CODEN: LUTFD2/(TFRT-5396)/1-55/(1989)

Compensation of Disturbances Caused by Coupled Mass Inertia

Kenneth Olsson

Department of Automatic Control
Lund Institute of Technology
February 1989

Department of Automatic Control
Lund Institute of Technology
P.O. Box 118
S-221 00 Lund Sweden

Document name
Master thesis

Date of issue
February 1989

Document Number
CODEN: LUTFD2/(TFRT-5396)/1-55/(1989)

Author(s)
Kenneth Olsson

Supervisor
Torgny Brogård, ABB and Lars Nielsen, LTH

Sponsoring organisation

Title and subtitle
Compensation of disturbances caused by coupled mass inertia.

Abstract

In today's robots with large working ranges and high flexibility, large variations in the robot dynamics are obtained. Transient disturbances caused by rapid changes in acceleration or retardation causes transient trajectory errors.

In this report transient disturbances caused by coupled mass inertia is studied and minimized by means of both simulations and experiments.

Two alternatives for decreasing the path deviation caused by the transient disturbances have been considered. The first one is to control the derivative of the acceleration in such a way that the frequency range of the disturbance is low enough to achieve a sufficient controller stiffness. The second alternative is to make torque feedforward from the disturbing axis to the disturbed axis.

Velocity feedforward has also been studied in order to decrease the controller error in the position controller but also to increase the performance of the robot.

The simulations and the experimental results indicate that it should be possible to increase the performance of a robot, both concerning path accuracy and average velocity.

Key words
Robotics, path deviation, disturbances, torque feedforward, velocity feedforward, acceleration derivative control, coupled mass inertia.

Classification system and/or index terms (if any)

Supplementary bibliographical information

ISSN and key title

ISBN

Language
English

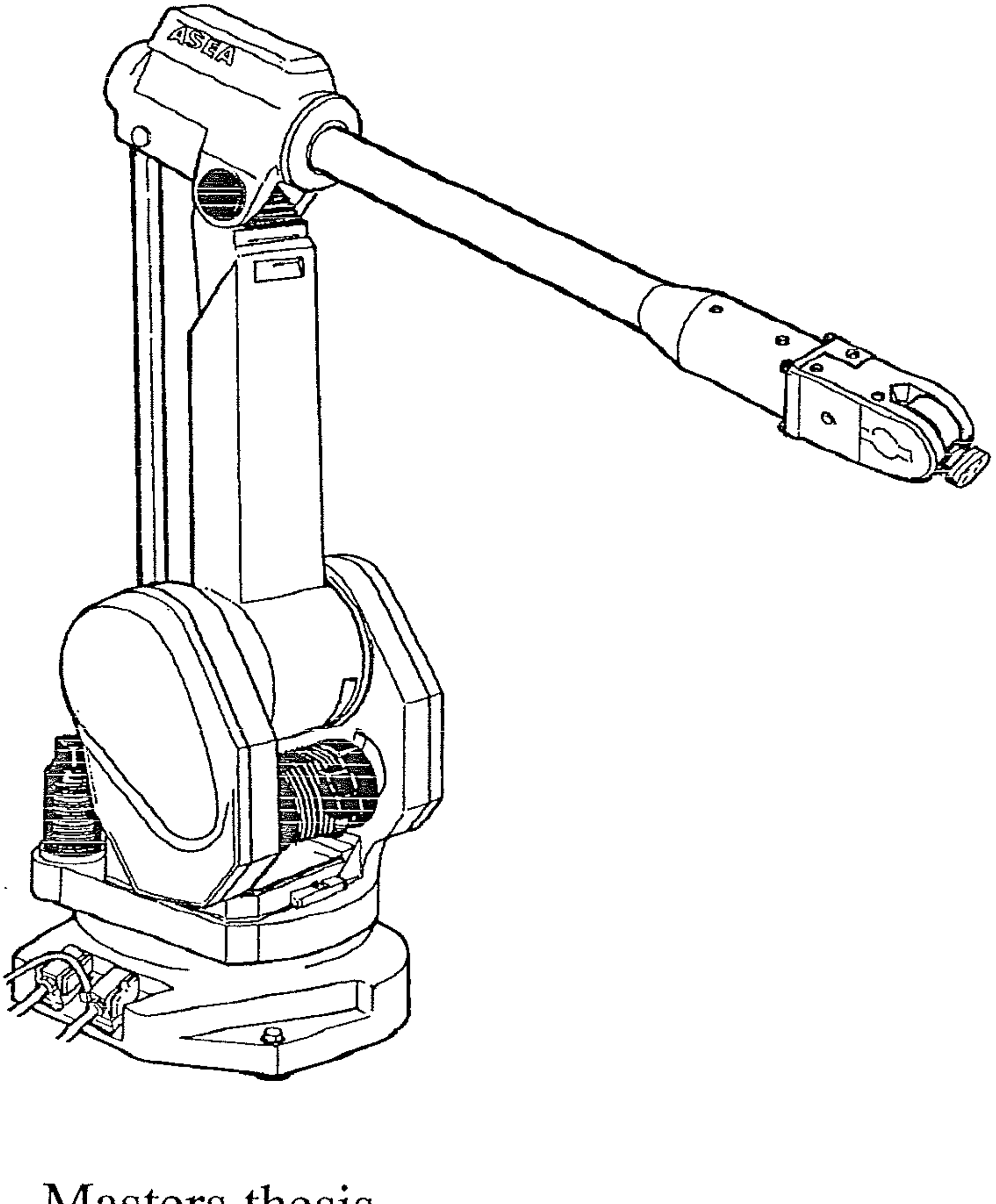
Number of pages
55

Recipient's notes

Security classification

The report may be ordered from the Department of Automatic Control or borrowed through the University Library 2, Box 1010, S-221 03 Lund, Sweden, Telex: 33248 lubbis lund.

Compensation of disturbances caused by coupled mass inertia.



Masters thesis

by Kenneth Olsson

Table of Contents

Chapters

Chapter 1.

Introduction

In this paper torque disturbances on robot axes have been studied by means of both simulations and experiments. Torque disturbances can be caused by gravitational forces, centrifugal forces, coriollis forces, coupled mass inertia, motor torque ripple and tool forces.

When the disturbance is of a low frequency characteristic, the stiffness of a PI controller is usually big enough to cancel the disturbance without reducing the path tracking accuracy.
However, transient disturbances with a high frequency characteristic can easily give rise to transient path deviations since the bandwidth of the controller is not large enough to counteract the disturbance. This is often the case with coupled mass inertia, motor torque ripple and tool forces.

The studies in this paper have been concentrated on coupled mass inertia since this is a significant problem for modern flexible multi purpose robots as IRB 2000, IRB 3000 and IRB 3200 from ABB Robotics. The coupled inertia can be seen as a reaction force from an accelerating axis.

The disturbance torque is proportional to the acceleration value of the disturbing axis and the larger the derivative of the acceleration, the higher the frequency content of the disturbance.
With a higher frequency content the path deviation increases, since the stiffness of the controller usually decreases with the frequency.

In this paper two alternatives for decreasing the path deviation have been considered.
The first one is to control the derivative of the acceleration in such a way that the frequency range of the disturbance is low enough to achieve a sufficient controller stiffness. The second alternative is to make a model based torque feedforward from the disturbing axis to the disturbed axis.

Chapter 2

Summary

This Masters Thesis deals with the problem of disturbances caused by coupled mass inertia. The studies in this paper have been concentrated on axis two and three in robot IRB 2000 from ABB Robotics.

The masters thesis contain the following activities:
* Creating a realistic computer model of the robot with its control system.
* Verifications on a robot.
* Construction of computer models in order to decrease disturbances caused by coupled mass inertia.
* Simulations.
* Robot implementation.
* Analysis of achieved results.

Computer modules describing the robot with its control system existed at ABB Robotics. These modules have been modified and new ones have been developed to create a realistic simulation model which could be used for synthesis and analysis.

To control the acceleration derivative in such a way that the frequency range is low enough to achieve sufficient controller stiffness able to prevent disturbances caused by the coupled mass inertia, the following methods were developed:

Acceleration Derivative Control, ADC:
* ADC of speed reference in the position loop
* ADC of position reference
* ADC in path generation

To prevent disturbances by means of torque feedforward, TFF, the following methods were developed:

Torque Feedforward, TFF
* Non-linear resonance TFF
* Linear resonance TFF
* Non-linear rigid body TFF
* Linear rigid body TFF

The rigid body TFF was implemented in the robot controller S3 and tested on IRB 2000.
To achieve high velocity performance simultaneously with low disturbance level, the following techniques were studied:

Velocity Feedforward, VFF
* VFF from the path generation
* VFF with a filtered position reference.

To study the disturbances a larger number of simnon simulations have been carried out. The simulations have been correlated with robot verifications to achieve realistic results.

For the friendly and qualified support in my Masters thesis, I want to thank my supervisor at ABB Robotics, Torgny Brogård but also Henrik Knobel who supported me with the computer simulations in Simnon, Staffan Elving who supported me with the robot implementation and other fellow workers at ABB Robotics. I also want to thank my supervisor at Lund Institute of Technology, Lars Nielsen, Department of Automatic Control.

Kenneth Olsson

Chapter 3

GENERAL ROBOTICS

Chapter 3.

GENERAL ROBOTICS

An industrial robot is a mechanical device that can be programmed to perform a wide variety of applications.

In order to describe the position and orientation off a body in space we attach a coordinate system, or frame, rigidly to the object. We then proceed to describe the position and orientation of this frame with respect to some reference coordinate system.

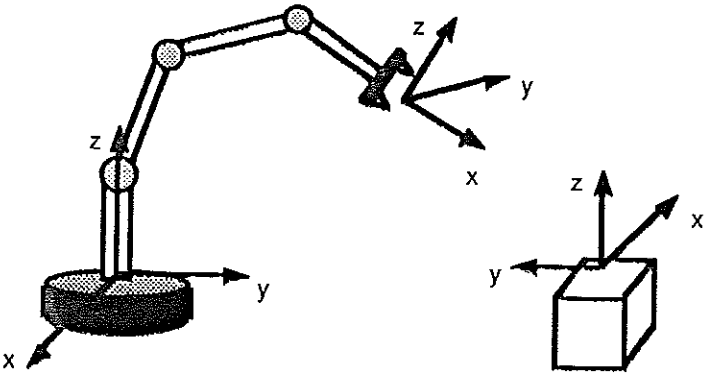


Figure 3.1. Coordinate system.

For convenience, the point whose position we describe, is chosen as the origin of the body-attached frame. The position and orientation pair is an entity called a frame, which is a set of four vectors giving position and orientation information. One vector locates the finger tip position and three more describe its orientation.

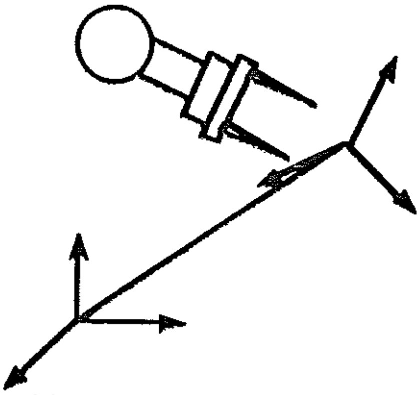


Figure 3.2. Locating an object in position and orientation.

Trajectory generation.

A common way of causing a manipulator to move smoothly between two positions is to cause each joint to move as specified by a smooth function of time. Commonly, each joint starts and ends its motion at the same time so that the manipulator motion appears coordinated. Exactly now to compute these motion function is the problem of trajectory generation. Often a path is described not only by a desired destination but also by intermediate locations, or via points, through which the manipulators must pass en route to the destination. The technique in which only some points in the trajectory are of importance, is called a rendez- vous problem. With a coordinated motion function, the rendez- vous problem may be extended, to a technique where every point in the trajectory is of importance. The designated trajectory has to be followed exactly by the robot manipulator.

Chapter 3.

GENERAL ROBOTICS

An industrial robot is a mechanical device that can be programmed to perform a wide variety of applications.

In order to describe the position and orientation off a body in space we attach a coordinate system, or frame, rigidly to the object. We then proceed to describe the position and orientation of this frame with respect to some reference coordinate system.

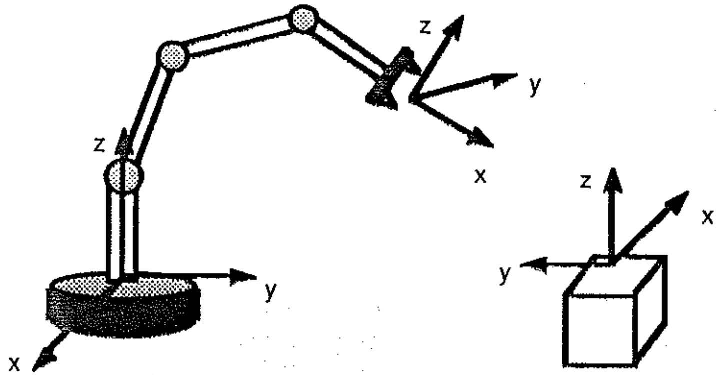


Figure 3.1. Coordinate system.

For convenience, the point whose position we describe, is chosen as the origin of the body-attached frame. The position and orientation pair is an entity called a frame, which is a set of four vectors giving position and orientation information. One vector locates the finger tip position and three more describe its orientation.

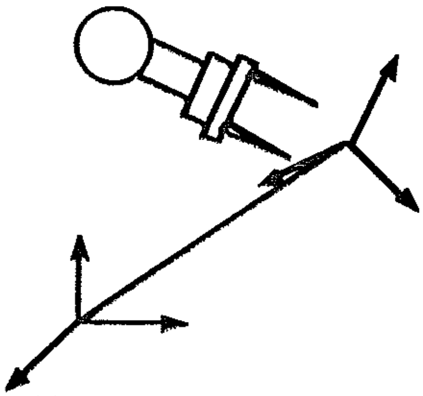


Figure 3.2. Locating an object in position and orientation.

Trajectory generation.

A common way of causing a manipulator to move smoothly between two positions is to cause each joint to move as specified by a smooth function of time. Commonly, each joint starts and ends its motion at the same time so that the manipulator motion appears coordinated. Exactly now to compute these motion function is the problem of trajectory generation. Often a path is described not only by a desired destination but also by intermediate locations, or via points, through which the manipulators must pass en route to the destination. The technique in which only some points in the trajectory are of importance, is called a rendez- vous problem. With a coordinated motion function, the rendez- vous problem may be extended, to a technique where every point in the trajectory is of importance. The designated trajectory has to be followed exactly by the robot manipulator.

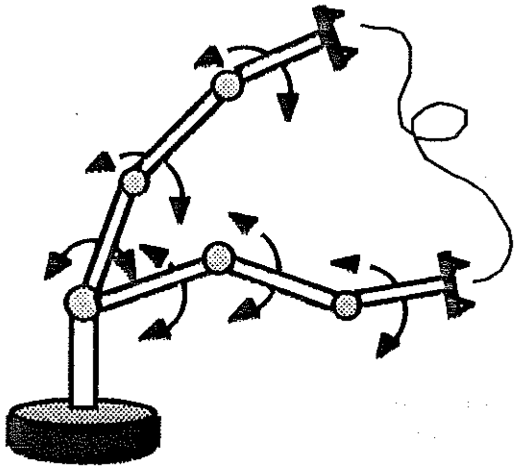


Figure 3.3. In order to cause the manipulator to follow the desired trajectory, a position control system must be implemented.Such a system uses feedback from joint sensors to keep the manipulator on course.

