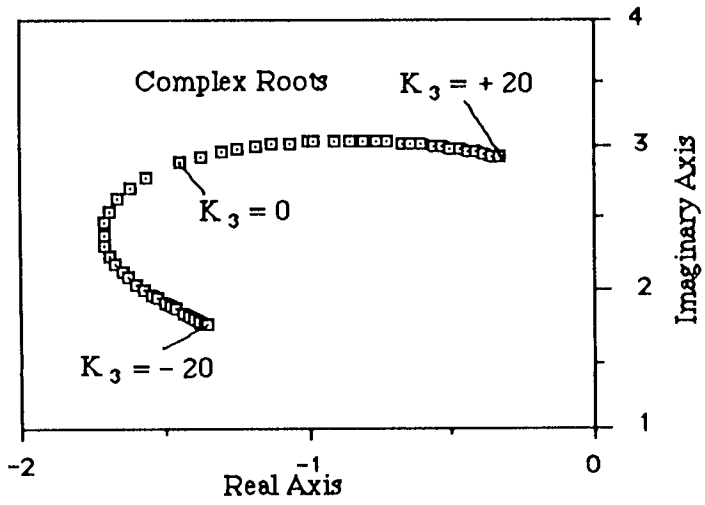


Figure 6.21. Root locus for the variation of K_3 in the range K_3=–20 to +20, showing the low frequency roots.

If Fig.6.21 is compared to Fig.6.15 for the PD system, this is very similar in shape for the range of K_3=-20 to + 20. The plot for the high frequency roots is also drawn as seen in Fig.6.22:

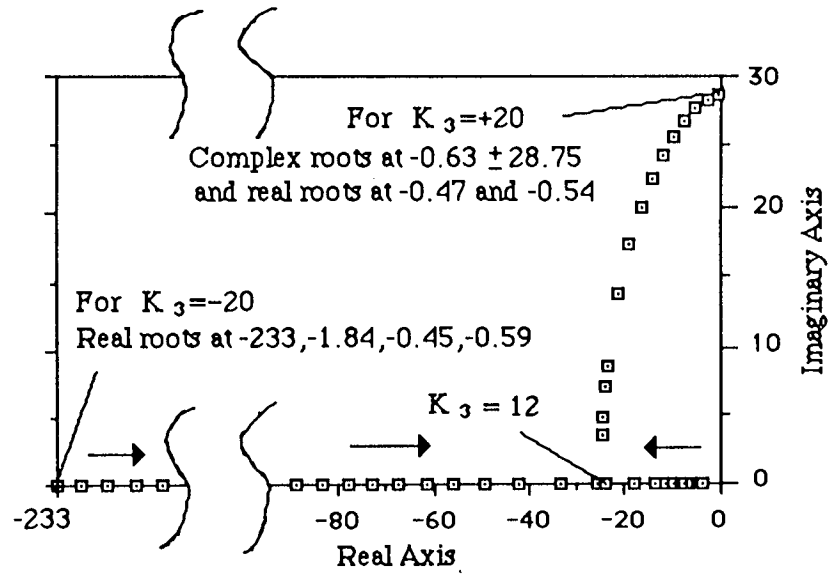


Figure 6.22. Root locus for the variation of K_3 in the range of K_3=–20 to +20, showing the high frequency roots.

Again if this plot is compared to the corresponding one for PD control it is very similar in shape.

The variation of K_1 is plotted below in Fig.6.23 (the real roots are not shown), this is seen to be similar in shape to the PD plot of fig.6.17:

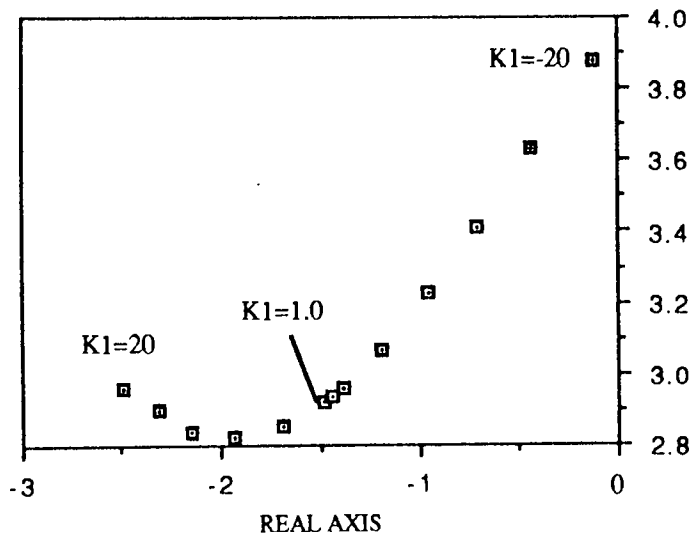


Figure 6.23. Root locus for the variation of K_1 in the range K_1=–20 to 20.

The root locus is now drawn for a range of joint velocities as seen in Fig.6.24. This shows the position of the high frequency roots and indicates that the roots go into the right hand half of the argand diagram at a combination of high joint velocities. The diagram has a similar form to the previous one for PD control of Fig.6.18.

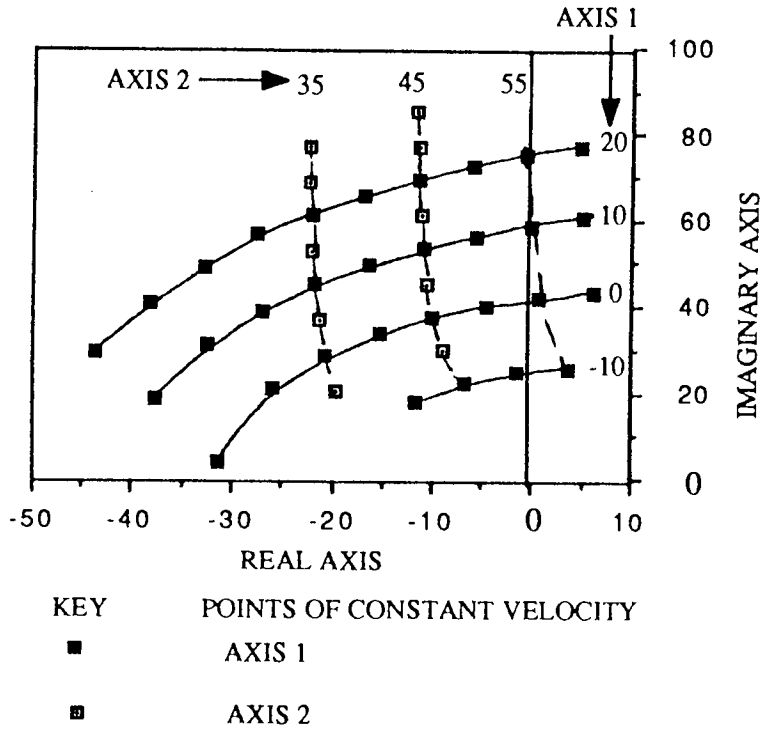


Figure 6.24. Root locus for the variation of the joint velocities showing the high frequency roots.

The low frequency roots are also plotted in Fig.6.25. This is a similar plot to that of the PD system showing at a particular combination of joint velocities the low frequency roots move to the right hand quadrant of the Argand diagram indicating instability.

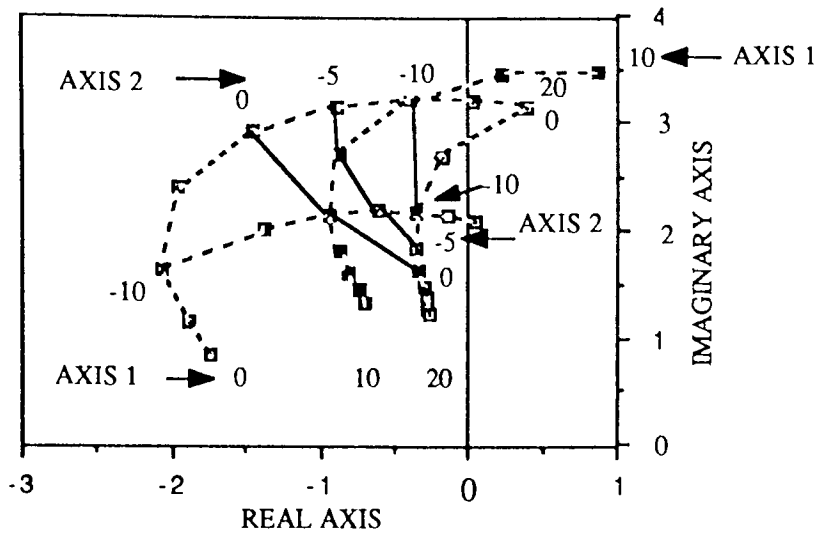


Figure 6.25. Root locus for the variation of the joint velocities showing the low frequency roots.

PID + Acceleration feedforward

The introduction of acceleration feedforward in the PID control system is analogous to the introduction of acceleration feedforward to the PD system. The numerator order is increased to the same order of the denominator i.e sixth order, although there in no cancellation due to the coupling terms. If the computed inertia for each axis is less than the actual inertia this causes the roots to move towards the right hand quadrant of the Argand diagram.

6.4.4 Degree of Inertial coupling

It was seen in chapter 3 on the dynamics of the manipulator how an acceleration in one axis causes a torque and hence acceleration in the opposite direction, due to the coupling inertia D_{12}, in the other axis. This is of some significance if the trajectory that the manipulator is required to follow is considered. As seen in chapter 5, dealing with off-line path and trajectory planning, in order for the manipulator to follow a path at a constant velocity between points, at each corner point it is required for the manipulator to accelerate at the corner points to meet the next required sector velocity in Cartesian space. This trajectory planned in

Cartesian space corresponds to rapid joint accelerations at the corner points in joint space as seen in fig.5.15 and fig.5.16 of chapter 5. The joint accelerations are analogous to a short step acceleration input but in opposite directions. In order to study how the inertial coupling terms affect the response of the system, the response of the PD system to a step acceleration input will be considered.

