Analysis of Velocity Estimation Methods for High-Performance Motion Control Systems

RUTH ALEKSANDRAUSKAITE
Analysis of Velocity Estimation Methods for High-Performance Motion Control Systems

RUTH ALEKSANDRAUSKAITE
Abstract

The majority of all commercial electronics hardware is manufactured using Surface Mount Technology (SMT). Nevertheless, the increased complexity and miniaturization of electronics impose tough performance requirements on the automation process.

The research in this paper concerns test and analysis of alternative velocity estimation methods for high-performance embedded motion control systems. The motion system in Mycronic’s pick and place machines is regulated by a motion controller consisting of a feedforward component and a feedback controller. The linear displacement is measured with an incremental encoder and the velocity is estimated with a state observer. Previous work suggests that the velocity estimation is inadequate.

Different observer designs including state and disturbance estimators were tested and evaluated through simulations in MATLAB SIMULINK®. After that, experiments were performed on a conveyor retrieved from a pick and place machine.

The results show that a Kalman filter is the best state estimator. However, the method requires extensive tuning to attain good performance. The tracking performance and robustness of the motion control system was highly improved when using a Perturbation observer with Kalman filtering. Nonetheless, the settling time for point-to-point movements was somewhat shorter when using a Kalman filter alone.
Sammanfattning

Majoriteten av all kommersiell elektronikhårdvara tillverkas med ytmonteringsteknik. Ökad komplexitet och miniatyrisering av elektronik medför hårdare prestandakrav på automationsprocessen.


Diverse tekniker för rekonstruktion av tillstånd har testats och utvärderats via simulering i MATLAB SIMULINK®. Därefter utfördes experiment på ett transportband hämtat från en pick- and placemaskin.

# Contents

**Acronyms and Abbreviations** vii

<table>
<thead>
<tr>
<th>1 Introduction</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 Problem</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Research Questions</td>
<td>2</td>
</tr>
<tr>
<td>1.3 Delimitations</td>
<td>2</td>
</tr>
<tr>
<td>1.4 Method</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2 Background</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1 System Overview</td>
<td>4</td>
</tr>
<tr>
<td>2.2 Motion Control System</td>
<td>5</td>
</tr>
<tr>
<td>2.2.1 Feedforward Controller</td>
<td>7</td>
</tr>
<tr>
<td>2.2.2 Feedback Controller</td>
<td>7</td>
</tr>
<tr>
<td>2.2.3 Observer</td>
<td>8</td>
</tr>
<tr>
<td>2.3 Disturbances and Uncertainties</td>
<td>9</td>
</tr>
<tr>
<td>2.3.1 Friction Force</td>
<td>9</td>
</tr>
<tr>
<td>2.3.2 Force Ripples</td>
<td>10</td>
</tr>
<tr>
<td>2.4 Analysis of Alternative Velocity Estimation Methods</td>
<td>10</td>
</tr>
<tr>
<td>2.4.1 Finite Difference</td>
<td>10</td>
</tr>
<tr>
<td>2.4.2 Direct Observer</td>
<td>12</td>
</tr>
<tr>
<td>2.4.3 Tracking Differentiator</td>
<td>12</td>
</tr>
<tr>
<td>2.4.4 Adaptive Windowing</td>
<td>13</td>
</tr>
<tr>
<td>2.4.5 Kalman Filter</td>
<td>14</td>
</tr>
<tr>
<td>2.4.6 Disturbance Observers</td>
<td>15</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>3 Simulations</th>
<th>18</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1 Simulation Model and Set-up</td>
<td>18</td>
</tr>
<tr>
<td>3.1.1 Test Cases</td>
<td>19</td>
</tr>
<tr>
<td>3.1.2 Measure of Performance</td>
<td>20</td>
</tr>
</tbody>
</table>
3.2 Simulation Results ........................................... 20
  3.2.1 Benchmark Performance ................................. 21
  3.2.2 One-step Predictor with Central Difference Approximation ......................... 22
  3.2.3 Direct Observer ........................................... 22
  3.2.4 Kalman Filter ............................................. 23
  3.2.5 Perturbation Observer ................................. 24
  3.2.6 Extended State Observer ......................... 24
  3.2.7 Evaluation and Discussion ......................... 25

4 Experiments .................................................. 29
  4.1 Test Rig ................................................... 29
  4.2 Test Cases ................................................. 29
  4.3 Measure of Performance ............................. 31
    4.3.1 Parameter Tuning ........................................ 31
  4.4 Results .................................................. 34
    4.4.1 Point-to-point ........................................ 34
    4.4.2 Constant Speed ........................................ 36
    4.4.3 Robustness Test ...................................... 39

5 Discussion ................................................. 40

6 Conclusions ............................................... 43

Bibliography .................................................. 44
## Acronyms and Abbreviations

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAN</td>
<td>Controller Area Network.</td>
</tr>
<tr>
<td>DO</td>
<td>Direct Observer.</td>
</tr>
<tr>
<td>ESO</td>
<td>Extended State Observer.</td>
</tr>
<tr>
<td>KF</td>
<td>Kalman Filter.</td>
</tr>
<tr>
<td>OSP</td>
<td>One-step Predictor.</td>
</tr>
<tr>
<td>OSPCD</td>
<td>One-step Predictor with Central Difference approximation.</td>
</tr>
<tr>
<td>PCB</td>
<td>Printed Circuit Board.</td>
</tr>
<tr>
<td>PO</td>
<td>Perturbation Observer.</td>
</tr>
<tr>
<td>POKF</td>
<td>Perturbation Observer with Kalman Filtering.</td>
</tr>
<tr>
<td>SMT</td>
<td>Surface Mount Technology.</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

One of the challenges in the electronics industry nowadays concerns fitting complex circuitry into small spaces. Hence, the vast majority of mass-produced electronics are manufactured using SMT. The components are mounted directly onto the circuit board, and the process is regulated by an embedded motion control system.

Precision and speed requirements on the motion control system become more strict due to the miniaturization and the increased complexity of electronics. Therefore, further development of the control system is of high importance in order to achieve increased performance.

The thesis work was performed on behalf of Mycronic AB located in Stockholm, which develops and produces equipment for electronics and display industries.

1.1 Problem

Surface mount equipment used for assembly of circuit boards are commonly called, pick and place machines. The movements in such machines are characterized by large accelerations and high speed. It is desired that the movements are performed with high precision and as fast as possible.

The motion system in Mycronic’s pick and place machines is driven by a linear motor, which is regulated by a state feedback controller. The velocity is estimated with a state observer and a feedforward controller is implemented...
to compensate for the friction force. Previous research in [1] has established that the observer part of the controller is inadequate. Consequently, enhanced velocity estimation could improve the overall performance.

1.2 Research Questions

The thesis work concerns analysis, testing, and evaluation of different methods that could potentially improve the performance of the embedded control system in a pick and place machine. One purpose of this project was to assess if increased performance can be achieved with the proposed modifications of the control system.

The following research questions were formulated: What methods can be used in order to achieve more accurate and precise velocity estimations? How does it affect the motion control, could an increased performance be achieved compared to the current existing control design? How robust are the proposed methods?

1.3 Delimitations

It is beyond the scope of this study to examine the motion control system of an entire machine, hence the project was limited to one subsystem. A conveyor retrieved from a pick and place machine was utilized in the experimental testing. Hence, the disturbances induced by the remaining moving part of the machines were omitted.

1.4 Method

In order to answer the research questions formulated in Section 1.2 the study was divided into seven stages:

1. Gain knowledge about the system and its embedded control system.
2. Literature study concerning modern observer design for high-performance motion control.
3. Implement the chosen controller design in MATLAB SIMULINK®.
4. Simulate and evaluate the theoretical performance of the velocity estimation and the control performance.