Workspace

When a manipulator has less than six degrees of freedom, it cannot attain general goal positions and orientations in the three dimensional (3D- space). Roughly speaking, workspace is that volume of space which the end effector of the manipulator can reach. The reachable workspace is that volume of space which the robot can reach in at least one orientation. Dexterous workspace - is that volume of space which the robot end effector can reach with all orientations. The dexterous workspace is a subset of the reachable workspace.

Example
Consider the workspace of the 2-link manipulator in the figure 3.4 below.

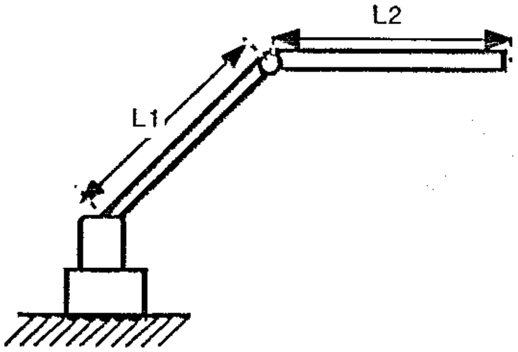


Figure 3.4. Two link manipulator with link lengths L1 and L2.

If L1=L2 the reachable workspace consists of a disc of radius 2*L1.
The dexterous workspace consists of only a single point, the origin. If L1≠L2 then there is no dexterous workspace, and the reachable workspace becomes a ring of outer radius L1+L2 and an inner radius of | L1-L2|. When the joint limits are limited by the actual mechanisms, the workspace is obviously correspondingly reduced.

Kinematics

Kinematics is the science of motion, which treats motion, without regard to the forces which cause it. Within the science of kinematics one studies the position, velocity, acceleration, and all higher derivatives of the position variables.

Inverse Kinematics.

Given the desired position and orientation of the tool relative to the station, we now compute the set of joint angels which will achieve this desired result. The solution to the problem of finding the required joint angles to place the tool frame relative to the station frame, is split into two parts. First, frame transformations are performed to find the wrist frame relative to the base frame, and then the inverse kinematics are used to solve for joint angels.

Multiple solutions.

A planar arm with three resolute joints has a large dexterous workspace in the plane, since any position in the interior of its workspace could be reached with any orientation.

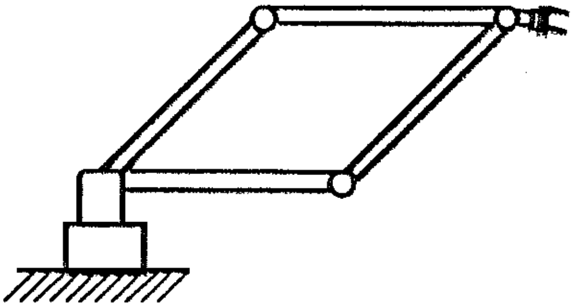


Figure 3.5. Multiple solutions.

The fact that a manipulator has multiple solutions may cause problems because the system has to be able to choose one.

Singularities

For this purpose, the singularities are classed into two categories.

1. Workspace boundary singularities are those which occur when the manipulator is fully stretched out or folded back on itself such that the end-effector is near or at the boundary of the workspace.
2. Workspace interior singularities are those which occur away from the workspace boundary and generally are caused by two or more joints axes lining up.

Mass distribution

The inertia tensors have to be defined as a distribution of mass of a rigid body relative to a reference frame. The inertia tensor relative to the reference frame can be expressed in matrix form as a 3*3 matrix.

$$J = \begin{bmatrix} Jxx & -Jxy & -Jxy \\ -Jxy & Jyy & -Jyz \\ -Jxz & -Jyz & Jzz \end{bmatrix}$$

where the scaler elements are given by:

$J_{xx} = \iiint_V (y^2 + z^2)*\rho \ dv$ $\qquad J_{xy} = \iiint_V x*y*\rho \ dv$

$J_{yy} = \iiint_V (x^2 + z^2)*\rho \ dv$ $\qquad J_{xz} = \iiint_V x*z*\rho \ dv$

$J_{zz} = \iiint_V (x^2 + y^2)*\rho \ dv$ $\qquad J_{yz} = \iiint_V y*z*\rho \ dv$

The rigid body is composed of different volume elements dv containing material density ρ.

In cases where a simplification of the 3- dimensional inertia tensor in the one dimensional case, the inertia is calculated as:

$$J = \int l^2 *\rho \ dv$$

Coupled mass inertia

The coupled mass inertia causes disturbances between links at high accelerations in mechanical constructions. In figure 3.6, the coupled mass inertia is shown. The equivalent mass of the upper link has its mass centre at a length L3 from the centre of rotation. The length of the lower link is L2 and the length from the rotation centre of the lower link to the centre of the equivalent mass of the upper link is L.

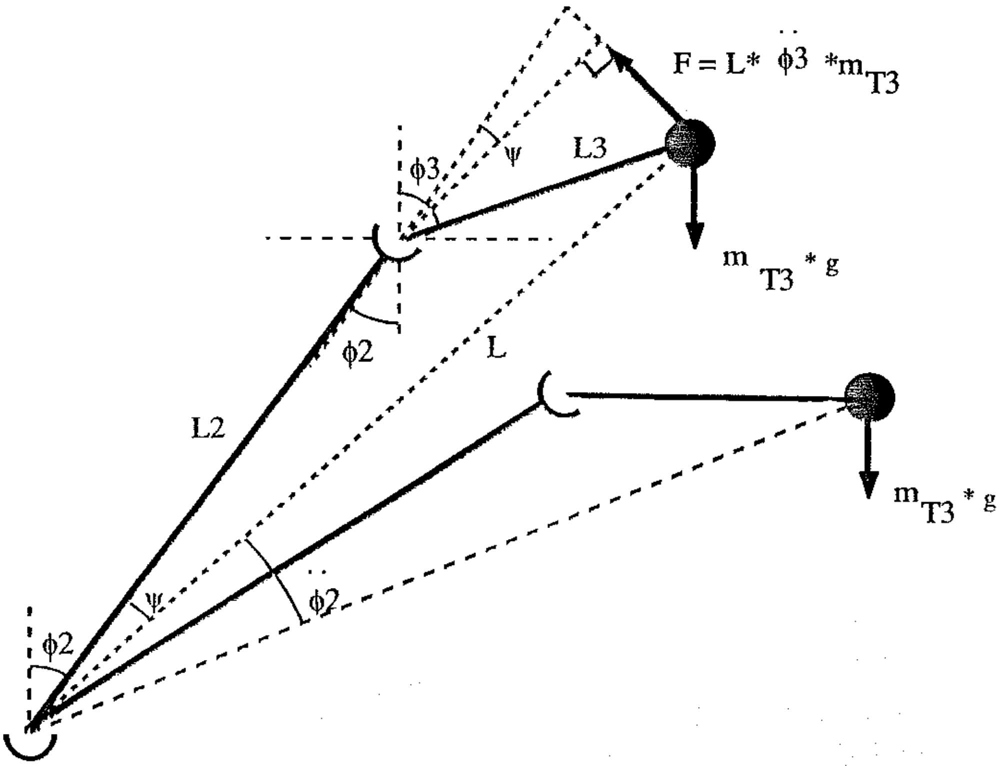


Figure 3.6. Coupled mass inertia of a simple robot structure.

The force F working on link 3, caused by the acceleration on link 2, is proportional to length L , the mass m_{T3} and the acceleration.

$$F = L*\ddot{\phi}2*m_{T3}$$

The torque M, caused by the force F on axis 3, is dependent on the length L3.

$$M = F*L_3*\cos(\phi_3 - \phi_2 - \psi)$$

The coupled mass inertia is by definition

$$M = J_{23} * \ddot{\phi}_2$$

The coupled inertia the can be calculated as

$$J_{23} = L*m_{T3}*L_3* \cos(\phi_2 - \phi_3 - \psi)$$

The coupled mass inertia between axis 2 and 3 is of special interest in this thesis and varies with different arm positions in the robot.
Figure 3.7 below shows the maximum working range for axis 3.

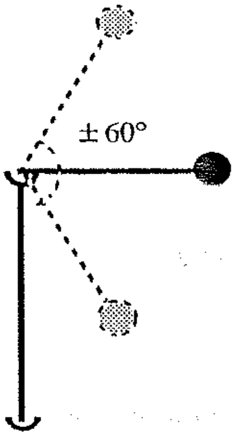


Figure 3.7. Working range of IRB 2000.

Data for IRB- 2 000

L_2	= 0.8 m
L_3	= 0.2 m
m_{T3}	= 54 kg

The length from the centre of rotation to the centre of gravity on link 3 is with a minimal angle (-60°) L_{min} = 0.635 m and at the maximum angel (+ 60°) L_{max} = 0.973 m.

The coupled mass inertia between axis 2 and 3 in IRB- 2000 varies between -0.0012 and 0.0019 $kg*m^2$.

Disturbances caused by coupled mass inertia.

Torque disturbances can be caused by gravity forces, centrifugal forces, coupled mass inertia, motor torque ripple and tool forces. Transient disturbances with a high frequency characteristic, can easily give rise to transient path deviations since the bandwidth of the robot control system is not big enough to cancel the disturbance.
Coupled mass inertia on an axis in a robot system, usually affects the disturbed axis with a position error in the given trajectory.

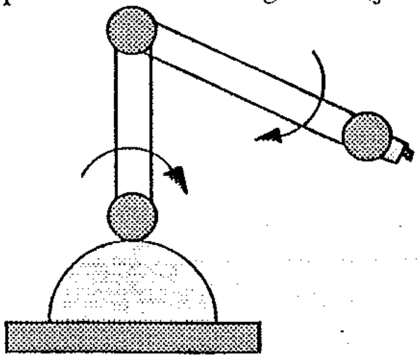


Figure 3.8. Simple robot model.

In the figure the lower robot arm influences the upper. The robot makes a circular movement around axis two. A forward acceleration of the lower arm forces the upper arm into a less horizontal position.
When the acceleration is slow, the controller manages to counteract the force and a trajectory error is avoided.

ABB-ROBOTS

ABB's IRB-2000 industrial robot, is an electrical-computer-based system for general robot applications.

The robot system primarily consists of two parts:
* The mechanical robot
* The robot control system.

The mechanical robot performs the movements commanded by the operator with support from the robot control system. The robot control system contains a programming unit, control panel, servo amplifiers and control electronics. Programming and manual operation, are performed via a portable programming unit and the operator's panel on the control cabinet.

The control system contains four microprocessors:
- main computer : for overall control
- servo computer : control of servo functions and robot movement
- axis computer : for individual control of robot axes.
- I/O computer : controls the communication with operators unit, peripheral equipment, host computer and floppy disc unit.

Programming unit

All operator communication, with the exception of selection of the operational mode for the robot system, are available on the portable programming unit.

The programming unit contains:
* A two-row alphanumeric display for messages in plain language.
* Joystick; The robot, or external axes, are positioned using the joystick together with safety pad and switches.

The programming unit, when not in use is placed is placed in a compartment on the door of the control cabinet or removed from the system.

Manual operation

The robot can move in the following coordinate systems:
* Rectangular base-oriented coordinate system (fixed coordinate system)
* Axis oriented coordinate system. (fixed coordinate system)
* Rectangular wrist-oriented coordinate system. (moving coordinate system)
* Rectangular tool-oriented coordinate system. (moving coordinate system)

In the rectangular coordinate system the trajectory calculations are based upon the arm position in a rectangular coordinate system. In the robot coordinate system the arm position is calculated from the axes angles in a Cartesian coordinate system. The Figures. 3.9 and 3.10, illustrate the difference.

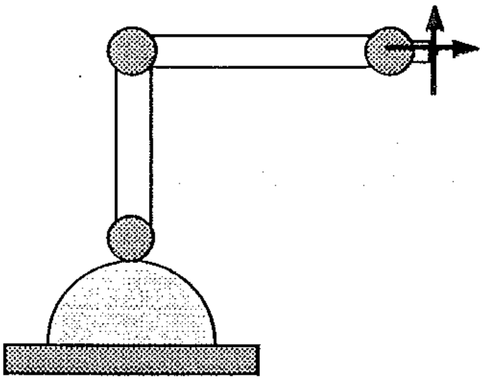


Figure 3.9. A robot in a rectangular coordinate system

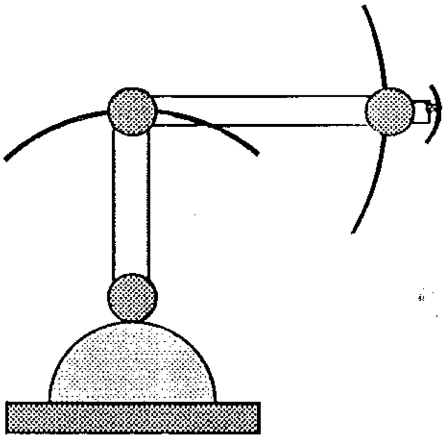


Figure 3.10. A robot in a robot coordinate system

Automatic operation

When the system is executes a program the robot will move from its current position to a programmed position stored in the memory as the position of the tool centre point, TCP.

TCP

Each position is programmed by the operator moving the robot to the desired position by manual operation. A specific point, known as the working point, or the tool centre point (TCP), can be defined in any selected position. The movements are made in an orientation system of the wrist or the tool.

Three different types of coordinate systems in automatic control are possible.

* Robot coordinate movement
All axes are individually moved at constant speed towards the programmed position. Any reorientation of the wrist will be performed continuously during the movement.

* Straight-line movement
The TCP moves in straight line and at constant speed towards the programmed position. Any reorientation of the wrist will be performed continuously during the movement.

* Modified straight-line movement
As straight- line movement but allowing a controlled deviation in the wrist orientation when passing a singular point.

Technical specification

A sample of the technical specification concerning the performance of the ABB industrial robot IRB-2000 follows.

Axis 1 Rotation movement	+180° - -180°	115°/sec
Axis 2 Arm movement	+100° - -110°	115°/sec
Axis 3 Arm movement	+ 60° - -110°	115°/sec
Axis 4 Wrist movement	+200° - -200°	280°/sec
Axis 5 Bend movement	+120° - -120°	300°/sec
Axis 6 Turn movement	+200° - -200°	300°/sec
Handling capacity	10 kg	

Movement structure

The robot studied, IRB-2000 has six degrees of freedom. Every degree of freedom corresponds to
In the robot a degree of freedom is represented with movements around an axis.
The robot movement can be briefly described as follows (see also the figure below).

Axis 1
Turning of the complete mechanical robot arm system.

Axis 2
Forward and reverse movement of the lower arm.

Axis 3
Up and down movement of the upper arm.

Axis 4
Turning the complete wrist centre.

Axis 5
Bending of wrist around the wrist centre.

Axis 6
Turning of mounting plate (tum disc).

