ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

ENERGY EFFICIENT VELOCITY TRAJECTORY OPTIMIZATION USING DYNAMIC PROGRAMMING FOR ELECTRIC VEHICLES

M.Sc. THESIS

Abdullah KIZIL

Department of Mechatronics Engineering

Mechatronics Engineering Programme

OCTOBER 2021

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

ENERGY EFFICIENT VELOCITY TRAJECTORY OPTIMIZATION USING DYNAMIC PROGRAMMING FOR ELECTRIC VEHICLES

M.Sc. THESIS

Abdullah KIZIL
518161030

Department of Mechatronics Engineering

Mechatronics Engineering Programme

Thesis Advisor: Assoc. Prof. Dr. Volkan SEZER

OCTOBER 2021

ISTANBUL TEKNİK ÜNİVERSİTESİ ★ LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

ELEKTRİKLİ ARAÇLAR İÇİN DİNAMİK PROGRAMLAMA KULLANILARAK ENERJİ VERİMLİ HIZ YÖRÜNGE OPTİMİZASYONU

YÜKSEK LİSANS TEZİ

Abdullah KIZIL
518161030

Mekatronik Mühendisliği Anabilim Dalı

Mekatronik Mühendisliği Programı

Tez Danışmanı: Doç. Dr. Volkan SEZER

EKİM 2021

Abdullah Kızıl, a M.Sc. student of İTU Graduate School student ID 518161030, successfully defended the thesis entitled "ENERGY EFFICIENT VELOCITY TRAJECTORY OPTIMIZATION USING DYNAMIC PROGRAMMING FOR ELECTRIC VEHICLES", which he prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor : **Assoc. Prof. Dr. Volkan SEZER** ....
İstanbul Technical University

Jury Members : **Assoc. Prof. Dr. Tufan KUMBASAR** ....
Istanbul Technical University

Assoc. Prof. Dr. M. Selçuk ARSLAN ....
Yildiz Technical University

Date of Submission : 02 October 2021
Date of Defense : 22 October 2021

To my spouse and family,

FOREWORD

I am so grateful to work with Assoc.Prof.Dr. Volkan SEZER and thank you to support me, also I would like to thank Oğuz ERÇAKIR, who support me during my thesis.

I would like to thank my family and my wife Nefise Begüm KIZIL for all their support in this journey.

October 2021 Abdullah KIZIL
(Mechatronics Engineer)

TABLE OF CONTENTS

Page

ABBREVIATIONS

ACC	**:** Adaptive Cruise Control
ADAS	**:** Advanced Driver Assistance System
BEV	**:** Battery Electric Vehicle
DP	**:** Dynamic Programming
GLOSA	**:** Green Light Optimal Speed Advisory
GNSS	**:** Global Navigation Satellite System
GPS	**:** Global Positioning System
HMI	**:** Human Machine Interface
MiL	**:** Model in the Loop
MPC	**:** Model Predictive Control
NLP	**:** Nonlinear Programming
OSM	**:** OpenStreetMap
PCC	**:** Predictive Cruise Control
PMP	**:** Pontrygin’s Minimum Principle
QP	**:** Quadratic Programming
SoC	**:** State of Charge
SOCP	**:** Second Order Cone Program
V2V	**:** Vehicle to Vehicle
VCU	**:** Vehicle Control Unit

SYMBOLS

a	: Acceleration
$\bar{a}$	: Average acceleration
A_{front}	: Vehicle’s cross-frame area
c_d	: Aerodynamic resistance coefficient
E	: Energy consumption
f_{iner}	: Inertia equivalent factor vehicle mass
f_r	: Road friction coefficient
g	: Gravity of earth
I_R	: Total rotational inertia of the rotating components
J	: “Cost-to-go” of transition
J^*	: The optimal value of “Cost-to-go”
M	: Number of nodes with speed as the state variable
m_{veh}	: Vehicle mass with payload
N	: Number of nodes for distance as the stage variable
P_{loss}	: Total power losses of powertrain components
s	: Distance
t	: Time
t^*	: Time
v	: Vehicle speed
v^*	: Optimal vehicle speed
v_{max}	: The maximum vehicle speed limit
v_{min}	: The minimum vehicle speed limit
α	: Road slope
ρ_{air}	: Density of air
β	: Weight factor for energy consumption and travel time in the cost function
γ	: Weight factor for comfort in the cost function

LIST OF TABLES

Page

LIST OF FIGURES

Page

ENERGY-EFFICIENT VELOCITY TRAJECTORY OPTIMIZATION USING DYNAMIC PROGRAMMING FOR ELECTRIC VEHICLES

SUMMARY

The electrification and autonomous systems developed in the automotive industry in the last decade bring different solutions. Many methods have been developed and still continue to be developed to reduce energy consumption in vehicles, especially with electrified, connected vehicle technologies and navigation systems. Speed trajectory optimization is part of these methods.