The inertial coupling torque either reinforces or opposes the control torque depending upon the direction of acceleration of the other axis. For instance, the equation for axis 1(excluding velocity coupling terms) is as follows:

$$T_{c1} = D_{11}\ddot{\theta}_1 + D_{12}\ddot{\theta}_2$$

If the acceleration demand in axis 2 is in the opposite direction to axis 1 then the control torque is increased:

$$T_{c1} + D_{12}\ddot{\theta}_2 = D_{11}\ddot{\theta}_1$$

This can be interpreted in another way, the acceleration torque available for link 1 is increased by the coupling torque due to the acceleration of axis 2. If the acceleration of axis 2 is in the same direction as axis 1 the coupling torque will now oppose the control torque to axis 1. The torque available to accelerate link 1 is now reduced:

$$T_{c1} - D_{12}\ddot{\theta}_2 = D_{11}\ddot{\theta}_1$$

During the initial period of time when a step acceleration input is applied to the system the acceleration and inertial coupling torque are the predominant portion of the total torque. This can be seen if the absolute value of the individual torque terms are expressed as a percentage of the total absolute torque and plotted against time, as seen in Fig.6.26:

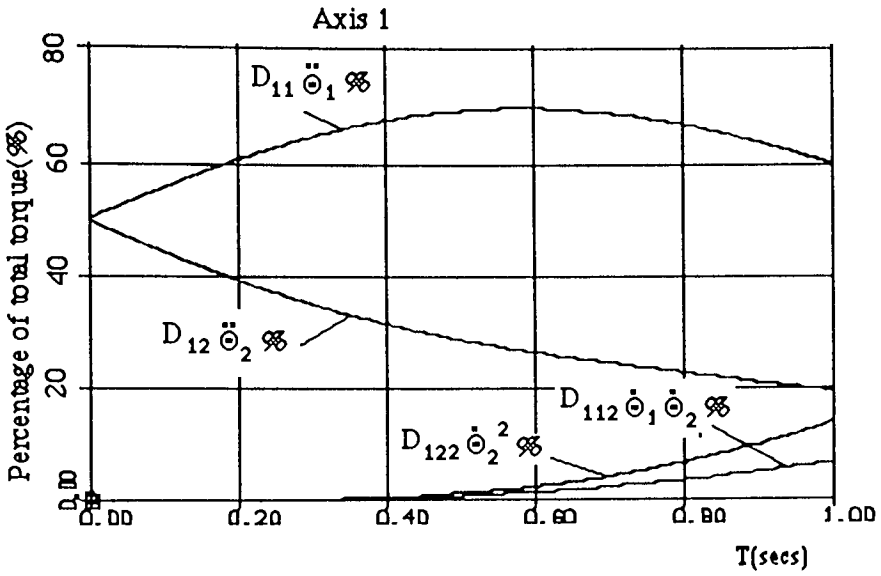


Figure 6.26 Showing the percentage of the total torque that the dynamic terms are for axis 1.

Fig.6.26 shows the percentage that each torque term takes for axis 1 for a step input of acceleration to axis 1 and a negative step acceleration input to axis 2. The velocity coupling terms are seen to be initially insignificant in the response until the velocity builds up. In axis 2 the inertia and coupling torque can be shown to dominate the response in a like manner to axis 1.

The acceleration response for axis 1 is now plotted for the following combinations of step input to the two axes:

A. Decoupled system response
B. Coupled system with +ve step to axis 1 and 2
C. Coupled system with +ve step to axis 1 and -ve step to axis 2.
D. Required step input

These are shown plotted in Fig.6.27.

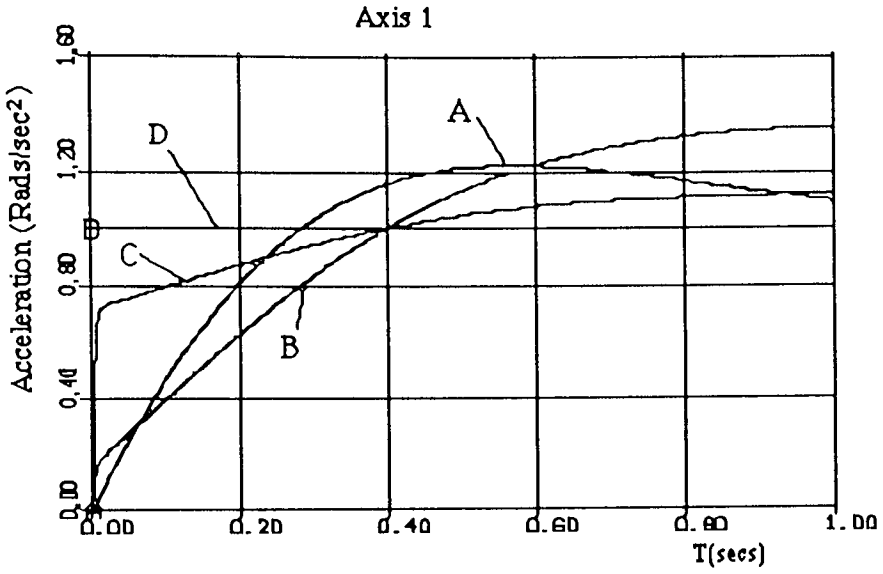


Figure 6.27. The acceleration response for axis 1.

As seen in Fig.6.27,the coupled response(C) reaches nearly 80% of the required step in a very small amout of time compared to the non-coupled response(A) and the response (B).

The acceleration response for axis 2 is now plotted for the following combinations of step input to the two axes:

A. Decoupled system response
B. Coupled system with +ve step to axis 1 and 2
C. Coupled system with -ve step to axis 1 and +ve step to axis 2.
D. Required step input

The responses are shown plotted in Fig.6.28, axis 2 overshoots the required step when the step inputs are of opposite sign in a very fast time(C). When the step inputs are of the same sign the response, due to the coupling, is seen to go negative initially(B) and is much slower than the decoupled response(A.).

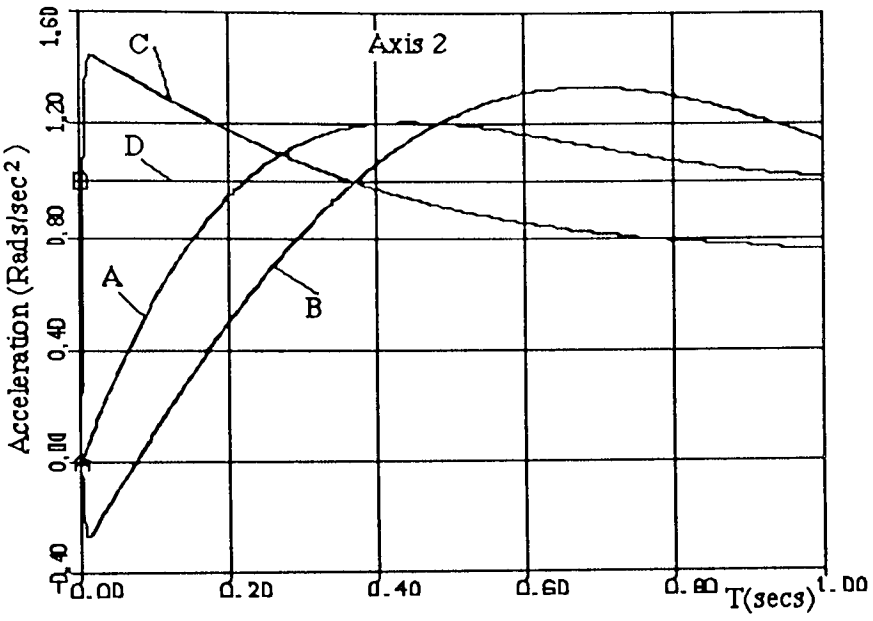


Figure 6.28. The acceleration response for axis 2.

The inertial coupling affects the ability of the system to track the planned trajectory as seen in Fig's.6.27 and 6.28. The coupling torque either reinforces or opposes the control torque and in our particular trajectory the acceleration demands at corner points are of opposite signs which means the control torque for each axis is reinforced. Previously in Table 6.5 the absolute positional error was obtained for each axis over the complete trajectory for the different control systems considered. The manipulator full dynamics were modelled and this was compared against the result for when the manipulator was modelled as a simple inertia. As seen in Table 6.5, for the PD control system with direct drive the error for the simple system is in fact larger than when the full dynamics of the manipulator are modelled. For instance in axis 1 the maximum error for the simple system is 0.0895 Rads while for the full system the error is less at 0.0562 Rads. This result is due to the effect of the inertial coupling between the two axes. The positional error is now plotted for the two systems for axis 1 and axis 2 as shown in Fig.6.29 and Fig.6.30.

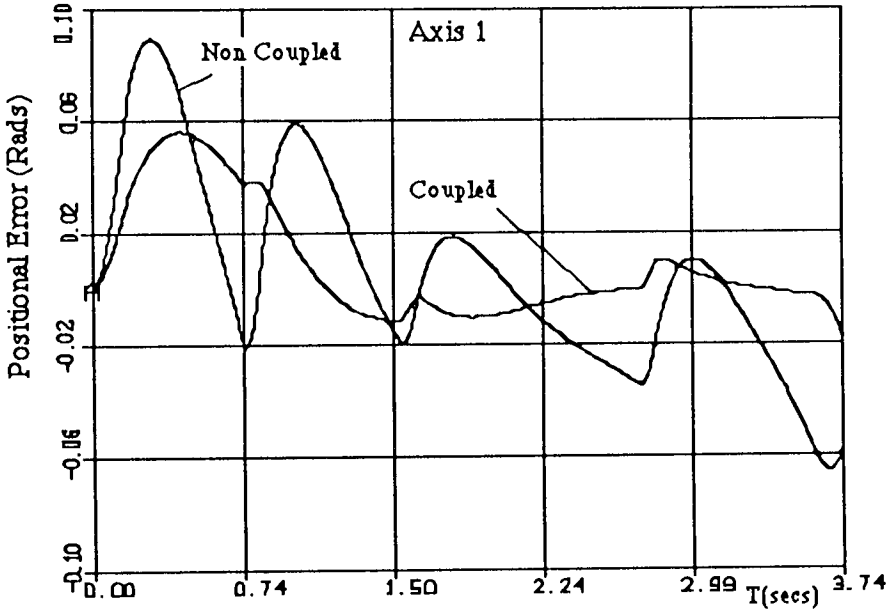


Figure 6.29. Showing the tracking error over the complete trajectory for axis 1.

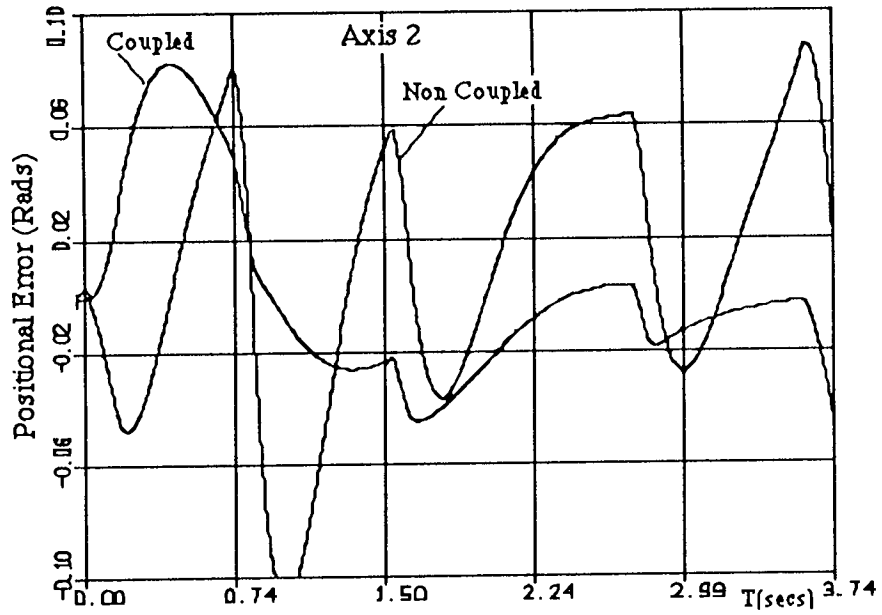


Figure 6.30. Showing the tracking error over the complete trajectory for axis 2.