5. Implement the modified controller in the existing embedded code and evaluate the motion control performance for the test rig.

6. Draw conclusions regarding the proposed methods.

7. Perform a robustness analysis.
Chapter 2

Background

In this chapter, a brief introduction of the system is presented, followed by a detailed description of the current motion control system. Finally, an extensive study is performed, based on the analysis of the existing controller design, recent research and development within the area. Different techniques that may improve the performance of the embedded control system were discussed and analyzed.

2.1 System Overview

A MY200 pick and place machine works with a split-axis motion, where the pick head travels along the x-axis, picks an electrical component from the storage unit and places it on a Printed Circuit Board (PCB). The Y-wagon is responsible for positioning the PCB along the y-axis. The thesis work is limited to the study of one subsystem of the MY200, namely the Y-wagon. A brief overview of the relevant parts of a MY200 is presented below and a visualization is seen in Figure 2.1.

The central computer controls all parts of the machine. The communication with the different nodes is handled via the Controller Area Network (CAN).

The controller board, called CMOT, is located near each subsystem that is controlled by the central computer. The CMOT contains the hardware that enables communication with the central computer. When a move command is received, the CMOT calculates the corresponding reference trajectories and
determines the control current. The current is then converted to an analog voltage signal, which is then sent to the motor drive.

The motor drive receives the voltage signal from the CMOT. The output is an electric current that is fed into the motor.

The motor used is a permanent magnet synchronous brushless linear motor with an iron core.

The Y-wagon positions the PCB along the y-axis. The linear motor is mounted underneath the wagon.

The incremental encoder is used for position measurement. The encoder has a resolution of 0.5 μm and the pulses are counted by the CMOT.

### 2.2 Motion Control System

In this section, a detailed description of the existing motion control system is presented and discussed. The controller design has two degrees of freedom consisting of a feedforward component and a feedback controller, which are further described in the forthcoming sections. A block diagram of the controller design is shown in Figure 2.2. The information was obtained mainly from [1, chap. 2] and by studying the embedded system code.

The system dynamics for the Y-wagon are presented in [1, p. 26]

\[
m\ddot{x} = K_f i - F_{fric}(\dot{x}) - F_{cog}(x) - F_{dis}
\]  

(2.1)

where \(x\) denotes the linear displacement, \(m\) is the mass, \(i\) denotes the current fed to the motor, and \(K_f\) corresponds to the motor force constant. Moreover,
Figure 2.2: Visualization of the control system, where \( \hat{x}_2 \) is the estimated velocity, \( u \) the control input, \( y \) is the measured position and \( r, \dot{r}, \ddot{r} \) is the reference signal and its derivatives.

\( F_{\text{fric}} \) is the combined static and dynamic friction, \( F_{\text{cog}} \) denotes the cogging force and \( F_{\text{dis}} \) denotes other disturbances. The motor drive is assumed to be an ideal current source, hence the electrical properties of the motor are not modeled.

It is desired that the system follows the reference signal \( r \). The tracking error \( e \) and its derivatives are defined as

\[
e = r - x, \quad \dot{e} = \dot{r} - \dot{x}, \quad \text{and} \quad \ddot{e} = \ddot{r} - \dddot{x}. \quad (2.2)
\]

Thus, the system in (2.1) can be written as

\[
\ddot{e} = \ddot{r} + \frac{1}{m} \left( F_{\text{fric}}(\dot{r} - \dot{e}) + F_{\text{cog}}(r - e) - K_f i + F_{\text{dis}} \right). \quad (2.3)
\]

Further on, using the notation \( e_1 = e \) and \( e_2 = \dot{e} \), the system can be written in state space form

\[
\dot{e}_1 = e_2, \quad \dot{e}_2 = \ddot{r} + \frac{1}{m} \left( F_{\text{fric}}(\dot{r} - e_2) + F_{\text{cog}}(r - e_1) - K_f i + F_{\text{dis}} \right), \quad (2.4)
\]

where the current \( i \) is the control input. A pick and place machine operates under high velocities and large accelerations. The influence from cogging is then relatively small compared to the dynamics (acceleration and deceleration) [12], therefore the cogging force is not considered in the existing controller design. The term \( F_{\text{dis}} \) corresponds to other unknown disturbances and nonlinearities and is therefore ignored.
2.2.1 Feedforward Controller

The feedforward controller calculates the current needed to obtain the desired acceleration \( \ddot{r} \) and compensates for the friction force. Hence, the feedforward control law corresponds to

\[
 u_{\text{ff}} = \frac{1}{K_f} \left( F_{\text{fric}}(\dot{r}) + m\ddot{r} \right), 
\]

(2.5)

where \( u_{\text{ff}} \) denotes the feedforward current. Note that, in order to achieve cancellations of the friction force in (2.4) with the feedforward control defined in (2.5), an accurate friction model is desired. Furthermore, the tracking error should ideally be equal to zero. The friction model used in the feedforward controller is a first order model

\[
 F_{\text{fric}}(v) = \text{sign}(v)(D|v| + F_{\text{static}}), 
\]

(2.6)

where \( D \) is the dynamic friction coefficient, \( v \) is the velocity and \( F_{\text{static}} \) is the static friction.

2.2.2 Feedback Controller

A state feedback control is used to drive the tracking error to zero. Integral action is added to the controller in order to eliminate the steady state error. The state vector is therefore extended with a third state \( e_3 \) which denotes the integrated position error, namely \( e_3 = \int_0^t e_1(\tau)d\tau \), with the dynamics

\[
 \dot{e}_3 = e_1. 
\]

(2.7)

The control law is thus

\[
 u_c = L_1 e_1 + L_2 \hat{e}_2 + L_3 e_3, 
\]

(2.8)

where \( L_1, L_2 \) and \( L_3 \) denote the feedback gains, \( u_c \) is the feedback controller current and \( \hat{e}_2 = \dot{r} - \hat{x}_2 \) is the velocity error. Note that \( \hat{x}_2 \) is the estimated velocity that is obtained from the observer, see Figure 2.2. The discrete-time implementation of the integrated position error follows

\[
 \int_0^t e_1(\tau)d\tau \approx T(e_1(t-T) + e_1(t-2T) + \ldots) 
\]

(2.9)
where $T$ is the sampling time. The tracking error dynamics correspond to
\[
\dot{\vec{e}} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \vec{e} + \begin{bmatrix} 0 \\ \frac{K_i}{m} \\ 0 \end{bmatrix} u_c,
\]
(2.10)
where $\vec{e} = (e_1, e_2, e_3)^T$ is the state vector. Pole placement technique [2, p. 124] is used to place the closed-loop poles in specific locations on the complex plane. By choosing the location of the poles, a desired system response can be achieved, provided that the sampled system is controllable. Note that the controller is basically a PID controller [5, p. 53 - 54] where the parameters $K_P$, $K_I$, and $K_D$ are chosen with the pole placement technique.

