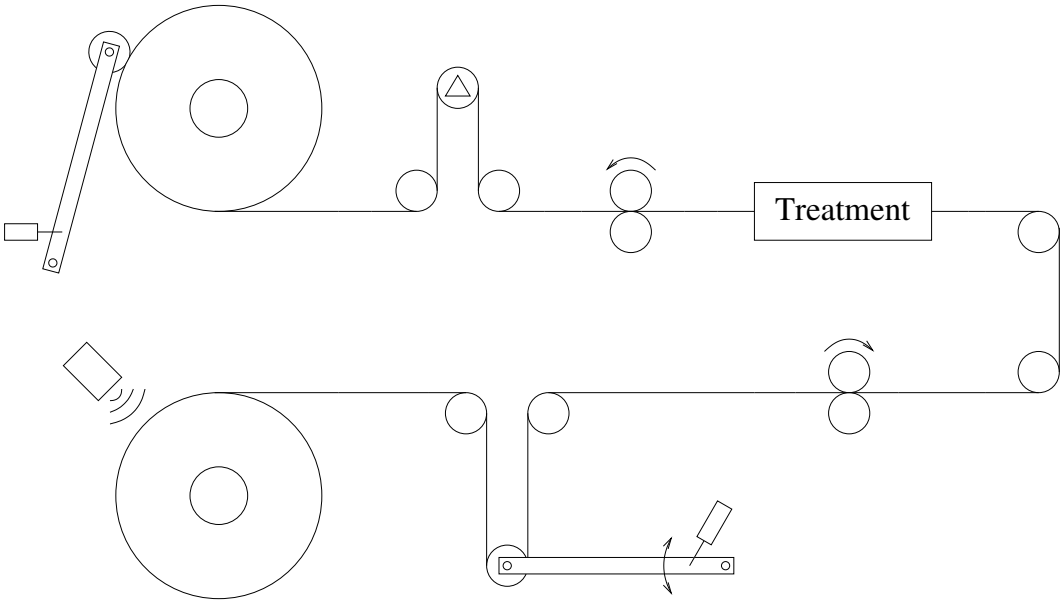
Adaptive Control with Self-Tuning for Center-Driven Web Winders



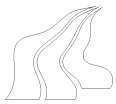
Group members of IAS10-1032a

Jeppe Søndergaard Larsen
Peter Kai Jensen

Master Thesis
Spring 2007

AALBORG UNIVERSITY, DEPARTMENT OF CONTROL ENGINEERING

The Faculty of Engineering, Science, and Medicine
Aalborg University



Intelligent Autonomous Systems

THEME:
Final Thesis

TITLE:
Adaptive Control with
Self-Tuning for Center-Driven
Web Winders

PROJECT PERIOD:
IAS10
Feb. 1. - Jun. 7. 2007

GROUP:
IAS10-1032a

GROUP MEMBERS:
Jeppe Søndergaard Larsen
Peter Kai Jensen

SUPERVISOR:
Roozbeh Izadi-Zamanabadi

NUMBER OF DUPLICATES:
5

NUMBER OF PAGES IN REPORT:
62

NUMBER OF PAGES IN APPENDIX:
28

TOTAL NUMBER OF PAGES:
90

Abstract:
This project deals with the adaptive control of a Center-Driven Web Winder system with dancer feedback.
A model was developed and implemented in Simulink. Several adaptive control methods were evaluated and a modified form of Model Identification Adaptive Control was used.
The automated controller design is integral optimal control, with a basis in the developed model and estimated plant parameters.
The developed adaptive controller was tested on a scale model provided by Danfoss, and it delivered acceptable performance if the control effort weighting parameter is high enough.
Two methods for determining the current winder roll radius and one method for determining total system inertia were also developed and tested.
Both radius estimation methods delivered acceptable performance, but the inertia estimator only worked reliably when there were enough disturbances to excite the dynamics of the system.

Preface

This report is written in the spring semester of 2007 at Aalborg University, at the institute for Electronic Systems, department of Process Control. The topic is Adaptive Control of a Center-Driven Web Winder System, and it is the completing work in the Master Program in Intelligent Autonomous Systems for group 1032a. Associate Professor Roozbeh Izadi-Zamanabadi was the supervisor and Michael Laursen was our contact person at Danfoss.

Citations in the report are marked with [], and are listed alphabetically in the Bibliography at the end of the report. The enclosed CD contains the report in PS and PDF format, as well as the Matlab and Simulink code used for design and implementation of the controller.

A special thanks is extended to Carsten Skovmose Kallesøe at Grundfos for lending us a dSpace rack, to Henrik Rasmussen at Aalborg University for assisting us in the motor laboratory, and to Jakob Stoustrup for his input on various mathematical issues.

Jeppe Søndergaard Larsen

Peter Kai Jensen

Table of Contents

Nomenclature

Symbol	Description	Unit
$\mathbf{A}$	System matrix	–
A, A_1, A_u	Web cross sectional area	m^2
$\mathbf{B}_C$	Controllable part of input matrix	–
$\mathbf{B}_D$	Uncontrollable part of input matrix	–
B_m	Coefficient of friction in drive train	$Nm/rad/s$
B_d	Coefficient of friction in dancer	kg/s
$\mathbf{C}$	System output matrix	–
$\mathbf{C}_d$	System output matrix for just the dancer	–
C	Web dampening coefficient	Pa/s
d	Dancer position	m
d_{ref}	Dancer position reference	m
E	E-modulus of the web	Pa
F_w	Force caused by dancer roll weight	N
F_L	Spring loading of the dancer at its minimum position	N
g	Force of gravity	m/s^2
G	Paper thickness	m
J	Total system inertia reflected to motor	kgm^2
$\mathbf{K}, \mathbf{K}_0$	Optimal gain	–
K_1	Integral gain	–
K_d	Spring constant of dancer	N/m
$\mathbf{L}_o$	Observer gain	–
L, L_u	Web length	m
L_N	Nominal web length, with dancer	m
m_d	Mass of dancer	kg
N	Gear ratio	–
$\mathbf{Q}_c$	State weighting matrix	–
$\mathbf{Q}_o$	State noise covariance matrix	–
$\mathbf{R}_c$	Control input weighting matrix	–

Symbol	Description	Unit
$\mathbf{R}_o$	Measurement noise covariance matrix	–
R	Radius of the winder roll	m
R_{core}	Radius of the winder roll core	m
T, T_1	Web tension	N
T_{ref}	Web tension reference	N
$\mathbf{u}_C$	Controllable input vector	–
$\mathbf{u}_D$	Measured disturbance input vector	–
V	Web surface velocity at the winder roll	m/s
V_1	Line speed	m/s
V_d	Dancer velocity	m/s
$\mathbf{x}$	System state vector	–
x_I	Integrator state	–
$\mathbf{y}$	System output vector	–
$\varepsilon, \varepsilon_1$	Strain of the web	–
ρ, ρ_1, ρ_u	Density of the web	kg/m^2
σ	Web stress	Pa
τ	Torque generated by motor	Nm
ω	Winder motor velocity	rad/s
ω_1	Unwinder motor velocity	rad/s
γ	Control effort scaling factor	–
φ	Signal vector for system identification	–
θ	Parameter vector for system identification	–

List of Acronyms

Acronym	Definition
ARE	Algebraic Riccati Equation
ARX	AutoRegressive with eXogeneous input
CDWW	Center-Driven Web Winder
LQE	Linear Quadratic Estimator
LQR	Linear Quadratic Regulator
LS	Least Squares
MIAC	Model Identification Adaptive Control
MRAC	Model Reference Adaptive Control
PID	Proportional Integral Derivative
PI	Proportional Integral
PWM	Pulse Width Modulation
RLS	Recursive Least Squares
TML	Torque Minor Loop
WTS	Web Transport System

Chapter 1

Introduction

1.1 Web Transport System

A Web Transport System (WTS) is used in many industries to move and process continuous strips of materials, such as metals, textiles, and paper. These materials are generically referred to as *web*, and it is the purpose of the WTS to maintain appropriate tension zones throughout the processing.

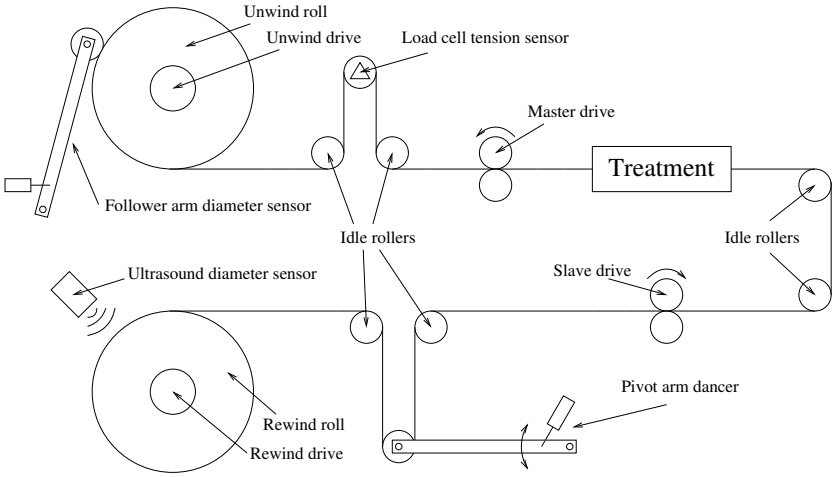


Figure 1.1: Generic WTS.

Figure 1.1 shows a generic WTS with examples of the various kinds of sensors and drives that are used to control the web tension throughout the process. The location between two drives will have a tension determined by the material and the

drives. Various treatments of the web may require different tensions for optimal results. Depending on the material, there may be more or less strict requirements to the tension control. For instance, textiles may wrinkle with too little tension, and thin strips of paper or plastic materials may break at too high a tension.

1.2 Center-Driven Web Winder

The focus of this project will be on the control of the Center-Driven Web Winder (CDWW), which is the final step of the process, where the material is rolled up again. Accurate tension control for this section is very important for the final quality of the product, as tension errors here can warp the web, or even crush the inner layers of the roll. To protect against these problems, some CDWW systems use what is known as *taper tension*, to gradually decrease the web tension with increasing roll diameter.

The tension in this section of the WTS is also one of the more difficult to control, since the plant dynamics are constantly changing. Specifically the roll diameter and inertia may increase by an order of magnitude or more. With a static controller design, this would require a design for the worst-case situation, which will be sub-optimal for large parts of the winding process.

1.3 Goals

In order to improve usability for the operator, it is desirable that the adaptive control should be able to automatically determine as many parameters as possible. Certain parameters, such as roll dimensions and weights, can usually be considered known and will be input by the operator. Web thickness, material density, and elasticity are more difficult, or even impossible, to measure. Once the web winder is running, the changing roll diameter will affect the dynamics of the system. The goal of the adaptive control is to ensure stable tension control, given this changing parameter.

Chapter 2

Problem Analysis

This analysis includes a description of the plant, provided from Danfoss, followed by an overview of some of the existing controller structure in use today. Based on this background a controller structure will be selected, which will create the basis for the modeling part in the present project.

2.1 Test Platform Description

A WTS model has been provided from Danfoss, which will be used in this project for developing and testing controller algorithms. A roll of paper is also provided for use as the web material. The WTS is composed of winder rolls, motors, sensors and frequency converters which will be described in the following.

2.1.1 Dancer

The WTS is equipped with one passive dancer used for measuring the tension in the web, as depicted in Figure 2.2. In this plant the passive dancer is a pivot arm connected to a potentiometer, whose output corresponds to the position of the dancer. Passive dancers provides a good tension feedback for low speed web lines. They are, however, limited in a wide range of dynamic condition in high speed web lines [Tre99].

An active dancer is driven by an actuator to control the translational velocity. This type of dancer gives more flexibility for attenuating disturbances and for maintaining lower tension fluctuations [Tre99].

The dancer did not originally have any springs attached and the potentiometer

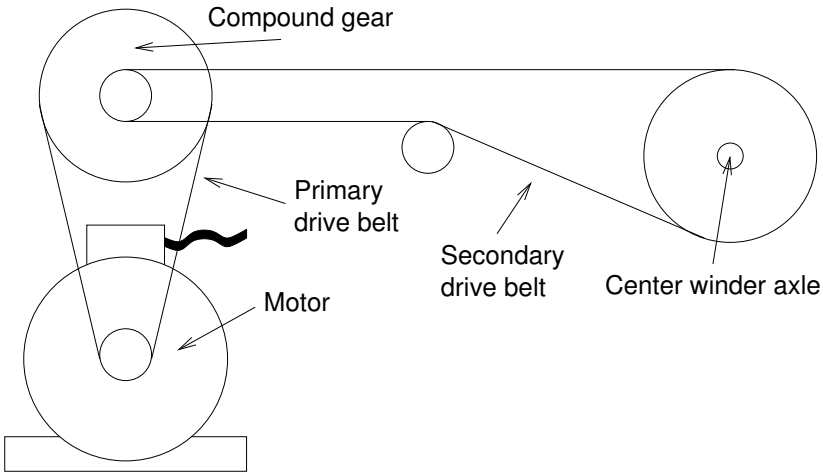


Figure 2.1: Transmission from motor to winder roll.

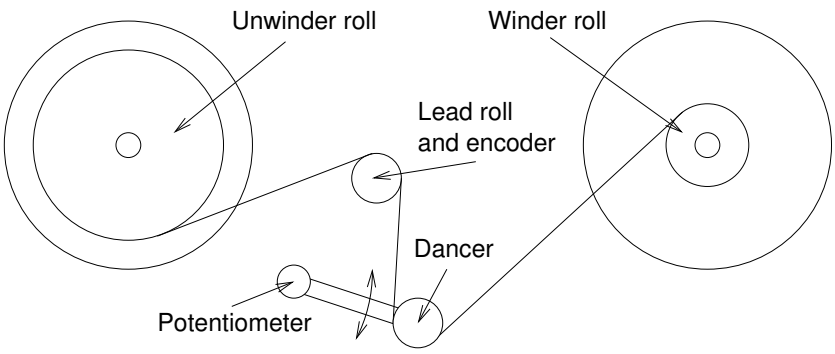


Figure 2.2: Web path from unwinder to winder roll, through dancer.

output was in a relatively narrow range. To compensate, an amplification circuit was added to the potentiometer feedback and springs were attached to the dancer roll axle. The hardware involved is described in Appendix A.

2.1.2 Motor

Two 0.75 kWh ATB motors drive the unwinder and winder rolls through gearings. Figure 2.1 shows the winder roll side. The unwind roll side is identical, but mirrored. An encoder is used for counting the revolutions of the winder motor, while the encoder on the unwinder motor is unconnected. One Danfoss frequency converter is used for controlling each motor.

2.1.3 Frequency Converter

One Danfoss frequency converter controls each of the two motors. The frequency converters are configured as master/slave. The master has a programmable MCO305 which also takes the encoder outputs as inputs. The MCO305 can be programmed with its own macro based programming language. It is also possible to control the frequency converters through analog reference inputs.

2.1.4 Gearing

Two gears are used between the motors and the winder and unwinder rolls, as depicted in Figure 2.1. The left side motor drives the right side roll and vice versa. From the left side motor to the right side roll axle, the gearing ratio is 10.75:1, while the other side is geared at 9:1.

2.1.5 Line Speed Sensor

The lead roll depicted in Figure 2.2 is connected with an encoder, so the line speed of the web can be measured. The roll is equipped with a rubber surface to prevent slipping.

2.1.6 dSpace

An embedded dSpace platform and interface rack is used to connect Simulink and the test plant. Two encoder feedbacks are used to measure winder motor speed

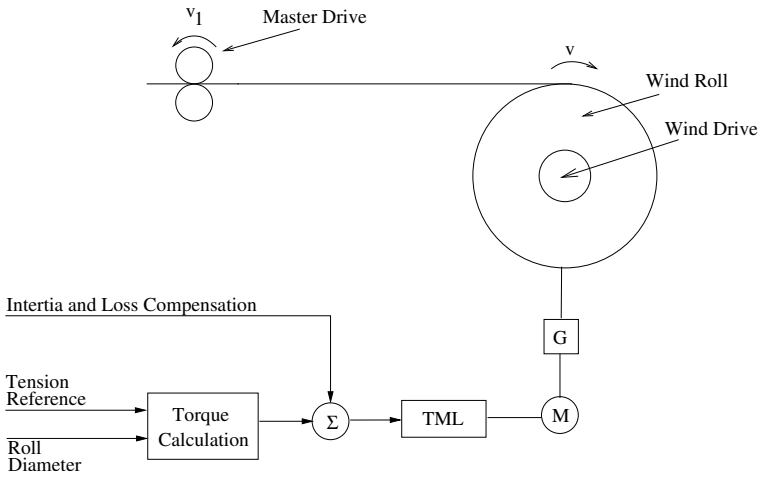


Figure 2.3: The torque at the winder is controlled in an open loop, based on a tension reference and the roll diameter.

and line speed, and one analog feedback is used for measuring the dancer position. The analog reference outputs are generated with the dSpace PWM output. This is filtered by the frequency converters to set an analog level proportional to the PWM period.

2.2 Existing Control Structures

There are different control strategies, in use today, for controlling a WTS. The choice of which one to use, depends on the plant setup and the requirements to the systems behavior and maybe the final product. In this section five of them are discussed from a control point of view, starting with the least advanced method. Using the previously plant description one of them is selected, later in the problem analysis, and used for the following plant modeling.

This section is primarily based on theory from [Liu99] and [Dam04].

2.2.1 Torque Controlled Winder

In the first controller structure, the torque at the winder roll is controlled in an open loop in order to obtain a correct web tension, as shown in Figure 2.3. The torque is calculated according to a tension reference and a roll diameter, and is fed into the Torque Minor Loop (TML) block, which compensates for the inertia and

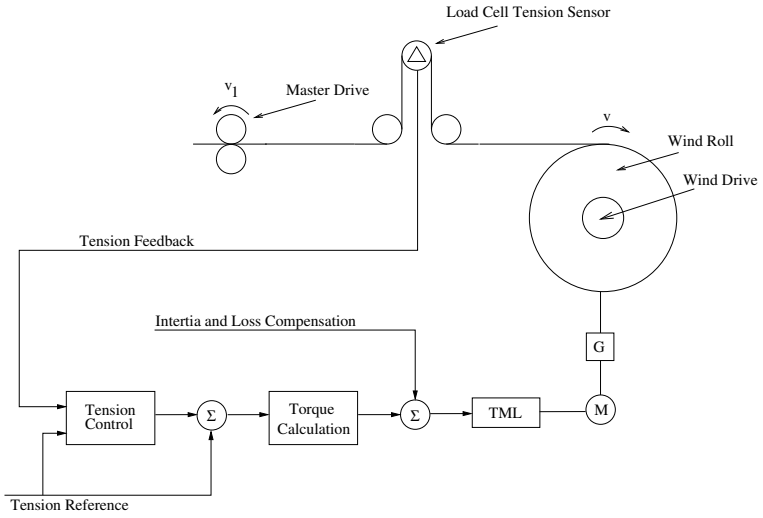


Figure 2.4: The torque at the winder is controlled based on feedback from a load cell measuring the web tension and a tension reference.

the torque loss in the system. The compensated torque is then used to control the torque at the winder through the motor and the gear.

This control structure is stable if the system has a high natural dampening or if the disturbances occur infrequently. Otherwise, the system become unstable. The natural dampening depends on the friction of the mechanical system, line speed, and web dampening caused by web properties. Such a control structure is only used in systems with low requirements of tension control.

2.2.2 Torque and Tension Controlled Winder

In this closed loop control structure, a load cell is used to measure the force applied to an idler roll due to the tension. The force is used to control the torque at the winder and hence the web tension, as depicted in Figure 2.4. The force is proportional to the wrap angle around the roll, and the lead-in and lead-out idler rolls are used because the wrap angles must not vary when the roll diameter changes, unless this is considered in the model [Dam04]. The tension controller is typically designed as PI or PID whose output signal is based on the tension feedback from the load cell and a tension reference defined by the operator [Liu99]. The controller output is then used for the torque calculations.

When using a PI controller, the crossover frequency needs to be less than the systems frequency with no dampening [Liu99]. This is also denoted as the natural

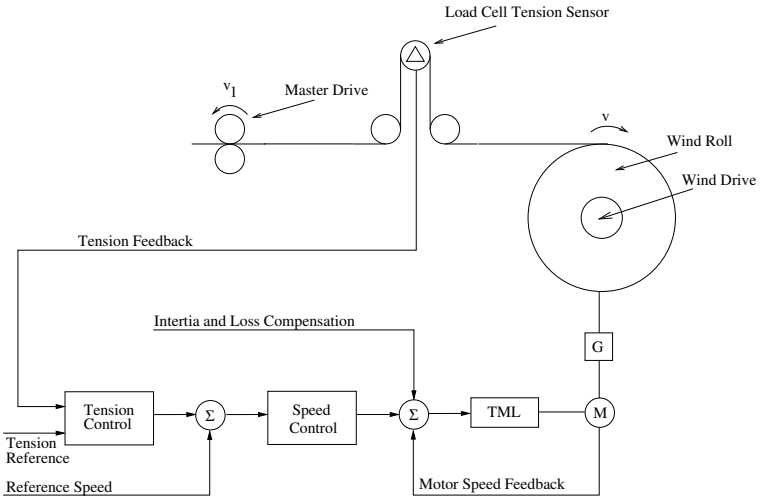


Figure 2.5: The speed at the winder roll is controlled based on a tension reference and a feedback from a load cell measuring the web tension.

frequency and is related to the mechanical configuration of the system, the properties of the web, and the line speed. The natural frequency has its minimum value either when the winder roll is full or empty, depending on the system configuration and the web properties. When the natural frequency becomes smaller, the system response becomes slower, which might cause oscillatory tension if the system poles become too dominating. The effect of the poles, can be dampened by using a derivative term. This adds an additional zero which moves the systems root-locus further to left part of the s-plane, if it is placed correctly. An incorrect placement of the zero might introduce noise to the system or make the system unstable.