As seen for both axes, the positional error peaks at the corner points for the non-coupled system. However, for the coupled system although there are small visible undulations at the corner points these are far less significant than in the non-coupled system. Certainly it appears that the positional error is decreased at the corner points due to the coupling torque reinforcing the control torque. The

largest positional error occurs at the second corner point in axis 2 for the non-coupled system this corresponds to the largest acceleration demand in the trajectory in axis 2 as seen in Fig.5.19 of chapter 5. However, this positional error has diminished considerably in the response for the coupled system as seen in Fig.6.30.

As seen in Fig.6.29 and Fig.6.30, the absolute error is large after a corner point, this occurs because the system cannot follow the rapid demand in acceleration at the corner point in each axis. In practical terms it is worth referring back to the example application of chapter 5 section 5.22 Fig.5.2. Although it is important for the manipulator to follow the planned trajectory for the whole of the path, the most crucial part is the segment during which it tracks the part on the conveyor and carries out an operation on the part. If the absolute error is large during this segment, the operation it carries out on the part may not be completed successfully. At the corner points of this segment the absolute error may not be important providing it follows the correct programmed position and velocity during the major portion of the segment. The other segments of the trajectory are not so critical providing the manipulator avoids any obstacles.

At high speed, when the velocity coupling terms are significant, it was shown , by a root locus analysis, that the roots of the system move into the right hand side of the Argand diagram indicating instability. The effect in practical terms when the manipulator is tracking at high speed is possibly for the arm to surge forward overshooting its path, then possibly reverse direction and surge back in the opposite direction. There could be significant high frequency oscillation of a much smaller magnitude due to the effect of the high frequency roots moving towards the right hand side of the Argand diagram.

In order to quantify the degree of inertial coupling the relative gain method(as descibed by Doebelin[29]), is used to specify the degree of interaction. The relative gain method requires the equations of the open loop system to be obtained at an equilibrium operating point where small changes in the the inputs Δm_1, Δm_2 cause small changes in the outputs Δc_1, Δc_2. For the two axis system the equations are obtained:

$$\Delta c_1 = K_{11}\Delta m_1 + K_{12}\Delta m_2 \qquad (6.58)$$

$$\Delta c_2 = K_{21}\,\Delta m_1 + K_{22}\,\Delta m_2 \qquad (6.59)$$

where $K_{11} = \left.\frac{\partial c_1}{\partial m_1}\right|_{m_2 \text{ constant}}$, $K_{12} = \left.\frac{\partial c_1}{\partial m_2}\right|_{m_1 \text{ constant}}$

$K_{21} = \left.\frac{\partial c_2}{\partial m_1}\right|_{m_2 \text{ constant}}$, $K_{22} = \left.\frac{\partial c_2}{\partial m_2}\right|_{m_1 \text{ constant}}$

The equations for the manipulator include the velocity terms associated with the Coriolis and centrifugal terms with the corresponding linearised coefficients. However, it has been seen that during a step acceleration the initial response of the system is governed primarily by the inertial coupling of the system so a quasi-equilibrium state is defined during this period where small changes in the input torque cause small changes in the ouput acceleration. With this assumption the equations for the two axis manipulator are obtained as follows:

$$\ddot{\theta}_1 = \frac{Jeff_2/N_2\ T_{M1} - N_1 D_{12} T_{M2}}{Jeff_1\, Jeff_2/N_1N_1 - N_1N_2 D_{12}^{\,2}} \qquad (6.60)$$

$$\ddot{\theta}_2 = \frac{Jeff_1/N_1\ T_{M2} - N_2 D_{12} T_{M1}}{Jeff_1\, Jeff_2/N_1N_1 - N_1N_2 D_{12}^{\,2}} \qquad (6.61)$$

If a feedback controller is applied to one of the outputs of the system the open loop gains for the uncontrolled variable change. For instance if the controller on axis 2 is in perfect steady state thus $\Delta c_2=0$. A new steady state gain is defined as :

$$a_{11} = \left.\frac{\partial c_1}{\partial m_1}\right|_{c_2 \text{constant}}$$

A relative gain λ_{11} is defined as:

$$\lambda_{11} = \frac{\left.\frac{\partial c_1}{\partial m_1}\right|_{m_2 \text{constant}}}{\left.\frac{\partial c_1}{\partial m_1}\right|_{c_2 \text{constant}}} \qquad (6.62)$$

The relative gains can be related to the K's and are obtained as follows:

$$\lambda_{11} = \frac{K_{11}K_{22}}{K_{11}K_{22} - K_{12}K_{21}} \quad , \lambda_{12} = \frac{K_{12}K_{21}}{K_{12}K_{21} - K_{11}K_{22}}$$

$$\lambda_{22} = \frac{K_{11}K_{22}}{K_{11}K_{22} - K_{12}K_{21}} \quad , \lambda_{21} = \frac{K_{12}K_{21}}{K_{12}K_{21} - K_{11}K_{22}}$$

It is now possible to display the relative gains in terms of the input and output variable as seen in table 6.7:

		Inputs	
		m_1	m_2
Outputs	C_1	λ_{11}	λ_{12}
	C_2	λ_{21}	λ_{22}

Table 6.7. Showing the relative gain terms.

When the relative gain terms are of similar magnitude this indicates strong interaction. It is now useful to calculate the relative gains for the two axis manipulator for the different gear ratio's considered. For the RTX manipulator where the gear ratio's are N_1=0.00114 and N_2=0.002281(e.g $N_1=\omega_{L1}/\omega_{M1}$ where L_1 is the load side of the gearbox, this number is small the gear ratio is referred to as being high) the relative gains are shown in table 6.8 below:

		Inputs	
		T_{M1}	T_{M2}
Outputs	$\ddot{\theta}_1$	1.107	-0.107
	$\ddot{\theta}_2$	-0.107	1.107

Table 6.8. Showing the relative gains for the RTX robot manipulator.

As seen in table 6.8 the interaction is relatively small between the two axes.

For the optimum ratio the gear ratio's are N_1=0.0.0037 and N_2=0.008 the relative gain terms are calculated as shown in table 6.9:

		Inputs	
		T_{M1}	T_{M2}
Outputs	$\ddot{\theta}_1$	3.164	-2.164
	$\ddot{\theta}_2$	-2.164	3.164

Table.6.9 Showing the relative gains for the optimum ratio.

The degree of interaction between the two axes is seen to be of significance for the optimum gear ratio system. Finally the relative gains are calculated for the direct drive system as shown in table 6.10.

		Inputs	
		T_{M1}	T_{M2}
Outputs	$\ddot{\theta}_1$	28.75	-27.75
	$\ddot{\theta}_2$	-27.75	28.75

Table.6.10. Showing the relative gains for the direct drive system.

For the direct drive sytem the interaction is very strong as seen by the fact that the relative gains are all of similar magnitude.

To conclude this chapter, in comparing the different control strategies it was seen that the presence of the dynamic coupling affected their performance. The coupling affects are reduced if there is a high gear ratio between the actuators of the manipulator and the link joints as in the RTX. Under these circumstances the controllers with acceleration feedforward perform far better than simple controllers

of PD and PID. However, when the gear ratio is low or the manipulator is directly driven the simple PD and PID perform far better than the acceleration feedforward controllers. The reason for this was seen by examining the symbolically derived transfer function equations for the coupled system where the numerator and denominator coefficients for the acceleration controllers no longer cancel.

The velocity coupling terms were seen to affect the stability of the system at high velocities and can cause the roots of the system to move to the right hand side of the Argand diagram which means the system is unstable. The inertial coupling terms affect the initial response of the control system and were seen to either reinforce or subtract from the control torque, depending on the sign of the demand acceleration signal for each axis.

Chapter 7:Discussion, Conclusions and Recommendations

Chapter 7- Discussion, Conclusions and Recommendations for future research

7.1 Discussion

Current manufacturing techniques place high demands upon manufacturing equipment. For a robot manipulator to replace existing equipment it must perform with a high degree of accuracy, at high speed and without significant vibration or becoming unstable, when following a desired trajectory. The design of control systems for robot manipulators is not an easy task since they are represented by second order, or higher, non-linear, coupled equations. It was identified in the literature survey that two particular aspects relating to robot manipulator control system design have not been addressed in any great detail by previous researchers, namely, how significant are the coupling terms in the dynamic equations, and what is the effect of the coupling terms on the performance of a number of typical independent axis control schemes?. These aspects are discussed in this chapter and conclusions are drawn from the results of the research, also suggestions for possible directions for future research are made.

In order to carry out this research it was necessary to choose a suitable example application. As outlined in the introductory chapter, the application of robot manipulators to parts transfer and assembly operations is becoming increasingly a more realistic and economic proposition. The reason for this is that robot manipulators are extremely flexible machines and can be easily reprogrammed to perform new tasks. However, the widespread application of robot manipulators is still thwarted by their high capital cost. This is due to the fact that their operating speed cannot compare or improve upon existing manufacturing equipment. By increasing operating speeds they will gain widespread application and hence manufacturers will be able to produce them in sufficient numbers to warrant a reduction in their price. The example chosen application thus reflects this strive for greater speeds. It is a typical task where the robot manipulator is required to track a part on a conveyor and carry out an operation on that part.

The dynamic model for a six degree of freedom manipulator is represented by a set of second order, or higher , non-linear coupled equations. The first three degrees of freedom of the manipulator provide the motion to place the hand in the vacinity of the target destination. This is often referred to as the gross motions of the manipulator. The three degrees of freedom of the wrist provide the motion to orientate the end effector in order to clasp the object correctly. This is often

referred as the fine motions of the manipulator. It can be argued that the wrist parameter variations(such as the inertia and postion of centre of gravity) , are small compared with the main body of the manipulator. The important parameters variations are in those providing the gross motion. To further reduce the complexity of the dynamic equations a two degree of freedom model of the robot manipulator is used. This retained the important characteristics of the six degree of freedom system namely the inertia coupling, Coriolis, and centrifugal velocity coupling terms in the dynamic equations for each axis.

The selection of a particular example application predetermined the desired trajectory for the manipulator to follow. The manipulator must track the part on the conveyor at a constant speed in a straight line. However, the manipulator is very often required to avoid obstacles in its path en route, so the manipulator must go via points to reach its destination as was shown in Fig. 5.2 of chapter 5.

The conveyor speed is usually dictated by how fast the manipulator can track the part accurately. Hence, in Fig. 5.2 the velocity V_c represents the maximum tracking speed that the manipulator can follow accurately. It must track the conveyor, which lies parallel to the Y axis at a velocity of V_c in the direction of the Y axis and have zero velocity in the direction of the X axis. However, the remaining path segments(where segment is the portion of the path between two corner or path points as seen in fig.5.3, chapter 5), are not necessarily parallel with any of the axes , so what is important is the resultant velocity at which the manipulator moves. In this application the resultant velocity would be set as V_c in order to minimise the cycle time. The manipulator must accelerate at each corner point to reach the next segment velocity. To do this smoothly a parabolic blend period was chosen for this purpose, this is a period of constant acceleration. It was shown in chapter 5 how the complete segment times and blend times can be calculated, based upon a chosen resultant velocity and acceleration for the complete path.

