MASTER

Robustness of a second order sliding mode controller

Blom, A.

Award date:
1993

Link to publication
Robustness of a Second Order Sliding Mode Controller

A. Blom

WFW Report 93.007

Professor: Dr. Ir. J.J. Kok
Coach: Dr. Ir. A.G. de Jager

January 1993

Eindhoven University of Technology
Department of Mechanical Engineering
Division of Mechanical Engineering Fundamentals
# Contents

Summary ........................................ iii  
Notation ....................................... iv  

1. Introduction .................................... 1  
2. Sliding control for nonlinear systems ........... 4  
   2.1 Introduction .................................. 4  
   2.2 Preliminaries .................................. 4  
   2.3 MIMO sliding control with  
       second order sliding condition ............... 7  
   2.4 Conclusions .................................. 15  

3. Control of RT-robot ............................ 16  
   3.1 Description RT-robot ......................... 16  
   3.2 Desired trajectories ........................... 17  
   3.3 Unmodeled dynamics ......................... 18  
   3.4 Control parameters ........................... 20  
   3.5 Simulations ................................... 23  
       3.5.1 First setting ............................. 23  
       3.5.2 Error partition ........................... 26  
       3.5.3 PD analogy ............................... 26  
       3.5.4 Second setting ......................... 28  
       3.5.5 Damping ................................. 29  
   3.6 Conclusions .................................. 29  

4. Control of XY-table ........................... 31  
   4.1 Description XY-table ......................... 31  
       4.1.1 Stiff model .............................. 32  
       4.1.2 Flexible model ......................... 34  
   4.2 Desired trajectories ........................... 34  
   4.3 Unmodeled dynamics ........................... 35  
       4.3.1 System analysis .......................... 35  
       4.3.2 Model extension ......................... 36  
   4.4 Control parameters ........................... 39  
       4.4.1 SMC-controller ......................... 39  
       4.4.2 PD-controller ......................... 42
### 4.5 Experiments

- 4.5.1 Robustness to unmodeled dynamics ........................................ 43
- 4.5.2 Robustness to parameter variations ........................................ 45
- 4.5.3 Initial conditions ..................................................................... 46
- 4.5.4 Control of end-effector position ........................................... 47

### 4.6 Conclusions .............................................................................. 48

### 5. Conclusions .............................................................................. 49

### 6. References .................................................................................. 51

## Appendices

<table>
<thead>
<tr>
<th>Appendix</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>I/O-linearization</td>
<td>54</td>
</tr>
<tr>
<td>B</td>
<td>Sliding mode control with first order sliding condition</td>
<td>58</td>
</tr>
<tr>
<td>C</td>
<td>RT-robot trajectory</td>
<td>63</td>
</tr>
<tr>
<td>D</td>
<td>Motor data RT-robot simulations</td>
<td>64</td>
</tr>
<tr>
<td>E</td>
<td>Simulation algorithm RT-robot</td>
<td>65</td>
</tr>
<tr>
<td>F</td>
<td>Flexible model XY-table</td>
<td>66</td>
</tr>
<tr>
<td>G</td>
<td>Transfer functions XY-table</td>
<td>68</td>
</tr>
<tr>
<td>H</td>
<td>Kalman observer XY-table</td>
<td>70</td>
</tr>
<tr>
<td>I</td>
<td>Controller settings and constants</td>
<td>73</td>
</tr>
</tbody>
</table>
Summary

Mechanical manipulators are controlled in order to make their end-effector track a desired trajectory, with a desired velocity. Many linear and non-linear control methods are available to achieve this goal, all with their own (dis)advantages. The control is often based on a mathematical model that represents the dynamic behaviour of the system.
In practice however, this model is never an exact representation of reality. There are always phenomena like unmodeled dynamics, wrongly tuned parameters and measurement noise that cause model imprecisions. These imprecisions may come from actual uncertainty about the system, or from the purposeful choice for a simplified representation of the system dynamics. The less the control system is affected by these modeling errors, the more robust the controller is said to be.

In this report the robustness of a 'Second Order Sliding Mode Controller' is investigated. This controller can be used when the model structure itself is inaccurate, but the inaccuracies are bounded with known bounds. Once we have gathered information about these model uncertainties we can (1) use our knowledge to estimate the uncertainty bounds, but it is possible also (2) to extend our model. Both methods are described in this report.

The investigation into the robustness consists of four parts:
- Theory
- Simulations (RT-robot)
- Experiments (XY-table)
A comparison with a PD-controller will be made.

Simulations and experiments were done with parametric uncertainties as well as unmodeled dynamics. During simulations these uncertainties are obtained by introducing a nominal design model and an evaluation model in which a motor has been added and the mass parameters can be changed. In the practical situation of the experiments (of course) many model imprecisions are present.

The main conclusions of this report are that as far as robustness to unmodeled dynamics is concerned there is no advantage in using a 'Second Order Sliding Mode Controller' compared to a PD controller, however when the performance and the robustness to parametric uncertainties are concerned, the 'Second Order Sliding Mode controller' is slightly better than a PD controller.
Finally we conclude that 'knowledge of the system to be controlled is better than knowledge of advanced control'.
Notation

\(a\) : scalar
\(A\) : matrix
\(a\) : column
\(\dot{a}, \ddot{a}\) : estimation of scalar or column
\(a_d, a_d\) : desired scalar or column
\(|A|, |a|\) : matrixnorm resp. vectornorm
\(A^{-1}\) : inverse of matrix
\(A^T, a^T\) : transposition of matrix or column
\(I\) : unity matrix
\(\dot{a}, \ddot{a}\) : first order time derivative of scalar or column
\(\ddot{a}, \dddot{a}\) : second order time derivative of scalar or column
\(a^{(n)}\) : \(n\)-th order time derivative of scalar
\(a^{(n)}\) : '\(n\)-th order time derivative' of column:
\([a_1^{(n_1)}, a_2^{(n_2)},..., a_k^{(n_k)}]^T\), with \(n_1 + n_2 + ... + n_k = n\)
\(g(x)\) : vectorfunction of \(x\)
\(g_i(x)\) : \(i\)-th element of \(g(x)\)
1. Introduction

Advanced manipulators (or robots) are increasingly applied in industry for several tasks, to eliminate unpleasant work and/or to increase the productivity and quality. Some important applications are welding, dyeing, assembling, glueing, testing, and loading of production machines.

In the early years of robot applications, designing a robot was mainly a construction problem: in order to design manipulators that could perform tasks accurately, designers used to make stiff but also heavy constructions. The price we had to pay for this accuracy was however a limitation in speed and high costs of energy and production.

Nowadays designing a robot has more and more become a control problem: to increase the speed, designers had to reduce the construction’s weight, often resulting in less stiff manipulators. Of course, the accuracy still had to be guaranteed. To fulfil this requirement on the dynamic behaviour, it is of advantage to use model based control.

As we know, a model often is only a simple representation of reality, since during the modeling process only a selection of the relevant aspects of the physical system can be made from an overwhelming number of possibilities (each relevant aspect can also not been taken into account exhaustively). Inevitable inaccuracies are made. According to Slotine and Li [1991 (pp. 267)] these inaccuracies can be classified into two major categories (we remark that other classifications are possible):

- **Structured (parameter) uncertainties:**
  e.g. imprecision on the mass properties or loads, inaccuracies on the torque constants of the actuators, friction, and so on;

- **Unstructured uncertainties (unmodeled dynamics):**
  i.e. underestimation of the system’s order, e.g. structural resonant modes or neglected timedelays (in the actuator for instance).

The presence of these inaccuracies often requires a robust control algorithm. 'Robustness is the sensitivity to effects which are not modeled in the design, such as disturbances, measurement noise, unmodeled dynamics, etc. The system should be able to withstand these neglected effects when performing the tasks of interest’ (Slotine and Li [1991] pp. 197).
In robust control, the controller is designed based on considerations of both the nominal model and some characterization (specification) of the model uncertainties. This does however not implicate that robust control will always perform better than non-robust control. Generally, a non-robust control algorithm (based on the nominal situation) will perform better in the absence of model errors. See for an impression of robustness to parameter variations fig. 1.1.

Adaptive control algorithms are often a very effective way of dealing with structured uncertainties, i.e. constant (or slowly-varying) parametric uncertainties. A disadvantage of this approach is however the inherent inability to adapt to unstructured uncertainties.

A class of controllers that is said to be robust (Chang [1990], Utkin [1976], Slotine and Li [1991], e.a.) to unstructured uncertainties are 'Sliding Mode Controllers'. These controllers can be used when the model structure itself is inaccurate, but the inaccuracies are bounded with known bounds.

Unfortunately, till now (using a first order sliding condition), the tuning capability was limited and in some cases there existed a trade-off between tracking accuracy and the robustness to model errors. Recently however, sliding control has been further enhanced in order to eliminate these limitations, applying a second order sliding condition. In this report an investigation is done into the robustness of this 'Second Order Sliding Mode Controller'.
In chapter 2 the theoretical properties of the sliding mode controller are explained and discussed. After an I/O linearization technique the sliding mode control with a second order sliding condition will be presented.

In chapter 3 the results of simulations with an RT-robot are presented. This robot is chosen because it can be described by two coupled differential equations (MIMO), containing Coriolis and centripetal forces (i.e. nonlinear). After the presentation of the mathematical model of the robot, results will be presented using a Second Order Sliding Mode Controller, with unmodeled dynamics caused by several motors. The results will be compared with a PD-controller.

In chapter 4 results of experiments with an XY-table are presented, to investigate the controller in an experimental environment. The nominal model consists of two uncoupled differential equations, containing Coulomb and viscous friction terms. Model errors are present, which make these experiments suitable for investigation into robustness. The nominal XY-table model has been extended also in an attempt to get better results. Again the results will be compared with a PD-controller.

In chapter 5 the main conclusions are drawn and recommendations are given for further research.
2. Sliding control for nonlinear systems

2.1 Introduction

The idea behind sliding mode control is to choose a suitable surface (hyperplane) in the state space, called the switching surface, and switch the control input on this surface. The control input is then chosen to guarantee that the trajectories from all initial states are directed towards the surface within a finite time. Once the system is trapped on the surface, the closed loop dynamics are completely governed by the equations that define the sliding surface. We expect that in this way the closed loop dynamics of the system will be independent of model errors and robustness is achieved.

To achieve zero tracking error despite model uncertainties and disturbances in the system, we will use a 'Second Order Sliding Mode Control'-strategy proposed by Chang [1990]. This control strategy will be described in the next paragraphs.

2.2 Preliminaries

To present the Sliding Mode Controller (SMC) for nonlinear systems, we consider a nonlinear system that can be described (exactly) by a MIMO model of general form, linear in the control \( u \) (affine):

\[
\dot{x} = f(x,t) + \sum_{i=1}^{m} g_i(x)u_i
\]

\[ y = h(x) \]  

\[
\text{with:} \quad x \in \mathbb{R}^n \quad \text{(state vector)} \\
\quad u \in \mathbb{R}^m \quad \text{(input vector)} \\
\quad y \in \mathbb{R}^k \quad \text{(output vector) \quad k \leq m} 
\]

The functions \( f(x,t) \), \( g_i(x) \) and \( h(x) \) are nonlinear functions of their arguments, that we assume to be differentiable a sufficient number of times.

We assume that to control the system a mathematical model of the system is available and written to be:
\[ \dot{x} = \hat{f}(x,t) + \sum_{i=1}^{m} \hat{g}_i(x)u_i \quad (2.2) \]

\[ y = h(x) \]

Notice that \( h(x) \) is assumed to be exact. Model imprecisions may come from actual uncertainty about the plant, or from the deliberate choice for a simplified representation of the system dynamics. We assume:

\[ f(x,t) = \hat{f}(x,t) + \Delta f(x,t) \quad (2.3) \]

\[ g_i(x) = \hat{g}_i(x) + \Delta g_i(x) \]

The modeling uncertainties \( \Delta f(x,t) \) and \( \Delta g_i(x) \) are unknown, but bounded (with known bounds).

Strictly we cannot make this assumption when we are dealing with unmodeled dynamics, i.e. when the system (eq. 2.1) is of higher order (say: \( l \)) than the order \( n \) of the model (eq. 2.2). In that case we assume the system to be described by eq. 2.4.

\[ \dot{x}^* = \begin{bmatrix} \dot{x} \\ \dot{\dot{x}} \end{bmatrix} = \begin{bmatrix} f_1(x^*,t) \\ f_2(x^*,t) \end{bmatrix} + \begin{bmatrix} G_1^*(x^*) \\ G_2^*(x^*) \end{bmatrix} u \quad (2.4) \]

with \( x^* \in \mathbb{R}^l \quad x \in \mathbb{R}^n \quad \dot{x}' \in \mathbb{R}^{l-n} \)

However, to avoid unnecessary complications we hold on to eq. 2.1.

The goal of the control problem is to guide \( y(x) \) along a desired trajectory \( y_d(t) \), keeping the whole state bounded.

Often a model is described by coupled and nonlinear differential equations. Therefore, it is not easy to see how the input \( u \) can be designed to control the tracking behaviour of the output \( y \). The difficulty of the control design can be reduced if we can find a simple relation between the system output \( y \) and the control input \( u \). To generate this relation we use the so-called Input/Output-linearization approach [Isidori, 1989], described in appendix A.
In the I/O-linearization process the dynamics of a general nonlinear MIMO system is decomposed into an external (I/O) part and an internal ('unobservable') part, called the 'internal dynamics'. Since the external part consists of a linear relation between $y$ and a 'synthetic input' $\upsilon$, it is easy (in the unperturbed case) to design this input $\upsilon$ so that the output $y$ behaves as desired. Then, the question is whether the internal dynamics will also behave well (bounded). This is guaranteed if the zero-dynamics of the system (intrinsic to the nonlinear system) behaves exponentially stable, i.e. if the system is asymptotically minimum phase (Slotine and Li [1991] pp. 224).

Once having a system in a so called 'normal form' (with exponentially stable zero dynamics) we can design a sliding mode controller, using a state feedback control law with synthetic input $\upsilon$ and using a linear sliding mode controller.

In order to keep up with notations used by many authors, who choose the 'controllability canonical form' (the external (I/O) part of the normal form) as a starting point, we rewrite equation 2.1 into (it is recommended to read appendix A first):

$$\dot{x} = f(x) + B(x)\mu$$
$$\dot{f}(x) = \hat{f}(x) + \Delta f(x)$$
$$B(x) = \hat{B}(x) + \Delta B(x)$$

The I/O-linearizing state feedback control law will be:

$$\mu = \hat{B}^{-1}(\upsilon - \hat{f})$$

In order to avoid unnecessary complications we usually restrict ourselves to square systems ($k = m$) so that $\hat{B}$ is square.