2.2.3 Speed and Tension Controlled Winder

In the control structure shown in Figure 2.5, the torque calculation is replaced by a speed loop. The speed loop calculates the control signal based on the actual motor speed, a speed reference, and a tension control signal based on the tension feedback signal. The output of the speed controller sets the speed for the winder through the TML, motor, and the gear.

The dampening of speed loops depends on the natural frequency and the system inertia, which are both changing. Depending on the system configuration, the natural frequency and the dampening decreases when the roll diameter and the inertia increases [Liu99]. Typically, the PI gain is optimized to a certain inertia

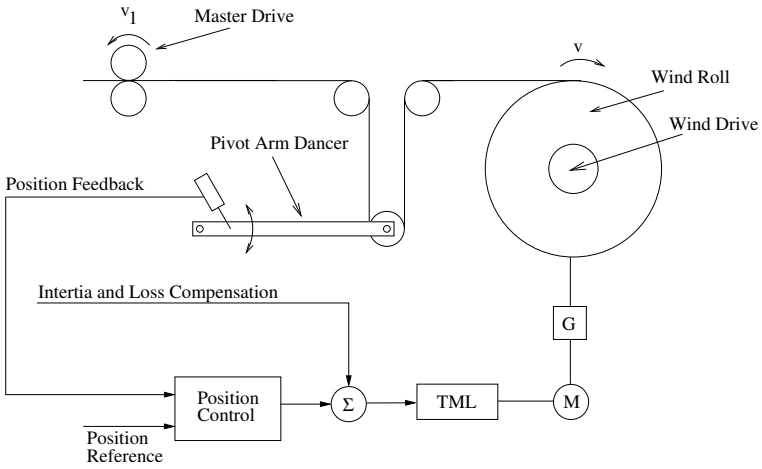


Figure 2.6: The torque at the winder roll is controlled based on the position of a dancer roll and a position reference.

range in which the system response is the best. To get optimized PI gains through the whole process, they should be adapted with the change of inertia.

2.2.4 Torque and Position Controlled Winder

Instead of using a load cell to control the tension, a dancer can also be used, as shown in Figure 2.6. The dancer shown in the figure is a pivot arm dancer which is also used in this project.

There exist different types of dancers, but they all attempt to keep the tension constant by having idler rolls that are loaded in one direction, while the web tends to move these in the opposite direction. The position of the dancer is measured by a sensor, whose output is compared to a position reference given by the operator [Dam04]. The result is used with the inertia and the loss compensation, to calculate the total torque used for controlling the motor and hence the web tension. The tension is also affected by the dancer friction, dancer roll inertia, and other nonlinear dancer elements, which are not considered in this analysis.

To use this controller structure a PID controller is needed as minimum. A PI controller will give an unacceptable response, because the pole is located very close to the origin which makes the position oscillatory. Adding a derivative term gives a more acceptable response, because this moves the root-locus toward the more stable part of the s-plan.

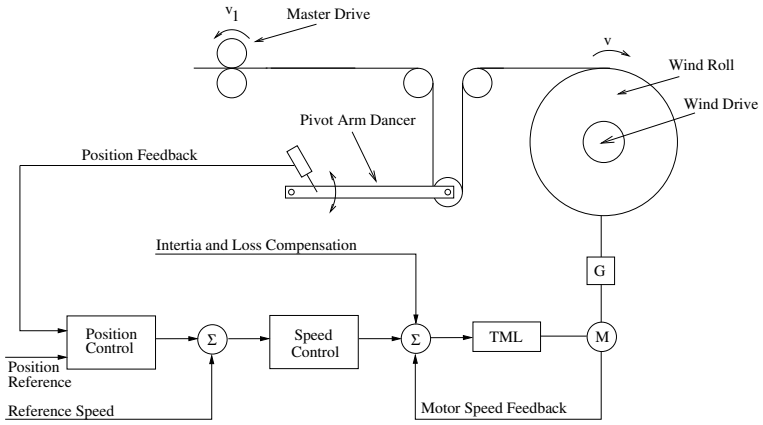


Figure 2.7: The speed at the winder roll is controlled based on a position reference and the position of a dancer roll.

2.2.5 Speed and Position Controlled Winder

The last controller structure, shown in Figure 2.7, is using a speed controller together with a position controller to calculate the correct torque at the winder. The actual position of the dancer and a reference position is used to calculate a control signal. This control signal is used to control the speed of the motor.

Through reasonable design of the speed loop, the poles can be placed far from the imaginary axis, so the controlled plant tends to have the behavior of an integral. The integral eliminates the steady state error, which gives better position response and therefore better control performance.

2.3 Adaptive Control

In control structures, like the previously described, the gain values are often required to be specified for different tension zones, so the tension response can be maintained within an acceptable level during the winding process. The tension is sensitive to changes in the winder roll radius, line speed, and the material. When the tension response changes, the gain values need to be updated. Typically, adaptive control is used for updating the gain values when the system is running, but other methods also exist in use today. This section gives an overview of some of the different methods.

2.3.1 Self-Initializing

To start the process the gains need to be initialized. Manual initializing can often be difficult and time consuming. Wrong parameters might lead to unstable control, which may cause a web break. Therefore, it is better to self-tune the parameters by letting the system oscillate controllably in the start-up phase, to gain information for estimating the gain values. [Liu98]

2.3.2 Manual Adjustment

One simple way of compensating for changing plant dynamics is to have an operator monitor the response of the system and manually adjust controller parameters as needed. This adjustment is typically based on the operators experience, and will usually be adequate but rarely optimal. In some situations, this may still be the most economic solution.

2.3.3 Gain Scheduling

A step up from manual adjustment is gain scheduling, where the operating range of the plant is divided and controllers designed to be optimal for the specific range. An automated system then monitors the progress and switches between gains to match the plant state. The downside is that for more complex plants, the number of regions can grow very large, and the gains will be specific to the web material and plant configuration. At the boundary of the control regions the performance will also be sub-optimal, and unpredicted areas of operation could lead to unstable operation.

2.3.4 Frequency Based

A more advanced adaptive control is to adapt the gains to the frequency. In [Liu98] the gains are found using the natural frequency to which a probing signal, with a specified frequency and a small amplitude, is applied. The resulting response is filtered and used to estimate the new set of gain values. If the new gains give a too dramatic change in the phase margin, the plant becomes unstable and user interruption is needed. Then the system parameters need to be initialized again. [Liu98]

2.4 Scope

In this project the controller will be developed for a plant with a dancer so that the test plant provided from Danfoss will not have to be changed significantly. An adaptive torque-based tension control method will be implemented and the internal dynamics of dSpace and the VLT will not be considered.

2.5 Requirements Specification

The design of the controller takes its starting point in a set of requirements, which also will create the basis for the final test. With suggestions from Danfoss, the requirements for this project are:

1. When ramping V_1 the tension error must be less than 5 %.
2. The tension settling time after a V_1 disturbance must not exceed 2 s.
3. The steady state tension error must not exceed 1 %.
4. The rise time for ramping V_1 from 10% to 90% of its maximum velocity must not exceed 2 s.

2.6 Project Procedure

This section describes the project procedure in the present project, starting with the Research and Analysis. The procedure is shown in Figure 2.8 and is explained stepwise in the following.

2.6.1 Research and Analysis

The first step of the project is to gather information about previous research in the field of both CDWW systems and adaptive control. Then follows an analysis of the test plant provided by Danfoss, to determine what possibilities and limitations exist.

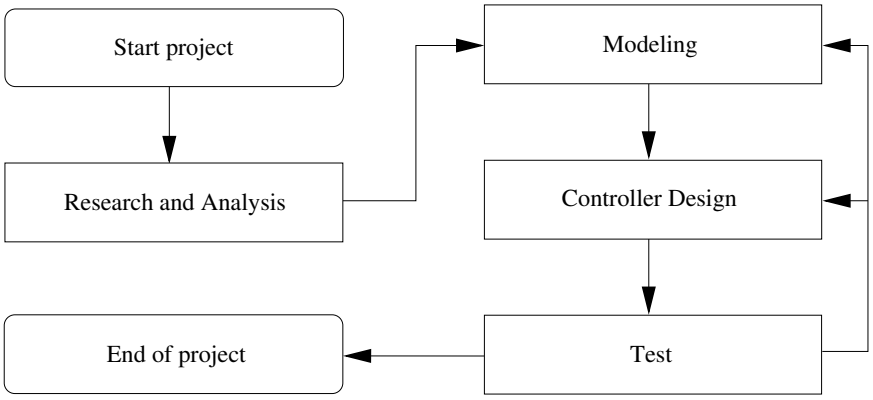


Figure 2.8: Project procedure flowchart.

2.6.2 Modeling

Based on an analysis of the physical plant structure, the plant should be divided into separate parts that can be modeled individually, including any non-linear elements. These submodels are then combined into a full model of the CDWW for use in the controller design.

2.6.3 Controller Design

Based on the model, an adaptive controller will be designed to ensure acceptable performance throughout the winding process. A supervisory control will also be designed to handle initialization and optimizing of line speed with regards to the possible motor speeds.

2.6.4 Test

The designed controller will be simulated and implemented on the Danfoss test plant. The test should demonstrate the controllers ability to adjust parameters throughout a full roll winding, while satisfying the requirement specification.

Chapter 3

Modeling

In this chapter a model will be created for the CDWW. To simplify the modeling, the entire system will be divided into smaller parts, which will be modeled individually and then combined. Finally, the model will be shown in block diagram form and its state space representation will be found. The developed model will be used for the controller design later in the project.

3.1 Web Material Models

Before modeling of the web tension dynamics can be done, a suitable mathematical model for the strain and stress relationships of the web material needs to be selected. The selected model needs to be general enough for use on a wide range of materials, while not being computationally complex. The following descriptions of lumped parameter models are based on information from [Roy01].

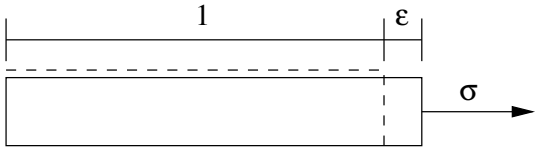


Figure 3.1: Definitions of strain, ε, and stress, σ.

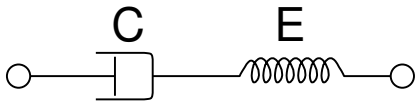


Figure 3.2: Maxwell model for viscoelastic materials.

3.1.1 Maxwell Model

The Maxwell model for viscoelasticity is a purely elastic spring in series with a viscous damper. The time-response for the strain with a given stress can be expressed as:

$$\dot{\varepsilon} = \frac{\sigma}{C} + \frac{\dot{\sigma}}{E} \tag{3.1}$$

The response to a constant applied force is then an immediate elastic deformation and a constantly increasing viscous deformation. When the force is removed, the elastic deformation springs back, while the viscous deformation remains. This model is best applied to very easily deformed materials, such as certain polymers. Since permanent deformation is generally not desired in a CDWW system, it is not appropriate to use this model.

3.1.2 Voigt Model

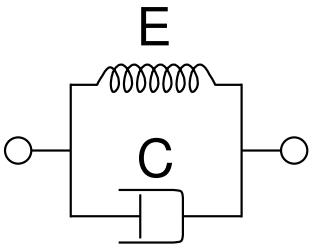


Figure 3.3: Voigt model for viscoelasticity materials.

The Voigt model for viscoelasticity is a purely elastic spring in parallel with a viscous damper. The stress can be expressed as:

$$\sigma = E\varepsilon + C\dot{\varepsilon} \tag{3.2}$$

When the strain is changed, the viscous component introduces additional stress. Alternatively, with a constant stress, the material relaxes over time and eventually reaches the strain defined by the purely elastic component. No permanent deformation is included in this model, and has been found to be a good model for many web materials [SF95],[Liu99].

3.1.3 Standard Linear Solid Model

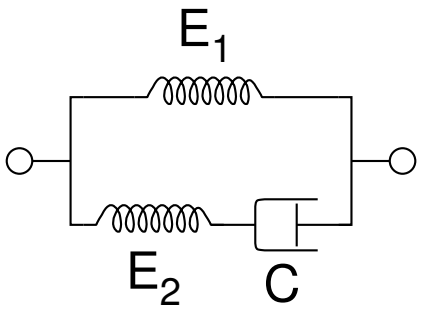


Figure 3.4: Standard linear solid model for viscoelasticity materials.

The standard linear solid model is a Maxwell model in parallel with a purely elastic spring. The time-dependent strain in response to applied stress can be expressed as:

$$\dot{\varepsilon} = \frac{\frac{E_2}{C}\left(\frac{C}{E_2}\dot{\sigma} + \sigma - E_1\varepsilon\right)}{E_1 + E_2} \tag{3.3}$$

While this model more accurately describes the dynamics of most viscoelastic materials, it is computationally expensive and can be numerically unstable.

3.1.4 Selection of Model

The most commonly used models described in the literature are simple purely elastic models and Voigt models for when web dampening is also to be modeled. The Voigt model is chosen because it most accurately describes the dynamic response, while remaining computationally simple.

3.2 Web Tension Dynamics

3.2.1 Assumptions

1. The web material is stiff, resulting in a small web strain, i.e. $\varepsilon \ll 1$.
2. The web elasticity can be accurately modeled with the Voigt-model for viscoelastic materials.
3. The air resistance and friction in the roller bearings can be ignored.
4. The line speed, V_1, can be considered constant over a period.
5. The tension in the previous web section, T_1, can be considered constant.
6. The web viscosity, density, and modulus of elasticity are constant, regardless of tension.
7. There is no slippage on any of the rolls.
8. The web material is stiff. Hence, the line speed V_1 and the winding speed V are very close.
9. The dancer movement is negligible compared to the length of the web in the winder zone.
10. The dancer velocity is negligible compared to the line speed.
11. The dancer moves linearly up and down, and is equipped with two idle rollers, to ensure a fixed web wrap angle.

3.2.2 Web Dynamics With Load Cell

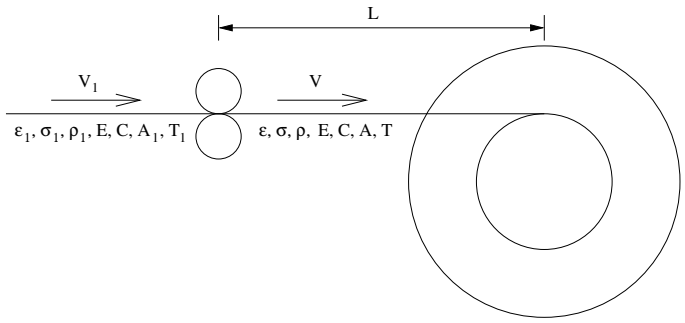


Figure 3.5: Simplified illustration of web winder tension zone.

Figure 3.5 shows the web winder tension zone, the previous tension zone, parameters, and variables for a CDWW system with a load cell type tension measurement.

The tension zone length, L, doesn't change with changes in tension, though the increasing winder roll radius will change the length over time.

The derivation of the web tension dynamics is based on the aforementioned assumptions, control volume analysis and continuum mechanics equations. The definition of mass continuity states that stretching the material does not change the mass of the stretched material:

$$\rho AL = \rho_u A_u L_u \tag{3.4}$$

Since the density is considered constant, this can be shortened:

$$AL = A_u L_u \Rightarrow \frac{L}{L_u} = \frac{A_u}{A} \tag{3.5}$$

The definition of strain gives:

$$\varepsilon = \frac{L - L_u}{L_u} = \frac{L}{L_u} - 1 \Rightarrow A = \frac{A_u}{1+\varepsilon} \tag{3.6}$$

The definition of mass conservation states that the change in mass of the control volume between the lead roll and the winder is equal to the difference between the mass that enters and exits the control volume:

$$\frac{d}{dt}(\rho AL) = \rho_1 A_1 V_1 - \rho AV \tag{3.7}$$

Using the definition of strain and mass continuity, this can be written:

$$\frac{d}{dt}\left(\frac{L}{1+\varepsilon}\right) = \frac{V_1}{1+\varepsilon_1} - \frac{V}{1+\varepsilon} \tag{3.8}$$

With assumption 1, the following first order approximation can be made:

$$\frac{1}{1+\varepsilon} \approx 1 - \varepsilon \tag{3.9}$$

The equation of mass conservation can then be rewritten:

$$\frac{d}{dt}(L\cdot(1-\varepsilon)) = V_1\cdot(1-\varepsilon_1)-V\cdot(1-\varepsilon) \tag{3.10}$$

$$L\frac{d\varepsilon}{dt} = V-V_1+V_1\varepsilon_1-V\varepsilon \tag{3.11}$$

$$L\varepsilon s = V-V_1+V_1\varepsilon_1-V\varepsilon \tag{3.12}$$

With assumption 2 and the definition of stress, the following relations exist:

$$\sigma = E\varepsilon + C\frac{d\varepsilon}{dt} = \frac{T}{A} \tag{3.13}$$

Rearranging with respect to ε in the Laplace domain gives:

$$T = EA\varepsilon + CAs\varepsilon \tag{3.14}$$

$$\varepsilon = \frac{T}{EA+CAs} \tag{3.15}$$

This is then inserted into the equation of mass conservation:

$$\begin{aligned} L\frac{sT}{EA+CAs} &= V-V_1+V_1\frac{T_1}{EA_1+CA_1s}-V\frac{T}{EA+CAs} \\ sT &= \frac{EA}{L}(V-V_1)+\frac{CA}{L}s(V-V_1)+\frac{V_1T_1}{L}\cdot\frac{A}{A_1}-\frac{V}{L}T \end{aligned} \tag{3.16}$$

The velocity difference, $V-V_1$, can be considered the input to a system where the output is the web tension, T. Given assumptions 4 and 5, the web dynamics are unaffected by the tension transfer from the previous section, so it is excluded from the transfer function.

$$T\cdot\left(s+\frac{V}{L}\right) = \left(\frac{EA}{L}+\frac{CA}{L}s\right)\cdot(V-V_1) \tag{3.17}$$

It can be seen that the pole of the system will be defined by V, which is variable. However, with assumption 8 the pole location will not be significantly moved by replacing V with V_1, which was assumed constant. The resulting transfer function is then:

$$H_{web}(s) = \frac{\frac{EA}{L}+\frac{CA}{L}s}{s+\frac{V_1}{L}} \tag{3.18}$$

Since this transfer function ignores the tension transferred from the previous section, it needs to be added back to get the actual tension that the web winder sees. Given assumptions 4 and 5, T_1 does not affect the dynamics and it can simply be added after the dynamics, as shown in Figure 3.6.

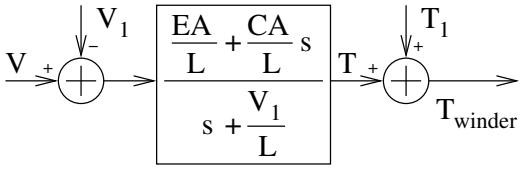


Figure 3.6: Block diagram for tension dynamics with a load cell.

In a real WTS it a reasonable assumption that T_1 is constant and known, as there will typically have been a processing section before the CDWW, which would have a controlled tension. However, in the test plant there is the possibility that T_1 varies, depending on how the unwind roll was previously wound. This tension variance will then have to be filtered through the dynamics, but comparison of the model shown in Figure 3.6 and one based on Equation 3.16 shows that if the tension varies sufficiently slowly, this effect can still be neglected.

3.2.3 Web Dynamics With Dancer

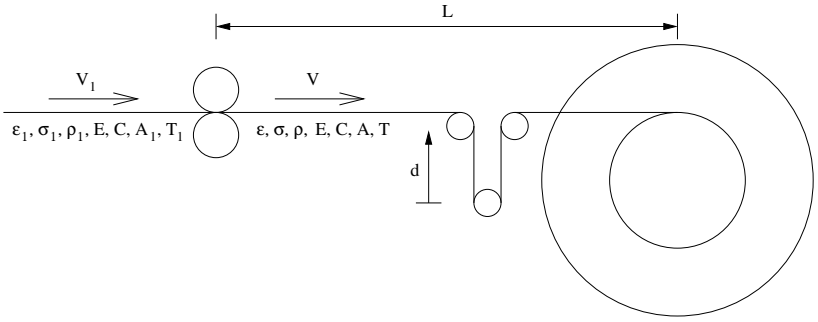


Figure 3.7: Simplified illustration of web winder tension zone, with a dancer.

Figure 3.7 shows the web winder tension zone, the previous tension zone, parameters, and variables for a CDWW system with a dancer type tension measurement. The tension zone length, L, is the length of web with the dancer at the minimum position.