The next question that had to be resolved was what resultant velocity and acceleration to choose for the complete cycle. This was answered by looking at the specification of the example application. The manipulator can only track the part and carry out an operation on that part while the part remains in the tracking window of the robot manipulator(where the "tracking window" is defined in

Groover[26] as the intersection of the robot's work volume with the line of travel of the workpart along the conveyor). It must therefore achieve the correct motion velocities as quickly as possible as soon as the part enters the tracking window. This determines the ratio Txy_{jk} / Td_{jk} where Txy_{jk} is the period of time that the segment is at constant velocity and Td_{jk} is the total segment time. This lead to a very important derivation, of the ratio between the resultant velocity for the path to the resultant acceleration for the path. It was shown that to retain the same ratio of Txy_{jk} / Td_{jk} for each segment:

If the velocity is increased by n then the acceleration must be incresed by n^2 .

Hence, with these important relationships established for the example application trajectory, this allowed us to adjust the trajectory to the required level but retaining the correct scaling.

The picture was not complete at this stage since the first question to be answered was, what is the significance of the coupling terms in the dynamic equations?. To establish this, typical trajectory speeds could be calculated based upon the chosen path and as mentioned in the previous paragraph could be scaled to what ever desired speed was required at the conveyor. However, the path chosen , referred to as p300/2000 (where 300 is the resultant path velocity in mm/sec and 2000 the resultant path acceleration in mm/sec^2) was arbitrarily chosen and there were many other possible routes that the manipulator could take. Drawing upon the knowledge in the kinematics of chapter 2, near the boundary of the manipulator's work space, which for a two d.o.f. manipulator is the perimeter of a circle of radius equal to the link lengths, the manipulator reaches a singularity. At a singularity the denominator of the Jacobian approaches zero hence, the magnitude of the joint angles become excessively large. It was shown(as seen in fig. 5.20 , section 5.6.1 of chapter 5), that restricting how close the manipulator goes near the boundary of the work space and by making the work space in to an outer and inner circle, and then arranging for one of the segments of the path to pass at a tangent to this circle, this yields the maximum joint velocities for the reduced work space. This established a path that is a means for gauging the significance of the coupling terms.

One essential criterion remains: What is the definition of the significance or relevance of the coupling terms?. In the computed torque controller of Paul[8] and Bejczy[9] the non-linear coupling terms are calculated seperately. They are fed forward as individual coupling terms. However, the torques can be positive or negative and if they were expressed as a percentage of the total they would possibly be greater than 100 % , since positive and negative terms in the total would cancel each other out. It thus makes sense express the absolute value of a coupling term as a percentage of the absolute total.

A realistic path, trajectory and the means of calculating the relevance the coupling terms was established. This uses however, the specific parameters for the dynamic equations of the manipulator considered namely the RTX. The relative magnitude of these parmeters should be typical for any manipulator considered. The computer program TORQUE calculates the relative maximum percentage that each term in the dynamic equations takes over the complete trajectory. For path p300/2000 it was seen that the inertial coupling terms are a high percentage at 86.6% and 84.3% for axis 1 respectively as seen in table 5.13. For axis 2 the inertial coupling terms are 94.8% and 82.7%. The Coriolis and centrfugal terms for axis 1 were seen to be 9.8% and 29.4% and for axis 2 the centrifugal term was 29.4%. Thus the velocity coupling terms are quite significant in relative terms. Near a singularity the velocity coupling terms are more significant as seen in table 5.14. The Coriolis term and centrifugal term have a relative significance of 47.2 % in axis 1 and the centrifugal term has a relative significance of 51.9% in axis 2.

The important fact in table 5.13 is seen to be that if the path is time scaled by increasing the velocity to 600 mm/sec and hence the acceleration to 8000 mm/sec^2 then the significance of the terms remains exactly the same. Hence, not only has time scaling occurred but also dynamic scaling. This dynamic scaling can be deduced intuitively if the dynamic equations of the manipulator are observed as seen in chapter 5. However, the absolute magnitudes of the joint velocities and accelerations do not remain the same. Their effect upon the stability of the control system will be discussed later in this chapter.

The independent axis control schemes that were tested over the planned trajectory were chosen to reflect typical industrial robot control schemes. These control schemes are also representative of the linear portion of the computed torque controller(as seen in section 4.2.1 of chapter 4). However, in the computed torque implementation the controller gains are normally computed on-line at each position along the trajectory rather than having fixed coefficients.

The choice of controller gains for the independent axis control schemes is quite straight forward when the the robot manipulator is represented by a non-coupled effective inertia at each axis. For the Proportion with derivative feedback controller(Pd) ,Propotional plus derivative controller(PD) and PD plus acceleration feedforward controller, the closed loop characteristic equations are of second order(as seen in table 6.1 of chapter 6 and the transfer function for the PD plus acceleration feedforward controller is shown in equation 6.14 of chapter 6). Each term of the characteristic equation is chooseable except the highest order term and the characteristic equation can be equated to the standard second order equation form(equation 6.19 of chapter 6). The undamped natural frequency and damping ratio may be chosen accordingly. For the Proportion and Integral controller(PI) with derivative feedback, Proportional, Derivative and Integral controller(PID) and PID with acceleration feedforward controller, the characteristic equation is of third order (as seen in table 6.2 of chapter 6 and the transfer function for the PID plus acceleration feedforward controller is shown in equation 6.16 of chapter 6), and all terms may be chosen except the highest order term. The characteristic equation may be equated to the standard form for the ITAE index(equations 6.28 to 6.30 of chapter 6) and a suitable value of undamped natural frequency chosen.

It is desirable to set the controller gains as high as possible in order to obtain a fast response, maximum disturbance rejection and minimum tracking error. However, the undamped natural frequency cannot be chosen arbitrarily and for conservative design must be set at half the structural frequency. For the RTX^4 manipulator the stiffness of the joints is low and it thus has a low structural frequency, the controller gains may not be set very high in this case.

The results for simulation, as seen in table 6.4 of chapter 6, are specific for the RTX robot manipulator. These results show that the acceleration feedforward controllers perform far better than their corresponding basic versions. There is no real significant difference between the PD with acceleration feedforward and the

PID with acceleration feedforward. Similarly there is no significant difference between the PD and PID control performance(as seen in table 6.4 of chapter 6). This is because the integral term in the PID control system does not play a part in the dynamic tracking ability since its action is slow. It only really affects the final steady state position error. Thus, the PID acts in very much the same way as the PD system. This was verified by comparing a plot of the tracking error of both schemes and these were seen to be very much the same, although these plots are not included in the thesis. The Pd controller shows very poor performance however, this is not entirely unexpected since the Pd system is a type 1. This means that for velocity and acceleration reference inputs there will be tracking errors. The PI with derivative feedback controller performs badly, as seen in table 6.4 of chapter 6. The closed loop characteristic equation for this system is the same as the PID controller but the numerator of the transfer function has one less term(as indicated in table 6.2 of chapter 6). This means that it is only a type 2 system rather than a type 3 for the PID system. The derivative feedback term causes an error to an acceleration input hence the poor performance in tracking the planned trajectory. The results for this simulation are specific for the RTX robot manipulator. The RTX manipulator has very high gearbox ratios for each axis. This means that when the coupling terms are reflected back, from load side to motor side through the gearbox, they are very small in comparison with the motor torque. Hence, for this particular system the manipulator modelled as a non-coupled inertia at each axis is an appropriate one.

In order to establish the effect that the coupling terms have upon the independent control schemes the gearbox ratio was modelled as:

a) an optimum ratio for the system.

b) a directly driven system i.e. the gearbox ratio was unity.

For the simulation, the chosen controller gains were based upon the maximum load configuration and the undamped natural frequency based upon the structural frequency at this configuration. The simulation results were also obtained for the situation when the manipulator is modelled as a non-coupled effective inertia, for comparison with the coupled system results. These results are shown in table 6.5 and show their comparitive performance has changed to that obtained for the RTX manipulator. The PD and PID with acceleration feedforward controllers no longer perform better than the PD and PID controllers. This is not surprising when the transfer function for the coupled system is considered. For instance for the PD

with acceleration feedforward the characteristic equation is of fourth order compared to the second order characteristic equation from which the controller gains were designed(equation 6.55,6.56). The numerator of the transfer function does not cancel with the denominator as is expected in the design theory for this independent scheme. The results for when the manipulator's gearbox ratios are optimum and unity show quite clearly that the coupling terms, that were formerly ignored in the design procedures for the independent axis control schemes, have an adverse effect upon the tracking performance. This is further substantiated if for instance the design position for the roots of the PD control system are compared to those of the coupled system. These were shown plotted previously in Fig.6.16 of chapter 6 and this plot is reproduced below:

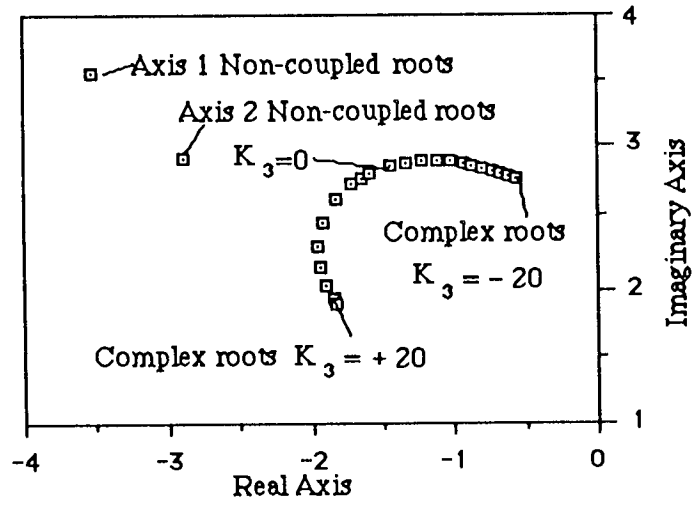


Figure 6.16. The root locus for the variation of K_3 for the range K_3=–20 to +20, showing the low frequncy roots.

For the non-coupled system the roots were placed at –3.52 –j3.58 , –3.52 +j3.58 and –2.89 –j2.95, –2.89 +j2.95 for axis 1 and 2 respectively with a damping ratio of 0.7 as shown in fig. 6.16 above. However, in reality the characteristic equation of the coupled system is that shown in equation (6.53) i.e. fourth order. The inertial coupled sytem roots are then at –1.6 –j2.8, –1.6 +j2.8 (not taking into account the velocity coupling i.e K_3=0 in fig.6.16) and two real roots of -3.37 and -111.5(not shown in fig.6.16) and the damping ratio for the complex roots is 0.15. One question that can be asked is can the roots be chosen based upon the fourth order characteristic equation rather the second order characteristic equation of the non-coupled system. Referring to the characteristic equation for the coupled system(equation 6.53), the first two terms can be chosen since they are

made up of the controller gains for the two axes. However, higher order terms are not easily adjustable. Setting the first two terms according to the ideal forms of the ITAE index for a fourth order system would not give suitable positioning of the roots. It is clear that for the coupled system, when the coupling terms are significant, choosing suitable controller gains to obtain a suitable performance is not as straight forward as the procedures established for the independent schemes.