Thus:

$$\dot{x} = \upsilon + \Delta f + \Delta B\hat{B}^{-1}(\upsilon - \hat{f})$$
Because sliding control requires that the uncertainties are bounded with known bounds, a general confinement is:

\[
\| \Delta B(x) \| \leq \beta \quad \forall \ x \\
\| \Delta f(x) \| \leq \alpha \quad \forall \ x
\]  

(2.8)

### 2.3 MIMO sliding control with second order sliding condition

In appendix B a sliding control with a first order sliding condition is given (for SISO systems), to show that the tuning capability is limited. In order to increase the capabilities, recently the sliding control has been further enhanced, resulting in a MIMO sliding control with a second order sliding condition.

We consider a general MIMO system, described in controllability canonical form by eq. 2.5. The output variables of interest form an output vector \( y \) (with \( y^{(n)} = x^{(n)} \); see notation). Note that when the system has internal dynamics, this dynamics must behave bounded.

The available mathematical model of the system is given by:

\[
x^{(n)} = \hat{f}(x) + \hat{B}(x)u
\]  

(2.9)

A second order Sliding Mode Control strategy by Chang [1991] defines a zero \( z_p \), in the 'error dynamics', described by eq. 2.10.

\[
\left( \frac{\partial}{\partial t} + z_0 \right) s_i = \prod_{j=1}^{n} \left( \frac{\partial}{\partial t} + \lambda_{ji} \right) \int_{0}^{t} e_i d\tau
\]  

(2.10)

\[
e_i = x_i - \hat{x}_{id}
\]

\( i = 1 \ldots k \)

This equation is a set of band-pass filters where the break frequencies are determined by the selection of the poles (\( \lambda_{ij} \)) and zero (\( z_0 \)). An integral term in the equation assures zero steady-state errors. Writing eq. 2.10 in the unfactored (polynomial) form we get (see Chang [1990]):
Sliding control for nonlinear systems

\[ \dot{\mathbf{s}} + Z_0 \mathbf{s} = \mathbf{g}^{(n-1)} + C_{n-1} \mathbf{g}^{(n-2)} + \ldots + C_1 \mathbf{g} + C_0 \int_0^t \mathbf{g} \, dt \] (2.11)

Where \( \dot{\mathbf{s}} = \mathbf{0} \) represents the sliding hyperplanes. Note that \( Z_0 \) and \( C_{n-1}, \ldots, C_0 \) are (diagonal) matrices.

The relation between \( \mathbf{g} \) and the control input can be obtained by differentiating equation 2.11.

\[ \dot{\mathbf{g}} + Z_0 \dot{\mathbf{g}} = \mathbf{g}^{(n)} + C_{n-1} \mathbf{g}^{(n-1)} + \ldots + C_1 \mathbf{g} + C_0 \mathbf{g} \]

\[ \dot{\mathbf{g}} + Z_0 \dot{\mathbf{g}} = \mathbf{g}^{(n)} + \mathbf{e}_p \] (2.12)

\[ \mathbf{e}_p = C_{n-1} \mathbf{g}^{(n-1)} + \ldots + C_1 \mathbf{g} + C_0 \mathbf{g} \]

\( \mathbf{e}_p \) can be computed with (full state) feedback signals \([\mathbf{x}, \ldots, \mathbf{x}^{(n-1)}]\), implying a full state measurement, however knowing the successive integrator relationship allows us to evaluate the successive derivatives as needed (differentiating).

Substituting equation 2.5 in 2.12 yields:

\[ \dot{\mathbf{g}} + Z_0 \dot{\mathbf{g}} = \mathbf{f} + B \mathbf{u} - \mathbf{x}_d^{(n)} + \mathbf{e}_p \] (2.13)

The role of the control \( \mathbf{u} \) is to control \( \mathbf{g} \) such that the sliding surface \( (\dot{\mathbf{g}} = \mathbf{0}) \) can be reached within a finite time. Once \( \mathbf{g} \) reaches the sliding surfaces, zero steady-state tracking error can be achieved on the sliding surface with stable error dynamics (eq. 2.10/2.11).

To investigate stability with the second method of Lyapunov, a positive definite Lyapunov function candidate is selected as:

\[ V = \frac{1}{2} \mathbf{g}^T \mathbf{g} + \frac{1}{2} \mathbf{g}^T \Omega \mathbf{g} \] (2.14)

\[ \Omega = \text{diag}(\omega_n^2) \]
Negative definiteness condition of the Lyapunov stability criterion can be written as:

\[ \dot{V} = \dot{s}^T (\dot{s} + \Omega s) < 0 \quad (2.15) \]

which constitutes the attractivity condition for the Sliding Mode Controller towards the sliding hyperplane i.e. \( \dot{s} = 0 \):

\[ \dot{V} = \dot{s}^T (\dot{s} + Z_0 \dot{s} + \Omega s - Z_0 \dot{s}) < 0 \]

\[ \dot{V} = \dot{s}^T (f + B u - x_d + \varepsilon_p + \Omega s - Z_0 \dot{s}) < 0 \quad (2.16) \]

To reach the sliding surface within a finite time (despite model errors) the control has to provide a switched action. This control is designed as:

\[ u = \hat{B}^{-1} (\hat{u} - k \text{sign}(\dot{s})) \quad (2.17) \]

\[ \hat{u} = -\hat{f} + x_d + \varepsilon_p + Z_0 \dot{s} + \Omega \dot{s} \]

Equation 2.17 is used to relate the gain \( k \) with the system uncertainties. Substitution in eq. 2.16 and reorganizing yields:

\[ \dot{V} = \dot{s}^T (\Delta f + \Delta B \hat{B}^{-1} \hat{u} - k(I + \Delta B \hat{B}^{-1}) \text{sign}(\dot{s})) \quad (2.18) \]

This equation indicates a condition for the gain \( k \) to guarantee the system stability. To quantify the gain, we assume:

\[ \dot{s}^T (\Delta f + \Delta B \hat{B}^{-1} \hat{u} - k(I + \Delta B \hat{B}^{-1}) \text{sign}(\dot{s})) \leq \]

\[ ||\dot{s}|| (||\Delta f|| + ||\Delta B|| ||\hat{B}^{-1} \hat{u}|| - k(1 - ||\Delta B|| ||\hat{B}^{-1} \text{sign}(\dot{s})||)) < 0 \quad (2.19) \]
Finally, the gain $k$ is found as:

$$
k > \frac{\alpha + \beta \| \hat{B}^{-1} \hat{u} \|}{1 - \beta \| \hat{B}^{-1} \text{sign}(\hat{s}) \|} \tag{2.20}
$$

$$
\beta \| \hat{B}^{-1} \text{sign}(\hat{s}) \| < 1
$$

For control design purposes, the minimum value of $k$ is selected, since the least control effort is desired. Notice that $k$ is strongly dependent on the desired trajectory and initial condition through $\hat{u}$.

Summarizing the expression for the control $u$ becomes:

$$
u = \hat{B}^{-1} \left[ x^o - e_p + Z_\psi \hat{s} - \Omega \hat{s} - k \text{sign}(\hat{s}) - \hat{f} \right] \tag{2.21}
$$

with $k$ according to equation 2.20. We get the 's-dynamics' by substituting eq. 2.21 in eq. 2.13:

$$
\hat{s} + kB\hat{B}^{-1} \text{sign}(\hat{s}) - \Delta B\hat{B}^{-1}Z_\psi \hat{s} + B\hat{B}^{-1} \Omega \hat{s} = \Delta \hat{f} + \Delta B\hat{B}^{-1} \left( x^o - \hat{f} - e_p \right) \tag{2.22}
$$

As we can see from eq 2.21 the control law is discontinuous across $\hat{s}$, which leads to chattering. In general, chattering must be eliminated for the controller to perform properly. This can be achieved by smoothing out the control discontinuity, in a boundary layer neighbouring the switching surface $(\hat{s} = 0)$. Therefore we don’t use the 'signum'-function (sign), but we apply the 'saturation'-function (sat) instead, with a boundary layer width $\Phi$:

$$
sat(\hat{s}, \Phi) = \begin{cases} 
1 & \hat{s} > \Phi \\
\frac{\hat{s}}{\Phi} & |\hat{s}| < \Phi \\
-1 & \hat{s} < -\Phi 
\end{cases} \tag{2.23}
$$

During the transient phase of the dynamics $\hat{s}$ may move in and out of this boundary layer. However, once Lyapunov function $V < \Phi^2/2$ is reached, $\hat{s}$ remains in this layer (see Elmali and Olgac [1992]).
The s-dynamics within the boundary layer become:

\[
\frac{d\hat{s}}{dt} + \frac{k}{\Phi} BB^{-1} \hat{s} - \Delta B \hat{B}^{-1} Z_{\omega} \hat{s} + B \hat{B}^{-1} \Omega \hat{s} = \Delta f + \Delta B \hat{B}^{-1} (\hat{x}_d^{(n)} - \hat{f} - e_p)
\]

(2.24)

This equation represents a set of 2nd order low-pass filters. In comparison with the 1st order s-dynamics derived from a 1st order sliding condition (appendix B), the 2nd order s-dynamics has more tuning parameters. Not only can \(\Phi\) be set, but also the zeros \(Z_\omega\) and the 'spring constants' \(\omega_m^2\). The overall control strategy is represented by the block diagram of fig. 2.1.

Figure 2.1; block diagram representation of SMC strategy

The loop closure in the synthesis view of sliding control is due to a non-zero \(\Delta B\) and therefore to a non-zero dummy variable \(\bar{b}\). The blocks in this synthesis view are illuminated by eq. 2.25.
model errors = $\Delta f + \Delta B\dot{B}^{-1}(\dot{\chi}_d^{(n)} - \dot{f})$

1) $s(p) = \left(p^2I + \left(\frac{k}{\Phi}B\dot{B}^{-1} - \Delta B\dot{B}^{-1}Z_o\right)p + B\dot{B}^{-1}\Omega\right)^{-1} u'(p)$

2) $e(p) = p(pI + Z_o) \left(p^nI + C_{n-1}p^{n-1} + \ldots + C_0\right)^{-1} s(p)$

3) $b(p) = \Delta B\dot{B}^{-1}(C_{n-1}p^{n-1} + \ldots + C_0) e(p)$

$p = \text{Laplace operator}$

(2.25)

To analyse the system dynamics we first take a close look at the $s$-dynamics in case the sign-function is used and the unmodeled dynamics are neglected. In that case it holds that:

$$\dot{s} + k\cdot \text{sign}(s) + \omega^2 s = 0$$

(2.26)

The phase portrait is given in fig 2.2. We see that after a transient, during which $s$ can oscillate, $s$ remains within an certain area: $|s| < k/\omega^2$ and $\dot{s}=0$.

One can derive from eq. 2.26 that the transient will last approximately $t_s$ seconds in which approximately $n_o$ oscillations will be made:

$$t_s = \frac{s_0\omega \pi}{2k}$$

$$n_o = \frac{s_0\omega^2}{4k}$$

(2.27)

It is difficult to find the exact mathematical solution of eq. 2.26, especially once $|s| < k/\omega^2$. We assume that after the transient $\dot{s}$ will be zero and $s$ has a certain (constant) value. In a practical implementation $\dot{s}$ will not remain zero, but will oscillate slightly (due to limited sample frequency) such that $s$ tends to zero.

When a boundary layer is defined (i.e. the sign-function is replaced by the sat-function) the $s$-dynamics outside the boundary layer stays the same. Inside the boundary layer the $s$-dynamics is described by a (standard) second order system.
Secondly we have a look at the error dynamics, described by eq. 2.10/2.11. Simple analysis yield that the step-response (in case of a second order system) will be:

\[ y(t) = \left(1 + (\omega - \lambda) t\right)e^{-\lambda t} e(t) \]  

(2.28)

In which \( e(t) \) is a unity-step.

The analysis of the dynamic nature of the control loop (synthesis view, fig 2.1) is not possible by observing the s-dynamics and error dynamics without considering the contributions of the feedback element. The lack of tools in nonlinear systems theory, however, creates a problem; a systematic way of selecting the parameters \( z, \omega, \Phi, C_{n-1}...C_n \) and \( k \) does not exist. However, the influence of the tuning parameters can be evaluated.

First we consider the gain \( k \). This control parameter is completely determined by the confinements on the uncertainties and can be calculated with eq. 2.20. As mentioned before, for control design purposes, the minimum value of \( k \) (according to eq. 2.20) is selected, since the least control effort is desired.
Secondly we investigate the influence of the zero $z_o$ in the error-dynamics. Chang [1991] suggested that greater values of the placed zeros provide more damping in the s-dynamics. As we can see from eq. 2.22/2.24, $z_o$ indeed contributes in the damping of the s-dynamics, but the exact influence is not clear, since the total damping can in- and decrease, depending on the unknown $\Delta B$. If $\Delta B$ is 0 (zero-matrix), $z_o$ has no influence at all in the s-dynamics.

Simulations, however, indeed show more 'damping' when $z_o$ is increased. This is very likely due to the error-dynamics, in whose response $z_o$ has a 'damping' contribution, recognized as a reduction of the rise-time in eq. 2.28 (strictly 'rise-time' would therefore be a better term than 'damping'). Besides, one can derive from eq 2.28 that $z_o$ is preferably smaller than two times $\lambda$ to prevent 'negative response' (like not-minimum-phase systems).

Third we investigate the influence of $\omega_n$. As we can see from the s-dynamics (eq. 2.22) $\omega_n$ sets the break-frequency, in contradiction with a first order SMC (appendix B), independent of $\Phi$. Preferably we choose this break-frequency (determining the s-dynamics bandwidth) smaller than the lowest unmodeled structural resonant mode.

To simplify the choice of the poles in the error dynamics (by setting $C_n$ ... $C_b$) we choose the bandwidth of the error dynamics the same as the bandwidth ($\omega_n$) of the s-dynamics. It is not necessary to set the bandwidth much higher (hoping for a faster response), because the input signal to the error dynamics doesn't contain higher frequencies than $\omega_n$. Besides setting the bandwidth of the error-dynamics higher than the lowest structural resonant mode yields a less effective filtering (but still 2nd order) of high frequency unmodeled dynamics, than in case of equal bandwidth's. On the other hand, a large error-dynamics-bandwidth has a positive effect on the tracking error as we will see at the end of this paragraph. Note also that the higher the bandwidth of the s- and error-dynamics, the higher the control effort will be.