Mass conservation for the control volume states:

$$\frac{d}{dt}\left(\rho A\left(L-2d\right)\right)=\rho_1 A_1 V_1-\rho A V \tag{3.19}$$

With the definition of strain and mass continuity, this can be written:

$$\frac{d}{dt}\left(\frac{L-2d}{1+\varepsilon}\right)=\frac{V_1}{1+\varepsilon_1}-\frac{V}{1+\varepsilon} \tag{3.20}$$

With assumption 1, this can be rewritten:

$$\begin{aligned}
\frac{d}{dt}\left(\left(L-2d\right)\cdot\left(1-\varepsilon\right)\right) &= V_1\cdot\left(1-\varepsilon_1\right)-V\cdot\left(1-\varepsilon\right) && (3.21)\\
-L\dot{\varepsilon}-2V_d+2d\dot{\varepsilon}+2V_d\varepsilon &= V_1-V_1\varepsilon_1-V+V\varepsilon && (3.22)\\
\left(L-2d\right)\dot{\varepsilon} &= V-2V_d-V_1+V_1\varepsilon_1-\left(V-2V_d\right)\cdot\varepsilon && (3.23)
\end{aligned}$$

Inserting the strain-tension relationship:

$$\frac{\left(L-2d\right)\cdot Ts}{EA+CAs}=V-2V_d-V_1+V_1\frac{T_1}{EA_1+CA_1 s}-\left(V-2V_d\right)\cdot\frac{T}{EA+CAs} \tag{3.24}$$

$$T\cdot\left(s+\frac{V-2V_d}{L-2d}\right)=\left(\frac{EA}{L-2d}+\frac{CA}{L-2d}s\right)\cdot\left(V-2V_d-V_1\right)+\frac{V_1T_1}{L-2d}\cdot\frac{A}{A_1} \tag{3.25}$$

With assumptions 8, 9 and 10, this can be simplified:

$$T\cdot\left(s+\frac{V_1}{L_N}\right)=\left(\frac{EA}{L_N}+\frac{CA}{L_N}s\right)\cdot\left(V-2V_d-V_1\right)+\frac{V_1T_1}{L_N}\cdot\frac{A}{A_1} \tag{3.26}$$

Where L_N is the web length with the dancer at the steady-state position dictated by the web tension set point. If one considers $V-2V_d-V_1$ to be the input to the system, T as the output, and takes into consideration assumptions 4 and 5, the transfer function can be written:

$$H_{web}=\frac{\frac{EA}{L_N}+\frac{CA}{L_N}s}{s+\frac{V_1}{L_N}} \tag{3.27}$$

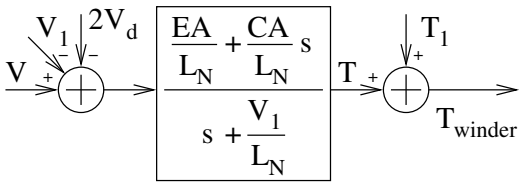


Figure 3.8: Block diagram for tension dynamics with a dancer.

Figure 3.8 shows the block diagram for the dynamic tension response, including the tension transferred from the previous section.

Assumptions 9 and 10 generally hold for a large-scale WTS, but on the test plant the dancer contains a significant part of the total web in the final tension zone. This means that the pole in the tension response may move significantly back and forth in response to disturbances. One possible solution is to use a dancer spring with a spring constant large enough to ensure that the expected disturbances at the nominal tension only move the dancer a small amount. However, this also has the cost of reducing the accuracy of the measurement.

The calculations in Appendix B show why assumption 11 is needed, as the relationship between web tension and dancer position would otherwise be nonlinear. An extra idler roll will be fitted on the test plant to avoid this kind of problem.

3.3 Dancer Dynamics

The dynamics of the dancer roll itself can be modeled with an analysis of the applied forces. Figure 3.9 shows a simplified model of the dancer, with the friction and attached spring modeled as an ideal spring in parallel with a viscous damper. F_w is the force caused by the weight of the dancer, while F_L is the spring loading of the dancer at its minimum position.

Using Newton's law, the equations of motion can be expressed as:

$$2T - F_w - F_L = m_d \cdot \ddot{d} + B_d \cdot \dot{d} + K_d \cdot d \tag{3.28}$$

A block diagram of this system is shown in Figure 3.10.

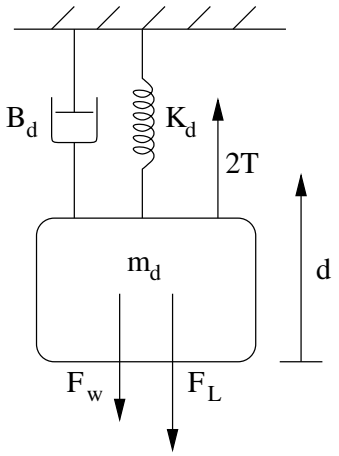


Figure 3.9: Mechanical model of dancer.

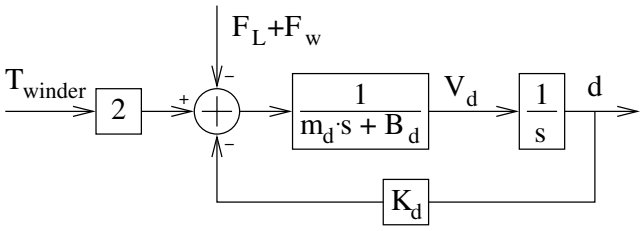


Figure 3.10: Block diagram for the dynamics of the dancer roll.

3.4 Drive Train

This section describes how the web velocity is found based on the torque. The equations are developed in the following sections.

3.4.1 Motor

In this project one motor drives the winder roll. The motor is considered a black box with one input and one output, and is modeled with a standard model for a motor with inertia and friction, as shown in Equation 3.29. The used model outputs the motors rotational velocity with the torque as input. The detailed calculation of the inertia for our test plant can be found in Appendix C.

$$\omega = \frac{\tau}{Js + B} \tag{3.29}$$

3.4.2 Web velocity

Next, the rotational velocity is converted to the web velocity, using Equations 3.30. The velocity depends on the motor rotational velocity, the radius of the winder roll, and the gear ratio from the motor to the winder roll. This ratio is 10.75:1.

$$V = \frac{\omega R}{N} \tag{3.30}$$

3.4.3 Block Diagram

Figure 3.11 shows the above equations combined in a block diagram which will be used in the full model.

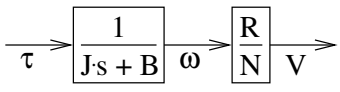


Figure 3.11: The block diagram shows the drive train.

3.5 Combined model

In the previous sections, the different parts of the system have been modeled. In Figure 3.12 and Figure 3.13 the combined system block diagrams are shown.

In the first diagram a load-cell is used for tension measurement, while a dancer is used in the second.

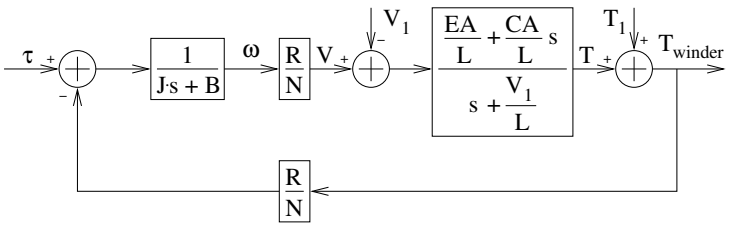


Figure 3.12: Combined model with load-cell type tension measurement.

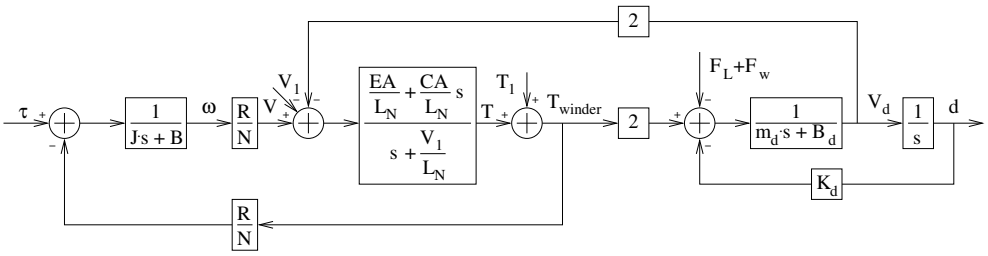


Figure 3.13: Combined model with dancer type tension measurement.

3.6 Model Selection

As the test plant already has a dancer, this is the model that will be the basis for the controller design. This means that the controller structure will be a torque-regulated position controller. The tension reference will have to be converted to a position reference with the use of Hooke's law:

$$d_{ref} = \frac{2T_{ref} - F_w - F_L}{K_d} \tag{3.31}$$

Before the model is put on state-space form, one simplification will be applied. Specifically, the web dampening coefficient will be removed. The rationale is that

since the material is very stiff, C is typically several orders of magnitude smaller than E[Liu98]. This places the associated zero far away from the rest of the plant dynamics and its effect is insignificant below 400 Hz. This places the effect of the dampening coefficient well outside the operational area.

3.7 State Space Model

Figure 3.14 is a restructuring of the simplified version of Figure 3.13, which serves to illustrate the state-space structure.

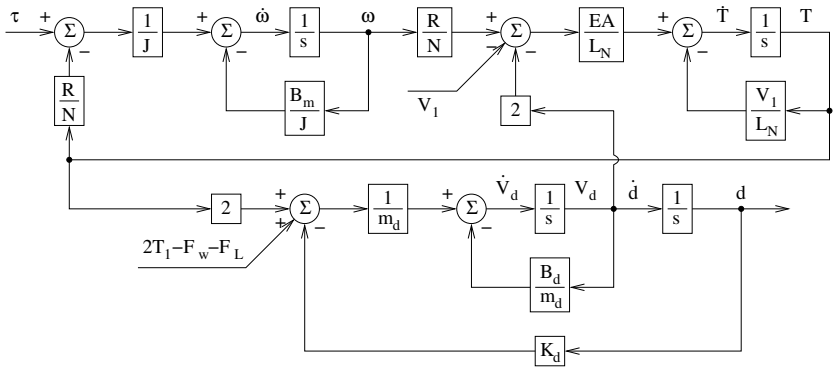


Figure 3.14: Restructured model for web winder system with dancer.

The following state equations are derived from Figure 3.14:

$$\dot{\omega} = -\frac{B_m}{J}\omega - \frac{R}{N}\frac{1}{J}T + \frac{1}{J}\tau \tag{3.32}$$

$$\dot{T} = \frac{R}{N}\frac{EA}{L_N}\omega - \frac{V_1}{L_N}T - 2\frac{EA}{L_N}V_d - \frac{EA}{L_N}V_1 \tag{3.33}$$

$$\dot{V}_d = \frac{2}{m_d}T - \frac{B_d}{m_d}V_d - \frac{K_d}{m_d}d + \frac{1}{m_d}F_x \tag{3.34}$$

$$\dot{d} = V_d \tag{3.35}$$

$$F_x = 2T_1 - F_w - F_L \tag{3.36}$$

Both V_1 and F_x enter the system as measured disturbance inputs. F_x is considered known and V_1 is measured. It is assumed that V_1 will remain constant for the most part, with occasional set point changes.

With the subscripts C and D denoting controllable and disturbance inputs, the state space form on which the controller design will be based is then:

$$\dot{\mathbf{x}} = \mathbf{A}\,\mathbf{x} + \mathbf{B}_C\,\mathbf{u}_C + \mathbf{B}_D\,\mathbf{u}_D \tag{3.37}$$

$$\mathbf{y} = \mathbf{C}\,\mathbf{x} \tag{3.38}$$

$$\mathbf{A} = \begin{bmatrix} -\frac{B_m}{J} & -\frac{R}{N}\frac{1}{J} & 0 & 0 \\ \frac{R}{N}\frac{EA}{L_N} & -\frac{V_1}{L_N} & -2\frac{EA}{L_N} & 0 \\ 0 & \frac{2}{m_d} & -\frac{B_d}{m_d} & -\frac{K_d}{m_d} \\ 0 & 0 & 1 & 0 \end{bmatrix},\ \mathbf{B}_C = \begin{bmatrix} \frac{1}{J} \\ 0 \\ 0 \\ 0 \end{bmatrix},\ \mathbf{B}_D = \begin{bmatrix} 0 & 0 \\ -\frac{EA}{L_N} & 0 \\ 0 & \frac{1}{m_d} \\ 0 & 0 \end{bmatrix}$$

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},\ \mathbf{x} = \begin{bmatrix} \omega \\ T \\ V_d \\ d \end{bmatrix},\ \mathbf{u}_C = \tau,\ \mathbf{u}_D = \begin{bmatrix} V_1 \\ F_x \end{bmatrix},\ \mathbf{y} = \begin{bmatrix} \omega \\ d \end{bmatrix}$$

Chapter 4

Controller Design

4.1 Controller Introduction

Adaptive controllers are used where parameters in the system change drastically. According to Oxford English Dictionary adaptive control is:

> *"A form of control in which the control parameters are automatically adjusted as conditions change so as to optimize performance."*

Systems where the performance is critical can be found in most of the processes from flight and ship control to process control, like the CDWW in this project. Common for these processes is that the controller must ensure certain performance criteria. In this project the tension must be kept constant through the winding process, even though the changing radius affects many plant parameters. Adaptive control can be divided into different categories of which some of the simpler were described in the Problem Analysis, Section 2.3. In this section two of the more advanced methods will be described. They are the Model Reference Adaptive Control (MRAC) and the Model Identification Adaptive Control (MIAC) [AW95, Page: 20-21], also known as Self Tuning Controller.

4.1.1 Model Reference Adaptive Control

The first type of controller is the Model Reference Adaptive Control (MRAC) which is depicted in Figure 4.1.

When using the MRAC the system response is compared to the desired response

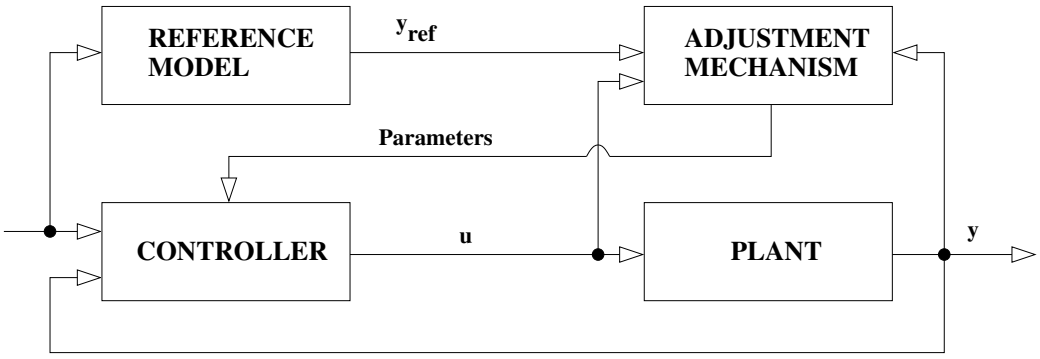


Figure 4.1: Block diagram of the Model Reference Adaptive Control.

from the reference model, which runs in parallel with the controller. The difference is used with the control signal to adjust and update the controller parameters.

The controller can be designed using different approaches, but the overall goal is to adjust the controller parameters so that the plant response matches the reference model, which does not necessarily have to be of the same order as the plant. Knowledge of plant structure and parameters can not be directly used for updating the controller parameters with a MRAC method. Since this method updates the controller parameters directly, it is commonly referred to as a direct adaptive control method.

4.1.2 Model Identification Adaptive Control

The plant model for the CDWW system has been studied and it is expected that many of the parameters can be identified online. The MIAC method can take advantage of this by identifying the plant parameters and use an automated controller design method to update the controller parameters.

Figure 4.2 shows how the MIAC first identifies the plant parameters and then transfers the estimated parameters, and optionally the uncertainty, to the controller generator. Based on this, the new controller parameters are calculated. The estimation method can be a least-squares method that is based on the plant input and output with a known structure, or it may simply be a parameter that can be measured directly or indirectly.

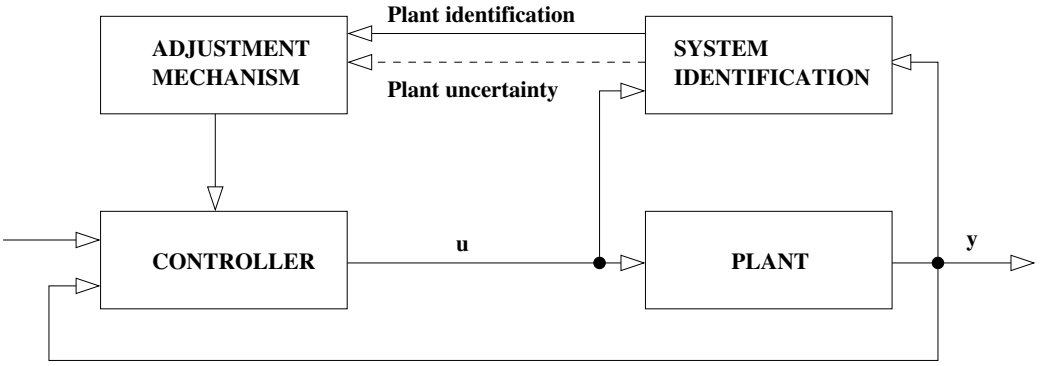


Figure 4.2: Block diagram of the Model Identification Adaptive Control.

4.2 Plant Controller

In this project the controller design is based on the model developed in Chapter 3. Development and testing has primarily been done on the Simulink model described in Appendix D, and unless otherwise noted, with the simulation constants described in Appendix E. The controller generator will be based on Optimal Control theory, since the updated controller can be calculated automatically, while performance can be adjusted through weighting parameters.

The controller gains should be updated periodically as the plant runs its course and the parameters change significantly or the need for different control weighting is detected. As the plant parameters are expected to change slowly, an update rate of once every second will probably be sufficient. It may also be a possible solution to adjust the update interval based on how large the change was, compared to the previous gain.

4.2.1 Controllability

To be able to influence the behavior of the plant, it needs to be controllable. This means the states can be taken to any desired value in finite time by applying a control input[SSJH02].

Ordinarily, controllability can be tested with the Matlab function $CTRB(\mathbf{A}, \mathbf{B}_C)$, which returns the controllability matrix of the system. If the rank of this matrix equals the number of states, the plant is controllable. However, since the system is expected to change, it is not sufficient to test at nominal parameter values. Instead

the controllability matrix is constructed:

$$\mathcal{C} = \begin{bmatrix} \mathbf{B}_C & \mathbf{A}\mathbf{B}_C & \mathbf{A}^2\mathbf{B}_C & \mathbf{A}^3\mathbf{B}_C \end{bmatrix} \tag{4.1}$$

It is seen that $\mathcal{C}$ is an upper triangular matrix with the following elements on the diagonal:

$$diag(\mathcal{C}) = \begin{bmatrix} \frac{1}{J} & \frac{R}{N}\frac{EA}{L_N}\frac{1}{J} & \frac{2}{m_d}\frac{R}{N}\frac{EA}{L_N}\frac{1}{J} & \frac{2}{m_d}\frac{R}{N}\frac{EA}{L_N}\frac{1}{J} \end{bmatrix} \tag{4.2}$$

Since neither of these elements can even theoretically become zero, the controllability matrix will always have a rank of four, which means that the system will always be controllable. In the following the controller will be developed for the plant.

4.2.2 Linear Quadratic Regulator

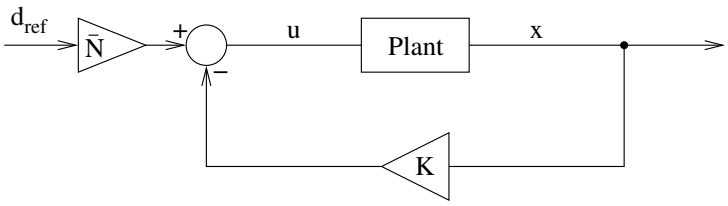


Figure 4.3: Basic LQR with reference input.

Figure 4.3 shows a basic full-state feedback system with reference input. Using optimal control theory the optimal gains, $\mathbf{K}$, can be found, such that the control law $\mathbf{u}_C = -\mathbf{K}\mathbf{x} + \bar{N}d_{ref}$ inserted in Equation 4.3 minimize the performance function in Equation 4.4[FPEN02, Page 532]:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}_C\mathbf{u}_C \\ \mathbf{y} &= \mathbf{C}x \end{aligned} \tag{4.3}$$

$$J = \int_0^\infty \left(\mathbf{x}^T\mathbf{Q}_c\mathbf{x} + \mathbf{u}_C^T\mathbf{R}_c\mathbf{u}_C\right)dt \tag{4.4}$$

Methods for solving the optimization problem have been developed extensively over the years. One common solution involves finding the positive-definite solution to the Algebraic Riccati Equation (ARE):

$$\dot{\mathbf{S}} = \mathbf{0} = \mathbf{A}^T\mathbf{S} + \mathbf{S}\mathbf{A} - \mathbf{S}\mathbf{B}_C\mathbf{R}_c^{-1}\mathbf{B}_C^T\mathbf{S} + \mathbf{Q}_c \tag{4.5}$$

Using the Matlab function $LQR(\mathbf{A}, \mathbf{B}_C, \mathbf{Q}_c, \mathbf{R}_c)$, the optimal gains can be found. The function uses the system in Equation 4.3 and the weighting matrices $\mathbf{Q}_c$ and $\mathbf{R}_c$. However, *LQR* is not part of the embeddable subset of the Matlab language, so an alternative method needs to be implemented. The procedure used in this project for solving the optimization problem using the ARE is described in Appendix F.