Another effect of the coupling terms is shown in the results of table 6.5. The tracking errors of the PD and PID control schemes, for when the manipulator is modelled as an effective inertia at each axis with no coupling, are in fact larger than when the full non-linear dynamics of the manipulator are modelled. This occurs as a result of the inertial coupling and the type of trajectory the manipulator is required to follow. At the corner points there is a period of constant acceleration in Cartesian space that the manipulator must achieve in order for it to reach the next segment Cartesian velocity. In joint space this corresponds to an abrupt acceleration demand in each axis of opposite direction for a short period of time (fig.5.15,5.16 of chapter 5). It was shown that when the manipulator must follow a step demand of acceleration the percentage of the total torque that the motor and inertial coupling terms take is initially equal to 100%. The other coupling terms are insignificant for a short period of time as seen in fig.6.26 . Hence, during the short period of time that the acceleration demand occurs the response is dominated by the inertial torque and inertial coupling torque. For this particular trajectory, at the corner points the inertial coupling between the two axes reinforces the control torque. This means that the system reaches the acceleration demand signal in each axis far faster and hence the positional error is reduced at the corner points, this is seen in fig.6.27 and 6.28 for axis 1 and 2 respectively. The largest error for the non coupled system is seen to occur at the corner points as seen in fig.6.29 and fig.6.30. The reinforcement of the control torque would not occur if the acceleration demands in each axis were in the same direction, in fact they would cause the system to be far more sluggish than the non coupled system as seen in Fig.'s 6.27 and 6.28. In between the corner points of the trajectory the velocity coupling terms become more prominant and they will reinforce or oppose the control torque depending upon their sign. In order to anticipate the degree of inertial coupling the relative gain[29] was calculated for the three gearbox ratio's considered. These are seen to reflect this inertial coupling, for instance the RTX has only a cross coupling value of -0.107 compared to 1.107 between the same axes as seen in table 6.8. However, the cross coupling for the direct drive system

is -27.75 compared to 28.75 between same axes as seen in table 6.10. Although the inertial coupling in this particular instance is beneficial in terms of the reduced tracking error the result is rather fortuitous and is not really controllable.

In order to look at how the velocity coupling terms affect the stability of the independent control schemes the non-linear equations of the manipulator were linearised about a particular configuration. The resulting linearised coefficients are velocity dependent and were referred to as K_1, K_2, K_3 and K_4(equations 6.43-6.46). The coefficients $K_1$and K_2 are a result of linearising the Coriolis term in the axis 1 equations. The K_3 coefficient is a result of linearising the centrifugal term in axis 1 and K_4 is the result of linearising the centrifugal term in axis 2. The linearised equations were shown to be a good representation of the non-linear system equations.

It was mentioned previously that although the relevant significance of the dynamic terms remains the same if the trajectory is scaled according to the established relationship, the absolute magnitude of the joint velocities and accelerations do not remain the same. The characteristic equation of the coupled system for PD control as seen in equation(6.53) involves the linearised coefficients K_1 to K_4. It can be anticipated that the position of the roots of the characteristic equation will be affected by the magnitude of these linearised coefficients. A root locus for the variation of K_3 shows that the roots move towards the right hand side of the Argand diagram as seen in fig.6.15, 6.16. Similarly if a root locus for the variation of K_1 is drawn the roots move to right hand side of the Argand diagram as seen in fig.6.17. However, this only occurs at very high magnitudes of joint 1 and joint 2 velocities in the order of + or – 25 rad/sec. For the test trajectory(referred to as p300/2000), the maximum joint velocities are 1.28 rads/sec for joint 1 and 1.5 rads/sec for joint 2 and these do not necessarily occur at the configuration chosen for linearisation. Hence, the roots will remain very close to the position for zero velocity of fig.6.16. However, if the trajectory is scaled up to a more realistic speed that would be expected by industrial applications to say 6 m/s i.e p6000/800000, the maximun joint velocities become 26 rads/sec and 30 rads/sec for joints 1 and 2 respectively. This is more in the range that was considered for the variation of K_1 and K_3.

The linearised coefficients are dependent upon both the joint velocities and thus the effect upon the roots of the characteristic equation will depend upon maginitude of these joint velocities. This lead to the idea of plotting the root locus for ranges of joint velocities rather that linearised coefficients. The resulting plots are shown in fig.s 6.18, 6.19. The root locus of fig.6.18 represents the high frequency roots(N.B. the high frequency roots represent a complex pair of roots of high frequency, there are also a pair of low frequency complex roots not shown in fig.6.18), and shows that at certain combinations of joint velocities the high frequency complex roots move towards the right hand side of the Argand diagram. For instance with a joint 2 velocity of 55rads/sec and a joint 1 velocity of 20rads/sec the high frequency roots are in the right hand side of the Argand diagram(the high frequency roots are at +6 –j77, +6 +j77 and the accompanying low frequency roots are at –0.5 –j0.6, -0.5 +j0.6). In fact for any joint 1 velocity the high frequency roots will probably be in the right hand side of the Argand diagram for that particular value of joint 2 velocity.

The low frequency root locus is a plot of the low frequency roots and does not show the other accompanying roots. The low frequency roots locus shows also, that for certain combinations of joint 1 and joint 2 velocity the low frequency roots will be in the right hand side of the Argand diagram. For instance at a joint 1 velocity of 20 rads/sec and a joint 2 velocity of -20 rads/sec the low frequency complex roots are at +0.19 –j2.9, +0.19 +j2.9 and the other two roots are real at -158 and -2.95. The position of the low frequency roots for this example are shown in fig.6.19, chapter 6, this is reproduced below:

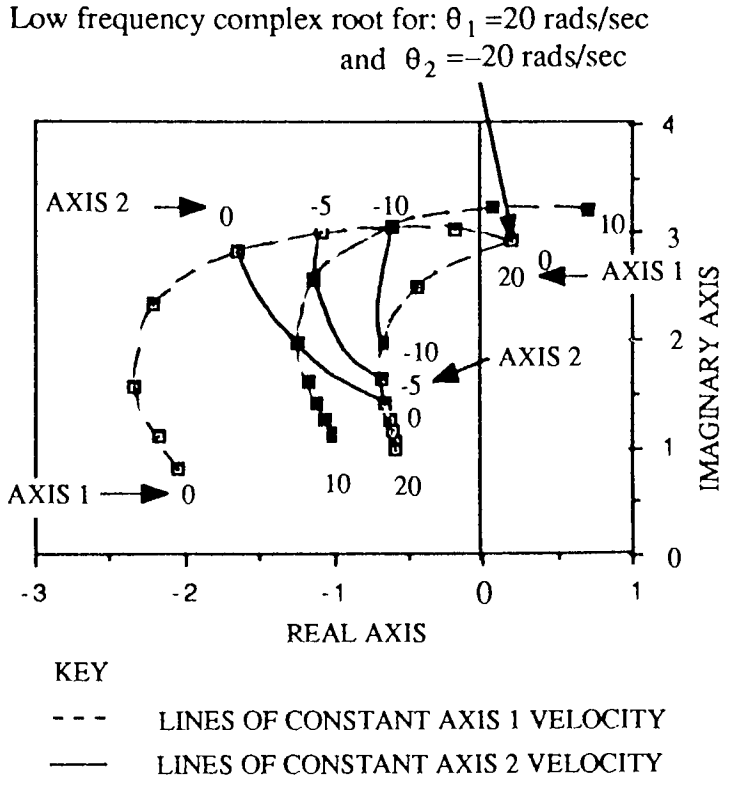


Figure 6.19. Root locus for the variation of the two joint velocities showing the low frequency roots.

The instability of the PD independent axis control scheme has thus been established at high speeds. A similar analysis for the PID control system showed the instability of the system at high speeds(see fig.6.24,6.25). The position of the roots will not be affected so much by the velocity coupling terms when each axis has a high gearbox ratio. It is very likely that if the RTX robot manipulator could operate at more typical industrial speeds that its stability would only be slightly affected by the velocity coupling terms.

It can be concluded that the question of how the coupling terms affect the performance of a number of typical independent axis control schemes is very much related to the initial question of how significant are the coupling terms. It has been shown that when the coupling terms are insignificant, as for example in the case of the RTX robot manipulator having a high gearbox ratio, the effect of the coupling terms on the independent axis control system performance is small. The coupling terms were shown to be significant when the manipulator is directly driven. The effect upon the independent axis control system performance in this instance is two fold. The inertial coupling was shown to reinforce the control torque at the corner points of the trajectory, where there is an abrupt demand in

acceleration in each axis of opposite sign. This subsequently reduces the tracking error at the corner points along the trajectory where normally the non coupled system was shown to exhibit maximum tracking error. However, this effect is not controllable and is therefore not desirable in a control system. The second effect is that due to the velocity coupling terms. Although it was shown that their relevant significance remains the same for a scaled trajectory their absolute magnitude does not remain the same. At high trajectory speeds the velocity coupling terms were shown, by the root locus analysis(as discussed in the preceding paragraph), to cause the independent control system schemes to become unstable.

The analysis carried out in this thesis was applied to a two degree of freedom manipulator. This analysis could be extended to consider the case of a six degree of freedom manipulator . However, it would be anticipated that similar results would be observed for this manipulator.

In the general approach to control system design for robot manipulators, it has been shown that the independent axis control designs will not work for robot manipulators that have significant coupling between links. The coupling effects become particularly pronounced at high speed and are not controllable. The question arises what alternatives are available to the designer of robot manipulators in order that the manipulator may operate at high speeds. These alternatives will now be discussed.

To some extent the application of many joint motion control schemes that were reviewed in chapter 4 is hindered by the characteristics of the design of many current robot manipulators. A similar problem was encountered in the design of suitable trajectory planning algorithms for manipulators. Many robot designers initially did not take in to consideration how important the geometry of the manipulator was to the solution of the inverse kinematics for the trajectory planning algorithm. In some earlier designs of robots the geometry was so complicated that the inverse kinematics solution was extremely difficult if not impossible to obtain. This lead to the idea of designing manipulators that had readily solvable inverse kinematics solutions. In particular for manipulators with six degrees of freedom with six revolute joints, with the last three joint axes intersecting at a point, a solution could be readily found. The design or implementation of suitable control systems is similarly hindered by the design of robot manipulators.

The application of the computed torque control system of Paul[8] and Bejczy[9] would be a possible contender for implementation if the designs of robot manipulators were changed. This would require the manipulator design characteristics to include high structural stiffness, little or no friction at the joints and no gearbox compliance. Also since the computed torque method is a model based controller the parameters of the manipulator would have to be accurately known. However, robot manipulators are very frequently characterised by high indeterminant friction at the joints, compliance in the gearbox and the parameters of the manipulator are not usually accurately determined by the manufacturer. The computed torque controller is fairly sensitive to parameter variations. This was seen for the acceleration feedforward controllers where inaccuracies in the computed inertia and uncompensated high coupling between joints cause the controller's perform to degrade considerably. Hence, the application of the computed torque method to a manipulator that cannot be accurately modelled would not give satisfactory results.