Fourth, the choice of $\Phi$. As we know, the damping in the s-dynamics ($k/\Phi$) affects the bandwidth in negative sense, because usually $k/\Phi>>1$ (heavily damped). Simple analysis, based on the response of standard second order systems, yields (see Cool, Schijff and Viersma [1985] pp. 170):

$$\omega_{bandwidth}^2 = \frac{-(k/\Phi)^2 + 2\omega_n^2 + \sqrt{(2\omega_n^2 - (k/\Phi)^2)^2 + 4\omega_n^2}}{2} \quad (2.29)$$
We will derive an expression for the tracking error, the same way we did for the traditional sliding mode controller (Appendix B; eq. B.15). With help of Lyapunov we guaranteed $\dot{s} < 0$ (after a transient). Noting that $s$ contains no frequencies higher than $\omega_{\text{bandwidth}}$ we find (as an approximation): $\dot{s} < \omega_{\text{bandwidth}} \Phi$, so that the maximal tracking error will be (once the $\dot{s} < 0$):

$$|e| < \frac{\Phi(\omega_{\text{bandwidth}} + z_{n})}{\lambda_1 \lambda_2 \ldots \lambda_n}$$ (2.30)

With a guaranteed precision $\varepsilon$.

However, we can't make the tracking error arbitrarily small, choosing $\Phi$ smaller and smaller. Since the implementation has a limited sample frequency it is possible that $s$ will move out of the boundary layer between two sample moments. Therefore $\Phi$ is bounded by the sample frequency. Besides we see from eq. 2.30 that the zero affects the tracking accuracy in negative sense.

### 2.4 Conclusions

Theoretically stability of the Second Order Sliding Mode Controller can be proven in the face of modeling imprecisions, as long as they are bounded with known bounds. The gain $k$ is a measure of these uncertainties and is determined by eq. 2.20. It is however strongly dependent on the desired trajectory and the initial condition.

The parameter $\omega_n$ (determining the bandwidth of the s-dynamics) must be smaller than the frequency of the lowest unmodeled structural resonant mode, guaranteeing robustness to high frequency unmodeled dynamics.

The problem of the limited tuning capability of a First Order Sliding Mode Controller has been solved and a possible trade-off between tracking accuracy and robustness to high-frequency unmodeled dynamics has been eliminated since $\omega_n$ can be set independently of $\Phi$.

The bandwidth of the error-dynamics is chosen equal to the bandwidth of the s-dynamics. $\Phi$ is limited by the sample frequency and must be chosen high enough to eliminate chattering, and small enough to achieve a small tracking error. The zero in the error-dynamics provides damping, but has negative influence on the tracking error.
3. Control of RT-robot

To investigate the second order sliding mode controller, simulations will be done with an RT-robot.

3.1 Description RT-robot

The RT-robot has two degrees of freedom ($r(t)$ and $\varphi(t)$) and consists of a disc with moment of inertia $J$ and a rigid bar with length $l$ and homogeneously distributed mass $m$. The payload at the end of the bar is a concentrated mass $m_1$. The bar can be moved inside the disc by means of a force $F(t)$. The disc is driven by a torque $T(t)$. In figure 3.1 a schematic top view representation of the RT-robot is given.

![Figure 3.1; RT-robot](image)

If there is no friction, the equations of motion can be derived with the method of Lagrange (see also Van Gerwen [1990]):

\[
P_1 \ddot{r} - (P_1 r - P_2) \dot{\varphi}^2 = F(t)
\]

\[
(P_3 - 2P_2r + P_1r^2)\ddot{\varphi} + 2(P_1 r - P_2)\dot{r} \dot{\varphi} = T(t)
\]
The parameter values are chosen the same as used in previous investigations (in order to allow comparison):

\[ P_1 = m + m_1 = 15 \, [kg] \]
\[ P_2 = \frac{1}{2} ml = 5 \, [kgm] \]  \hspace{1cm} (3.2)
\[ P_3 = J + \frac{1}{3} ml^2 = 8 \frac{1}{3} \, [kgm^2] \]

Writing the equations of motion in state space description yields:

\[ \dot{x}_1 = x_2 \]
\[ \dot{x}_2 = (x_1 - \frac{P_2}{P_1})x_4^2 + \frac{F}{P_1} \]
\[ \dot{x}_3 = x_4 \]
\[ \dot{x}_4 = \frac{2(P_2 - P_1x_1)x_2x_4 + T}{P_3 - 2P_2x_1 + P_1x_1^2} \]  \hspace{1cm} (3.3)

with \( x = [r, \dot{r}, \varphi, \dot{\varphi}]^T \)

### 3.2 Desired trajectories

The goal of the control problem is to guide the effector (i.e. payload \( m_i \)) along a circular trajectory with constant velocity. In cartesian end-effector space the desired trajectory is:

\[ x_d = \begin{bmatrix} x_c + r_e \cos(\omega t) \\ y_c + r_e \sin(\omega t) \end{bmatrix} \]

\[ x_c = 0.5 \, [m] \]
\[ y_c = 0 \, [m] \]
\[ r_e = 0.25 \, [m] \]
\[ \omega = \pi \, [rad/s] \]  \hspace{1cm} (3.4)

The description of this trajectory in polar coordinates is given in appendix C as well as the first and second time derivatives.
3.3 **Unmodeled dynamics**

Equation 3.1 shows that the systems input consists of a force $F(t)$ and a torque $T(t)$. In reality, however, the robot is driven by (servo)motors, whose output (force/torque) are generally not proportional to their input (voltage). In order to approach reality as closely as possible, and to investigate robustness, the simulations are performed with a servomotor having its own dynamics, neglected in the controller design process (i.e. unmodeled dynamics). So to assess robustness, different models are used for design and evaluation, i.e. a nominal design model and an evaluation model are introduced. The blockdiagram of the evaluation model is given in figure 3.2.

![Blockdiagram of the evaluation model](image)

**Figure 3.2; evaluation model**

- $U$ = control input [Nm]
- $U'$ = motor voltage [V]
- $I$ = motor current [A]
- $k_m$ = motor constant [Nm/A] or [Vs/rad]
- $L$ = (self)induction [H]
- $R$ = resistance [$\Omega$]
- $T$ = motor torque [N]
- $i$ = transmission ratio
- $\varphi_m$ = angular displacement motor [rad]
- $\varphi_s$ = angular displacement system [rad]
- $p$ = Laplace operator
For the linear movement we assume a gearwheel/gearrack transmission with transfer ratio \( i_1 = 0.05 \, [\text{m/rad}] \). For the rotational movement we assume a gear reduction (for instance a gearbox) with transfer ratio \( i_2 = 0.04 \, \text{i} \) (\( i = \omega_{\text{out}}/\omega_{\text{in}} \). The moments of inertia of the transmission are neglected).

From figure 3.2 we conclude:

\[
\dot{i} = \frac{1}{L} \left( \frac{iR}{k_m} \ - \ \frac{k_m}{i} \dot{\phi}_{\text{syst}} \ - \ RI \right)
\]

\[
T_{\text{syst}} = U - \frac{k_m^2}{i^2 R^2} \ddot{\phi}_{\text{syst}} - \frac{k_m}{iR} \dot{i}
\]  

(3.5)

Analogously this equation holds for the translational movement \( (F_{\text{syst}} \; \text{instead of} \; T_{\text{syst}}) \).

In state space description, the evaluation model is represented by:

\[
\begin{bmatrix}
\dot{x}_1 \\
\dot{x}_2 \\
\dot{x}_3 \\
\dot{x}_4 \\
\dot{x}_5 \\
\dot{x}_6
\end{bmatrix} =
\begin{bmatrix}
0 \\
0 \\
0 \\
1 \\
1 \\
1
\end{bmatrix}
- \frac{1}{L_1} \left( \frac{k_{m_1} x_2 - R_1 x_3}{i_1} \right) \begin{bmatrix}
0 \\
0 \\
0 \\
i_1 R_1 \frac{U_r}{L_1 k_{m_1}} \\
i_2 R_2 \frac{U_q}{L_2 k_{m_2}}
\end{bmatrix}
\]

\[
\begin{bmatrix}
x_2 \\
x_4 \\
x_6
\end{bmatrix}
= \frac{\left( P_2 - \frac{P_1 x_1}{P_1} \right) x_2 x_4 + \frac{k_{m_2}}{i_2} x_5}{P_3 - 2 P_2 x_1 + \frac{x_1}{P_1}}
\]

(3.6)

with \( \dot{x} = [ r, \dot{r}, \phi, \dot{\phi}, I_r, I_\theta ]^T \)

Since the design model is given by equation 3.3, we see that the evaluation model is of higher order than the design model, i.e. we indeed have unstructured uncertainties / unmodeled dynamics (referring to eq. 2.4). Three servo motors have been selected to perform the simulations (Electro Craft, Honeywell and Mavilor). The motor data is listed in appendix D.
Control of RT-robot

For servomotors we can define two timeconstants:

**Electrical timeconstant**, the first-order time constant of the electrical switching on phenomenon (transient) with fixed motor \((\varphi_m = \theta)\):

\[
\tau_e = \frac{L}{R}
\]  

**Mechanical timeconstant**, the first-order switching on phenomenon (transient) on a free turning motor for which the (self)induction is neglected:

\[
\tau_m = \frac{J_{motor} R}{k_m^2}
\]

3.4 Control parameters

To set the control parameters we need information about the unmodeled dynamics. In this case we have at our disposal a simplified/reduced (=design) model and an evaluation model. We therefore try to find a mathematical relation for the unmodeled dynamics. Additionally we assume the payload \(m_l\) and inertia \(J\) not to be known exactly \((\Delta m_l = 0.5 \text{ [kg]}, \Delta J = 0.5 \text{ [kgm}^2]\) , so that besides the unstructured uncertainties (i.e. unmodeled dynamics) structured (parametric) uncertainties are introduced. We set:

\[
P_1 = \hat{P}_1 + \Delta P_1
\]

\[
P_3 = \hat{P}_3 + \Delta P_3
\]

\[
\Delta P_1 = \frac{J_{motor}}{i_1^2} + 0.5 \quad \text{[kg]}
\]

\[
\Delta P_3 = \frac{J_{motor}}{i_2^2} + 0.5 \quad \text{[kgm}^2]\]
Note that the inertias of the unmodeled motors are 'compensated' in the structured uncertainties (recall that the inertia of the transmission is neglected).

We can rewrite the evaluation model (eq. 3.6) to:

\[
\begin{align*}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= (x_1 - \frac{P_2}{P_1})x_4^2 + \frac{1}{P_1}U + \frac{k_{m_1}}{i_1^2 R_1 P_1} x_2 - \frac{L_1}{R_1 P_1} \dot{\hat{p}} \\
\dot{x}_3 &= x_4 \\
\dot{x}_4 &= \frac{1}{P_3 - 2P_2 x_1 + P_1 x_1^2} \left( 2(P_2 - P_1 x_1)x_2 x_4 + U - \frac{k_{m_2}}{i_2^2 R_2} x_4 - \frac{L_2}{R_2} T \right) 
\end{align*}
\]

(3.10)

Notice that the state vector is reduced to four elements (the same as used in the design model (eq. 3.1)), referring to eq. 2.4.

The design model is given by a modification of equation 3.3 (by using estimations of the parameters instead of the real (but unknown) parameters):

\[
\begin{align*}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= (x_1 - \frac{P_2}{\hat{P}_1})x_4^2 + \frac{F}{\hat{P}_1} \\
\dot{x}_3 &= x_4 \\
\dot{x}_4 &= \frac{2(P_2 - \hat{P}_1 x_1)x_2 x_4 + T}{\hat{P}_3 - 2P_2 x_1 + \hat{P}_1 x_1^2}
\end{align*}
\]

(3.11)

Simple calculation yields the equations for the model errors (notice that they obey the matching condition (Appendix A; eq. A.5)): 
\[
\Delta f_1 + \Delta B_1 u = 0 \\
\Delta f_2 + \Delta B_2 u = -\frac{k_{m_2}^2}{i_1^2 R_1 P_1} x_2 - \frac{L_1}{R_1 P_1} \dot{T} + \frac{P_2 \Delta P_1 x_4}{P_1^2} - \frac{\Delta P_1}{P_1^2} U_r \\
\Delta f_3 + \Delta B_3 u = 0 \\
\Delta f_4 + \Delta B_4 u = -\frac{1}{P_3 - 2 P_2 x_1 + P_1 x_1^2} \left( \frac{L_2}{R_2} \dot{T} + \frac{k_{m_2}^2}{i_2^2 R_2} x_4 - 2 \Delta P_1 x_1 x_2 x_4 \right) \\
-\frac{\Delta P_1 x_1^2 + \Delta P_3}{(P_3 - 2 P_2 x_1 + P_1 x_1^2)^2} (2 (P_2 - P_1 x_1) x_2 x_4 + U_\phi + 2 \Delta P_1 x_1 x_2 x_4)
\]

(3.12)

With this equations we can find approximations for \(\alpha\) and \(\beta\) according to equation 2.8 for the three selected motors. It is quite obvious that the uncertainties are time varying and therefore it is very difficult to find the exact values of \(\alpha\) and \(\beta\). Furthermore the force/torque (and its derivative) is not known beforehand. Observing various simulation results however can give insight in the maximal values, so that a rough approximation for the gain \(k\) according to equation 2.20 will be:

- Electro Craft: \(k > 2\) \([\text{m/s}^2]\) (r), \([\text{rad/s}^2]\) (\(\varphi\))
- Honeywell: \(k > 2\) \([\text{m/s}^2]\) (r), \([\text{rad/s}^2]\) (\(\varphi\))
- Mavilor: \(k > 50\) \([\text{m/s}^2]\) (r), \([\text{rad/s}^2]\) (\(\varphi\))

We set the gain \(k\) at 4 and expect good results for the Electro Craft and Honeywell motor and no guaranteed tracking stability for the Mavilor motor. Afterwards we can calculate the model errors (eq. 3.12) and check whether the inequality eq. 2.20 is fulfilled.