The main motivation of speed trajectory optimization is to prevent excessive energy consumption due to driver driving style. In order to prevent this, information such as the slope and speed limit of the road to be traveled is used over the navigation system.

When we consider only energy while optimizing the speed trajectory, the prolongation of the driving time will appear as a concern. Because if the vehicle goes faster, the energy consumed will increase quadratically. Therefore, optimization will always demand the vehicle to go slower in order to consume less energy and there must be a balance between energy and travel time.

In this thesis, a study has been carried out that periodically updates the speed trajectory, which will ensure that the destination point and arrival time information are provided into the navigation system by the driver while consuming the least energy in the given time.

Dynamic Programming (DP) method is used to solve this problem. Dynamic programming always presents the global optimum behavior under the given boundary conditions. The speed of the vehicle was used as the only state variable and its optimization was performed separately over the distance stages.

The average speed required to reach the destination on time, based on the destination point and travel time information obtained from the navigation system, is given as an input to the optimization, and the DP state space is constantly updated. The main reason for this is to reduce the memory load required by DP. Thus, a fixed number of states are scanned. But the scanned range values are updated according to this speed input.

A longitudinal vehicle model was used for optimization. The limits of the powertrain are also part of the optimization as a boundary condition. Before the optimization is run, a pre-calculation is also made to include the states where the transition between states is possible only in the optimization. Thus, it is aimed to shorten the calculation time by not including the unreachable situations in the optimization.

Optimization takes place along a certain horizon. The speed trajectory calculated for this horizon is transmitted to the vehicle speed control unit as an input. The vehicle follows this speed profile. The optimization is updated again after a certain period of time and transmits the speed trajectory calculated for the next horizon to the vehicle. The purpose of this is if the vehicle cannot follow the given speed for any reason during

real driving, the optimization is performed again based on the new conditions. This allows the vehicle to progress in real-time using the speed trajectory closest to the global optimum.

In the study, simulation and analysis of the all-electric truck were carried out on two different slope routes. Tests were performed with different fixed velocity values and velocity profiles produced by velocity trajectory optimization in both routes. As a result of the simulations carried out, it has been observed that up to 4% of energy consumption and up to 2.5% of the targeted time are saved. Thanks to the proposed adaptive weight factor, it has been observed that the time-energy balance is maintained for different routes, arrival times, and vehicle parameters.

ELEKTRİKLİ ARAÇLAR İÇİN DİNAMİK PROGRAMLAMA KULLANILARAK ENERJİ VERİMLİ HIZ YÖRÜNGE OPTİMİZASYONU

ÖZET

Son 10 yılda otomotiv endüstrisindeki geliştirilen elektrifikasyon ve otonom sistemleri farklı çözümleri de beraberinde getirmektedir. Özellikle bağlantılı araç teknolojileri ve navigasyon sistemleriyle beraber araçlardaki enerji tüketimini azaltmaya yönelik bir çok metot geliştirilmiş ve halende geliştirilmeye devam edilmektedir. Hız yörünge optimizasyonu da bunlardan bir tanesidir.

Hız yörünge optimizasyonu, optimal bir kontrol problemi olarak formüle edilebilir ve çözümü elde etmek için farklı yöntemler vardır. Genellikle, bu yöntemler üç farklı gruba ayrılır: dolaylı yöntemler, doğrudan yöntemler ve dinamik programlama (DP).

İlk grup, dolaylı yöntemler, Pontryagin'in Minimum İlkesini (PMP) takip eder. Bu yöntemler, problemi çok noktalı Hamiltonian sınır değer problemine dönüştürerek ve sayısal olarak çözerek problemi "dolaylı olarak" çözer. Bu konuyla ilgili yapılan ilk çalışmada, içten yanmalı motorun yakıt akış hızının motorun hız ve gücünün fonksiyonu olarak yaklaşık bir polinom denklemi ile sabit yol eğimi için doğrusal olmayan şekilde modellenmiş ve bir aracın yakıt tüketimini en aza indirme problemini çözmek için PMP'i kullanılmıştır.