The design of robot manipulators could change in order to make possible the implementation of the computed torque controller. For instance with the advent of more efficient electric motors, with high power to weight ratio the links of the manipulator could be driven directly. This would effectively eliminate the friction at the joints and there would be no problems associated with compliance in the drive chain. Another benefit of direct drive is that the structural stiffness is considerably increased since there is very little compliance between the link and motor. This means that the controller gains could be set at higher values which would correspondingly increase the tracking performance. However, obtaining motors with suitable performance to be used with direct drive may be a problem. In order to make use of existing motors the links of the manipulator could be redesigned to reduce their inertia values. For instance the use of light weight , high strength alloys or carbon fibre materials as used in the aircraft industry. In a directly driven robot manipulator the coupling terms are significant and can not be ignored. If the computed torque method of Paul[8] and Bejczy[9] was used these terms may be accurately compensated for providing that the parameters of the manipulator are known. The design method of the manipulator should take in to account that the manipulator parameters need to be known. For certain types of manipulator such as the SCARA type(where the parameters of the arm links only operate in the horizontal plane), the inertia about a single axis only needs to be determined, hence in this particular type the parameters can be easily be

determined. An advantage of computed torque controller is that the controller gains can be easily chosen when the modelled parameters equal the actual parameters. Each axis is then represented by a second order differential equation in error space as was shown in equation(4.9) chapter 4.

The alternative control system designs available to robot manipulator manufacturers are to some extent the re-application of control laws established for linear systems. In order to apply them to robot manipulators the equations representing the manipulator are linearised in a suitable form for application . For instance in the application of the minimum-time control method of Kahn[12], a direct solution to the time optimal problem can only be found using numerical techniques. The solution must be computed for each trajectory. This means that it does not lend itself very well to practical implementation. In order to over come this problem the manipulator equations are linearised and a solution can then be found that approximates the time optimal problem and is referred to as near minimum control. The solution is only an approximation to the time optimal problem and its performance depends upon how accurate the linearised equations(used in the solution) are representitive of the system. To over come this problem it would be necessary to go back to using the full non-linear equations to obtain a numerical solution. If the computing power was available then the solution could be precomputed for each tyrajectory. This method would not lend itself to the real time tracking application situation.

Other alternative joint motion control schemes are the Cerebellar model articulation controller of Albus[18] and the Fuzzy logic controller of Scharf[20]. These I consider, can really be classified as learning controllers. In order to achieve accurate tracking the task cycle must be repeated a number of times in order for the system to choose and optimize upon the controller gains. The control structures are normally independent axis schemes, and it has been shown that it is not easy to choose the controller gains when there is strong coupling between axes. So these schemes are restrained by the controller structure and can only choose a compromised set of gains rather placing the poles. It is questionable whether at high speeds the controller gains can be chosen at all to give stability. The Fuzzy logic controller has only been tested at low speed by Scharf[20]. Their suitability to all robot manipulator applications is restricted by their need to learn from repetitive motions. For instance in the real time tracking situation it is likely the vision sensor would identify the postion of a randomly positioned part approaching the

tracking window of the robot. The coordinates would then be relayed to the robot that would immediately need to move meet the part. This would not allow the learning type controller to operate successfully as the same exact motion would not be repeated.

Another set of alternative control designs are broardly classed as adaptive control techniques. These were established from the need to obtain good control performance in the presence of uncertain modelled parameters. The application of adaptive techniques to current robot manipulators is a logical one since very often the parameters of the manipulator are not known with sufficient accuracy for model based controllers to be applied. The adaptive control system is employed to identify the configuration dependent effective inertia of the manipulator. When the manipulator picks up an object of unknown mass the adaptive system must identify the effective inertia change and adjust controller gains to sustain a suitable performance. The situation could arise in the example application when the parts approaching the tracking window of the manipulator are of unknown mass. Therefore the adaptive controller would have to adjust to this parameter variation.

There are a number of adaptive methods that have been postulated by previous researchers. For instance Self Tuning Control(STC) of Koivo[24], this method is designed to identify changes in the parameters and then adjust controller gains to retain system performance. The structure of this controller is thus an identification algorithm, a control synthesis algorithm and a control implementation algorithm. A linear model non-coupled model of each axis is established and the parameters of this model are obtained by the identification algorithm. The configuration dependent effective inertia and changes in the inertia due to unknown loads that are picked up are identified. At each stage the identified parameters are fed to the control synthesis algorithm that computes the new controller gains. In the identified model the coupling between axes is ignored, therefore when the system converges on the parameters of the system it is really converging on an independent axis control scheme. It is questionable how well this control system can perform in the presence of high coupling effects and whether it can remain stable. If the manipulator is operating at very high speeds the parameters of the system are changing extremely quickly. This leaves only a very short period of time to identify the change in parameters. If it can not identify the new parameters fast enough then the performance of the system will operate as a fixed term control system and may become unstable.

It is my opinion that it does not make sense to be identifying the configuration dependent effective inertia parameter changes. If the parameters were determined accurately prior to assembly, the effective inertia changes due to configuration could then be calculated from the dynamic equations of the manipulator. A modelled based scheme such as the computed torque technique could then be used when the manipulator operates in no load conditions. This would operate far faster than a STC system that must identify the configuration dependent inertia parameters. However, as mentioned earlier, the computed torque system performance will degrade considerably in the presence of uncertainty in the modelled parameters. This would occur in the case when the manipulator is made to pick up an object of unknown load. It is during this period of the task cycle that the adaptive scheme would operate with far better efficiency than the computed torque control system. This leads to the idea that if the manipulator is required to operate in a situation where it is required to pick up unknown loads why not combine the two methods in a sequential form. For instance, if the parameters of the manipulators are accurately known then whilst the manipulator moves unloaded a computed torque scheme would be employed. However, when the manipulator picks up an unknown load and the performance is measured to degrade below an acceptable norm, the STC could be switched in and the computed torque controller switched off. The STC then takes over the control of the manipulator until such time that the manipulator unclasps the unknown load. In order to obtain a smooth transition between the switching of the two control systems the identification algorithm and controller synthesis algorithm of the STC are made to operate during the operation of the computed torque controller as well. This type of controller design would thus encapsulate both of the desireable characteristics of the computed torque controller and the STC.

Other adaptive control strategies include Model Reference Adaptive Control (MRAC) of Dubowsky[23]. This method does not identify the modelled parameters of the manipulator in order to adapt. Instead, each axis of the manipulator is modelled by a second order time invarient differential equation. The control system (e.g. PD, PID), feeds signals to the model and the actual system. The error between the model and the actual system is used by an algorithm that adjusts the controller gains for each independent axis based upon a stability criteria. The validity of using a linear time invarient model for each axis is questionable since for some manipulators the configuration dependent parameters vary quite considerably. This method ignores the coupling terms between axes that can cause

instability at high speeds when they are significant. The MRAC is sometimes referred to as a model following method so can the non-linear manipulator be expected to follow a linear model of the system. This suggests the idea of employing a non-linear model of the system for it to follow.

Another adaptive control scheme is the Adaptive Perturbation Controller(APC) of Lee[25]. This goes some way to being a hybrid controller since in consist of two parts, an algorithm to compute the nominal torques based upon known modelled parameters, and an adaptive controller. This is not quite the same as the idea of the hybrid controller since the two parts of the controller operate simultaneously and the algorithm calculating the nominal torques does not include a linear controller. Parameter variations are dealt with by the adaptive part of the controller. The adaptive part of the controller uses a model of the manipulator that is linearised about the trajectory. It thus identifies changes in the model about the trajectory. An optimal control strategy is used for the control synthesis algorithm. The APC is probably the most promising of the adaptive strategies. However, it was shown to perform better than a model-based controller only when there was uncertainty in the modelled parameters. It is also questionable how quickly it would be able to react to changes in the modelled parameters at high speeds and whether the nominal torques would cause instability if the parameter changes were not identified fast enough.

It has been seen that if a control system for a robot manipulator is to be successful it must be able to compensate for the non linear dynamics of the manipulator. The model-based control system designs available to the robot manipulator manufacturer rely on the parameters of the robot to be accurately known and to have certain desirable characteristics e.g. high stiffness, low friction at joints etc. , in order that their performance will be acceptable. These characteristics must be aimed for by the robot designer if the model-based control system design are to be implemented. In order to over come the problem of uncertain modelled parameters in robot manipulators adaptive techniques have been proposed. It is evident that these schemes must be shown to perform at high speeds if they are to be implemented successfully.

7.2 Conclusions

The conclusions are made in the light of the initial objectives which were:

1. How significant are the coupling terms in the dynamic equations of a robot manipulator.

2. What is the effect of the coupling terms on the performance of a number of typical independent axis schemes.

1.

i) The significance of the coupling terms have been determined for two degrees of freedom of the RTX robot manipulator and are based upon the parameters for that specific manipulator.

ii) The significance of a coupling term is the absolute torque for that term expressed as a percentage of the total sum of all the absolute torque terms for a single axis.

iii) The significance of the coupling terms has been calculated over a realistic path and trajectory and the maximun significance that each term attains in that path and trajectory(p300/2000) was found to be as follows:

Axis 1

$$\frac{|D_{11}\ddot{\theta}_1|}{|T_1|} * 100\% = 87\% \quad \text{Inertia term}$$

$$\frac{|D_{12}\ddot{\theta}_2|}{|T_1|} * 100\% = 84\% \quad \text{Inertia coupling term}$$

$$\frac{|D_{122}\dot{\theta}_2^{\,2}|}{|T_1|} * 100\% = 29\% \quad \text{Centrifugal term}$$

$$\frac{|D_{112}\dot{\theta}_1\dot{\theta}_2|}{|T_1|} * 100\% = 10\% \quad \text{Coriolis term}$$

Axis 2

$$\frac{|D_{22}\ddot{\theta}_2|}{|T_2|} * 100\% = 95\% \quad \text{Inertia term}$$

$$\frac{|D_{12}\ddot{\theta}_1|}{|T_2|} * 100\% = 83\% \quad \text{Inertia coupling term}$$

$$\frac{|D_{211}\dot{\theta}_1^{\,2}|}{|T_2|} * 100\% = 29\% \quad \text{Centrifugal term}$$

iii) When the path was planned near a singularity the significance of the velocity coupling terms in axis 1 were found to increase to 47% and in axis 2 to 52%.

iv) The significance of the coupling terms is reduced when they are reflected back to the motor if the gear box ratio (expressed as ω_L/ω_M where ω_L is loadside of gearbox and ω_M is motor side of gearbox.) is less than 1.

v) The significance of the coupling terms remain the same if the path and trajectory is scaled by increasing the resultant velocity by n and the resultant acceleration by n^2. This retains the same ratio of Txy_{jk}/Td_{jk} for each segment of the path(where Txy_{jk} is the period of time that the segment is at constant velocity and Td_{jk} is the total segment time).

vi) The absolute joint velocities and accelerations do not remain the same when the path and trajectory is scaled. For p300/2000 the maximum joint velocities are: 1.25 rad/sec and 1.5 rads/sec for joints 1 and 2 respectively.If scaled by n=20 the coresponding joint velocities are 26 rads/sec and 30 rads/sec for joints 1 and 2 respectively.