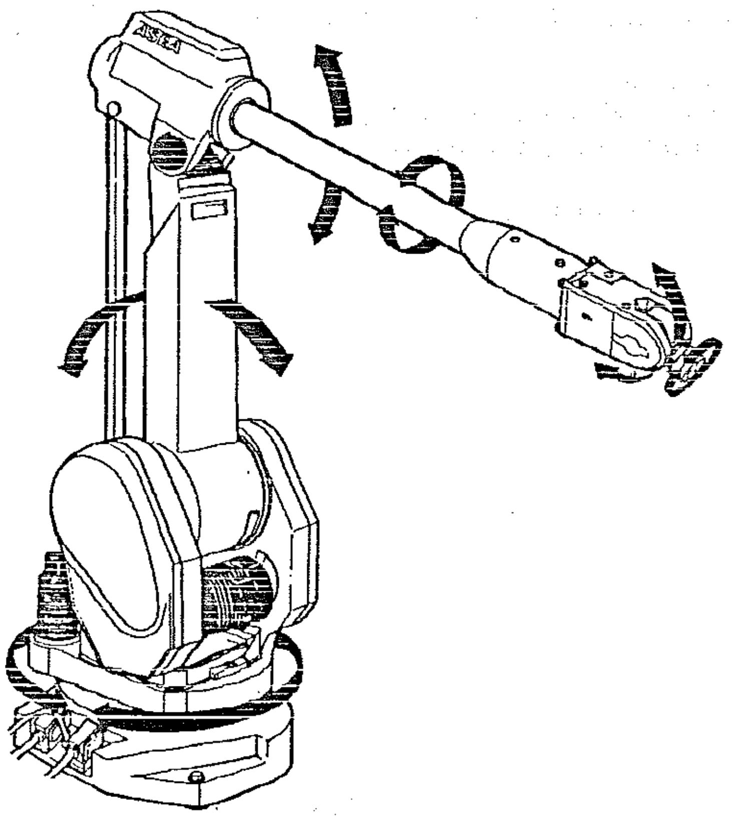


Figure. Movement structure.

Chapter 4.

Robot models

4.1 Linear rigid body model

4.3 Linear rigid body model with viscous damping

4.4 Non-linear rigid body model

4.6 Linear resonance robot model

4.8 Non-linear resonance robot model

Chapter 4.

ROBOT MODELS

A robot is a non linear system, containing different non-linearities.
Two basic non linearities are the backlash and the band of friction that a robot arm has to pass through.

The analysis of one non linearity can be carried out using, for example a describing function. With several non-linearities this becomes a complex problem. Therefore, it is essential to use simulation techniques in the analysis work.

The analysis of the robot can be made in five steps. In the first, the simplest model possible with no non-linearity is studied. The last step treats the problem of a complete robot model, with static and dynamic friction, rigidity, viscous damping and backlash.

Models studied:

1. Linear rigid body model
2. Linear rigid body model with viscous damping
3. Non-linear rigid body model
4. Linear resonance robot model
5. Non-linear resonance robot model

The stability characteristics were studied for every robot model.

Linear rigid body model

The rigid body model has a rigid axis between the motor and the arm.
The most simple rigid body model is a model of a robot axis without any kind of friction, backlash or viscous damping. The two masses, on the motor and on the axis are represented in with inertia J1 and J2. See figure below.

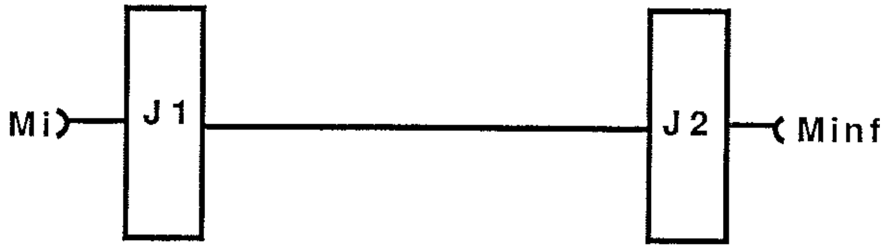


Figure 4.1. A simple rigid body model of a robot axis.

J1	:	Motor inertia
J2	:	Axis inertia
Mi	:	Input torque
Minf	:	Axis torque caused by coupled inertia (influence)

The arm acceleration is proportional to the input torque on the motor.

$$\ddot{s}\,(J1+J2) = Mi - Minf$$

In this model with a rigid axis the motor torque and the axis torque, the velocities and positions of inertia 1 and 2 are identical.

A state space transformation of the state variables s1 and s2 to x gives:

$\ddot{x} = (Mi - Minf)/(J1+J2)$

With the state space variables

$x1 = s$ and $x2 = \dot{x}1 = \ddot{s}$

a system on the form

$$\dot{X} = A^*X + B^*U$$

is achieved, which results in:

$$\dot{x}1 = x2$$

$$\dot{x}2 = (Mi-Minf)/(J1+J2)$$

Written in matrix form

$$\dot{X} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} {}^* X + \begin{bmatrix} 0 & 0 \\ 1/(J1+J2) & -1/(J1+J2) \end{bmatrix} \begin{bmatrix} Mi \\ Minf \end{bmatrix}$$

The stability of the system is given by the poles of the A-matrix.
This system has two poles in the origin.

Linear rigid body model with viscous damping.

This rigid body model is a model of a robot axis without any kind of non-linearities (friction or backlash) but with viscous damping, which is the simplest extension of the simplest rigid body model.
The two masses, on the motor and on the axis, are represented by inertia J1 and J2. See figure below.

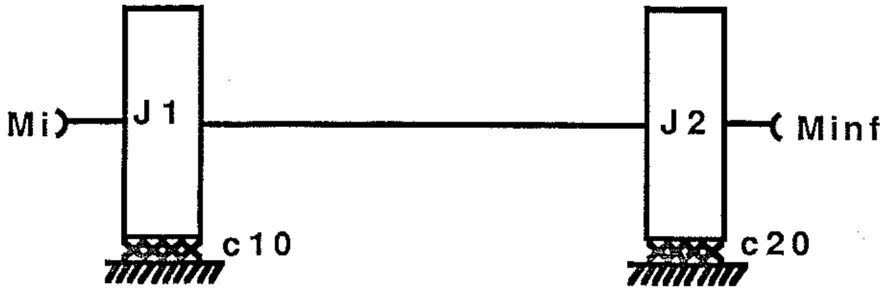


Figure 4.2. Rigid body model of a robot axis.

J1	:	Motor inertia
J2	:	Axis inertia
Mi	:	Input torque
Minf	:	Axis torque caused by coupled inertia. (influence)
c10	:	Viscous damping, inertia 1.
c20	:	Viscous damping, inertia 2.

The positions s1 and s2 are both identical, as in the linear rigid body model.

The differential equation describing this system states:

$$\ddot{s}(J1 + J2) = Mi - Minf - (c10 + c20)*\dot{s}$$

The state space transformation of the state variables s1 and s2 to x , with the state space variables

$x1 = s$ and $x2 = \dot{x}1 = \dot{s}$

gives the system

$$\dot{x}1 = x2$$

$$\dot{x}2 = -((c10 + c20)/(J1 + J2))*x2 + (Mi-Minf)/(J1+J2)$$

Compare the result with system on the form

$$\dot{X} = A*X + B*U$$

$$\dot{X} = \begin{bmatrix} 0 & 1 \\ 0 & -(c10+c20)/(J1+J2) \end{bmatrix} * X + \begin{bmatrix} 0 & 0 \\ 1/(J1+J2) & -1/(J1+J2) \end{bmatrix} \begin{bmatrix} Mi \\ Minf \end{bmatrix}$$

In this model one pole in the A-matrix will be negative , but the other one still stays in the origin.

Non-linear rigid body model

The non-linear rigid body model contains friction and viscous damping. The backlash is of less interest with a rigid axis. The two masses, on the motor and on the axis are represented by inertia J1 and J2.

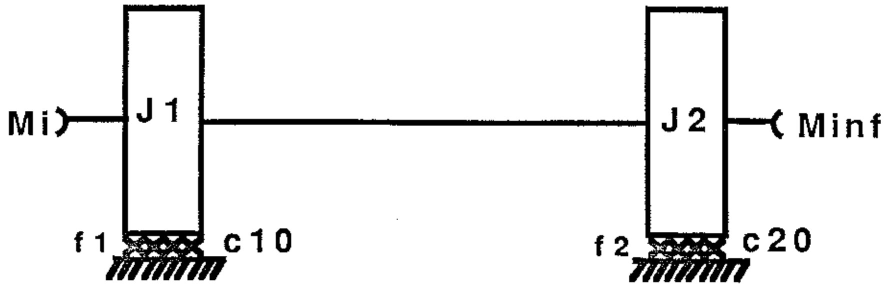


Figure 3.3. A simple rigid body model of a robot axis.

J1	:	Motor inertia
J2	:	Axis inertia
Mi	:	Input torque
Minf	:	Axis torque caused by coupled inertia.(influence)
c10	:	Viscous damping, inertia 1.
c20	:	Viscous damping, inertia 2.
f1	:	Friction, inertia 1.
f2	:	Friction, inertia 2.

The differential equation describing the robot model:

$$\ddot{s}(J1+J2) = \left[\begin{array}{ll} 0 & ;\ |M1| < f1+f2 \\ Mi - Minf - (c10+c20)*\dot{s} - (f1+f2)*SIGN(\dot{s}) & ;\ |M1| > f1+f2 \end{array} \right.$$

where

$M1 = Mi - Minf - (c10+c20)*\dot{s}$

The non-linearity in this model

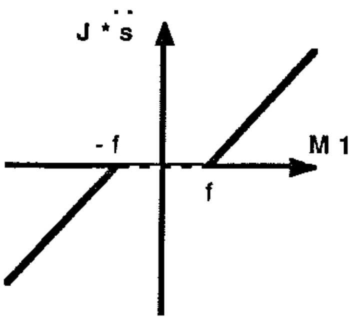


Figure 4.4. Non linear model.

J	=	J1 + J2
f	=	f1 + f2

A state space transformation of the state variables s1 and s2 to x, with the state space variables $x1 = s$ and $x2 = \dot{x}1 = \ddot{s}$

gives

$$\dot{x}1 = x2$$

$$\dot{x}2 = -((c10 + c20)/(J1 + J2))*x2 + (Mi-Minf)/(J1+J2)$$

In matrix form:

$$\dot{X} = \begin{bmatrix} 0 & 1 \\ 0 & -c/J \end{bmatrix} * X + \begin{bmatrix} 0 \\ 1 \end{bmatrix} *(1/J)*(Mi - Minf - f1 - f2)$$

c = c10 + c20
J = J1 + J2

This system has one pole in the origin and one on the negative real axis.

Linear resonance robot model

The most simple linear resonance robot model contains the two inertia, dynamic friction and a spring with rigidity k1. The model does not contain any backlash or any static friction. A simple figure of this robot is shown below.

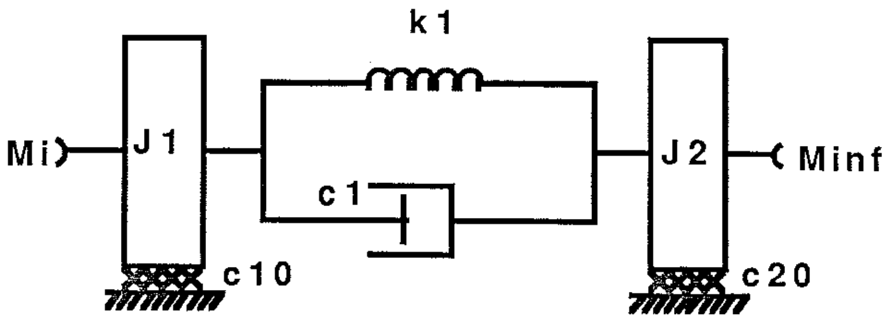


Figure 4.5. Linear robot model

In the figure following signs are used:

J1	:	Motor inertia
J2	:	Axis inertia
k1	:	Rigidity
c1	:	Viscous friction, inertia 1 to 2.
c10	:	Viscous friction, inertia 1 to case.
c20	:	Viscous friction, inertia 2 to case.
Mi	:	Input torque
Minf	:	Axis torque

The difference equations of the system

$$J1*\ddot{s}1 = Mi - Mv1 - c1*(\dot{s}1 - \dot{s}2) - c10*\dot{s}1$$

$$J2*\ddot{s}2 = -Minf + Mv1 + c1*(\dot{s}1 - \dot{s}2) - c20*\dot{s}2$$

with the shortenings

$$M1 = Mi - Mv1 - c1*(\dot{s}1 - \dot{s}2) - c10*\dot{s}1$$

$$M2 = -Minf + Mv1 + c1*(\dot{s}1 - \dot{s}2) - c20*\dot{s}2$$

M1 is the torque on inertia 1 and M2 is the torque on inertia 2.

$$\ddot{s}1 = -\dot{s}1*(c1 + c10)/J1 - \dot{s}2*c1/J1 - (s1 - s2)*k1/J1 + Mi/J1$$

$$\ddot{s}2 = \dot{s}1*c1/J2 - \dot{s}2*(c1 + c20)/J2 + (s1 - s2)*k1/J2 - Minf/J2$$

A state space transformation of these equations in the form

$$\dot{\mathbf{X}} = \mathbf{A*X + B*U}$$

With the new state variables

$x1 = \dot{s}1,\ x2 = \dot{s}2\ ,\ x3 = s1$ and $x4 = s2$

$$X = \begin{bmatrix} x1 \\ x2 \\ x3 \\ x4 \end{bmatrix}$$

gives the following equations:

$$\dot{X}(t) = \begin{bmatrix} -(c1+c10)/J1 & c1/J1 & -k1/J1 & k1/J1 \\ c1/J2 & -(c1+c20)/J2 & k1/J2 & -k1/J2 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} X(t) + \begin{bmatrix} Ml/J1 \\ 0 \\ 0 \\ 0 \end{bmatrix} U(t)$$

The parameters of the studied axis, on IRB - 2000:

Parameter	axis 2	axis 3	
J1+J2	0.0094	0.00625	$kg*m^2$
J1	0.0043	0.004	$kg*m^2$
J2	0.0051	0.00225	$kg*m^2$
k1	180	180	Nm/rad
b1	0.001	0.001	rad
c1	0.12	0.12	kg*m/s
c10	0.008	0.008	kg*m/s
c20	0.00001	0.00001	kg*m/s

Table 4.1. Parameters for axis 2 and 3.

With realistic values in the matrix above, the poles have been calculated for axis 2 and 3 in the robot by means of an expanded matlab, developed at ETH in Zürich for Macintosh computers.

The poles:
Axis 2
p1,2 = -26.2 ± 277*i
p3 = -0.85
p4 = 0.00

Axis 3
p12 = -43.2 ± 351*i
p3 = -2.56
p4 = 0.00

Non-linear resonance robot model

This robot model contains, static and dynamic friction, rigidity , viscous damping and a backlash. The dynamic friction works in the opposite direction as the resulting torque on each mass.

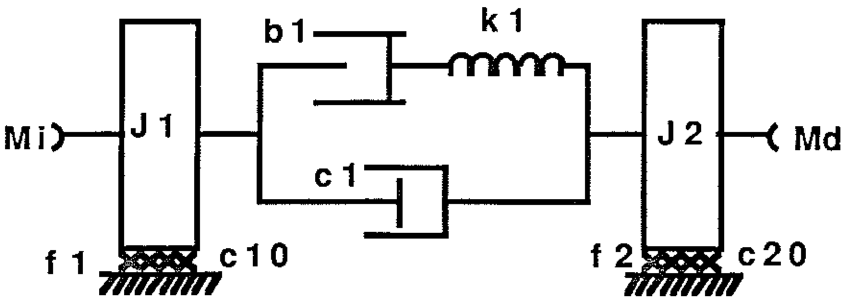


Figure 4.6. Non-linear robot model

J1	:	Motor inertia
J2	:	Axis inertia
k1	:	Rigidity
b1	:	Backlash
c1	:	Dynamic (viscous) gear friction, between inertia 1 to 2
c10	:	Dynamic (viscous) friction, inertia 1.
c20	:	Dynamic (viscous) friction, inertia 1.
f1	:	Sliding friction , inertia 1.
f2	:	Sliding friction, inertia 2.
Mi	:	Input torque
Minf	:	Axis torque

Balance equations for the non linear robot model:

$$M1 = Mi - Mv1 - c1*(\dot{s}1 - \dot{s}2) - c10*\dot{s}1$$

$$M2 = -Minf + Mv1 + c1*(\dot{s}1 - \dot{s}2) - c20*\dot{s}2$$

$$J1*\ddot{s}1 = \begin{cases} 0 & ; \ |M1| < f1 \\ M1 \quad -Mf1 & ; \ |M1| > f1 \end{cases}$$

$$J2*\ddot{s}2 = \begin{cases} 0 & ; \ |M2| < f2 \\ M2 \quad -Mf2 & ; \ |M2| > f2 \end{cases}$$

Mf1 = f1*SIGN(M1)
Mf2 = f2*SIGN(M2)

When the position difference between the motor and the arm is smaller than the half of the backlash, no torque is transferred between the motor and the arm. If the position difference is bigger than half of the backlash a torque proportional to the position difference minus half the backlash and multiplied with the rigidity k1.

$$Mv1 = \begin{cases} 0 & ; |s1-s2| < b1/2 \\ k1*SIGN(s1-s2)*(|s1-s2| - b1/2) & ; |s1-s2| > b1/2 \end{cases}$$

The analysis of this non linear axis model contains three groups of non linearities, see diagram.