### 2.2.3 Observer

An observer is used in order to estimate the velocity $\dot{x}$ of the system. The system dynamics are approximated as the linear relation between the acceleration and the input current, namely
\[
m\ddot{x} = K_f i,
\]
(2.11)
known as the double integrator. Defining the states as $x_1 = x$, $x_2 = \dot{x}$ and the input $u = i$, the model is written in state space form
\[
\begin{bmatrix} \dot{\vec{x}} \\ y \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{K_i}{m} \end{bmatrix} u,
\]
(2.12)
where $\vec{x} = [x_1, x_2]^T$ is the state vector and $y$ denotes the output. The discrete time description [20, p. 80], with the sampling time $T$ and zero-order hold, corresponds to
\[
\begin{aligned}
\vec{x}[k + 1] &= \Phi \vec{x}[k] + \Gamma u[k], \\
y[k] &= C \vec{x}[k],
\end{aligned}
\]
(2.13)
where $k = 0, 1, 2...$ and $\vec{x}[k + 1]$ corresponds to $\vec{x}(kT + T)$ in continuous time. The matrices in (2.13) are defined below
\[
\Phi = e^{\Delta T} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}, \quad \Gamma = \int_0^T e^{\Delta s} ds B = \frac{K_i T}{m} \begin{bmatrix} 1 \\ \frac{1}{2} \end{bmatrix}.
\]
(2.14)
The estimated state vector is denoted \( \hat{x} \) and the observer dynamics follows
\[
\hat{x}[k+1|k] = \Phi \hat{x}[k|k-1] + \Gamma u[k] + K(y[k] - C \hat{x}[k|k-1])
\] (2.15)
where \( K = [k_1, k_2]^\top \) are the observer gains. The notation \( \hat{x}[k+1|k] \) implies that the estimate at time \( k+1 \) is determined using the measurements that are available at time \( k \). The observer is interpreted as a one-step predictor, which predicts the system states at the next time instance \( k+1 \) using the estimated states at \( k \) with additional feedback from the estimation error, weighted by the observer gains. Note that the feedback term in (2.15) gives no contribution if the predicted output from the observer is the same as the measured one.

Provided that the system is observable i.e. the pair \( (\Phi, C) \) is observable, the observer gains can be selected to ensure that estimation error \( \hat{x} - \hat{x} \) converges to zero. The closed-loop poles correspond to the eigenvalues of the system matrix \( (\Phi - KC) \) [4] hence, choosing the observer gains as
\[
\begin{align*}
k_1 &= 2 - p_1 - p_2, \\
k_2 &= \frac{1}{T}(1 - p_1)(1 - p_2),
\end{align*}
\] (2.16)
will place the discrete-time poles \( p_1, p_2 \) of (2.15) at the specified locations. Stable continuous-time system poles (Laplace domain) are mapped into a unit circle on the \( z \)-plane through \( z = e^{sT} \) [20, p. 89]. The selection of observer poles is a compromise between the sensitivity to measurement errors and how fast the estimated states converge to the true ones [2]. Hence, the desired position of the observer poles are in fact tuning parameters.

2.3 Disturbances and Uncertainties

Disturbances and model uncertainties are always present in real systems. To get a wider understanding of the system, the main sources of disturbance and their influence on the motion control are further discussed.

2.3.1 Friction Force

Friction is always present in all moving mechanics. Moreover, it is well-known that the friction phenomena is hard to accurately describe on an explicit form. Currently, a simple first order friction model (2.6) is used in the feedforward controller. The model is however only a rough approximation of the actual friction force.
2.3.2 Force Ripples

The ripple forces are generated due to the magnetic structure of the permanent magnet linear motor. There are two main components in the force ripple, namely the cogging force and the reluctance force. The cogging arise due to the attraction between the permanent magnets and the iron core. Even in the absence of current, the phenomenon is still present and varies along the magnet track. The reluctance force occurs due to variations in the self-inductance of the windings. Consequently, the two components constitute the force ripple [17]. As mentioned earlier, the phenomenon is more dominant at low speed. Hence, considering e.g. point-to-point control, the controller oscillates around the desired position as it settles, where the ripple has a significant effect.

2.4 Analysis of Alternative Velocity Estimation Methods

The goal of this thesis work was to improve the performance of the motion control system used for the control of a Y-wagon. The project at hand concerns a noisy environment, both measurement noise and disturbances from the surroundings are present. Furthermore, even if the linear dynamics are dominant, unmodelled nonlinear dynamics do affect the performance and have to be dealt with in the new controller design. In order to make the implementation feasible one needs to keep in mind that the embedded hardware has limited resources. For that purpose, an analysis of potential velocity estimation techniques for high-performance motion control applications is presented in the forthcoming sections. The methods presented in this report were acquired from other literature. Nonetheless, some combinations of the different techniques are believed to be unique.

2.4.1 Finite Difference

The experiments performed in [1] on the Y-wagon revealed a significant undershoot during the settling process, which can be seen in Figure 2.3. The behavior is typical for non-minimum phase systems which are characterized by zero(s) in the right half plane [4, p.173]. This undesired phenomenon was
concluded to be caused by the current observer since it vanishes when the observer was replaced with the differentiated velocity \( v \)

\[
v[k] \approx \frac{y[k] - y[k-1]}{T}.
\]  

(2.17)

This, however, resulted in noisy velocity estimations, which caused a chatter on the input signal and is therefore not applicable in practice. The noise was reduced by including more samples when differentiating the velocity as described in [10]. Unfortunately, the delay imposed by the filter caused oscillations at low speed [1], [22].

![Figure 2.3: Response of a 20 mm move by the Y-wagon.](image)

The position is measured by an incremental encoder, hence the position measurements are contaminated with a quantization error as well as measurement noise. Using finite difference (2.17) the noise is amplified with a factor \( 1/T \) and therefore results in noisy velocity estimations.

Consequently, it is desired that the noise is attenuated and the delay is minimized. A prediction of future output \( \hat{x}_1[k+1] \) is obtained using the observer presented in Section 2.2.3. Note though that \( u[k] \) has been replaced with \( u[k-1] \) in order to make the observer causal. Then the velocity can be estimated with a central difference approximation

\[
\hat{x}_2[k] = \frac{\hat{x}_1[k+1] - \hat{x}_1[k-1]}{2T},
\]  

(2.18)
which should result in less noisy estimations and a shorter delay compared to backward difference estimation (2.17). The method will be referred to as the One-step Predictor with Central Difference approximation (OSPCD).

### 2.4.2 Direct Observer

The current feedback controller design uses the velocity estimated by the observer presented in Section 2.2.3. Note though that the observer has a delay, since the estimation \( \hat{x}_2 \) at time sample \( k \) only depends on measurements up to \( k - 1 \). It is well-known that that long time delays result in inaccurate estimations, and in precision tracking the accuracy of the estimates is essential in order to achieve high performance. Therefore, an alternative observer, the Direct Observer (DO) is proposed in [2, p. 140], which include the current position measurement in the estimation of the states

\[
\hat{x}[k|k] = (I - KC)(\Phi\hat{x}[k - 1|k - 1] + \Gamma u[k - 1]) + Ky[k]
\]  

(2.19)

where \( K = [k_1, k_2]^T \) denotes the observer gains. Then, provided that the pair \((\Phi, C\Phi)\) is observable, the closed-loop system matrix \((\Phi - KC\Phi)\) can be given arbitrary eigenvalues [4] by choosing

\[
k_1 = 1 - p_1p_2, \quad k_2 = \frac{1}{T}(1 + p_1p_2 - p_1 - p_2),
\]

(2.20)

where the notation \( p_1 \) and \( p_2 \) denotes the location of the discrete-time poles on the \( z \)-plane.