\(\Phi\) is set to 0.05 in order to avoid undesirable chattering. \(Z_\theta\) is set to 0.2, to keep the tracking error small (eq. 2.30). \(\omega_\varphi\) is set to 10 [rad/s], so that the real break-frequency (bandwidth) will be 1.62 [rad/s] (eq. 2.29). The bandwidth of the error dynamics will be set approximately the same, so \(c_\varphi = c_\varphi = 4\) (bandwidth 2 [rad/s]). According to equation 2.30 the maximal tracking error will be less than 0.02 [m] (r) or 0.02 [rad] (\(\varphi\)).
3.5 Simulations

The simulations are performed with a TURBO-PASCAL program, with a discrete controller implementation. The calculation time is assumed to be very small in relation to the sampling time, so no Kalman observer is needed to estimate the next state. Measurement and control take place at the same time! Schematic the simulation algorithm is given in appendix E. The used parameters and controller settings are summarized in appendix I.

3.5.1 First setting

Simulation results with the three motors of appendix D and the controller settings of paragraph 3.4 are shown in figure 3.3. In the upper left corner of this figure simulation results with the Electro Craft (solid), Honeywell (solid) and Mavilor (dashed) motor are shown. As expected there is no tracking stability for the Mavilor motor and therefore this motor will not be used during further investigation. The three other plots in fig 3.3 are results of simulations with the Electro-Craft motor (shown as a typical simulation result). Solid lines represent the $r$-direction and dashed lines represent the $\varphi$-direction.

![Figure 3.3; simulation result](image-url)
Notice the input signal is smooth (no chattering) and that the tracking error obeys eq. 2.30 (i.e. the tracking error is smaller than 0.02 [m] for the r-direction and smaller than 0.02 [rad] in \( \varphi \)-direction). Besides we see that \( \delta \) remains inside the boundary layer, set by \( \Phi \).

To investigate robustness, simulations were done. Each time the same desired trajectory had to be followed (two revolutions) and the same controller setting was used. By changing the motor timeconstants \( \tau_e \) and \( \tau_m \) research was done to the influence of unmodeled dynamics on the tracking error Root Mean Square (RMS). Only the second revolution is used to calculate the RMS value, to eliminate the effect of incorrect initial conditions. As mentioned before, eq. 3.12 was calculated afterwards to check whether the condition described by eq. 2.20 was fulfilled.

\( \tau_e \) has been changed by choosing various (self)inductions \( L \), \( \tau_m \) has been changed by varying the motor constant \( k_m \) (both not influencing one another (see eq. 3.7 and 3.8)).

The results will be presented in double logarithmic plots with the unmodeled dynamics scaled along the x-axis and the error RMS scaled along the y-axis. This way of presenting the results appeared to be the most convenient to give a view of robustness.

Checking afterwards yielded that inequality 2.20 is fulfilled when:

\[
\begin{align*}
\text{Electro Craft:} & \quad \tau_e < 1.30e-2 \ [s] \\
& \quad \tau_m > 3.00e-3 \ [s] \\
\text{Honeywell:} & \quad \tau_e < 1.25e-2 \ [s] \\
& \quad \tau_m > 7.47e-4 \ [s]
\end{align*}
\]

Notice that the mechanical time constant \( \tau_m \) has a minimum value. Looking at figure 3.2 we can find the explanation for this fact, since small \( \tau_m \)-values imply high \( k_m \)-values (see eq. 3.8), yielding high (unmodeled) feedback gains.

The simulations with a Second Order Sliding Mode Controller are compared with a PD-controller (with acceleration feed-forward), whose setting is chosen such, that it yields the same tracking error RMS in case of the nominal motor (unchanged \( L \) and \( k_m \) (see appendix D)), yielding a critically damped controlled system. The simulation program for PD controller is almost the same as for SMC, only the input signal is different (see eq. 2.21):
\[ U = \hat{B}^{-1} \left[ X_d^{(s)} - \varepsilon_p - \dot{\varepsilon} \right] \]

\[ \varepsilon_p = C_t \varepsilon + C_0 \varepsilon \]

Therefore a good comparison can be made. The results are shown in figure 3.4.

**Figure 3.4:** robustness plots

From the definition of robustness, we concluded that robust controllers yield a smaller tracking error in the presence of model errors than non-robust controllers. When we consider variations of the electrical time constant (eq. 3.7) we see that the Sliding Mode Controller (for unmodeled dynamics obeying eq. 2.20) indeed is more robust than an equivalent PD controller (as expected), however when we consider variations of the mechanical time constant (eq. 3.8) the robustness of the SMC is approximately the same as an equivalent PD controller. Apparently a high unmodeled feedback gain is not compensated effectively. For unmodeled dynamics not obeying eq. 2.20 we see that the PD controller is more robust, but for these unmodeled dynamics the SMC controller has not been designed.
3.5.2 Error partition

The partition of the tracking error RMS into several classes has been investigated for the SMC controller and is shown in fig 3.5. Along the x-axis the unmodeled dynamics, caused by variation of the electrical time constant, is scaled. Two horizontal lines will appear, because the parameter errors are not affected by the electrical time constant.

<table>
<thead>
<tr>
<th>linestyle</th>
<th>unm. dyn.</th>
<th>par. err.</th>
</tr>
</thead>
<tbody>
<tr>
<td>solid</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>dashed</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td>dotted</td>
<td>no</td>
<td>yes</td>
</tr>
<tr>
<td>dashdot</td>
<td>no</td>
<td>no</td>
</tr>
</tbody>
</table>

![Figure 3.5; error partition](image)

From this figure we conclude that structured uncertainties (parameter errors) as well as unstructured uncertainties (unmodeled dynamics) have a considerable (negative) influence on the tracking error RMS. When both types of uncertainties are present, however, the tracking error RMS is not an addition of RMS-values obtained by treating both types of uncertainties separately. The controller compensates for structured and unstructured uncertainties at the same time.

3.5.3 PD-analogy

It is possible to choose the PD controller setting such that it yield a more robust controller than a SMC (first setting). The PD controller setting $c_I = 169$ and $c_D = 26$ (accidently chosen, wanting a critically damped controlled system with bandwidth 13 rad/s, by switching the parameters) yields the result as shown in figure 3.6 for variations of the electrical time constant of the Electro Craft motor.

We can clearly see a more robust PD controller in comparison to a SMC controller. Analysis yielded that, the more damping in the controlled system, the more robust the PD controller was.
Then the question arises if it is possible to tune the SMC controller such that the robustness is the same as in case of an equivalent PD controller. For a second order system this appears to be the case when we set:

\[
\begin{align*}
-z_0^2 + \frac{k}{\Phi} z_0 &= \omega^2 \\
c_{1_{(SMC)}} &= c_{1_{(PD)}} - \frac{\omega^2}{z_0} \\
c_{1_{(SMC)}} &= c_{0_{(PD)}} \frac{z_0}{\omega^2} \\
c_{0_{(SMC)}} &= 0
\end{align*}
\]

The last line implies that for the SMC controller the steady state property is lost (see pp. 7). When we consider an overdamped controlled system ($\beta > I$) it is always possible to set the control parameters of the second order sliding mode controller in such way that the system (inside the boundary layer) behaves exactly the same as controlled by a PD-controller. This has been confirmed by simulations. Setting the sliding mode controller as described by eq. 3.14 yield exactly the same tracking error RMS for electrical timeconstants below a certain value. Above this value for the SMC controller instability occurs (of course eq. 2.20 still holds), which is caused by inaccuracies in the calculation of the switching parameter $s$, because this is the only difference between the two controllers.
3.5.4 Second setting

To tune up the performance a new SMC parameter setting is chosen:

\[ k = 10 \text{ [-]} \]
\[ \omega = 15 \text{ [rad/s]} \]
\[ \Phi = 0.47 \text{ [s]} \]
\[ c_\theta = 225 \]
\[ c_I = 30 \]
\[ z_0 = 0.2 \]

The control bandwidth is 15 [rad/s] and the relative damping in the s-dynamics is approximately 0.71 [-]. Simulations with this controller (with Electro Craft motor) yield a smaller tracking error RMS in the nominal case. When we compare this controller with an equivalent PD controller (same error RMS in nominal case (bandwidth 35 rad/s)), the SMC is less robust to variations of the electrical time constant \( \tau_e \), but more robust to variations of the mechanical time constant \( \tau_m \) (when \( \tau_m > 2.1e-4 \)), although the difference is small. The results are shown in figure 3.7.

Strange enough, these results do not match with fig. 3.4. This is probably due to \( \Phi \), setting the damping in the s-dynamics at 0.71, while in fig. 3.4 the damping in the s-dynamics was 50, having a negative influence on the robustness to variation of \( \tau_e \), but a positive influence on the robustness to variation of \( \tau_m \).

![Figure 3.7; robustness plots](image-url)
### 3.5.5 Damping

In the preceding simulations the zero $z_0$ in the error dynamics has been set very small, to achieve a small tracking error (see eq. 2.30). From chapter 2.3 however we conclude that this choice of $z_0$ does not provide any significant damping in the error dynamics. Yet this damping is very important, especially during the transient. In fig 3.8 several transients are shown with different settings of $z_0$. We see that large values of $z_0$ indeed provide more damping. Besides the correctness of eq. 2.28 can be checked and appears to be OK.

![Damping Diagram](image)

$z = 0.2$ resp. $2$ resp. $20$

**Figure 3.8:** damping during transient

### 3.6 Conclusions

We have seen that a Second Order Sliding Mode Controller, designed according to eq. 2.20, is stable and more or less robust in the presence of model errors obeying the matching conditions. The results in comparison to a PD controller however are not substantially better.

The controlled system bandwidth in case of a SMC controller can be chosen lower than a PD controller to achieve the same tracking error. This is due to the fact that the SMC control contains more corrector-terms than a PD controller.
We concluded from the partition of the tracking error that structural (parameter) uncertainties have a strong influence on the total error. It is probably possible to lessen this effect by combining this controller with an adaptive controller (during this investigation this will however not be done).

Large values of the zero $z$ in the error dynamics provide more damping during the transient, but have negative influence on the tracking error. We could try to modify $z$ on line from high values during transient, to low values afterwards (notice that stability is not guaranteed during modification by the choice of the Lyapunov function described by eq. 2.14).

The robustness to parameter variations has not been investigated thoroughly, only an impression has been given in fig. 3.5. A more profound investigation will be done in the next chapter.
4. **Control of XY-table**

To investigate the second order sliding mode controller in an experimental environment, the control law was implemented in the software of the XY-table.

4.1 **Description XY-table**

In figure 4.1 a schematic top view representation of the XY-table is given, which one can imagine as a big plotter type machine. The end effector is a slide with mass \( m_e \), which can move in the XY-plane by means of three slideways. Two of them slides in x-direction and one in y-direction. The belt wheels of both slideways are driven by servomotors, exerting torques \( T_1 \) and \( T_2 \). Coulomb friction appears in all slides and is represented by friction-torques \( W_1, W_2 \) and \( W_3 \) [Nm]. Viscous damping is represented by \( D_1, D_2 \) and \( D_3 \) [Nm].

Unfortunately it is not possible to measure the position of the end effector, since only three measurement signals are available: \( x, x' \) and \( y \).

We therefore have to restrict ourselves to one of the following options to control the system:
- control of motor positions.
- control of estimated end-effector position:

\[ x_{\text{est}} = x + \frac{x'' - x}{d} y \]

\[ y_{\text{est}} = y \] (4.1)

The belt wheels of the slideways in x-directions can be connected in two ways:

(i) With a rigid bar \((k_z = \infty)\), resulting in a (stiff) model with two degrees of freedom: \(x, y\), since the translations \(x\) and \(x''\) are equal the estimated end-effector-position is the same as the motor position).

(ii) With a torsion spring with stiffness \(k_z\), resulting in a (flexible) model with three degrees of freedom: \(x, x''\) and \(y\).

### 4.1.1 Stiff model

The equation of motion in case of the two degrees of freedom (stiff) model can easily be derived and are represented in state space description by:

\[
\begin{align*}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= \frac{F_1 - a_3 \text{sign}(x_2) - a_2 x_2}{a_1} \\
\dot{x}_3 &= x_4 \\
\dot{x}_4 &= \frac{F_2 - a_4 \text{sign}(x_4) - a_6 x_4}{a_2}
\end{align*}
\]

\( \mathbf{x} = [x \ x' \ y \ y']^T \)

with:

\[
\begin{align*}
F_1 &= T_1 / r_x & [N] \\
F_2 &= T_2 / r_y & [N] \\
a_1 &= J_1 / r_x^2 + 2*m_x + m_e + m_y & [kg] \\
a_2 &= J_2 / r_y^2 + m_e & [kg] \\
a_3 &= (W_1 + W_2) / r_x & [N] \\
a_4 &= W_2 / r_y & [N] \\
a_5 &= (D_1 + D_3) / r_x & [Ns/m] \\
a_6 &= D_2 / r_y & [Ns/m]
\end{align*}
\]
Unfortunately the parameters $a_1, \ldots, a_6$ are not known exactly, so a few experiments have been done to identify them globally. At first a stable SMC-controller (setting of no importance) was implemented on the XY-table. Then the end-effector had to track a straight line with several constant velocities. As we can see from eq. 4.2 it is in that case very easy to calculate the friction and damping parameters by averaging out the input signals, while knowing the velocity. In fig. 4.2 the exerted force is given as function of the velocity.

![Figure 4.2; exerted force for constant velocity](image)

From this picture it is clear that we have to introduce viscous damping in the model to approach reality as closely as possible. In previous investigations however this phenomenon has not been accounted for.

Secondly the mass-parameters have been approximated by matching the (im)pulse-response of the uncontrolled model with the (im)pulse-response of the uncontrolled system***. From these experiments (shown in fig 4.3) the following parameters are found:

$$
\begin{align*}
  a_1 &= 34 \text{ [kg]} \\
  a_2 &= 2.7 \text{ [kg]} \\
  a_3 &= 36 \text{ [N]} \\
  a_4 &= 9 \text{ [N]} \\
  a_5 &= 50 \text{ [Ns/m]} \\
  a_6 &= 8 \text{ [Ns/m]} 
\end{align*}
$$

It should be noticed that these parameter estimates do approximately match with the ones in a recent investigation by Van de Wal [1992].

*** Instead of applying a real impulse to the system (impossible of course), a pulse is applied instead (lasting 0.03 seconds).
4.1.2 Flexible model

The equations of motion in case of the three degrees of freedom model (with torsion spring $k_t$) have been derived by Van den Molengraft [1989] and are much more complex. This model can be found in appendix F.