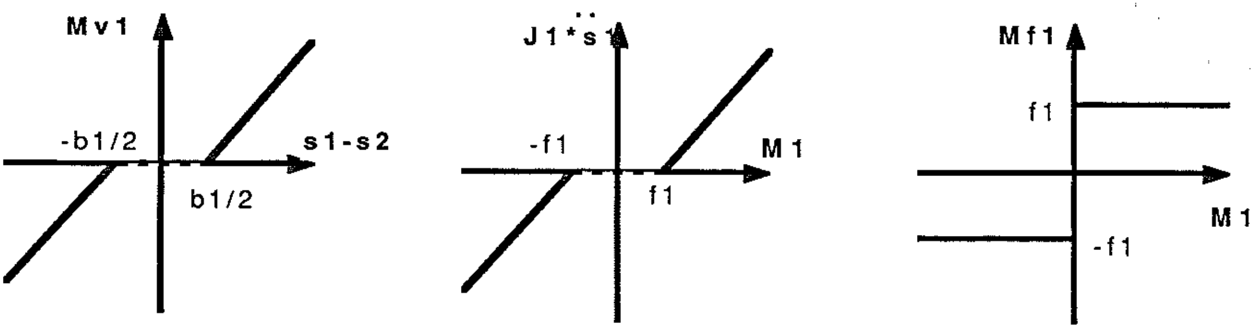


Figure 4.7 Non-linearities.

A state space transformation of the linear part of the equations in the form:

$$\dot{X} = A*X + B*U$$

with the state space variables

$x1 = \dot{s}1,\ x2 = \dot{s}2\ ,\ x3 = s1$ and $x4 = s2$

Results in the same state space transformation as for the linear resonance robot model.

Chapter 5.

Acceleration Derivative Control

Chapter 5.

Acceleration Derivative Control (ADC)

In this chapter the technique of making smooth signals in a control system is studied. The smoothness can be increased by filters placed in the control loop or in the reference generator. In the present ABB controller, the acceleration derivative is controlled in a module SMACC, (Smoothness acceleration control), located in the position control loop.

The different methods studied:
1. SMACC limited acceleration derivative in the position loop.
2. SMACC limited acceleration and retardation derivative in the position loop.
3. Acceleration derivative control in the position reference.
4. Non linear feedback filter on the position reference.

The basic structure of the controller system is show in figure 5.1 below.

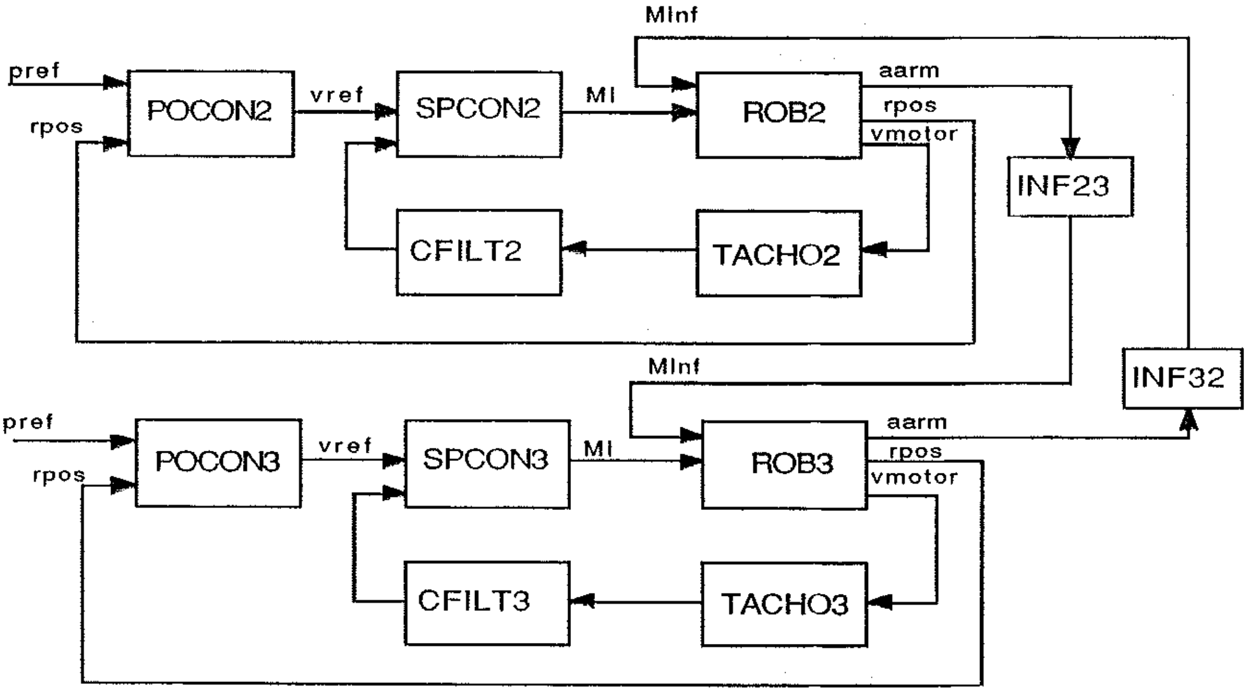


Figure 5.1. Robot system

POCON	:	Position controller
SPCON	:	Speed controller
ROB	:	Robot model
TACHO	:	Tachometer
CFILT	:	Filter

Present controller system

The SMACC module is located in the position control loop in order to limit the second derivative of the velocity reference. The purpose of the SMACC function is to avoid excitation of the mechanical resonances of the robot arm in the beginning of an acceleration. SMACC is thus only used during acceleration and not during retardation.
By means of the SMACC function, it is, however, also possible to decrease the path deviation because of unsufficient controller stiffness to high frequency disturbances. Since SMACC lowers the frequency range of the axis control, it will also lower the frequency range of the disturbances.

In order to decrease the path deviation also during retardation, the SMACC has to work also in this case. However, then the SMACC will decrease the stability of the position loop giving a unacceptable control with big overshoots.
To get rid of this problem, two other ADC solutions were studied. In the first, a path generator module with a limited acceleration/ retardation derivative was used on the position reference, in the other a non linear feedback filter, DACC was used.

1. SMACC limited acceleration derivative

The purpose of a SMACC is to limit the acceleration derivative of the axes.

Limited signals in the position loop limit the motor torque, and thus the arm acceleration. In the same manner a limited derivative of the motor torque will give a limited derivative of the arm acceleration. The SMACC, according to this reasoning, should be easier to place in the velocity loop. However, simulations with a SMACC in the velocity loop resulted in an unstable system.

In the present robot control system, S3, a SMACC is implemented. A SMACC prevents the robot from excitation of mechanical resonances. Steep flanks in a reference signal contain a larger frequency spectrum.

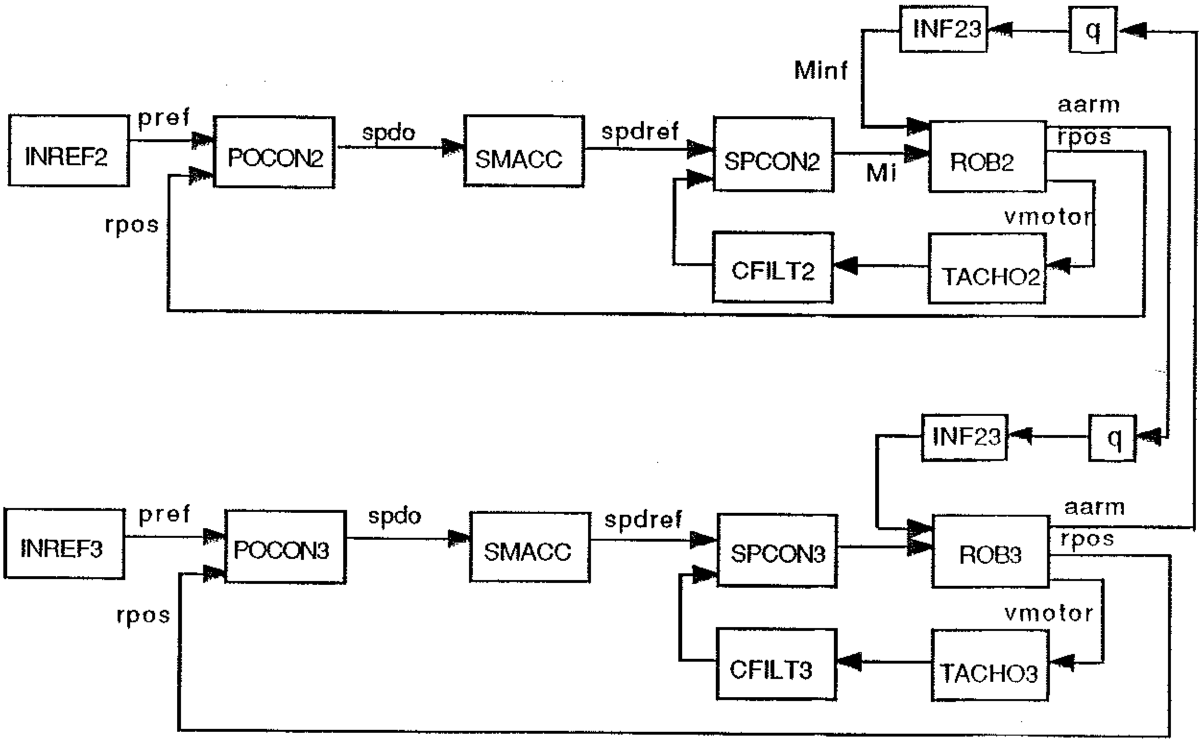


Figure 5.2. SMACC limitations.

Simulation results (SMACC)

For input reference generation the module INREF is used with an acceleration reference of 1000 rad/s^2, see PLOT 1. ADC Input Reference.The time of constant speed gives a controlled transition from acceleration to retardation. The plots are shown in Appendix B1.
PLOT 2. to 4. ADC, show simulations with a SMACC module working on the acceleration only.
In PLOT 2. ADC, the acceleration derivative is 9500 rad/s2. The trajectory error (the position deviation of axis 3) caused by coupled inertia is 0.05 rad and the velocity error is 0.6 rad/s .
With a SMACC limitation of 6000 rad/s2 the velocity error on arm 3 decreases to 0.4 rad/s but the position error is still 0.05 rad (in the acceleration limited part of the movement), see PLOT 3. ADC.
The disturbances are measured under the limitation and not as the top to top value.
The SMACC causes in this case the controller to build up an extra position error a LAG, during the limitation. The harder limitation of the acceleration derivative, the larger LAG is build up. The top to top disturbances are increasing with the limitations in these simulations. This is caused by increased LAG at harder limitations.

In PLOT 4. ADC the acceleration derivative is 4000 rad/s2. The velocity of arm 3 is decreased during the acceleration phase but the position error hardly decreases.
In this case the disturbance peak in the transition between acceleration and retardation, caused by the hard limited acceleration derivative, is increasingly larger, compare with PLOT 3. ADC. This causes in both cases larger path deviations during the retardation.

Robot verification

Since SMACC is implemented in the robot control system, S3, the robot can be tested with different SMACC limitations to see how it works in reality. The results from these verifications are described in Appendix B2. Experimental results.

In Robot verification 3, the motor torque on axis 2 and the disturbance velocity v3 on axis 3 are shown.The different Limitations in robot verification 3 are, without a SMACC, with a limitation on 9 500 rad/s^2 and 4 250 rad/s^2. Comparing the robot verification without a SMACC with a limitation of 9500 rad/s, the velocity error of axis 3 is bigger during acceleration but smaller during the transmission from acceleration to retardation. The motor torque has a steeper edge on the falling edge in crossings between acceleration and retardation. With harder limitations in the SMACC the influence velocity decreases. This is caused by the decreased performance of the robot. The robot control system, S3, softens the control signals and cycle time is increased.

In the simulations the peak caused during the steep transition between acceleration and retardation, increases with harder limitations. In the simulations the pulse duration and the magnitude in the reference signals in the simulations is kept constant with the different limitations in the simulations to have comparable simulations, in opposite to the robot verification where the pulse duration is increased and the reference magnitudes are decreases.
With softer signals by hard limitations in the simulations as in the robot the simulations also would result in decreased disturbances.

2. SMACC limited acceleration and retardation derivative.

With a SMACC working both on the acceleration and retardation, the disturbances caused by mass coupled inertia should be decreased.

The characteristics of a SMACC, working both on the acceleration and retardation is shown in the upper plot in PLOT 5 ADC. SMACC. The speed reference, spdref and the output from the SMACC, spdo is clearly limited.
During the acceleration/ retardation phase, the speed is limited with a maximum derivative, d2mxa and d2mxr.

Plot 6. ADC shows a simulation with an acceleration derivative limitation of 4 000 rad/s^2. In the plot the limitation is obvious in the motor torque. However, the same problems that occurred with the SMACC on the acceleration only, also occurs in this case. The controller builds up a position error LAG, on harder limitations. At crossings between acceleration and retardation, a peak , caused by the LAG, increases the influence disturbances on the disturbed axis.
In the lower plots in plot 5. ADC the phenomena of overshoot is shown. With a reference not followable by the controller, an overshoot, caused by the integral part in the speed controller occurs. With softer references this is provided. When the reference contains a period of constant velocity, the speed controller reaches up with the reference. The controller can get rid of the LAG, an overshoot is provided.

3. Position reference derivative acceleration control.

An external module does not change the phase margin of a system. This means that it does not cause stability problems in the control loop.
The reference generator, POSDACC, is constructed to calculate the trajectories with a limited second derivative of the position reference. The output characteristic of the position generator, POSDACC is shown in plot 9. ADC. The implementation in simnon is based on references in the acceleration derivative. The position is calculated with three integrations.

Simulation results (POSDACC).

The parameters in POSDACC, the pulse times dat, at and delayt are chosen to result in a maximum acceleration of 1 000 rad/s2 and a velocity of maximum 200 rad/s for different acceleration derivatives. The pulse duration is thereby increased for harder limitation in d2mxa=d2mxr. The mathematical analysis of POSDACC is found in Appendix A3.
Plot 7 and 8 ADC show simulations with limited acceleration derivatives of 15 000 and 8 000 rad/s3. Both without a band of friction and a backlash in the robot.