### 2.4.3 Tracking Differentiator

As mentioned earlier, the critical part of the motion is in the low-speed region. Therefore, the velocity estimation algorithm developed in [14] is further analyzed. The method is based on the fact that numeric integration can provide more accurate and precise estimations compared to numeric differentiation which is noise amplifying. The enhanced tracking differentiator in [14] is proposed as an improved velocity estimator in the low-speed regions, and is the successor to the tracking differentiator developed in [15], [16]. Su et al. [14] claim that the enhanced tracking differentiator achieves accurate velocity estimations at low speed. Moreover, the method is said to be very robust
against noise, which was verified with simulations and experimental result in the control of a permanent magnet synchronous motor. The dynamics of the discrete enhanced differentiator can be expressed as

\[
\begin{align*}
\dot{x}_1[k+1] &= \dot{x}_1[k] + T\dot{x}_2[k] \\
\dot{x}_2[k+1] &= \dot{x}_2[k] + TR^2\left[-\alpha_0(\dot{x}_1[k] - y[k]) - \alpha_1(\dot{x}_1[k] - y[k])^{n/m} - \alpha_2 \left(\frac{\dot{x}_2}{R}\right)^{n/m}\right]
\end{align*}
\] (2.21)

where \( R > 0, \alpha_0, \alpha_1, \alpha_2 > 0 \) and \( m > n \) are positive odd numbers. Nonetheless, the drawback of this method is the large amount of tuning parameters. The article does not mention how to choose the parameters in order to achieve a desired response. On the other hand, the proposed observer is model free and therefore easy to implement for the different subsystems of a pick and place machine.

### 2.4.4 Adaptive Windowing

Various filtering techniques have been developed in order to achieve more accurate velocity estimations. In [8] an adaptive windowing method is presented. The windowing technique is supposed to minimize the error variance while maximizing the accuracy of the estimates. Lower variance in the estimation error can be obtained if the position measurement are far apart e.g. averaging over the \( n \in \mathbb{N}_{>0} \) latest velocity estimates

\[
\dot{x}_2 = \frac{y[k] - y[k-n]}{nT},
\] (2.22)

where \( n \) corresponds to the window size. Note though that a large window size causes a delay and also reduces the accuracy of the estimate. For that reason, the window size is selected adaptively, depending on the signal. A short window size is suggested in [8] when the velocity is high, which then results in more accurate estimates. The window size should be large when the velocity is low, yielding less noisy estimates. Hence, the noise reduction and the accuracy requirements of the estimates put a lower and upper bound on the window size. Considering that the time delay in time-critical systems is particularly harmful, the proposed method may result in a rather low upper bound which will be more or less equivalent to the noisy backward difference approximation.
2.4.5 Kalman Filter

In order to handle the process and measurement noise in the physical system, Kalman filtering was considered [13]. Stochastic disturbance (Gaussian white noise) was therefore introduced [11, p.193-203] in the discrete linear model (2.13), yielding

\[
\tilde{x}[k+1] = \Phi \tilde{x}[k] + \Gamma u[k] + \tilde{w}[k]
\]

\[
y[k] = C \tilde{x}[k] + n[k]
\]

(2.23)

where \(\tilde{w}[k]\) denotes the process noise with covariance matrix \(Q\) and \(n[k]\) is the measurement noise with variance \(r\). Since the system is viewed as a double integrator the disturbance only acts on the acceleration and thus the structure of \(Q\) becomes

\[
Q = \begin{bmatrix} 0 & 0 \\ 0 & q \end{bmatrix}
\]

(2.24)

where \(q\) denotes the variance of the process noise that enters the second state equation. Notice that the magnitude of \(q\) reflects the disturbance on the input signal.

The choice of the observer gains is a trade-off between sensitivity to measurement noise and how fast the estimation error converges to zero. The optimal state observer, known as the Kalman filter provides the optimal estimate of \(\tilde{x}\) minimizing the covariance matrix \(P\) of the estimation error \(\tilde{x}\) using series of noisy measurements [18, p. 126-131]. Henceforth, the discrete Kalman Filter (KF) presented in [8], [13] follows

\[
\hat{x}[k|k-1] = \Phi \hat{x}[k-1|k-1] + \Gamma u[k-1],
\]

\[
P[k|k-1] = \Phi P[k-1|k-1] \Phi^T + Q,
\]

(2.25)

where \(P\) is the covariance matrix of the estimation error. The Kalman gain is determined as

\[
H[k] = P[k|k-1]C^T(CP[k|k-1]C^T + r)^{-1},
\]

(2.26)

and finally the prediction in (2.25) is updated according to

\[
\hat{x}[k|k] = \hat{x}[k|k-1] + H[k](y[k] - C \hat{x}[k|k-1]),
\]

\[
P[k|k] = (I - H[k]C) P[k|k-1].
\]

(2.27)

Further on, the dynamics of the reconstruction error corresponds to

\[
\bar{x}[k] = \bar{x}[k] - \tilde{x}[k] = (\Phi - HC) \bar{x}[k-1] + \bar{w}[k-1] - Hn[k-1],
\]

(2.28)
where one can explicitly see how the noise terms affect the error. The measurement noise is multiplied with the Kalman gain and thus can be highly amplified. Therefore, the parameter $r$ should be tuned in order to reflect the magnitude of the measurement noise. The gain can be reduced if the resulting estimates are considered too noisy by choosing a higher value on $r$, see relation (2.26).

### 2.4.6 Disturbance Observers

In all physical systems, disturbances and uncertainties exist to some extent. Different techniques are discussed in [3] that deal with the influence of model uncertainties and unknown disturbance on the control performance. The rejection of these effects can increase both the robustness but also the overall tracking performance. The methods developed has been applied in various industrial sectors such as chemical and aerospace [3] to mention a few.

**Perturbation Observer**

The perturbation model presented in [9] includes unmodeled dynamics, external disturbances as well as model uncertainties which are lumped together and denoted $d_l$. The perturbation model is composed of the input signal, nominal model parameters and state signals as follows

$$ d_l[k, \bar{x}] = \Gamma^+(\bar{x}[k+1] - \Phi \bar{x}[k]) - u[k], \quad (2.29) $$

where $\Gamma^+ = (\Gamma^T \Gamma)^{-1} \Gamma^T$ is the pseudo inverse of $\Gamma$. Then assuming that the actual perturbation is quite smooth, a causal disturbance observer is constructed

$$ \hat{d}_l[k] = Q(\Gamma^+(\hat{x}[k] - \Phi \hat{x}[k-1]) - u[k-1]), \quad (2.30) $$

where $Q$ is a discrete low-pass filter, which prevent high amplification of measurement noise. A first order discrete low-pass with a cutoff frequency $f_c$ is suggested in [9]

$$ Q(z) = \frac{a}{1 + b z^{-1}}, \quad \text{where} \quad a = \frac{1}{1 + \alpha}, \quad b = -\frac{\alpha}{1 + \alpha}, \quad (2.31) $$

and $\alpha = 1/(2\pi f_c T)$. The complete perturbation observer combined with a state estimator is formulated

$$ \hat{d}_l[k] = -b \hat{d}_l[k-1] + a(\Gamma^+(\hat{x}[k] - \Phi \hat{x}[k-1]) - u[k-1]), $$

$$ \hat{x}[k+1] = \Phi \hat{x}[k] + \Gamma (u[k] + \hat{d}_l[k]) + K (y[k] - C \hat{x}[k]), \quad (2.32) $$
which in this paper will be referred to as the Perturbation Observer (PO). By including the disturbance estimation in the second row in (2.32), the overall performance of the state estimation could be improved [9]. Note though, that the design of the $Q$-filter is essential in order to achieve robustness and good disturbance suppression [19].