4.2 Desired trajectories

The desired trajectory to be tracked by the end-effector during all experiments is chosen to be a circle:

\[
\begin{align*}
x_d &= 0.5 - r \cos (\omega t) \\
y_d &= 0.5 + r \sin (\omega t)
\end{align*}
\]

with:

\[
\begin{align*}
r &= 0.25 \\
\omega &= \pi
\end{align*}
\]

The initial conditions will be discussed in paragraph 4.5.3.
4.3 **Unmodeled dynamics**

The dynamics of the XY-table is much more complex than described by the two models mentioned before. Some phenomena that have an additional influence on the dynamic response of the XY-table are:

- Harmonic friction terms in x- and y-direction, caused by some bad bearings and crooked shafts.
- Backlash in the attachment of one of the belt-wheels in x-direction
- Dynamics of motors and amplifiers
- Extra flexibility caused by springs (with stiffness $k_2$) used to attach the slides to the belts.
- Limited sampling rate in controller implementation
- Saturation of control input

As mentioned before, sliding mode control can be used when the structure of the model is inaccurate, or if the model parameters are unknown. The controller setting is, among other things, determined by the uncertainty of the additional dynamics and must be estimated. We therefore need information about the system.

4.3.1 **System analysis**

One way of gathering some information is to compare the impulse-response of the system with the impulse-response of the model. In figure 4.3 the impulse-response of the stiff model of the xy-table was already given as well as the impulse-response of the real system. In appendix G the transfer functions (impulse-response in frequency domain) are given. We can clearly see 'unmodeled dynamics' with an eigenfrequency of approx. 14 Hz in the x-direction. In the y-direction the 'unmodeled dynamics' is less clear. As mentioned in paragraph 2.3 we choose the controlled system bandwidth smaller than the lowest unmodeled structural mode, so it is probably this eigenfrequency that limits the controlled system bandwidth, as mentioned and investigated by Visser [1992]. In the following we will focus on the dynamics in the x-direction, because of the relatively low unmodeled eigenfrequency.
To assess robustness to unmodeled dynamics in an experimental environment we have several torsion springs $k_l$ at our disposal. However, experiments showed that the additional eigenfrequency, caused by these springs is not dominant, though this should be the case (theoretically) for springs with low $k_l$-values. For all $k_l$-values in our experiments we only recognize an unmodeled eigenfrequency of approx. 14 Hz. as in appendix G.

4.3.2 Model extension

To get a clear view of robustness, we try to model the dominant unmodeled eigenfrequency. Analyzing the system yielded that this eigenfrequency is probably caused by the springs used to attach the slides to the belts in $x$-direction. In appendix G we see also the impulse response of the system with removed springs, confirming the hypothesis.

There are two ways of modeling this effect. One way yields the following state space description, in normal form, according to eq. 2.9 (Notice that we make the simplifying assumption that the $x$-slides move simultaneously; $k_l = \infty$):

$$
\dot{\xi}_1 = \xi_2
$$

$$
\dot{\xi}_2 = \frac{1}{a_{1,1}} (F_1 - 4k_2 (\xi_1 - \eta_3) - a_{3,1} \text{sign(}\xi_2) - a_{5,1} \xi_2)
$$

$$
\dot{\xi}_3 = \xi_4
$$

$$
\dot{\xi}_4 = \frac{1}{a_2} (F_2 - a_4 \text{sign(}\xi_4) - a_6 \xi_4)
$$

$$
\dot{\eta}_5 = \eta_6
$$

$$
\dot{\eta}_6 = \frac{1}{a_{1,2}} (- a_{3,2} \text{sign(}\eta_6) - 4k_2 (\eta_5 - \xi_1) - a_{5,2} \eta_6)
$$

$$
\vec{\xi} = [x \ \dot{x} \ y \ \dot{y}]^T
$$

$$
\vec{\eta} = [x' \ \dot{x'} \ y' \ \dot{y'}]^T
$$
with:

$F_1 = T_1 / r_x \quad [N]$

$F_2 = T_2 / r_y \quad [N]$

$a_{1,1} = J_1 / r_x \quad [kg]$

$a_{1,2} = 2m_y + m_y + m \quad [kg]$

$a_2 = J_3 / r_y^2 + m_y \quad [kg]$

$a_{3,1} = c_1 * (W_1 + W_2) / r_x \quad [N]$

$a_{3,2} = (I - c_1) * (W_1 + W_2) / r_x \quad [N]$

$a_4 = W_3 / r_y \quad [N]$

$a_{5,1} = c_2 * (D_1 + D_2) / r_x \quad [Ns/m]$

$a_{5,2} = (I - c_2) * (D_1 + D_2) / r_x \quad [Ns/m]$

$a_5 = D_2 / r_y \quad [Ns/m]$

$k_2 = k_2 \quad [N/m]$

$0 < c_1, c_2 < 1 \quad [-]$

$a_{1,1} + a_{1,2} = a_1 \quad [kg]$

$a_{3,1} + a_{3,2} = a_3 \quad [N]$

$a_{5,1} + a_{5,2} = a_5 \quad [Ns/m]$

The parameters $c_1$ and $c_2$ are introduced to divide friction and damping between $x$ and $x'$.

We can show that the zero dynamics is exponentially stable by choosing the Lyapunov function:

$$V = \frac{1}{2} \eta^2 + 2k_2 \frac{\eta^2}{a_{1,2}}$$  \quad (4.4)$$

Another way of modeling would be to choose the position of the end-effector ($x'$) to be the output of the system. This yields a normal form without internal dynamics, since the relative degree in $x$-direction is '4' in that case. This description is not chosen because we cannot measure the end-effector position directly, which would make the control problem unnecessarily complex.

The parameter value of the friction-term $a_{3,2}$ is estimated by decoupling the belts from the pulleys and measuring the exerted force, necessary to move with a constant velocity (manually). The mass-parameters $a_{1,1}, a_{1,2}$ are set according to Van de Wal [1992] and the spring constant $k_2$ is set in such way, that the eigenfrequency is approx. 14 Hz. (Measurement of the spring constant yields approximately the same result). The damping is 'equally divided' between $a_{5,1}$ and $a_{5,2}$. This yields:
We see from these parameter-estimations that the (Coulomb) friction in the slides in x-direction \( a_{3,2} \) is relatively small in comparison to the total friction in x-direction \( a_{3,1} + a_{3,2} \) (viscous damping is the same). When we attach to friction a 'damping' action, this could be an explanation for the fact that the eigenfrequency due to spring \( k_2 \) can be seen clearly in the impulse-response (lightly damped), while the eigenfrequency due to spring \( k_1 \) is not visible (heavily damped). This hypothesis is supported by the fact that the friction in the bearings of \( x \) and \( x'' \) is quite high.

As mentioned before, it is not possible to measure the position of the end-effector, so we have to estimate this position with for instance a Kalman observer. We can show that the system is observable by measuring \( x \) and \( y \), applying (discrete) linear control theory as in Kok [1990]. An observer that estimates the position and speed one step ahead (necessary due to a discrete controller implementation) is given in appendix H.

The first result that will be presented is a comparison between the two described models eq. 4.2 and 4.3. In fig. 4.4 for both models (implemented in a PD controller with acceleration feed forward and friction compensation) the tracking error RMS obtained with an experiment is plotted against the controlled system bandwidth.

We see that the "extended model" (eq. 4.3) is more accurate, because it yields a smaller tracking error RMS and above all that this model remains stable when the controlled system bandwidth is chosen in the neighbourhood of 14 Hz. We therefore conclude that we have succeeded in our attempt to model the lowest structural resonant mode, in order to get a clear view of robustness during the experiments.

\[
\begin{align*}
a_{1,1} &= 17.5 \text{ [kg]} \\
a_{1,2} &= 16.5 \text{ [kg]} \\
a_2 &= 2.7 \text{ [kg]} \\
a_{3,1} &= 28 \text{ [N]} \\
a_{3,2} &= 8 \text{ [N]} \\
a_4 &= 9 \text{ [N]} \\
a_{5,1} &= 25 \text{ [Ns/m]} \\
a_{5,2} &= 25 \text{ [Ns/m]} \\
a_6 &= 8 \text{ [Ns/m]} \\
k_2 &= 23300 \text{ [N/m]}
\end{align*}
\]
Control parameters

Five different control problems will be considered, using:

1) Second order Sliding Mode Controller
2) Second order Sliding Mode Controller with on-line modification of the zero \( z_e \) in the error-dynamics
3) Second order Sliding Mode Controller with on-line modification of the zero \( z_e \) in the error-dynamics, controlling the estimated end-effector-position.
4) Second order Sliding Mode Controller, using an extended model in the controller implementation (see paragraph 4.3.2).
5) PD-controller with acceleration feed-forward and friction-compensation

Again the PD-controller has been investigated to allow comparison. A comparison with a first order SMC controller has not been performed. For all the experiments a sampling time of 0.005 seconds is chosen.

4.4.1 SMC-controller

The setting of the SMC-controller-setting is tuned up, to get the best results, i.e. the controlled system bandwidth is chosen maximal, with a relative damping in the error dynamics of \( \beta = 0.71 \). As we can see from eq. 2.21 the second order sliding mode control consists of an PD-part and a Sliding Mode-part. It is because of this PD-part that we can speak of an 'bandwidth'. In fig. 4.5 the tracking error RMS is plotted against the controlled system bandwidth (critically damped s-dynamics inside boundary layer). For high bandwidth's chattering occurs and the system becomes unstable.
For the stiff model (2 dof) probably the unmodeled dynamics due to $k_1$ is exerted, while for the extended (3 dof) model other unmodeled dynamics is the cause of this behaviour.

![Graph showing tracking error RMS vs bandwidth for SMC control. Solid line: 3 dof model (eq. 4.3). Dashed line: 2 dof model (eq. 4.2).](image)

**Figure 4.5; Sliding mode control**

To achieve damping during the transient, the zero in the error-dynamics is chosen twice the controlled system bandwidth. This yields for the stiff (2 dof) model:

\[
\begin{align*}
c_t &= 63.6 \\
c_\theta &= 2025 \\
\omega_n &= 45 \\
z_\theta &= 90
\end{align*}
\]

and for the extended (3 dof) model ($k_2$):

\[
\begin{align*}
c_t &= 99 \\
c_\theta &= 4900 \\
\omega_n &= 70 \\
z_\theta &= 140
\end{align*}
\]

We have to make assumptions on the maximal model uncertainties to determine $k$ and therefore consider the three models available:

- nominal (stiff) model (eq. 4.2)
- extended (3 dof) model with torsion spring $k_1$ (appendix F)
- extended (3 dof) model with springs $k_2$ (eq. 4.3)

It is not difficult to derive the model with $k_1$ as well as $k_2$ (five degrees of freedom), which will of course yield a better estimation of the uncertainties. It is however not sensible to derive complicated evaluation models, only to estimate uncertainties in the design model, because when we did, we could use this evaluation model as a new design model, probably again yielding a better estimation, and so on.
We therefore try to find estimations of the uncertainties assuming the 'three degrees of freedom models' to be an exact representation of the real system (the evaluation model). This method yields a rough approximation for the uncertainties in case of the stiff design model (eq. 4.2):

Evaluation model \( k_1 \) (appendix F):

\[
\alpha = \left| \frac{(m_{13} + m_{33})k_{11}(x'' - x) + m_{13}W_x\text{sign}(\dot{x}'')} + m_{33}W_x\text{sign}(\bar{x})}{m_{11}m_{33} - m_{13}^2} + \frac{a_3}{a_1}\text{sign}(\bar{x}) \right|
\]

\[
\beta = \left| \frac{m_{33}}{m_{11}m_{33} - m_{13}^2} - \frac{1}{a_1} \right|
\]

(4.5)

where we assumed that the centrifugal and Coriolis forces are small enough to be neglected. The same is assumed for the cross-terms (between \( x \) and \( y \)) in the mass matrix.

Evaluation model \( k_2 \) (eq. 4.3):

\[
\alpha = \left| \frac{a_3}{a_1}\text{sign}(\bar{x}) - \left( \frac{a_{1,2}}{a_{1,1}}\dot{x}' + \frac{a_{3,2}}{a_{1,1}}\text{sign}(\dot{x}') + \frac{a_{3,1}}{a_{1,1}}\text{sign}(x) \right) \right|
\]

(4.6)

Once we have these expressions we can make an estimation of the gain \( k \) (according to eq. 2.20), in case we have a stiff design model and an 3 dof evaluation model. Therefore we have to make an estimation for the \( \hat{B}^T\hat{u}_{\text{max}} \) value. Strictly this value cannot be calculated beforehand, but only an approximation (see eq. 2.17). This yields:

<table>
<thead>
<tr>
<th>evaluation model</th>
<th>( \alpha )</th>
<th>( \beta )</th>
<th>( \hat{B}^T\hat{u}_{\text{max}} )</th>
<th>( k )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model ( k_1 ) (app. F)</td>
<td>( \approx 4 )</td>
<td>( \approx 0.015 )</td>
<td>( \approx 550 )</td>
<td>( \approx 19.0 )</td>
</tr>
<tr>
<td>Model ( k_2 ) (eq. 4.3)</td>
<td>( \approx 2 )</td>
<td>( \approx 0.030 )</td>
<td>( \approx 550 )</td>
<td>( \approx 39.0 )</td>
</tr>
</tbody>
</table>
Notice that we can conclude from the estimations of the gain $k$ that the evaluation model with $k_1$ is closer to the stiff design model than the evaluation model with $k_2$. This is as to be expected, since we concluded before that the unmodeled dynamics due to springs $k_2$ is dominant. The values in this table are estimated for one trajectory only (see paragraph 4.2) and it should be noticed that all these values, but particularly $\hat{B}^{-1}\hat{u}_{\text{max}}$ are strongly dependent on this trajectory. From this table we can derive that the gain $k$ in case of the stiff design model (eq. 4.2) should be higher than 39. To have a kind of safety margin we choose the gain $k = 70$.

To prevent undesired chattering we apply the saturation function and set $\Phi$ in such way, that we get a critically damped second order s-dynamics inside the boundary layer (see eq. 2.24).

The control law is given by eq. 2.21.

When we use the extended (3 dof) model (eq. 4.3) in the controller (i.e. design model) we can expect the uncertainties to be smaller than in case of the stiff (2 dof) model (eq. 4.2), since our model is more accurate. It is however not possible to calculate the $k$-value as in the previous case, since we don't have the exact mathematical description of the system or an even more extended (> 3 dof) evaluation model. To be on the safe side we choose our $k$- and $\Phi$-setting the same as was done in case of the stiff design model, described above.