$\partial a/\partial t\,|_{max}$	Friction	dat	at	delayt	v3	p3	Controller	PLOT
(rad/s^2)		(s)	(s)	(s)	(rad/s)	(rad)	error	
15 000	Yes	0.0667	0.133	0.2	0.90	0.093	1.40	
10 000	Yes	0.1	0.1	0.2	0.81	0.090	1.35	
8 000	Yes	0.125	0.075	0.2	0.80	0.090	1.30	
6 000	Yes	0.1667	0.0333	0.2	0.60	0.08	1.1	
15 000	No	0.0667	0.133	0.2	1.02	0.123	1.21	7. ADC
10 000	No	0.1	0.1	0.2	0.87	0.074	0.84	
8 000	No	0.125	0.075	0.2	0.78	0.071	0.83	8. ADC
6 000	No	0.1667	0.0333	0.2	0.61	0.067	0.75	

Table 5.1. Simulation results with a limited position reference
Glossary:

Friction : Friction and backlash in the robot model.
dat : Time during, derivative acceleration, see Appendix A3.
at : Time during constant acceleration, see Appendix A3.
delayt : Time during constant velocity, see Appendix A3.
v3 : Influence velocity on arm 3.
p3 : Trajectory error.
Controller
error : The controller error in the speed controller.

With a position generator, limiting the acceleration derivatives the influence can be damped. In the simulations above the trajectory error can be decreased with a factor two, using a derivative limitation of 6 000 instead of 15 000 rad/s^2.

The advantages with an acceleration derivative control in the path generation are:
* The stability of the servo is not affected.
* In the path generation it is possible to limit both the acceleration - and the retardation derivative.

With a limited acceleration derivative the disturbances, velocity error and the trajectory error could be decreased. However the reduction is not big enough to achieve a position error < 1 in a typical case.

4. Closed loop filter for derivative acceleration control, DACC

The control system contains information of the arm positions and the desired positions. With this information the robot control computer calculates the trajectory for the arm. If the trajectory calculations have to be based upon a limited third derivative of the position reference, the computational effort would increase and the trajectory generation procedures would be complex.
An external module, separated from the trajectory calculations for the position reference would therefore be desired. A closed loop derivative acceleration control filter, DACC was thus constructed.
The DACC-filter developed contains a model rigid body robot. The model of the robot, is supposed to make the position reference signal more followable for the robot control system. A closed DACC- filter, limits the derivative of acceleration and retardation in the position reference. The closed loop DACC-filter is coupled without feedforward to the position reference on the position controller.

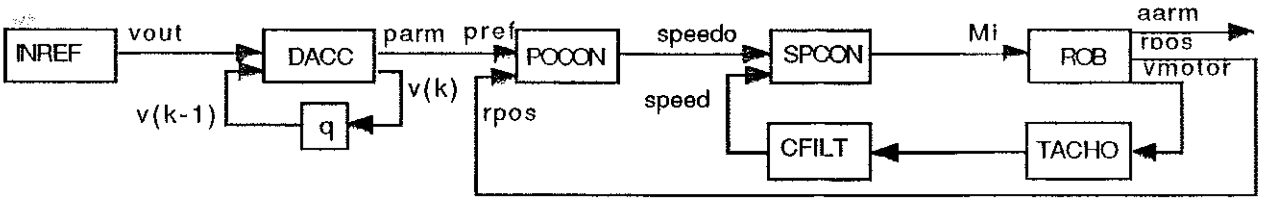


Figure 5.3. Filtered position reference.

The position reference, leaving the closed loop model, is filtered, so that the derivative in the acceleration will be limited, DACC.

Non linear feedback filter.

The derivative of the acceleration is limited with a parameter, d2mxa, in the module DACC2<x>. With a square root filter the derivative of the retardation is limited.
With a limitation of the derivatives in the acceleration and retardation the position reference will be possible to follow for the control system, simultaneously with reduced disturbances on robot axes.

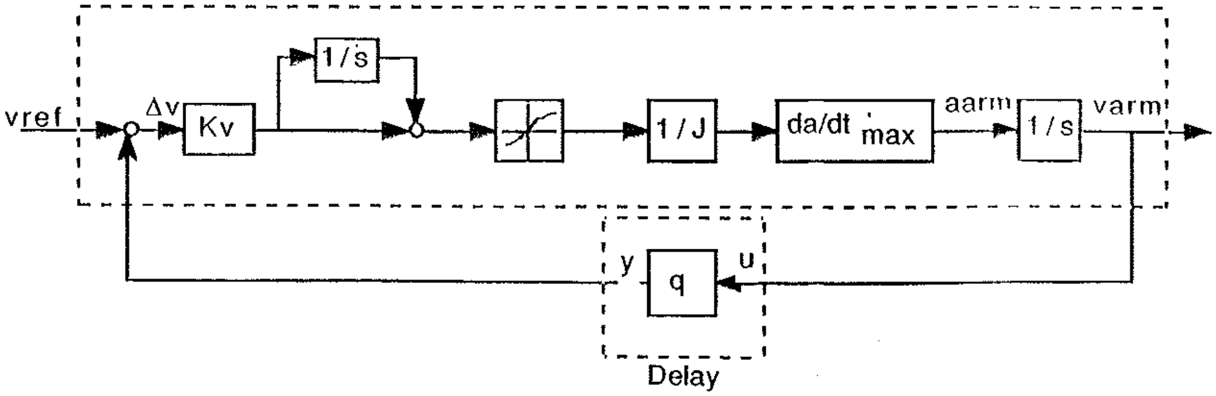


Figure 5.4. Closed loop-filter of the robot.

Square-root filter

The derivative of the retardation is limited with a square-root filter, which is represented by the non linear box in the figure above.

The formulas describing the square root function:

$$e_v = v_{ref} - v_{real}$$

$$a = 1/J * \sqrt{(K_v * e_v)}$$

$$\frac{da}{dt} = \frac{1}{2} * \frac{1}{\sqrt{(K_v * e_v)}} * \frac{K_v}{J} * \frac{de_v}{dt}$$

$$\frac{de_v}{dt} \approx \frac{dv_{real}}{dt} \approx a$$

==>

$$\frac{da}{dt} = \frac{1}{2} * \frac{K_v}{J^2}$$

With the parameters Kv and J, it is possible to control the the maximal derivative in the retardation.

A lag is build up in the filter, to prevent an overshoot, the filter has to be able to reach up with the reference signal before the retardation phase starts, like in PLOT 5 Overshoot, where the reference signal is delayed until the speed controller has reached up with the reference signal. PLOT 10. ADC, shows the total behaviour of dacc2 as a closed loop filter.
The reference signal is almost identical with those from POSDACC, compare with PLOT 9.ADC.

The advantages with this closed loop feedback filter are:

* The stability of the system is not changed with an external module. With smoother references it might even be improved, they don't contain a wide frequency spectrum as the stiffer flanks, and thereby don't excite the system into mechanical resonances.
* Complicated algorithms for trajectory calculations are avoided.
* The disturbances caused by mass coupled inertia is decreased. (Compared with an internal SMACC).

Conclusions, ADC :

* With a SMACC limited acceleration derivative in the position loop the disturbances can be decreased during the limitation. However, hard limitations give rise to other phenomenon, as a lag in the position controller that causes a disturbance peak in crossings between acceleration and retardation.
* A SMACC limitation on both acceleration and retardation easily causes stability problems. To prevent stability problems, the performance has to be decreased to such an extent that this approach is not interesting.
* Acceleration derivative control in the position generation makes it possible to limit both the acceleration - and retardation derivative without causing stability problems.
* A non linear feedback filter for acceleration derivative control is a simple implementation of the acceleration derivative in the position reference.

Chapter 6.

Torque Feedforward

Chapter 6

Torque Feedforward, TFF

Torque feedforward between axis two and axis three is used to compensate for disturbances caused by coupled mass inertia and minimize the path deviation. The compensating torque should have the same curve form as the disturbance torque, but with the opposite sign.

Four kinds of torque feedforward, TFF principles, were studied:

* Non-linear resonance torque feedforward, TFF
* Linear resonance torque feedforward, TFF
* Non-linear rigid body TFF
* Linear rigid body TFF

In the simulations robot axis 2 disturbs axis 3. Robot verification 1 is an experimental test where the robot axis two disturbs axis three. Axis two moves the arm between two end points at a high speed to influence the other axes as much as possible, see Robot verification 1 in Appendix B2. Robot verification 1 was also used as a reference for the simulations, see Appendix A3. MOMGEN, MOMREF.

The coupled inertia between axis two and three are the only influence studied in this work even if the other axes are also influenced by these movements.

According to the simnon simulations the derivative of the acceleration/ retardation is the critical part for the transient path deviations caused by the coupled inertia.

Non-linear resonance Torque Feedforward

The robot model ROBMOD, used for the feed forward generation, contains rigidity, backlash, dynamic viscous damping and static and dynamic friction.

Sign changes on the motor torque makes the robot passing through the band of friction. At small values of the motor torque the arm can get caught in the band of friction. Leaving and passing the band of friction could create disturbances with a high bandwidth.

Figure 6.1 shows this kind of coupled inertia compensation.

Linear resonance Torque Feedforward

The simplest linear resonance torque feedforward module contains rigidity and viscous damping but not a band of friction and a backlash. In the simulations the module, ROBMOD is used for these simulations with the friction and the backlash set to zero.

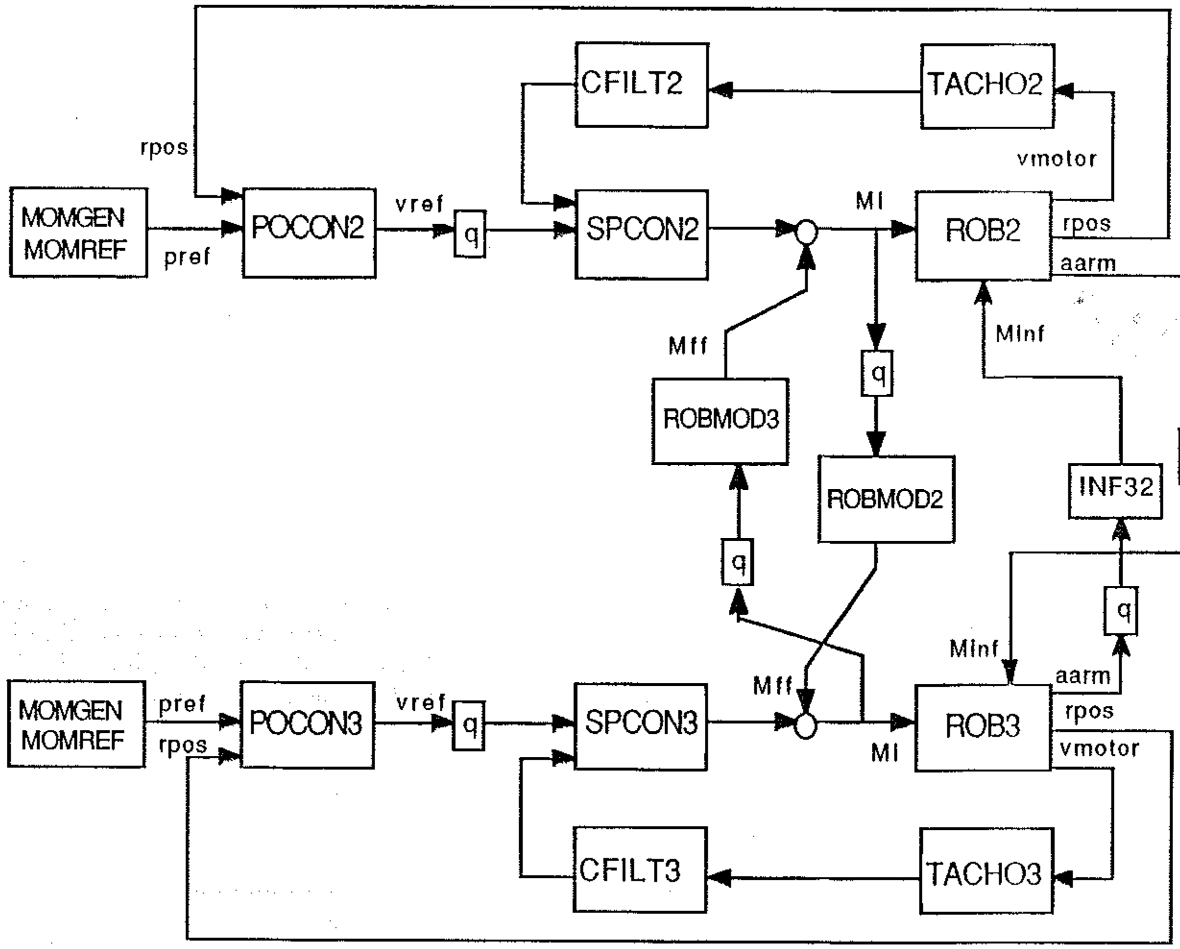


Figure 6.1. Resonance TFF.

POCON	:	Position controller
SPCON	:	Speed controller
ROB	:	Model of a robot axis
INF	:	Disturbances caused by coupled inertia
TACHO	:	Tachometer generator
CFILT	:	FILTER
ROBMOD	:	Non-linear resonance torque feedforward.
MOMGEN		
MOMREF	:	Trajectorie generator
q	:	Delay

Non-linear rigid body Torque Feedforward.

To improve the linear rigid body torque feedforward, a band of friction is added to the model and the non-linear rigid body TFF is achieved.

To move a mass m lying on the ground, a force greater than the static friction has to be applied before the mass starts moving, see figure 6.2 below.

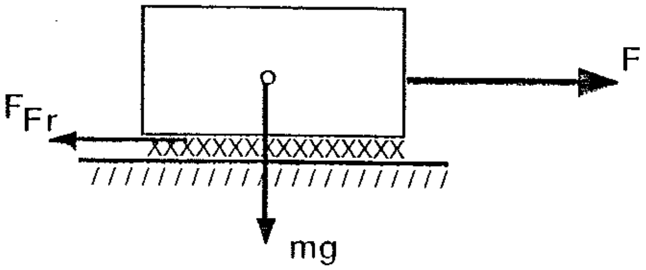


Figure 6.2. Body of mass m with static and dynamic friction.

When $|F| < F_{fr}$, the mass, m, lies still. The same principal is also valid for robot arms. Measurements on the robot result in a band of friction in between $|M_{Fr}| < 0.55$ Nm. The rigid body model was extended with this band of friction, to be more like a real robot.

Linear rigid body Torque Feedforward.

This feed forward is based on a rigid body model of the robot axis. In order to minimize the computational effort, simpler algorithms for the Feedforward are desired than in the non-linear resonance TFF. In non-linear resonance TFF, with four state variables, the computer has to solve four difference equations (these are v1, s1, v2 and s2.) for each axis. With a simple model based on a rigid body , see figure 6.3 below, the calculations are minimised to one multiplication by a constant.