**Perturbation Observer with Kalman Filtering**

During the experiment phase of the project, the state estimator in (2.32) was replaced with a Kalman filter, named the Perturbation Observer with Kalman Filtering (POKF). The observer dynamics the POKF corresponds to

$$\hat{x}[k|k-1] = \Phi\hat{x}[k-1|k-1] + \Gamma(u[k-1] + d_l[k-1]),$$

$$P[k|k-1] = \Phi P[k-1|k-1] \Phi^T + Q,$$

where the predictions are updated according to (2.27) with the Kalman gain defined in (2.26). Lastly, the disturbance estimation at time instance $k$ is determined as follows

$$\hat{d}_l[k] = -b\hat{d}_l[k-1] + a(\Gamma^+(\hat{x}[k|k] - \Phi\hat{x}[k-1|k-1]) - u[k-1]).$$

**Extended State Observer**

The Extended State Observer (ESO) proposed by Han and described in [6] is a widely applied method for estimation and attenuation of unknown disturbances and model uncertainties [3]. An extended-state-observer-based controller was successfully designed for e.g. hydraulic system [21]. The general method introduces the lumped disturbance $d_l = f(t)$ as an additional state in the observer. The system is now approximated as

$$\dot{x}_1 = x_2, \quad \dot{x}_2 = \frac{K_t}{m}u, \quad \dot{x}_3 = \hat{f}(t),$$

where $\hat{f}(t)$ is unknown. Henceforth, the discrete-time ESO was constructed for the specific problem as described in [6] with $\epsilon$ defined as the difference between the measured and estimated position

$$\hat{x}_1[k+1] = \hat{x}_1[k] + T\hat{x}_2[k] + \frac{K_t T^2}{2m}(u[k] + \hat{x}_3[k]) - \beta_1 \epsilon$$

$$\hat{x}_2[k+1] = \hat{x}_2[k] + \frac{K_t T}{m}(u[k] + \hat{x}_3[k]) - \beta_2 \text{fal}(\epsilon, \alpha, \delta)$$

$$\hat{x}_3[k+1] = \hat{x}_3[k] - \beta_3 \text{fal}(\epsilon, \alpha/2, \delta)$$

(2.36)
where the function \( \text{fal}(\epsilon, \alpha, \delta) \) is defined as

\[
\text{fal}(\epsilon, \alpha, \delta) = \begin{cases} 
|\epsilon|^{\alpha}\text{sign}(\epsilon) & \text{if } |\epsilon| > \delta, \\
\epsilon\delta^{\alpha-1} & \text{if } |\epsilon| \leq \delta,
\end{cases}
\]  

(2.37)

where \( \beta_1, \beta_2, \beta_3 \) are the observer gains. The parameters \( \alpha \in [0, 1) \) and \( \delta \) are design parameters. Han [6] mentions that there exist several ways to choose the observer gains and suggests e.g.

\[
\beta_1 = 1, \quad \beta_2 = \frac{1}{2T^{0.5}}, \quad \beta_3 = \frac{2}{25T^{1.2}}.
\]  

(2.38)

It is also possible to simplify the implementation by replacing the nonlinear function \( \text{fal}(\epsilon, \alpha, \delta) \) with \( \epsilon \). Then the gains can be chosen as

\[
\beta_1 = 1, \quad \beta_2 = \frac{1}{3T}, \quad \beta_3 = \frac{1}{32T^2},
\]  

(2.39)

note though that the use of the nonlinear feedback function can sometimes result in significantly better results [6].
Chapter 3

Simulations

In order to draw conclusions regarding the suitability of the methods presented in Chapter 2, simulations were performed. This chapter is organized as follows; Firstly, a model overview is presented followed by a description of the test cases as well as the measure of performance. Secondly, the theoretical performance of the existing controller design is discussed. Finally, the simulation results are presented and the performance of the different methods is evaluated.

3.1 Simulation Model and Set-up

The model of the Y-wagon used for simulation was developed in MATLAB SIMULINK® by Amann and presented in [1]. The model includes an improved friction model, cogging from the motor and limitations on maximum voltage that can be provided. Process and measurement noise was added to the simulation model to make it more realistic.

A schematic simulation set-up is shown in Figure 3.1, where one can observe that the output from the model $y(t)$ is in continuous time. The signal is sampled, quantized, and contaminated with white Gaussian noise. Each discrete position measurement $y[k]$ is sent to the controller which calculates the current needed to follow the reference trajectory. Thereupon the control signal $u[k]$ is delayed, which represent the computational delay in the real case. The delayed control signal is made continuous through a zero-order hold, meaning that the signal is held constant during each sampling period. Likewise, the
control signal is exposed to white Gaussian noise before it enters the model of the system.

This model was used in order to draw conclusions regarding the controller ability to suppress disturbances, estimate the true velocity and track the reference trajectory. For that purpose the model was considered to be good enough, reflecting the dominating features of the true system.

### 3.1.1 Test Cases

Simulations of different movements were performed. The standard movements of the axis in a pick and place machine are listed below.

- Short point-to-point movement 20-100 mm
- Long point-to-point movement 1-1.1 m
- Constant velocity movement

In normal operation, the Y-wagon can accelerate up to 10 m/s² with a speed up to 2 m/s. The reference trajectories are designed such that maximum acceleration is obtained in point-to-point movements.
3.1.2 Measure of Performance

The potential of each observer design will be compared with a MATLAB SIMULINK® implementation of the existing controller design described in Section 2.2. This will correspond to the benchmark performance. The tracking performance can be measured by computing the difference between the position of the wagon and the reference position. Furthermore, the observer’s ability to estimate the true velocity can be evaluated by calculating the difference between the real and estimated velocity. The following definitions for the errors will be used in this chapter.

**Tracking error:** The difference between the actual position and the reference position, denoted $e$.

**Velocity estimation error:** The difference between the real and estimated velocity, denoted $\hat{e}_2$.

For each movement, the maximum absolute error (the worst case) and the RMS error (mean deviation) was recorded and compared with the benchmark performance. Furthermore, the velocity estimation should be sufficiently smooth, to prevent chatter in the control signal.

The focus of this study was to assess how well the proposed methods manage to estimate the velocity and track the reference trajectory by suppressing the impact of disturbances, computational delay, and noise. The controller is not tuned to achieve minimal overshoot and fast settling time. Instead, the study is focused on the controller’s ability to track the reference and estimate the velocity during the whole movement.

3.2 Simulation Results

In this section, the simulation results are presented followed by an analysis and discussion of the methods presented in Section 2.4. To make a fair comparison, the controller poles were kept in the same location when simulating the movements.

Unfortunately, a stable implementation of the tracking differentiator described in Section 2.4.3 could not be attained. One issue was the numerous design parameters that had to be set randomly because no information on how to choose them was presented in the article. The adaptive windowing technique men-
tioned in Section 2.4.4 was concluded to be not feasible for this application. Averaging over the two latest velocity estimates resulted in large oscillations, which limits the window size and results in noisy estimations. Notice that the POKF described in Section 2.4.6 was not simulated, because the idea of combining Kalman filtering with disturbance estimation was brought in rather late in the project.

Simulations were performed for several movements described in Section 3.1.1. It was concluded that the performance did not vary significantly when comparing different movements. For that reason, only plots showing 100 mm point-to-point movements are presented in this section. The results from constant velocity movements are shown and discussed in Section 3.2.7.

### 3.2.1 Benchmark Performance

The tracking as well as the velocity estimation performance obtained with the motion control system described in Section 2.2 is shown in Figure 3.2. The observer used for estimation of the velocity will be referred to as the One-step Predictor (OSP), which is presented in detail in Section 2.2.3. By studying the graph in Figure 3.2b one can observe that the magnitude of the velocity error is initially large during the acceleration. Further, the error magnitude decreases until the deceleration phase is reached, where it once again increases.

![Figure 3.2: Simulation of 100 mm point-to-point movement using a one-step predictor.](image_url)
3.2.2 One-step Predictor with Central Difference Approximation

In Section 2.4.1, the idea of combining a one-step prediction observer and finite difference approximation, or the OSPCD, in order to improve the accuracy of the estimate was developed. Initial results showed that the simple modification reduced both the tracking error and the estimation error by half or more compared to the benchmark performance. However, the estimate was not smooth enough even though it was less noisy compared to the backward difference approximation (2.17). The chatter on the estimate was reduced by placing the continuous time observer poles closer to the origin, making the observer slower, yet less sensitive to noise. The evolution of the tracking and estimation error is shown in Figure 3.3.