### 4.4.2 PD-controller

The setting of the PD-controller is chosen such that the relative damping $\beta = 0.71$ and the controlled system bandwidth is as high as possible. This bandwidth has already been discussed before and the results are plotted in fig 4.4. We therefore conclude that we can set the controlled system bandwidth in case of the stiff design model to 45 rad/s and the extended design model to (at least) 70 rad/s.

Notice that the controlled system bandwidth can be chosen higher than in case of a Sliding Mode Controller. This is caused by the fact that the SMC control action is 'wilder', containing more corrector terms, causing chattering for the chosen setting.

The control law is given by eq. 3.13.
4.5 Experiments

To assess robustness of the Second Order Sliding Mode Controller experiments are done with:
- several springs $k_I$
- additional masses connected to the end-effector

Two models are used in the controller implementation:
- stiff model (eq. 4.2)
- extended model (eq. 4.3)

Two 'types' of Second Order Sliding Mode Controllers are considered:
- as described in previous paragraph.
- as described in previous paragraph with on-line modification of the zero $z_0$ in the error-dynamics. After 0.5 second (after transient) the zero $z_0$ is reduced to half the initial value in approximately 1 second. Of course a more advanced $z_0$-modification-algorithm is possible.

These controllers are compared with a PD-controller with acceleration feed-forward (2 models) and friction-compensation. Besides experiments were done, controlling the estimated end-effector-position (eq 4.1) with a Second Order Sliding Mode Controller with on-line modification of the zero $z_0$ in the error-dynamics.

During all the experiments the end-effector had to track the same (desired) trajectory twice. Only the second revolution is used to calculate the tracking error RMS, to eliminate the effect of incorrect initial conditions.

The used parameters and controller settings are summarized in appendix I.

4.5.1 Robustness to unmodeled dynamics

To investigate robustness to unmodeled dynamics we did experiments with different torsion springs $k_I$. The stiffness of the torsion springs $k_I$ can be computed with a standard formula. However, this formula is only valid for small torsion angles (assuming a linear relation between exerted torque and angular displacement), while the real torsion angles are far beyond these values. In a recent investigation by Van de Wal [1992] the stiffnesses have therefore been identified during experiments and because of that they seem to be more trustworthy. For not identified spring constants, the calculated values are chosen.
Notice that the stiffness of torsion spring '6' is higher identified than torsion spring '5' and that for large spring-constants the difference between calculation and identification is eminent. The last effect is due to additional flexibilities that occur in practice, for instance in the driving shaft and toothed belts.

In fig 4.6 the results in x-direction are presented for both models. To have a good look at robustness in the same figure the results are normalized, i.e. the tracking error RMS in case of the most stiff torsion spring $k_I$ is set to 1 (100%). From these plots we see that the robustness to stiffness $k_I$ of both SMC-controllers (with and without modification of $z_F$) is almost the same, as to be expected, but we see also that the PD-controller yields better results (not expected), though the tracking error RMS is higher. The results in y-direction are not shown because for all controllers the tracking error RMS in this direction is hardly affected by the torsion springs $k_I$.

A strange phenomenon is the fact that the weakest spring $k_I$ yields a small tracking error RMS. This phenomenon can be seen in all experiments and is probably due to the fact that the torsion angles are that high, that the 'effective mass' to be displaced (at $x$) decreases. This can be illustrated by an (imaginary) experiment for which no spring $k_I$ (i.e. $k_I = 0$) is used. In that case the displacement of $x''$ is zero and the mass at $x$ is 'reduced'.

<table>
<thead>
<tr>
<th>spring no.</th>
<th>calculated $k_I$ [Nm/rad]</th>
<th>identified $k_I$ [Nm/rad]</th>
<th>used $k_I$ [Nm/rad]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>213</td>
<td>-</td>
<td>213</td>
</tr>
<tr>
<td>2</td>
<td>28</td>
<td>14</td>
<td>14</td>
</tr>
<tr>
<td>3</td>
<td>3.77</td>
<td>3.16</td>
<td>3.16</td>
</tr>
<tr>
<td>4</td>
<td>0.92</td>
<td>1.28</td>
<td>1.28</td>
</tr>
<tr>
<td>5</td>
<td>0.79</td>
<td>0.73</td>
<td>0.73</td>
</tr>
<tr>
<td>6</td>
<td>0.69</td>
<td>0.86</td>
<td>0.86</td>
</tr>
<tr>
<td>7</td>
<td>0.45</td>
<td>0.50</td>
<td>0.50</td>
</tr>
<tr>
<td>8</td>
<td>0.19</td>
<td>-</td>
<td>0.19</td>
</tr>
</tbody>
</table>
4.5.2 Robustness to parameter variations

To investigate robustness to parameter variations, experiments are done with the most stiff $k_i$ and different additional masses attached to the end-effector. The results in x-direction are shown in fig. 4.7. In this case we see again that the robustness of both SMC-controllers is approximately the same. However, now the robustness (to mass-variation) of the SMC-controller is better. In y-direction the results are also shown in fig. 4.7, for which we remark that the bandwidth had to be set lower (to 30 rad/s) to prevent instability (caused by relatively large parameter errors). Again we see the same trend. We remark that for the y-direction the stiff and extended model are the same (compare eq. 4.3 and eq. 4.2), since the springs used to attach the belt to the slide do not play a dominant role in y-direction.

Figure 4.6; robustness to unmodeled dynamics
Control of XY-table

4.5.3 Initial conditions

A remark that must be made is that the SMC-controller is very 'sensitive' to wrongly set initial conditions (in the controller). When we set the initial condition as described in paragraph 4.2, the controlled system is unstable for the chosen bandwidth, in contradiction to the PD-controllers. This is due to two phenomenon:

Figure 4.7; robustness to parameter variations
At first as we can see from eq. 2.20 the gain \( k \) is set with supposed knowledge of the synthetic input \( \hat{u} \). This input in case of wrong initial conditions can differ greatly from the nominal case. It is possible that the minimum gain \( k \) according to eq. 2.20 is therefore chosen to low and the controlled system becomes unstable.

Secondly saturation of control is cause of much trouble. Due to excess of correction-terms in the control law over a PD-controller, the SMC-controller makes very strong control actions. Result is a quick saturation of the control input, also yielding an unstable controlled system.

The problems mentioned above are prevented by choosing a slightly different trajectory, for which the initial velocity is \( \theta \) and is as in paragraph 4.2 after 0.5 seconds, and by choosing the initial condition as good as possible:

\[
\omega = 2\omega_d t - \frac{1}{2\pi} \omega_d \sin(4\pi t) \quad 0 < t < 0.5 \quad [s]
\]

When we track this modified desired trajectory, the control does not saturate, as expected, and therefore eliminates the effect of wrongly set initial conditions. All the experiments to assess robustness tracked this trajectory during the transient.

4.5.4 Control of estimated end-effector position

A few experiments were done with control of the estimated end-effector position. Using the stiff model (eq. 4.2) in the controller, with the same setting as used in the previous experiments. The results are shown in fig 4.8.

![Figure 4.8; sliding mode control of end-effector position](image)
We see that instability occurs, which is caused by the use of a too simple model (2 dof) in the Kalman filter. Probably better results can be obtained by using the 3 dof model of appendix F. The extended model of eq. 4.3 would not yield better results in this case, because it does not account for springs $k_I$. A combination of both models (as suggested earlier) would also be a good alternative.

### 4.6 Conclusions

As expected the trade-off between tracking accuracy and damping during transient can be solved, as suggested in paragraph 3.6, by modifying $z_0$ on line from high values during transient (damping), to low values afterwards (tracking accuracy).

The gain $k$ depends very much on the desired trajectory and initial condition. Wrongly set initial conditions cause instant instability.

Second order sliding mode control is very 'sensitive' to saturation of control, causing instability. The only remedy to account for this effect is to set the gain $k$ as low as possible (but still according to 2.20) and the bandwidth as low as possible, resulting in less control effort.

A PD-controller with an extended model yields better results, concerning robustness and performance than an advanced SMC-controller with a simple model. We therefore conclude that 'knowledge of the system to be controlled is better than knowledge of advanced control' (De Jager [1992]).

As far as robustness to unmodeled dynamics is concerned, the PD controller performs better, especially in case of a simple model. When we look at robustness to parameter errors, the results of a SMC-controller are slightly better.

The performance (i.e. tracking accuracy) of a Second Order Sliding Mode controller is often better than the performance of a PD controller with acceleration feed-forward and friction compensation.
5. Conclusions and Recommendations

In the previous chapters the robustness of a 'Second Order Sliding Mode Controller' has been investigated and compared with a traditional PD controller.

In chapter 2 we saw that theoretically stability can be proven in the presence of model errors, as long as they obey the matching conditions and are bounded with known bounds. The gain $k$ is a measure for these modeling errors, but is strongly dependent on the desired trajectory and initial condition.

In chapter 3 simulation results were presented for a 'Second Order Sliding Mode' and a PD controller in the presence of structured and unstructured model uncertainties. We did not see a substantial improvement in robustness of the sliding mode controller, but instead the bandwidth of the 'Sliding Mode Controller' could be set lower than of a PD controller, realizing the same tracking error.

In chapter 4 experimental results were presented, while investigating robustness to structured and unstructured model uncertainties separately. Besides the problem of the trade-off between tracking accuracy and damping during transient has been solved.

In this chapter the conclusions and recommendations will be given for further research.

The final conclusions are:

* In the introduction we stated that a 'Second Order Sliding Mode Controller' should be robust to unstructured uncertainties (unmodeled dynamics). After simulations and experiments however, we cannot confirm this statement, since a traditional PD controller often yields better results. We therefore conclude that a 'Second Order Sliding Mode Controller' is not more robust to unstructured uncertainties than a PD controller.

* The 'Second Order Sliding Mode Controller' is more robust to structured uncertainties (parameter variations) than a PD controller. A comparison with an adaptive controller has not been performed.

* With a 'Second Order Sliding Mode Controller' a smaller tracking error can be realized than with a PD controller, having the same bandwidth, or the same tracking error, having a smaller bandwidth.
Large values of the zero in the error dynamics provide more damping during transient, but have negative influence on the tracking error. This trade-off has been eliminated by modifying the zero online, from high values during transient (damping) to low values afterwards (tracking error).

'Second Order Sliding Mode Control' is not robust to variations in the initial condition, since the gain $k$ depends (very much) on the desired trajectory, and therefore on the initial condition. This is a disadvantage. Besides 'Second Order Sliding Mode Control' is very sensitive to saturation of control, causing instant instability.

The performance of a 'Second Order Sliding Mode Controller' with relatively small model uncertainties is often better than the performance of a PD controller.

A traditional PD controller with an extended model yields better results, concerning both robustness and performance than an advanced 'Second Order Sliding Mode Controller' with a simple model. To design a 'Second Order Sliding Mode Controller' we need information about the system, to determine the uncertainty bounds. It is better to use this information to extend the model and apply a PD-controller: 'knowledge of the system to be controlled is better than knowledge of advanced control' (De Jager [1992]).

A disadvantage of the Second Order Sliding Mode Controller is the large number of parameters that must be tuned.

The recommendations for further research are:

For further investigation it is recommended to equip the XY-table with a end-effector measurement system, such that the end-effector can track a real trajectory and the influence of springs $k_1$ and $k_2$ can be seen more clearly.

It is of advantage to use the extended model of the XY-table (as described in chapter 4) in the controller implementation, or maybe an even more extended model by combining the two available 3 dof models into a 5 dof model.

To lessen the effect of parameter errors on the tracking error RMS it is probably possible to combine the 'Second Order Sliding Mode Controller' with an adaptive controller.

A comparison with a first and/or higher order sliding mode controllers could be performed.
6. References

Asada, H and J.-J. E Slotine,  

Breevoord, G.,  
'Composite computed torque control of the XY-table with an elastic motor transmission', Master's thesis, Eindhoven University of Technology, Department of Mechanical Engineering, WFW Report 92.054, 1992

Chang, L.-W.,  
'A MIMO Sliding Control with a Second-Order sliding condition', ASME WAM. paper no. 90 WA/DSC-5, Dallas, Texas, 1990

Chang, L.-W.,  
'A MIMO Sliding Control with a First-order plus Integral Sliding Condition', Automatica, Vol. 27, No. 5, pp. 853-858, 1991

Chern, T.-L. and Wu, Y.-C.,  

Cool, J.C., Schijff, F.J., Viersma, T.J.,  
'Regeltechniek', Delta Press BV, Overberg, The Netherlands, 1985

Elmali, H and Olgac, N.,  

Gerwen, L. J. W. van,  

Heikoop, R.  
'Robust control based on μ-synthesis, applications for dynamic systems', master's thesis, Eindhoven University of Technology, Department of Mechanical Engineering, WFW report 92.128, 1192
Isidori, A.,
'Nonlinear control systems, an introduction', Berlin: Springer-Verlag, 1985

Jager, B. de, Lammerts, I., Veldpaus, F.,
'Course on advanced control', Lecture notes: 4708, Eindhoven University of technology, Department of Mechanical Engineering, 1991

Jager, B. de,

Jager, B. de,

Jager, B. de,

Kok, J. J.,
'Werktuigkundige Regeltechniek II', Lecture notes: 4594, Eindhoven University of Technology, Department of Mechanical Engineering, 1990

Koster, M. P.,
'Het ontwerpen van elektromechanische servo-systemen', Lecture notes: 124151, Twente University, Department of Electrical Engineering, January 1991

Molengraft, M. J. G.,
'Identification of non-linear mechanical systems', Phd dissertation, Eindhoven University of Technology, Department of Mechanical Engineering, 1990

Mulders, P. C.,
'Geavanceerde besturingstechnologie', Lecture notes: 4603, Eindhoven University of Technology, Department of Mechanical Engineering, 1990
Nievergeld, A. J. L.
'Robotregeling met behulp van Sliding mode', master's theses, Eindhoven University of Technology, Department of Electrical Engineering, Report no. 5706, 1990

Slotine, J.-J. E. and Li, W.,

Spong, M. W. and Vidyasagar, M.,

Utkin, V. I.,

Vijverstra, F. J.,
'Direct, Indirect and Composite Adaptive Control of Robot Manipulators', master's thesis, Eindhoven University of Technology, Department of Mechanical Engineering, WFW report 92.076, 1992

Visser, R. W.,
'The improvement of controller robustness using acceleration feedback', master's thesis, Eindhoven University of Technology, Department of Mechanical Engineering, WFW report 92.093, 1992

Wal, M. van de,
'Identificatie van de XY-tafel ten behoeve van regeldoeleinden', Eindhoven University of Technology, Department of Mechanical Engineering, WFW report 92.110, 1992
Appendix A

I/O linearization

As described by Slotine and Li ([1991] pp. 225), a controller design based on I/O-linearization can be designed in three steps:

- differentiate the output \( y \) until the input \( u \) appears (determine (vector) relative degree and normal form by suitable transformation);
- choose \( u \) to cancel the nonlinearities and guarantee tracking convergence;
- study the stability of the internal dynamics.