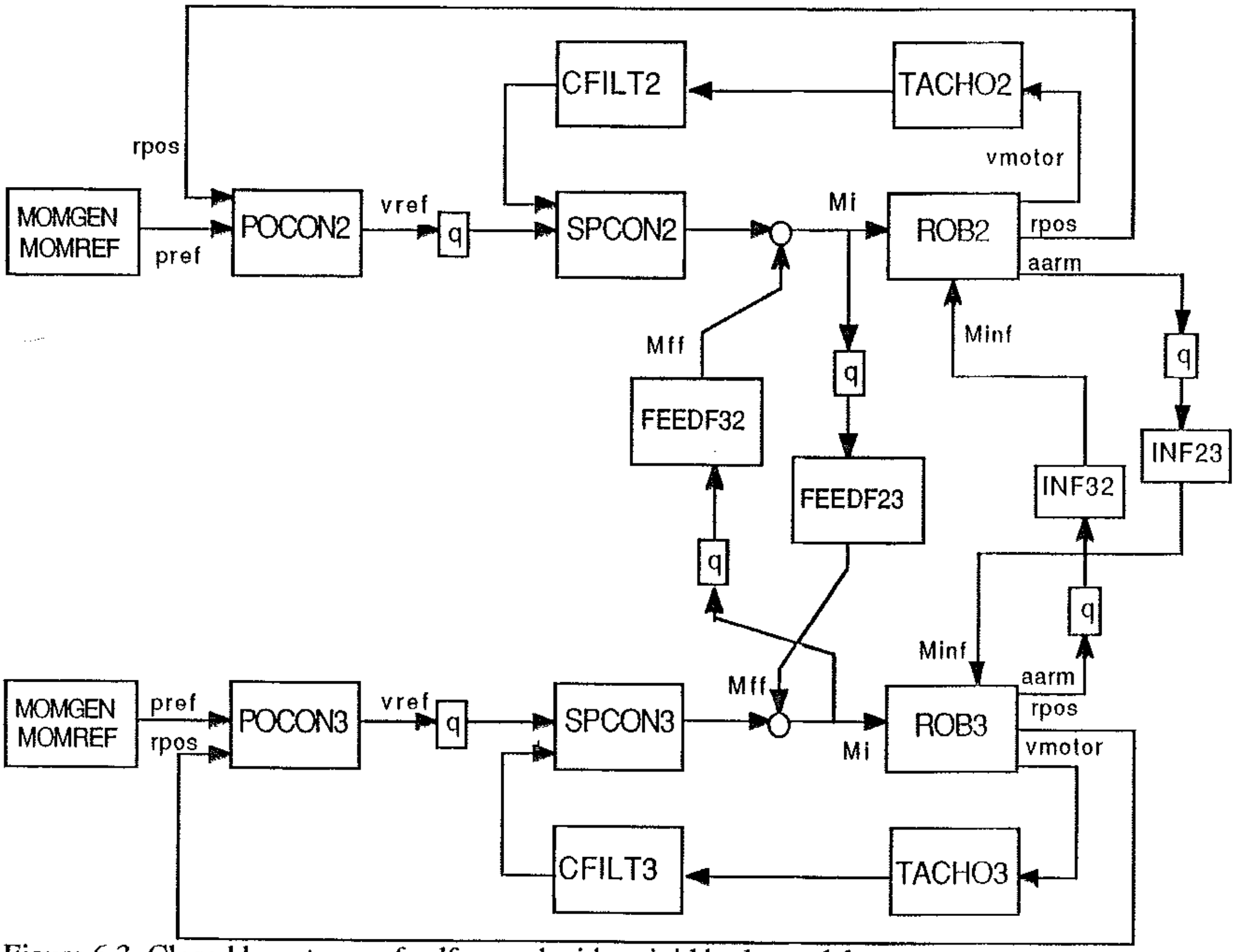


Figure 6.3. Closed loop torque feedforward with a rigid body model.

A rigid body model is based on the mathematics below.(See also the description of the linear rigid body model in chapter 4).

The acceleration of arm two is proportional to the input torque.

$$\frac{d^2x_2}{dt^2} = M_i/J_2$$

The coupled inertia torque between the axes two and three is

$$M_{inf23} = J_{23}* \frac{d^2x_2}{dt^2} = M_2*J_{23}/J_2$$

The speed controller , spcon, calculates the torque M2, which is the input to the motor on axis two. Inertia J_2 is the common inertia for the motor and for the arm on axis two, J_{23} is the coupled inertia between axis two and three. These are calculated in a dynamic robot model in the present robot control system, S3, from ABB Robotics. Compensation with TFF can be don with the actual values in the working position.
A linear rigid body Feedforward module is able to compensate for linear coupled inertia effects, but unable to compensate for non linearities, since the non-linear resonance torque feedforward is sufficient.

Cross coupled and closed loop TFF.

Torque feedforward can be done in two different ways, one closed loop torque feedforward and one cross coupled torque feedforward. See figure 6.4 and 6.5 below.

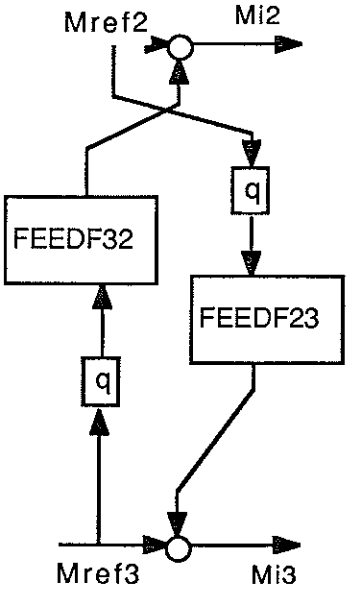


Figure 6.4. Cross coupled TFF

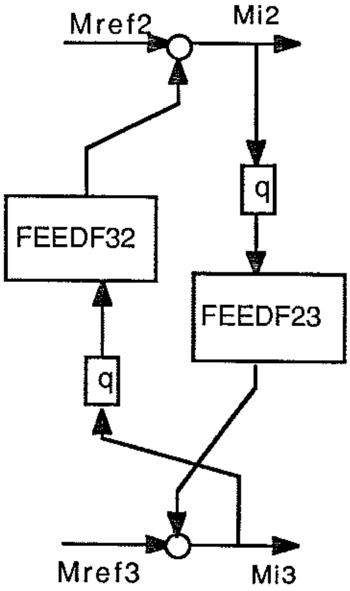


Figure 6.5. Closed loop TFF

The advantage with the closed loop feed forward is that the input signal to the TFF is exactly the same as that to the influencing motor axis. However, a closed loop system might cause stability problems when the poles of the closed loop system are not well damped. In this case the poles are located close to the origin, see Figure 6.6 below, and thereby very well damped.

Calculations:

$$M_{i2} = M_{ref2} + K_{32}*q*M_{i3}$$
$$M_{i3} = M_{ref3} + K_{23}*q*M_{i2}$$

$$M_{i2} = M_{ref2} + K_{32}*q*(M_{ref3} + K_{23}*q*M_{i2})$$
$$M_{i3} = M_{ref3} + K_{23}*q*(M_{ref2} + K_{32}*q*M_{i3})$$

$$M_{i2} = \frac{M_{ref2} + K_{32}*q*\ M_{ref3}}{1\ -\ q*K_{23}}$$

$$M_{i3} = \frac{M_{ref3} + K_{23}*q*\ M_{ref2}}{1\ -\ q*K32}$$

Poles:

$q_2 = 1/K_{23}$
$q_3 = 1/K_{32}$

$K_{23} = K_{TFF}*J_{23}/J_2$
$K_{32} = K_{TFF}*J_{32}/J_2$

Parameters:

$K_{TFF} = 1$
$J_2 = 0.0094$
$J_3 = 0.00625$
$J_{23} = 0.0015$
$J_{32} = 0.0015$

===>

$q2 = 0.16$
$q_3 = 0.24$

The poles are placed near the origin at a distance of about 0.16 and 0.24.

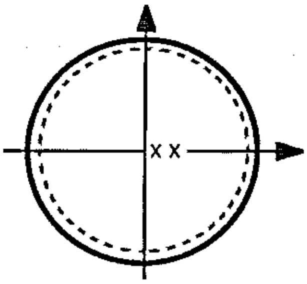


Figure 6.6. Poles of the closed loop torque feedforward.

The results of these two kinds of torque feedforward do not differ much in the simulations. This might be explained by the fact that the poles of the closed loop torque feedforward are well damped and that the TFF signal is relatively small compared with the feedback signal of the disturbing axis. With large TFF signals the controller is stiff enough to counteract the differences.

The conclusion we can draw from these simulations with closed loop and cross coupled torque feedforward, is that in an implementation we can use the one that is the easiest to implement, since the results do not differ significantly in the different cases.

Simulations with Torque Feedforward.

The simulations with and without torque feedforward were made for these cases:
* Non-linear resonance torque feedforward
* Linear resonance torque feedforward
* Non-linear rigid body feedforward.
* Linear rigid body feedforward.

The input signal in all the torque feedforward simulations is a signal that should have the same curve shape and pulse duration as the reference signal in Robot verification 1, see Appendix B2, Experimental results.
Momref and Momgen is the reference generator used for these simulations. This to create signals like the real ones in a robot, see Appendix A3 and Appendix B2. The simulations in this chapter are carried out with and without a band of friction and a backlash in the robot model. Plot 3 TFF shows a simulation with a robot containing friction and backlash. Plot 1 TFF is without.

The measurements in the robot gives the opportunity to measure the motor torque and the motor speed from test signals in the control system. The trajectory error also can be studied with a pen on the robot arm. A pen fixed to the robot arm draws the complete trajectory when the robot moves in one plane, a paper on a board can then be easily placed in the same plane as the movements. The difference in the drawn trajectories gives the position error caused by coupled inertia.

The system was simulated with different feedforward factors, the one that minimizes the influence of the most is the factor one, full feedforward. Increasing or decreasing the feedforward factor only increases the trajectory error.

The friction (static and dynamic) is supposed to have the same value as in the ABB-robot IRB-2000. In the simulation with friction and backlash the measured signals are 'discontinuous'. Discontinuous in this case means that the measured signals show up a curve form with almost infinite derivatives. The simulations without friction, have a soft curve form without any kind of corners. In this matter the simulations done without friction and backlash have a curve form more according to the real one measured on the robot, see Appendix B.

Resonance Torque Feedforward

The damping of the disturbances with this non-linear resonance torque feedforward is 3.8 in the arm velocity and 5,7 in the arm position. (The damping is calculated as the influence in a simulation without feedforward divided by the achieved one with feedforward.)
A table containing the most important results from these simulations, compares the results with different kinds of feedforward with and without a band of friction and with and without backlash, see tables below.
In the Table 6.1 the robot model does not contain a band of friction or a backlash.

PLOT 1. torque feedforward, shows a simulation without TFF and Plot 2. TFF, with non-linear resonance torque feedforward. In the simulations showed in PLOT 1 and 2, the robot does not contain a band of friction or a backlash. These simulations without friction and backlash are done because, they have a curve form more like the real ones. Compare these simulations with the simulations in PLOT 3 and 4, where the robot contains friction and backlash.
However, the damping of these simulations are well comparative to the ones without friction, see the tables below.

The Plots contains:
Motor torque on axis 2 - in the upper left corner.
Torque on inertia 1 and 2 - in the lower left corner.
The position error - in the upper right corner.
The velocity error - in the lower right corner.

Inertia 1 and 2, mentioned above, is the inertia the motor inertia and for the arm. The torque on the arm inertia is basically the influence torque.

Compare PLOT 3 and 4. This shows that, with a feedforward, creating a compensation torque with the opposite sign of the disturbance on, the position error and the velocity error are substantially reduced.

As mentioned above the velocity on axis 3 appears less continuous with non linearities than in the simulations without.
The simulations without non linearities might be more realistic than the ones with.
In the book, Introduction to Robotics, Mechanics & control, the problematic of creating realistic robot models is discussed. The author John. J Craig, proposes that, computer simulations should be done without non linearities.
The most important factor for this simulation is, though ,the fact that both of them gives a similar result in the damping of the influence.

Rigid body Torque Feedforward.

In these simulations the modules Feedf23 and Feedf32 are used.
With a linear rigid body torque feedforward(Lin r.b TFF) without friction and backlash the influence is damped a factor 5.3 in the arm velocity and by a factor 6.8 in the position error, when the robot does not contain any friction or backlash, see table below. Influence damping without non linearities, below and PLOT 5. TFF.

When the robot contains friction and backlash a non-linear rigid body model (Non lin. r.b TFF) damps the influence speed with a factor 4 and a linear with 3.6. The trajectory error is damped with a factor 3.6 - 3.8, see table. Influence damping with friction.

TFF	Max arm velocity	Max position fault	Damping	
	v3 (rad/s)	parm (rad)	v3,ref/v3	parm,ref/parm
None	1.2	0.058	- - -	- - -
Lin. resonance	0.21	0.0087	5.7	6.7
Lin. r.b TFF	0.23	0.0085	5.3	6.8

Table 6.1. Influence damping without non linearities.

	Max arm velocity	Max position fault	Damping	
TFF	v3 (rad/s)	parm (rad)	v3,ref/v3	parm,ref/parm
None	1.12	0.068	- - -	- - -
Resonance TFF	0.27	0.012	4.1	5.8
Lin. r.b TFF	0.28	0.012	4.0	5.8
Non lin. r.b TFF	0.31	0.012	3.6	5.8

Table 6.2. Influence damping with non linearities.

Simulations with cross coupled and closed loop Torque Feedforward.

Simulations have been done to study the effect of cross coupled and closed loop torque feedforward. In the simulations, the robot contained, friction, backlash, rigidity and viscous damping. The torque feedforward module was a rigid body model, simulated with and without a band of friction. The results from these simulations appears in the tables below.

TFF	Max arm velocity v3 (rad/s)	Max position fault parm (rad)	Damping v3,ref/v3	 parm,ref/parm
None	1.12	0.068	- - -	- - -
Cross coupled	0.22	0.0085	5.4	6.8
Closed loop	0.23	0.0085	5.3	6.8

Table 6.3. Influence damping without non linearities in torque feedforward with a rigid body model.

Way of Feedforward	Max arm velocity v3 (rad/s)	Max position fault parm (rad)	Damping v3,ref/v3	 parm,ref/parm
None	1.12	0.068	- - -	- - -
Cross coupled	0.31	0.012	3.6	5.7
Closed loop	0.28	0.012	4.0	5.8

Table 6.4. Influence damping with friction in torque feedforward with a rigid body model.

In plot 6. torque feedforward two simulations with cross coupled and closed loop torque feedforward is shown. Both of them do not contain a band of friction or a backlash.
As shown in the table and in the plots no larger differences between the to different principal ways of TFF exists.

Conclusions:

Torque feedforward minimizes the disturbances caused by coupled mass inertia. The simulations also show that with acceleration derivative control the influence can be further decreased to some extent.
With TFF the path deviation and the velocity error can be drastically decreased. With the different dynamical models in the TFF the results are comparable, there exists no larger differences with the different TFF modules simulated. The average damping (without non linearities) of the velocity error, v_3 is 5.5 and of the position error, p_3 is 6.8.
The average damping of disturbances with non linearities is in velocity 3.9 and in position 5.8.

Sampled Torque Feedforward

The sample time can affect the stability and performance of a system. To investigate the sensitivity of different sample times, simulations have to be done to see whether the sample time affects the sensitivity of our system. In order to test the sensibility, simulations are done with and without a band of friction.
A realisation of a Torque Feedforward module in the ABB robots, has to be, at least in the first step in the servo computer. This because that the axis computer is very busy with other routines, and has no memory left for new routines. The servo computer is sampled with 12 ms, while the axis computer is sampled with 2 ms.
To investigate the sensibility for different sample times, a simulation is made with and without a band of friction. In plot 7 and 8. TFF. Sampled torque Feedforward, a sample of the simulation results are shown. The system with friction in the robot and an sampled non linear rigid body Torque Feedforward. The result of the simulations is shown in the diagrams below.