![Figure 3.3: Simulation of 100 mm point-to-point movement using a OSPCD.](image)

3.2.3 Direct Observer

Instead of predicting future outputs using the one-step predictor, the direct observer was implemented to estimate the states. The proposed observer includes the most recent position measurement in the estimation, which should shorten the delay, see Section 2.4.2. The observer poles of the DO were placed in the same location as for the OSP. In Figure 3.4 the tracking and estimation error is shown.
3.2.4 Kalman Filter

The Kalman filter aims to reduce the variance of estimation error while maximizing the accuracy of the estimate. The observer dynamics are presented in Section 2.4.5. The parameter values $r, q$ represent the variance of measurement and process noise. In a simulation, the noise characteristics are known making it easy to choose the parameter values which were set to $r = 10^{-7}$ and $q = 0.01$. A simulation of point-to-point movement with the Kalman filter as the state estimator is presented in Figure 3.5.

Figure 3.4: Simulation of 100 mm point-to-point movement using a direct observer.

Figure 3.5: Simulation of 100 mm point-to-point movement using a Kalman filter.
3.2.5 Perturbation Observer

In order to capture the effect of disturbances, the perturbation observer described in Section 2.4.6 was tested. The state observer poles were kept in the same location as for the one-step predictor however, there is an additional parameter that had to be tuned, namely the cutoff frequency of the low-pass filter. One should keep in mind when choosing the cutoff frequency, that it is desired to filter out the high-frequent measurement noise while including as much as possible of the low-frequent disturbance. Different values were tested, too high frequencies resulted in an unstable observer. On the other hand, too low values yielded to poor performance. Hence, the cutoff frequency was at the end set to $f_c = 70$ Hz. The progression of the estimation and tracking error using the modified controller is shown in Figure 3.6.

![Figure 3.6: Simulation of 100 mm point-to-point movement using a perturbation observer.](image)

(a) Tracking error  
(b) Velocity estimation error

3.2.6 Extended State Observer

An alternative disturbance observer to the perturbation observer is the ESO presented in Section 2.4.6. The observer gains were chosen as suggested in [6] which resulted in a stable observer for a wide range of values on $\alpha$ and $\delta$. The tracking and estimation error is presented in Figure 3.7 were $\delta = T$ and $\alpha = 0.69$. 
3.2.7 Evaluation and Discussion

In this section the performance of the alternative observer designs are compared with each other when considering typical movements for a pick and place machine, see Section 3.1.1. Furthermore, the differences between the methods are discussed.

The tracking and estimation errors obtained when using different observer designs mentioned in Section 3.2 are presented in Table 3.1. One can observe that, in simulation, the tested methods manage to reduce both the estimation and tracking error when comparing to the benchmark performance. Moreover, significant improvement was noted when using a OSPCD for the velocity estimation. Still, the velocity estimate may still not be smooth enough for practical implementation, which is further tested and discussed in Chapter 4.

Table 3.1: Tracking and estimation error when simulating a 100 mm point-to-point movement using different velocity estimation methods.

<table>
<thead>
<tr>
<th></th>
<th>OSP</th>
<th>OSPCD</th>
<th>DO</th>
<th>KF</th>
<th>PO</th>
<th>ESO</th>
</tr>
</thead>
<tbody>
<tr>
<td>$e$ RMS [$\mu$m]</td>
<td>1.4</td>
<td>0.6</td>
<td>1.3</td>
<td>0.9</td>
<td>0.9</td>
<td>0.9</td>
</tr>
<tr>
<td>$e$ Max. [$\mu$m]</td>
<td>5.8</td>
<td>3.2</td>
<td>5.3</td>
<td>4.2</td>
<td>5.1</td>
<td>5.3</td>
</tr>
<tr>
<td>$\tilde{e}_2$ RMS [mm/s]</td>
<td>1.2</td>
<td>0.3</td>
<td>1.1</td>
<td>0.5</td>
<td>0.3</td>
<td>0.2</td>
</tr>
<tr>
<td>$\tilde{e}_2$ Max. [mm/s]</td>
<td>2.1</td>
<td>0.8</td>
<td>1.8</td>
<td>1.0</td>
<td>1.5</td>
<td>1.3</td>
</tr>
</tbody>
</table>

Figure 3.7: Simulation of 100 mm point-to-point movement using an extended state observer.
The Kalman filter was able to increase the accuracy of the velocity estimate without inducing unnecessary chatter on the input signal when properly tuned. Nonetheless, for each sampling period the covariance matrix of the estimation error, as well as the Kalman gain, have to be recalculated; Which in turn increases the computational time in practical applications.

Considering the two disturbance observers (PO and ESO), an additional state was introduced to represent the estimated disturbance as seen in Figure 3.8a. One can observe that both the ESO and PO manage to capture similar disturbance characteristics, even though the estimate from the ESO is noisier. Keep in mind that additional noise is usually undesired in practice. The magnitude of the disturbance is largest during acceleration and deceleration, similar to the velocity estimation error of the OSP seen in Figure 3.2b. The disturbance estimation is later used by the state observer part of the ESO and PO in order to improve the estimation of the velocity as seen in Figure 3.8b.

![Figure 3.8: The estimated disturbance and the velocity estimation error obtained with the two disturbance observers during a simulation of a point-to-point motion.](image)

In general, a disturbance observer aims to estimate the lumped disturbance caused by e.g. unmodeled dynamics of deterministic nature. The Kalman filter, on the other hand, aspires to estimate the states from a series of measurements that are contaminated with stochastic disturbance, e.g. sensor noise. The different approaches are compared in Figure 3.9 and Table 3.1.
As mentioned earlier, the cogging phenomenon affects the dynamics more at low speed, making it more difficult to control. For that reason, it is of high interest to compare the performance of the different observers when the aim is to keep the wagon at constant low velocity. The results can be viewed in Table 3.2. Note that the OSPCD, KF, PO and ESO manage to reduce the velocity estimation error significantly compared to the benchmark performance.

In Figure 3.10 the effect of disturbance estimation and compensation is shown. The perturbation observer seen in Figure 3.10b, as well as the extended state observer manage to increase the accuracy of the velocity estimate while keeping it quite smooth.

Table 3.2: Tracking and estimation error when simulating a constant speed motion of 0.02 m/s.

<table>
<thead>
<tr>
<th></th>
<th>OSP</th>
<th>OSPCD</th>
<th>DO</th>
<th>KF</th>
<th>PO</th>
<th>ESO</th>
</tr>
</thead>
<tbody>
<tr>
<td>e RMS [µm]</td>
<td>0.5</td>
<td>0.3</td>
<td>0.5</td>
<td>0.3</td>
<td>0.5</td>
<td>0.4</td>
</tr>
<tr>
<td>Max. [µm]</td>
<td>4.7</td>
<td>2.9</td>
<td>4.5</td>
<td>3.2</td>
<td>4.1</td>
<td>4.2</td>
</tr>
<tr>
<td>( \tilde{e}_2 ) RMS [mm/s]</td>
<td>2.2</td>
<td>0.2</td>
<td>1.9</td>
<td>0.9</td>
<td>0.2</td>
<td>0.1</td>
</tr>
<tr>
<td>Max. [mm/s]</td>
<td>2.5</td>
<td>1.0</td>
<td>2.3</td>
<td>1.2</td>
<td>1.5</td>
<td>1.3</td>
</tr>
</tbody>
</table>

Figure 3.9: Simulation of 100 mm point-to-point movement using a perturbation observer and a Kalman filter.