The diagonal matrices $\mathbf{Q}_c$ and $\mathbf{R}_c$ are used for weighting the accuracy of state tracking and the control effort. $\mathbf{Q}_c$ must be positive semi definite and $\mathbf{R}_c$ must be positive definite. In practice the initial values of $\mathbf{Q}_c$ and $\mathbf{R}_c$ can be selected, using Bryson's rule[FPEN02, Page 537] such that:

$$\begin{aligned} \mathbf{Q}_{c_{ii}} &= 1/\text{maximum acceptable value of } [\mathbf{x}_i^2] \\ \mathbf{R}_{c_{ii}} &= 1/\text{maximum acceptable value of } [\mathbf{u}_{C_i}^2] \end{aligned}$$

The weighting can be modified to achieve an acceptable trade off between performance and control effort. One common way to weigh the control accuracy / cost ratio is to multiply the $\mathbf{Q}_c$ matrix with a scaling factor, γ. The more expensive the control, the lower γ needs to be, and vice versa.

It may be possible to detect an unacceptable response in the closed loop system by measuring the peak absolute error while the V_1 reference is being adjusted, and increasing γ to compensate. This would allow the controller to determine the minimum amount of control effort required to have acceptable performance. Likewise, a detection of actuator saturation might be a good indication that γ needs to be decreased.

The $\bar{N}$ scaling factor is used to bring the plant state to a point where the output equals the reference signal, d_{ref}. It is, however, not robust with regard to changing plant parameters and a non-zero steady state error is very likely. One way to ensure robust tracking of a reference signal with no steady state error is to use integral control[FPEN02, Page 586].

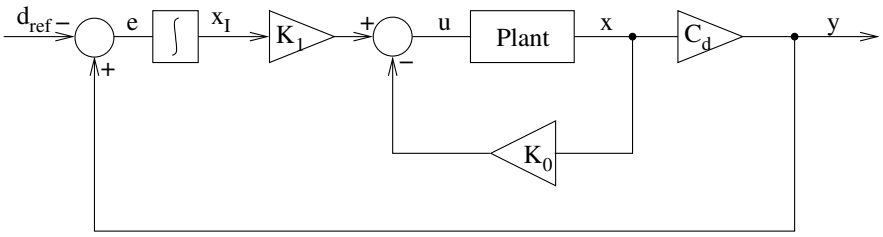


Figure 4.4: LQR with integral control.

4.2.3 Integral Control

Figure 4.4 shows the LQR controller with a reference input introduced through integral control. The integral control gives an extra state, representing the error between the measured and the estimated output. Integral control is obtained by augmenting the state space representation of the plant model. The augmented matrix can be written in the following general form, where $\mathbf{C}_d$ is the row of the $\mathbf{C}$ matrix related to the dancer position:

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{x_I} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{C}_d & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ x_I \end{bmatrix} + \begin{bmatrix} \mathbf{B}_C \\ \mathbf{0} \end{bmatrix} \mathbf{u}_C - \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} d_{ref}$$

The feedback law, for the integral control, is:

$$\mathbf{u}_C = -\begin{bmatrix} \mathbf{K}_0 & K_1 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ x_I \end{bmatrix}$$

When using the integral control, the feedback gains are calculated in the same way as before for the Linear Quadratic Regulator.

Based on Bryson's rule, the weighting matrices used in the project are:

$$\mathbf{Q}_c = diag\left(\begin{bmatrix} \frac{1}{105^2} & \frac{1}{100^2} & \frac{1}{1^2} & \frac{1}{0.1^2} & \frac{1}{0.001^2} \end{bmatrix}\right)$$

$$\mathbf{R}_c = \begin{bmatrix} \frac{1}{0.1^2} \end{bmatrix}$$

This sets the range for the winder motor rotational velocity to around 1000 RPM. The allowable tension is roughly in the range where it will not break. There is essentially no limit on V_d, as the controller should not penalize faster movement

towards the setpoint. The dancer limit is set so it can move freely in the range defined by the physical plant. The final limit is for the allowable integrator state, which is a way of limiting the allowable error. While it is tempting to set this limit much lower, there is the risk that the controller will over-react to disturbances and give an overshoot in the response. The allowable control input is initially set in the range that a simulation showed would be sufficient.

4.2.4 Integrator Resetting

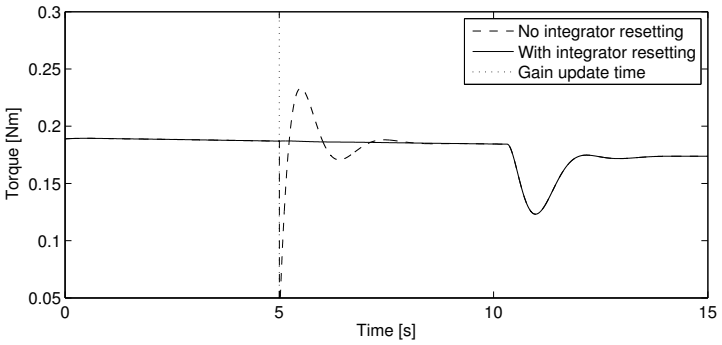


Figure 4.5: Control output with and without integrator resetting.

One problem with updating a state feedback gain and/or the integral gain with integral optimal control is that as soon as the gain is changed, the control input will have a discontinuous jump, as shown in Figure 4.5. This happens because the integrator state now no longer matches the updated controller.

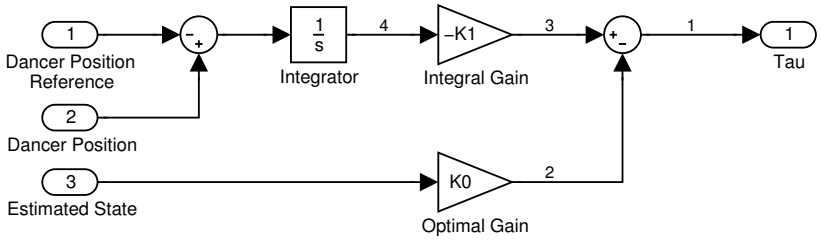


Figure 4.6: Basic Integral Optimal Control.

Therefore it is necessary to detect when the gains are changed and calculate a new integrator state. Figure 4.6 shows the basic Integral Optimal Control structure for the web winder system. The goal is to avoid discontinuity at the point 1. A change

in K_1 means that the state at point 4 needs to be scaled by the ratio between the old and new K_1 to avoid a discontinuity at point 3. A change in $\mathbf{K}_0$ will also cause a discontinuity at point 2, so the difference between the value calculated with the old and new $\mathbf{K}_0$ needs to be added to point 3 to compensate.

Equation 4.6 shows how to calculate the new integrator state, and Figure 4.7 shows the Simulink model used to both detect any change in gains and reset the integrator to the correct state.

$$x_{I_{new}} = x_{I_{old}} \frac{K_{1_{old}}}{K_{1_{new}}} + \frac{\mathbf{K}_{0_{old}} - \mathbf{K}_{0_{new}}}{K_{1_{new}}} \mathbf{x} \tag{4.6}$$

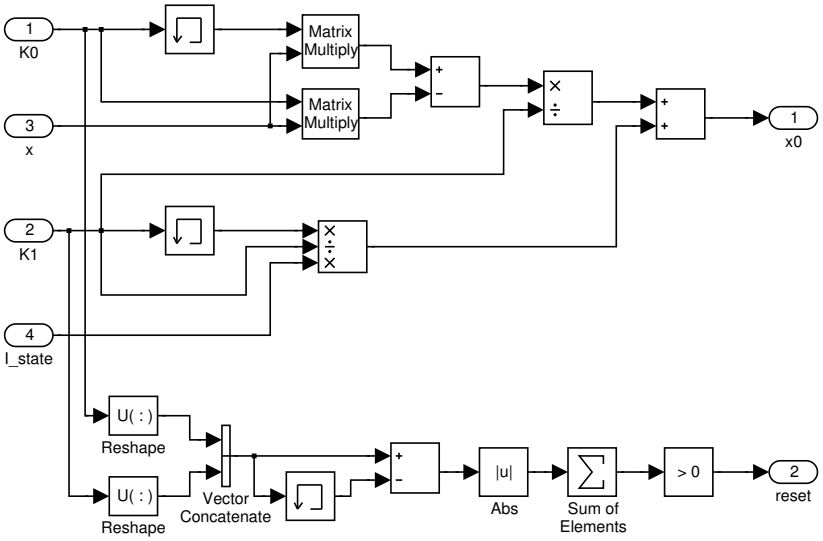


Figure 4.7: Resetting circuit.

4.3 State Estimator

This section describes the design of the Kalman estimator, also is known as an optimal observer. This is needed to provide full state feedback to the LQR when not all states can be measured.

4.3.1 Observability

To build a state estimator, the system must be observable which means that any earlier value of the state vector is determinable by watching the output[SSJH02, Page 589].

Ordinarily, observability can be tested with the Matlab function $OBSV(\mathbf{A},\mathbf{C})$, which returns the observability matrix of the system. If the rank of this matrix equals the number of states, the plant is observable. As with controllability, it is not sufficient to test nominal parameter values, so the observability matrix is constructed:

$$\mathcal{O} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \mathbf{CA}^3 \end{bmatrix} \tag{4.7}$$

For this system, the first four rows of $\mathcal{O}$ are:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{B_m}{J} & -\frac{R}{N}\frac{1}{J} & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{4.8}$$

Since R can never be 0, this is already enough to show that the rank of $\mathcal{O}$ will always be four, which means that the states will always be observable.

4.3.2 Full-Order Estimator

In principle an open loop estimator could be used, but this is very sensitive to incorrect initial conditions, or modeling error. Either the estimation error would grow continually or go to zero very slowly. Therefore it is better to use a closed loop estimator, as shown in Figure 4.8. This includes feedback from the measured and estimated output, which is used for correcting the state estimate equation [FPEN02, Page 541]:

$$\dot{\hat{\mathbf{x}}} = \mathbf{A}\,\hat{\mathbf{x}} + \mathbf{B}_C\,\mathbf{u}_C + \mathbf{B}_D\,\mathbf{u}_D - \mathbf{L}_o\,(\mathbf{y} - \mathbf{C}\,\hat{\mathbf{x}}) \tag{4.9}$$

Using the Matlab function $LQE\,(\mathbf{A},\mathbf{G},\mathbf{C},\mathbf{Q}_o,\mathbf{R}_o)$, the optimal Kalman gain, $\mathbf{L}_o$, can be found, where $\mathbf{G}$ is the identity matrix and $\mathbf{Q}_o$ and $\mathbf{R}_o$ are covariance matrices that must be positive semidefinite and positive definite, respectively. $\mathbf{Q}_o$

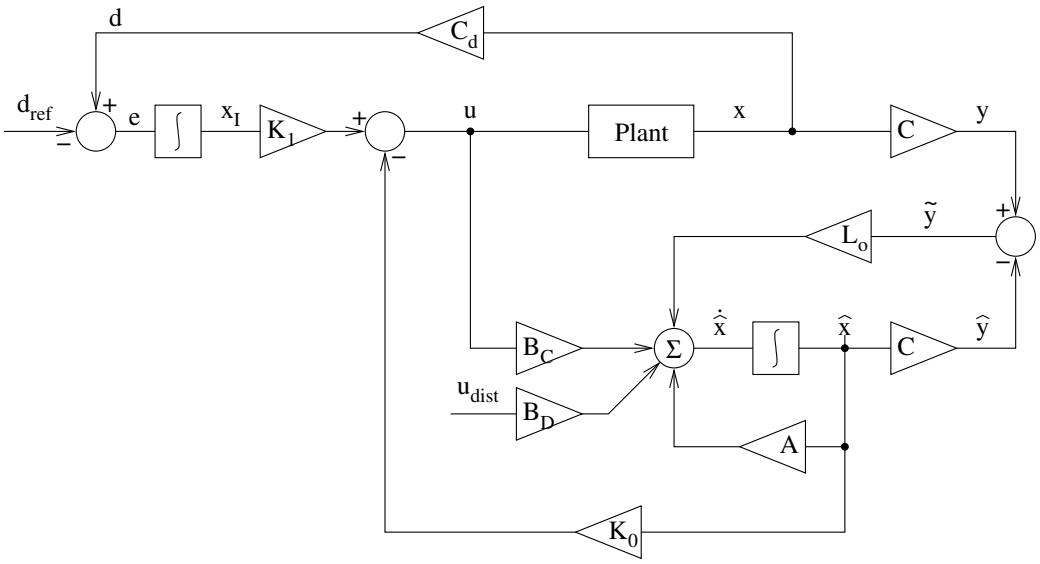


Figure 4.8: The plant with integral optimal control and state estimator.

and $\mathbf{R}_o$ represent the intensity of the process and the sensor noise[SSJH02, Page 686]. A relatively low value for $\mathbf{R}_o$ would place a great deal of confidence in the accuracy of the sensor measurement, but if the measurement is noise-filled, the estimator would incorrectly attempt to track this. Tuning the $\mathbf{R}_o$ and especially the $\mathbf{Q}_o$ matrix can be difficult if little is known about the process noise, and it is usually a recursive process. The initial estimates for $\mathbf{Q}_o$ and $\mathbf{R}_o$ are:

$$\mathbf{Q}_o = diag\left(\begin{bmatrix} 10 & 1 & 0.1 & 0.001 \end{bmatrix}\right)$$

$$\mathbf{R}_o = diag\left(\begin{bmatrix} 0.1 \cdot 10^{-3} & 0.1 \cdot 10^{-3} \end{bmatrix}\right)$$

4.3.3 Separation Principle

Using an estimator does not influence the control of the plant. This is because of a property known as the Separation Principle [SSJH02, Page 646-647].

The plant with the controller and the estimator can be represented as a block triangular matrix, where $\tilde{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}}$:

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\tilde{\mathbf{x}}} \end{bmatrix} = \begin{bmatrix} \mathbf{A} - \mathbf{B}_C \mathbf{K} & \mathbf{B}_C \mathbf{K} \\ 0 & \mathbf{A} - \mathbf{L}_o \mathbf{C} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \tilde{\mathbf{x}} \end{bmatrix}$$

The eigenvalues of the matrix are the eigenvalues of the matrices along the diagonal blocks. This means the closed-loop eigenvalues or poles are the union of the observer and the controller poles. Because the system is controllable, the poles of $\mathbf{A} - \mathbf{B}_C \mathbf{K}$ can be placed arbitrarily without affecting the estimator poles. Likewise, because of the observability, the poles of $\mathbf{A} - \mathbf{L}_o \mathbf{C}$ can be placed arbitrarily without affecting the controller poles.

4.3.4 Alternative State Estimator

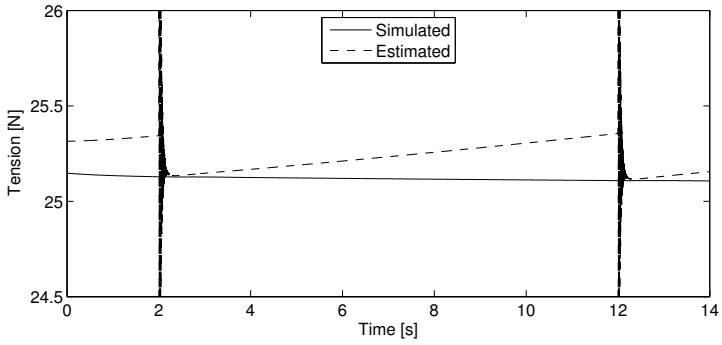


Figure 4.9: Gradual divergence of web tension estimate.

Simulations and tests on the plant have shown that the piecewise linear Kalman filter solution is unable to correctly track the plant state. As seen in Figure 4.9, the estimated tension rapidly diverges between system updates. This is because the tension calculation according to the model involves the very small difference between V and V_1, where V is calculated based on the measured ω and a R that is considered static, but is actually increasing. Since this error tends to cascade through the model, it renders the entire state estimate unreliable.

One possible solution might be to continually update the value of R in the observer model. However, in order to have an accurate tension estimate the estimated R also needs to be very accurate, due to the very small velocity differences involved.

Another solution is to use knowledge of the plant physics and note that when the dancer is not accelerating, the tension is exactly equal to:

$$T = \frac{K_d \cdot d - F_x}{2} \tag{4.10}$$

Using this estimate would be inaccurate while the dancer is accelerating, due to the extra force required to accelerate m_d. However, as the simulation in Figure

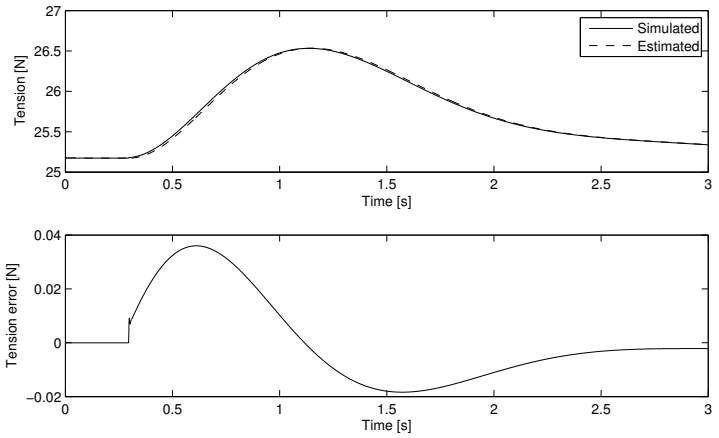


Figure 4.10: Alternative tension estimate error during disturbance.

4.10 shows, the tension estimate error is negligible with the type of disturbances typically seen.

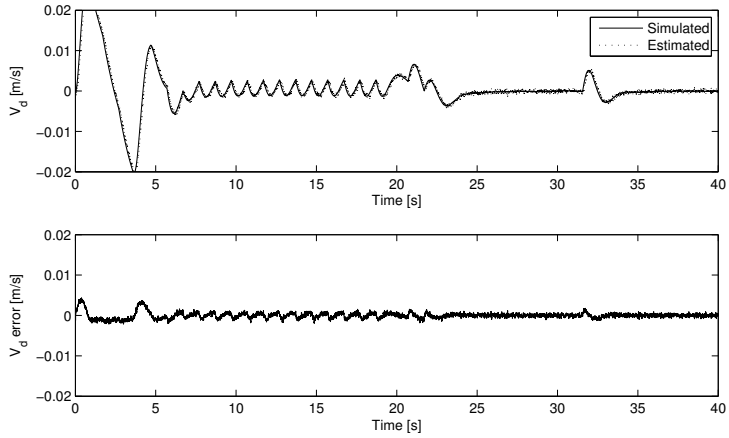


Figure 4.11: Alternative V_d estimate error during simulation with noisy d.

An alternative way to estimate the dancer velocity, V_d, is to differentiate the dancer position. However, because the measured signal is likely to contain noise, it should be filtered in some way before being differentiated. In Figure 4.11, the system was simulated with Gaussian white noise with a standard deviation of 0.1 *mm* added to the dancer position. The samples are averaged over a window of 0.1 seconds and filtered through a first-order filter with a time constant of $\frac{1}{20}$.

This is considered to be an acceptably good estimate and will be used instead of the Kalman filter estimate.

4.4 Startup

When starting the system, the dancer will be at its minimum position. The dancer will not start moving before a minimum tension is reached, determined by the dancer weight and the prestressing of the dancer springs. This minimum force has been measured to be approximately 22 N, or 11 N of web tension.

When the system starts, only the winder motor is activated, corresponding to a nonzero V and zero web line speed, V_1. This tightens the web and lifts the dancer to a higher starting position, as illustrated in Figure 4.12.

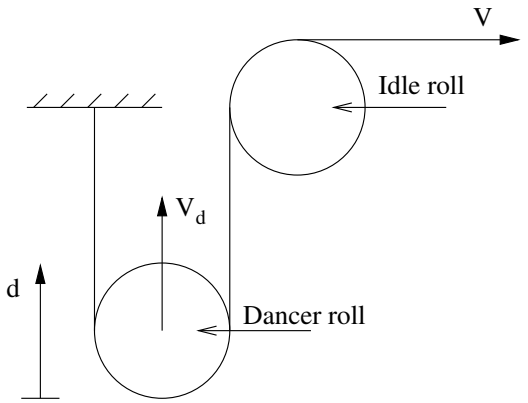


Figure 4.12: In the startup phase V_1 is zero and V is nonzero. This will raise d.

When the dancer has reached the desired starting position, the V_1 controller can be activated, without risk of the web losing contact with the dancer and idle rolls.

4.5 V_1 controller

4.5.1 Control loop

To control the line speed of the web entering the system, the unwinder motor needs to be controlled. The motor itself has no feedback, and can only be given an open-loop speed reference. However, the V_1 speed is measured with an encoder, and a controller can be built around the system.

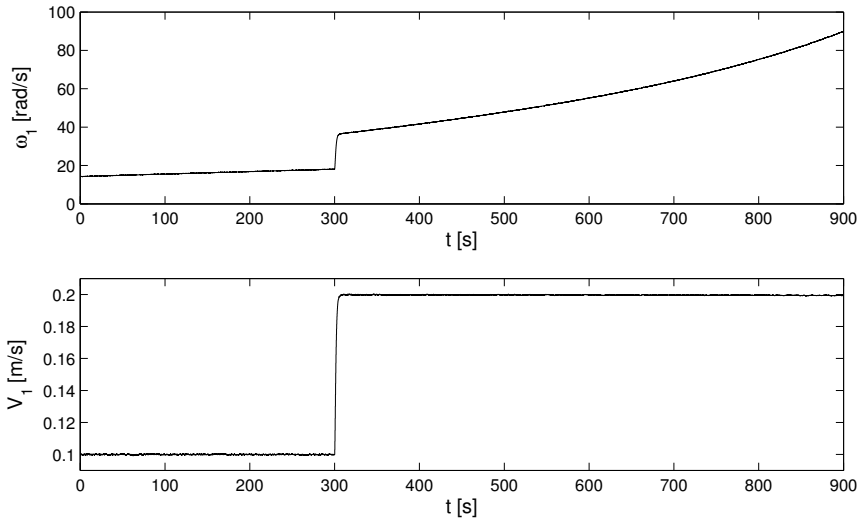


Figure 4.13: Simulation of V_1 controller.

As the web unwinds, the changing radius requires the motor speed to increase gradually. To maintain optimal performance, the controller would require some sort of adaptation, either through simple gain scheduling or more advanced adaptive control. However, since the accuracy and performance of this controller is not of particular importance to the main goal of the project, a simpler solution is to use a PI controller with static gains.

Testing on a simulation of the system shows that pure integral control with a gain in the order of 300-400 gives acceptable performance, even in the presence of significant noise.

Figure 4.13 shows a full run with a reference step at 300 seconds. A uniform random noise signal between -0.01 and 0.01 $^{m}/_{s}$ was added to the V_1 feedback signal to show that even severe quantization noise from the encoder will not degrade the performance significantly.

4.5.2 Speed supervisor

Since the radius ratio between the unwinder and winder roll is constantly changing, the line speed occasionally needs to be adjusted to allow the motors to run at the fastest speed possible. Since the winder roll typically starts with the smallest radius, it is this motor that will initially determine the line speed.

As the web builds up, the rotational speed will slow down with a constant line speed. Therefore the line speed set point should be increased whenever the rotational speed drops below a certain threshold. Likewise, the unwinder roll will have to rotate faster as the web is unrolled. When it reaches its maximum speed, the line speed set point should be decreased to compensate. Since the natural tendency is for the winder roll to move slower and the unwinder roll to move faster over time, the unwinder roll limit needs to have priority over the winder roll. This is implemented by never increasing the line speed again, after the unwinder roll has once reached its speed limit.

Figure 4.14 shows a pseudo-code representation of the line speed supervisor and 4.15 shows a simulation run with the supervisor activated. The winder roll minimum limit and unwinder roll maximum limit are set to the same speed.

```
while ω1 < ω1max
  if ω < ωmin
    V1ref = V1ref + stepsize
    pause settlingtime
while not stopped
  if ω1 > ω1max
    V1ref = V1ref − stepsize
    pause settlingtime
```

Figure 4.14: Line speed supervisor.

4.6 Radius Estimation

The radius of the winder roll is constantly changing and must therefore be measured or estimated. The test plant has no sensor for this, and it is generally desirable to avoid adding sensors if it can be avoided. Therefore, a few methods for estimating the radius are examined.

$$R_{est_{vel}} = \frac{N(V_1 + 2V_d)}{\omega} \tag{4.11}$$

In the first method, Equation 4.11 is used, which is simply based on the ratio between the tangential and rotational velocity of the winder roll. The method can be sensitive to sensor noise at low speeds, so it should not be used before V_1 has reached a certain minimal speed. When the radius is at its smallest, ω will be large

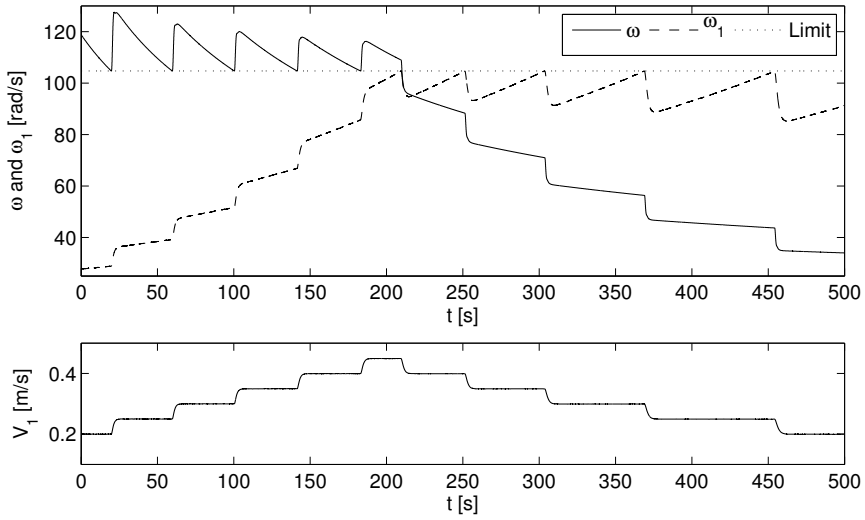


Figure 4.15: Test of line speed supervisor.

and the sensor noise will be comparatively small. As the radius builds up, ω will typically have to decrease, thus increasing the relative importance of the sensor noise. This estimate is therefore best suited for the initial part of the winding process.

Another method for estimating the radius is based on an approximation of the length of an Archimedean spiral, commonly used for calculations on inductor coil windings. With n_{winder} as the total number of rotations of the winder roll and G as the web material thickness, the approximated length is:

$$L_{web} = \pi \cdot n_{winder} \cdot (2 \cdot R_{core} + n_{winder} \cdot G) \tag{4.12}$$

$$G = \frac{L_{web} - 2 \cdot \pi \cdot n_{winder} \cdot R_{core}}{\pi \cdot n_{winder}^2} \tag{4.13}$$

For use in radius estimation, L_{web} and n_{winder} can be found by integrating V_1 and $\frac{\omega}{2\pi \cdot N}$, respectively. The length approximation can then be used to find a radius estimate:

$$R_{est_{coil}} = G \cdot n_{winder} + R_{core} = \frac{L_{web}}{\pi \cdot n_{winder}} - R_{core} \tag{4.14}$$

This approximation is unfortunately not optimal for materials that are very thin,

relative to the core radius. It is necessary to test for impossible estimates, such as anything less than the core radius. Once significant amounts of web material have been wound, this estimate does approach the real radius asymptotically. Since it also integrates the velocities, sensor noise tends to get averaged out. This results in a radius estimate that gets increasingly accurate towards the end of the winding process.

The startup for the two estimation methods are shown in Figure 4.16, where they are compared to the real radius. To compensate for the above mentioned deficiencies, it would be useful to combine the two estimation methods. One way is to use $R_{est_{vel}}$ until $R_{est_{coil}}$ comes sufficiently close to $R_{est_{vel}}$, then using $R_{est_{coil}}$ from that point on.

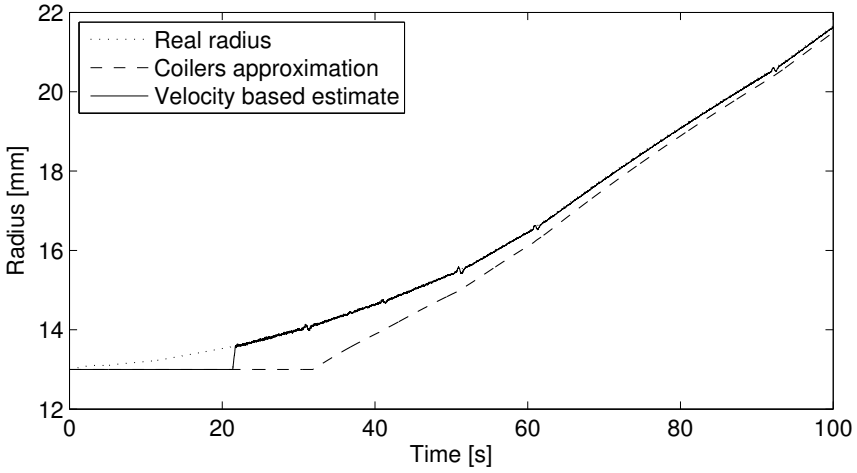


Figure 4.16: Two methods for estimating the radius compared to the real radius.

4.7 Inertia Estimation

When winding the web, the inertia is growing as more material is rolled onto the winder roll and both the mass and radius increases. As shown in Appendix G, a controller optimized for a particular system inertia might give poor performance, or even cause instability, if the real system inertia differs significantly. As shown in Appendix C, this is not an issue for the test plant, as the inertia caused by the web is insignificant. However, it is not unreasonable to expect the increasing inertia caused by the web to be the dominant source of plant inertia in a real application.

4.7.1 Mathematical Estimate

One way to estimate the current plant inertia could be to calculate it mathematically, based on a measurement or estimate of the current winder roll radius and parameters entered by the operator.

In order to calculate the current system inertia, the operator would have to enter the weight and radius of an empty core, m_{core} and R_{core}, the weight and radius of a full roll, m_{full} and R_{full}, the unchanging plant inertia, J_{sys}, and the gearing ratio, N.

The inertia of the core roll is:

$$J_{core} = \frac{1}{2} \cdot m_{core} \cdot R_{core}^2 \tag{4.15}$$

The density per meter of roll length for the particular web material can then be calculated as:

$$\lambda_{web} = \frac{m_{full} - m_{core}}{\pi \cdot \left(R_{full}^2 - R_{core}^2\right)} \tag{4.16}$$

With a winder roll radius measurement or estimate, R_{est}, the estimated mass of the web currently on the winder roll can be calculated:

$$m_{web_{est}} = \lambda_{web} \cdot \pi \cdot \left(R_{est}^2 - R_{core}^2\right) \tag{4.17}$$

The inertia is then:

$$J_{web_{est}} = \frac{1}{2} \cdot m_{web_{est}} \cdot \left(R_{est}^2 + R_{core}^2\right) \tag{4.18}$$

$$= \frac{1}{2} \cdot \frac{\left(m_{full} - m_{core}\right) \cdot \left(R_{est}^4 - R_{core}^4\right)}{R_{full}^2 - R_{core}^2} \tag{4.19}$$

For the total system inertia, the core and web inertia needs to be reflected back to the motor and added to the static system inertia:

$$J_{total} = J_{sys} + \frac{J_{core} + J_{web_{est}}}{N^2} \tag{4.20}$$

4.7.2 System Identification

Another way to determine the system inertia is through system identification. Only the subsystem between the torque input and the motor rotational velocity needs to be considered, and it is modeled as the following first order system:

$$H_{motor}(s) = \frac{1}{Js + B_m} \tag{4.21}$$

The system identification model is chosen to be a first order AutoRegressive with eXogeneous input (ARX) model:

$$\begin{aligned} A(q)\,y(t) &= B(q)\,u(t) + e(t) && (4.22)\\ \left(1 + a_1 q^{-1}\right) y(t) &= b_1 q^{-1} u(t) + e(t) && (4.23) \end{aligned}$$

The signal and parameter vectors for this system are:

$$\varphi(t) = \begin{bmatrix} -y(t-1) \\ u(t-1) \end{bmatrix}, \quad \hat{\theta} = \begin{bmatrix} \hat{a}_1 \\ \hat{b}_1 \end{bmatrix} \tag{4.24}$$

Since the ARX model has a linear predictor, the analytical parameter estimate, also known as the Least Squares (LS) solution, can be determined if a set of n samples have been collected:

$$\hat{\theta} = \left[\sum_{t=1}^{n} \varphi(t)\,\varphi(t)^T \right]^{-1} \sum_{t=1}^{n} \varphi(t)\,y(t) \tag{4.25}$$

The application calls for continuous estimation, and one way to do this would be to calculate the parameter estimate on a sufficiently large moving window of samples. However, this is computationally expensive and also weighs the influence of all samples equally. Since the tracked parameters will be changing over time, a better solution is to use the Recursive Least Squares (RLS) algorithm with exponential forgetting. An algorithm for doing this without the need for matrix inversion is derived in [Lju99, Page 365]:

$$\mathbf{L}(t) = \frac{\mathbf{P}(t-1)\,\varphi(t)}{\lambda+\varphi(t)^T\,\mathbf{P}(t-1)\,\varphi(t)} \tag{4.26}$$

$$\hat{\theta}(t) = \hat{\theta}(t-1)+\mathbf{L}(t)\left(y(t)-\varphi(t)^T\,\hat{\theta}(t-1)\right) \tag{4.27}$$

$$\mathbf{P}(t) = \frac{1}{\lambda}\left[\mathbf{P}(t-1)-\frac{\mathbf{P}(t-1)\,\varphi(t)\,\varphi(t)^T\,\mathbf{P}(t-1)}{\lambda+\varphi(t)^T\,\mathbf{P}(t-1)\,\varphi(t)}\right] \tag{4.28}$$

If any initial parameter estimate, θ_{est}, is known, $\hat{\theta}(1)$ should be set to θ_{est} to ensure faster convergence. The estimate will, however, converge anyway. $\mathbf{P}(1)$ should be initialized with a diagonal matrix. The magnitude should reflect the confidence in θ_{est}, with larger values indicating lower confidence.

The λ parameter is a measure for how fast previous samples will be forgotten. It can be approximated to an effective window of samples:

$$n_{eff} \approx \frac{1}{1-\lambda},\ 0.95<\lambda<1 \tag{4.29}$$

$$\lambda \approx \frac{n_{eff}-1}{n_{eff}} \tag{4.30}$$

Determining the appropriate λ requires some consideration. One important factor is the chosen sampling time, which should be fast enough to capture the primary dynamics. It is also important to consider how fast the identified parameters are expected to change. A faster forgetting factor could mean that the noise and other disturbances have too large an influence. In the same way, a slower forgetting factor might result in an inability to follow the parameter development sufficiently fast.

Assuming a good estimate with little noise, the ARX model can be expressed as an equivalent discrete transfer function:

$$H_d(z) = \frac{b_1 z^{-1}}{1+a_1 z^{-1}} \tag{4.31}$$

$$= \frac{b_1}{z+a_1} \tag{4.32}$$

Discrete transfer functions of this form have no simple direct version in the Laplace domain. Several approximation techniques can, however, be applied. One of the simpler is known as Euler's method[FPW98, Page 59]:

$$\dot{x}(t) = \lim_{\Delta t \to 0} \frac{x(t+\Delta t) - x(t)}{\Delta t} \tag{4.33}$$

$$s \cdot x(t) \approx \frac{x(t+T_s) - x(t)}{T_s} \tag{4.34}$$

$$s \cdot x(k) \approx \frac{x(k+1) - x(k)}{T_s} \tag{4.35}$$

$$sT_s \cdot x(k) \approx x(k)(z-1) \tag{4.36}$$

$$z \approx sT_s + 1 \tag{4.37}$$

The continuous time version of H_d can then be approximated to:

$$H_c(s) = H_d(sT_s + 1) \tag{4.38}$$

$$= \frac{b_1}{sT_s + 1 + a_1} \tag{4.39}$$

$$= \frac{\frac{b_1}{T_s}}{s + \frac{1+a_1}{T_s}} \tag{4.40}$$

Extracting the desired continuous time plant parameters is then trivial:

$$H_c(s) = \frac{\frac{1}{J}}{s + \frac{B_m}{J}} \tag{4.41}$$

$$J = \frac{T_s}{b_1} \tag{4.42}$$

$$B_m = \frac{1 + a_1}{b_1} \tag{4.43}$$

4.7.3 Practical Issues

In a real system it is likely that there will be considerable noise on both the control signal and the measured motor speed. Depending on the chosen effective window, the noise might dominate over the dynamics caused by the system inertia. Filtering the input before feeding it to the RLS algorithm may alleviate this effect.

Another practical issue is the nonlinearity caused by the unmodeled Coulomb friction, which introduces a roughly constant torque loss. Since this may adversely

affect the RLS algorithm, the data needs to be detrended online. One way to do this could be to subtract the mean of a window of samples from the data.

One final difficulty with recursive estimation algorithms is that they forget the already seen dynamics after a while, with drifting estimated parameters as a result. To avoid this, there needs to be sufficient excitation of the system to maintain a reliable estimate. In the CDWW test plant, this excitation may come from just the changing radius and set point changes on the line speed. If this is insufficient, it may also be necessary to occasionally apply some extra disturbances to give the RLS algorithm some more data to work with. The probing disturbances could for instance be a sinusoid with a small amplitude, added to the dancer reference position.

4.8 Adaptive Controller Structure

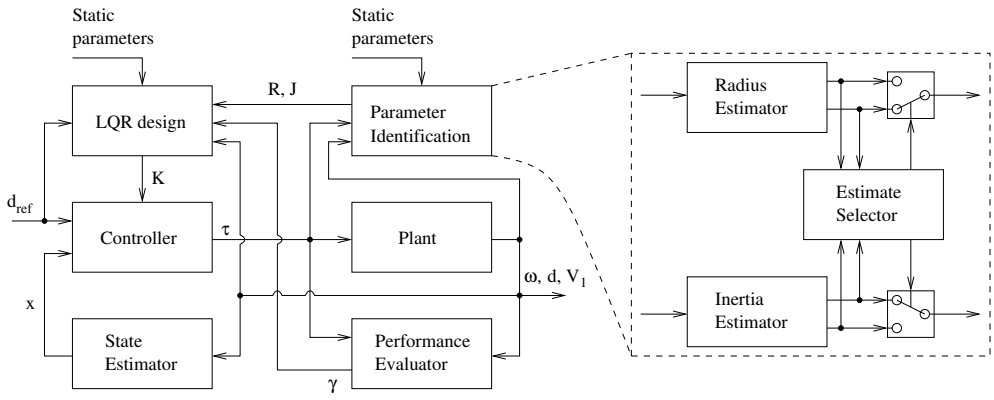


Figure 4.17: Final control structure.

The final adaptive control structure can be seen in 4.17. This is essentially a MIAC structure, with integral LQR as the control method. A V_1 speed controller and supervisor has also been implemented, and is considered as an autonomously running part of the plant that occasionally introduces disturbances to V_1.

Chapter 5

Practical Test

5.1 Implementation Issues

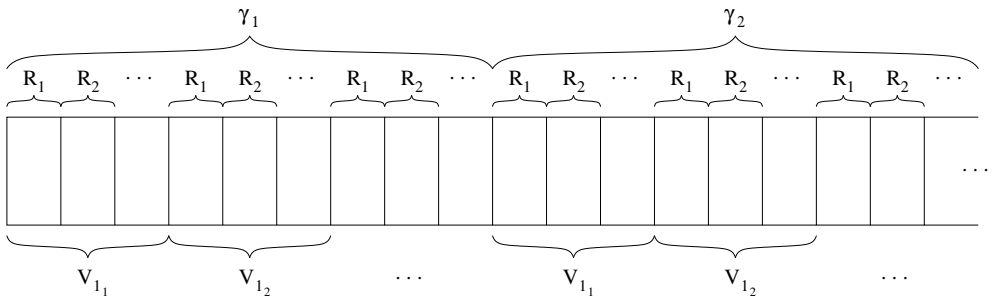


Figure 5.1: Gain scheduling lookup table

In the previous section it was described how the ARE could be used for solving the optimization problem, and this is used in all the simulations. This method does, however, require an eigenvalue decomposition which is not supported in the version of Simulink used for implementing the controller with dSpace. Therefore a lookup table will be used, containing gains corresponding to different values of V_1, R and γ. This method is also known as gain scheduling. The lookup table is illustrated in Figure 5.1 and the gains are calculated using the *LQR* function in Matlab. The updating functionality scans a set of index vectors for the nearest matching value for the parameters provided and looks up the appropriate optimal gain.

The performance evaluation algorithm has not been implemented, so the control effort is set manually in the tests. Only the velocity-based radius estimate is used

in the tests, as no estimate selector has been implemented. Additionally, only the RLS inertia estimator is implemented, as the other method is purely mathematical. The estimated inertia is not actually used for the controller design, as the plant has only minimal inertia development for a full test run.

5.2 Controller Performance

In this section the controllers performance will be tested against the requirements specification in Section 2.5. For this the reference tension is selected to 25 N. Preliminary tests showed that the errors caused by positive steps in V_1 generally have a larger impact than steps in the other direction, so only these will be shown here.

To verify Requirement 1, two tests have been carried out with a low, $\gamma = 1$, and a high, $\gamma = 25$, control effort. When using the low control effort, the tension error is about 10 % in the start of the ramping phase, after which it decreases to about 8 % as shown in Figure 5.2. The outlier in V_1 at 4 seconds and the corresponding disturbance on the tension is due to slipping of the web on the unwinder roll.

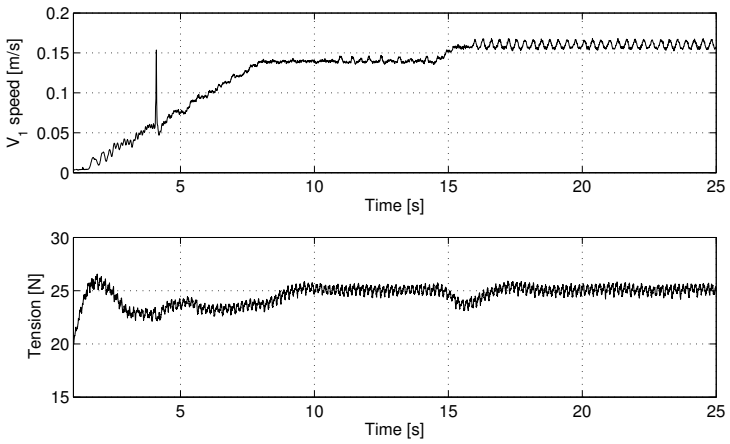


Figure 5.2: In the startup the tension error is relative high when using $\gamma = 1$.

At 83 seconds, the V_1 supervisor starts applying larger steps to V_1, which results in an error of approximately 10 %, as shown in 5.3. The requirements have therefore not been fulfilled for this control effort.

Requirement 2, about the settling time, is difficult to verify, as the measurements are almost entirely masked by noise and a steady state oscillation, as seen in Figure

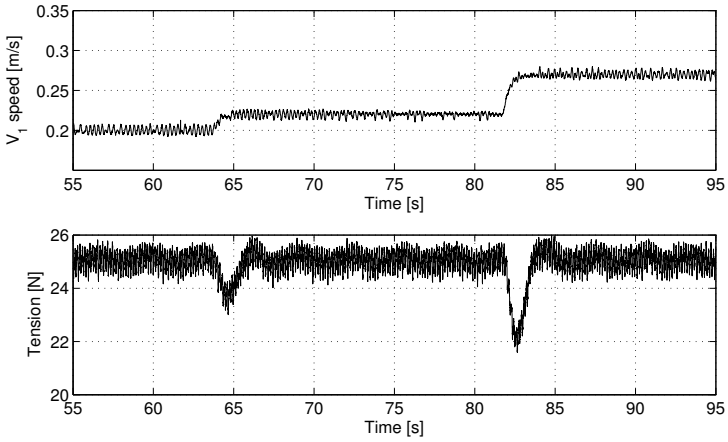


Figure 5.3: When the steps in V_1 increase, the tension error also increases. $\gamma = 1$

5.4. However, it is assessed that the tension is stabilized approximately 3 s after V_1 has ramped up, which is too slow.

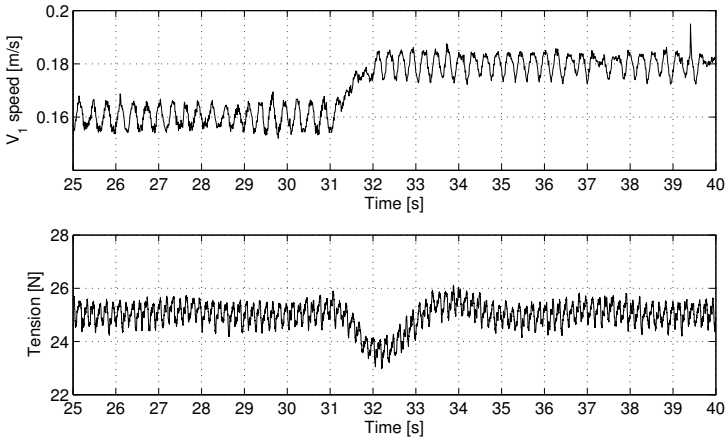


Figure 5.4: The settling time is difficult to extract as the signal is very noisy.

It is difficult to extract the steady state error for test of Requirement 3. The mean value over a second is at steady state is 25 N, but the amplitude of the steady state oscillation exceeds the 1 % allowed deviation.

Next, the control effort is increased to 25, which decreases the tension error to the acceptable level at 2 % for low steps in V_1, as shown in Figure 5.5. The larger

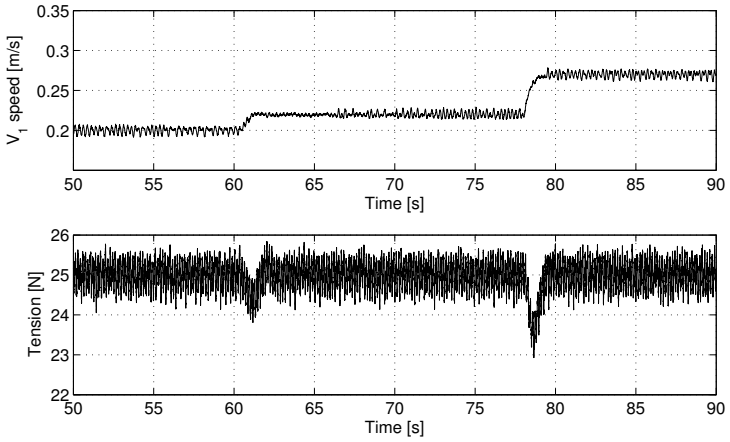


Figure 5.5: When using $\gamma = 25$ the tension error is penalized more.

step still gives an unacceptable 6 % error, but it is a definite improvement over the result with $\gamma = 1$.

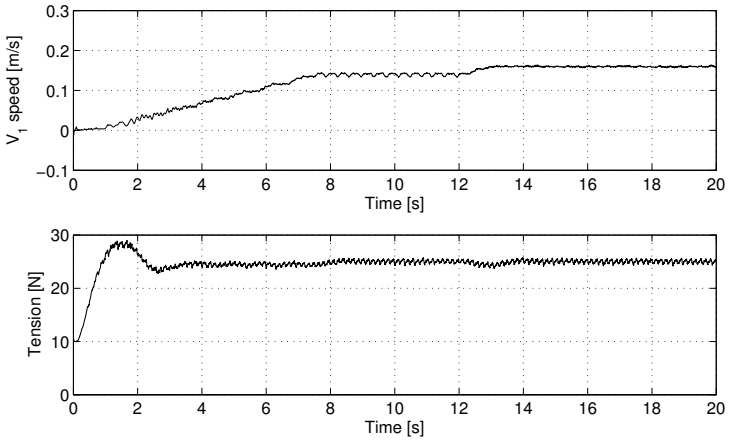


Figure 5.6: For $\gamma = 25$ the tension error is worse during startup.

At the startup the controller is now relatively more aggressive, so V ramps up faster and hence increases the initial overshoot to 6 %, as shown in Figure 5.6, because V_1 has not had a chance to start yet.

To minimize this overshoot V_1 can be started at a lower dancer position, or the control effort could be kept low until the dancer is at its starting position and the

V_1 supervisor has started.

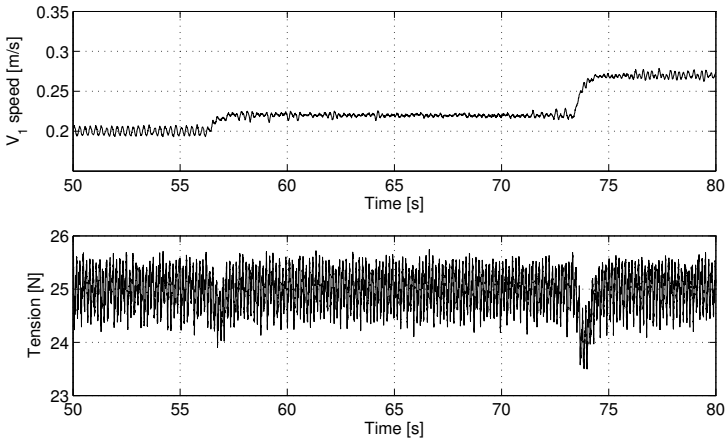


Figure 5.7: When using $\gamma = 100$, the tension error is no more than 4 %.

Figure 5.7 shows that further increasing the control effort to 100 results in a tension error of 2 % for small steps in V_1 and 4 % for larger steps in V_1. However, in the startup the initial overshoot increases to about 19 %, as shown in 5.8, further necessitating a change in the way the startup is handled.

The control inputs for the tests at $\gamma = 1$ and $\gamma = 25$ are shown in Figure 5.9. It can be seen that the larger control effort gives a control signal that varies more. At some point, increasing γ further would obviously bring the actuator into saturation, resulting in a loss of control.

As dictated by Requirement 4 the rise time for V_1 must not exceed 2 s. A test showed that this requirement cannot be fulfilled with the designed controller with the current setup of the VLT. This requirement has not been in focus in this project. Instead, the focus has been on handling disturbances during the winding phase.

5.3 Radius Estimator

The velocity based method and the coilers approximation has been tested on the plant. The velocity based method is generally more noise filled than the coilers approximation as it is directly based on the noisy measurements from V_1 and ω. When the winder roll approaches its maximum radius, V_1 and ω become slower and the noise becomes more significant, which gives a more noise filled radius

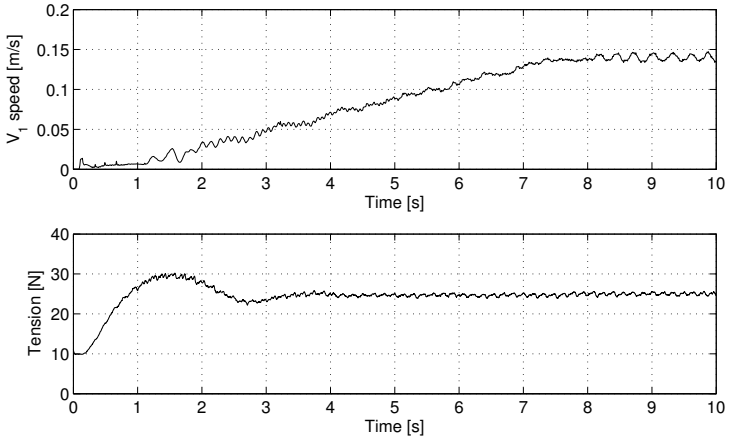


Figure 5.8: When using $\gamma = 100$, the initial overshoot becomes 19 %.

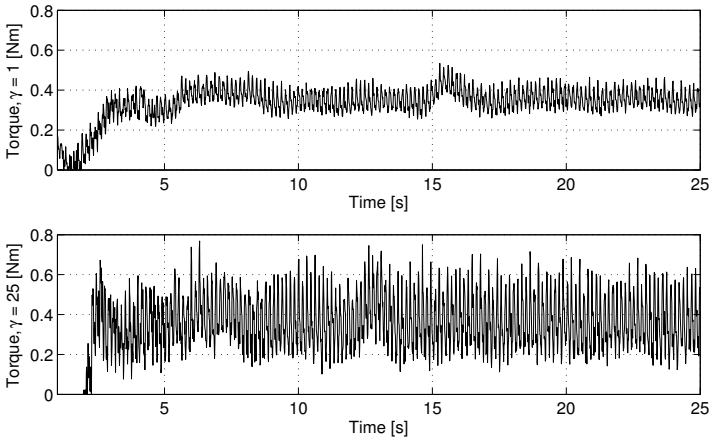


Figure 5.9: Control input at $\gamma = 1$ and $\gamma = 25$

estimate. This is shown in the upper plot in Figure 5.10. To improve the estimate, either the V_1 and ω or the estimated radius must be filtered.

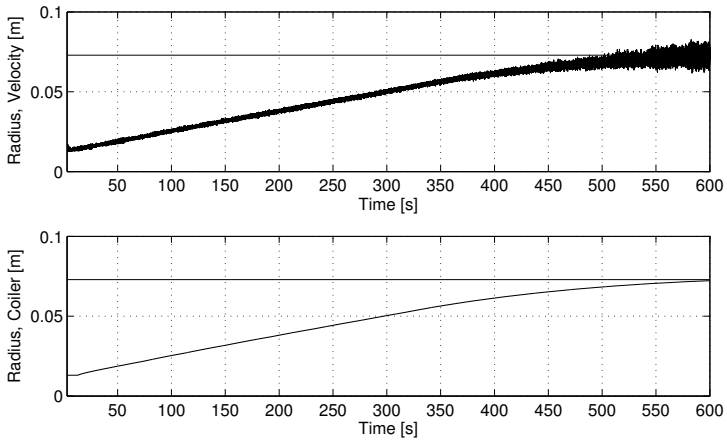


Figure 5.10: The radius is estimated using the velocity based method and the coilers approximation.

In the lower plot the radius estimated using the coilers approximation is shown. It can be seen that this is not as noisy as the velocity based method as this is based on an integration of the noisy V_1 and ω, which tends to average out measurement noise. This approximation become more accurate over time. However, this approximation does not give an accurate radius estimate in the first many seconds, because the approximation is not particularly good with materials that are very thin, relative to the core radius.

Both the estimators eventually go to the measured final roll radius, indicated with the vertically line.

Figure 5.11 shows the web thickness estimates. Like with the radius estimate, the velocity based estimate is very noisy, while the one based on the coilers approximation is initially very bad, but gradually approaches the measured web thickness. The rate of convergence could potentially be used to switch estimates, by setting the switching point when this rate falls below an appropriately small threshold.

Figure 5.12 shows the radius estimates when the system is started from a larger core radius. It can be seen that this causes the estimate based on the coilers approximation to converge more slowly, while the velocity based estimate is unaffected. Given the scale of real CDWW applications, it is very likely that both estimation methods need to be used.

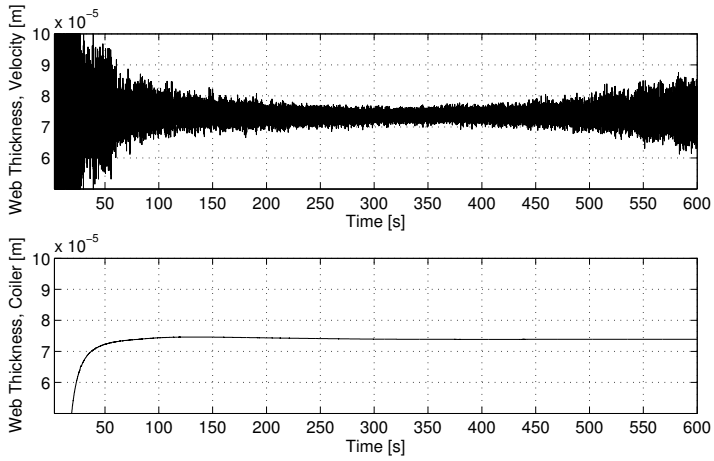


Figure 5.11: The web thickness estimated using the velocity based method and the coilers approximation.

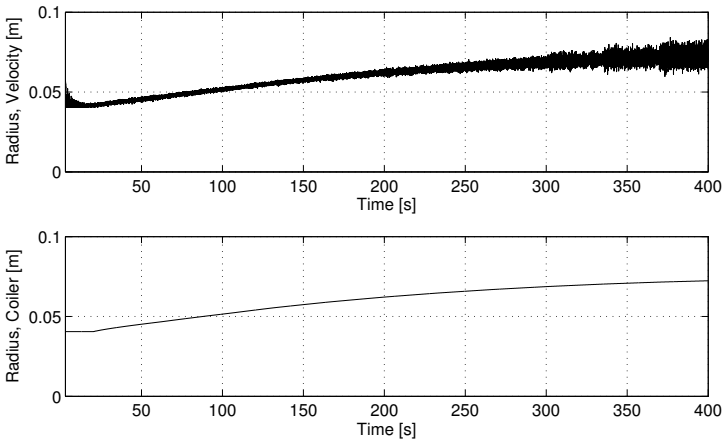


Figure 5.12: The radius estimate, when started from a larger core.

5.4 Inertia Estimator

In this section the RLS inertia estimator is tested. γ is set to 100.

Figure 5.13 shows the inertia estimate in a test where a probing signal, consisting of short pulses of added tension to the reference, is used over the whole period. The horizontal line corresponds to the calculated system inertia. In the first graph, the inputs to the RLS algorithm are unfiltered, while filtering is used in the second graph. It can be seen that the estimate with no filtering gives a poor estimate that eventually starts to drift away from the calculated system inertia. It can be seen that the estimate is much improved with filtering of the inputs

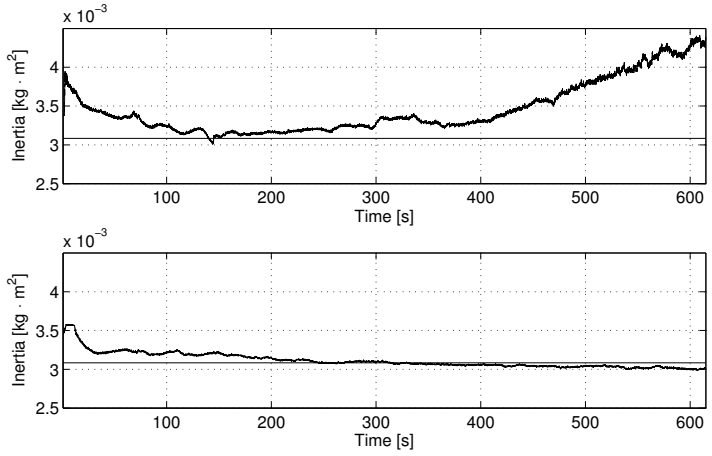


Figure 5.13: Comparison of inertia estimate without and with filtered inputs.

Figure 5.14 shows the inertia estimate with intermittent manually applied probing signals, as evidenced by the tension measurement. It can be seen that the estimate has a tendency to drift when there are no disturbances, whether they be caused by steps in V_1 or the probing signal.

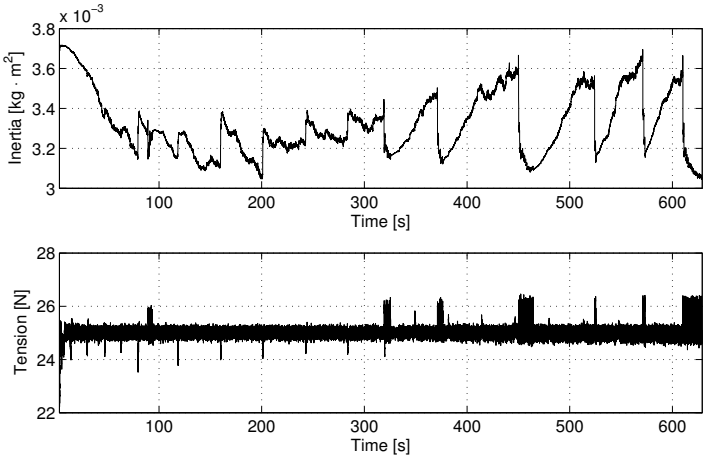


Figure 5.14: Inertia estimate with intermittent probing.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In this project, a nonlinear model of a CDWW plant with dancer feedback has been developed and implemented in Simulink. Several adaptive control methods were evaluated and a modified form of MIAC was chosen. A method was developed for designing an piecewise linear integral optimal controller online, based on known and estimated plant parameters and desired control effort.

Implementation issues prevented the automated controller designer from being ported to the test plant. Instead, a few of the more influential parameters were selected for creating a gain scheduling matrix, so that the online adaptation could still be tested.

Tests showed that the requirements for disturbance rejection were fulfilled for some values of the control effort parameter. An automated method for adjusting this parameter based on the performance has been described, but not implemented.

Tests also showed that the tension oscillates slightly around the set point at all times. Some of this may be caused by the V_1 controller, which seems unable to keep V_1 constant. It is also possible that the integrator error has not been weighted sufficiently in the $\mathbf{Q}_c$ matrix, and the controller therefore responds too slowly to minor disturbances that push the dancer away from the set point.

The V_1 controller has been designed for gradual steps, and was found to be unable to track the kind of ramp necessary to test requirement 4, so this test was not performed.

Two radius estimation techniques have been tested, and it was found that the velocity-based estimate tends to suffer more from sensor noise. This is partic-

ularly pronounced at the start and end of the winding process, where the line speed was the lowest. The method based on the approximation of the length of an Archimedes spiral shows the most promise.

The RLS based inertia estimate has been designed and tested on the plant, and it was found that the estimate tends to diverge, due to too little excitation of the plant dynamics. When a test probing signal was applied to the dancer position reference, the estimate rapidly converges to the calculated value for the plant inertia.

6.2 Future Work

The developed model shows that there are several non-linear relations between overall plant states. In this project, this was dealt with by continually updating a linear controller with estimates of how the plant is changing. It may be worth researching how other adaptive or nonlinear control methods can be applied to the problem.

A method for determining poor performance or actuator saturation should be developed, so that the control effort parameter can be updated automatically in response to the plant behavior.

The controller is currently designed for a CDWW system with dancer feedback. It should be determined if a similar adaptive method can be applied to a system with load cell feedback.

A reliable inertia estimate currently requires relatively large probing disturbances to prevent estimation drift. It should be determined if the estimator can be made to work if it is only run periodically or when other disturbances excite the dynamics. It should also be investigated if it is possible to get a reliable estimate with a less harmful probing signal than the one used for the tests.

Appendix A

Dancer Changes

A.1 Amplification Circuit

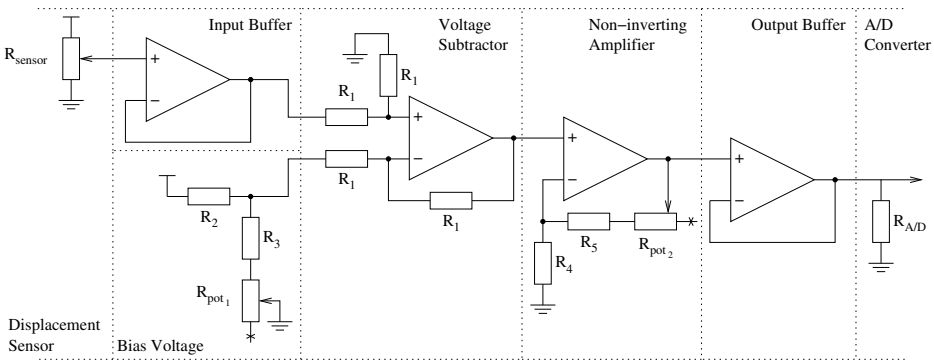


Figure A.1: Level conversion circuit.

The sensor for detecting dancer displacement is a simple 5.58 kΩ potentiometer, connected to the A/D converter of the VLT. However, due to the range of possible movement, the detected voltage range is only approximately 1 V of the full 10 V range. Furthermore, it is non-linear due to the 10 kΩ input resistance of the A/D converter.

One way to solve these problems would be to use a more appropriate potentiometer and place a buffer between the potentiometer and the A/D converter. However, it is also possible to amplify and buffer the existing signal in a way that allows fine-tuning of the desired input voltage range, without changing the potentiometer.

	R_1	R_2	R_3	R_4	R_5	R_{pot_1} / R_{pot_2}
Value	12 kΩ	1.2 kΩ	820 Ω	1 kΩ	3.3 kΩ	1 kΩ

Table A.1: Resistor values.

The circuit used to convert the voltage levels is shown in Figure A.1, with the chosen resistor values as shown in Table A.1. The circuit is described in detail in the following.

Displacement Sensor
A simple 5.58 kΩ potentiometer. Resistance varies from 2.94 kΩ to 3.84 kΩ when going from minimum to maximum dancer position.

Input Buffer
Voltage follower, to prevent the subsequent 0 from introducing non-linearity to the displacement sensor.

Bias Voltage
Adjustable offset voltage to be subtracted from the minimum output voltage of the sensor, which is approximately 5.25 V at a 10 V supply. Using a R_2, R_3, and R_{pot_1} value of 1.2 kΩ, 820 Ω, and 1.0 kΩ respectively, allows for adjusting the bias voltage between approximately 4 V and 6 V.

Voltage Subtractor
Subtracts the bias voltage from the buffered sensor input. The amplification of this signal could also be done in this step, but to make it adjustable, two resistors would have to be adjusted synchronously. Instead, the four R_1 resistors were chosen to be of the same size, to provide a unity gain on the voltage difference.

Non-inverting Amplifier
Adjustable amplification of the output of the Voltage Subtractor. Using a R_4, R_5, and R_{pot_2} value of 1.0 kΩ, 3.3 kΩ, and 1.0 kΩ respectively, it is possible to adjust the amplification between 4.3 and 5.3 times. This amplification range, in combination with the voltage subtractor allows extension of the analog range to between 1 V and 9 V with a 10 V supply.

Output Buffer
Voltage follower, to prevent any input resistance of the A/D converter from affecting the Non-inverting Amplifier.

A.2 Dancer Spring Attachment

To allow the attachment of one or more springs to the rotating dancer roll, the device depicted in Figure A.2 was manufactured and attached to the dancer roll through a ball bearing.

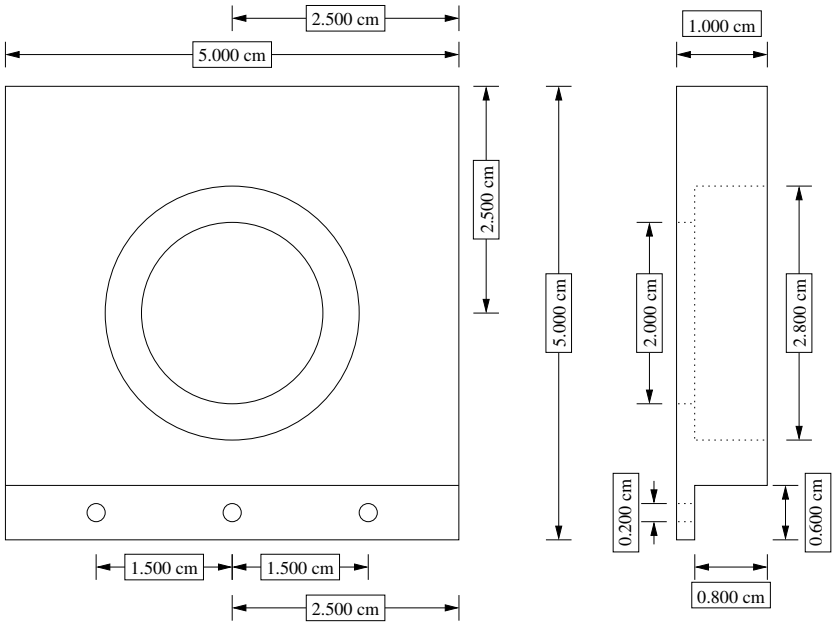


Figure A.2: Spring attachment device.

Appendix B

Nonlinear Dancer Tension

To illustrate the necessity of two idle rolls for the dancer, the tension to position relationship will be derived for the original test plant dancer.

B.1 Assumptions

- The dancer moves linearly up and down.
- The spring force is linear with respect to dancer displacement.
- The system has no friction.
- The web is moving with a constant velocity.
- Dynamics will not be considered.

B.2 Tension Calculation

Figure B.1 shows the original dancer configuration.

$$d = d_1 + d_2 = \sqrt{l_1^2 + (l_2 - x)^2} \tag{B.1}$$

$$sin(\theta) = \frac{r_1}{d_1} = \frac{r_2}{d_2} \tag{B.2}$$

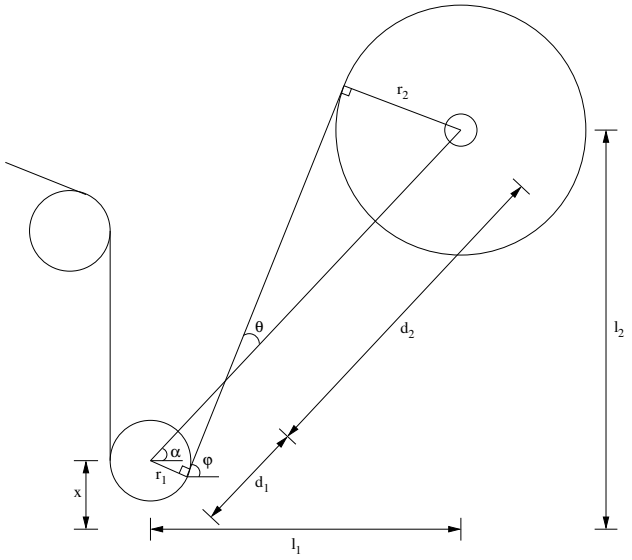


Figure B.1: Geometry of dancer and winder roll.

$$d_1 = \frac{r_1}{r_2} \cdot d_2 = \frac{r_1}{r_2} \cdot (d - d_1) \tag{B.3}$$

$$d_1 \cdot \left(1 + \frac{r_1}{r_2}\right) = \frac{r_1}{r_2} \cdot d \tag{B.4}$$

$$d_1 = \frac{\frac{r_1}{r_2}}{1 + \frac{r_1}{r_2}} \cdot \sqrt{l_1^2 + (l_2 - x)^2} = \frac{r_1}{r_2 + r_1} \cdot \sqrt{l_1^2 + (l_2 - x)^2} \tag{B.5}$$

$$sin(\theta) = \frac{r_1}{d_1} = \frac{r_1 + r_2}{\sqrt{l_1^2 + (l_2 - x)^2}} \tag{B.6}$$

$$tan(\alpha) = \frac{l_2 - x}{l_1} \tag{B.7}$$

$$\varphi = \alpha + \theta = tan^{-1}\left(\frac{l_2 - x}{l_1}\right) + sin^{-1}\left(\frac{r_1 + r_2}{\sqrt{l_1^2 + (l_2 - x)^2}}\right) \tag{B.8}$$

It can be seen that the dancer wrap angle will vary with both the current dancer position and the amount of material rolled up on the winder roll.

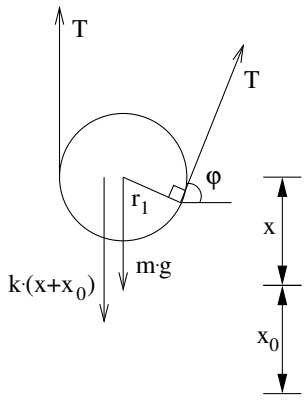


Figure B.2: Force diagram for dancer.

The tension as a function of dancer position and winder roll radius can be found by analysis of the force diagram in Figure B.2.

$$m \cdot g + k \cdot (x + x_0) = T \cdot (1 + sin(\varphi)) \tag{B.9}$$

$$T = \frac{m \cdot g + k \cdot (x + x_0)}{1 + sin(\varphi)} \tag{B.10}$$

$$T = \frac{m \cdot g + k \cdot (x + x_0)}{1 + sin\left(tan^{-1}\left(\frac{l_2 - x}{l_1}\right) + sin^{-1}\left(\frac{r_1 + r_2}{\sqrt{l_1^2 + (l_2 - x)^2}}\right)\right)} \tag{B.11}$$

Such a nonlinear relationship would needlessly complicate the use of the dancer for feedback, so an extra idle roll will be introduced to ensure a fixed wrap angle.

Appendix C

Inertia in the Plant

In the following the inertia will be found for all the major rotating parts in the plant, shown in Figure C.1. The letters A to D refer to the gear wheels. Later on the total inertia is projected to the motor shaft.

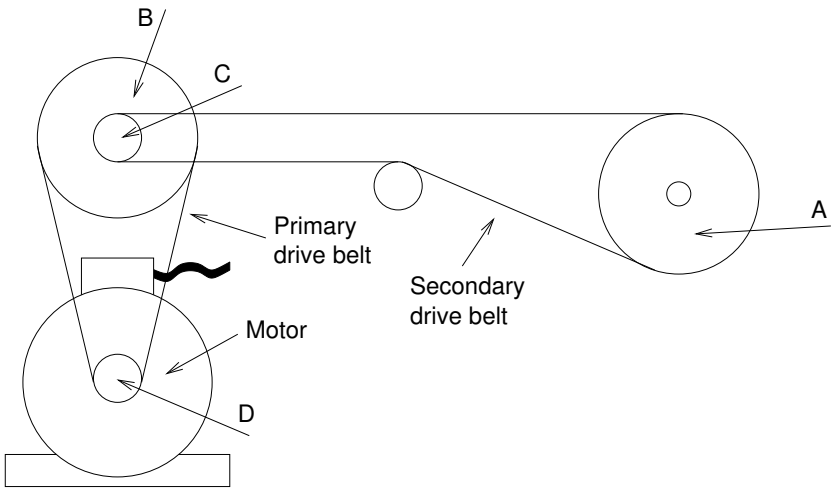


Figure C.1: The motor shaft is affected by the inertia of the rotating parts in the plant.

C.1 Motor

The inertia for the motor is given by Danfoss:

$$J_M = 1.4000 \cdot 10^{-3} \ \left[kg \cdot m^2\right] \tag{C.1}$$

C.2 Gear Wheel

The inertia of the gear wheels are calculated separately using the same equation as previously. The gear wheels, are labeled A, B, C, and D, respectively, as seen Figure C.1. The small center gear wheel, between A and B, is not considered as this only has insignificant effect on the total inertia.

C.2.1 Gear Wheel A

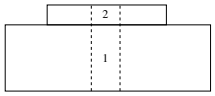


Figure C.2: Gear wheel A is placed at the same shaft as the winder roll. The drive belt touches cylinder 1.

Gear wheel A is modeled as two composed cylinders as depicted in Figure C.2. Their inertia are as following:

$$J_{GWA1} = \frac{1}{2} \cdot 0.5293 \cdot ((6.00 \cdot 10^{-3})^2 + (46.00 \cdot 10^{-3})^2) = 0.5695 \cdot 10^{-3} \ \left[kg \cdot m^2\right] \tag{C.2}$$

$$J_{GWA2} = \frac{1}{2} \cdot 0.1003 \cdot ((6.00 \cdot 10^{-3})^2 + (26.00 \cdot 10^{-3})^2) = 0.1855 \cdot 10^{-3} \ \left[kg \cdot m^2\right] \tag{C.3}$$

The inertia of the entire gear wheel is then found:

$$J_{GWA} = J_{GWA1} + J_{GWA2} = 0.6349 \cdot 10^{-3} \ \left[kg \cdot m^2\right] \tag{C.4}$$

C.2.2 Gear Wheel B

This gear wheel is considered as one cylinder and the inertia is then calculated as:

$$J_{GWB} = \frac{1}{2} \cdot 0.0597 \cdot ((6.00 \cdot 10^{-3})^2 + (16.00 \cdot 10^{-3})^2) = 0.0087 \cdot 10^{-3} \ \left[kg \cdot m^2\right] \tag{C.5}$$

C.2.3 Gear Wheel C

This gear wheel is also modeled as two composed cylinders with the following inertia.

$$J_{GWC1} = \frac{1}{2} \cdot 0.3871 \cdot ((6.00 \cdot 10^{-3})^2 + (47.00 \cdot 10^{-3})^2) = 0.4345 \cdot 10^{-3} \ \left[kg \cdot m^2\right] \tag{C.6}$$

$$J_{GWC2} = \frac{1}{2} \cdot 0.0465 \cdot ((6.00 \cdot 10^{-3})^2 + (25.00 \cdot 10^{-3})^2) = 0.0154 \cdot 10^{-3} \ \left[kg \cdot m^2\right] \tag{C.7}$$

The inertia of the entire gear wheel is then calculated as:

$$J_{GWC} = J_{GWC1} + J_{GWC2} = 0.4490 \cdot 10^{-3} \ \left[kg \cdot m^2\right] \tag{C.8}$$

C.2.4 Gear Wheel D

Gear Wheel D is mounted on the motor and is composed of an aluminium and a steel part, which are shown in Figure C.3 and Figure C.4. The inertia for both of them are calculated in the following by using Equation C.9 for a cylinder.

$$J = \frac{1}{2} m (r_0^2 + r_1^2) \ \left[kg \cdot m^2\right] \tag{C.9}$$

The mass, m, is found using the equation for finding the volume of a cylinder multiplied with the materials density. This equation is also used in the remaining part.

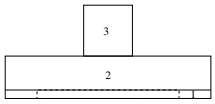


Figure C.3: The aluminium part of the gear wheel D at the motor shaft. The drive belt touched cylinder 1.

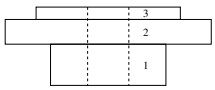


Figure C.4: The steel part of the gear wheel D at the motor shaft.

Aluminium Part

Likewise, the aluminium part, shown in Figure C.3, is divided into three cylinders. The inertia for each of them are then calculated in the following.

$$J_{GWDA1} = \frac{1}{2} \cdot 0.0504 \cdot ((35.00 \cdot 10^{-3})^2 + (49.90 \cdot 10^{-3})^2) = 0.0937 \cdot 10^{-3} \ \left[kg \cdot m^2\right] \tag{C.10}$$

$$J_{GWDA2} = \frac{1}{2} \cdot 0.3223 \cdot ((2.50 \cdot 10^{-3})^2 + (49.90 \cdot 10^{-3})^2 = 0.4032 \cdot 10^{-3} \ \left[kg \cdot m^2\right] \tag{C.11}$$

$$J_{GWDA3} = \frac{1}{2} \cdot 0.0280 \cdot ((2.50 \cdot 10^{-3})^2 + (12.00 \cdot 10^{-3})^2) = 0.0021 \cdot 10^{-3} \ \left[kg \cdot m^2\right] \tag{C.12}$$

The inertia for the aluminium part is:

$$J_{GWDA} = J_{GWDA1} + J_{GWDA2} + J_{GWDA3} = 0.4981 \cdot 10^{-3} \ \left[kg \cdot m^2\right] \tag{C.13}$$

Steel Part

The steel part of the gear wheel D is shown in Figure C.4. The dotted lines indicates the center hole for the motor shaft. The figure shows how this part is

divided into three cylinders denoted 1, 2 and 3, from which three different inertia are calculated. The inertia for the three parts are calculated in the following:

$$J_{GWDS1} = \frac{1}{2} \cdot 0.2953 \cdot ((9.00 \cdot 10^{-3})^2 + (27.00 \cdot 10^{-3})^2) = 0.1196 \cdot 10^{-3} \ \left[kg \cdot m^2\right] \tag{C.14}$$

$$J_{GWDS2} = \frac{1}{2} \cdot 0.7113 \cdot ((9.00 \cdot 10^{-3})^2 + (50.00 \cdot 10^{-3})^2) = 0.9180 \cdot 10^{-3} \ \left[kg \cdot m^2\right] \tag{C.15}$$

$$J_{GWDS3} = \frac{1}{2} \cdot 0.1161 \cdot ((9.00 \cdot 10^{-3})^2 + (35.00 \cdot 10^{-3})^2) = 0.0759 \cdot 10^{-3} \ \left[kg \cdot m^2\right] \tag{C.16}$$

The inertia for the steel part is:

$$J_{IBS} = J_{IBS1} + J_{GWDS2} + J_{GWDS3} = 1.11 \cdot 10^{-3} \ \left[kg \cdot m^2\right] \tag{C.17}$$

Steel and Aluminum Part

The inertia for the steel part and the aluminium part constitutes the total inertia for gear wheel D:

$$J_{IB} = J_{GWDA} + J_{GWDS} = 1.60 \cdot 10^{-3} \ \left[kg \cdot m^2\right] \tag{C.18}$$

C.3 Winder Roll

The inertia of the winder roll is composed of the inertia of the core, the web and the roller sides. The inertia of the web is changing as the web radius changes.

C.3.1 Core

The inertia of the core is calculated as:

$$J_{core} = \frac{1}{2} \cdot 0.0254 \cdot ((6 \cdot 10^{-3})^2 + (13 \cdot 10^{-3})^2) = 0.0026 \cdot 10^{-3} \ \left[kg \cdot m^2\right] \tag{C.19}$$

C.3.2 Web

The inertia of the web is calculated using the same equation as previously.

$$J_{paper} = \frac{1}{2}m(r_0^2 + r_1^2) \ \left[kg \cdot m^2\right] \tag{C.20}$$

The mass, m, is the product of the cross-sectional area of the web, the wound web length and the density of the web.

$$m = A \cdot l \cdot \rho \tag{C.21}$$

The radius, r_1, is approximated to the integral of the web thickness divided with 2π and the rotational velocity of the roll. The rotational velocity is also defined as the web velocity divided with the radius of the winder roll.

$$r_1 = \int \frac{\lambda}{2\pi}\omega \tag{C.22}$$

C.3.3 Roller Sides

The inertia of the two identical roller sides are calculated as follows:

$$J_{RS} = 2 \cdot \frac{1}{2} \cdot 0.3535 \cdot ((6.00 \cdot 10^{-3})^2 + (100.00 \cdot 10^{-3})^2) = 3.5000 \cdot 10^{-3} \ \left[kg \cdot m^2\right] \tag{C.23}$$

C.4 Total Inertia

The inertia from the different parts must all be projected to the motor shaft through the two gears in the drive chain. The inertia is projected by dividing it by the gear ratio as in Equation C.24.

$$J = \frac{J_1}{N^2} \tag{C.24}$$

First, the inertia for the rotating parts at gear wheel A are projected to the shaft at

gear wheel B and C in the drive train. The gear ratio is $N_{horizontal} = 3$.

$$J_1 = \frac{J_{RS} + J_{core} + J_{paper} + J_{GWA}}{N_{horizontal}^2} = 0.4597 \cdot 10^{-3} + 0.1111 J_{paper} \ \left[kg \cdot m^2\right] \tag{C.25}$$

Then, the inertia, J_1, is added to the inertia of gear wheel B and C and the sum is then projected to the motor shaft through the second part of the drive train. The gear ratio of this gear is $N_{vertical} = 3.58$.

$$J_2 = \frac{J_1 + J_{GWB} + J_{GWC}}{N_{vertical}^2} = 0.0716 \cdot 10^{-3} + 8.6690 \cdot 10^{-3} J_{paper} \ \left[kg \cdot m^2\right] \tag{C.26}$$

Finally, the inertia, J_2, is added to the inertia of the inertia block, gear wheel D at the motor, and the motor itself.

$$J_3 = J_2 + J_M + J_{GWD} = 3.0830 \cdot 10^{-3} + 8.669 \cdot 10^{-3} J_{paper} \ \left[kg \cdot m^2\right] \tag{C.27}$$

The plot in Figure C.5 shows the development in the inertia when the winder rolls diameter increases at maximum web velocity. It is seen the generated inertia from the winded web is minor related to the inertia from the other rotating parts in the plant. Therefore, the inertia generated from the winded web, J_{paper}, is assumed zero. Hence, the total inertia at the motor shaft becomes a constant:

$$J = 3.0830 \cdot 10^{-3} \ \left[kg \cdot m^2\right] \tag{C.28}$$

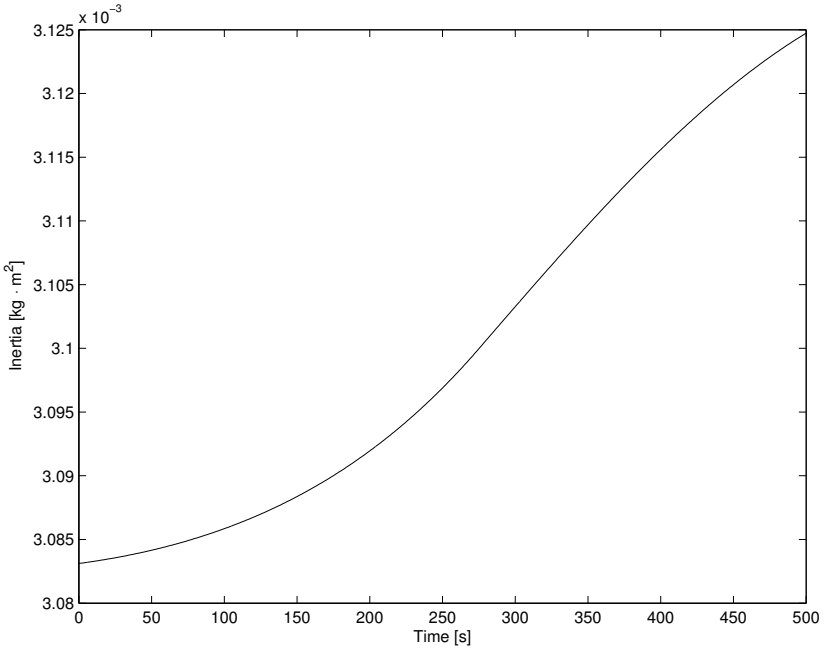


Figure C.5: The total inertia in the plant.