2.

i) A number of independent axis control schemes have been tested over the planned path and trajectory(p300/2000) and these include Pd,PD,PID,PI with derivative feedback, PD with acceleration feedforward and PID with acceleration feedforward. These control schemes were tested in simulation using a model of RTX robot manipulator. The effect of the coupling terms upon these independent axis control schemes was assessed for manipulator when the drive ratio was :

1) High ratio
2) optimum ratio
3) directly driven.

ii) With a high drive ratio the follow results were observed:

a. The significance of the coupling terms was small.
b. The PD with acceleration feedforward and PID with acceleration feedforward control systems had comparable performance. They performed far better than the simple PD and PID controllers.
c. The PD and PID controllers had comparable performance.
d. Pd and PI with derivative feedback controllers performed less well than any of the other control schemes.
e. There were very little coupling effects evident in the independent axis control schemes response for the high drive ratio.

iii) When an optimum and direct drive ratio was used in the manipulator model the following coupling effects were observed:

a. The PD with acceleration feedforward and PID with accleration feedforward control systems were shown to perform less well than the PD and PID control systems.
b. PD and PID control systems performed the best in the presence of high significant coupling.
c. The inertial coupling between axes was shown to reinforce the control torque at the corner points of the trajectory, where there is an abrupt acceleration demand in each axis but of opposite direction.
d. The inertial coupling effect subsequently reduces the tracking error at the corner points along the trajectory where normally a non coupled system would exhibit the

maximum tracking errors. This effect is not a controllable one.
e. The velocity coupling terms were shown, by means of a root locus analysis, to cause the roots of the independent axis control schemes to move to the right hand side of the Argand diagram. This effect occurs at high tracking speeds where the magnitude of the joint velocities is high.

7.3 Recommendation for future research

The concept of a model-based robot manipulator control system is a good one. However, in order that this type of control can be implemented suuccessfully requires the design characteristics of robot manipulators to be changed. The model-based controller requires the manipulator to have a number of desirable chracteristics these include little or no: backlash, compliance and friction in the drive chain and accurately determined manipulator parameters. These characteristics should be aimed at with research in to the design of direct drive robots by making use of state of the art technology in electric motor and materials.

The other possible direction for research is the idea of combining model-based controllers with adaptive controllers. The idea is for the manipulator to operate under the control of a model based controller when it is unloaded or handling a part of known load(this assumes the parameters of the manipulator are accurately known). When the manipulator is required to pick up parts of unknown load and the model based controller performance significantly degrades then the adaptive control system will be switched in. The adaptive control system will operate until the manipulator unclasps the object of unknown load. Research in to this idea would bring together the desirable chracteristics of both of the control systems.

References

1. Denavit, J., Hartenberg, R.S., "A kinematic notation for lower-pair mechanisms based on matrices.", Journal of Applied Mechanics Vol.22, pp.215-221, 1955.

2. Pieper, D, Roth, B., "The kinematics of manipulators under computer control.", Proceedings of the Second International Congress on theory of Machines and Mechanisms,Vol.2, pp.159-169, 1969, Zakopane, Poland.

3. Paul, R.P., Robot Manipulators: Mathematics, Programming, and Control., The MIT Press Series in Artificial Intelligence, 1981, USA.

4. Universal Machine Intelligence(U.M.I), Inside RTX, U.M.I, 1985, England.

5. Asada, H., Slotine, J.E., Robot analysis and control, John Wiley and Sons Inc., 1986, USA.

6. Craig, J., Introduction to Robotics: mechanics and control, Addison-Wesley Pub. Company, 1986, USA.

7. FU, K.S., Gonzalez, R.C, Lee, C.S.G., Robotics Control, Sensing, Vision, and Intelligence., McGraw-Hill Book Company, 1987, Singapore.

8. Paul, R.P., "Modelling, trajectory calculation and sensing of a computer controlled arm.", Memo AIM 177, Stanford Artificial Intelligence Laboratory, Stanford University, 1972, Palo Alto, California.

9. Bejczy, A.K., "Robot arm dynamics and control", Technical Memo 33-669, Jet Propulsion Laboratory, 1974, Pasadena, California.

10. Luh, J.Y.S., "Conventional controller design for industrial robots-a tutorial.", IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-13, pp.298-316, 1983.

11. An, C.H., Atkeson, C.G., Hollerbach, J.M., "Determination of the effect of feedforward control for trajectory tracking errors(robot arm).", Proceedings 1986 IEEE International Conference on Robotics and Automation, 7th-10th Sept. 1986, San Francisco, California, USA, Pub. 1986 Washington DC, IEEE Computer Society Press, Vol.1, pp.55-60.

12. Kahn, M.E., Roth, B., "The near-minimum-time control of open-loop articulated kinematic chains.", Transactions of the ASME Journal of Dynamic Systems, Measurement, and Control, Vol.93, pp.164-172, 1971.

13. Young, K.D., "Controller design for a manipulator using theory of variable structure systems.", IEEE Transactions on Systems, Man, and Cybernetics, Vol.8, No.2, pp.101-109, 1978.

14. Staszulonek, A., Brussel, H.V., "Inertially decoupled, sliding mode controller design for trace and pickup robot.", Proceedings 16th International Symposium on Industrial Robots- 8th International Conference on Industrial Robot Technology, 3oth Sept.-2nd Oct. 1986, Brussels, Belgium, Pub. 1986 Bedford, BIFS Publications.

15. Harashima, F, Hashimoto, H, Maruyama, K., "Practical robot control of robot arm using variable structure system.", Proceedings 1986 IEEE International Conference on Robotics and Automation, 7-10th April 1986, San Francisco, California, U.S.A, Pub. 1986 Washington DC, IEEE Computer Society Press, Vol.1, pp.532-9.

16. Freund, E., "Fast nonlinear control with arbitrary pole placement for industrial robots and manipulators.", International Journal of Robotics Research, Vol.1, No.1, pp.65-78, 1982.

17. Luh, J.Y.S., Walker, M.W., Paul, R.P., "Resolved acceleration control of mechanical manipulators.", IEE Transactions on Automatic Control, AC-25, No.3, pp.468-474, 1980.

18. Albus, J.S., "A new approach to manipulator controls: The celebellar model articulation controller.", Transactions of ASME Journal of Dynamic Systems, Measurement, and Control, Vol.97, pp.220-227, 1975.

19. Albus, J.S., "Data storage in the cerebellar model articulation controller.", Transactions of ASME Journal of Dynamic Systems, Measurement, and Control, Vol.97, pp.228-233, 1975.

20. Scharf, E., "Fuzzy logic could redefine robot control.", Automation, Robotics, Vol.21, No.2, pp.11-14, 1985.

21. Scharf, E., Mandic, N.J., Mamdani, E.H., "A self-organizing algorithm for the control of a robot arm.", Proceedings of the 23rd International Symposium on Mini and Microcomputers and their applications, Dec. 1983, San Antonio, U.S.A.

22. Whitney, D.E., "Resolved motion rate control of manipulators and human prostheses.", IEEE Transactions Man-Machine Systems, Vol. MMS-10, No.2, pp.47-53, 1969.

23. Dubowsky, S., DesForges, D.T., "The application of model reference control to robotic manipulators.", Transactions ASME Journal of Dynamic Systems, Measurement and Control, Vol. 101, pp.193-200, 1979.

24. Koivo, A.J., Guo, T.H., "Adaptive linear controller for robotic manipulators.", IEEE Transactions Automatic Control, Vol. AC-28, No.1, pp162-171, 1979.

25. Lee, C.S.G, Chung, M.J., "Adaptive perturbation control with feedforward compensation for robot manipulators.", Simulation, Vol.44, No.3, pp.127-136, 1985.

26. Groover, M.P., Weiss, M., Nagel, R.N., Godrey, N., Industrial Robotics: Technology, Programming and applications, McGraw-Hill Book Company, 1986, USA.

27. Think Technology Inc., Light Speed Pascal user's guide and reference manual for the Macintosh, Think Technology Inc., 1986, USA.

28. Apple Computer Inc., Macintosh Plus, Apple Computer Inc., 1986, USA.

29. Doebelin, E.O., Control system principles and design, John Wiley and Son Inc., 1985, USA.

30. Book, W.J., Maizza-Neto, O., Whitney, D.E., "Feedback control of two beam, two joint systems with distributed loads.", ASME Transactions Journal of Dynamic systems, Measurement and Control, Vol.97, No.4, pp.424-431, 1975.

31. Graham, D, Lathrop, R.C, "The synthesis of optimum transient response criteria and standard forms.", Transactions AIEE, Vol.72, part 2, pp.273-288, 1953.

32. Mitchell and Gauthier Associates, Advanced Continuous Simulation Language(A.C.S.L), Concord, 1986, USA.

33. Systems Control Inc., Ctrl-C Users Guide Version 4.0, Systems Control Inc., 1986, USA.

34. Firoozian, R., Report on the computer program simulation of dynamic design of control systems, Mechanical Eng. Departmental Report, University of Aston in Birmingham, 1983, England.

Appendix 1: Notation

Chapter 2

θ_n manipulator joint n, angle

X X position coordinate of the manipulator wrist or end effector with respect to the base or reference XYZ Cartesian frame

Y Y position coordinate of the manipulator wrist or end effector with respect to the base or reference XYZ Cartesian frame

Z Z position coordinate of the manipulator wrist or end effector with respect to the base or reference XYZ Cartesian frame

ϕ_X rotational angle about the X axis Cartesian frame

ϕ_Y rotational angle about the Y axis Cartesian frame

ϕ_Z rotational angle about the Z axis Cartesian frame

${}^nT_o$ A 4 by 4 homogeneous transform relating the coordinate frame n to the base coordinate frame o.

where $${}^nT_o = \begin{bmatrix} n_x & s_x & a_x & p_x \\ n_y & s_y & a_y & p_y \\ n_z & s_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$[\, n_x \; n_y \; n_z \,]^T$ a vector describing the orientation of the X axis of a second coordinate frame with respect to the reference coordinate frame

$[\, s_x \; s_y \; s_z \,]^T$ a vector describing the orientation of the Y axis of a second coordinate frame with respect to the reference coordinate frame

$[\, a_x \; a_y \; a_z \,]^T$ a vector describing the orientation of the Z axis of a second coordinate frame with respect to the reference coordinate frame

$[\, p_x \; p_y \; p_z \,]^T$ a vector describing the origin of a second coordinate frame with respect to the reference coordinate frame

Rot(X,θ) A transform corresponding to a rotation about the X axis an angle θ

$$\text{Rot}(X,\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Rot(Y,θ) A transform corresponding to a rotation about the Y axis an angle θ

$$\text{Rot}(Y,\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Rot(Z,θ) A transform corresponding to a rotation about the Z axis an angle θ

$$\text{Rot}(Z,\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$\text{Trans}(a_n,0,d_n)$ A translation transform corresponding to a translation a_n along the X axis and d_n along the Z axis

$$\text{Trans}(a_n,0,d_n) = \begin{bmatrix} 1 & 0 & 0 & a_n \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_n \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$^{n-1}A_n$ defines the general homogeneous transformation that relates joint n-1 to joint n and is defined as:

$^{n-1}A_n = \text{Rot}(Z,\theta)\ \text{Trans}(0,0,d)\ \text{Trans}(a,0,0)\ \text{Rot}(X,\alpha)$

L_n length of link n

Sθ an abreviation of sinθ

Cθ an abreviation of cosθ

S12 an abreviation of $\sin(\theta_1 + \theta_2)$

C12 an abreviation of $\cos(\theta_1+\theta_2)$

$S\theta C\alpha$ an abreviation of $\sin\theta\cos\alpha$ etc.