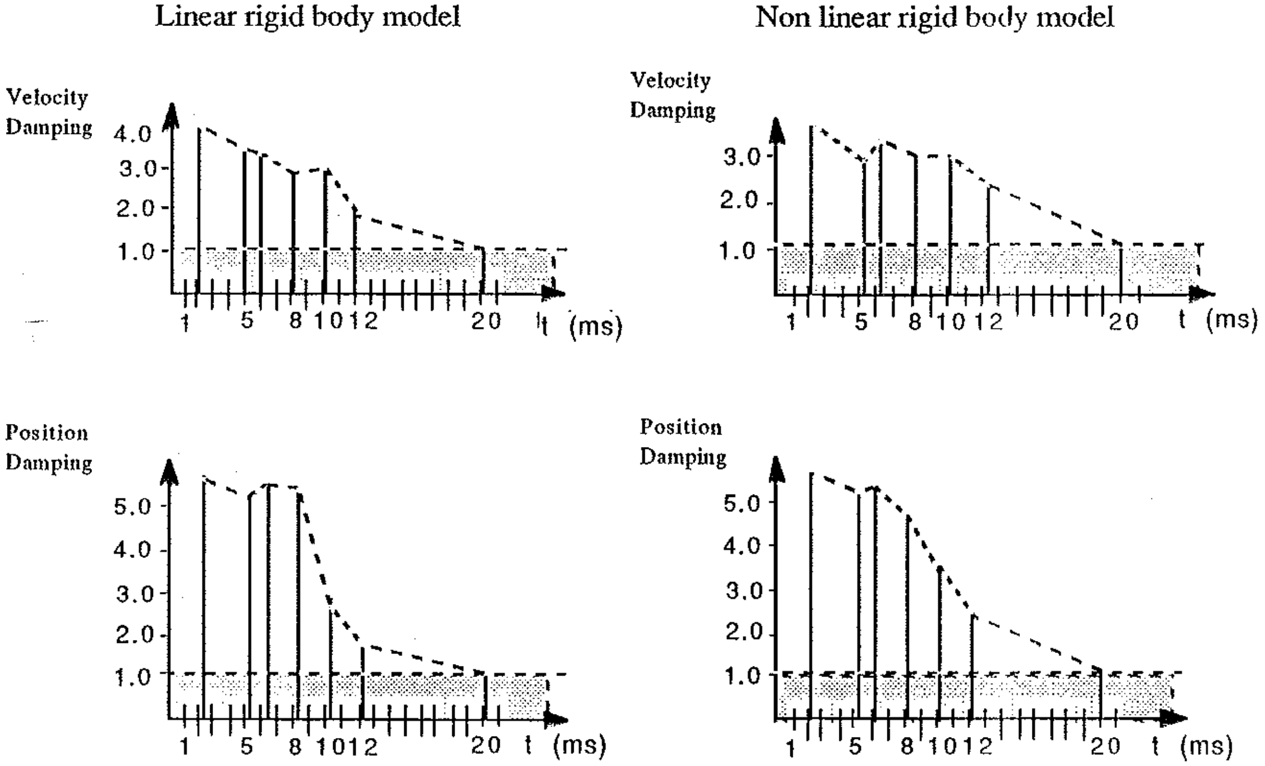


Diagram 6.1. Influence damping with sampled Torque Feedforward

Velocity damping is the damping achieved in the velocity of the disturbed axis and position damping is the damping of the trajectory error. The damping of the influence is greatest with a sample time of 2 ms, and damps the position fault with a factor of about 6 and the influenced velocity by a factor of about 4. With a sample time of 12 ms, the damping of the position error and the influenced velocity is about two. With a sample time of 20 ms, Torque Feedforward cannot improve the influence fault any longer.

Conclusions

According to the simulations the damping of the disturbances can be increased by a factor three, just by choosing a faster sample time than 12 ms.
In a final implementation TFF has to be implemented in a computer with a sample time of 6 ms or less.

Robustness

The regulator and the robot itself contains two major non linearities, the band of friction and a backlash. Calculating the theoretical stability for a system with several non linearities would need a lot of computational effort. When avoiding a larger mathematical study, simulations to determine the stability can be done. In the simulations to determine the robustness of the system divide the robustness in two parts, stability and parameter sensitivity and consider more about practical stability and practical sensitivity. The simulations have to cover different cases in the working range. These simulations can in no way cover an theoretical study with phase portraits but it suits the purpose of a test for our robot application, when we can prove if it is stable in the working range or not.

Stability

One possible way of testing the stability of a non linear system, is to give different step amplitudes on the input of the system, and then study the state variables. All state variables have to be stable for all the different step amplitudes. With a step on the input, the system respond with a step respond. The system seam to be asymptotical stable for the step amplitudes investigated, even though some of them were unrealistic high.

Parameter sensitivity

The inertia varies in the working range and with different lasts. The inertia of axis two varies ±4% and of axis three ±19% at the most. The inertia for axes two varies with the last and the inertia on axis three varies with the working position.

To study the sensitivity within the working range and with different lasts, simulations have been done to cover the different possibilities in the inertia variation. Simulations with the most extreme variations of the inertia on axes two and three are presented in plot 9 and 10 in Appendix B1. TFF. Inertia variations in axis two with ±4 % and in axis three with ±19 % are simulated with the four permutations of this extreme inertia variation. In the simulations presented, the model contain non linearities.

In simnon the motor inertia is separated from the axis inertia with a rigidity. The variation of ± 4 % and ± 19 % is calculated for both the axis and the motor inertia. This means that the inertia for the axis varies the more. The given inertia in PLOT 9 and 10 TFF is the common inertia for both the motor and the axis.
The simulation done with different arm loads only differ marginally in the peak to peak value in the velocity error and the position error on the disturbed axis. The variation of the position is 5 mrad, it varies between 13 mrad and 18 mrad, which can be considered as a small variation.

Chapter 7.

Velocity Feedforward

Chapter 7

Velocity Feedforward, VFF

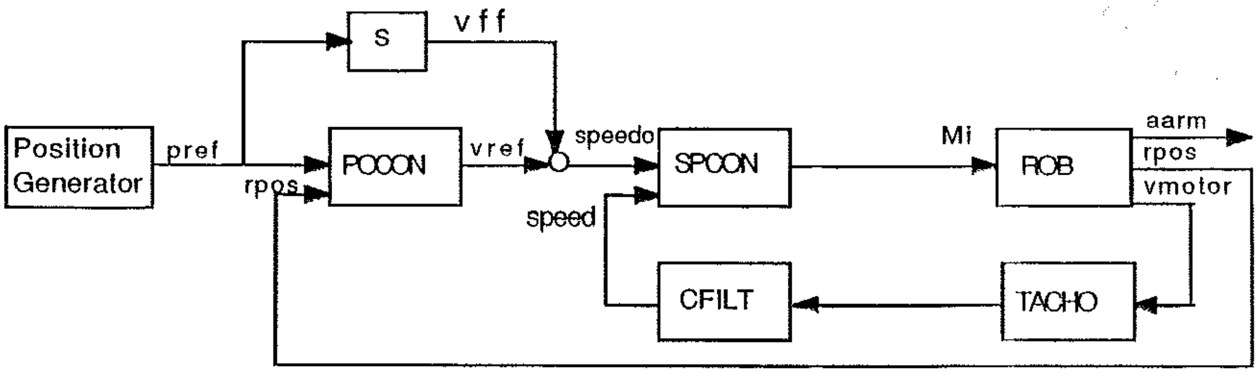


Figure 7.1. Velocity feedforward.

With TFF between axes two and three, as described in the previous chapter, the path deviation caused by high derivatives in the acceleration were damped in the simnon simulations by a factor of almost 7.
With the robot a trajectory error was measured, at the most 3 mm on the reference trajectory, see Robot verification 1. Appendix B2. With a damping of this trajectory error by a factor 7, the achieved trajectories differ less than 0.5 mm.

With a position controller and a speed controller in series, the position controller has to build up a position error before the speed controller reacts. This will give delays in the control and the servo will not be as fast as the robot performance allows.

In this chapter a technique based on followable references for the robot, combined with an axis individual velocity feedforward from the position reference to the speed reference is introduced.
An axis individual feedforward from the position reference to the speed controller, here called velocity feedforward, VFF, makes the speed controller react sooner to given references and decreases the position error in the position controller.

The technique of velocity feedforward, VFF, speed reference combined with the position reference is used to control the robot. The speed controller reacts immediately to the reference value, and the performance of the robot is increased. The average speed of the robot arms, is then also increased.
To prevent a drastic increase of the disturbances from steeper control signals, they are made more followable with the non linear feedback filter for acceleration derivative control, see Chapter 5.
The limited derivative of the acceleration and the retardation in the position reference also limits the torque derivative for smoother movements, reduced arm vibrations due to resonance excitation and reduced disturbances on other axes.

The speed reference is the integral of the arm acceleration calculated in the non linear feedback filter for acceleration derivative control that contains a rigid body robot model.

Direct Velocity Feedforward.

An axis individual feedforward will increase the performance of the system. In this step a feedforward from the position generation to the speed controller is added. The speed controller will then react quicker to new references.

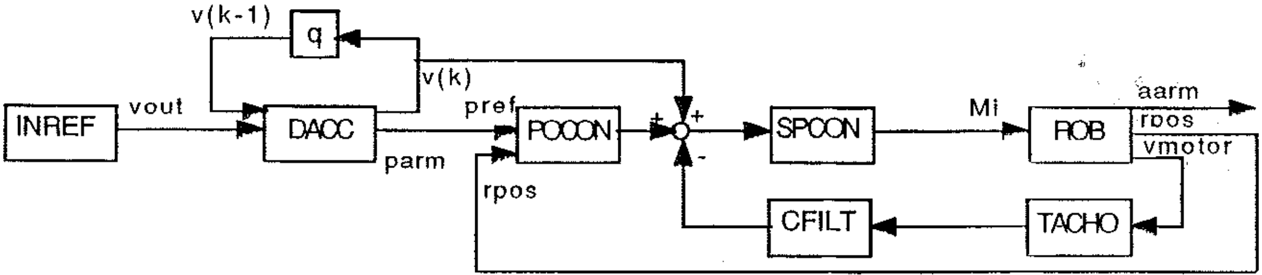


Figure 7.2. Direct velocity feedforward.

With a closed loop derivative control filter, DACC, the average velocity can be increased, see the Tables, 1 - 4 Velocity feedforward.

The first simulation in every Table is a simulation without velocity feedforward, VFF and without torque feedforward, TFF.
With VFF the average speed increases by 70 - 100 %. With a direct velocity feedforward the controller error will decrease by a factor 15 at the most. However the path deviation increases with VFF only but with both VFF and TFF the disturbances are well damped.

Velocity Feedforward with filtered position reference.

A model of the robot system is added to the reference generation to create lag adapted and more followable references for the robot. This should give a smaller position error because of more accurate position reference.

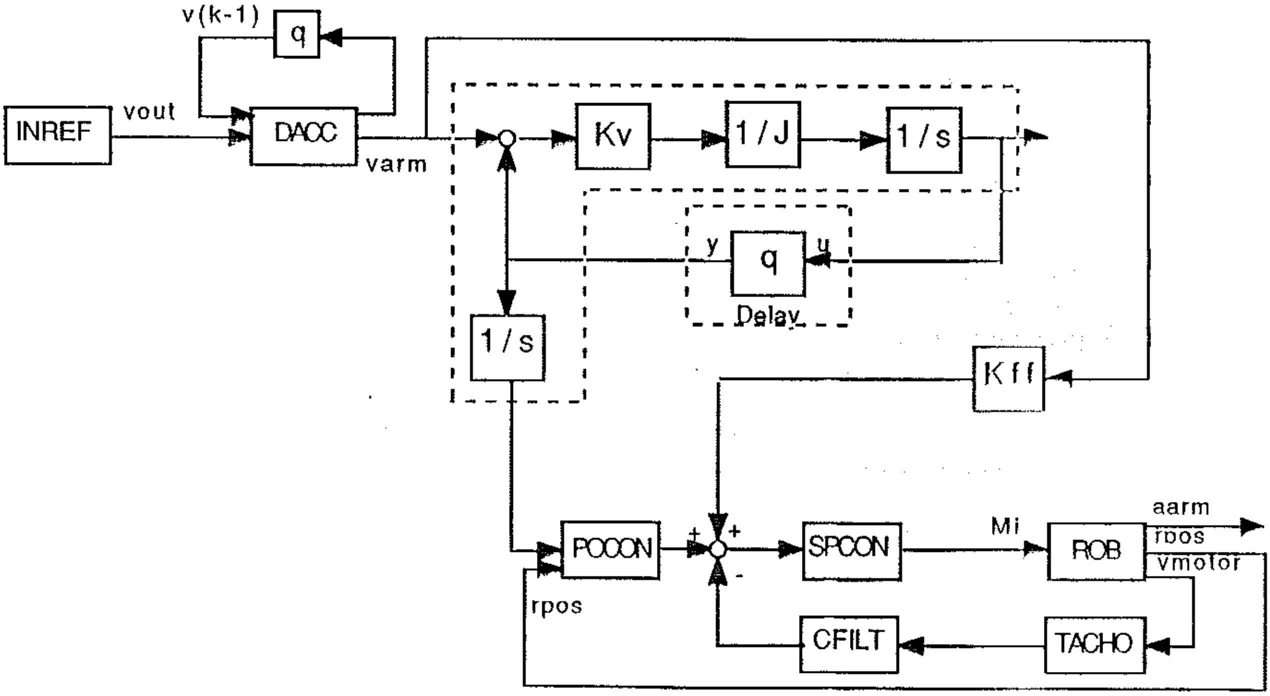


Figure 7.3. Velocity feedforward with followable references.

The average speed is calculated as the position movement, Δp, divided by the time for the controller to stabilize the LAG to a maximum error of 0.1 rad. The average speed increases with 30 - 50 % with velocity feedforward. With a filtered velocity feedforward the controller error decrease by a factor 20 at the most.

The velocity of axis 3 is marginally lower and in the trajectory error of axis 3 no differences to the simulations with direct velocity feedforward is seen.

The Plots 1 - 8 shows simulations without a band of friction and a backlash in the robot. Plot 1 - 4 is with a tough reference with direct crossings from acceleration to retardation, without a delay to achieve constant velocity. Plot 5 - 8 is with a delay in the references to achieve period of constant velocity. This to prevent rapid crossings from acceleration to retardation.

Conclusions:

- Velocity feedforward decreases the lag drastically
- Velocity feedforward increases the average speed of a robot.
- With a more followable position reference,the performance can be even better.
- With VFF the disturbances increase but with TFF the disturbances are well damped and the performance is increased.

Simulation results, VFF :

The simulations below show some simulations with a model of the robot and the control system. Simulations are done with and without a band of friction and a backlash. In the simulations with friction and backlash the estimated values for the robot is used. These simulations result in discontinuous curve forms, the derivatives are infinite in some points. The simulations are done with and without velocity feedforward, VFF and Torque Feedforward,TFF.

Simulation characteristics:

a = 750 rad/s^2 : Acceleration reference
Kv2 = 2.65 (DACC2<2> ==> d2mxa = 15 000)
Kv3 = 1.17 (DACC2<3> ==> d2mxa = 15 000)
v3ttt = Velocity on arm 3 , top to top value, (also called velocity error).
p3ttt = Position error on arm 3, top to top value, (also called trajectory error).
Δt : Estimated as the time for the controller to control the references to a maximum position error less than 0.1 rad.
t12 : Time for constant acceleration / retardation in the position reference. The parameter is to be set in the position generator, INREF.
t23 : Time during constant velocity. This is a delay time for creating less tough references.

Direct Velocity Feedforward.

The references, position and velocity, both pass through a closed loop filter; DACC2 to limit the acceleration derivatives .