Before we can perform these steps, we first have to introduce some mathematical tools.

Given a scalar function \( h(x) \) and a vector field \( f(x) \) we can define a new scalar function, called the Lie derivative of \( h \) with respect to \( f \):

\[
L_f h = \nabla h f = \frac{\partial h}{\partial x} f
\]  
(A.1)

Repeated Lie derivatives can be defined recursively:

\[
L_f^0 h = h
\]
(A.2)

\[
L_f^i h = L_f (L_f^{i-1} h) f = \nabla(L_f^{i-1} h) f
\]

Similarly, if \( g(x) \) is another vector field, then the scalar function \( L_g L_f h(x) \) is:

\[
L_g L_f h = \nabla(L_f h) g
\]  
(A.3)

To perform I/O linearization we consider a general nonlinear MIMO system (eq 2.1). The model of the system is said to have a (vector) relative degree \([r_p, \ldots, r_k]\) at \( x_o \) if:
(i) \[ L_y L_j^p h_i(x) = 0 \]

for all \( 1 \leq j \leq m \)
\[ I \leq i \leq k \]
\[ p \leq r_i - 2 \]

for all \( x \) in a neighbourhood of \( x_0 \)

(ii) the \( k \times m \) matrix (\( k \leq m \))

\[
\begin{bmatrix}
L_x L_j^{r_i-1} h_i(x) & \cdots & L_x L_j^{r_i-1} h_i(x) \\
\vdots & \ddots & \vdots \\
L_x L_j^{r_i-1} h_i(x) & \cdots & L_x L_j^{r_i-1} h_i(x)
\end{bmatrix}
\] (A.4)

has rank \( k \) at \( x = x_0 \)

Physically this means that \( r_i \) is exactly the number of times one has to differentiate the \( i \)-th output \( y_i(t) \) in order to have at least one component of the input vector \( u \) explicitly appearing.

One can show that the (vector) relative degree is unchanged by the addition of disturbances if the model errors are inside the kernel of the map defined by the vector fields (see Elmali and Olgac [1992]):

\[
\frac{\partial h_i}{\partial x}, \frac{\partial L_j h_i}{\partial x}, \ldots, \frac{\partial L_j^{r_i-2} h_i}{\partial x}
\] (A.5)

for \( i = 1 \ldots k. \)

This is called the 'matching condition'. It assures that the disturbances do not appear in derivatives of \( y_i \) of order less than \( r_i \).

If the above assumptions on the perturbation vector fields are fulfilled there is a diffeomorphic coordinate transformation (i.e. invertible and continuous partial derivatives of any order) \( (\xi, \eta) = \Gamma(x) \), which transforms the system (eq. 2.1) into a so-called 'normal form' (see Elmali and Olgac [1992]):
Where:

\[ \xi_1^i = h_i \left( T^{-1}(\xi, \eta) \right) \]
\[ \xi_2^i = L_f h_i \left( T^{-1}(\xi, \eta) \right) \]
\[ \vdots \]
\[ \xi_{r_i}^i = L_f^{r_i-1} h_i \left( T^{-1}(\xi, \eta) \right) \]

\[ \dot{\xi}_i \left( \xi, \eta \right) = L_f^r h_i \left( T^{-1}(\xi, \eta) \right) \]
\[ \Delta b_i \left( \xi, \eta \right) = L_f^{r+1} h_i \left( T^{-1}(\xi, \eta) \right) \]
\[ \Delta a_{ij} \left( \xi, \eta \right) = L_f^{r_i+1} h_i \left( T^{-1}(\xi, \eta) \right) \]

\[ i = 1 \ldots k, \quad j = 1 \ldots m \]

\[ r = r_1 + r_2 + \ldots + r_k \quad (r \leq n). \]

These equations are rewritten in matrix form, suppressing \((\xi, \eta)\):

\[ \xi^{(r)} = \dot{\xi} + \Delta \ddot{\xi} + (\dot{A} + \Delta A) u \]
\[ \eta = q \]  

(A.8)

Where \( \xi^{(r)} = [\xi^{(r)}_1, \xi^{(r)}_2, \ldots, \xi^{(r)}_k]^T \)

(superscripts in parenthesis indicate the order of time derivatives).
Note that the first $r$ equations of the normal form have a 'companion' or 'controllability canonical' form, while the last $(n-r)$ equations are not directly related to the system output $y$ and the last states remain 'unobservable'.

Suppose now the following state feedback control law is chosen:

$$ u = \hat{A}^{-1} (v - \hat{b}) $$  \hspace{1cm} (A.9)

where $v$ is the 'synthetic input' ($v \in \mathbb{R}^k$).

This control linearizes the nominal part of equation A.8, and yields:

$$ \xi^{(v)} = v + \Delta b + \Delta A \hat{A}^{-1} (v - \hat{b}) $$  \hspace{1cm} (A.10)

$$ \dot{\xi} = \eta $$

The overall strategy (in the unperturbed case) is depicted in figure A.1, in which $v$ is generated by a compensator.

---

*** If $A$ is not square ($k < m$) we cannot calculate $A^{-1}$. In that case we take the pseudoinverse of $A$ instead. The pseudoinverse of a matrix $A$ is a matrix $X$ of the same dimensions as $A^T$, so that $A^T X A = A$, $X^T A X = X$ and $X^T A$ and $A^T X$ are Hermitian (definition from software package MATLAB).
Appendix B

Sliding mode control with first order sliding condition

We consider the nonlinear SISO system in controllability canonical form, with \( x \) the output of interest:

\[
\begin{align*}
    x^{(n)} &= f(x) + B(x)u(t) \\
    \dot{x} &= [x, \dot{x}, \ldots, x^{(n-1)}]
\end{align*}
\] (B.1)

Sliding mode controller design starts with the definition of the error dynamics:

\[
\begin{align*}
    e^{(n-1)} + c_{n-1}e^{(n-2)} + \ldots + c_1e &= s \\
    e &= x - x_d
\end{align*}
\] (B.2)

When we take one time derivative we get:

\[
\begin{align*}
    \dot{e}^{(n)} + c_{n-1}\dot{e}^{(n-1)} + \ldots + c_1\dot{e} &= \dot{s} \\
    \dot{e}^{(n)} + \dot{e}_p &= \dot{s} \\
    \dot{e}_p &= c_{n-1}\dot{e}^{(n-1)} + \ldots + c_1\dot{e}
\end{align*}
\] (B.3)

Substitution of eq. B.1 in eq. B.3 yields:

\[
\dot{s} = Bu + f - x_d^{(n)} + e_p
\] (B.4)

The state feedback control law is proposed as:

\[
\begin{align*}
    u &= \dot{s}^{-1}(\dot{u} - k\text{sign}(s)) \\
    \dot{u} &= -f + x_d^{(n)} - e_p
\end{align*}
\] (B.5)
To investigate stability with the second method of Lyapunov, a positive definite Lyapunov function candidate $V$ is chosen:

$$V = \frac{1}{2}s^2$$  \hspace{1cm} (B.6)

Uniform asymptotic stability is guaranteed if the time-derivative of $V$ is negative definite:

$$\dot{V} = ss' < 0$$  \hspace{1cm} (B.7)

Therefore:

$$s[Bu + f - x_d^{(e)} + e_p] < 0$$

$$s[BB^{-1}[\dot{u} - k\text{sign}(s)] + f - x_d^{(e)} + e_p] < 0$$  \hspace{1cm} (B.8)

$$s[\Delta BB^{-1}a - BB^{-1}k\text{sign}(s) - \dot{f} + f] < 0$$

$$s[\Delta f + \Delta BB^{-1}a - k(1 + \Delta BB^{-1})\text{sign}(s)] < 0$$

Equation B.8 indicates a condition for the gain $k$ to guarantee the system stability. To quantify the gain we assume:

$$s[\Delta f + \Delta BB^{-1}a - k(1 + \Delta BB^{-1})\text{sign}(s)] \leq$$

$$|s| \cdot [|\Delta f| + |\Delta B|\cdot|B^{-1}a| - k(1 - |\Delta B|\cdot|B^{-1}\text{sign}(s)|)] < 0$$  \hspace{1cm} (B.9)

Finally the gain $k$ is found as:

$$k > \frac{\alpha + \beta |\hat{B}^{-1}a|}{1 - \beta |\hat{B}^{-1}|}$$  \hspace{1cm} (B.10)

if $\beta |\hat{B}^{-1}| < 1$
And $V$ indeed appears to be a true Lyapunov function. The Lyapunov stability is guaranteed as long as $k$ is chosen according to eq. B.10. Furthermore, we conclude from this equation that $k$ is a measure of the uncertainties. We see that $k$ depends on $\theta$ and therefore on the desired trajectory and the initial condition (through the tracking error $e_p$).

We can find the s-dynamics by substituting equation B.5 in B.4. Simplifying yields:

$$\dot{s} + kB\tilde{B}^{-1}\text{sign}(s) = \Delta B\tilde{B}^{-1}(x_d^{(m)} - \dot{\hat{f}} - e_p) + \Delta f \quad (B.11)$$

As we can see from eq B.5 the control law has to be discontinuous across $s$, which leads to chattering. In general, chattering must be eliminated for the controller to perform properly. This can be achieved by smoothing out the control discontinuity, in a boundary layer neighbouring the switching surface ($s = \theta$). Therefore we don't use the 'signum'-function (sign), but we apply the 'saturation'-function (sat) instead:

$$\text{sat}(s, \phi) = \begin{cases} 
1 & s > \phi \\
\frac{s}{\phi} & |s| < \phi \\
-1 & s < -\phi 
\end{cases} \quad (B.12)$$

When we do so, the s-dynamics inside the boundary layer become:

$$\dot{s} + \frac{k}{\phi}B\tilde{B}^{-1}s = \Delta B\tilde{B}^{-1}(x_d^{(m)} - \dot{\hat{f}} - e_p) + \Delta f \quad (B.13)$$

So the smoothing out of control discontinuity inside the boundary layer essentially assigns a low-pass filter structure to the s-dynamics of which the break-frequency is determined by the boundary layer width $\Phi$ and the gain $k$.

If we rewrite (factorize) the error-dynamics (equation B.2) we get without loss of generality ($\lambda_i$ can be complex!):
We see that the error dynamics can be thought of as a sequence of 1st order filters with input $s$.

Noting that after a transient $s<\Phi$ (guaranteed with Lyapunov’s second method) we can show that this operation leads to tracking to within a guaranteed precision $\epsilon$ (The derivation can among others be found in Asada and Slotine [1985]):

$$|\epsilon| < \frac{\Phi}{\lambda_1 \lambda_2 \ldots \lambda_{n-1}}$$

(B.15)

Figure B.1 shows schematically the sliding control from a synthesis and implementation point of view.

To filter high frequency unmodeled dynamics we have to set the bandwidth of the s-dynamics lower than the lowest structural resonant mode. Most authors (Slotine and Li [1991] (pp. 295) and Chang [1990]) suggest to set the bandwidth of the error-dynamics equal to the bandwidth of the s-dynamics ($\lambda=k/\Phi$).
When we do so, there is a trade-off between tracking accuracy and robustness to unmodeled dynamics. This is caused by the fact that the control parameter $\Phi$ is (over)loaded with two tasks which are controlling the accuracy and filtering the unwanted dynamics. A constant thin layer (small $\Phi$) may lead to a fast response and accurate tracking, but has no control over rejecting the unmodeled frequencies. With greater $\Phi$, the high-frequency unmodeled dynamics are filtered at the sacrifice of the tracking accuracy.

It is however not necessary to choose both bandwidth's the same. It is possible to set the bandwidth of the error-dynamics higher than the bandwidth of the s-dynamics, thus eliminating the trade-off between tracking accuracy and robustness to unmodeled dynamics (see eq. B.15). We only have to pay attention to the fact that when we set the bandwidth of the error-dynamics higher than the lowest structural resonant mode, the unmodeled dynamics (in $e$) are filtered only to the first order (see fig B.1) by the s-dynamics, resulting in a less effective filtering.

Summarizing we conclude:

(i) A variable $s$ is introduced, that is merely a weighted sum of the tracking error and it's derivatives. The weighting factors are chosen such that the error-dynamics is exponentially stable.

(ii) We guarantee (Lyapunov) that within a finite time interval it holds that $s < \Phi$, by a proper selection of the gain $k$.

(iii) Inside the boundary layer, $s$ can be thought of as a $1^{st}$ order filtering of among others the unknown disturbances, so if we want to filter high frequency unmodeled dynamics we have to set the 'bandwidth' low, implying $\Phi$ to be large.

(iv) Tracking accuracy is guaranteed to within a precision $\epsilon$, implying $\Phi$ to be small, given an available 'bandwidth', or large $\lambda$'s, resulting in a lower order filtering of the unmodeled dynamics.

(v) In case of equal bandwidth's (error- and s-dynamics) there exists a trade-off between tracking accuracy and robustness to unmodeled dynamics. In case of unequal bandwidth's the filtering of the high-frequency unmodeled dynamics is of lower order and therefore less effectively.