Until now, only the point-to-point movement was considered. Thus, one may wonder how well the alternative controller designs manage to control the wagon at a constant velocity.
The direct observer has similar performance as the one-step predictor when tested on the cases presented in Section 3.1.1, which is expected due to the similarities of the two approaches.

In conclusion, the simulation results indicate that the proposed methods are able to reduce the velocity estimation error and improve the tracking performance. Thus, the alternative observer designs were implemented in the embedded code to test if the methods have potential in a practical application.
Chapter 4

Experiments

In this chapter, the effect of the alternative velocity estimation methods is further analyzed. At first, a description of the test rig is presented followed by a specification of the methods to measure and compare performance. Thereafter, the experimental results for typical movements are presented for the purpose of performance comparison between the observer designs.

4.1 Test Rig

The test rig used in the experiments is a Y-wagon from a MY200 machine seen in Figure 4.1, which in total weighs about 100 kg. The wagon was loaded with approximately 30 kg in order to resemble the conditions of a typical Y-wagon. Note that the performance of the test rig was not optimized, meaning that the controller poles were not changed from what is used in the embedded code today. The purpose of the project was to compare different controller designs and how they may affect the motion control performance rather than optimizing the performance of the test rig.

4.2 Test Cases

Movements performed in a pick and place machine are mainly point-to-point movements i.e. moving from one point to another, as fast as possible. In some cases, constant velocity movements are desired e.g. during initialization of the
machine. It is generally harder to control the wagon at low speed due to the fewer number of encoder ticks per sampling time. A summary of the test cases used for comparison of performance are listed below:

- Short point-to-point movement 20 mm and 100 mm
- Long point-to-point movement 1 m
- Constant velocity 0.02 m/s and 0.2 m/s

It is of high interest to test the robustness of the proposed methods where it is desirable that the controller remains stable and that the performance is not affected too much. Thus, in order to test how sensitive a method is to change in parameter values, the total mass of the conveyor $m$, and the motor force constant $K_f$ were altered in the embedded code from their initial values, as stated below:

- Change of mass $m$ with $\pm$ 15%  
- Change of motor force constant $K_f$ with $\pm$ 5%

The choice of parameters was not arbitrary in fact, the loaded mass on the conveyor can sometimes vary up to $\pm$30%. Furthermore, the specifications of the linear motor, including $K_f$ can vary within $\pm$10% [7].
4.3 Measure of Performance

In point-to-point movements, certain features are desired. The over- and undershoot should be as small as possible. In addition, the settling time should be minimized as well. The definitions of the performance measures for point-to-point movements are visualized in Figure 4.2.

![Graphical illustration of the over- and undershoot as well as the settling time.](image)

The ability to keep the wagon at constant speed was measured by studying the velocity error.

**Velocity error:** The difference between the reference and the estimated velocity.

The maximum absolute error, as well as the RMS error, were used to compare the performance of constant velocity movements.

4.3.1 Parameter Tuning

Several tests were performed with each observer design in order to choose the appropriate parameter values. The feedback controller poles were kept in the same location initially to reduce the number of tuning parameters.
One-step Predictor with Central Difference Approximation

There were doubts whether the velocity estimate obtained with this method is smooth enough to be applicable in practice. Without any tuning, the control signal was too noisy as seen in Figure 4.3b, which resulted in an unpleasant sound from the motor, especially at low speed. By placing both the continuous-time observer poles and the feedback controller poles closer to the origin in the complex plane, and thus reducing the observer- and controller gains, the chatter on the input signal was reduced to acceptable levels. Unfortunately, at the expense of the control performance, see Section 4.4.

![Settling](a) Settling  ![Control signal](b) Control signal

Figure 4.3: A 100 mm point-to-point movement run with OSPCD and OSP.

Direct Observer

The poles of the direct observer were placed in the same locations as for the one-step predictor. More tuning of this observer was concluded unnecessary, due to the smooth velocity estimate and acceptable motion control performance.

Kalman Filter

Starting with the parameter values $r$ and $q$ used in the simulation, the parameters were tuned until desired performance was attained. Keeping in mind that the parameters $r$, $q$ have a physical interpretation, the values were in the end chosen as $q = 0.006$ and $r = 10^{-8}$. 
Perturbation Observer

The state estimator gains were set to the same values as for the OSP. The cutoff frequency of the low-pass filter had to be tuned as well. Starting with the value used in the simulation, the frequency was increased until acceptable tracking and noise attenuation was obtained, thus $f_c = 130$ Hz.

Perturbation Observer with Kalman filtering

During the experimental phase of the project, the idea of combining the perturbation observer and a Kalman filter for state estimation was developed, see Section 2.4.6 for a more detailed description. It was tuned accordingly and the final parameters values were chosen to $q = 0.002$, $r = 8 \cdot 10^{-8}$, and the cutoff frequency $f_c = 120$ Hz.

Extended State Observer

The extended state observer has many parameters that have to be tuned, see Section 2.4.6. At first, the simplified version of the extended state observer

![Figure 4.4: Settling of a 1 m movement using different versions of the extended state observer.](image)
was implemented and tuned. Later on, the version with nonlinear feedback was implemented as well, which introduced two additional parameters. After tuning, the resulting control performance of the two methods was very similar, see Figure 4.4.

### 4.4 Results

In this section, the results collected during the experimental phase of the project are summarized. First and foremost the results for the point-to-point movements are presented, followed by constant velocity movements. Finally, the results from the robustness test described in Section 4.2 are included as well.

#### 4.4.1 Point-to-point

The results for the different observer designs are shown in Table 4.1. The performed point-to-point test cases are described in Section 4.2.

<table>
<thead>
<tr>
<th></th>
<th>OSP</th>
<th>OSPCD</th>
<th>DO</th>
<th>KF</th>
<th>PO</th>
<th>POKF</th>
<th>ESO</th>
</tr>
</thead>
<tbody>
<tr>
<td>20 mm</td>
<td>Overshoot [μm]</td>
<td>10</td>
<td>15</td>
<td>10</td>
<td>6</td>
<td>9</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>Undershoot [μm]</td>
<td>23</td>
<td>26</td>
<td>22</td>
<td>13</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Settling time [ms]</td>
<td>37</td>
<td>48</td>
<td>37</td>
<td>29</td>
<td>50</td>
<td>45</td>
</tr>
<tr>
<td>100 mm</td>
<td>Overshoot [μm]</td>
<td>8</td>
<td>10</td>
<td>7</td>
<td>5</td>
<td>9</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Undershoot [μm]</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>10</td>
<td>7</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Settling time [ms]</td>
<td>38</td>
<td>40</td>
<td>36</td>
<td>36</td>
<td>63</td>
<td>46</td>
</tr>
<tr>
<td>1 m</td>
<td>Overshoot [μm]</td>
<td>6</td>
<td>9</td>
<td>6</td>
<td>5</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Undershoot [μm]</td>
<td>13</td>
<td>14</td>
<td>13</td>
<td>10</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>Settling time [ms]</td>
<td>38</td>
<td>40</td>
<td>36</td>
<td>30</td>
<td>52</td>
<td>35</td>
</tr>
</tbody>
</table>

The dynamics of the settling process for point-to-point motion when using different velocity estimation methods are shown in Figures 4.6-4.7.
Figure 4.5: Settling of a 20 mm movement using a OSP, OSPCD and DO.

Figure 4.6: Settling of a 20 mm movement using a OSP, KF and POKF.
4.4.2 Constant Speed

In some cases it is desired that a movement of an axis in a pick and place machine is held at constant speed. Experiments were performed and the velocity error defined in Section 4.3 was calculated and displayed in Table 4.2.

Table 4.2: Velocity error of constant speed motion using different observers.