Simulations with friction and backlash in the robot.

The simulations in Table 1 and 2 are done with a robot model containing friction and backlash. The only difference in the conditions between the two simulations is that the simulation in Table 1. has a tougher references. In the simulations in Table 2 the velocity reference reaches a stable end value before the retardation starts. The parameter t23 sets the period of constant velocity. With this an overshoot caused by a tough reference and a lag in the position controller. Lag is the position error in the controller, which occurs when the controller is unable to follow the reference given. When the controller contains a time of stable reference the controller can get rid of the lag that is built up during the acceleration phase.

				Average	v3ttt	p3ttt	Max lag
VFF	TFF	Δp (rad)	Δt (s)	speed (Δp/Δt)	(rad/s)	(rad)	(rad)
NO	NO	7.47	0.49	15.1	0.97	0.057	3.9
YES	NO	7.47	0.25	29.8	4.0	0.13	0.37
NO	YES	7.47	0.25	15.1	0.62	0.013	3.9
YES	**YES**	**7.47**	**0.49**	**29.8**	**1.11**	**0.042**	**0.37**

Table 7.1. Velocity feedforward
Simulation characteristics:

t12 = 0.1 : Acceleration time
t23 = 0 : Time during constant velocity
The robot contains friction and backlash.

				Average	v3ttt	p3ttt	Max lag
VFF	TFF	Δp (rad)	Δt (s)	speed (Δp/Δt)	(rad/s)	(rad)	(rad)
NO	NO	15	0.58	25.9	1.13	0.075	6
YES	NO	15	0.35	43.4	2.8	0.11	0.38
NO	YES	15	0.58	25.9	0.31	0.011	6
YES	**YES**	**15**	**0.35**	**43.4**	**1.08**	**0.021**	**0.38**

Table 7.2. Velocity feedforward.

Simulation characteristics:

t12 = 0.1 : Acceleration time
t23 = 0.1 : Time during constant velocity
The robot contains friction and backlash.

In the first simulation in the tables the system is simulated without TFF and without VFF, this to get a reference for the other simulations. With VFF the controller error, lag is decreased from 3.9 rad to 0.37 rad in table 1. and from 6 to 0.38 rad in table 2.

With velocity feedforward,VFF the average speed is increased with 70 - 100 %. As shown in the tables above the controller error drastically decreases with VFF which was the objective. When VFF is used the position controller doesn't need to build up a larger position error before the controller reacts. The disturbances on arm 3 is increased with VFF. The arm position 3, velocity error increases from 0.97 rad/s to 4.0 rad/s with VFF, see Table 1.

When comparing the results in table 1 and 2 with VFF and TFF an overshoot caused by the tougher references. In the simulations in Table 1. a lag is build up. Normally the disturbances are supposed to be larger in the simulations in table 2, the average speed is larger, the trajectory length is longer. The delay time, t23 helps the controller to get rid of the lag build up in the simulations in Table 1, before the retardation starts. The overshoot is prevented.

Simulations without friction and backlash in the robot.

The simulations without friction and backlash are carried out because these have a curve form more according to the experimental results measured on the robot. They don't contain same number of discontinuouties, points with infinite derivatives, as the ones with friction and backlash.

				Average	v3ttt	p3ttt	Max lag
VFF	TFF	Δp (rad)	Δt (s)	speed (Δp/Δt)	(rad/s)	(rad)	(rad)
NO	NO	7.47	0.5	14.9	0.96	0.046	4.0
YES	NO	7.47	0.24	30.6	4.0	0.10	0.33
NO	YES	7.47	0.5	14.9	0.189	0.0073	4.0
YES	**YES**	**7.47**	**0.24**	**30.6**	**0.95**	**0.018**	0.33

Table 7.3. Velocity feedforward

Simulation characteristics:

t12 = 0.1 : Acceleration time
t23 = 0 : Time during constant velocity
The robot doesn't contain friction or backlash

				Average	v3ttt	p3ttt	Max lag
VFF	TFF	Δp (rad)	Δt (s)	speed (Δp/Δt)	(rad/s)	(rad)	(rad)
NO	NO	15	0.57	26.5	0.79	0.055	6
YES	NO	15	0.29	51.4	2.8	0.10	0.35
NO	YES	15	0.57	26.5	0.147	0.0075	6
YES	**YES**	**15**	**0.29**	**51.4**	**0.6**	**0.016**	**0.35**

Table 7.4. Velocity feedforward.

Simulation characteristics:

t12 = 0.1 : Acceleration time
t23 = 0.1 : Time during constant velocity
The robot doesn't contain friction or backlash

VFF increases the average velocity with at the most 100 % in the simulations presented above. VFF minimises the lag in both the simulations with and without a band of friction in the robot model. The overshoot caused by the lag is more obvious in the simulations with friction as without.

Simulation results with Velocity Feedforward with filtered position reference.

The simulations presented below are done with a filter between the position reference to the position controller.
The references from the path generator MOMGEN and MOMREF passes through the non linear feedback filter for acceleration derivative control, DACC2. The position reference from DACC2 is filtered with a rigid body model of the system, DAMOD. The filter, DAMOD containing a rigid body model of the robot and the gain Kv. This filter creates a position reference delayed as if it passed through a real robot and amplified by the velocity gain, Kv. The parameters J and K in the filter can set so that the position reference from the filter can be in phase and amplify identical with the real robot position. These modifications have been done to minimize the lag even more than in the simulations above with direct VFF.

Simulations with friction and backlash in the robot.

				Average	v3ttt	p3ttt	Max lag
VFF	TFF	Δp (rad)	Δt (s)	speed (Δp/Δt)	(rad/s)	(rad)	(rad)
NO	NO	7.44	0.50	14.9	0.98	0.056	4
YES	NO	7.44	0.30	25.0	3.74	0.12	0.22
NO	YES	7.44	0.50	14.9	0.76	0.015	4
YES	**YES**	**7.44**	**0.30**	**25.0**	**0.81**	**0.016**	**0.22**

Table 7.5. Velocity feedforward with filtered position reference.

Simulation characteristics:

t12 = 0.1 : Acceleration time
t23 = 0 : Time during constant velocity

VFF	TFF	Δp (rad)	Δt (s)	Average speed (Δp/Δt)	v3ttt (rad/s)	p3ttt (rad)	Max lag (rad)
NO	NO	15	0.57	28.5	1.13	0.076	6
YES	NO	15	0.42	36.0	2.64	0.11	0.28
NO	YES	15	0.57	28.5	0.36	0.011	6
YES	**YES**	**15**	**0.42**	**36.0**	**0.74**	**0.013**	**0.28**

Table 7.6. Velocity feedforward with filtered position reference.

Simulation characteristics:

t12 = 0.1 : Acceleration time
t23 = 0.1 : Time during constant velocity

Simulations without friction and backlash in the robot.

VFF	TFF	Δp (rad)	Δt (s)	Average speed (Δp/Δt)	v3ttt (rad/s)	p3ttt (rad)	Max lag (rad)
NO	NO	7.44	0.50	14.9	0.96	0.046	4.0
YES	NO	7.44	0.30	36.0	3.84	0.097	0.22
NO	YES	7.44	0.50	14.9	0.187	0.0072	4.0
YES	**YES**	**7.44**	**0.30**	**36.0**	**0.93**	**0.017**	**0.22**

Table 7.7. Velocity feedforward with filtered position reference.

Simulation characteristics:

t12 = 0.1 : Acceleration time
t23 = 0 : Time during constant velocity

The simulation results from Table 7 is shown in the Plots 1 - 4, VFF.

VFF	TFF	Δp (rad)	Δt (s)	Average speed (Δp/Δt)	v3ttt (rad/s)	p3ttt (rad)	Max lag (rad)
NO	NO	15	0.57	28.5	0.79	0.046	6.0
YES	NO	15	0.40	37.1	2.61	0.094	0.31
NO	YES	15	0.57	28.5	0.15	0.0072	6.0
YES	**YES**	**15**	**0.40**	**37.1**	**0.57**	**0.016**	**0.31**

Table 7.8. Velocity feedforward with filtered position reference.
Simulation characteristics:

t12 = 0.1 : Acceleration time
t23 = 0.1 : Time during constant velocity

PLOT 5 - 8, shows the results from the table above.

The best improvement, comparing the results achieved with velocity feedforward, is achieved with the simulations with friction and backlash. When both VFF and TFF are used, as in, for example, Table 2 and 6, the trajectory error is decreased by a factor 6, compare also Table 1 and 5.

In the simulations with only VFF the disturbances caused by the mass coupled inertia increase drastically. TFF is able to decrease the disturbances in position and velocity. However, the disturbance damping with TFF is more efficient in the simulations with a period of constant velocity. This means that VFF cannot be used separately in a robot implementation. When VFF is implemented TFF also has to be implemented. If not the path deviation will increase. Of special interest is the path deviation, p3ttt. With VFF the path deviation increases. However, with TFF the disturbances can be damped to the original value and lower.

PLOT 1 to 8. Velocity Feedforward shows the simulations with a filtered position reference. They are all simulated without friction and backlash.
In PLOT 4, an overshoot caused by a tough reference signal is obvious in the velocity on arm 3, v3. The peak in the velocity, v3 caused by a lag, is the reason why no improvement is seen in this simulation compared to the one without the filter in the position reference.

Conclusions, VFF :

With VFF the lag can be drastically decreased and the performance can be increased. However, with the increased performance the disturbances on other axes increase. To minimize the disturbances TFF was used.

With direct VFF and TFF :
* The lag can be decreased with a factor 15.
* The path deviation can be decreased with a factor 6, Table 2.
* The average speed can be increased with a factor 2.

With filtered VFF and TFF :
* The lag can be decreased with a factor 20, see Table 6.
* The path deviation can be decreased with a factor 5.8.
* The performance, and the average speed can be increased with a factor 2.4, see Table 7.

Chapter 8

Conclusions

Different methods were studied in order to decrease the path deviation caused by coupled mass inertia.

Acceleration derivative control, ADC
Transient controller disturbances can be decreased using smooth references.
By limiting the acceleration derivative in the position loop, some disturbances could be decreased. However, the path generation has to be adapted to the limitation to prevent overshoots.Limitation of both acceleration and retardation can cause stability problems. To limit both the acceleration and retardation derivative, the limitation has to be implemented as an external module and not in the position loop, so as not to cause stability problems. With a limited acceleration and retardation reference, either in the path generation or with an external closed loop feedback filter, stability problems are avoided. These two methods give about the same decrease of the disturbances.

Torque Feedforward, TFF
Torque feedforward between the disturbing and the disturbed axis is a possible method for compensating of the disturbances and the path deviation caused by coupled mass inertia.
Simulations have been carried out with different TFF models.

- Non-linear resonance TFF
- Linear resonance TFF
- Non-linear rigid body TFF
- Linear rigid body TFF

According to the simulation results, the path deviation can be significantly reduced with TFF. The four TFF modules studied gave similar results: the path deviation can be damped by a factor of six. However, the sample time is of great importance: with a sample time of 20 ms the path deviation cannot be damped at all; with a sample time of 12 ms the path deviation is damped by a factor of 2.0 - 2.5, depending on the TFF model used.

The linear rigid body TFF was implemented in the robot controller S3 and tested on IRB 2000, an industrial robot from ABB Robotics. The sample time in the implementation was 12 ms. The path deviations were studied with a pen, mounted at the tool centre point,TCP, drawing the trajectories on a paper, but also with recordings of the controller error in the position controller.
With TFF, the path deviation of the trajectories described on the paper were clearly improved and the controller error could be improved by almost a factor two, as predicted in the simulations.

Velocity Feedforward

Velocity feedforward has also been studied in order to increase the velocity performance of the robot. The velocity in the path generation is fed forward to the speed controller. With VFF, the controller error in the position loop is largely decreased a factor 15 - 40; the speed controller reacts quicker on references and the performance can be increased (75 - 100 % increase in average velocity),

The result from this work can be summarized as follows:

* Acceleration derivative control in the position reference, is a possible method of decreasing the disturbances in a controlled manner without stability problems.

* Non-linear feedback filter for acceleration derivative control of the position reference, can be used to implement an acceleration derivative control outside the position loop without causing stability problems. With a closed loop feedback filter, complicated algorithms for the path generation with limited acceleration derivatives can be avoided.

* Torque feedforward enables the path deviation caused by coupled mass inertia to be minimized.

* Simulations with different kinds of TFF modules, where the most complex one was one with a band of friction , a backlash and viscous damping and the simplest one a linear rigid body model, did not differ greatly. All of them were able to damp the path deviation by a factor six in the simulations with a sample time of 2 ms.

* It is of great importance that a definite implementation of TFF uses a sample time less than 6 ms.

* Velocity feedforward is a possible way of increasing the performance.
In the simulations the average speed can be increased 75 - 100 % with VFF.

* Combining VFF and TFF the average speed and the path deviation can be decreased simultaneously.

Discussion

The simulations and the experimental results indicate that it should be possible to increase the performance of a robot, both concerning path accuracy and average velocity, by means of torque feedforward and velocity feedforward. The torque feed forward should be made between axes with big coupled mass inertia and the velocity feedforward should be made on all axes. The torque feedforward was connected from the torque reference of the disturbing axis to the torque reference of the disturbed axes. The feedforward factor should be determined by the dynamic robot model already implemented in S3, which computes mass inertia.
In this work, gravity was not taken into account, which will give a big offset feedforward in the general case. When a final implementation is made, gravity and centripetal and coriolis torque should be subtracted from the torque reference of the disturbing axis, before the acceleration and the disturbance signal is estimated.

Another possibility to estimate the acceleration of the disturbing axes is to make use of the second derivative of the measured position signal. This will, however, give a too noisy or a too much delayed signal for the feedforward purpose.
Extrapolating the results of this work, it seems logical to proceed the developments according to the following steps:

1. Verification of the simulated TFF results at a sample time of 6 ms or less.
2. Implementation of ADC with a non-linear filter on the position reference. Verification of the simulated improvements, compared with the present SMACC- function.
3. Implementation of VFF and verification of the simulated results.
4. Evaluation of the improvements above of path accuracy and average velocity for typical trajectories.

The control schemes developed can be looked upon as a kind of computed torque. In the classic computed torque, the necessary motor torques are calculated from the reference path, solving the dynamic equations of the robot. In order to achieve correct computed torque signals, the inverse of the position and velocity loops must be used in the torque computations, which will probably be very difficult. In the control schemes developed in this work, the real path is used instead of the reference path as an input to the torque computations. Thereby, the inverse controller dynamics is not necessary and even non-linearities can be compensated for.
This technique can also be used for minimizing the error of the velocity loop. Instead of using the acceleration of the reference path, the velocity reference from the position loop can be used, after filtering in a model of the velocity loop. Combined with the VFF presented in this report, a very accurate and fast control should be possible.

Appendix

References

[1] Introduction to Robotics
Mechanics & Control
Addison - Wesly Publishing Company
John J. Craig

[2] Industrial robot system
IRB - 2000
ASEA Robotics

[3] Systemidentifiering av industrirobotsystem IRB - 6
Technical Report J-83-03
Klas Nilsson

[4] Tachofri reglering
Masters thesis
Stig Moberg

[5] A simnon tutorial
Department of Automatic Control
Lund Institute of Technology, July 1985
Karl Johan Åström

[6] Simnon
An interactive simulation program for non- linear systems
User's manual

[7] Computer Controlled Systems:
Theory and design
Prentice-Hall International Editions
Karl Johan Åström
Björn Wittenmark

[8] Programmeringsmanual
IRB 2000/3000