Sorunu çözmek için başka bir yöntem grubu ise, doğrudan yöntemler olarak adlandırılır. Bu yöntemlerle, durumlar ve kontrol değişkenleri, optimizasyon problemini bir Doğrusal Olmayan Programlama (NLP) problemine yaklaştıracak şekilde parametreleştirilir. Spesifik formlarda tanımlanan maliyet fonksiyonu ve sınır koşulları ile birlikte, problemler çok çeşitli son teknoloji NLP çözücülerle "doğrudan" çözülebilir. Örnek olarak, İkinci Derece Koni Programı (SOCP) ve Kuadratik Programlama (QP), araç hızı ve akü enerjisi durum değişkenleri olarak tanımlanarak hibrit araçlar için kestirimci enerji yönetiminde kullanılmıştır. Optimal durum yörüngeleri, yaklaşımlar ve buluşsal yöntemler kullanarak dışbükey olarak yeniden formüle ettikten sonra, bir SOCP çözücü tarafından daha hızlı hesaplanacak bir şekilde üretilir. Bundan sonra, sorun bir QP çözücünün gereksinimine uyacak şekilde yeniden formüle edilir. Belirli bir test senaryosu için gerçekleştirilen çalışmalar, iki farklı çözücüden de neredeyse aynı sonuçları verir. Elektrikli araçlar için ise QP kullanılarak Model Öngörülü Kontrol (MPC) tabanlı bir Yeşil Işık Optimum Hız Tavsiye (GLOSA) işlevi de geliştirilmiştir. Burada üst düzey bir işlev, birden fazla trafik ışığını durmadan geçebilen bir referans araç hızı yörüngesini hesaplarken, daha düşük düzeydeki bir MPC denetleyicisi, bir enerji optimal araç hızı yörüngesini hesaplar. Problemi dışbükey ve basit bir formda tutmak için maliyet fonksiyonunda sadece araç ivmesi ve referans yörüngeden hız sapması dikkate alınmaktadır.

DP, problemi Bellman'ın Optimallik İlkesini kullanarak özyinelemeli olarak çözülebilen alt problemlere böler. Araç hızı yörünge optimizasyonu ile ilgili gözden geçirilen literatürler arasında DP, optimizasyon problemini çözmek için çok yaygın olarak benimsenen bir yöntemdir. Farklı çalışmalarda aşama değişkeni olarak zaman

kullanıldığında ileri özyinelemeli DP’yi veya mesafe aşama değişkeni olarak alındığında da yolculuk süresinin de maliyet fonksiyonun dahil edildiği yaklaşımlarda görülmektedir. Gerçek zamanlı uygulamalar için, arama alanını azaltmak, hesaplama maliyetini azaltmak için de etkili bir yaklaşımdır. Bu yaklaşımla kısıtlamaları karşılamayan durumlar ortadan kaldırılarak hesaplama maliyeti düşürülür. Yaygın olarak uygulanan ayrık durum uzayına kıyasla, sürekli durum kullanan DP'de önerilen çalışmalar vardır. Bu çalışmalarda, durum değişkeni kaba bir aralıkla birkaç "kutuya" ayrılmıştır. "Kutular" içinde, durum değişkeni daha sonra sezgisel yaklaşım kullanılarak yerel ve sürekli olarak optimize edilir. Bu çalışmalara göre, bu yaklaşım hesaplama maliyetini düşürür ve aynı zamanda ayrık DP tarafından sıklıkla karşılaşılan enterpolasyon problemlerini çözer.

Yinelemeli DP ise, bir uyarlamalı arama uzayında optimal sonuçları yinelemeli olarak yakınsayarak sorunu çözmeye çalışan diğer bir DP yaklaşımıdır. Bu yaklaşımda, sonraki yinelemeler, hesaplama maliyetini düşük tutarken doğruluğu artırmak için daha ince bir ızgaralı durum alanı oluşturmak için önceki yinelemelerin sonucunu kullanır. Bu yaklaşımın, teorik olarak hesaplama çabasını önemli ölçüde azaltabilen, önceki optimizasyon adımından elde edilen tarihsel hesaplanmış maliyetin yeniden kullanılmasını önermektedir.