$R_{\phi,\theta,\psi}$ Euler angles defining all possible orientations in terms of a rotation ϕ about the Z axis, then a rotation θ about the new Y axis y' and finally a rotation ψ about the new Z axis Z".

$R_{\phi,\theta,\psi}$= Rot(Z,ϕ) Rot(Y,θ) Rot(Z,Ψ)

$\underline{P}$ is the position vector $[X,Y,Z,\phi_x,\phi_y,\phi_z]^T$ defining the coordinates of the manipulator end effector p in terms of the position X,Y,Z and orientation ϕ_x,ϕ_y,ϕ_z with respect to the base coordinate frame

$\underline{P}_\theta$ the joint position vector $[\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6]^T$ of joint angles of the manipulator

$\underline{D}$ a vector of differential displacements $[d_x, d_y, d_z, \delta_x, \delta_y, \delta_z]^T$

where d_x, d_y, d_z are differential displacements along the X,Y,Z axes and

$\delta_x, \delta_y, \delta_z$ are differential rotations about the X,Y,Z axes.

$\underline{D}_\theta$ is a vector of differential joint displacements $[d_{\theta1}, d_{\theta2}, \text{-----}, d_{\theta n}]^T$ of the six joint angles

J defines the 6 by 6 Jacobian matrix, relating the differential displacements in Cartesian space to the differential dispalcements in joint space e.g.

$\underline{D}= J\,\underline{P}_\theta$

where $J_{11}=\frac{\partial X}{\partial\theta_1}$, $J_{21}=\frac{\partial Y}{\partial\theta_1}$ etc.

$\dot{\underline{X}}$ is a velocity vector $[\dot{X},\dot{Y},\dot{Z},\dot{\phi}_x,\dot{\phi}_y,\dot{\phi}_z]^T$

where $\dot{\underline{X}}=(\frac{1}{\Delta t})\,\underline{D}$ and $\dot{X}$, $\dot{Y}$, $\dot{Z}$ are the Cartesian velocities in the X,Y,Z directions and $\dot{\phi}_x, \dot{\phi}_y, \dot{\phi}_z$ are the rotational velocities about the X,Y,Z axes

$\dot{\underline{\theta}}$ is a vector of joint velocities $[\dot{\theta}_1, \dot{\theta}_2, \text{------}, \dot{\theta}_6]^T$

|J| is the determinant of matrix J

adj(J) is the adjoint matrix of J

J^{-1} is the inverse of matrix J

$F_{n-1,n}$ the linear force vector acting at a point O_{i-1} that is the origin of the coordinate frame $O_{i-1},X_{i-1},Y_{i-1},Z_{i-1}$

$F_{n,n+1}$ the linear force vector applied to link i+1 by link i

m_ig the gravity force acting at the centroid of link n, where m_i is the mass of link i and g the vector representing the gravitational acceleration

m_na_n is the inertial force vector and a_n is the acceleration of link n

$T_{n-1,n}$ is the torque applied to link n at joint n-1

$T_{n,n+1}$ is the torque applied to link n+1 at joint n

$r_{n-1,n}$ is the vector distance of force $F_{n-1,n}$ from the centroid

$r_{n,n+1}$ is the vector distance of force $F_{n,n+1}$ from the centroid

ω_n is the angular velocity vector of link n

I_n is the centroidal inertia tensor of link i

$\dot{\omega}_n$ is the rate of change of angular velocity vector of link n

H_{ij} Inertia matrix

h_{ijk} coupling term matrix

G_i gravity term matrix

q_i joint displacement(generalised)

$\dot{q}_i$ joint velocity (generalised)

$\ddot{q}_i$ joint acceleration (generalised)

L Lagrangian,i.e. the difference between the total kinematic energy of a system and the total potential energy of a system

K total kinematic energy of a system

P total potential energy of a system

F_i generalised force or torque of the system with respect to coordinate displacement

L_n length of link n

θ_n angular displacement of link n

M_n mass of link n

T_n torque applied to link n at joint n-1

K_{en} kinematic energy of link n

V_n resultant linear velocity of link n

ω_{on} angular velocity of link n

ω_{mn} angular velocity of link n referred to the motor shaft

A_n the position of the centre of gravity of link n

T_p tangential force of link at joint 1

R_p perpendicular force of link at joint 1

D_{ii} is the effective inertia at joint i, with an acceleration at joint i, causing a torque at joint i equal to $D_{ii}\ddot{\theta}_i$

D_{ij} is the coupling inertia between joints i and j, with an acceleration at joint i or j causing a torque at joint i or j equal to $D_{ij}\ddot{\theta}_j$

$D_{ijj}\dot{\theta}_j^2$ is the centripetal force acting at joint i, due to the velocity at joint j

$D_{ijk}\dot{\theta}_j\dot{\theta}_k$ is the Coriolis force acting at joint i, due to velocities j and k

D_i is the torque representing the gravity forces at joint i

$M(\theta)$ is an n by n inertia matrix

$V(\theta,\dot{\theta})$ is an n by 1 vector of centripetal and Coriolis terms

$G(\theta)$ is an n by 1 vector of gravity terms

Chapter 4

α the computed inertia gain vector

β the vector of the computed non linear terms that are fedforward

E absolute error between desired angular position and actual angular position i.e. $E=\theta_d-\theta_o$

$\dot{E}$ is the absolute velocity error i.e $\dot{E}=\dot{\theta}_d-\dot{\theta}_o$

K_{In} is the integral gain of joint n motion controller

K_{Pn} is the position gain of joint n motion controller

K_{Vn} is the velocity gain of joint n motion controller

T' is the torque generated by the servo portion of the computed torque controller

$I(\theta)$ diagonal inertia matrix

$I_o(\theta)$ off diagonal inertial coupling matrix

$I_c(\theta)$ computed diagonal inertia matrix

$I_{oc}(\theta)$ computed off diagonal inertial coupling matrix

S Laplace transform operator

C ratio between $I_c(\theta)/I(\theta)$

ω_n natural frequency of oscillation

ζ damping ratio

Chapter 5

td_{jk} the time period of segment jk

t_{jk} the linear portion of segment jk i.e. that at a constant velocity

t_i the time duration for the blend region at corner point j

X_j X Cartesian position at corner point j (also P_x)

Y_j Y Cartesian position at corner point j (also P_y)

$\dot{X}_{jk}$ the velocity along the X axis between corner points j and k

$\dot{Y}_{jk}$ the velocity along the Y axis between corner points j and k

$\ddot{X}_j$ the acceleration along the X axis at corner point j

$\ddot{Y}_j$ the acceleration along the Y axis at corner point j

S_i reference to corner point i

$XV_{jk}(i)$ $\dot{X}_{jk}$ the velocity in the X axis between corner points j and k

$YV_{jk}(i)$ $\dot{Y}_{jk}$ the velocity in the Y axis between corner points j and k

XA(i) $\ddot{X}_i$ the acceleration in the X axis at corner point i

YA(i) $\ddot{Y}_i$ the acceleration in the Y axis at corner point i

$TXY_{jk}(i)$ t_{jk} the time of the path at constant velocity between corner points j and k

TXY(i) t_i the time of the blend period at corner point i

V_R the resultant velocity i.e. $\sqrt{\dot{X}_{jk}^2 + \dot{Y}_{jk}^2}$

A_R the resultant acceleration i.e. $\sqrt{\ddot{X}_j^2 + \ddot{Y}_j^2}$

TH1 the angular displacement of joint 1 , link 1

TH2 the angular displacement of joint 2, link 2

TH1V the angular velocity of joint 1, link 1

TH2V the angular velocity of joint 2, link 2

TH1A the angular acceleration of joint 1, link 1

TH2A the angular acceleration of joint 2, link 2

TL_1 the torque at load side of the gearbox for link 1

TL_2 the torque at load side of the gearbox for link 2

Tm_1 the torque at motor side of the gearbox for link 1

Tm_2 the torque at motor side of the gearbox for link 2

$T_{L1}A_1$ $D_{11}\ddot{\theta}_1$ inertial torque axis 1

$T_{L1}A_3$ $D_{12}\ddot{\theta}_2$ inertial coupling torque axis 1

$T_{L1}A_4$ $D_{122}\dot{\theta}_2^2$ centripetal coupling torque axis 1

$T_{L1}A_5$ $D_{112}\dot{\theta}_1\dot{\theta}_2$ Coriolis coupling torque axis 1

$T_{L2}A_1$ $D_{22}\ddot{\theta}_2$ inertial torque axis 2

$T_{L2}A_3$ $D_{12}\ddot{\theta}_1$ inertial coupling torque axis 2

$T_{L2}A_4$ $D_{211}\dot{\theta}_1^2$ centripetal coupling torque axis 2

$$T_{L1}A_1(\%) = \frac{abs(T_{L1}A_1)}{\sum_1^5 abs(T_{L1}A_n)} * 100\ \%$$

Chapter 6

J_{eff1} the effective inertia of axis 1 referred to the motor side of the gearbox

e.g $J_{eff1}=(J_{A1}+J_{m1}+n_1^2D_{11})$

where J_{A1} and J_{m1} are the actuator and motor fittings

and $n_1^2D_{11}$ is the reflected inertia of link 1 at actuator 1 through gearbox 1

V_f voltage applied to the direct current motor at the field windings

I_f current through field windings of direct current motor

R_f resistance of foeld windings of direct current motor

L_f inductance of field windings of direct current motor

V_a voltage applied to armature of direct current motor

I_a current through armature of direct current motor

R_a resistance of armature windings of direct current motor

L_a magnetic flux constant = $K_f I_f$

B viscous damping coefficient

T_m motor time constant for a field controlled d.c. motor i.e T_m=J/B

(or RJ/K_tK_b for an armature controlled d.c. motor)

T_f field time constant for a field controlled d.c. motor i.e. T_f=L_f/R_f

K_t motor constant for an armature controlled d.c. motor

K_b back emf constant for an armature controlled d.c. motor

V_b back emf voltage for an armature controlled d.c. motor

T_a armature time constant for an armature controlled d.c motor

γ damping factor for an armature controlled d.c. motor

T_d test disturbance load torque

θ_r reference or desired angular position input

K_{eff} the effective stiffness of joint of the manipulator

J_{eff} the effective inertia of joint of the manipulator

ω_o natural frequency of ITAE index standard forms

ω_{nn} normalised natural frequency

ω_s structural resonant frequency

ω_n natural frequency or desired

$B(\theta)$ a matrix of Coriolis coefficients of dimensions $n*n(n-1)/2$

$C(\theta)$ a matrix of centipetal coefficients