We see that the tuning capability of a First Order Sliding Mode Controller is limited.
Appendix C

RT-robot trajectory

Description of trajectory RT-robot in polar coordinates:

\[ r = \sqrt{x_e^2 + y_e^2 + r_e^2 + 2r_e(x_e \cos(\omega t) + y_e \sin(\omega t))} \]

\[ \phi = \arctan\left( \frac{y_e + r_e \sin(\omega t)}{x_e + r_e \cos(\omega t)} \right) \]  

(C.1)

For control purposes the first and second time derivatives are also needed:

\[ \ddot{r} = \frac{r_e \omega (y_e \cos - x_e \sin)}{\sqrt{h}} \]

\[ \dot{\phi} = \frac{r_e \omega (x_e \cos + y_e \sin + r_e)}{h} \]  

(C.2)

\[ \ddot{r} = -\frac{r_e \omega^2 (x_e \cos + y_e \sin)}{\sqrt{h}} - \frac{r_e^2 \omega^2 (y_e \cos - x_e \sin)^2}{h^{3/2}} \]

\[ \ddot{\phi} = \frac{r_e \omega^2 (y_e \cos - x_e \sin)}{h} - \frac{2r_e^2 \omega^2 (y_e \cos - x_e \sin)(x_e \cos + y_e \sin + r_e)}{h^2} \]  

(C.3)

with:

\[ si = \sin(\omega t) \]

\[ co = \cos(\omega t) \]  

(C.4)

\[ h = 2r_e x_e \cos + 2r_e y_e \sin + x_e^2 + y_e^2 + r_e^2 \]

Numerical values can be found in appendix I.
Appendix D

Motor data RT-robot simulations

<table>
<thead>
<tr>
<th>type:</th>
<th>Electro Craft B-540 SA</th>
<th>Honeywell 33VM62-200-3</th>
<th>Mavilor MO 4500</th>
</tr>
</thead>
<tbody>
<tr>
<td>$k_m$ [Nm/A]</td>
<td>70e-3</td>
<td>82e-3</td>
<td>0.67</td>
</tr>
<tr>
<td>R [Ω]</td>
<td>1.60</td>
<td>2.89</td>
<td>0.45</td>
</tr>
<tr>
<td>L [H]</td>
<td>4e-3</td>
<td>5.78e-4</td>
<td>8.98e-5</td>
</tr>
<tr>
<td>J [kgm²]</td>
<td>27e-6</td>
<td>3.72e-6</td>
<td>9e-3</td>
</tr>
<tr>
<td>$\tau_m$ [s]</td>
<td>8.6e-3</td>
<td>1.6e-3</td>
<td>9.0e-3</td>
</tr>
<tr>
<td>$\tau_c$ [s]</td>
<td>2.5e-3</td>
<td>2.0e-4</td>
<td>2.0e-4</td>
</tr>
</tbody>
</table>
Appendix E

Simulation algorithm RT-robot

start

initialization

t=t_0

e(t)=x(t)-x_d(t)
s=f(e)

u=f(e,s)

ZOH

x(t+\Delta t)=f(x(t),u)

t>t_a ?

yes

no

t=t+\Delta t
x(t)=x(t+\Delta t)

end

; state measurement
; inverse error dynamics

; calculation of control

; zero order hold (u)

; evaluate next state (ode23)

; make a timestep
Appendix F

Flexible model XY-table (see figure 4.1)

\[
M(q)\ddot{q} + C(q,\dot{q})\dot{q} + Kq + n(\dot{q}) = F \quad [N] \tag{F.1}
\]

\[
q = \begin{bmatrix} x \\ y \\ x'' \end{bmatrix}; \quad n = \begin{bmatrix} n_1 \\ n_2 \\ n_3 \end{bmatrix}; \quad F = \begin{bmatrix} T_1/r^2 \\ T_2/r^2 \\ 0 \end{bmatrix} \tag{F.2}
\]

\[
M = \begin{bmatrix} m_{11} & 0 & m_{13} \\ 0 & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix}; \quad C = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & 0 & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix}; \quad K = \begin{bmatrix} k_{11} & 0 & -k_{11} \\ 0 & 0 & 0 \\ -k_{11} & 0 & k_{11} \end{bmatrix} \tag{F.3}
\]

\[
m_{11} = \frac{J_1}{r^2} + m_s + \frac{1}{3}m_y \left( \frac{l}{d} \right)^2 + m_e \left( \frac{y}{d} \right)^2
\]

\[
m_{22} = \frac{J_2}{r^2} + m_e
\]

\[
m_{33} = m_s + m_y - m_y \left( \frac{l}{d} \right) + \frac{1}{3}m_y \left( \frac{l}{d} \right)^2 + m_e - 2m_e \left( \frac{y}{d} \right) + m_e \left( \frac{y}{d} \right)^2
\]

\[
m_{13} = m_{31} = \frac{1}{2}m_y \left( \frac{l}{d} \right) - \frac{1}{3}m_y \left( \frac{l}{d} \right)^2 + m_e \left( \frac{y}{d} \right) - m_e \left( \frac{y}{d} \right)^2
\]

\[
m_{23} = m_{32} = m_e \left( \frac{x - x''}{d} \right) \tag{F.4}
\]
\begin{equation}
\begin{gathered}
c_{11} = -c_{13} = m_e \left( \frac{1}{d^2} \right) \ddot{y} \\
c_{22} = -c_{23} = m_e \left( \frac{1}{d^2} \right) \ddot{x} - m_e \left( \frac{1}{d} \right) \dot{y} \\
c_{12} = c_{21} = -c_{21} = m_e \left( \frac{1}{d^2} \right) \ddot{y} (\dot{x} - \dot{x}') \\
c_{32} = m_e \left( \frac{1}{d} \right) (\dot{x} - \dot{x}') - m_e \left( \frac{1}{d^2} \right) \dot{y} (\dot{x} - \dot{x}')
\end{gathered}
\end{equation}

\begin{equation}
\begin{gathered}
n_1 = W_1 \text{sign}(\dot{x}) \\
n_2 = W_2 \text{sign}(\dot{y}) \\
n_3 = W_3 \text{sign}(\dot{x}') \\
k_{11} = \frac{k_1}{r^2}
\end{gathered}
\end{equation}

Numerical values can be found in appendix I.
Appendix G

Transfer functions XY-table

Figure G.1; transfer function x-direction

Figure G.2; transfer function y-direction
solid: actual system

Figure G.3; transfer function x-direction (fixed springs $k_j$)
Appendix H

Kalman observer XY-table

To estimate the position and speed one sample ahead, a Kalman observer is designed, based on a discrete time model (see Kok [1990] and Van Gerwen [1990]). At first we will consider a Kalman observer for the 'two degrees of freedom model', described in continuous time state space description by eq. 4.2. Because this model is only a simplification of the actual system there will be measurement noise \( v(t) \) and process noise \( w(t) \) present. The continuous time model can be rewritten as:

\[
\begin{align*}
\dot{x} &= Ax + B( u - E_f + w ) \\
y &= Cx + v \\
A &= \begin{bmatrix}
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & \frac{a_5}{a_1} & 0 \\
0 & 0 & 0 & \frac{a_6}{a_2}
\end{bmatrix} \\
B &= \begin{bmatrix}
1 \\
0 \\
0 \\
0
\end{bmatrix} \\
C &= \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0
\end{bmatrix}
\end{align*}
\]

\( E_f = \begin{bmatrix}
\frac{a_5 \text{sign}(x_3)}{a_1} \\
\frac{a_6 \text{sign}(x_4)}{a_2}
\end{bmatrix} \) (H.1)

Notice that the process noise is modelled as noise on the input \( u(t) \). The discrete time state space description can be written as:

\[
\begin{align*}
x(n+1) &= A_d \, x(n) + B_d \left[ u(n) - E_f(n) + w(n) \right] \\
y(n) &= C_d \, x(n) + v(n)
\end{align*}
\]

(H.2)

The discrete matrices \( A_d, B_d \) and \( C_d \) can be found with help of the software package MATLAB (c2d; continuous to discrete) and will not be demonstrated. To estimate the state one sample ahead, the observation update will be chosen according to Kok [1990]:
Appendix H: Kaiman observer XY-table

When we assume the measurement and process noise to be white noise sequences, we can find an optimal gain matrix $K^o$, and the observer will yield a minimum variance estimation. Again the software package MATLAB will be used to calculate this optimal gain matrix. To do so we have to determine the covariances of the process and measurement noise:

$$Q = \text{covariance}( \nu )$$

$$R = \text{covariance}( \nu )$$

In a previous investigation by Van Gerwen [1990] the covariance of the measurement noise has already been estimated by determining the measurement inaccuracy due to the incremental encoders. We have no reason to believe these values have changed, so we assume them to be correct:

$$R = \begin{bmatrix} 3.5e^{-12} & 0 \\ 0 & 8.4e^{-11} \end{bmatrix} \quad [m^2]$$

Due to a slightly different model in comparison to previous investigations (different parameters and viscous friction), we have to estimate the process noise ourselves. As described by Van Gerwen [1990] this is done by an experiment. After the experiment the actual velocity and acceleration can be determined using a central difference scheme. An input force is calculated such that the discrete time model follows the measured speed exactly. The difference between this calculated input force and the actually applied input force is the process noise. We determined the covariance as:

$$Q = \begin{bmatrix} 250 & 0 \\ 0 & 9.3 \end{bmatrix} \quad [N^2]$$

As mentioned before the optimal gain matrix $K^o$ can now be calculated with the software package MATLAB (dlqe; discrete linear quadratic estimator).
This yields:

\[ K^O = \begin{bmatrix}
1.9 & 0 \\
0 & 1.7 \\
185.4 & 0 \\
0 & 159.2
\end{bmatrix} \quad (H.7) \]

This result matches approximately with Van Gerwen [1990].

Secondly we consider the case when we use the 'three degrees of freedom model' (eq. 4.3) in the controller implementation. To determine the discrete Kalman observer the same method is used as described before in case of the 'two degrees of freedom model'. The only trouble is to determine the process noise of the last system equation (the internal dynamics). Because this equation doesn't have an input, we cannot use the method of determining \( Q \) as described before in this case and therefore we tried several values and chose the one yielding the best results. The covariance of the 'three degrees of freedom model' is chosen as:

\[ Q = \begin{bmatrix}
250 & 0 & 0 \\
0 & 9.3 & 0 \\
0 & 0 & 25000
\end{bmatrix} \quad [N^2] \quad (H.8) \]

Notice the very high value of the covariance of the process noise of the last system equation. Apparently this equation is inaccurate.

The optimal filter gain matrix is found as:

\[ K^O = \begin{bmatrix}
3.2 & 0 \\
0 & 1.7 \\
12.3 & 0 \\
562.8 & 0 \\
0 & 159.2 \\
275.2 & 0
\end{bmatrix} \quad (H.9) \]
Appendix I

Controller settings and constants

RT-robot

parameters

\[
\begin{align*}
P_1 &= 15 \quad \text{[kg]} & m_1 &= 5 \quad \text{[kg]} \\
P_2 &= 5 \quad \text{[kgm]} & m &= 10 \quad \text{[kg]} \\
P_3 &= 8.333 \quad \text{[kgm}^2\text{]} & J &= 5 \quad \text{[kgm}^2\text{]} \\
I &= 1 \quad \text{[m]} &
\end{align*}
\]

trajectory

\[
\begin{align*}
\xi &= 0.5 \quad \text{[m]} \\
\gamma &= 0 \quad \text{[m]} \\
r_c &= 0.25 \quad \text{[m]} \\
\omega &= \pi \quad \text{[rad/s]} \\
\end{align*}
\]

transfer ratio

\[
\begin{align*}
i_1 &= 0.05 \quad \text{[m/rad]} \\
i_2 &= 0.04 \quad [-] \\
\end{align*}
\]

controller setting 1

\[
\begin{align*}
k &= 4 \\
\omega_n &= 10 \\
z_0 &= 0.2 \\
c_0 &= 4 \\
c_1 &= 4 \\
\phi &= 0.05 \\
\end{align*}
\]

controller setting 2

\[
\begin{align*}
k &= 10 \\
\omega_n &= 15 \\
z_0 &= 0.2 \\
c_0 &= 225 \\
c_1 &= 30 \\
\phi &= 0.47 \\
\end{align*}
\]

disturbances

see motor data RT-robot simulation (appendix D)
# Appendix I: controller settings and constants

## XY-table

### parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m_s$</td>
<td>3.8</td>
<td>kg</td>
</tr>
<tr>
<td>$m_a$</td>
<td>2.3</td>
<td>kg</td>
</tr>
<tr>
<td>$m_2$</td>
<td>7.6</td>
<td>kg</td>
</tr>
<tr>
<td>$J_1$</td>
<td>1.75e-3</td>
<td>kgm$^2$</td>
</tr>
<tr>
<td>$J_2$</td>
<td>4e-5</td>
<td>kgm$^2$</td>
</tr>
<tr>
<td>$l$</td>
<td>1</td>
<td>m</td>
</tr>
<tr>
<td>$d$</td>
<td>1.25</td>
<td>m</td>
</tr>
<tr>
<td>$r_x$</td>
<td>0.01</td>
<td>m</td>
</tr>
<tr>
<td>$r_y$</td>
<td>0.01</td>
<td>m</td>
</tr>
</tbody>
</table>

### parameters stiff model (4.2)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_1$</td>
<td>34</td>
<td>kg</td>
</tr>
<tr>
<td>$a_2$</td>
<td>2.7</td>
<td>kg</td>
</tr>
<tr>
<td>$a_3$</td>
<td>36</td>
<td>N</td>
</tr>
<tr>
<td>$a_4$</td>
<td>9</td>
<td>N</td>
</tr>
<tr>
<td>$a_5$</td>
<td>50</td>
<td>Ns/m</td>
</tr>
<tr>
<td>$a_6$</td>
<td>8</td>
<td>Ns/m</td>
</tr>
</tbody>
</table>

### controller setting stiff model (4.2)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$k$</td>
<td>70</td>
</tr>
<tr>
<td>$\omega_n$</td>
<td>45</td>
</tr>
<tr>
<td>$z_0$</td>
<td>90</td>
</tr>
<tr>
<td>$c_0$</td>
<td>2025</td>
</tr>
<tr>
<td>$c_1$</td>
<td>63.6</td>
</tr>
<tr>
<td>$\Phi$</td>
<td>1.11</td>
</tr>
</tbody>
</table>
parameters extended model (4.3)

\[
\begin{align*}
    a_{1,1} &= 17.5 \quad \text{[kg]} \\
    a_2 &= 2.7 \quad \text{[kg]} \\
    a_{1,2} &= 16.5 \quad \text{[kg]} \\
    a_{3,1} &= 28 \quad \text{[N]} \\
    a_4 &= 9 \quad \text{[N]} \\
    a_{3,2} &= 8 \quad \text{[N]} \\
    a_{5,1} &= 25 \quad \text{[Ns/m]} \\
    a_6 &= 8 \quad \text{[Ns/m]} \\
    a_{5,2} &= 25 \quad \text{[Ns/m]} \\
    k_2 &= 23300 \quad \text{[N/m]}
\end{align*}
\]

controller setting extended model (4.3)

\[
\begin{align*}
    k &= 70 \\
    \omega_a &= 70 \\
    z_0 &= 140 \\
    c_0 &= 4900 \\
    c_1 &= 99 \\
    \Phi &= 0.7
\end{align*}
\]