Bu farklı yöntemler karşılaştırıldığında, dolaylı ve doğrudan yöntemler, dinamik programlama yöntemine kıyasla, özellikle daha fazla sayıda durum değişkeni olan problemleri çözerken, daha kısa hesaplama süresi avantajına sahiptir. Bununla birlikte, verimli çözücüler kullanarak daha kısa hesaplama süresini elde etmek için, problemin önce dışbükey veya doğrusal bir biçimde formüle edilmesi gerekir. Bu genellikle, çözümün kesinliğini etkileyecek olan basitleştirmeler ve yaklaşımlarla sağlanır. Ayrıca, problemin formülasyonuna bağlı olarak, global optimal çözüm her zaman garanti edilmez. Öte yandan, dinamik programlama, küresel bir optimal çözüme ulaşmak için alt problemleri özyinelemeli olarak çözer. Herhangi bir karmaşıklık seviyesindeki problemleri çözebildiğinden, problemi belirli bir formda formüle etmeye gerek yoktur. Muhtemelen DP'nin incelenen literatürlerde yaygın olarak benimsenmesinin nedenlerinden biri budur. Bununla birlikte, "boyutluluğun laneti" olarak adlandırılan durum nedeniyle, hesaplama karmaşıklığı durum sayısı ile katlanarak artar ve bu nedenle DP'nin uygulanması, daha az sayıda durumla ilgili problemlerle sınırlıdır. Ayrıca, yapımı daha kolay olan DP çözücüler, ekonomik açıdan pahalı olan bazı ticari NLP çözücülere kıyasla ekonomik açıdan avantajlıdır.

Hız yörünge optimizasyonun temel motivasyonu, sürücü sürüş stilinden kaynaklı aşırı enerji tüketimini engellemektir. Bunu engellemek için ise gidilecek olan yolun eğim, hız limiti gibi bilgileri navigasyon sistemi üzerinden kullanılmaktadır.

Yukarıda da bahsedildiği gibi hız yörünge optimizasyonu yaparken sadece enerjiyi dikkate aldığımızda karşımıza sürüş süresinin uzaması bir problem olarak çıkacaktır. Çünkü aracın daha hızlı gitmesi durumunda tüketilen enerji karesiyle artacaktır. O yüzden optimizasyon daha az enerji tüketmek için hep aracın daha yavaş gitmesini isteyecektir. Bundan dolayı enerji ve seyahat süresi arasında bir dengeleme olması gerekmektedir.

Bu tez çalışmasında, sürücü tarafından navigasyon sistemine girilen hedef nokta ve varış süresi bilgileri kullanılarak, bu noktaya verilen sürede en az yakıt tüketerek gidilmesini sağlayacak olan hız profilini periyodik bir şekilde güncelleyen bir çalışma gerçekleştirilmiştir.

Bu problemi çözmek için yukarıdaki karşılaştırmalar sonucu DP metodu kullanılmıştır. Dinamik programlama bize verilen sınır koşulları altında her zaman global optimum davranışı vermektedir. Tek durum değişkeni olarak ise aracın hızı kullanılmış ve optimizasyonu da mesafe kademeleri üzerinden ayrık olarak gerçekleştirilmiştir.

Navigasyon sisteminden alınan hedef nokta ve seyahat süresi bilgileri üzerinden sürekli olarak gidilmesi gereken yere zamanında varılması için gereken ortalama hız optimizasyona giriş olarak verilip DP durum uzayını sürekli güncellenmektedir. Bunun temel sebebi DP ihtiyaç duyduğu hafıza yükünü azaltmaktır. Böylelikle sabit sayıda bir durum aralığı taranmaktadır. Fakat taranan aralık değerleri bu hız girişine göre güncellenmektedir.

Optimizasyon için boylamsal taşıt modeli kullanılmıştır. Güç aktarım sisteminin limitleri de sınır koşulu olarak optimizasyonun bir parçasıdır. Optimizasyon koşmadan öncede durumlar arası geçişin mümkün olduğu durumları sadece optimizasyona dahil etmek için ayrıca bir ön hesaplama yapılmaktadır. Böylelikle erişilemeyecek olan durumları optimizasyona dahil etmeyerek hesaplama süresinin kısaltılması hedeflenmiştir.

Optimizasyon belirli bir ufuk boyunca gerçekleşmektedir. