

Institutionen för systemteknik

Department of Electrical Engineering

Examensarbete

Data Requirements for a Look-Ahead System

Examensarbete utfört i Fordonssystem
vid Tekniska högskolan i Linköping
av

Erik Holma

LITH-ISY-EX--07/4002--SE

Linköping 2007



Linköpings universitet
TEKNISKA HÖGSKOLAN

Department of Electrical Engineering
Linköpings universitet
SE-581 83 Linköping, Sweden

Linköpings tekniska högskola
Linköpings universitet
581 83 Linköping

Data Requirements for a Look-Ahead System

Master's thesis

performed in **Vehicular Systems,**
Dept. of Electrical Engineering
at **Linköpings universitet**

by **Erik Holma**

Reg nr: LiTH-ISY-EX--07/4002--SE

Supervisor: **Ph.D. Student Maria Ivarsson**
Scania / Linköpings Universitet

Examiner: **Professor Lars Nielsen**
Linköpings Universitet

Södertälje, September 23, 2007

Abstract

Look ahead cruise control deals with the concept of using recorded topographic road data combined with a GPS to control vehicle speed. The purpose of this is to save fuel without a change in travel time for a given road. This thesis explores the sensitivity of different disturbances for look ahead systems. Two different systems are investigated, one using a simple precalculated speed trajectory without feedback and the second based upon a model predictive control scheme with dynamic programming as optimizing algorithm.

Defect input data like bad positioning, disturbed angle data, faults in mass estimation and wrong wheel radius are discussed in this thesis. Also some investigations of errors in the environmental model for the systems are done.

Simulations over real road profiles with two different types of quantization of the road slope data are done. Results from quantization of the angle data in the system are important since quantization will be unavoidable in an implementation of a topographic road map.

The results from the simulations shows that disturbance of the fictive road profiles used results in quite large deviations from the optimal case. For the recorded real road sections however the differences are close to zero. Finally conclusions of how large deviations from real world data a look ahead system can tolerate are drawn.

Keywords: Look Ahead, Cruise Controller, MPC, Topographic Road map, Input data

Preface

With this thesis I complete my studies for a Master of Science degree in Applied Physics and Electrical Engineering International German. It has been an interesting thesis work and a time which I have enjoyed very much. Previous works have shown that by the use of GPS, a topographic road map and control theory it is possible to control vehicle speed over the topography to save fuel. My master thesis focuses on exploring the effects of non optimal input data to a look ahead system and set requirements on input data to the control system.

Thesis outline

The purpose of this thesis is to outline the sensitivity of a look ahead system and put requirements on input data to the system. Also the effects of disturbed input data are investigated. The purpose and method are closer handled in the introductory Chapter 1. The vehicle model used is presented in Chapter 2. Chapter 3 is devoted to explain how the optimal speed profiles look like and the gain of using these. In Chapter 4 results from simulations of the optimal speed trajectories calculated with flawed input data are presented. Chapter 5 is used to explain how the model predictive control and dynamic programming in DP-tool works. Results from simulations with DP-tool are presented in chapter 6. Finally in Chapter 7 conclusions are drawn. Extensions and future work to this thesis are discussed in Chapter 8.

Acknowledgment

First I want to express my deepest gratitude to my dedicated supervisor at Scania CV, Maria Ivarsson. Henrik Pettersson at Scania CV and Erik Hellström at Vehicular Systems at Linköping University are acknowledged for their help with the final steps of the report. I also want to acknowledge my examining professor Lars Nielsen at Vehicular Systems at Linköping University. The Staff at NEC at Scania CV are acknowledged for inspiring discussions and for letting me become a part of the group. I also want to thank my fellow master thesis workers Peter Andersson and Anders Zakrisson for interesting and humorous discussions. I want to thank Erik Hellström for all help regarding DP-tool and also for his works on this subject since it is the ground for a large part of this thesis. Anders Fröberg is also acknowledged for his works on the subject. Finally i want to express my gratitude for the support from family, friends and Helena.

Contents

Abstract	v
Preface and Acknowledgment	vii
1 Introduction	1
1.1 Thesis Objectives	1
1.2 Method	2
2 Vehicle Model	3
2.1 Engine	3
2.2 Driveline	7
2.3 Gearbox	8
2.4 Forces	8
2.5 Complete Driveline	9
2.6 Fuel Consumption	10
2.7 Fuel and Time	11
2.7.1 One Delta Function	11
3 Analytically Derived Optimal Speed Profiles	13
3.1 Optimal Speed Profiles on Level Road and Small Gradients	14
3.2 Optimal Speed Profile for Steep Downhill Slopes	14
3.3 Optimal Speed Profile for Steep Uphill Slopes	16
3.4 Combining Up- and Downhill Speed Profiles	16
4 Disturbance of Analytical Speed Profiles	19
4.1 Types of Disturbances	19
4.2 Effects of Disturbed Input Data in Steep Uphill Slopes	20
4.2.1 Position Bias Errors	20
4.2.2 Angle Bias Faults	20
4.3 Effects of Disturbed Input Data in Steep Downhill Slopes	23
4.3.1 Position Bias Errors	23
4.3.2 Angle Bias Errors	23
4.4 Effects of Disturbed Input Data on a Plateau Road Profile	25
4.5 Mass Error	27
4.6 Wheel Radius Errors	27
4.7 Changes to the Aerodynamic Drag Force and Rolling Resistance	28
4.8 Combined Disturbances	31
4.9 Gearshifts Caused by Disturbances	34
4.10 Conclusions	34

5 MPC with Dynamic Programming, DP-tool	36
5.1 Model Predictive control	36
5.2 Dynamic Programming	37
5.3 Complete Control Algorithm	37
5.3.1 Cost Function	38
6 Results DP-tool	40
6.1 Basic Roadprofiles	40
6.2 Disturbance of Basic Road Profiles	43
6.2.1 Position Bias Faults	43
6.2.2 Angle Bias Faults	44
6.2.3 Mass Errors	46
6.2.4 Wheel Radius Error	48
6.2.5 Changes in Aerodynamic Drag and Rolling Resistance	49
6.2.6 Combining Disturbances	52
6.3 Real Road Profiles	56
6.3.1 Position Bias Errors on Real Road Sections	57
6.3.2 Mass and Wheel Radius Errors	57
6.3.3 Roll Resistance and Aerodynamic Drag Force Errors	58
6.4 Map Quantisation	58
6.4.1 Uniform Quantisation	59
6.4.2 Non Uniform Quantisation	62
7 Conclusions	66
8 Extensions and Future Works	68
References	69
A Model Parameters	70
A.1 Simulink Model	70
A.2 Shared Parameters	71
A.3 Cruise Controller	72
A.4 Gearbox	72
A.4.1 Gearbox Model 1	72
A.4.2 Gearbox model in DP-tool	73
B More Results	74
B.1 More Results from DP-tool	74
C DP-Tool GUI	77

Chapter 1

Introduction

The increasing cost of fuel has raised the interest in ways to reduce fuel consumption. In Hellström (2005) and Hellström et al. (2007), it has been shown that by using GPS with a topographic road map and optimizing the velocity over the topography it is possible to lower fuel consumption for a given road without increasing the travel time. For example the fuel injection could be cut before a downhill slope where it is known that the vehicle then will accelerate over its reference speed. In steep uphill slopes it can be advantageous to accelerate before the slope. There are increasing interests in a system like this but there are still no solutions available commercially. A basic requirement is a commercially available topographic road map, today this doesn't exist either. This thesis strives toward outlining requirements for such a map and other input data. It is also in the thesis line to point out effects of disturbed input data.

1.1 Thesis Objectives

The main objective of this thesis is to line out which requirements that should be set on input data to a look-ahead system. How bad input data can be accepted before performance of the system is too degraded? How good does a topographic road map for a look ahead system have to be? Which are the effects on performance from speed trajectories calculated with disturbed input data?

1.2 Method

Two simulation environments are used in this thesis. The first one is a basic model using precalculated speed trajectories to control the vehicle over simple road profiles like the one presented in Figure 1.1. These trajectories are calculated from the results presented in Fröberg et al. (2006) and Fröberg and Nielsen (2007). The second model uses dynamic programming to optimize vehicle speed over given road profiles making it possible to handle real world roads. This simulation environment, named DP-tool, is made by Erik Hellström and described in Hellström (2005) and Hellström et al. (2007). For this thesis some modifications have been done to DP-tool since it is of interest to optimize over one road profile and then run the vehicle simulation over another road profile.

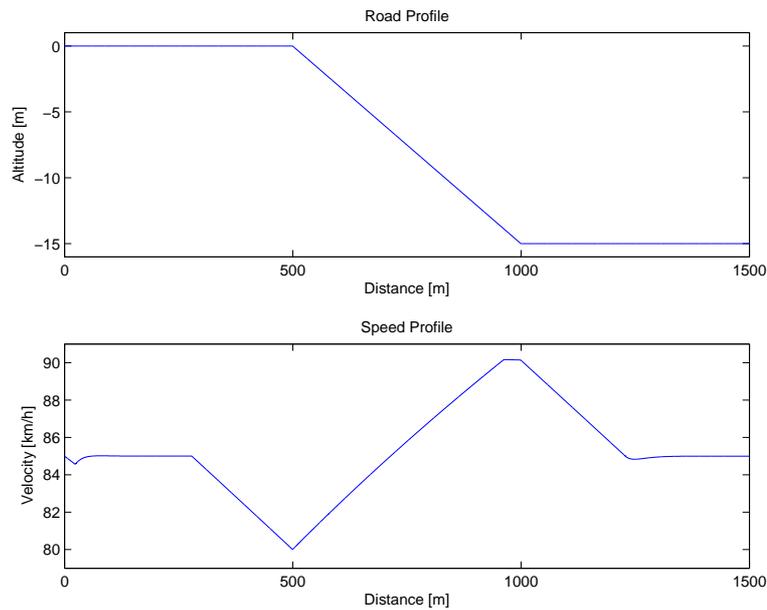


Figure 1.1: A simple downhill road profile with the optimal speed trajectory.

Chapter 2

Vehicle Model

The modeling of the driveline is thoroughly covered in Kiencke and Nielsen (2005) and Nielsen and Eriksson (2005). Since a lot of the work in this thesis is based on Fröberg et al. (2006) and Hellström (2005) the model used is similar to those used in these two papers. The engine torque is modeled as an affine torque map dependent on fuel and engine speed. Further the driveline is assumed stiff. Finally the different forces acting on the wheel of the truck are modeled. Model constants are presented in Appendix B. The model described in this chapter is used both for the simulation and prediction environment utilized in Chapter 3 and 4. It is also used in its complete form in the DP-tool prediction environment utilized in Chapter 6. However the simulation environment in DP-tool uses the measured fuel map to model engine torque instead of the approximation described in Section 2.1.

2.1 Engine

The engine chosen for this thesis work is a 12 liter Scania DT1211 L02 diesel combustion engine. This engine is a euro IV emission class engine and outputs 420 horsepower and has a maximum output torque of 2100 Nm. For use in this thesis a linear model of the engine is constructed. The model for the generated torque minus the internal friction is constructed from a torque map made up of steady state measurements of the generated torque for a given engine speed and amount of injected fuel. The engine map is expressed as $\hat{T}_{map}(N, \delta)$. From this map two functions are approximated one for the maximum output torque, $\hat{T}_{max}(N, \delta)$. The second approximation is a function for the engine drag torque, $\hat{T}_{drag}(N)$ which is the brake torque received from the engine when fueling is cut off and the driveline is engaged.

The engine map is assumed to be affine and the output torque is therefore approximated with a linear model as following:

$$\hat{T}_{map}(N, \delta) = a_e N + b_e \delta + c_e \quad (2.1)$$

Where N is the engine speed and δ is the amount of injected fuel. This model provides a good approximation to measured data. The engine constants are calculated with the least square method.

A model for the drag torque is constructed from the torque map using only the data where $\delta = 0$, $T_{drag}(N) = T_{map}(N, 0)$ and the approximation is given in Equation 2.2, where N is the engine speed. This approximation plotted against measured data can be seen in Figure 2.1.

$$\hat{T}_{drag}(N) = a_d N + b_d \quad (2.2)$$

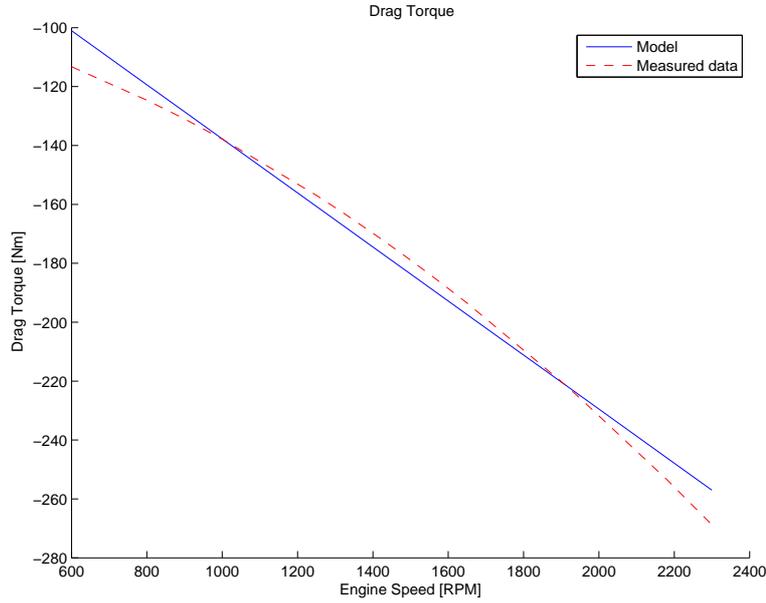


Figure 2.1: Engine drag torque, linear model compared to measured data.

The fueling function is modeled as given in equation 2.3. To control the fueling function a signal P , representing normalized fueling taking a value between zero and one is introduced. The signal P is in other words the driver demand which here is controlled by the cruise controller. Also a signal G , the gear number, an integer between 1 and 12 representing the current gear, is introduced. It is assumed that the engine runs on idle control when the driveline is disengaged, $G = 0$.

$$\delta(N, P, G) = \begin{cases} P\delta_{max}(N) & G \neq 0 \\ \delta_{idle} & G = 0 \end{cases} \quad (2.3)$$

The upper torque bound is approximated by a second order function given in Equation 2.4 and the upper fueling bounds by the function in Equation 2.5.

$$\hat{T}_{max} = a_t N^2 + b_t N + c_t \quad (2.4)$$

$$\hat{\delta}_{max} = a_\delta N^2 + b_\delta N + c_\delta \quad (2.5)$$

The constants of the maximum fueling function are calculated using a least square approximation on measured data for maximum fueling. The approximated fueling function and the measured maximum fueling are plotted in normalized form in Figure 2.2. There are some differences between the real and the approximated max fueling which might have an impact on the result. Since this thesis is about comparing different simulated driving scenarios to compare fuel consumption and travel time this model is adequate.

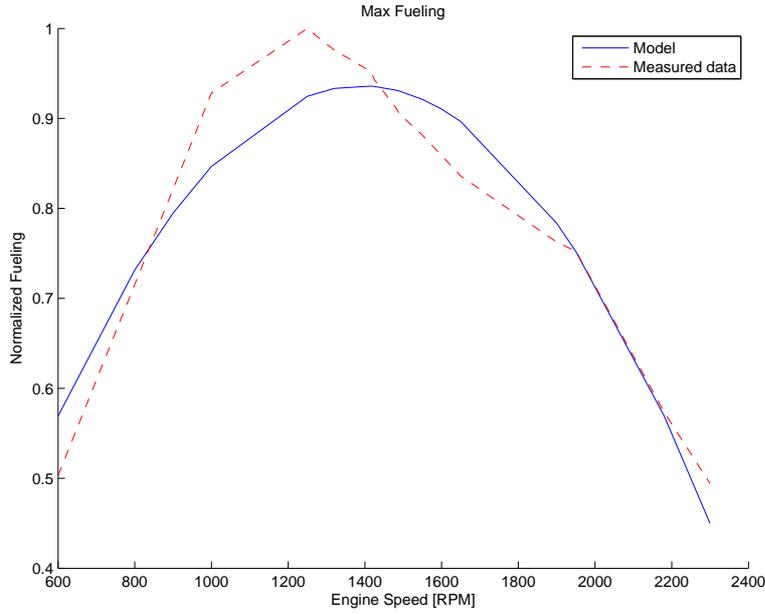


Figure 2.2: The modeled maximum fueling (solid) and measured data (dashed).

To simplify matters even more the upper torque bound is approximated as a function of the maximum fueling bound, see Equation 2.6. In Figure 2.3 this approximation, Equation 2.6, is plotted against the function specified in Equation 2.4. Figure 2.4 contains a plot of the modeled max torque and the measured max torque. The model of the max torque using the max fueling function differs slightly from the least square approximation of the max torque made directly from the engine map.

$$\hat{T}_{max}(N) = \hat{T}_{map}(N, \hat{\delta}_{max}(N)) \quad (2.6)$$

$$\hat{T}_{map}(N, \hat{\delta}_{max}(N)) = a_e N + b_e \hat{\delta}_{max}(N) + c_e \quad (2.7)$$

Finally by combining the Equations 2.1, 2.2, 2.3 and 2.6 an approximation for the output torque, produced torque minus internal friction, can be stated, Equation 2.8.

$$T_e(N, P, G) = \begin{cases} a_e N + b_e P \delta_{max} + c_e & P > 0, G \neq 0 \\ a_d N + b_d & P \leq 0, G \neq 0 \\ 0 & G = 0 \end{cases} \quad (2.8)$$

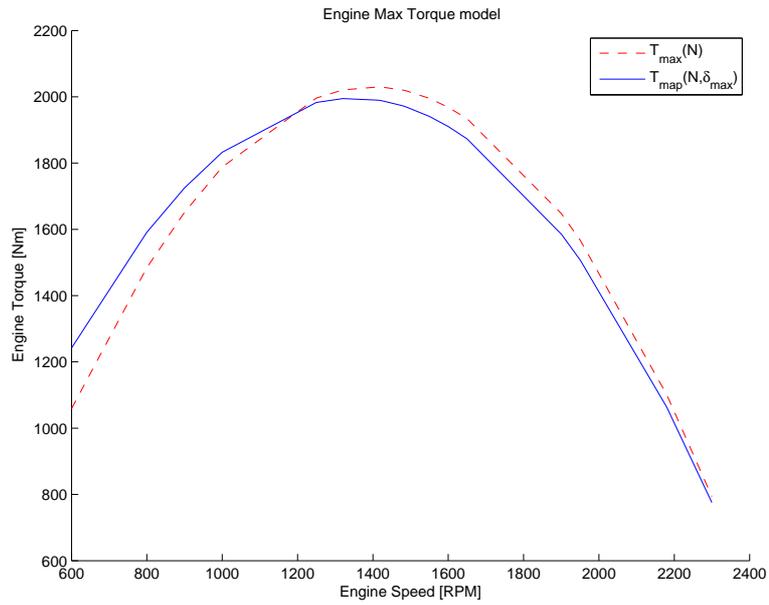


Figure 2.3: The max torque function approximated directly from the engine map (dashed) and by using the maximum fueling function (solid).

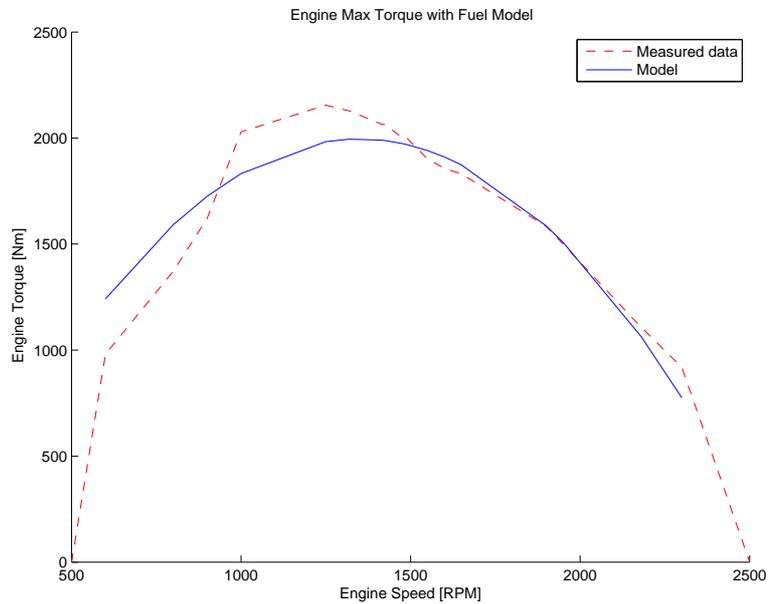


Figure 2.4: The maximum engine torque approximated using the maximum fueling function compared to measured data.

2.2 Driveline

The driveline in this paper is assumed to be stiff and the gearbox and final drive are modeled as tables of ratios and efficiency constants. Previous section, Section 2.1, presents a model of a diesel combustion engine which with its produced output torque drives the vehicle forward through the driveline. In Figure 2.5 the driveline configuration can be viewed.

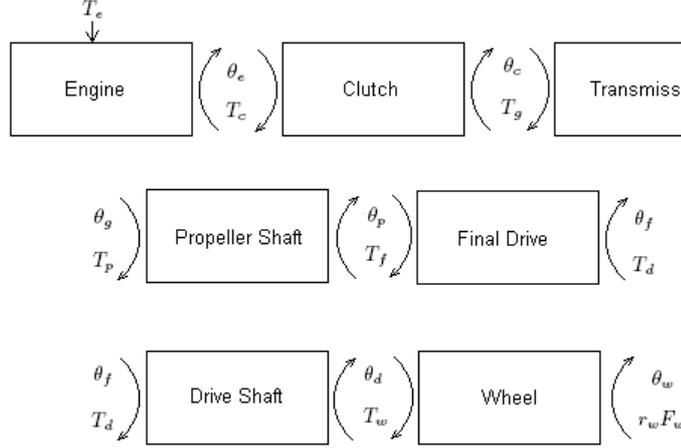


Figure 2.5: The driveline of the vehicle.

The produced torque of the engine minus the internal friction of the engine is represented by T_e , the external load comes from the clutch, T_c . The mass moment of inertia of the engine is J_e and the angle of the flywheel is θ_e which gives:

$$J_e \ddot{\theta}_e = T_e - T_c \quad (2.9)$$

The clutch is assumed to be stiff and therefore:

$$T_c = T_g \quad (2.10)$$

$$\theta_e = \theta_c \quad (2.11)$$

The gearbox is modeled by ratios and efficiency constants for each gear giving, with neglected inertia:

$$T_g i_g \eta_g = T_p \quad (2.12)$$

$$\theta_c = i_g \theta_g \quad (2.13)$$

The propeller shaft is also assumed to be stiff and will therefore not influence the equations, thus:

$$T_p = T_f \quad (2.14)$$

$$\theta_g = \theta_p \quad (2.15)$$

The final drive is, like the gearbox, modeled as a ratio and an efficiency constant with neglected inertia:

$$T_f i_f \eta_f = T_d \quad (2.16)$$

$$\theta_p = i_f \theta_f \quad (2.17)$$

The drive shaft is presumed stiff which gives:

$$T_w = T_d \quad (2.18)$$

$$\theta_f = \theta_d \quad (2.19)$$

Finally the equation of motion for the wheel is:

$$J_w \ddot{\theta}_w = T_w - k_b B - r_w F_w \quad (2.20)$$

$$\theta_d = \theta_w \quad (2.21)$$

Where J_w is the wheel inertia, r_w the wheel radius and k_b the brake constant where B is the brake pedal, taking a value between zero and one. F_w is the friction force at the wheel in other words the force which will drive the vehicle forward.

2.3 Gearbox

The gearbox is as stated implemented as gear ratios i_g and efficiencies η_g when modeled in the powertrain. Gear selection is based upon the current engine speed. Two different engine speed values are stored for each gear, a shift up point and a shift down point. The gearbox is limited to only shift up or down one step at a time. Since only a limited part of the operating range, the top gears, of the gearbox will be utilized in this thesis this limitation should not be a problem. When the gear is shifted the output torque from the gearbox will be zero for one second to model the loss of output torque during a gear shift. Gear shift points are presented in Appendix A.

2.4 Forces

The truck is affected by several different longitudinal forces: aerodynamic drag, rolling resistance, and gravity due to the road slope angle, see Figure 2.6.

The aerodynamic drag force, F_a , is modeled as:

$$F_a = \frac{1}{2} c_w A_a \rho_a v^2 \quad (2.22)$$

Where c_w is the air drag coefficient, A_a the vehicle cross section area, ρ_a the air density and v the velocity of the vehicle.

The rolling resistance force, F_r , is modeled as a simple function dependent only on the road angle, α :

$$F_r = c_r F_N \quad (2.23)$$

$$F_N = mg \cos(\alpha) \quad (2.24)$$

Where α is the road angle, m the mass of the truck and g the gravitational constant and F_N the normal force.



Figure 2.6: The forces acting on the vehicle.

The gravitation force, F_g , is:

$$F_g = mg \sin(\alpha) \quad (2.25)$$

Where α is the road angle, m the mass of the truck and g the gravitational constant.

Finally by using Newtons second law:

$$m\dot{v} = F_w - F_a - F_r - F_g \quad (2.26)$$

Where F_w is the resulting friction force at the wheel which drives the vehicle forward.

2.5 Complete Driveline

The equations given earlier in this chapter can be combined together to state the complete driveline model. The model is implemented in Simulink, a picture of the system is available in Appendix A. Assume that the gear is another than the neutral gear, then the velocity of the truck can be expressed as:

$$v = \dot{\theta}_w r_w = \frac{r_w}{i_g i_f} \dot{\theta}_e \quad (2.27)$$

Using this result and combining Equations 2.8, 2.20 and 2.26 earlier in this chapter the following differential equation is received:

$$\dot{v} = \frac{r_w}{J_w + mr_w^2 + \eta_g i_g^2 \eta_f i_f^2 J_e} \left(\eta_g i_g \eta_f i_f T_e(v, P, G) - k_b B - \frac{1}{2} c_w A_a \rho_a r_w v^2 - mgr_w (c_r \cos \alpha + \sin \alpha) \right) \quad (2.28)$$

Where T_e is the engine torque that is given in Equation 2.8 and the engine speed N can be rewritten as a function of v according to equation 2.29.

$$N = \frac{60i_g i_f}{2\pi r_w} v \quad (2.29)$$

If the neutral gear is used the output torque of the engine is zero which leads to the following differential equation:

$$\dot{v} = \frac{r_w}{J_w + mr_w^2} \left(-k_b B - \frac{1}{2} c_w A_a \rho_a r_w v^2 - mgr_w (c_r \cos \alpha + \sin \alpha) \right) \quad (2.30)$$

For the model used in Chapter 3 and 4 the implementation in Simulink is distance based instead of the standard time based one. Meaning that the Simulink time variable is used as distance instead of time. Therefore Equation 2.28 and 2.30 have to be rewritten into functions of distance instead of time. This is done according to Equation 2.31.

$$\frac{dv}{dt} = \frac{dv}{ds} \frac{ds}{dt} = v \frac{dv}{ds} \quad (2.31)$$

This results in that Equation 2.28 can be rewritten to Equation 2.32 and Equation 2.30 into 2.33.

$$\begin{aligned} \frac{dv}{ds} = \frac{1}{v} \frac{r_w}{J_w + mr_w^2 + \eta_g i_g^2 \eta_f i_f^2 J_e} & \left(\eta_g i_g \eta_f i_f T_e(v, P, G) - k_b B - \right. \\ & \left. - \frac{1}{2} c_w A_a \rho_a r_w v^2 - mgr_w (c_r \cos \alpha + \sin \alpha) \right) \end{aligned} \quad (2.32)$$

$$\frac{dv}{ds} = \frac{1}{v} \frac{r_w}{J_w + mr_w^2} \left(-k_b B - \frac{1}{2} c_w A_a \rho_a r_w v^2 - mgr_w (c_r \cos \alpha + \sin \alpha) \right) \quad (2.33)$$

2.6 Fuel Consumption

The fuel consumption, see Equation 2.36, is calculated by integrating the fuel mass flow given in Equation 2.37. The fuel mass flow itself, Equation 2.34, is determined from the fueling function δ described in Equation 2.3. Where \dot{m}_f is the fuel mass flow [g/s], δ the fueling [mg/stroke], N the engine speed [RPM] and c_f is defined by equation 2.35 where n_{cyl} is the number of cylinders and n_r the number of revolutions of the crankshaft per stroke. It is assumed that the engine runs on idle fueling during gear shifts since an automatic manual transmission ramps engine torque up and down during a gear shift and therefore some fuel will be used.

$$\dot{m}_f = c_f N \delta \quad (2.34)$$

$$c_f = \frac{1}{60 \cdot 10^3} \cdot \frac{n_{cyl}}{n_r} \quad (2.35)$$

$$m_f(N, P, G) = \int \dot{m}_f(N, P, G) dt \quad (2.36)$$

Where

$$\dot{m}_f(N, P, G) = \begin{cases} c_f N P \delta_{max}(N) & G \neq 0 \\ c_f N_{idle} \delta_{idle} & G = 0 \end{cases} \quad (2.37)$$

2.7 Fuel and Time

To be able to compare different simulations over the same road profile a difference in fuel consumption and travel time has to be measured. These quantities are called Δt and Δf . They are defined according to Equation 2.38 and 2.39, where t_{case} and t_{ref} are the travel times of the current simulation and a reference simulation. $m_{f_{case}}$ and $m_{f_{ref}}$ stands for the fuel consumption of the current simulation and a reference simulation. Most times the standard PI cruise controller is chosen as the reference simulation.

$$\Delta t = \frac{t_{case} - t_{ref}}{t_{ref}} \quad (2.38)$$

$$\Delta f = \frac{m_{f_{case}} - m_{f_{ref}}}{m_{f_{ref}}} \quad (2.39)$$

There is one quite difficult question that will be raised when different speed profiles for a given road are evaluated, how can difference in travel time be compared to a difference in fuel consumption. For example if the fuel consumption for a given road decreases with 1 % and the travel time increases with 1 %, is this system better or worse than the original cruise controller? The problem here is that there are no direct answers. The struggle for a look ahead cruise controller is to keep the travel time constant while lowering the fuel consumption. Due to the optimization problem on real road profiles this is hard to achieve therefore some type of weight or penalty function has to be constructed.

2.7.1 One Delta Function

The idea is to construct one single Δ -function from Δ_{time} and Δ_{fuel} or rather convert the difference in time to a difference in fuel. A test on level road with a base speed of $85km/h$ was done, simulation were then made to compare what happens with the fuel consumption if the road was driven faster or slower. The results are collected in the diagram presented in figure 2.7. In this figure, 2.7, also a least square estimation of the measured data, see equation 2.40 is presented. The offset in the least square approximation was too small to have any impact on the results later on in this report and was removed to simplify the approximation slightly. The result for this case is that a -1% decrease travel time can be seen as a 0.903% increase in fuel consumption in other words resulting in $c = 0.903$. It should be noted that a changed speed interval or set speed would lead to a changed c as well.

$$\Delta_{tot} = \Delta f + c \Delta t \quad (2.40)$$

$$\text{Here: } \Delta_{tot} = 0 \iff$$

$$\begin{aligned} c &= -\frac{\Delta f}{\Delta t} \\ c &= 0.903 \end{aligned} \quad (2.41)$$

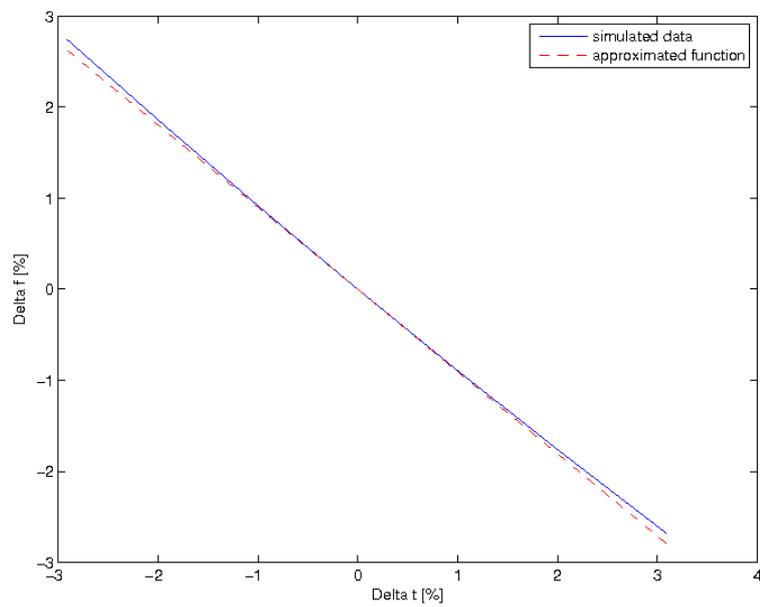


Figure 2.7: Δf as a function of Δt , solid line represents measured data while the dashed line represents the least square approximation.

Chapter 3

Analytically Derived Optimal Speed Profiles

It has previously been shown by Fröberg et al. (2006) that optimal speed profiles for simple road profiles can be derived analytically. The focus of this thesis is to analyze which effects disturbed input data has on a look ahead cruise controller given the knowledge of optimal speed profiles for different roads. This chapter will handle how these optimal speed profiles look like. It should also be stated since traveling at a lower speed saves fuel so time also has to be taken into consideration when calculating the optimal speed profiles.

A look ahead system could allow for more than just the velocity to be optimized over a given road. For example if it is known that the vehicle is almost at the top of a hill a gearshift could be avoided by allowing the vehicle to drag it self over the top at the current gear. When a gearshift really is necessary it could instead be done before an up- or downhill slope. For example the vehicle could shift the gear down one step and enter a steep uphill slope at a slightly higher engine speed. This could allow the vehicle to finish the slope without being forced to shift down two steps while in the slope.

The road profiles accounted for in this chapter are of theoretical interest since it is possible to find analytical solutions to the optimal speed profiles, this comes with the drawback that they do not reassemble real road profiles. Effects of real world road profiles will be explored later in this thesis. The reference speed is set to 85km/h , minimum speed to 80km/h and maximum velocity to 90km/h . The vehicle mass is set to $40\,000\text{kg}$ and the gear number is 12 if nothing else is stated. In this chapter it is assumed that the control system has perfect knowledge of the road topography, vehicle parameters and environmental conditions. Optimizations are done to keep an average speed of 85km/h over the road sections.

3.1 Optimal Speed Profiles on Level Road and Small Gradients

First speed profiles on level roads will be handled. By intuition one can argue that with a given travel time it is optimal to keep a constant velocity. Unnecessary accelerations should by intuition be seen as non fuel optimal behaviour. Also by traveling at a higher speed the aerodynamic drag force, which has a v^2 part in it, will quickly increase in size. Fröberg et al. (2006) also shows that it is optimal to keep the velocity constant on level road when the travel time is given.

These arguments also applies to small gradients, which are defined as downhill slopes where the gravity is not enough to accelerate the vehicle and uphill slopes where the engine is capable to keep the vehicle at reference speed. There is no reason to accelerate the vehicle in a small downhill slope if the gravity is not enough to speed up the truck, the same applies for small uphill gradients if the engine can keep the velocity constant then that is the optimal solution for a given travel time. This is also shown in Fröberg et al. (2006).

It should be noted that the solution explained here is true for an affine engine map. There is no guarantee that a non affine engine map will give this result in the optimal case. A non affine engine map makes it a lot more difficult to analytically derive the optimal trajectory and solutions with such engine maps are not considered in this thesis.

3.2 Optimal Speed Profile for Steep Downhill Slopes

It has been shown in for example Hellström (2005) and Fröberg et al. (2006) that controlling the speed of a vehicle in steep gradients give the potential to save some per cent of fuel. The idea is in a downhill slope to cut the fuel injection before the slope and then make use of the gravity to accelerate the vehicle. For a typical speed profile see Figure 3.1. This will if compared to a standard cruise controller take some extra time since the vehicle will travel at a lower average speed. To make up for this loss of time the vehicle will be accelerated before uphill slopes which will be seen later in this chapter. The 500m, 3% slope is steep enough to accelerate the vehicle but still short enough to point out most effects of disturbances which in longer slopes wont be possible to notice. It should be noticed that if the vehicle would be allowed to reach a higher speed than 90km/h in downhill slopes a lot of the gain of using this optimization would be lost. This is because energy is lost when the vehicle is forced to brake when 90km/h is reached. However the top velocity should realistically be restricted due to safety reasons and legislation. The cruise controller parameters are the same for both the optimization and the standard cruise controller.

In Figure 3.2 a standard PI cruise controller is compared to the optimal speed profile for the given downhill slope. The result is an increased travel time by $\Delta t = +1.02\%$ and $\Delta f = -12.65\%$ less consumed fuel.

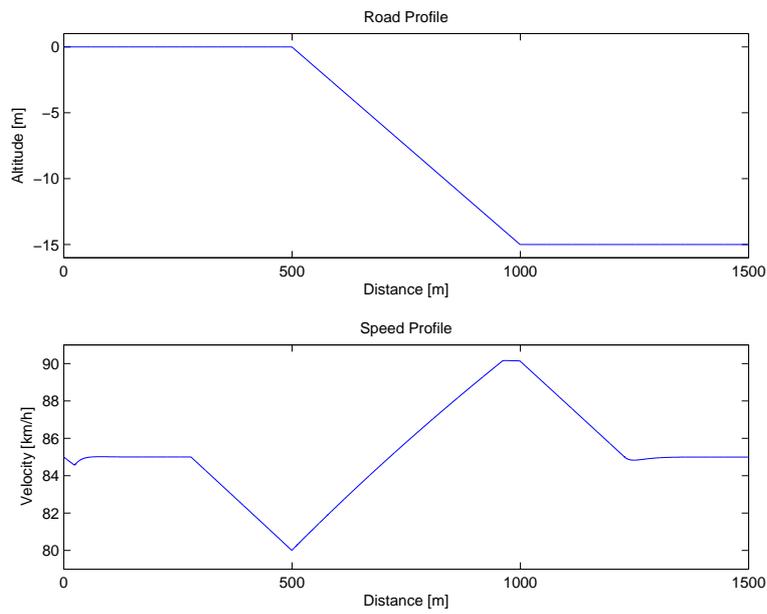


Figure 3.1: The optimal speed profile on the given downhill slope.

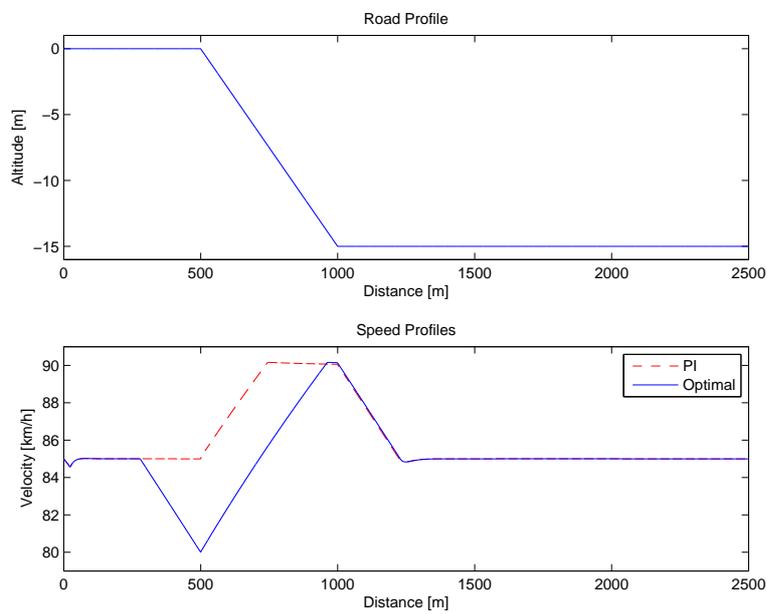


Figure 3.2: The optimal speed profile (solid) and the PI regulated speed profile (dashed) on the given road profile. $\Delta t = +1.02\%$ and $\Delta f = -12.65\%$.

3.3 Optimal Speed Profile for Steep Uphill Slopes

Steep uphill slopes are defined as slopes in which the engine isn't strong enough to keep the vehicle at reference velocity. The optimal solution is to accelerate the vehicle before the uphill slope, see Figure 3.3 for a road and speed profile. This allows the vehicle to enter the slope at a higher speed and therefore also finish the slope at a higher speed, something which will shorten the travel time but of course cost more fuel. This is in other words not fuel optimal but makes it possible to regain some of the time lost because of the decreased vehicle speed before steep downhill slopes. So when the uphill slope is combined with the profile of a downhill slope the travel time will be more or less constant but still a few percent fuel can be saved. The complete idea of the system is to save fuel by cutting the injection before steep downhill slopes and regain the lost travel time by accelerating the vehicle before steep uphill slopes.

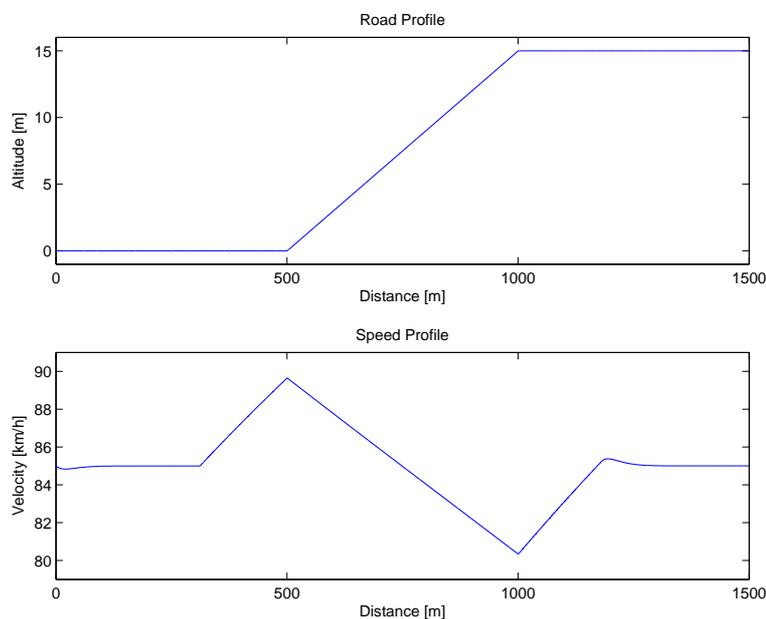


Figure 3.3: The optimal speed profile on the given uphill slope.

In Figure 3.4 a PI regulated cruise controller is compared to the optimal speed profile. In this case the distance will be driven in $\Delta t = -1.92\%$ less time but with the cost of $\Delta f = +1.18\%$ extra fuel.

3.4 Combining Up- and Downhill Speed Profiles

If the two road profiles explained in section 3.2 and 3.3 are combined and also the optimal velocities a case like the one in Figure 3.5 is constructed. For this road profile it is not only possible to save fuel but also to drive the distance faster than what the standard PI cruise controller is capable of.

Figure 3.6 shows a comparison between the PI regulated cruise controller and the optimal cruise controller. In this case both the travel time and the fuel consumption are less for the optimal controller, $\Delta t = -0.90\%$ and $\Delta f = -5.75\%$.

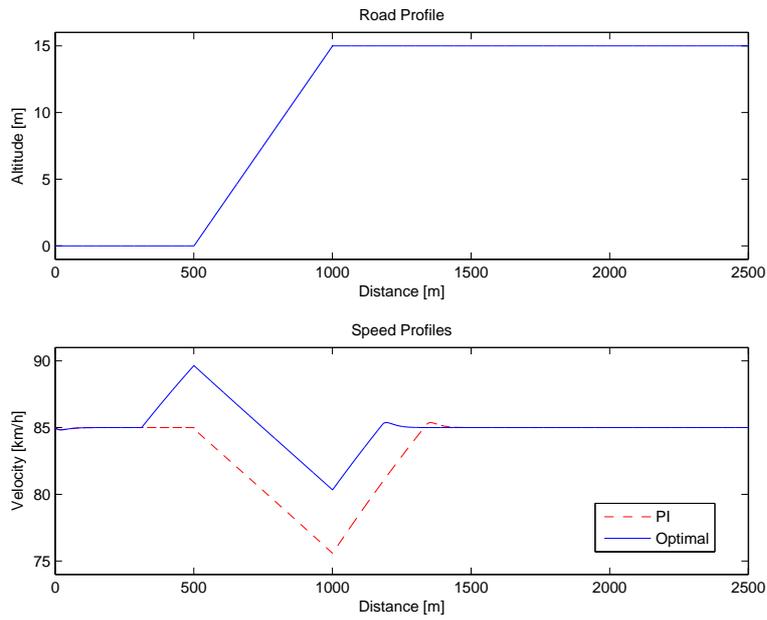


Figure 3.4: The optimal speed profile (solid) and the PI regulated speed profile (dashed) on the given road profile. $\Delta t = -1.92\%$ and $\Delta f = +1.18\%$

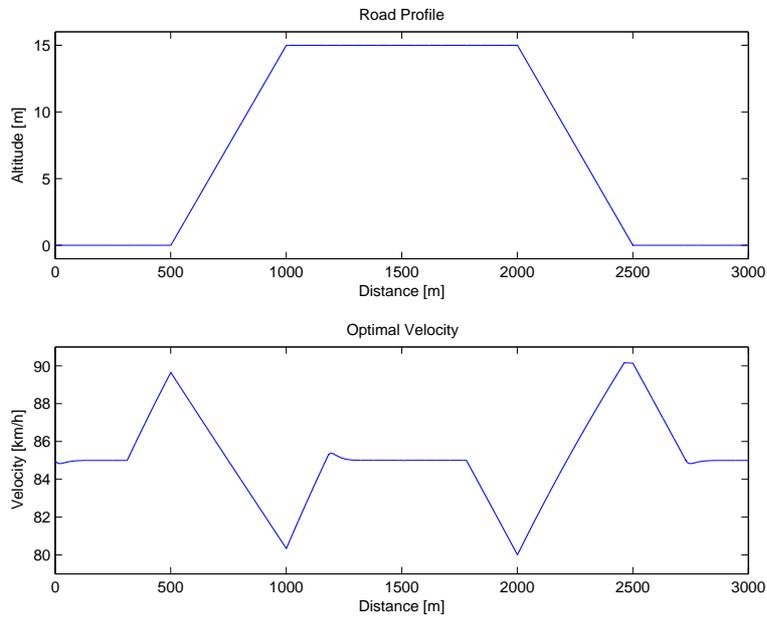


Figure 3.5: The optimal speed profile on the given plateau road profile.

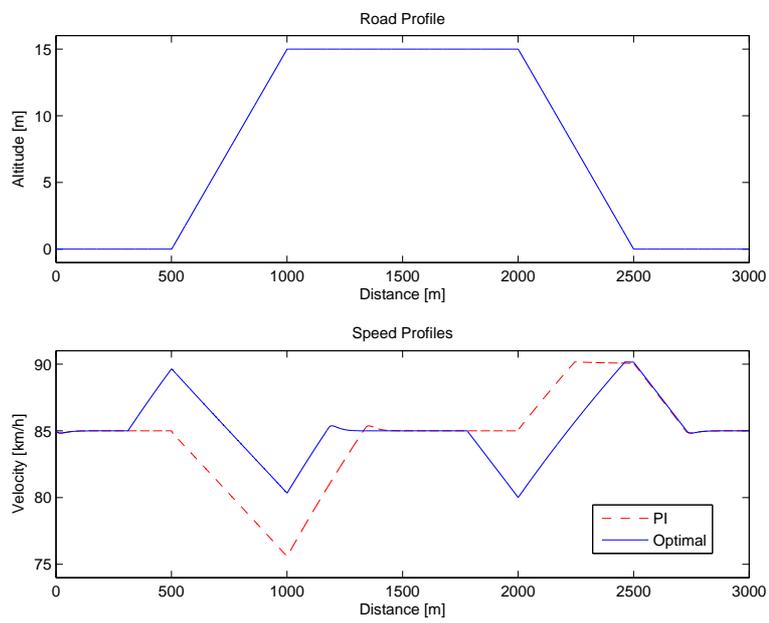


Figure 3.6: The optimal speed profile (solid) and the PI regulated speed profile (dashed) on the given road profile. $\Delta t = -0.90\%$ and $\Delta f = -5.75\%$.

Chapter 4

Disturbance of Analytical Speed Profiles

In Chapter 3 the analytically calculated speed profiles for steep up- and downhill slopes were presented. Of great interest is the capability of the system to handle disturbed input data. How large differences can be tolerated on the input data compared to the real road? There are several reasons why the system should be able to handle quite big disturbances, higher precision on map data will be more expensive and require more storage space and some parameters in the system, like vehicle mass are not possible to estimate perfectly. Due to the limitations of road profiles for which it are possible to analytically calculate the speed profiles there will also be limitations to which types of disturbances it are possible to simulate for this road profiles.

Therefore in this chapter only bias faults of different types will be given attention. To handle disturbances of types like white noise or quantization of map data some type of numerical optimization will be necessary. Another notation that should be made is that all faults are implemented in the prediction since this leaves the simulation conditions untouched for fair comparisons. It should also be noted that unless anything else is mentioned all simulations are done with the highest gear, number 12. The reference speed is set to 85km/h , minimum speed to 80km/h , maximum speed to 90km/h and the vehicle mass to $40\,000\text{ kg}$. For the simulated scenarios the difference in travel time and fuel consumption are presented and also the combined delta function defined in Section 2.7. Both the travel time and the fuel consumption are compared to a standard PI cruise controller.

4.1 Types of Disturbances

This chapter will handle several types of disturbances, primary bias faults in position and in the angle data of steep up- and downhill profiles. Effects of mass and wheel radius errors in the optimization and some environmental disturbances like badly estimated rolling resistance and aerodynamic drag will also be handled for these simple road profiles.

4.2 Effects of Disturbed Input Data in Steep Uphill Slopes

4.2.1 Position Bias Errors

In Figure 4.1 a plot of the results of a positive or negative bias error in the position data can be seen. A $50m$ bias is a quite large error and is approximately twice the length of a long haulage truck. The difference in travel time and fuel consumption compared to the standard PI cruise controller can be seen in Table 4.1. Figure 4.1 clearly shows the effect, if the system believes that the uphill slope comes early, the case marked as $-50m$, it will start to accelerate too early and reach maximum speed before the slope begins. The effect is higher fuel consumption but also slightly shorter travel time.

There is also another problem which is not directly visible. If the vehicle has already reached its maximum speed it will stop accelerate and reduce throttle level and therefore lose turbo pressure, something which should be avoided just before steep uphill slopes. The lost turbo pressure is not modeled in this thesis work but is very likely to lead to a loss in travel time and possibly also increased fuel consumption. In the opposite situation when the system believes that the hill starts further away, the scenario marked as $+50m$, than it actually does the vehicle will start to accelerate too late and does therefore not reach the optimum speed before the slope and therefore also receives a lower average speed on the distance.

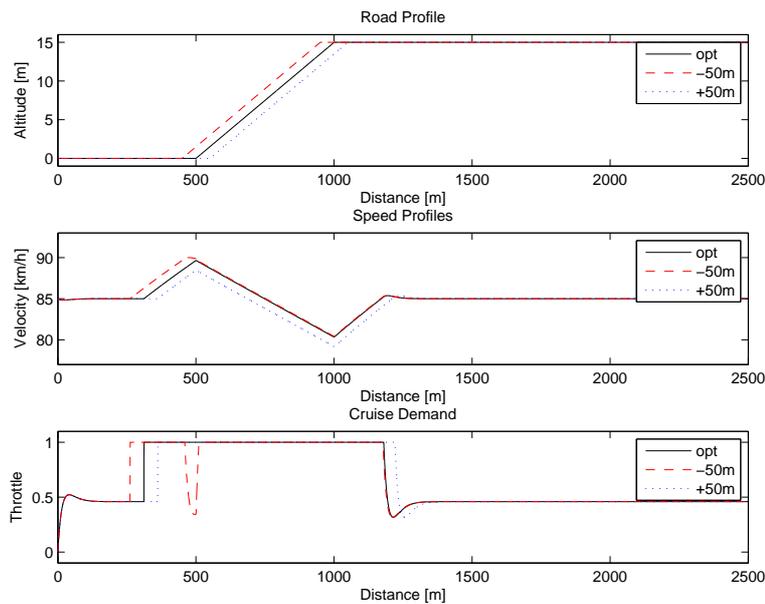


Figure 4.1: This figure shows simulations with position bias errors compared to the optimal scenario. The disturbed system will, as can be seen, either react too early or too late.

4.2.2 Angle Bias Faults

An angular bias fault will result in a similar result as the position bias fault. If the system receives data that says that the hill is less steep, here 2.4% instead

Uphill 3% 500m

Δ -Function	Value	Δ -Function	Value
Optimal Scenario		Angle Bias -20%	
Δf_{opt}	+1.17%	Δf_{-20}	+0.61%
Δt_{opt}	-1.92%	Δt_{-20}	-1.03%
$\Delta f_{opt} + c\Delta t_{opt}$	-0.56%	$\Delta f_{-20} + c\Delta t_{-20}$	-0.32%
Position Bias -50m		Angle Bias +20%	
Δf_{-50}	+1.27%	Δf_{+20}	+1.27%
Δt_{-50}	-2.05%	Δt_{+20}	-2.05%
$\Delta f_{-50} + c\Delta t_{-50}$	-0.58%	$\Delta f_{+20} + c\Delta t_{+20}$	-0.58%
Position Bias +50m			
Δf_{+50}	+0.87%		
Δt_{+50}	-1.45%		
$\Delta f_{+50} + c\Delta t_{+50}$	-0.44%		

Table 4.1: Results from a 500m 3% uphill slope with bias faults in the position data or in the angle data, a standard PI cruise controller is used as reference.

of 3%, it will accelerate to reach an expected optimal top speed which is lower than the optimal top speed for the real slope. If the situation is the opposite the control system will accelerate the vehicle to a higher speed before the slope than what is optimal. In this case optimization is done for a 3.6% slope but the vehicle is run over a 3% slope. These behaviours are shown in Figure 4.2 and the result in time and fuel consumption compared to a standard cruise controller are presented in Table 4.1. The reason for the small difference between the optimal solution and the disturbed solution with an expected 3.6% hill is because of the hard maximum speed limit of 90km/h for the vehicle. Thus since the vehicle never accelerates to a velocity over 90km/h a steeper uphill slope will not affect the result by much compared to the 3% 500m uphill slope where the top speed is almost reached before the slope.

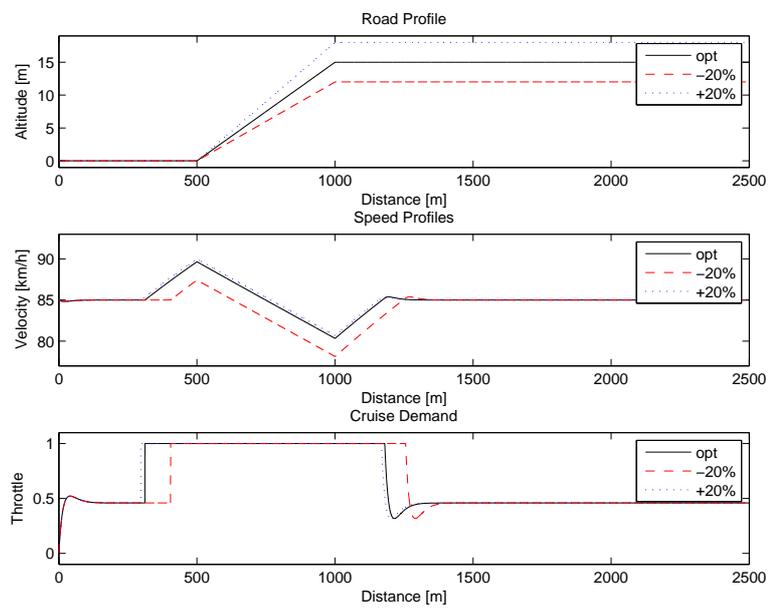


Figure 4.2: Simulations with a bias error in the angle forcing the system to optimize for a too steep or too flat slope. The system reacts too late when velocity is optimized for a assumed flatter slope but the difference is minor when the slope is assumed steeper due to the hard speed limit at 90km/h.

4.3 Effects of Disturbed Input Data in Steep Downhill Slopes

4.3.1 Position Bias Errors

The effects of a bias error in the position data in steep downhill slopes are similar to those in steep uphill slopes. If the system believes that the slope starts later than it really does it will cut the fuel injection too late and therefore waste fuel that could otherwise be saved, see Figure 4.3, the plot marked as $+50m$. In the opposite situation fuel injection will be cut too early which will cause the vehicle to reach minimum speed too early resulting in a slightly longer travel time but also a slightly lower fuel consumption, see Figure 4.3 and the scenario marked as $-50m$.

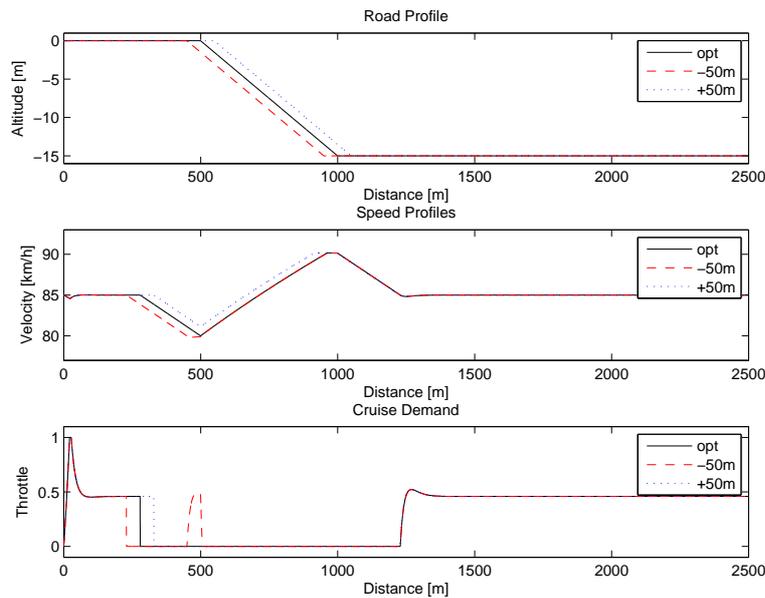


Figure 4.3: Simulations of a downhill slope with position bias faults in the optimization compared to the optimal speed trajectory. Again it is obvious that with a disturbance like these the system reacts either too early or too late which results in lost time or wasted fuel.

4.3.2 Angle Bias Errors

Scenarios with positive and negative angle bias faults are plotted in Figure 4.4 and the differences in fuel and travel time are presented in Table 4.2. As can be seen a positive bias fault of 0.6 percentage units, marked as $+20\%$ in Figure 4.4, in the angle of the slope will not affect the outcome since a 500 m 3% slope is enough to accelerate the vehicle to maximum speed, $90km/h$, from minimum speed, $80km/h$. A negative bias fault of -0.6 percentage units, marked as -20% in Figure 4.4, on the other hand will affect the behaviour of the system and some potential saved fuel will be wasted since fuel injection is cut too late before the slope.

Downhill 3% 500m

Δ -Function	Value	Δ -Function	Value
Optimal Scenario		Angle Bias -20%	
Δf_{opt}	-12.65%	Δf_{-20}	-8.33%
Δt_{opt}	+1.02%	Δt_{-20}	+0.56%
$\Delta f_{opt} + c\Delta t_{opt}$	-11.73%	$\Delta f_{-20} + c\Delta t_{-20}$	-7.82%
Position Bias -50m		Angle Bias +20%	
Δf_{-50}	-12.85%	Δf_{+20}	-12.65%
Δt_{-50}	+1.16%	Δt_{+20}	+1.02%
$\Delta f_{-50} + c\Delta t_{-50}$	-11.80%	$\Delta f_{+20} + c\Delta t_{+20}$	-11.73%
Position Bias +50m			
Δf_{+50}	-9.81%		
Δt_{+50}	+0.07%		
$\Delta f_{+50} + c\Delta t_{+50}$	-9.75%		

Table 4.2: Results from a 500m 3% downhill slope with bias faults in the position data or in the angle data. The standard PI cruise controller is used as reference.

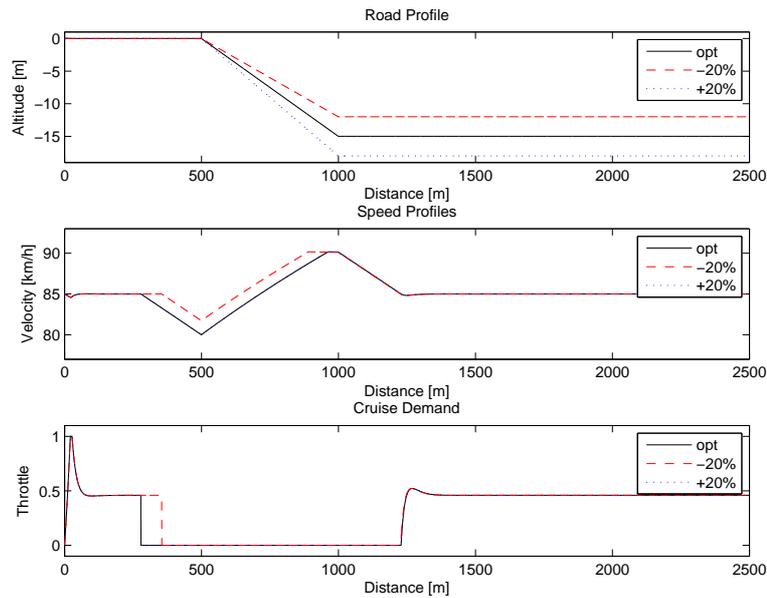


Figure 4.4: Plots of simulations with angle bias errors compared to the optimal scenario. With a assumed flatter slope the system reacts to late and a lot of the fuel that could be saved is wasted. For the assumed steeper slope there is no difference since the 3% downhill slope is enough to accelerate the vehicle from 80 to 90km/h.

4.4 Effects of Disturbed Input Data on a Plateau Road Profile

When a steep uphill slope is combined with a steep downhill slope the effects on the speed profiles are the same as in the single cases for these two slopes, see Figure 4.5. More interesting is the fact that when the control system acts too early the fuel consumption and travel time are almost unaffected but in the opposite situation the losses in both travel time and the amount of fuel used are facts, see Table 4.3. When the system reacts too early the minor difference can be explained by the fact that the vehicle will travel at top speed for some extra time before the uphill slope and at minimum speed for a longer time before the downhill slope. These two errors will cancel each other and therefore the small difference. In the case where the control system does not cut the fuel injection in time some of the potential fuel that could be saved before the downhill slope is wasted and therefore the degraded performance.

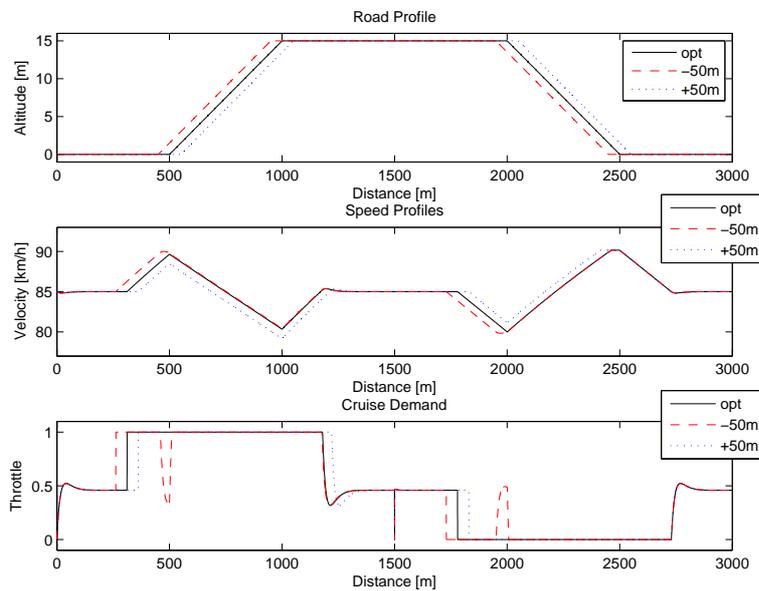


Figure 4.5: Simulations done over a 3% plateau with position bias errors with the optimal scenario as reference.

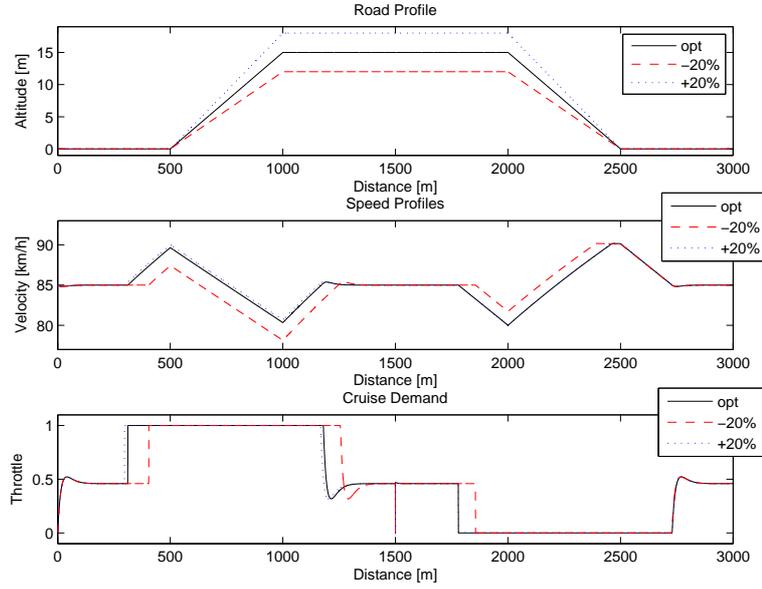


Figure 4.6: Simulations done over a 3% plateau with angle bias faults compared to the optimal speed trajectory.

Plateau 3% 500m

Δ -Function	Value	Δ -Function	Value
Optimal Scenario		Angle Bias -20%	
Δf_{opt}	-5.75%	Δf_{-20}	-3.96%
Δt_{opt}	-0.78%	Δt_{-20}	-0.42%
$\Delta f_{opt} + c\Delta t_{opt}$	-6.45%	$\Delta f_{-20} + c\Delta t_{-20}$	-4.34%
Position Bias -50m		Angle Bias +20%	
Δf_{-50}	-5.75%	Δf_{+20}	-5.65%
Δt_{-50}	-0.79%	Δt_{+20}	-0.90%
$\Delta f_{-50} + c\Delta t_{-50}$	-6.46%	$\Delta f_{+20} + c\Delta t_{+20}$	-6.46%
Position Bias +50m			
Δf_{+50}	-4.50%		
Δt_{+50}	-0.65%		
$\Delta f_{+50} + c\Delta t_{+50}$	-4.56%		

Table 4.3: Results from simulations on an 500m 3% uphill slope combined with a 500m 3% downhill slope, the standard PI cruise controller is used as reference.

4.5 Mass Error

To accurately calculate how to control the velocity in up and downhill slopes the mass of the vehicle has to be known. A mass estimation is available to the trucks control systems. This function will estimate the mass of the vehicle with an error of approximately $\pm 10\%$. A disturbance in the mass estimation of the vehicle could be seen as a disturbance in the road slope angle since gravitational force can be approximated as in Equation 4.1. If both errors are large this may result in large prediction errors for the look ahead cruise controller. A mass error however will affect other parts of the speed profile calculation as well and will therefore have a larger impact than a pure angle fault of the same size, see Equation 2.32. Figure 4.7 shows a plot of the speed profile when calculated with a $+10\%$ and with a -10% error in the vehicle mass, which initially was $40\,000\text{kg}$. The impacts on fuel consumption and travel time are presented in Table 4.4.

$$F_g = mg \sin \alpha \approx mg \alpha \quad \text{For small angles } \alpha \quad (4.1)$$

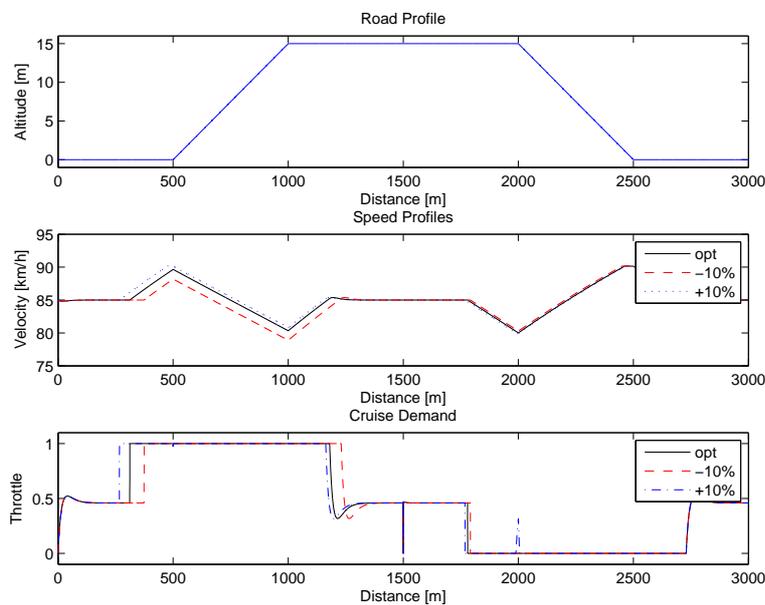


Figure 4.7: Plots of the speed trajectories when optimizations are done with a mass fault of $\pm 10\%$.

4.6 Wheel Radius Errors

The effects of wheel radius errors might at first seem hard to predict since it affects several different parts of the complete vehicle model. The force at the wheel from the engine torque will be wrongly predicted, see Equation 2.8 and 2.29, but also the aerodynamic drag force and the roll resistance force will be affected, see Equation 2.22 and 2.29. Still more parts of the prediction will be affected as can be seen in the complete vehicle model given in Equation 2.28. The effects of a wheel radius fault do not get as big as could be expected initially with all affected parts of the driveline equations. Wheel radius errors

Plateau 3% 500m			
Δ -Function	Value	Δ -Function	Value
Optimal Scenario $r = 52cm$		Radius $-5cm, r = 47cm$	
Δf_{opt}	-5.75%	Δf_{r47}	-5.86%
Δt_{opt}	-0.78%	Δt_{r47}	-0.42%
$\Delta f_{opt} + c\Delta t_{opt}$	-6.45%	$\Delta f_{r47} + c\Delta t_{r47}$	-6.24%
Mass Error -10%		Radius $+5cm, r = 57cm$	
$\Delta f_{-10\%}$	-5.77%	Δf_{r57}	-4.78%
$\Delta t_{-10\%}$	-0.36%	Δt_{r57}	-0.97%
$\Delta f_{-10\%} + c\Delta t_{-10\%}$	-6.47%	$\Delta f_{r57} + c\Delta t_{r57}$	-5.66%
Mass Error $+10\%$			
$\Delta f_{+10\%}$	-4.92%		
$\Delta t_{+10\%}$	-0.95%		
$\Delta f_{+10\%} + c\Delta t_{+10\%}$	-5.78%		

Table 4.4: Results from simulations with wheel radius errors (right column) and mass errors (left column) on an 500m 3% uphill slope combined with a 500m 3% downhill slope. The standard PI cruise controller is used as reference.

will only have an larger effect in steep uphill slopes where the optimal speed profile will be falsely predicted. Steep downhill slopes are almost unaffected.

The reason for this can be explained by looking at the equations referred to earlier in this section. A positive fault in the wheel radius will predict a lower engine torque as long as the engine speed is kept under approximately 1 500 rpm due to the v^2 dependent maximum torque function, see Section 2.1. The aerodynamic drag and roll resistance will on the other hand be estimated higher than what they really are. This results in a predicted slower acceleration in uphill slopes but only with minor effects in steep downhill slopes due to an estimation of decreased engine drag torque. In the opposite situation with a negative fault in the wheel radius the control system will predict an increased engine torque and decreased roll resistance and aerodynamic drag. This causes the control system to accelerate the vehicle too late in steep uphill slopes. Again steep downhill slopes remain relatively unaffected due to an increased engine drag torque which makes up for the lower rolling resistance and aerodynamic drag. In Figure 4.8 these two cases can be seen compared to the optimal case, the wheel radius is 52cm in the vehicle model and optimizations are done with a wheel radius of 47cm, the optimal 52cm and 57cm. Results from the simulations are presented in Table 4.4.

4.7 Changes to the Aerodynamic Drag Force and Rolling Resistance

The last model errors presented directly in this thesis are changes to the aerodynamic drag force and the rolling resistance in the optimization. It could be argued that good models are needed for these two forces since both directly affect the calculation of when to accelerate and decelerate to reach the optimal speed. The aerodynamic drag force is heavily dependent on weather conditions or more exactly wind speed and direction compared to the truck. The same can be said about the rolling resistance which is dependent on road condition, road material and also the tyres and the condition of those. Errors in both these resistances will show similar effects on the model since both are

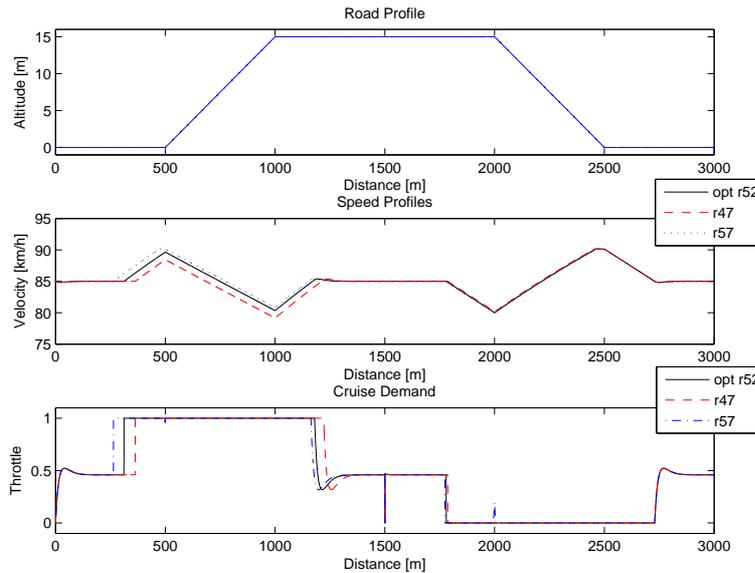


Figure 4.8: Simulations where optimization are done with wheel radius faults of $\pm 5\text{cm}$ resulting in a wheel radius of 47cm or 57cm instead of the vehicles real 52cm .

modeled as negative forces, or rather torques, in the complete driveline model, see Equation 2.28. Due to that the roll resistance force model is only dependent on weight and road angle and the v^2 dependence in the aerodynamic drag force an error in the rolling resistance will have larger impact in low velocities while the aerodynamic drag force fault will have larger effects the faster the truck is traveling. It is interesting to note that even though at first glance it looks bad with large errors in these two negative forces on the vehicle the impact is less serious than expected. A 10% error in roll resistance does not affect the calculation of the optimal speed profile very much, the effect is approximately the same as of a 20m bias fault in position in an uphill slope and that of a 10m bias fault in a downhill slope, see Figure 4.9. With a 10% error in the aerodynamic drag force the effects are even less, still steep uphill slopes suffer more from this type of error, approximately the same as a 15m bias fault while steep downhill slopes remain relatively unaffected, see Figure 4.10.

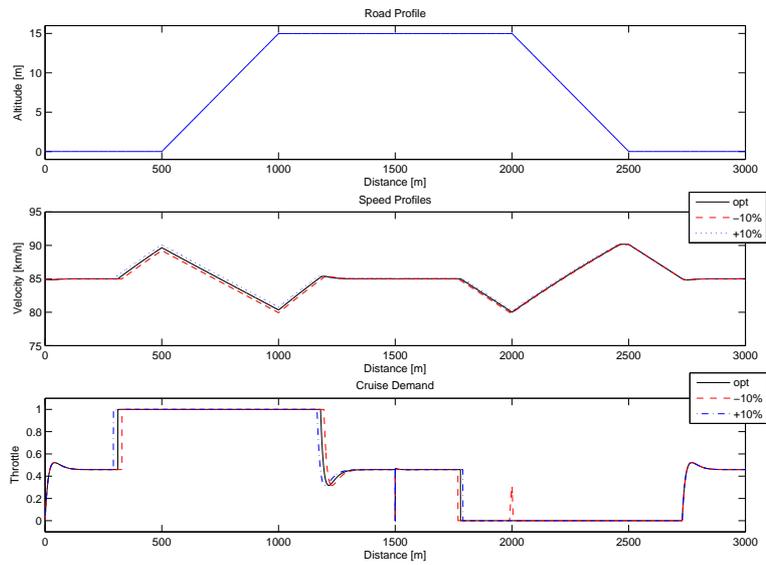


Figure 4.9: Simulations done over a 500m, 3% plateau with model faults in the rolling resistance.

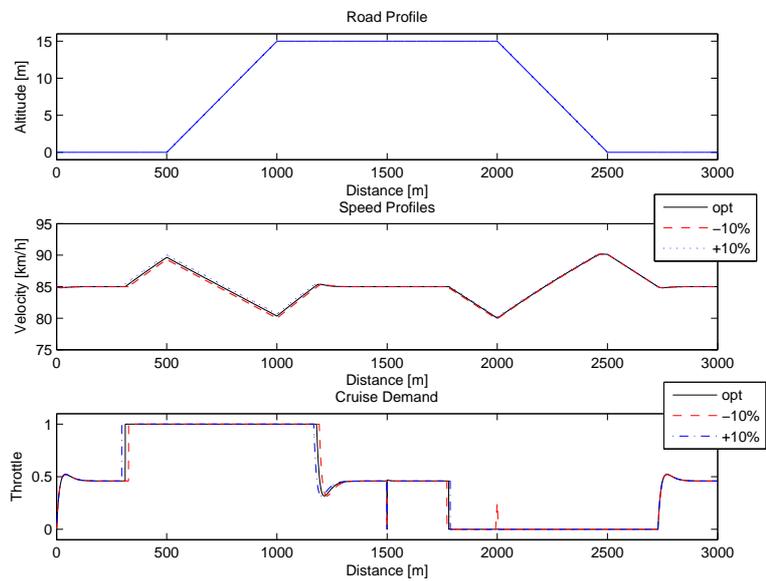


Figure 4.10: Simulations done over a 500m, 3% plateau with model faults in the aerodynamic drag force.

Plateau 3% 500m			
Δ -Function	Value	Δ -Function	Value
Optimal Scenario		Aerodynamic drag -10%	
Δf_{opt}	-5.75%	$\Delta f_{-10\%}$	-5.86%
Δt_{opt}	-0.78%	$\Delta t_{-10\%}$	-0.42%
$\Delta f_{opt} + c\Delta t_{opt}$	-6.45%	$\Delta f_{-10\%} + c\Delta t_{-10\%}$	-6.24%
Rolling resistance -10%		Aerodynamic drag +10%	
$\Delta f_{-10\%}$	-6.02%	$\Delta f_{+10\%}$	-4.78%
$\Delta t_{-10\%}$	-0.60%	$\Delta t_{+10\%}$	-0.97%
$\Delta f_{-10\%} + c\Delta t_{-10\%}$	-6.56%	$\Delta f_{+10\%} + c\Delta t_{+10\%}$	-5.66%
Rolling resistance +10%			
$\Delta f_{+10\%}$	-5.31%		
$\Delta t_{+10\%}$	-0.98%		
$\Delta f_{+10\%} + c\Delta t_{+10\%}$	-6.19%		

Table 4.5: Results from simulations with falsely estimated rolling resistance (left column) and falsely estimated aerodynamic drag (right column) in a 500m 3% uphill slope combined with a 500m 3% downhill slope. The reference is the standard PI cruise controller.

4.8 Combined Disturbances

Of course in real world applications the system will be under the influence of many different disturbances at the same time. Therefore it is of interest to add several types of disturbances together. When earlier results in this chapter are taken into account it is easily realized that some faults will add together in a way which will result in even larger loss of travel time or fuel compared to scenarios with perfect input data. In other cases the effects will more or less cancel each other.

If a downhill slope is considered, from fuel consumption perspective the worst possible scenario is when a large positive position bias fault, in other words the slope is coming earlier than expected, is combined with a large negative angular bias error meaning that the hill is steeper in reality than in the recorded data. In this case the look ahead cruise controller will cut the injection way too late compared to optimum wasting a lot of the potential saved fuel. The same scenario in an uphill slope will instead cause longer travel time since the vehicle will not accelerate early enough to reach optimal speed before the uphill slope. In Figure 4.11 the result of a position bias fault of 50 m and an angle error of -20 % in a 3% downhill slope can be seen. The result is a loss of more than 50% of the potential fuel that could be saved, see Table 4.6.

The problem is that it does not end there. If an error of -10% is added to the mass approximation the result will be even worse, see Figure 4.12 and 4.13. Still there are more possible faults that can be added to the system which will degrade performance even more. In Figure 4.13 the same faults are implemented in a simulation over a road profile with a 500m, 3% uphill slope combined with a 500m, 3% downhill slope. The results from these simulations are presented in Table 4.6.

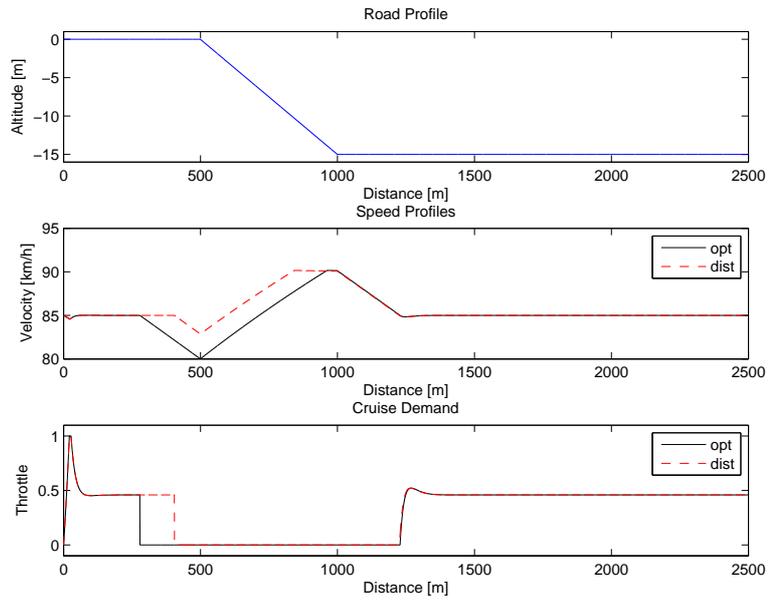


Figure 4.11: Plot of a simulation done in a 500m, -3% downhill slope with both an angle bias fault and a position bias fault.

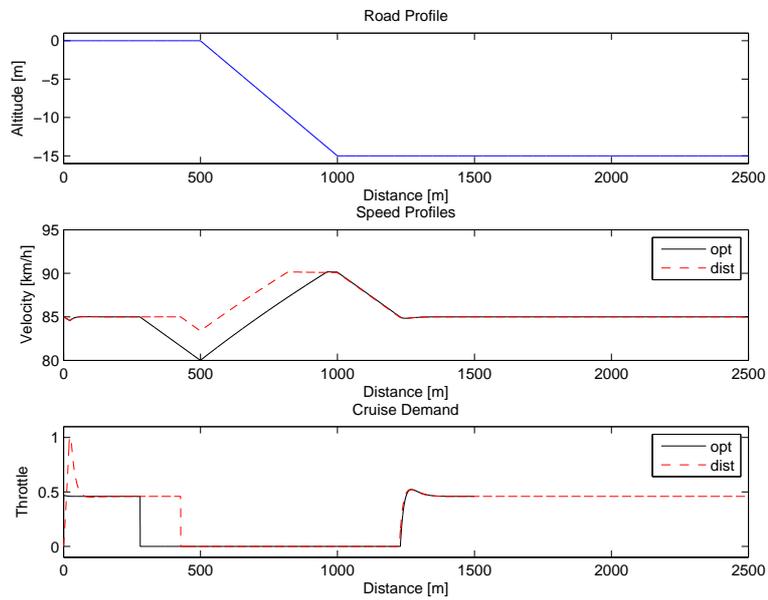


Figure 4.12: Plot of a simulation done in a 500m, 3% downhill slope with an angle bias fault, a position bias fault and a mass estimation error.

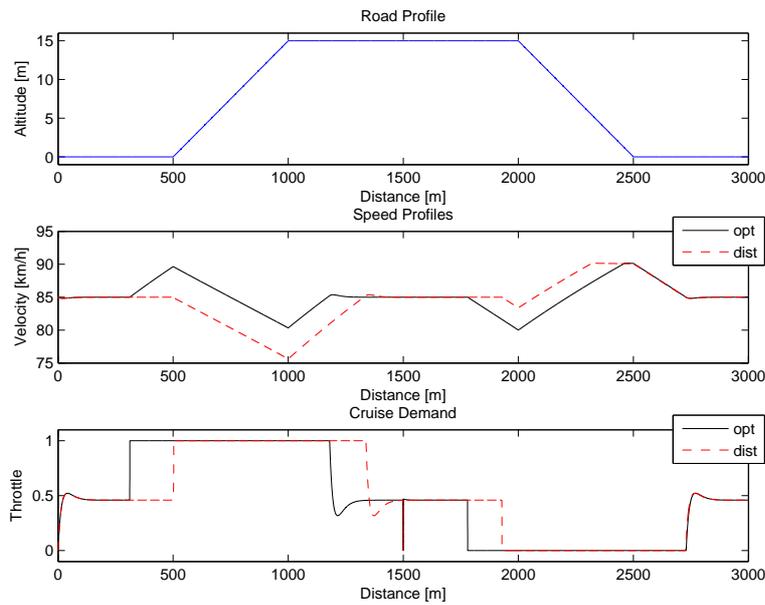


Figure 4.13: Plot of a simulation done with an angle bias fault, a position bias fault and a mass estimation error in a 500m, 3% uphill slope combined with a 500m, 3% downhill slope.

Several faults added together

Downhill 500m, -3%	Value	plateau, -3%	Value
Δ -Function		Δ -Function	
Optimal Scenario		Optimal Scenario	
Δf_{opt}	-12.65%	Δf_{opt}	-5.75%
Δt_{opt}	+1.02%	Δt_{opt}	-0.78%
$\Delta f_{opt} + c\Delta t_{opt}$	-11.73%	$\Delta f_{opt} + c\Delta t_{opt}$	-6.45%
Position bias +50m, angle bias -20%		Position bias +50m, Angle bias -20%, mass -10%	
Δf_{dist1}	-5.50%	Δf_{dist3}	-2.32%
Δt_{dist1}	+0.32%	Δt_{dist3}	+0.17%
$\Delta f_{dist1} + c\Delta t_{dist1}$	-5.21%	$\Delta f_{dist3} + c\Delta t_{dist3}$	-2.17%
Position bias +50m, Angle bias -20%, mass -10%			
Δf_{dist2}	-4.21%		
Δt_{dist2}	+0.22%		
$\Delta f_{dist2} + c\Delta t_{dist2}$	-4.01%		

Table 4.6: Results from simulations with several different faults, position bias, angle bias and mass fault, added together on a 500m 3% downhill slope (left column) and on a 500m 3% uphill slope combined with a 500m 3% downhill slope (right column). The standard PI cruise controller is used as reference.

Function	Value
Δf_{opt}	-5.43%
Δt_{opt}	-1.49%
Disturbed Scenario with Gearshift:	
Δf_{dws}	-3.23%
Δt_{dws}	-1.35%
Disturbed Scenario without Gearshift	
Δf_{dns}	-4.89%
Δt_{dns}	-0.94%

Table 4.7: Results from a disturbed scenario which forces a gearshift and a scenario where the gear shifting point has been lowered to avoid this unnecessary gear shift. The Δ values are calculated with the PI cruise controller as reference.

4.9 Gearshifts Caused by Disturbances

A disturbance could affect the vehicle in such a way that it will shift gear in an uphill slope. If the look ahead system calculates wrong due to false input data and accelerates the truck too late before a steep uphill slope the truck might lose enough velocity to be forced to shift down one step. With correct input data this unnecessary gear shift would have been avoided. There is also another possible scenario here, the system could be constructed to lower the shifting points slightly if the uphill slope is about to reach its end. This section will compare both scenarios to the optimal case. In Figure 4.14 the optimal speed trajectory is compared with a PI cruise controller and the disturbed optimal controller. According to the values received, see Table 4.7 this gearshift wastes more than 2 percentage units of the potential fuel that could be saved. There is also a minor loss in travel time, therefore avoiding this gearshift is valuable. In Figure 4.15 the down shift point has been lowered to avoid a gear shift at the end of the uphill slope. The results in Table 4.7 show that this is a much better solution than the previous one. Travel time is increased slightly compared to the scenario with the gearshift and fuel consumption is lower, not as good as in the optimal case but the loss is more acceptable. A look ahead system could and most likely should incorporate a system for gear selection based upon the knowledge of upcoming road, this however is outside the subject of this master thesis.

4.10 Conclusions

Several different disturbances covering positioning errors, faulty angle data, bad environmental models and mass errors have been simulated with this model. It can be seen that the environmental errors have less effect on the system than large angle or position faults. Wrong wheel radius almost only has an impact in steep uphill slopes. Miscalculations in the mass estimations also has a quite large impact on the system. A mass error combined with an angle fault and a position fault results in severely degraded performance compared to the optimal case. From the results in the section, since there is known that a mass error of 10% is hard to avoid, it definitely is of interest to keep the angle fault under 10%. Also by keeping the position error under 25m a lot has been won.

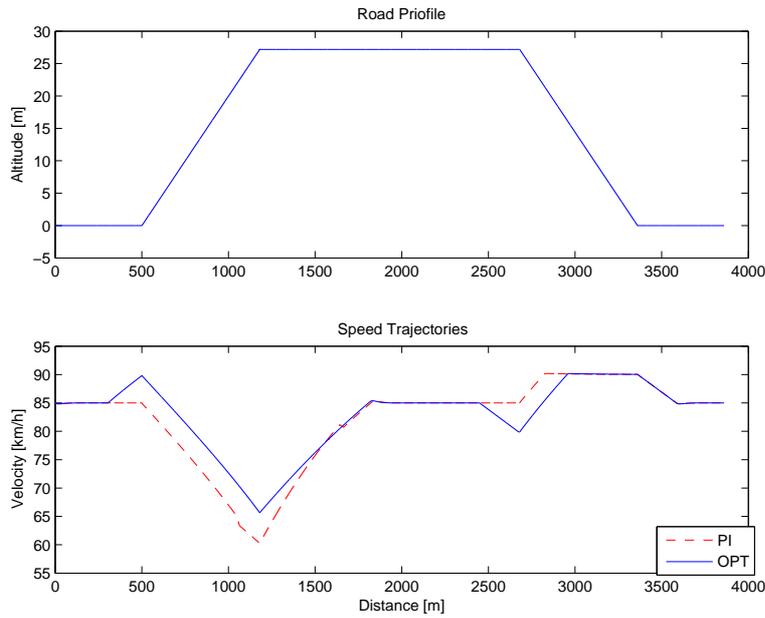


Figure 4.14: Speed profile with a gear shift for the PI cruise controller but not for the optimal strategy.

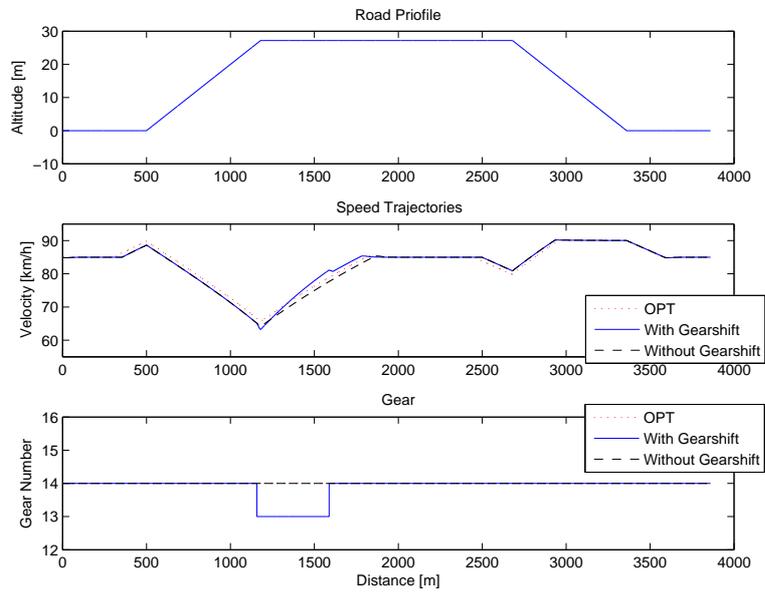


Figure 4.15: Speed profile with a gear shift due to disturbance and a speed profile with changed gear shifting point to avoid shifting gears due to disturbances.

Chapter 5

MPC with Dynamic Programming, DP-tool

DP tool is a program written by Ph.D. Student Erik Hellström at Linköping University. It uses a model predictive control (MPC) scheme with dynamic programming (DP) as optimizer to achieve an optimal control strategy of how to control the velocity and gear selection for a vehicle on a given road profile. The control algorithm and optimization are thoroughly described in Hellström (2005) and Hellström et. al. (2007). This approach is not bound to the simple road profiles which the analytic approach in previous chapters was limited to. Therefore it is possible to calculate optimal speed trajectories for real roads. To suite the purpose of this thesis some modifications to DP-tool are made, making it possible to optimize over defect input data but simulating the vehicle over undisturbed data. This chapter will briefly note the algorithms behind the DP and MPC used in this program. The vehicle model used in DP-tool is almost the same model as the one used in the previous chapters. Only a slightly different model of the gearbox is used, see Appendix A and the simulation uses a recorded torque map instead of the approximation in Section 2.1.

5.1 Model Predictive control

The idea of model predictive control is to use a model to predict future outputs of a system. This puts some requirements on the model since if its power to predict is not good enough the optimization algorithm would not be able to accurately choose the optimal strategy. The algorithm for MPC is explained in Ljung and Glad (2003) and is as follows:

1. At each instant t calculate predictions for the number of outputs y specified by a given horizon M , $\hat{y}(t+k|t)$, $k = 1, \dots, M$. These outputs will be based upon future control signals, $u(t+j)$, $j = 0, 1, \dots, N$, and at time t known measurable values.
2. Formulate a criterion based on these prediction and optimize in regard of the control signals, $u(t+j)$, $j = 0, 1, \dots, N$.
3. Send the optimal control signal $u(t)$.
4. Wait for the instant $t+1$ and repeat the algorithm from step 1.

5.2 Dynamic Programming

DP-tool utilizes dynamic programming (DP) to find the optimal control strategy. Basically what the DP part of DP-tool does is to find a solution to a shortest path problem. Dynamic programming is covered in Bertsekas (2000) and Appelgren, Holvid and Zachrisson (1972) here only the shortest path algorithm will be noted.

Shortest Path DP Algorithm (Bertsekas (2000) and Hellström (2005))

$a_k^{i,j}$, transition cost at step k from state $i \in S_k$ to state $j \in S_{k+1}$

$a_N^{i,t}$, terminal cost of state $i \in S_N$

$g_k(i, u_k^{i,j}, w_k)$, weighting function to define the cost of a policy

w_k , a random disturbance

J , the cost function

The cost $a_k^{i,j}$ is equal to $g_k(i, u_k^{i,j}, w_k)$ where $u_k^{i,j}$ is the control that causes the transition between state i and j . The terminal cost of state i is equal to $g_N(i)$. In DP-tool the cost of the final stage $g_N(i) = 0$ and w_k is the road slope angle and is considered as known.

1.

$$J_N(i) = a_N^{i,t}, \quad i \in S_N$$

2.

$$k = N - 1$$

3.

$$J_k(i) = \min_{j \in S_{k+1}} \left\{ a_k^{i,j} + J_{k+1}(j) \right\}, \quad i \in S_k$$

repeat for $k = N - 2, N - 3, \dots, 1$

The optimal cost $J_0(s)$ is equal to the cost of cheapest trajectory between s and t . The control sequence of the cheapest trajectory is the optimal control sequence.

5.3 Complete Control Algorithm

The complete control algorithm for the system can be found in Hellström (2005) but since this thesis work was published the system has evolved and some changes have been made. Basically the system looks at a horizon of $1500m$ with a step size of $50m$. The optimal speed for the vehicle every $50m$ of the entire horizon is calculated and a control is used to get the vehicle to the optimum speed at the next step. The optimal control for the next step is defined by the cheapest way to reach the end point of the horizon. Then after $50m$ the entire optimization is redone with a new horizon and also fault correction based upon current speed and the during last optimization predicted speed is applied and so on.

5.3.1 Cost Function

A cost function is introduced to weigh fuel consumption and travel time this cost function is given in Equation 5.1 and was already noted in Section 2.7.

$$I = M + \beta T \quad (5.1)$$

To make the cost function in 5.1 usable in dynamic programming the look ahead horizon is divided into N steps of length h . This brings the expression given in Equation 5.2.

$$J = \sum_{k=0}^{N-1} g_k(v_k, v_{k+1}, u_k, g_k, g_{k+1}, \alpha_k) \quad (5.2)$$

Where g_k is given in Equation 5.3. For the optimization algorithm the cost function is including not only consumption, $m_{f,k}$ and travel time t_k , but also velocity changes $|v_k - v_{k+1}|$, and gear shifts $|G_k - G_{k+1}|$. v_k represents velocity at stage k , u_k the control at stage k .

$$g_k(v_k, v_{k+1}, \underline{u}_k, \underline{u}_{k+1}, \alpha_k) = [Q_1, Q_2, Q_3, Q_4] \begin{bmatrix} m_{f,k} \\ t_k \\ |v_k - v_{k+1}| \\ \chi(|G_k - G_{k+1}|) \end{bmatrix} \quad (5.3)$$

$$k = 0, 1, \dots, N - 1$$

Where χ is the unit step:

$$\chi(t) = \begin{cases} 1 & , t > 0 \\ 0 & , t \leq 0 \end{cases} \quad (5.4)$$

The penalty factors have to be decided for use in the optimization. First the penalty on the amount of fuel used is assumed, $Q_1 = 1$. The two parameters penalizing gear shift and velocity changes are chosen on experience with the system. Still the penalizing factor for travel time has to be calculated or decided in some way. In Hellström et al. (2007) a way to calculate $\beta = Q_2$ is introduced and goes as follows:

If Equation 2.1 is considered with δ as a control signal, u , the equation can be written as in Equation 5.6. Together with Equation 2.28 and 2.29, u can be written as in Equation 5.7. This calculation of β is only correct as long as the vehicle is traveling in a slope where it can keep it is reference velocity.

$$I = M_f + \beta T \quad (5.5)$$

where M_f is the integrated amount of fuel and T the total time.

$$\hat{T}_{map}(N, u) = a_e N + b_e u + c_e \quad (5.6)$$

$$\hat{u} = c_v \hat{v}^2 + c_v \hat{v} + f(\alpha) \quad (5.7)$$

The fuel mass flow into the cylinders is stated in Equation 2.34 and 2.35 which together with Equation 2.29, 2.31 and $\delta = u$ gives Equation 5.8.

$$\frac{dm}{ds}(v, u) = c_3 u \quad (5.8)$$

The cost function $I = M + \beta T$ is then:

$$\hat{I}(\hat{v}) = \int_{s^0}^{s^f} \left(c_3 (2c_{v^2} \hat{v}^2 + c_v \hat{v} f(\alpha)) + \frac{\beta}{\hat{v}} \right) ds \quad (5.9)$$

A stationary point to \hat{I} is desired and is found by taking the derivative and set it equal to zero:

$$\frac{\hat{I}}{d\hat{v}} = \int_{s^0}^{s^f} \left(c_3 (2c_{v^2} \hat{v} + c_v) - \frac{\beta}{\hat{v}^2} \right) ds = 0 \quad (5.10)$$

Finally by solving Equation 5.10, Equation 5.11 is received.

$$\beta = c_3 \hat{v}^2 (2c_{v^2} \hat{v} + c_v) \quad (5.11)$$

Chapter 6

Results DP-tool

This chapter contains results of simulations done with DP-tool over different road profiles with different types of disturbances. Since a numerical approach is used to solve the optimization problem almost any type of disturbance are theoretically possible to simulate. However due to limitations in the implementation some types of disturbances aren't possible to implement without doing changes in the optimization algorithm. In some other cases the results are somewhat odd and one should consider them with the knowledge that this is a numerical approach in mind. The control system calculates the optimal speed trajectory based upon the coming 1 500m of the road. The calculation is redone every 50m. Reference speed is set to 85km/h and the vehicle is allowed to speed up to 90km/h which is a hard limit and the soft lower limit is set to 80km/h. In other word if the vehicle cannot hold the minimum velocity in a steep uphill slope it is allowed to drop to a lower speed, but it will be punished in the cost function. All simulations starts at reference speed and uses gear number 12 from the beginning.

6.1 Basic Roadprofiles

To be able to compare the results from DP-tool to those given in Chapter 3 and 4 simulations with the same road profiles and disturbances will also be done in DP-tool. For the cases with perfect input data see Figure 6.1, 6.2 and 6.3. The results from the simulations shown in these plots are in line with those from the analytical optimization approach, see Chapter 3.

Looking at the throttle level plots for these three cases something worth noting is obvious, the PI cruise controller used to set the speed has trouble handling the discrete speed steps given to it by the control algorithm. To reduce this ripple effect the signal containing the reference speed to the cruise controller passes through a low pass filter. Still however not all undesired variations are removed. The low pass filter has an unwanted effect in that it has a small negative influence on the result from the MPC. The effects of the ripple and the low pass filter are not very large but it should not be neglected because it reduces the potential of the MPC slightly. Changing the PI controller parameters can solve this problem but while improving MPC controller performance the standard cruise controller performance gets degraded. Thus a balance where performance is acceptable for both cases has to be found. When DP-tool has been tested on board this problem has not been seen most likely due to the more advanced cruise controller available there.

DP-tool seems to suffer in the same way from disturbances as the system in Chapter 4 but the difference between the optimal scenario and the disturbed ones are smaller. This is interesting and DP-tool does seem to be a lot less sensitive to disturbances which will be seen in this chapter. There are most likely several reasons why this system is more robust. First of all it is not as exact as the precalculated speed profile from the beginning something which has to do with the numerical optimization and the MPC. Secondly since it does calculate a trajectory every 50 m and also has a feedback with the current speed it has the opportunity correct prediction errors. This is most likely the greatest part to why it is so much more robust. Also DP-tool does not set the fuel injection to zero or max directly instead the driver demand is ramped up and down. This with the help of the feedback built into the system allows for changes in the ramps if it is discovered that the model and the real world differs.

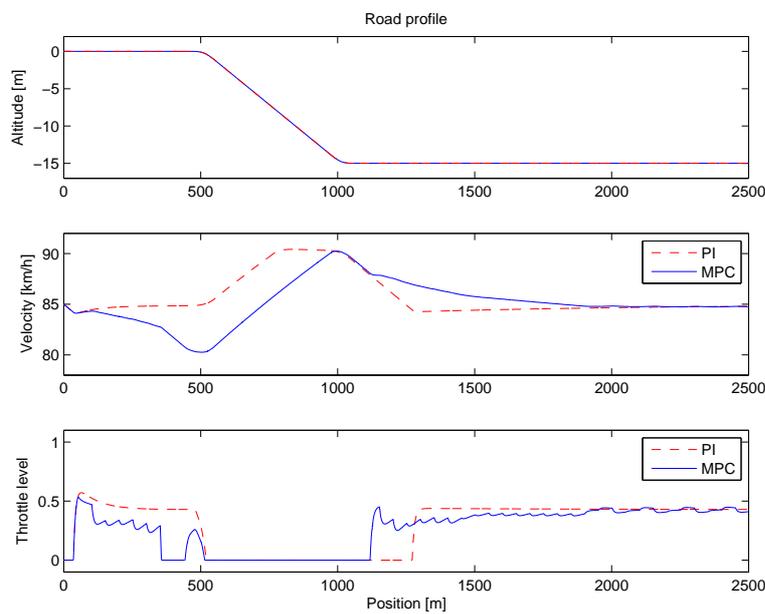


Figure 6.1: A 500m, 3% downhill slope simulated in DP-tool with perfect input data, the results are: $\Delta f = -12.51\%$ and $\Delta t = 0.71\%$.

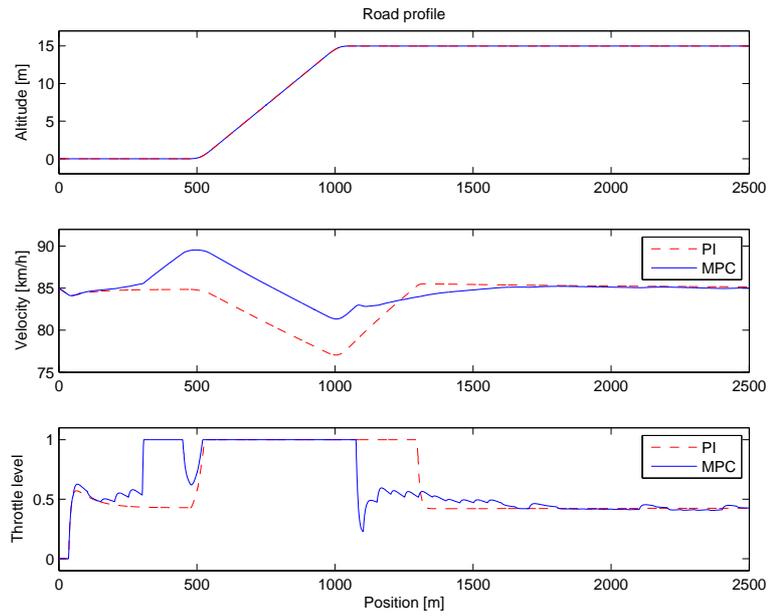


Figure 6.2: A 500m, 3% uphill slope simulated in DP-tool with perfect input data, the results are: $\Delta f = 0.36\%$ and $\Delta t = -1.52\%$.

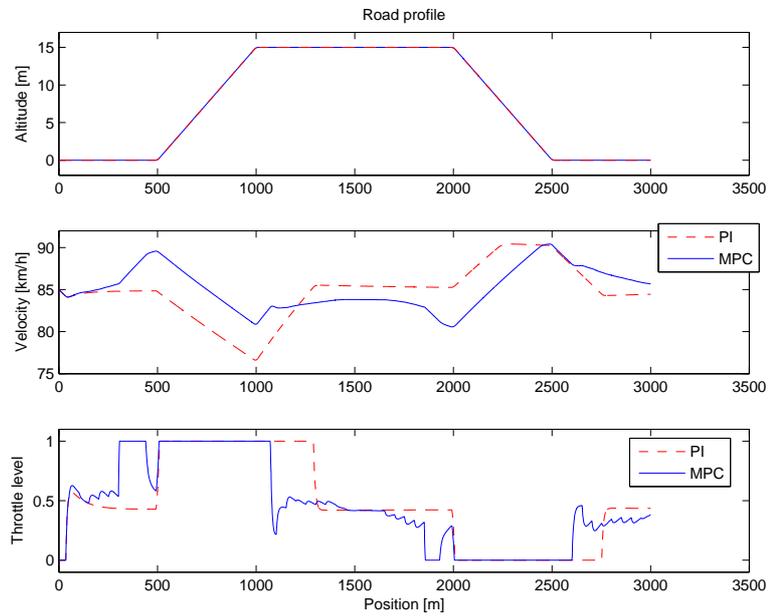


Figure 6.3: A 500m, 3% plateau simulated in DP-tool with perfect input data, the results are: $\Delta f = -4.48\%$ and $\Delta t = -0.49\%$.

6.2 Disturbance of Basic Road Profiles

6.2.1 Position Bias Faults

The result of a position bias fault in one of these simple road profiles is similar to the effect of having the same fault in the simple model used earlier. However the difference in fuel consumption and travel time between the disturbed road profile and the optimal one are not as large as in the simple optimization used earlier. This is most likely due to the feedback in the MPC. In Figure 6.4 the behaviour of the dynamic programming based optimization is shown when a position bias fault is added in a steep downhill slope. The behaviour in a steep uphill slope is presented in Figure 6.5. The resulting changes in travel time and fuel consumption are collected in Table 6.1.

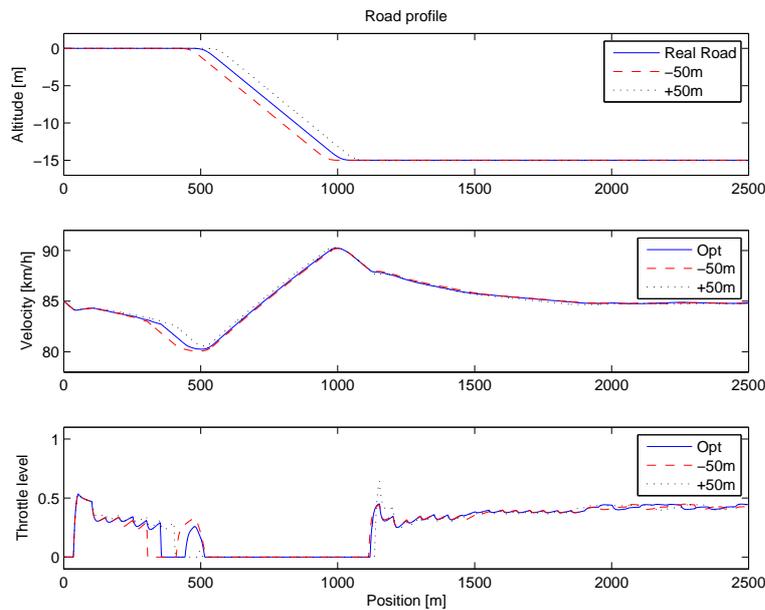


Figure 6.4: The Optimal solution calculated by DP-tool compared to solutions calculated with position bias errors in the input data.

Downhill		Uphill	
Δ -Function	Value	Δ -Function	Value
Optimal scenario		Optimal scenario	
Δf_{opt}	-12.27%	Δf_{opt}	+0.36%
Δt_{opt}	+0.70%	Δt_{opt}	-1.52%
+50m position bias:		+50m position bias:	
Δf_{+50m}	-11.63%	Δf_{+50m}	+0.46%
Δt_{+50m}	+0.62%	Δt_{+50m}	-1.46%
-50m position bias:		-50m position bias:	
Δf_{-50m}	-12.80%	Δf_{-50m}	+0.47%
Δt_{-50m}	+0.78%	Δt_{-50m}	-1.72%

Table 6.1: The resulting changes in fuel consumption and travel time corresponding to the cases simulated with position bias errors, Figure 6.4 and 6.5. The standard PI cruise controller is used as reference.

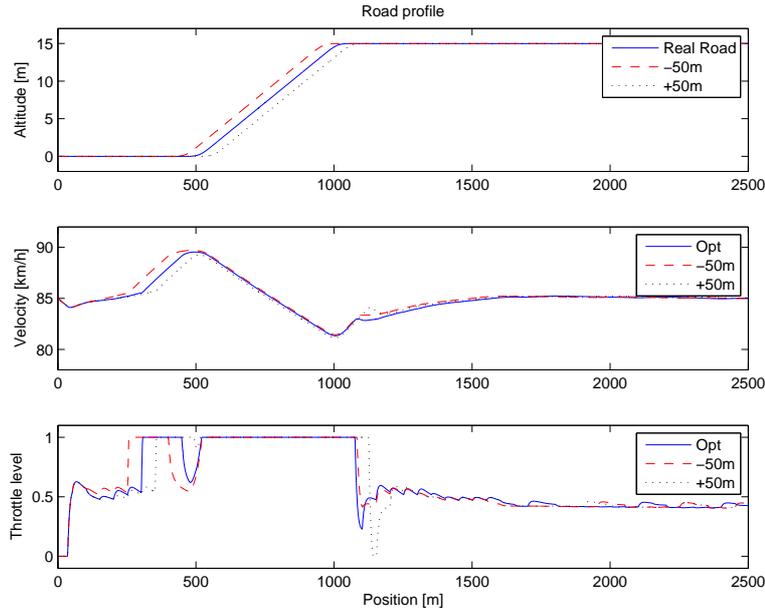


Figure 6.5: The Optimal solution calculated by DP-tool compared to solutions calculated with position bias errors in the input data.

6.2.2 Angle Bias Faults

Adding a bias error to the angle data in a steep uphill or downhill slope also gives similar behaviour as with the precalculated speed trajectory, see Figure 6.6 and Figure 6.7. Again it is obvious that the MPC with dynamic programming is much better at handling inaccurate input data which is confirmed by the Δf and Δt values in Table 6.2. The reason for the non existent difference between the optimal case and the case where the downhill slope appears steeper than it really is to the system is because that a 500m 3% downhill slope is enough to accelerate the vehicle from 80km/h to 90km/h. Thus because of the 80km/h lower speed limit the control system cannot slow the vehicle more even if it still would be possible to accelerate to maximum allowed velocity, 90km/h.

Downhill		Uphill	
Δ -Function	Value	Δ -Function	Value
Optimal scenario		Optimal scenario	
Δf_{opt}	-12.27%	Δf_{opt}	+0.36%
Δt_{opt}	+0.70%	Δt_{opt}	-1.52%
+20% angle bias		+20% angle bias	
$\Delta f_{+20\%}$	-12.27%	$\Delta f_{+20\%}$	+0.52%
$\Delta t_{+20\%}$	+0.70%	$\Delta t_{+20\%}$	-1.67%
-20% angle bias		-20% angle bias	
$\Delta f_{-20\%}$	-7.72%	$\Delta f_{-20\%}$	+0.03%
$\Delta t_{-20\%}$	+0.20%	$\Delta t_{-20\%}$	-0.76%

Table 6.2: The resulting difference in fuel consumption and travel time from simulations with angle bias faults, Figure 6.6 and 6.7. The standard PI cruise controller is used as reference.

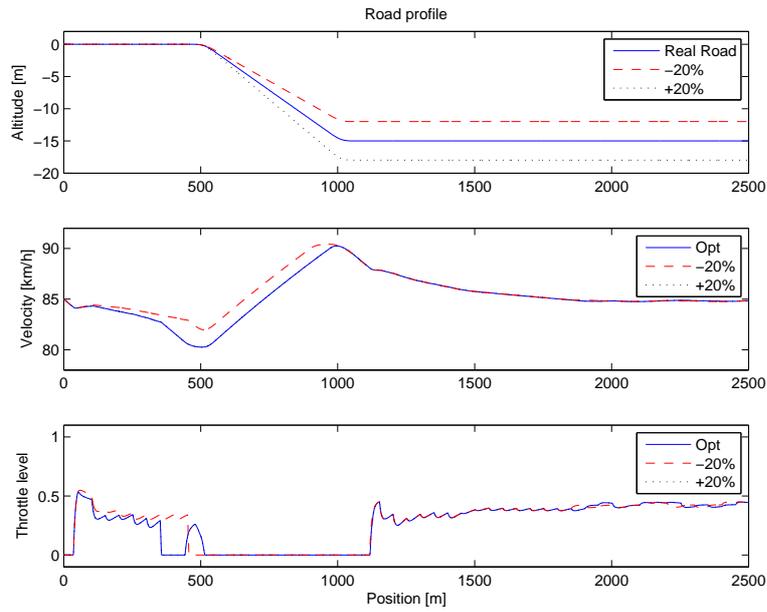


Figure 6.6: Simulations done over a step downhill slope with angle bias errors in the input data compared to the optimal solution.

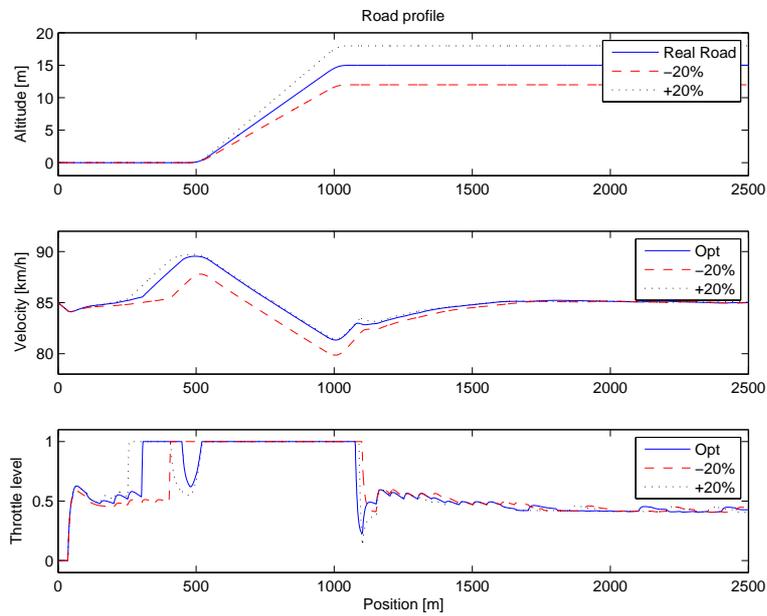


Figure 6.7: Simulations done over a step uphill slope with angle bias errors in the input data compared to the optimal solution.

6.2.3 Mass Errors

Next fault to take a look at using DP-tool is a vehicle mass estimation error. One problem, as explained earlier, with the mass estimation error is that it is hard to get rid off since the estimation has to be made on board. Thus the estimation has to be done with available resources in the truck. Therefore a mass error of at least 10% should be accounted for when simulating a look ahead system. A mass error will bring a direct error in the normal and gravitational force calculations, Equation 2.24 and 2.25. Also the vehicle inertia will be affected by a mass error. In the downhill slope the outcome is hardly influenced by the a mass error, see Figure 6.8. A higher or lower gravitational force will be canceled by a lower or higher rolling resistance. In an uphill slope there are on the other hand quite large differences. This because of a for the control system increased mass results in estimations of an increased gravitational force, an increased rolling resistance and an increased vehicle inertia. If the vehicle mass is estimated lower than it really is the situation is the opposite. In an uphill slope all faults from parts that are influenced by a mass error are added together resulting in quite large deviations from the optimal speed trajectories. In downhill slopes the increased or decreased resistance forces are canceled by an increased or decreased gravitational force.

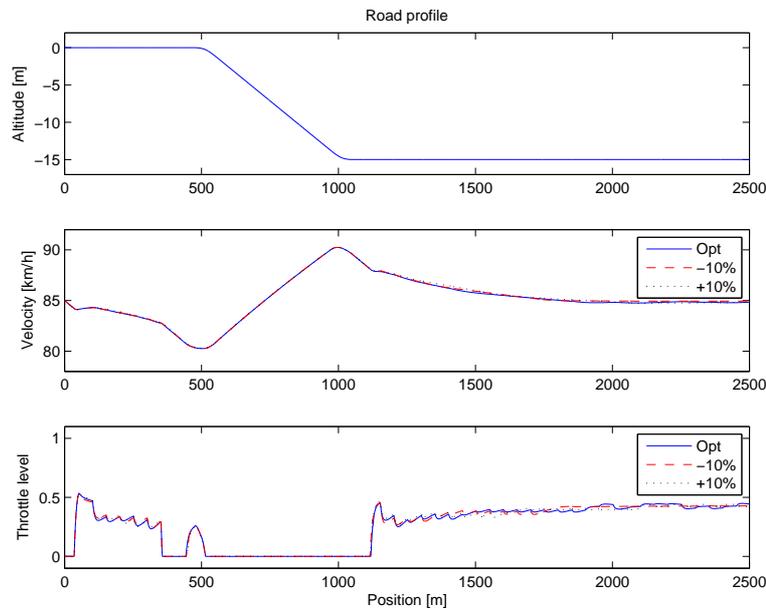


Figure 6.8: A 3%, 500m downhill slope where optimization has been done with mass errors.

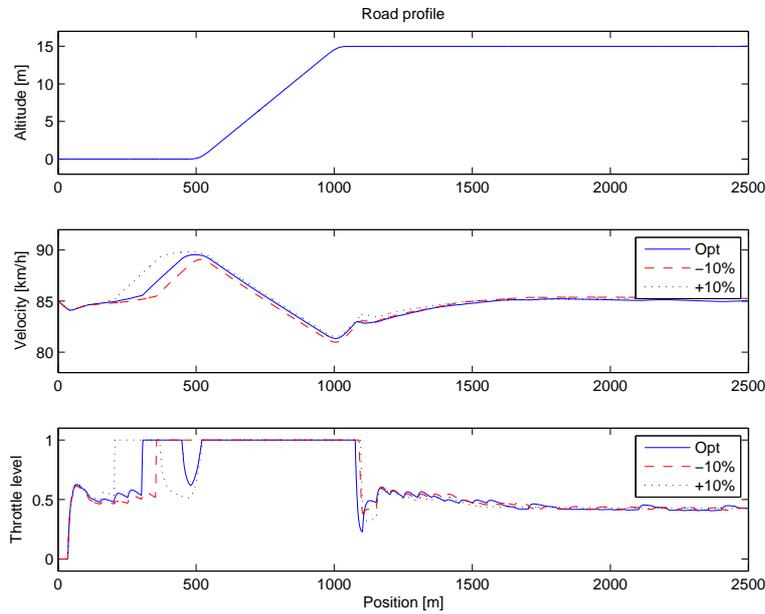


Figure 6.9: Simulations with mass faults done over a 3%, 500m uphill slope.

Downhill		Uphill	
Δ -Function	Value	Δ -Function	Value
Optimal scenario		Optimal scenario	
Δf_{opt}	-12.27%	Δf_{opt}	+0.36%
Δt_{opt}	+0.70%	Δt_{opt}	-1.52%
+10% mass fault:		+10% mass fault:	
$\Delta f_{+10\%}$	-12.24%	$\Delta f_{+10\%}$	+0.80%
$\Delta t_{+10\%}$	+0.65%	$\Delta t_{+10\%}$	-1.87%
-10% mass fault:		-10% mass fault:	
$\Delta f_{-10\%}$	-11.97%	$\Delta f_{-10\%}$	+0.58%
$\Delta t_{-10\%}$	+0.65%	$\Delta t_{-10\%}$	-1.43%

Table 6.3: Results for simulations done with mass errors, Figure 6.8 and 6.9. The reference is the standard PI cruise controller.

6.2.4 Wheel Radius Error

Continuing through the basic model errors, optimization with the wrong wheel radius in DP-tool gets a closer look. As said in Chapter 4 the outcome of a wheel radius error is slightly harder to predict. The pattern though is again the same with downhill hardly affected and quite large differences in steep uphill slopes. Values and plots from the simulations are presented in Table 6.4 and Figure 6.10 and 6.11. A false wheel radius will result in wrongly calculated torque from the aerodynamic drag force, the gravitational force and the rolling resistance force. Also the engine torque will be wrongly predicted. In steep uphill slopes these errors will add to each other while in a downhill slope the errors carries different signs and more or less cancel each other.

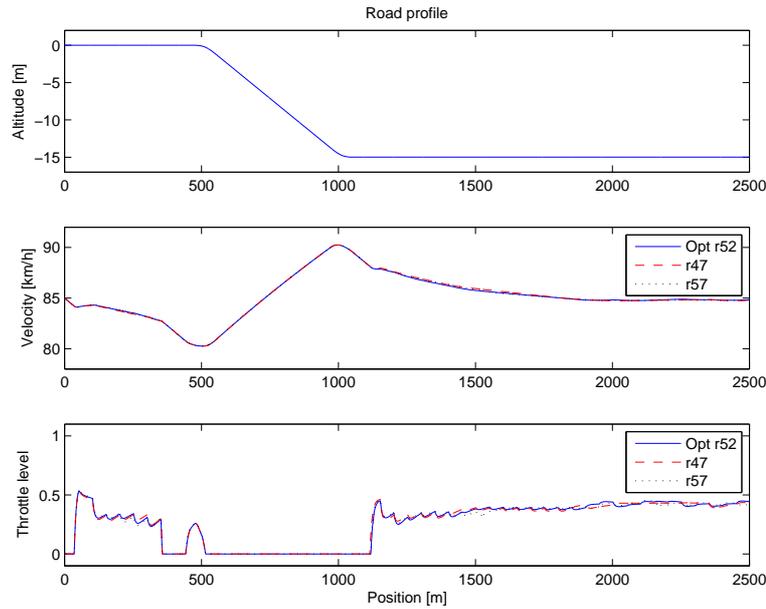


Figure 6.10: The optimal solution calculated by DP-tool compared to solutions calculated with wheel radius errors in the input data.

Downhill		Uphill	
Δ -Function	Value	Δ -Function	Value
Optimal scenario		Optimal scenario	
Δf_{opt}	-12.27%	Δf_{opt}	+0.36%
Δt_{opt}	+0.70%	Δt_{opt}	-1.52%
+5cm radius fault, r57		+5cm radius fault, r57	
Δf_{r57}	-12.47%	Δf_{r57}	+0.58%
Δt_{r57}	+0.70%	Δt_{r57}	-1.87%
-5cm radius fault, r47		-5cm radius fault, r47	
Δf_{r47}	-12.45%	Δf_{r47}	+0.30%
Δt_{r47}	+0.68%	Δt_{r47}	-1.35%

Table 6.4: The resulting delta values corresponding to the simulated cases in Figure 6.10 and 6.11. The PI cruise controller is used as reference.

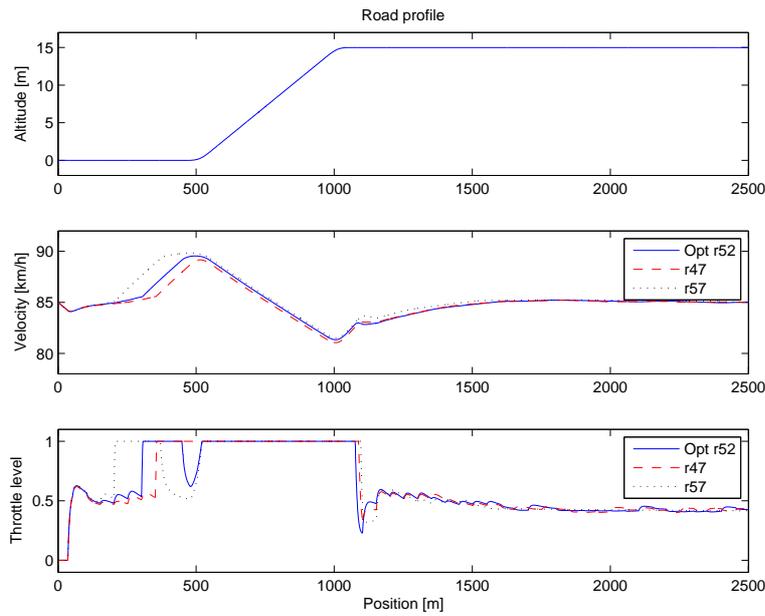


Figure 6.11: The optimal solution calculated by DP-tool compared to solutions calculated with wheel radius errors in the input data.

6.2.5 Changes in Aerodynamic Drag and Rolling Resistance

The first of the two disturbances to be studied in this subsection is the aerodynamic drag force. The fault is implemented as a change in the constant c_w in Equation 2.22. In the simulation environment the constant is left at its nominal value, $c_w = 0.6$, while it is changed to $c_w = 0.66$ and $c_w = 0.54$ in the optimization. As indicated by Table 6.5 and Figure 6.12 and 6.13. The effects could be seen as within the margin of error especially since one of the disturbed cases gives better performance than the optimal scenario. In the case of a rolling resistance error the result is following the same pattern as for the aerodynamic drag force. The disturbance is introduced in the rolling resistance force in the optimization, see Equation 2.23. The differences are a bit larger between the optimal solution and the disturbed ones than what was the case with the aerodynamic drag force, see Figure 6.14 and 6.15 and Table 6.6.

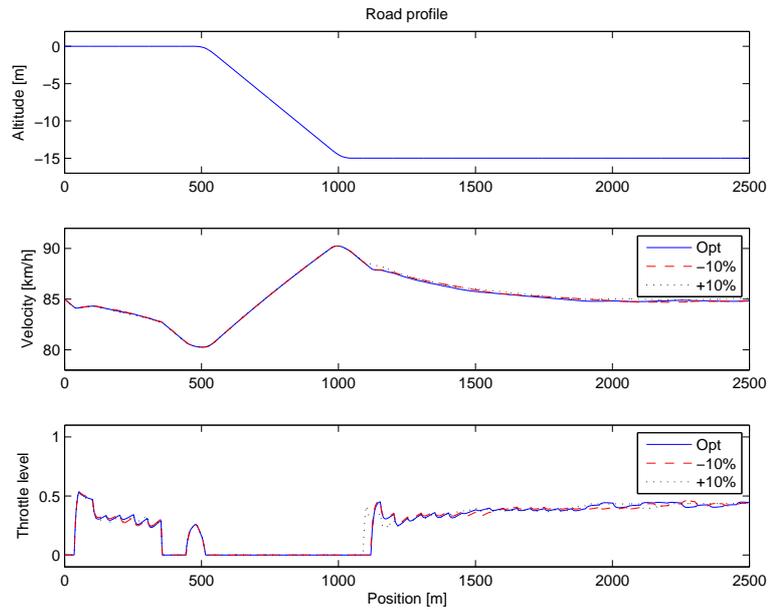


Figure 6.12: The optimal solution calculated by DP-tool compared to solutions calculated with a fault in the aerodynamic drag force in the input data.

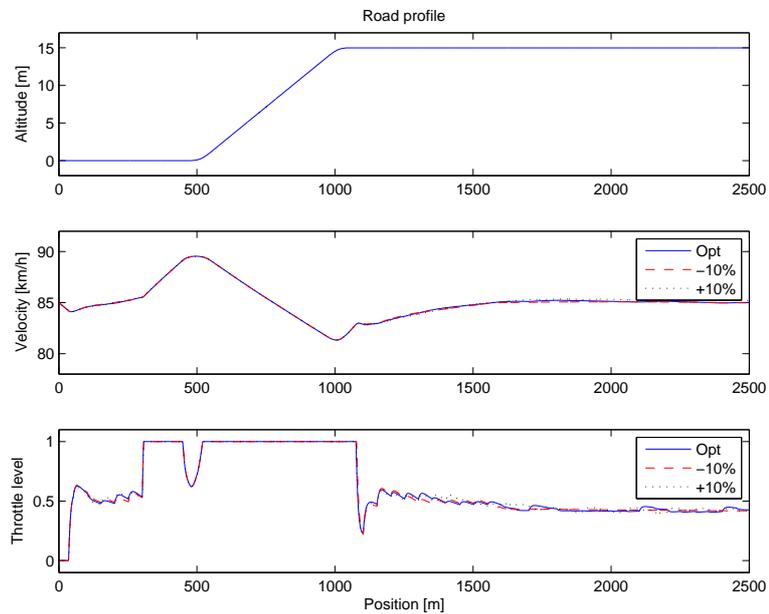


Figure 6.13: The optimal solution calculated by DP-tool compared to solutions calculated with a fault in the aerodynamic drag force in the input data.

Downhill		Uphill	
Δ -Function	Value	Δ -Function	Value
Optimal scenario		Optimal scenario	
Δf_{opt}	-12.27%	Δf_{opt}	+0.36%
Δt_{opt}	+0.70%	Δt_{opt}	-1.52%
+10% aerodynamic drag		+10% aerodynamic drag	
$\Delta f_{+10\%}$	-11.57%	$\Delta f_{+10\%}$	+0.62%
$\Delta t_{+10\%}$	+0.59%	$\Delta t_{+10\%}$	-1.58%
-10% aerodynamic drag		-10% aerodynamic drag	
$\Delta f_{-10\%}$	-12.30%	$\Delta f_{-10\%}$	+0.28%
$\Delta t_{-10\%}$	+0.66%	$\Delta t_{-10\%}$	-1.50%

Table 6.5: The difference in fuel consumption and travel time for simulation with bad aerodynamic drag force estimation compared with the optimal case. The PI cruise controller is used as reference.

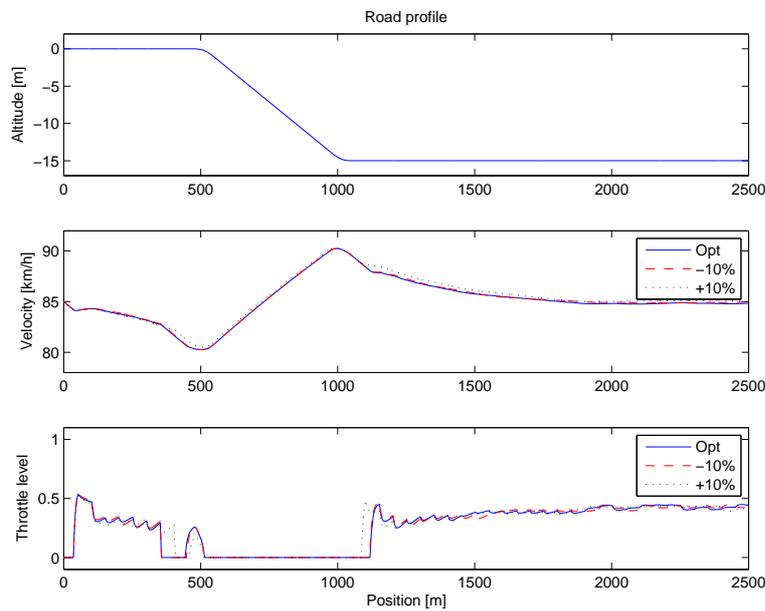


Figure 6.14: The optimal solution in a steep downhill slope compared to solutions calculated with a fault in the rolling resistance in the input data.

Downhill		Uphill	
Δ -Function	Value	Δ -Function	Value
Optimal scenario		Optimal scenario	
Δf_{opt}	-12.27%	Δf_{opt}	+0.36%
Δt_{opt}	+0.70%	Δt_{opt}	-1.52%
+10% rolling resistance		+10% rolling resistance	
$\Delta f_{+10\%}$	-11.05%	$\Delta f_{+10\%}$	+0.74%
$\Delta t_{+10\%}$	+0.44%	$\Delta t_{+10\%}$	-1.71%
-10% rolling resistance		-10% rolling resistance	
$\Delta f_{-10\%}$	-11.92%	$\Delta f_{-10\%}$	+0.67%
$\Delta t_{-10\%}$	+0.64%	$\Delta t_{-10\%}$	-1.47%

Table 6.6: The difference in fuel consumption and travel time for simulation with bad rolling resistance estimation compared with the optimal case. The standard PI cruise controller is used as reference.

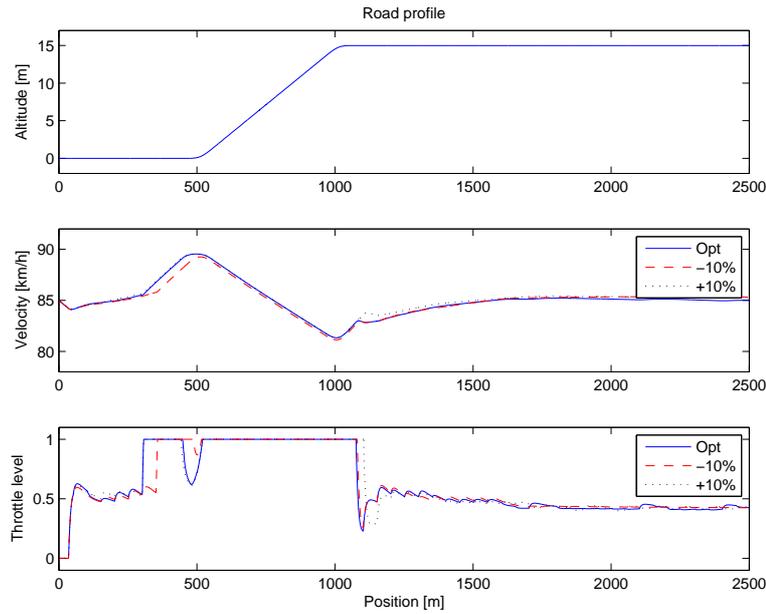


Figure 6.15: The optimal solution in a steep uphill slope compared to solutions calculated with a fault in the rolling resistance in the input data.

6.2.6 Combining Disturbances

To find a worst case scenario several different disturbances of the input data to the optimization are simulated simultaneously. A look at Figure 6.16 and 6.17 shows that with both a positive position bias error and a negative angle fault the vehicle hardly accelerates before a steep uphill slope or decelerates before a steep downhill slope. As a result a lot of the possible gains in fuel or time are lost due to the disturbances. This is also confirmed by the values in Table 6.7. Adding a -10% mass fault to the system does not make that much difference but the result is slightly worse. Plots of a steep downhill slope and a steep uphill slope with a $+50m$ position bias, a -20% angle bias and a -10% mass error are presented in Figure 6.18 and 6.19. The corresponding differences in travel time and fuel consumption are presented in Table 6.8.

Downhill		Uphill	
Δ -Function	Value	Δ -Function	Value
Optimal scenario		Optimal scenario	
Δf_{opt}	-12.27%	Δf_{opt}	$+0.36\%$
Δt_{opt}	$+0.70\%$	Δt_{opt}	-1.52%
Disturbed scenario		Disturbed scenario	
Δf_{Dist}	-5.24%	Δf_{Dist}	$+0.12\%$
Δt_{Dist}	$+0.04\%$	Δt_{Dist}	-0.29%

Table 6.7: Results from simulations done with both a $+50m$ bias fault and a -20% angle fault. To the left downhill and to the right uphill. The standard PI cruise controller is used as reference.

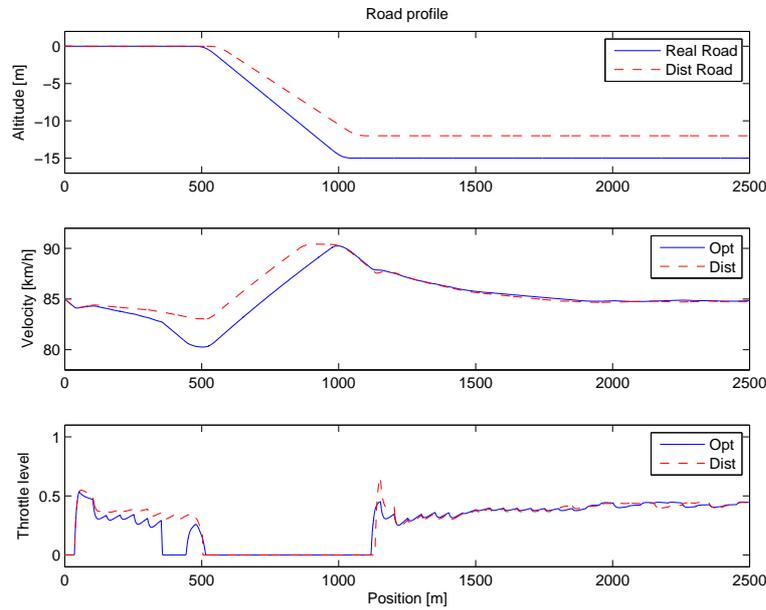


Figure 6.16: The optimal solution in a downhill slope calculated by DP-tool compared to solutions calculated with a $+50m$ bias fault in the position and a -20% bias fault in the angle.

Downhill		Uphill	
Δ -Function	Value	Δ -Function	Value
Optimal scenario		Optimal scenario	
Δf_{opt}	-12.27%	Δf_{opt}	$+0.36\%$
Δt_{opt}	$+0.70\%$	Δt_{opt}	-1.52%
Disturbed scenario		Disturbed scenario	
Δf_{Dist}	-5.01%	Δf_{Dist}	-0.08%
Δt_{Dist}	$+0.01\%$	Δt_{Dist}	$+0.26\%$

Table 6.8: Results from simulations done with a $+50m$ position error, a -20% angle fault and a -10% mass error. To the left downhill and to the right uphill. As reference is the standard PI cruise controller used.

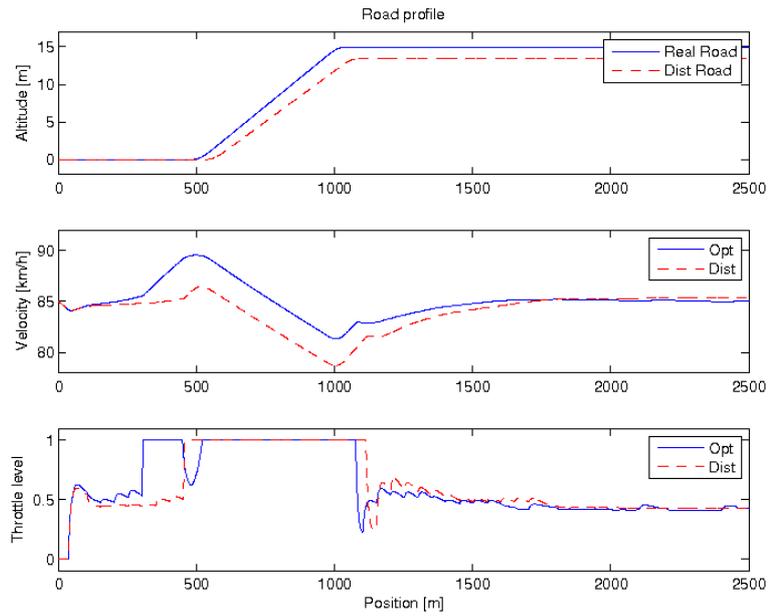


Figure 6.17: The optimal solution in an uphill slope calculated by DP-tool compared to solutions calculated with a $+50m$ bias fault in the position and a -20% bias fault in the angle.

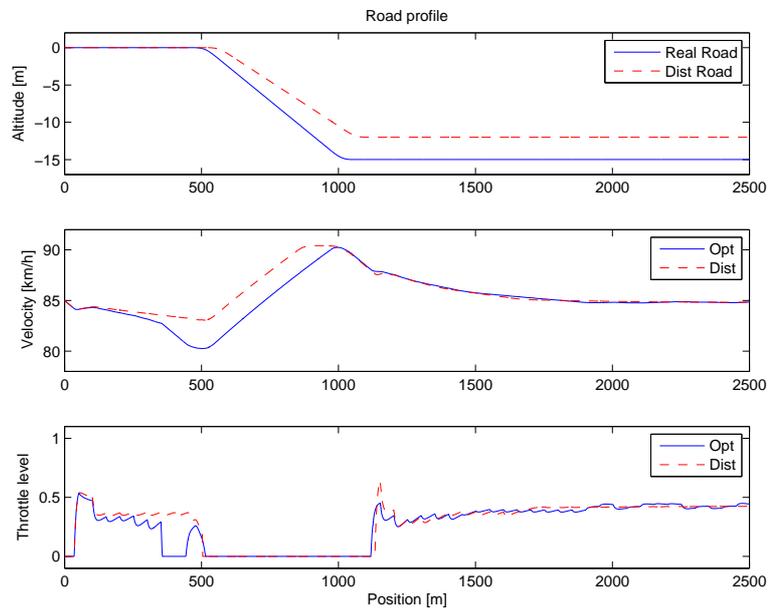


Figure 6.18: The optimal solution in a downhill slope calculated by DP-tool compared to solutions calculated with a $+50m$ bias fault in the position, a -20% bias fault in the angle and a -10% fault in the vehicle mass.

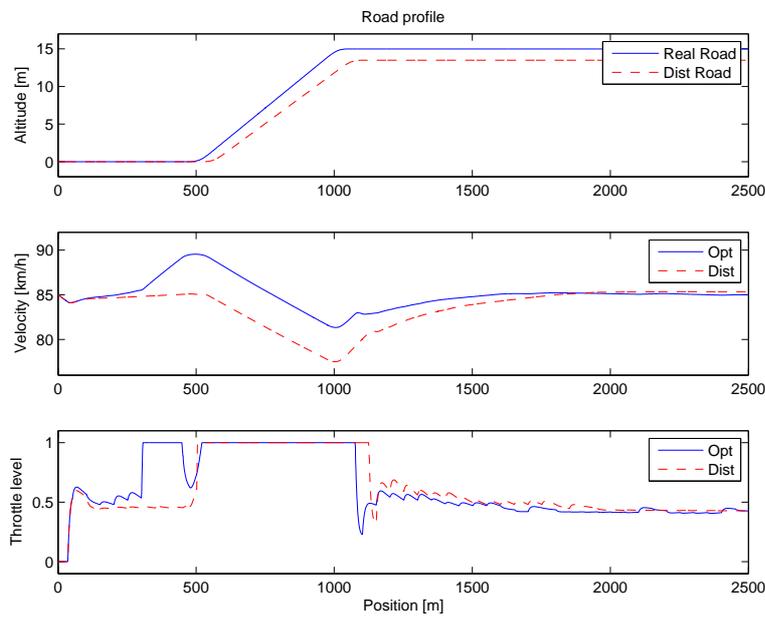


Figure 6.19: The optimal solution in an uphill slope calculated by DP-tool compared to solutions calculated with a $+50m$ bias fault in the position, a -20% bias fault in the angle and a -10% fault in the vehicle mass.

6.3 Real Road Profiles

While the simple road profiles are interesting from an analytical point of view the real interest is of course in using a look-ahead system on real roads. This section contains simulations done over the road Södertälje toward Norrköping (the E4), see Figure 6.20. This road, especially the first part of it, is good for speed optimizations of this kind which can be seen in the results and also by looking at the altitude plot of the road. The first part of the road has up- and downhill slopes of a length around $1 - 2\text{km}$ each which are quite optimal for this type of optimizations. In Appendix B results from simulations done over the road between Uppsala and Gävle are presented. Uppsala toward Gävle is a road which isn't really suited for this type of speed optimizations. As can be seen in the altitude plot there are only smaller local variations and it is not possible to optimize over a slowly climbing or descending road with the look ahead horizon used here. The starting gear in all simulation is gear number 12, the reference speed is set to 85km/h and the vehicle mass to $40\,000\text{kg}$. Again the vehicle is allowed to accelerate to a maximum of 90km/h and the minimum speed is set to 80km/h , a boundary which if necessary may be broken.

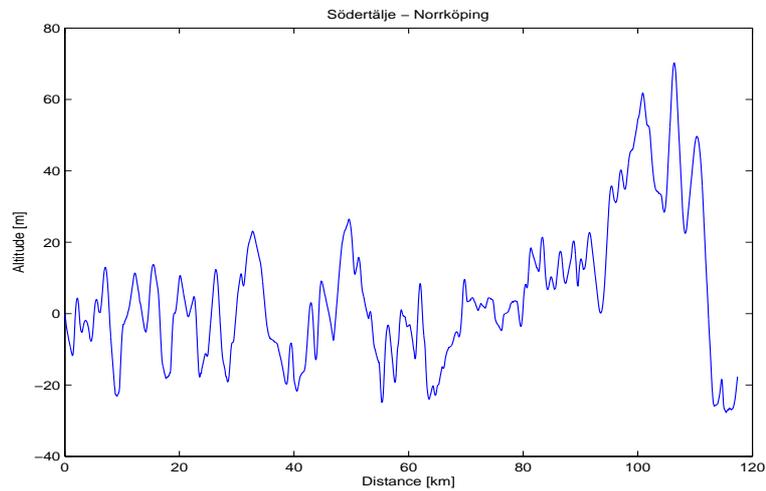


Figure 6.20: The road between Södertälje and Norrköping.

6.3.1 Position Bias Errors on Real Road Sections

Adding position bias faults to the system on a real road simulates an inexact map or, bad GPS signal or possibly delays in the calculation of the trajectory. DP-tool is only mildly affected of a 50m bias fault like this road section. The results in fuel consumption and travel time are presented in Table 6.9.

Södertälje - Norrköping			
Δ -Function	Value	Δ -Function	Value
Optimal scenario		+25m position bias	
Δf_{opt}	-1.74%	Δf_{+25m}	-1.74%
Δt_{opt}	-0.04%	Δt_{+25m}	-0.02%
$\Delta f_{opt} + c\Delta t_{opt}$	-1.78%	$\Delta f_{+25m} + c\Delta t_{+25m}$	-1.71%
-50m position bias		+50m position bias	
Δf_{-50m}	-1.72%	Δf_{+50m}	-1.65%
Δt_{-50m}	-0.12%	Δt_{+50m}	-0.06%
$\Delta f_{-50m} + c\Delta t_{-50m}$	-1.83%	$\Delta f_{+50m} + c\Delta t_{+50m}$	-1.70%
-25m position bias			
Δf_{-25m}	-1.68%		
Δt_{-25m}	-0.12%		
$\Delta f_{-25m} + c\Delta t_{-25m}$	-1.79%		

Table 6.9: The results from simulations of the first 40 km of Södertälje toward Norrköping with position bias faults. The standard PI cruise controller is used as reference.

6.3.2 Mass and Wheel Radius Errors

A mass error gives some minor effects on the outcome when this road is simulated. The benefits or losses in time and fuel are slightly reorganized but totally the change is down to a one tenth of a percentage unit. When a false wheel radius is used in the optimization slightly larger differences in fuel consumption and travel time appears but they are still small. DP tool is apparently very good at handling model errors when real roads are used to. Results are presented in Table 6.10.

Södertälje - Norrköping			
Δ -Function	Value	Δ -Function	Value
Optimal scenario $r_w = 52cm$		Wheel Radius, $r_w = 47cm$	
Δf_{opt}	-1.74%	Δf_{r47}	-1.79%
Δt_{opt}	-0.04%	Δt_{r47}	0.15%
$\Delta f_{opt} + c\Delta t_{opt}$	-1.78%	$\Delta f_{r47} + c\Delta t_{r47}$	-1.65%
-10% mass fault		Wheel Radius, $r_w = 57cm$	
$\Delta f_{-10\%}$	-1.79%	Δf_{r57}	-1.57%
$\Delta t_{-10\%}$	0.14%	Δt_{r57}	-0.32%
$\Delta f_{-10\%} + c\Delta t_{-10\%}$	-1.67%	$\Delta f_{r57} + c\Delta t_{r57}$	-1.86%
+10% mass fault			
$\Delta f_{+10\%}$	-1.67%		
$\Delta t_{+10\%}$	-0.25%		
$\Delta f_{+10\%} + c\Delta t_{+10\%}$	-1.89%		

Table 6.10: Results from simulations of Södertälje - Norrköping with mass errors to the left and false wheel radius to the right. The reference is the standard PI cruise controller.

6.3.3 Roll Resistance and Aerodynamic Drag Force Errors

The impact of environmental disturbances like changes in the aerodynamic drag force or the rolling resistance do not have much effect on the system either when running simulations on recorded real world road profiles. It should be remembered that a change in the aerodynamic drag force in the optimization also affects the value of β , see Equation 5.11. The differences in travel time and fuel consumption are presented in Table 6.11.

Södertälje - Norrköping			
Δ -Function	Value	Δ -Function	Value
Optimal scenario		-10% rolling resistance	
Δf_{opt}	-1.74%	$\Delta f_{-10\%}$	-1.91%
Δt_{opt}	-0.04%	$\Delta t_{-10\%}$	0.19%
$\Delta f_{opt} + c\Delta t_{opt}$	-1.78%	$\Delta f_{-10\%} + c\Delta t_{-10\%}$	-1.73%
-10% aerodynamic drag		+10% rolling resistance	
$\Delta f_{-10\%}$	-1.88%	$\Delta f_{+10\%}$	-1.61%
$\Delta t_{-10\%}$	0.16%	$\Delta t_{+10\%}$	-0.25%
$\Delta f_{-10\%} + c\Delta t_{-10\%}$	-1.73%	$\Delta f_{+10\%} + c\Delta t_{+10\%}$	-1.83%
+10% aerodynamic drag			
$\Delta f_{+10\%}$	-1.61%		
$\Delta t_{+10\%}$	-0.21%		
$\Delta f_{+10\%} + c\Delta t_{+10\%}$	-1.79%		

Table 6.11: Difference in fuel consumption and travel time for simulations over Södertälje toward Norrköping with wrongly predicted aerodynamic drag force (left) or rolling resistance (right). The standard PI cruise controller is used as reference.

6.4 Map Quantisation

One of the key elements of this master thesis is to define how good the topographic road map has to be. To lower the amount of memory needed to store the map some sort of quantization of the road gradient has to be done. The system should not change the velocity for angles where it can keep reference velocity. With the engine used in this thesis, a reference velocity of 85km/h and with a truck mass of $40\,000\text{kg}$, angles between approximately -1.5% and 1.5% should be possible to handle without losing or gaining speed due to gravity. Choosing a minimum angle in the quantization too close to these values will result in optimization problems in DP-tool. Border angles of -1% and 1% with a zero level in between did however work quite well. Two different types of quantization will be investigated in this section, an uniform quantization and a non uniform quantization with increasing step sizes and a gap between $\pm 1\%$ but still with a zero level.

One company which currently is looking into the construction of a commercial topographic road map has said that they at the moment are looking at implementing a solution where the road slope angle is represented by five bits. This limits the number of different angles that can be described to $2^5 = 32$, which at first look might seem like a dangerously small number. The results from simulations though are not really what could be expected when a quite "rough" quantization is applied. It should be noted that in applications like

sound and video encoding there is often of interest to calculate the signal to noise ratio (SNR). The SNR could be an interesting measurement of how good the quantized signal actually is but it doesn't work very well for the non uniform quantization used in this thesis due to the noise building up in the gap around zero.

6.4.1 Uniform Quantisation

The first form of quantization which is studied is uniform quantization. A uniform quantization is easy to implement and has the benefit of the same difference in angle between every step. The big drawback is primary that the relative errors for narrow angles will be much larger than for wider angles. This also, if the number of levels are limited to 32, leads to a quite narrow interval of angles if the errors for small angles should be left at a reasonable level. It could be said here that it possibly would not matter if small angles have a quite large fault since the angles of interest are generally over 1.5% for the engine used in this thesis.

It will be noticed in the simulations that the system is not even closely as sensitive as one could expect to quantization with relatively large step sizes. The results speak for themselves, see Figure 6.22, 6.23, 6.24 and Table 6.12. The system handles map data with quantization step sizes of up to 1.2% extremely well in this case which is kind of surprising since with a step size of 1.2% on this road, Södertälje - Norrköping, the system is limited to only seven different angle values, from -3.6% to $+3.6\%$. It should be said though that the results in some simulations start to vary a bit already at a stepsize of 0.8%. The figures shows that when utilizing a step size of 0.4% or 0.8% still the resulting road is very similar to the real road, see Figure 6.22 and 6.23. At a step size of 1.2% the resulting road is differing quite a bit from the real road but still most dynamics is reconstructed by the quantized road and performance is still good, Figure 6.24.

It should be remembered that the system is only looking at the coming 1.5 km section of the road so slowly climbing or falling roads will not affect the performance of the system much. Once a quantization step size of over 1.2% is used system performance becomes unstable leading to random results which can be seen in Table 6.12. Whether a stepsize of 0.8% should be tolerated or not can be discussed, there are deviations from the optimal result but it cannot be concluded that it is performing worse. There might be situations where the result is greatly degraded instead. A stepsize of 0.4% is no problem for the result other than the limited amount of angles that can be covered if a saturation is set. Simulations over the road between Uppsala and Gävle are available in Appendix B.

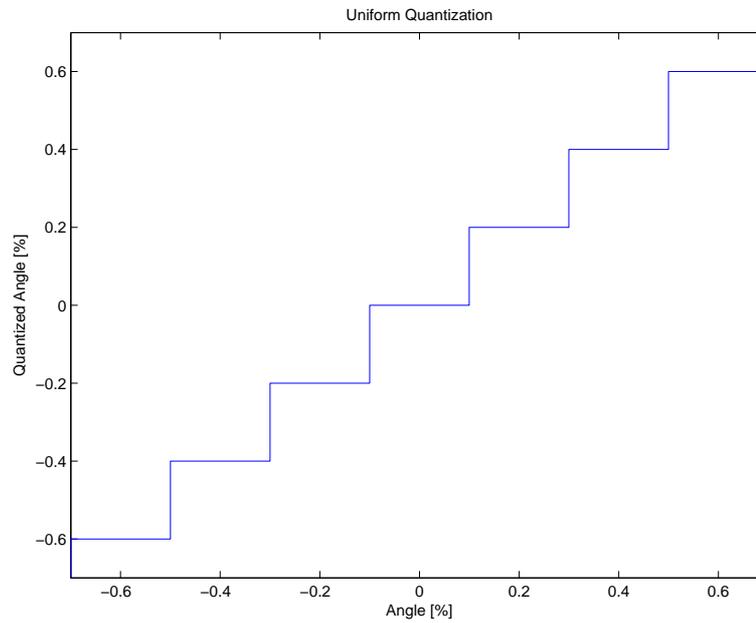


Figure 6.21: Uniform quantization, here with an angle step size of 0.2%.

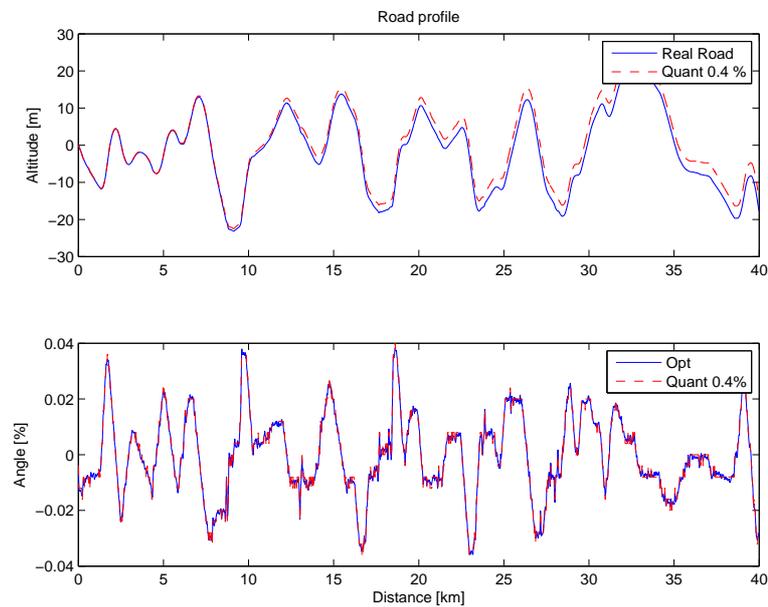


Figure 6.22: Uniform quantization with a step size of 0.4% applied on the first 40 km of the road between Södertälje and Norrköping.

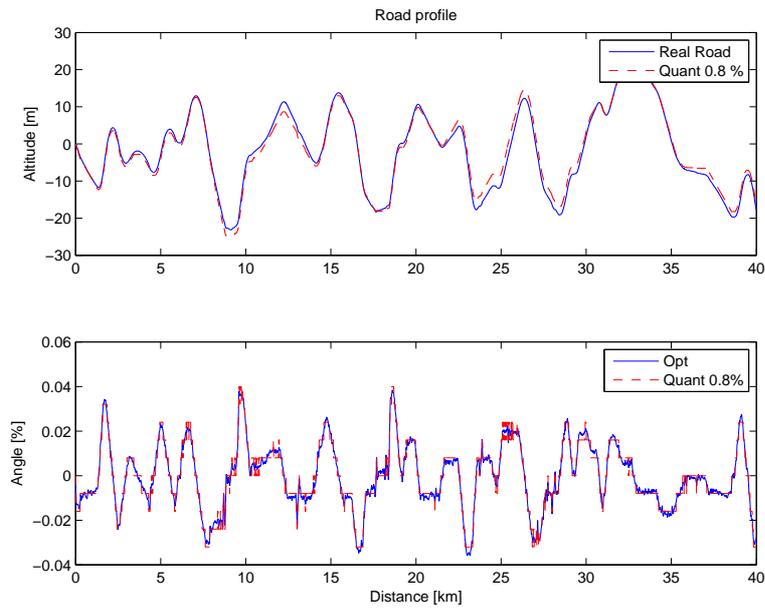


Figure 6.23: Uniform quantization with a step size of 0.8% applied on the first 40 km of the road between Södertälje and Norrköping.

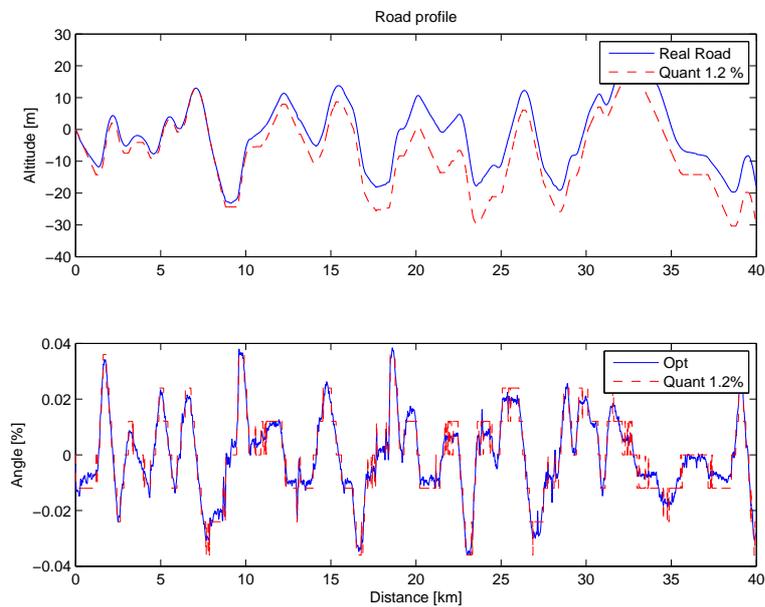


Figure 6.24: Uniform quantization with a step size of 1.2% applied on the first 40 km of the road between Södertälje and Norrköping.

Södertälje - Norrköping

Δ -Function	Value	Δ -Function	Value
Optimal scenario		Stepsize 0.8%	
Δf_{opt}	-1.74%	Δf_{q08}	-1.80%
Δt_{opt}	-0.04%	Δt_{q08}	+0.02%
$\Delta f_{opt} + c\Delta t_{opt}$	-1.78%	$\Delta f_{q08} + c\Delta t_{q08}$	-1.78%
Stepsize 0.2%		Stepsize 1.2%	
Δf_{q02}	-1.77%	Δf_{q12}	-1.84%
Δt_{q02}	-0.01%	Δt_{q12}	+0.02%
$\Delta f_{q02} + c\Delta t_{q02}$	-1.76%	$\Delta f_{q12} + c\Delta t_{q12}$	-1.82%
Stepsize 0.4%		Stepsize 1.6%	
Δf_{q04}	-1.72%	Δf_{q16}	-2.03%
Δt_{q04}	-0.06%	Δt_{q16}	+0.42%
$\Delta f_{q04} + c\Delta t_{q04}$	-1.77%	$\Delta f_{q16} + c\Delta t_{q16}$	-1.65%

Table 6.12: Results from uniform quantization of the angle vector from the road between Södertälje and Norrköping (E4). As reference is the standard PI cruise controller used.

6.4.2 Non Uniform Quantisation

The second type of quantization which is used in this thesis is a type of non uniform quantization with a gap between $\pm 1\%$ but still the zero level is kept, see Figure 6.25. Uniform quantization has the drawback that the precision for small angles is much worse than for larger angles. For example if an uniform quantization stepsize of 0.4% is used, a road slope angle of 1.01% would be rounded off to 1.2% which is almost a 20% error. But for an angle of 2.19%, which will be rounded off to 2.0%, the error will only be close to 10%. Thus the step size is forced to be small enough to handle narrow angles which due to data storage size limitations results in a low saturation of the angle data. For example uniform quantization of a system with a step size of 0.2% and 32 levels will run into saturation at an angle of 3.2% which is not enough for most roads.

Therefore it might be advantageous to look at a non uniform quantization which allows small step sizes for small angles and larger step sizes the larger the angle gets. This will result in better precision for smaller angles but lower precision for larger angles compared to the uniform quantization. The system will also not run into the saturation of the quantization as soon as for the uniform quantization.

The idea with the gap is that since the system will be limited in the number of angle sizes that it can represent, why waste space on angles that are not of interest. For angles where reference speed can be kept it should be kept. A good solution would be to let the vehicle run on standard cruise controller for small angles where the reference speed can be kept. The size of the gap has to be chosen so that all possible vehicle configurations can make use of the road map. Here the minimum levels are chosen as $\pm 1\%$ which is quite a bit lower than what is needed for the chosen vehicle configuration, 420hp and 40 000kg. Most vehicle configurations should be able to handle road slopes of $\pm 1\%$ but it might be needed to close the gap slightly to avoid performance reductions in some cases.

For this to work of course the optimization algorithm has to be able to handle the loss of information for angles between $\pm 1\%$. With a step size resulting

in maximum of a 10% fault, in other words steps increasing with 20% each time, and a gap between $\pm 1\%$ the quantized angle now spans over the interval $[-12.84\% + 15.41\%]$ when 5 bits (32 levels) are used. This should be more than enough for most roads. Possibly a stepsize where the fault is maximum of 20% could be used but with larger steps than that performance is degraded. Using 5 bits, a maximum fault of 20% and a gap between $\pm 1\%$ allows a span of angles over $\pm 100\%$. Since angles between $\pm 12.84\%$ should be more than enough for this system there is no reason to lower the precision more when 32 angle levels are available. It should be remembered that a mass fault also directly will add to the induced fault for the angle from the quantization, therefore keeping this fault low should be a priority.

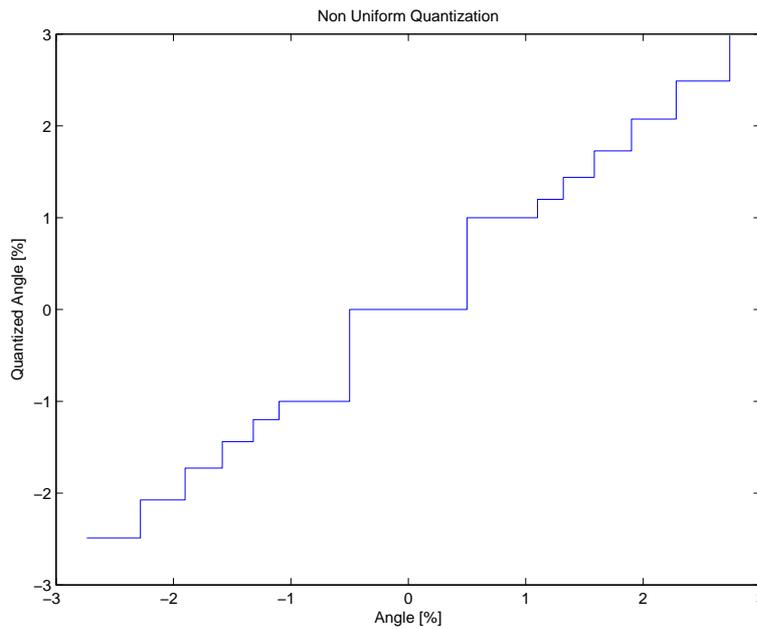


Figure 6.25: Non uniform quantization with gap between $\pm 1\%$ but with a zero level and with a step size increasing 20% from the current step.

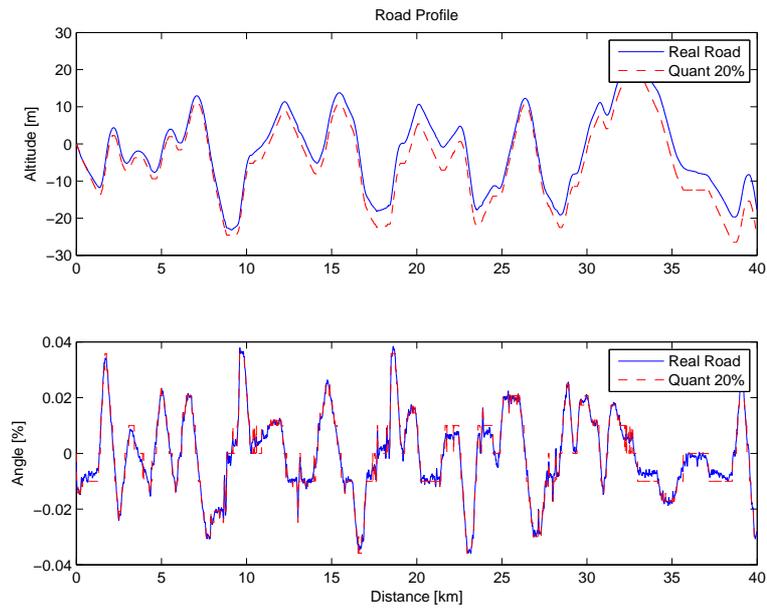


Figure 6.26: Non uniform quantization with an increasing step size of 20% applied on the first 40 km of the road between Södertälje and Norrköping.

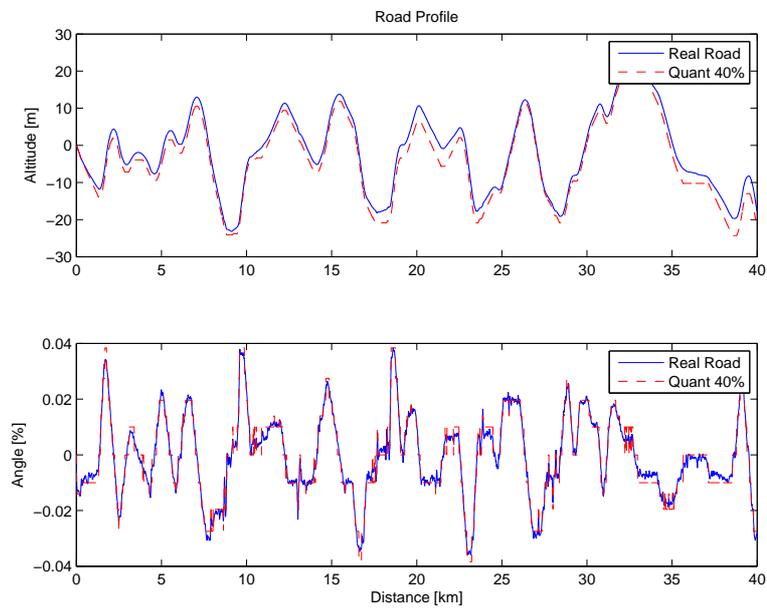


Figure 6.27: Non uniform quantization with an increasing step size of 40% applied on the first 40 km of the road between Södertälje and Norrköping.

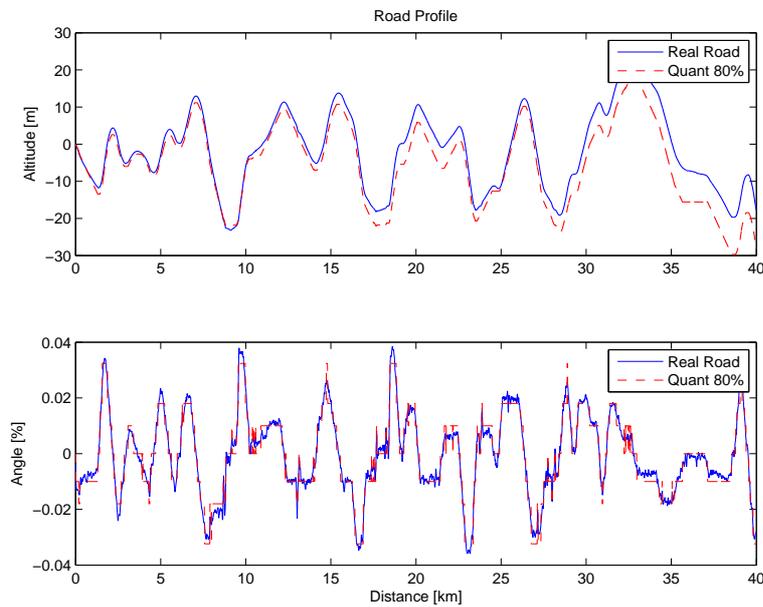


Figure 6.28: Non uniform quantization with an increasing step size of 80% applied on the first 40 km of the road between Södertälje and Norrköping.

Södertälje - Norrköping

Δ -Function	Value	Δ -Function	Value
Optimal scenario		Stepsize 80%	
Δf_{opt}	-1.74%	Δf_{q80}	-1.89%
Δt_{opt}	-0.04%	Δt_{q80}	+0.20%
$\Delta f_{opt} + c\Delta t_{opt}$	-1.78%	$\Delta f_{q80} + c\Delta t_{q80}$	-1.71%
Stepsize 20%		Stepsize 120%	
Δf_{q20}	-1.72%	Δf_{q120}	-1.62%
Δt_{q20}	-0.06%	Δt_{q120}	-0.22%
$\Delta f_{q20} + c\Delta t_{q20}$	-1.78%	$\Delta f_{q120} + c\Delta t_{q120}$	-1.81%
Stepsize 40%		Stepsize 160%	
Δf_{q40}	-1.80%	Δf_{q160}	-1.61%
Δt_{q40}	+0.02%	Δt_{q160}	-0.26%
$\Delta f_{q40} + c\Delta t_{q40}$	-1.78%	$\Delta f_{q160} + c\Delta t_{q160}$	-1.85%

Table 6.13: Results from non uniform quantization of the angle vector from the road Södertälje toward Norrköping (E4). The standard PI cruise controller is used as reference.

Chapter 7

Conclusions

This thesis looks into the sensitivities of two different types of look ahead systems. The first system uses analytical solutions of the optimal speed trajectory for very simple road profiles. The second system uses model predictive control and dynamic programming to be able to calculate the optimal solution and thus gives the opportunity to optimize the velocity over complex road profiles such as real recorded roads.

The results from the simple road profiles looked first upon in the two different implementations are quite clear. With disturbed road data the system either reacts too early or too late, with a loss of fuel and/or time as a result. While both the tested systems still are better than the PI cruise controller in a case with road data with quite large deviations from the real world data a lot of the potential fuel that otherwise could be saved is wasted. On these road profiles position faults of maximum of $25m$ and angle faults of max 10% of current angle seems to be quite realistic tolerances. Especially since there is a mass estimation error that can be upto around 10%. The analytical solution often gives a better result in non disturbed cases but the model predictive control with dynamic programming handles disturbed cases better. It should also be kept in mind that dynamic programming is a numerical approach which sometimes does not result in a perfect solution compared to an analytical solution. Also the MPC has the benefit of a built in feedback in the system.

However when the system using MPC and DP is run over recorded real roads the disturbances do not have nearly the same influence on the result as was the case with the simple road profiles. The system turns out to be extremely tolerant toward defect input data. For example quantization of the map data where a fault of up to 40% in the angle data is tolerated hardly has any effect on the outcome. Also a position fault of $50m$ seems tolerable which stands in contrast to the results from the simple road profiles. There are most likely several parts to why this is the case. First of all real roads are more forgiving since they will not change from level road to a 3% down or uphill slope in a single sample. Second there is a feedback in the system allowing for some correction of model errors. DP-tool ramps the driver demand up and down which works well together with the feedback in the system to correct model errors.

There should not be any problems to keep the position error under $50m$ and the smaller fault the better and to be realistic it should not be a problem to keep it under $25m$ either. A modern GPS gives a position error of under $5m$, 95% of the time and with a modern integrated computer and the right sampling requirements on the road map, giving a maximum position error at

around $\pm 20m$, this should be possible. Map quantization should be done so the error stays under 20% for the angle and it is definitely preferred to use a non uniform quantization. A non uniform quantization will either save data space and therefore cost of the system or allow better precision for most angles than the uniform quantization. With a non uniform quantization of the kind described in this thesis it is possible to keep the angle fault under 10% and still keep the span of angles between $[-12.84 + 15.41]$ % which is more than enough. In an implementation it most likely would be better to close the gap slightly and use a smaller stepsize to increase precision. Generally angles over around 8% are of less interest since there are very few road slopes steeper than that.

Chapter 8

Extensions and Future Works

Still there are no topographic road maps available which will be needed if a system like this is going to be constructed. While the dynamic programming and model predictive control approach certainly works well it, at least in the current implementation, requires too much computational power to be implemented on board. In his licentiate thesis (2007) Erik Hellström has shown that it is possible to implement a dynamic programming algorithm that is $N \log_2 N$ dependent instead of the current N^2 dependant implementation. This opens for new possibilities for the use of dynamic programming on board especially if a dedicated unit can be added for this use. In time dynamic programming might be a good choice but for use in trucks of today a simpler optimization method is needed. When an on board implementation is constructed new simulations and tests have to be done to investigate its robustness toward degraded input data. Also tests of the system on board with scrambled input data especially with a quantized map have to be done to ensure the results are in line with simulations of the system. Quantization of the road gradient might not be the only possible solution instead a library of different road sections could be built up and then combined together to form a complete topographic road profile. The next step for look ahead control might very well be to implement it in a hybrid driveline which should allow for very good control of when to use the electrical engine and when it is advantageous to charge the energy storage.

References

- [1] Leif Appelgren, Anders Holvid, and Lars Erik Zachrisson. *Optimeringsmetoder*. Studentlitteratur, 1972.
- [2] Dimitri P. Bertsekas. *Dynamic Programming and Optimal Control*, volume 1. Athena Scientific, 2 edition, 2000.
- [3] Anders Fröberg, Erik Hellström, and Lars Nielsen. Explicit fuel optimal speed profiles for heavy trucks on a set of topographic road profiles. Number 2006-01-1071 in SAE World Congress, 2006.
- [4] Anders Fröberg and Lars Nielsen. Optimal fuel and gear ratio control for heavy trucks with piece wise affine engine characteristics. 2007.
- [5] Torkel Glad and Lennart Ljung. *Reglerteori: Flervariabla och olinjära metoder*. Studentlitteratur, 2 edition, 2003.
- [6] Erik Hellström. Explicit use of road topography for model predictive cruise control in heavy trucks. Master's thesis LiTH-ISY-EX-05/3660-SE, Vehicular Systems, Department of Electrical Engineering, Linköpings Universitet, Linköping, Sweden, 2005.
- [7] Erik Hellström. *Look-ahead Control of Heavy Trucks utilizing Road Topography*. Licentiate thesis 1319, Vehicular Systems, Department of Electrical Engineering, Linköping University, 2007.
- [8] Erik Hellström, Maria Ivarsson, Jan Åslund, and Lars Nielsen. Look-ahead control for heavy trucks to minimize trip time and fuel consumption. 2007.
- [9] Uwe Kiencke and Lars Nielsen. *Automotive Control Systems, For Engine, Driveline and Vehicle*. Springer-Verlag, 2 edition, 2005.
- [10] Ermin Kozica. Look ahead cruise control: Road slope estimation and control sensitivity. Master's thesis IR-RT-EX-0524, Department of Signals, Sensors and Systems, Kungliga Tekniska Högskolan (KTH), 2005.
- [11] Lars Nielsen and Lars Eriksson. *Vehicular Systems*. Vehicular Systems, ISY Linköping Institute of Technology, 2005.

Appendix A

Model Parameters

A.1 Simulink Model

In Figure A.1 the Simulink model used in the first part of this thesis is shown.

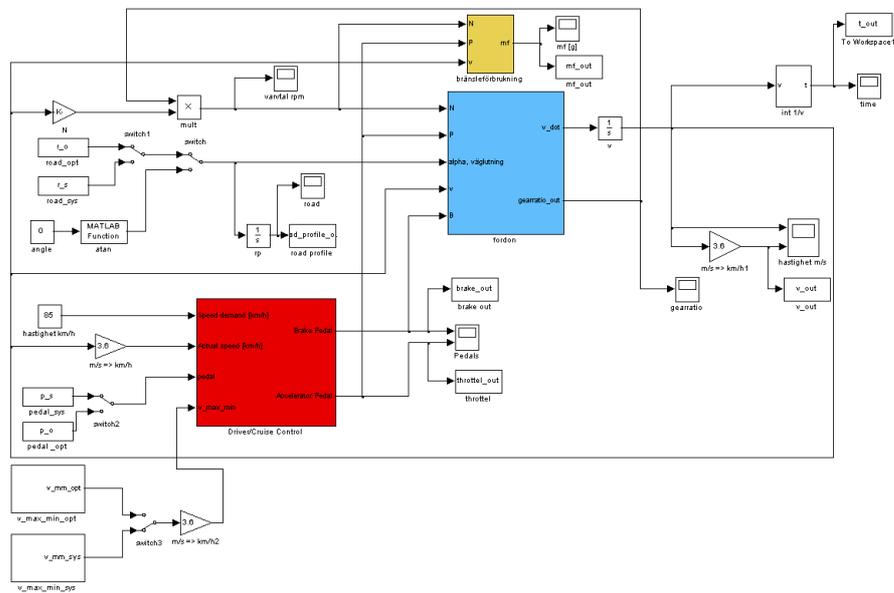


Figure A.1: Simulink model used in the first simulations, see Chapter 3 and 4.

A.2 Shared Parameters

Model Parameters

Parameter	Value	Unit
Engine Parameters		
Number of Cylinders n_{cyl}	6	—
Number of Revolutions per stroke n_r	2	—
Engine Inertia J_e	3.5	kgm^2
Idle Engine Speed N_{idle}	600	rpm
Idle Fueling δ_{idle}	10.83	$\frac{mg}{stroke}$
Engine Map		
a_e	-0.1135	$\frac{Nm}{rpm}$
b_e	9.4263	$\frac{Nm \cdot stroke}{mg}$
c_e	5.6282	Nm
Maximum Fuel Flow		
a_δ	$-1.429 \cdot 10^{-4}$	$\frac{mg}{stroke \cdot rpm^2}$
b_δ	0.3973	$\frac{mg}{stroke \cdot rpm}$
c_δ	-48.5649	$\frac{mg}{stroke}$
Drag Torque		
a_d	-0.0917	$\frac{Nm}{rpm}$
b_d	-46.0140	Nm
Forces		
Gravity g	9.81	$\frac{m}{s^2}$
Rolling Resistance Coefficient c_r	$7 \cdot 10^{-3}$	—
Air Drag Coefficient c_w	0.6	—
Air Density ρ_a	1.29	$\frac{kg}{m^3}$
Cross-section Area A_a	10	m^2
Wheel Inertia J_w	32.9	kgm^2
Wheel Radius r_w	0.52	m
Final Drive		
Ratio i_f	3.27	—
Efficiency η_f	0.97	—
Brake		
Brake Constant k_b	$20 \cdot 10^3$	—

Table A.1: Model Parameters

A.3 Cruise Controller

The cruise controller used in this thesis is a standard PI regulator. Brake usage is controlled by a second PI controller.

The cruise controller constants are:

$$K_p = 2.0$$

$$K_i = 0.05$$

The brake controller constants are:

$$K_p = 5.0$$

$$K_i = 0.02$$

To avoid integrator windup saturation levels are set for both PI controllers. The cruise integrator part is allowed to take values between 0 and 30 and the brake integrator part is allowed to take values between 0 and 35.

A.4 Gearbox

The gearboxes differs slightly in the two models used in this thesis due to different ways of implementation.

Gear Ratios

Gear	Ratio	Efficiency	Gear	Ratio	Efficiency
1	11.27	0.93	7	3.01	0.96
2	9.14	0.93	8	2.44	0.96
3	7.17	0.94	9	1.91	0.96
4	5.81	0.95	10	1.55	0.96
5	4.62	0.95	11	1.23	0.96
6	3.75	0.95	12	1.00	0.97

Table A.2: Gear ratios and efficiencies for a GRS 900 gearbox, the efficiency values stated here are only used in DP-tool.

A.4.1 Gearbox Model 1

The gearbox used by the simulation environment in chapter 3 and 4. The efficiency is assumed to be $\eta_g = 0.97$ for all gears. The gearbox is only able to shift the gear one step at a time but since all simulations but one with this gearbox model only uses gear number 12 this is adequate for the purpose of this thesis. Changing gear will drop the from the engine generated torque after the gearbox to zero for one second.

Gear shift points model 1

Gear Number	1 – 2	2 – 3	6 – 7	7 – 8	11 – 12
Shift Up Engine Speed	1400	1400	1550	1550	1650
Shift Down Engine Speed	1150	1135	1120	1110	1100

Table A.3: Gear shift points for a GRS 900 gearbox, down shift points modified for maximum load. For gear shifts that are not listed linear interpolation between listed points is applied.

A.4.2 Gearbox model in DP-tool

The gearbox used in DP-tool is a slightly better model.

Gear shift points (Engine Speed)

Gear	Up [rpm]	Down [rpm]	Gear	Up [rpm]	Down [rpm]
1 – 2	1500	950	7 – 8	1497	1006
2 – 3	1501	960	8 – 9	1489	1012
3 – 4	1502	970	9 – 10	1481	1018
4 – 5	1503	980	10 – 11	1473	1024
5 – 6	1504	990	11 – 12	1465	1030
6 – 7	1505	1000			

Table A.4: Gear shift points used in DP-tool

Modifier at Max and Min Load

Load	Up [rpm]	Down [rpm]
Min	-70	-100
Max	150	175

Table A.5: Gear shift points used in DP-tool

Appendix B

More Results

B.1 More Results from DP-tool

This section contains more results from simulations with DP-tool. The roads simulated are Uppsala toward Gävle and in the opposite direction, this road is plotted in Figures B.1. Also some simulations from Norrköping toward Södertälje are presented in the tables in this section. The reference run was done with the standard PI cruise controller.

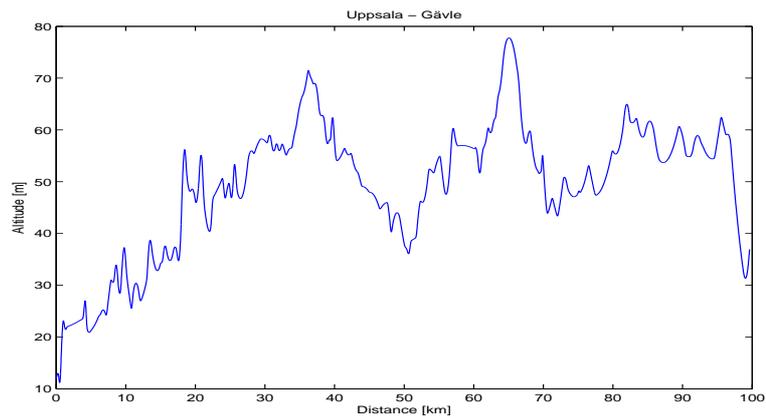


Figure B.1: The road between Uppsala and Gävle.

Södertälje - Norrköping		Norrköping - Södertälje	
Δ -Function	Value	Δ -Function	Value
Optimal scenario		Optimal scenario	
Δf_{opt}	-1.74%	Δf_{opt}	-2.08%
Δt_{opt}	-0.04%	Δt_{opt}	+0.39%
$\Delta f_{opt} + c\Delta t_{opt}$	-1.78%	$\Delta f_{opt} + c\Delta t_{opt}$	-1.72%
-50m position bias		-50m position bias	
Δf_{-50m}	-1.72%	Δf_{-50m}	-2.18%
Δt_{-50m}	-0.12%	Δt_{-50m}	+0.50%
$\Delta f_{-50m} + c\Delta t_{-50m}$	-1.83%	$\Delta f_{-50m} + c\Delta t_{-50m}$	-1.73%
+50m position bias		+50m position bias	
Δf_{+50m}	-1.65%	Δf_{+50m}	-1.86%
Δt_{+50m}	-0.06%	Δt_{+50m}	+0.30%
$\Delta f_{+50m} + c\Delta t_{+50m}$	-1.70%	$\Delta f_{+50m} + c\Delta t_{+50m}$	-1.59%

Table B.1: Results from the road between Södertälje and Norrköping in different directions with position bias faults.

Uppsala - Gävle		Gävle - Uppsala	
Δ -Function	Value	Δ -Function	Value
Optimal scenario		Optimal scenario	
Δf_{opt}	-0.05%	Δf_{opt}	-0.48%
Δt_{opt}	+0.21%	Δt_{opt}	+0.25%
$\Delta f_{opt} + c\Delta t_{opt}$	+0.15%	$\Delta f_{opt} + c\Delta t_{opt}$	-0.26%
-50m position bias		-50m position bias	
Δf_{-50m}	-0.07%	Δf_{-50m}	-0.56%
Δt_{-50m}	+0.19%	Δt_{-50m}	+0.29%
$\Delta f_{-50m} + c\Delta t_{-50m}$	+0.10%	$\Delta f_{-50m} + c\Delta t_{-50m}$	-0.30%
+50m position bias		+50m position bias	
Δf_{+50m}	-0.06%	Δf_{+50m}	-0.44%
Δt_{+50m}	+0.22%	Δt_{+50m}	+0.25%
$\Delta f_{+50m} + c\Delta t_{+50m}$	+0.13%	$\Delta f_{+50m} + c\Delta t_{+50m}$	-0.21%

Table B.2: Results from simulations of the between Uppsala and Gävle in different directions with position bias faults.

Södertälje - Norrköping		Norrköping - Södertälje	
Δ -Function	Value	Δ -Function	Value
Optimal scenario		Optimal scenario	
Δf_{opt}	-1.74%	Δf_{opt}	-2.08%
Δt_{opt}	-0.04%	Δt_{opt}	+0.39%
$\Delta f_{opt} + c\Delta t_{opt}$	-1.78%	$\Delta f_{opt} + c\Delta t_{opt}$	-1.72%
-10% mass fault		-10% mass fault	
$\Delta f_{-10\%}$	-1.79%	$\Delta f_{-10\%}$	-2.15%
$\Delta t_{-10\%}$	+0.14%	$\Delta t_{-10\%}$	+0.50%
$\Delta f_{-10\%} + c\Delta t_{-10\%}$	-1.67%	$\Delta f_{-10\%} + c\Delta t_{-10\%}$	-1.70%
+10% mass fault		+10% mass fault	
$\Delta f_{+10\%}$	-1.67%	$\Delta f_{+10\%}$	-1.98%
$\Delta t_{+10\%}$	-0.25%	$\Delta t_{+10\%}$	+0.26%
$\Delta f_{+10\%} + c\Delta t_{+10\%}$	-1.89%	$\Delta f_{+10\%} + c\Delta t_{+10\%}$	-1.75%

Table B.3: Simulations with mass faults on the road between Södertälje and Norrköping in both directions.

Uppsala - Gävle		Gävle - Uppsala	
Δ -Function	Value	Δ -Function	Value
Optimal scenario		Optimal scenario	
Δf_{opt}	-0.05%	Δf_{opt}	-0.48%
Δt_{opt}	+0.21%	Δt_{opt}	+0.25%
$\Delta f_{opt} + c\Delta t_{opt}$	+0.15%	$\Delta f_{opt} + c\Delta t_{opt}$	-0.26%
-10% mass fault		-10% mass fault	
$\Delta f_{-10\%}$	-0.27%	$\Delta f_{-10\%}$	-0.54%
$\Delta t_{-10\%}$	+0.20%	$\Delta t_{-10\%}$	+0.29%
$\Delta f_{-10\%} + c\Delta t_{-10\%}$	-0.09%	$\Delta f_{-10\%} + c\Delta t_{-10\%}$	-0.28%
+10% mass fault		+10% mass fault	
$\Delta f_{+10\%}$	+0.09%	$\Delta f_{+10\%}$	-0.46%
$\Delta t_{+10\%}$	+0.11%	$\Delta t_{+10\%}$	+0.25%
$\Delta f_{+10\%} + c\Delta t_{+10\%}$	+0.19%	$\Delta f_{+10\%} + c\Delta t_{+10\%}$	-0.24%

Table B.4: Simulations with mass faults on the road between Uppsala and Gävle in both directions.

Uppsala - Gävle

Δ -Function	Value	Δ -Function	Value
Optimal scenario		Stepsize 0.8%	
Δf_{opt}	-0.05%	Δf_{q08}	-0.28%
Δt_{opt}	+0.21%	Δt_{q08}	+0.24%
$\Delta f_{opt} + c\Delta t_{opt}$	+0.15%	$\Delta f_{q08} + c\Delta t_{q08}$	-0.06%
Stepsize 0.2%		Stepsize 1.2%	
Δf_{q02}	-0.15%	Δf_{q12}	-0.10%
Δt_{q02}	+0.23%	Δt_{q12}	+0.34%
$\Delta f_{q02} + c\Delta t_{q02}$	+0.05%	$\Delta f_{q12} + c\Delta t_{q12}$	+0.21%
Stepsize 0.4%		Stepsize 1.6%	
Δf_{q04}	+0.08%	Δf_{q16}	-0.49%
Δt_{q04}	+0.18%	Δt_{q16}	+0.62%
$\Delta f_{q04} + c\Delta t_{q04}$	+0.25%	$\Delta f_{q16} + c\Delta t_{q16}$	+0.07%

Table B.5: Results from uniform quantization of the angle vector from the road between Uppsala and Gävle (E4).

Uppsala - Gävle

Δ -Function	Value	Δ -Function	Value
Optimal scenario		Stepsize 80%	
Δf_{opt}	-0.05%	Δf_{q80}	-0.21%
Δt_{opt}	+0.21%	Δt_{q80}	+0.28%
$\Delta f_{opt} + c\Delta t_{opt}$	+0.15%	$\Delta f_{q80} + c\Delta t_{q80}$	-0.05%
Stepsize 20%		Stepsize 120%	
Δf_{q20}	-0.28%	Δf_{q120}	+0.07%
Δt_{q20}	+0.27%	Δt_{q120}	+0.08%
$\Delta f_{q20} + c\Delta t_{q20}$	-0.03%	$\Delta f_{q120} + c\Delta t_{q120}$	+0.15%
Stepsize 40%		Stepsize 160%	
Δf_{q40}	-0.03%	Δf_{q160}	+0.15%
Δt_{q40}	+0.30%	Δt_{q160}	+0.03%
$\Delta f_{q40} + c\Delta t_{q40}$	+0.24%	$\Delta f_{q160} + c\Delta t_{q160}$	+0.18%

Table B.6: Results from non uniform quantization of the angle vector from the road Uppsala toward Gävle (E4).

Appendix C

DP-Tool GUI

The original graphical user interface (GUI) for DP-tool was designed by Erik Hellström. To be able to control disturbances the GUI was modified. The original GUI allows the user to set the look ahead horizon, penalty parameters, reference speed and vehicle parameters. The modified GUI also allows the user to control position bias faults, mass estimation errors, angular bias faults and add low pass filtered white noise to position and altitude data. Another possibility for the user is to load an already disturbed road profile and run it against the real one. The position bias fault slider is controlling the position bias fault in meters. The mass fault is given in percent of the current mass and also the angle bias is given in percent of the current angle, meaning that a +20% bias fault on a 3% slope equals an angle of 3.6%. Quantization is given in road slope angle percent and the control variable for white noise is the variance.

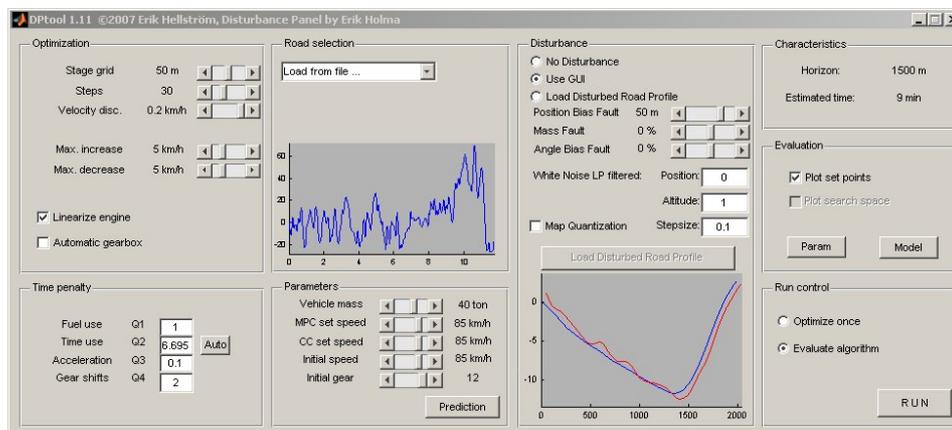


Figure C.1: The DP-Tool GUI

Copyright

Svenska

Detta dokument hålls tillgängligt på Internet - eller dess framtida ersättare - under en längre tid från publiceringsdatum under förutsättning att inga extra-ordinära omständigheter uppstår.

Tillgång till dokumentet innebär tillstånd för var och en att läsa, ladda ner, skriva ut enstaka kopior för enskilt bruk och att använda det oförändrat för ickekommersiell forskning och för undervisning. Överföring av upphovsrätten vid en senare tidpunkt kan inte upphäva detta tillstånd. All annan användning av dokumentet kräver upphovsmannens medgivande. För att garantera äktheten, säkerheten och tillgängligheten finns det lösningar av teknisk och administrativ art.

Upphovsmannens ideella rätt innefattar rätt att bli nämnd som upphovsman i den omfattning som god sed kräver vid användning av dokumentet på ovan beskrivna sätt samt skydd mot att dokumentet ändras eller presenteras i sådan form eller i sådant sammanhang som är kränkande för upphovsmannens litterära eller konstnärliga anseende eller egenart.

För ytterligare information om Linköping University Electronic Press se förlagets hemsida: <http://www.ep.liu.se/>

English

The publishers will keep this document online on the Internet - or its possible replacement - for a considerable time from the date of publication barring exceptional circumstances.

The online availability of the document implies a permanent permission for anyone to read, to download, to print out single copies for your own use and to use it unchanged for any non-commercial research and educational purpose. Subsequent transfers of copyright cannot revoke this permission. All other uses of the document are conditional on the consent of the copyright owner. The publisher has taken technical and administrative measures to assure authenticity, security and accessibility.

According to intellectual property law the author has the right to be mentioned when his/her work is accessed as described above and to be protected against infringement.

For additional information about the Linköping University Electronic Press and its procedures for publication and for assurance of document integrity, please refer to its WWW home page: <http://www.ep.liu.se/>

© Erik Holma
Linköping, September 23,
2007