Appendix D

Simulink Model

In this section the CDWW model developed in Simulink will be described. Figure D.1 shows the overall model with three inputs and four outputs. The inputs are the reference torque, F_x and the line speed. The outputs are the motor velocity, tension, dancer velocity and dancer position.

D.0.1 CDWW Model

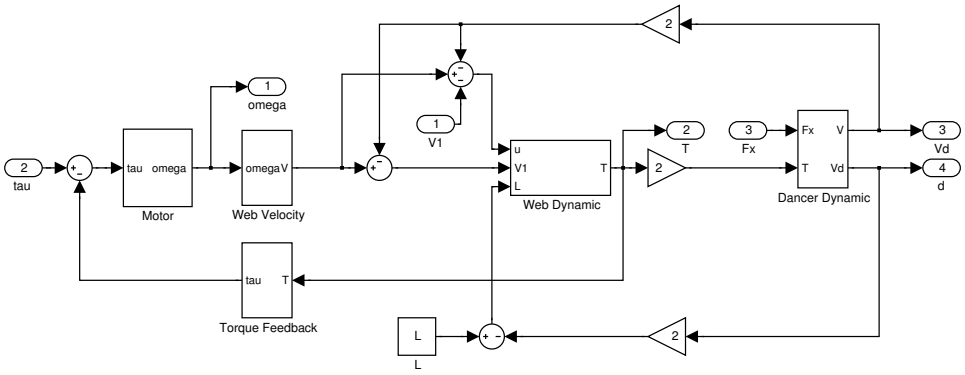


Figure D.1: The overall CDWW model representation in Simulink.

The first subsystem is the motor model given in Equation 3.29 which represents the relation between the motors rotational velocity, ω, and the control signal, τ. This signal is given as the reference signal and the torque feedback calculated from the tension of the web. Next, Equation 3.30 is applied to ω from which the web velocity at the winder roll, V, is obtained. V is then fed into the next subsystem,

which calculates the tension. Figure D.2 shows the elaborated subsystem, which corresponds to Equation 3.25. This is the model of the web dynamics before some of the simplification assumptions are taken. Compared to the simplified model, this model also takes $L-2d$ and $V-2V_d-V_1$ as inputs.

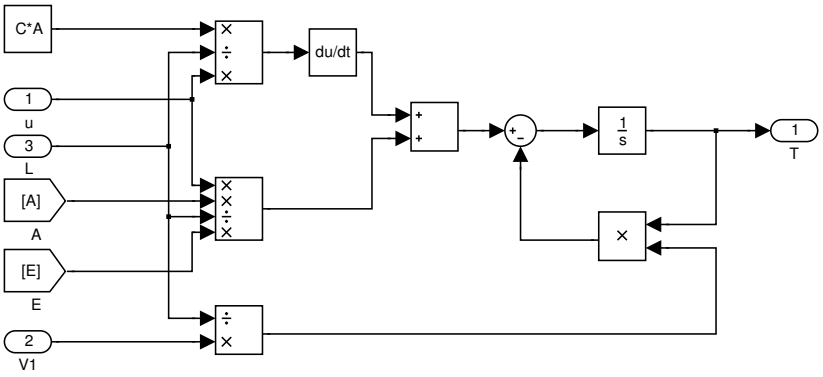


Figure D.2: The overall CDWW model representation in Simulink.

In Figure D.2 the web cross sectional area and E-modulus are considered as constants. The output tension is then multiplied by two to give the total web tension affecting the dancer. This, together with F_x is given as input to the next block that calculates the dancer position and velocity. This block corresponds to Equation 3.28.

Appendix E

Simulation Constants

Symbol	Value	Origin
A	$4.35 \cdot 10^{-6}\ m^2$	Measured
B_d	$15\ kg/s$	Estimated from spring dampening test
B_m	$0.5 \cdot 10^{-3}\ {}^{Nm}/_{rad/s}$	Estimated from closed loop motor test
d_{ref}	$25.9 \cdot 10^{-3}\ m$	Calculated
E	$8 \cdot 10^{9}\ Pa$	Commonly used value for newsprint
F_x	$-22\ N$	Measured
G	$0.075 \cdot 10^{-3}\ m$	Measured
J	$3.09 \cdot 10^{-3}\ kgm^2$	Calculated
K_d	$1080\ {}^{N}/_{m}$	Measured
L	$0.55\ m$	Measured
L_N	$0.4981\ m$	Calculated
m_d	$0.6\ kg$	Measured
N	10.75	Measured
R_{core}	$13 \cdot 10^{-3}\ m$	Measured
T_{ref}	$25\ N$	Defined

Appendix F

Algebraic Riccati Equation

The Algebraic Riccati Equation for optimal controller design in continuous time systems can be written as:

$$\dot{\mathbf{S}} = \mathbf{A}^T\mathbf{S} + \mathbf{S}\mathbf{A} - \mathbf{S}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S} + \mathbf{Q} \tag{F.1}$$

From this a Hamiltonian matrix can be constructed, which has the property that if λ is an eigenvalue of H, then $-\lambda$ is also an eigenvalue:

$$\mathbf{H} = \begin{bmatrix} \mathbf{A} & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ -\mathbf{Q} & -\mathbf{A}^T \end{bmatrix} \tag{F.2}$$

If X is a set of eigenvectors of H, belonging to the diagonal eigenvalue matrix, Λ, then the eigenvalue equation is:

$$\mathbf{H}\mathbf{X} = \mathbf{X}\Lambda \tag{F.3}$$

Some rearrangement gives:

$$\mathbf{H}\begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \end{bmatrix}\Lambda \tag{F.4}$$

$$\mathbf{H}\begin{bmatrix} \mathbf{I} \\ \mathbf{S} \end{bmatrix}\mathbf{X}_1 = \begin{bmatrix} \mathbf{I} \\ \mathbf{S} \end{bmatrix}\mathbf{X}_1\Lambda \tag{F.5}$$

$$\mathbf{S} = \mathbf{X}_2\mathbf{X}_1^{-1} \tag{F.6}$$

$$\mathbf{H}\begin{bmatrix}\mathbf{I}\\ \mathbf{S}\end{bmatrix} = \begin{bmatrix}\mathbf{I}\\ \mathbf{S}\end{bmatrix}\mathbf{X}_1\Lambda\mathbf{X}_1^{-1} \tag{F.7}$$

$$\mathbf{Z} = \mathbf{X}_1\Lambda\mathbf{X}_1^{-1} \tag{F.8}$$

Since $\mathbf{Z}$ is a similarity transformation of Λ, they have the same eigenvalues.

$$\begin{bmatrix}\mathbf{A} & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\\ -\mathbf{Q} & -\mathbf{A}^T\end{bmatrix}\begin{bmatrix}\mathbf{I}\\ \mathbf{S}\end{bmatrix} = \begin{bmatrix}\mathbf{Z}\\ \mathbf{S}\mathbf{Z}\end{bmatrix} \tag{F.9}$$

$$\begin{cases}\mathbf{A}-\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S} &= \mathbf{Z}\\ -\mathbf{Q}-\mathbf{A}^T\mathbf{S} &= \mathbf{S}\mathbf{Z}\end{cases} \tag{F.10}$$

Isolating $\mathbf{Z}$ and rearranging gives:

$$-\mathbf{Q}-\mathbf{A}^T\mathbf{S} = \mathbf{S}\left(\mathbf{A}-\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}\right) \tag{F.11}$$

$$\mathbf{0} = \mathbf{A}^T\mathbf{S}+\mathbf{S}\mathbf{A}-\mathbf{S}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}+\mathbf{Q} \tag{F.12}$$

This proves that $\mathbf{S}$ is a stabilizing solution to the algebraic Riccati equation.

If the state feedback gain is defined as,

$$\mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{S} \tag{F.13}$$

then the closed loop system, $\mathbf{Z} = \mathbf{A}-\mathbf{B}\mathbf{K}$, will have the eigenvalues of $\mathbf{H}$ which were chosen for constructing Λ and $\mathbf{X}$. A stabilizing optimal gain can be obtained by selecting only the eigenvalues with negative real part from $\mathbf{H}$.

$\mathbf{S}$ is supposed to be real symmetric matrix, but rounding errors may introduce non-symmetry, so a practical solution should enforce symmetry and realness:

$$\mathbf{S} = Re\left(\frac{1}{2}\left(\mathbf{S}+\mathbf{S}^*\right)\right) \tag{F.14}$$

Appendix G

Parameter Influence Analysis

In this appendix the result of five different tests are shown to illustrate what happens to the poles in the system when different parameters are changed. Also, a simulation of a badly optimized controller is shown. In table G the respective parameters and how much they are changed in the tests is shown. Only one parameter is changed for every test, while the rest remain at the nominal values shown in Appendix E.

The results are shown in Figure G.1 to Figure G.7 where only one pair of each complex conjugate pole pair is shown. The arrows shows in which direction the poles are moved by the increasing parameter. Generally, the poles are not affected significantly by the varying parameters.

One notable result, is the fourth test, which shows that the pole nearest the imaginary axis is moved closer to zero when the inertia is increased. If the controller is designed for a plant with a much smaller inertia than what is actually the case, this pole may go towards the RHP of the coordinate system, as shown in Figure G.5. The effect of this can be poor performance or even instability. This has been tested in the Simulink model, with the results shown in Figure G.6 and G.7.

Test nr.	Figure	Parameter	Range
1	G.1	V_1	0.1 - 1
2	G.2	E	$2 \cdot 10^9$ - $12 \cdot 10^9$
3	G.3	L	0.55 - 2.00
4	G.4	J	$0.309 \cdot 10^{-2}$ - 0.309
5	G.5	J	$0.309 \cdot 10^{-2}$ - 0.309

Table G.1: List of tests.

During the first two seconds the controller is designed with the actually inertia for the plant, resulting in correct and stable tracking of the set point. At two seconds the plant inertia is stepped up by a factor of ten, without updating the controller. It is seen that the system becomes unstable as a result. Eventually the control signal saturates, resulting in an a constant oscillation.

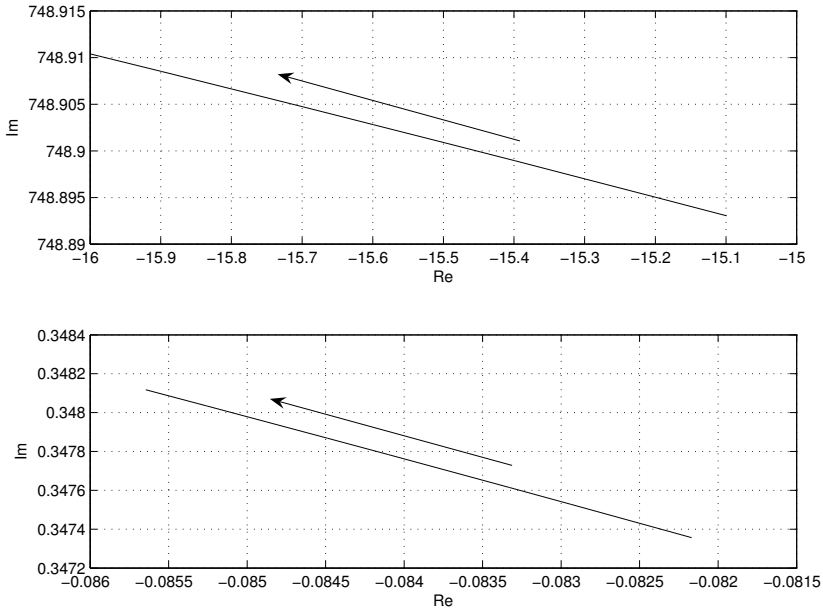


Figure G.1: The pole movement with varying line speed.

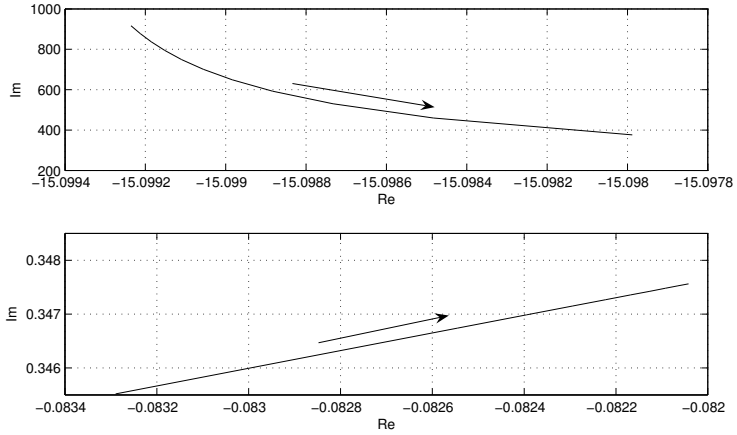


Figure G.2: The pole movement with varying E-modulus.

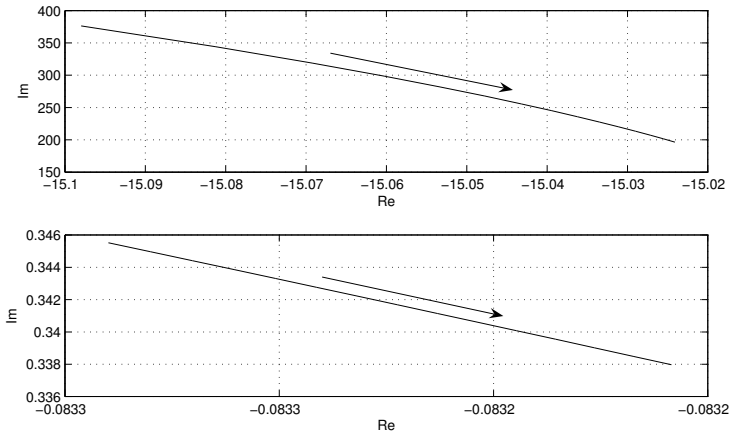


Figure G.3: The pole movement with varying web length.

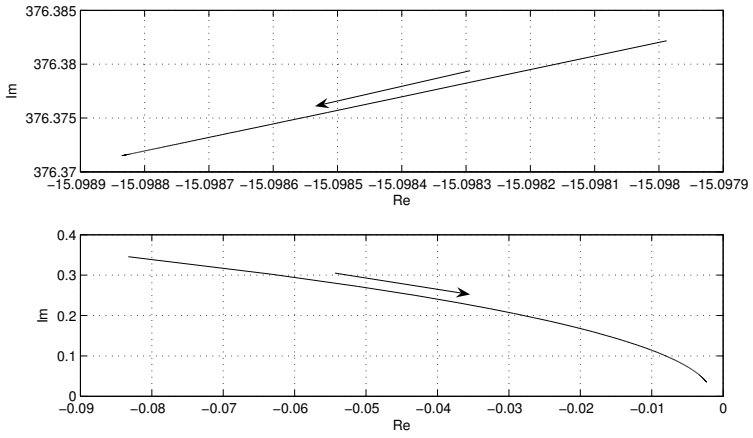


Figure G.4: The pole movement with varying inertia.

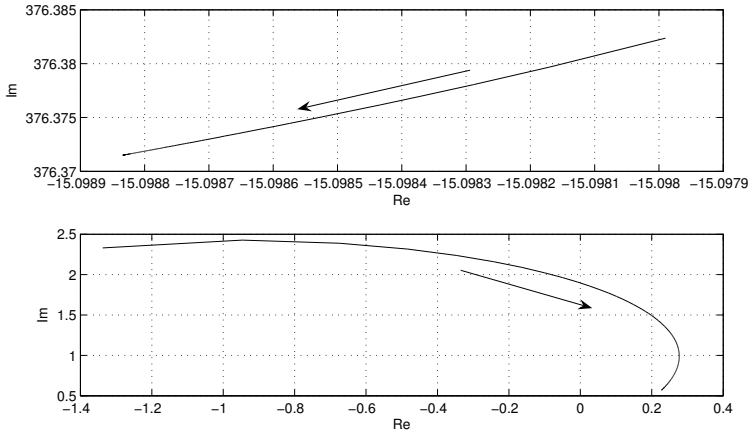


Figure G.5: Some poles go to the RHP when the plant inertia is much larger than the controller is designed for.

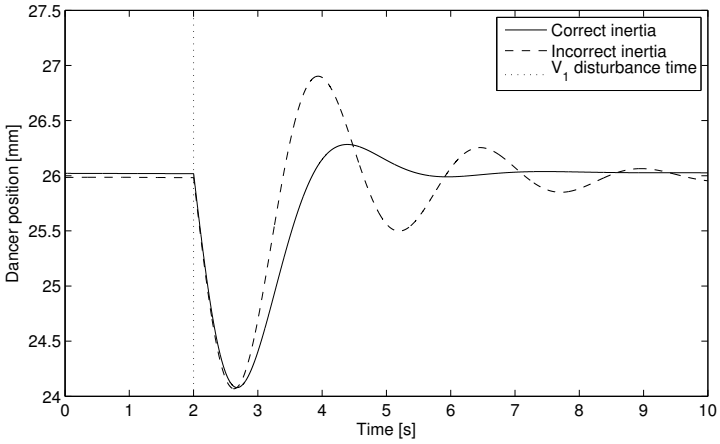


Figure G.6: Poor disturbance response with controller designed for lower inertia.

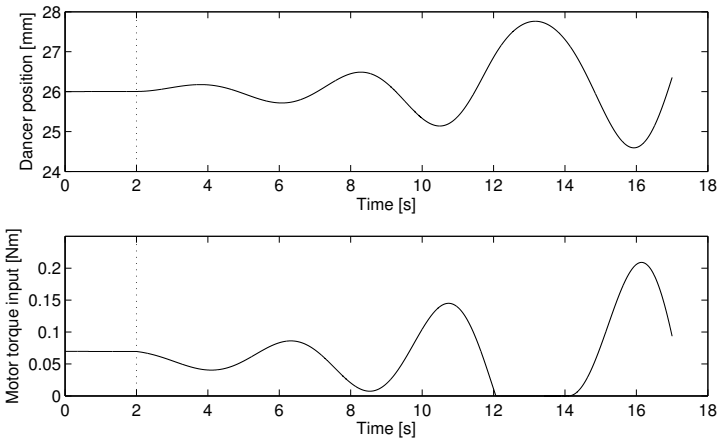


Figure G.7: The plant inertia is stepped up at two seconds without updating the controller, resulting in instability.

Bibliography

[AW95] Karl J. Åström and Björn Wittenmark. *Adaptive Control*. Addison-Wesley Publishing Company, 2nd edition, 1995.

[Dam04] Jeff Damour. The Mechanics of Tension Control, 2004.

[FPEN02] Gene F. Franklin, J. David Powell, and Abbas Emami-Naeini. *Feedback Control of Dynamic Systems*. Prentice Hall, 4nd edition, 2002.

[FPW98] Gene F. Franklin, J. David Powell, and Michael Workman. *Digital Control of Dynamic Systems*. Addison-Wesley Publishing Company, 3rd edition, 1998.

[Liu98] Zhijun Liu. A Frequency Response Based Adaptive Control for Center-Driven Web Winders. In *American Control Conference*, volume 3, pages 1493–1497, 1998.

[Liu99] Zhijun Liu. Dynamic Analysis of Center-Driven Web Winder controls. In *Industry Applications Conference*, volume 2, pages 1388–1396, 1999.

[Lju99] Lennart Ljung. *System Identification*. Prentice Hall, 2nd edition, 1999.

[Roy01] David Roylance. Engineering Viscoelasticity, 2001.

[SF95] Tetsuzo Sakamoto and Yoshikazu Fujino. Modeling and Analysis of a Web Tension Control System. In *Industrial Electronics*, volume 1, pages 358–362, 1995.

[SSJH02] Raymond T. Stefani, Bahram Shahian, Clement J. Savant Jr., and Gene H. Hostetter. *Design of Feedback Control Systems*. Oxford University Press, 4nd edition, 2002.

[Tre99] D. Alan Tree. Web Handling Research Center (WHRC), 1999. http://www.osu-ours.okstate.edu/report98/CEAT/ceatwhrc.html [Downloaded 01-05-2007].