<table>
<thead>
<tr>
<th>Speed</th>
<th>Error</th>
<th>OSP</th>
<th>OSPCD</th>
<th>DO</th>
<th>KF</th>
<th>PO</th>
<th>POKF</th>
<th>ESO</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.02 m/s</td>
<td>RMS [mm/s]</td>
<td>0.9</td>
<td>0.3</td>
<td>0.9</td>
<td>0.5</td>
<td>0.2</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td></td>
<td>Max. [mm/s]</td>
<td>2.1</td>
<td>1.6</td>
<td>2.1</td>
<td>1.6</td>
<td>0.8</td>
<td>0.5</td>
<td>0.8</td>
</tr>
<tr>
<td>0.2 m/s</td>
<td>RMS [mm/s]</td>
<td>0.8</td>
<td>0.3</td>
<td>0.9</td>
<td>0.5</td>
<td>0.2</td>
<td>0.1</td>
<td>0.2</td>
</tr>
<tr>
<td></td>
<td>Max. [mm/s]</td>
<td>2.1</td>
<td>1.6</td>
<td>2.1</td>
<td>1.7</td>
<td>0.8</td>
<td>0.5</td>
<td>0.5</td>
</tr>
</tbody>
</table>

The velocity estimation obtained with the tested observers is shown in Figures 4.8 - 4.10.
Figure 4.8: Estimated velocity with OSP, OSPCD and DO.

Figure 4.9: Estimated velocity with OSP, KF and POKF.
Figure 4.10: Estimated velocity with OSP, PO and ESO.
4.4.3 Robustness Test

In practical applications, it is required that the motion controller is robust. A description of the robustness test is presented in Section 4.2. The test was performed on the two observer designs that showed the best results in the earlier experiments, namely the KF and the POKF. The test was carried out for several movements however, no substantial difference was noted between them. Hence, the results for 100 mm point-to-point movement are presented in Figures 4.11 & 4.12.

![Graph 1](image1)

(a) Change in the motor force constant $K_f$.

![Graph 2](image2)

(b) Change in total mass $m$.

Figure 4.11: Robustness test of a Kalman filter.

![Graph 3](image3)

(a) Change in the motor force constant $K_f$.

![Graph 4](image4)

(b) Change in total mass $m$.

Figure 4.12: Robustness test of a Perturbation observer with Kalman filtering.
Chapter 5

Discussion

In the following chapter, the results obtained through simulations and experiments will be further analyzed and discussed. Potential error sources and their effect on the results will be covered as well.

The numerical results of the point-to-point experiments presented in Table 4.1 show an increase in the settling time for the majority of the tested observers compared to the benchmark performance of the OSP. In fact, during the parameter tuning of each observer, it turned out to be quite difficult to obtain faster settling without inducing too much chatter on the control signal. Nevertheless, the shortest settling time and therefore the best performance was obtained using a Kalman filter. The reason could be the fact that the Kalman filter takes into account the disturbances of stochastic nature, which are not handled by the other techniques presented in this paper. On the other hand, the drawback of using Kalman filtering is the time-consuming tuning of the parameters \( r \) and \( q \). The exact characteristics of the stochastic disturbances are seldom known, hence one may need to tune the observer for quite some time before desired control is attained. Moreover, a Y-wagon that operates in a machine, is subjected to different kind of disturbances and thus will most likely need retuning. Another drawback is the increased computational time due to the additional calculations, as discussed in Section 3.2.7.

The effect of a change in the parameters \( m \) and \( K_f \) when using a Kalman filter is shown in Figure 4.11. Clearly, the tracking is affected by the change although the settling time is alike. Note though that an increase of the mass \( m \) and the reduction of the motor force constant \( K_f \) causes more chatter in the control signal, which is undesirable in practice.
The majority of the tested velocity estimation techniques manage to decrease the over- and undershoot compared to the OSP, see Table 4.1. Remarkably good results were obtained for the POKF as seen in Figure 4.6. Although, the settling time is slightly longer for short point-to-point movements. The robustness analysis presented in Figure 4.12 show that the controller retains similar performance, which indicates that the method has a capability to cope with uncertainties. In practice, it is not unusual that the wagon is loaded with a mass that can vary up to $\pm 30\%$. Hence, the robustness feature is in fact very useful for this sort of application.

The performance between varying lengths of point-to-point movements should ideally be constant. It is interesting to note in Table 4.1 that the undershoot for the 20 mm movement increased by $\approx 10 \mu m$ for the OSP, OSPCD and DO compared with the 100 mm and 1 m movements. Whereas the KF and POKF have a more similar point-to-point performance as seen in Table 4.1.

The results from constant speed experiments in Table 4.2 show that the performance of 0.02 m/s and 0.2 m/s movements is similar. The worst results were obtained with the direct observer, which is almost identical to the benchmark. However, even though the accuracy was improved when using the OSPCD, still the estimate oscillates around the reference quite much as seen in Figure 4.8.

The Kalman filter did improve the accuracy of the velocity estimate, although a bias error is seen Figure 4.9. Nonetheless, by including a disturbance estimator in the observer (as in the PO, POKF and ESO), smooth estimates with no bias were obtained as seen in Figures 4.9 & 4.10. The numerical values presented in Table 4.2, entail that the POKF is the best observer for constant velocity movements.

The disturbance observers ESO and PO show similar dynamics as seen in Figures 4.7 & 4.10. Nonetheless, there are differences between the two methods: how the observer gains of the state estimator are chosen and also how the disturbance is estimated, see Section 2.4.6 for more details. It is worth to acknowledge that the PO was easier to tune because there is a physical interpretation of the tuning parameters (the observer poles and the cutoff frequency), unlike the ESO where it was hard to attain intuition on how the different parameters affect the performance.

Even though the simulation model is not a perfect replica of the wagon dynamics, still the simulation model is valuable. In Section 3.2.7 the ability of a perturbation observer to remove the bias was shown in Figure 3.10. The
feature was observed during the experiments as well, see e.g. Figure 4.10. Furthermore, the OSPCD method results in noisy estimations which were noted during the simulation phase of the project. Also, the simulation model was helpful for identifying how different parameters affect the motion control, thus facilitating the parameter tuning for the test rig.

Potential error sources that may affect the results are mainly the fact that there was a time limitation regarding the tuning of each observer. The tuning parameters highly influence the overall control performance. Thus, with more tuning of each observer, better performance may be obtained. The implementation of each observer in the embedded code can most likely be improved as well, which in turn could decrease the execution time.

It is assumed in the observer design that the effect of friction force is compensated for through the feedforward controller. However, the test performed on the Y-wagon in [1] indicate that the simple friction model (2.6) is inaccurate especially for low speed. Therefore, it is recommended to improve the current friction model. The actual friction force may be better reflected by the Tustin friction model [17, p.21]. The model includes the characteristics of static friction, dynamic friction and the Stribeck effect which originates at low sliding velocities.

Future work would be to further evaluate the potential of the KF and POKF in an operating pick and place machine. Further improvements on the control performance could be achieved with additional work on the implementation of the new observer design where the objective is to minimize the computational time each sample period. Finally, the new observer could be implemented in the motion control system used for the control of the x-axis as well.
Chapter 6

Conclusions

The proposed methods were tested in both simulations and on real hardware in order to identify their main features. The use of a Kalman filter (KF) is favorable to handle with stochastic disturbances which are present even in larger extent in an operating pick and place machine. Nonetheless, the method requires extensive tuning to achieve good performance.

The robustness and the tracking performance of the control system were enhanced when the Perturbation observer with Kalman filtering (POKF) was used. Furthermore, the method performed really well when movements at constant speed were desired. However, the settling time of point-to-point movements was slightly shorter when using the Kalman filter alone.
Bibliography


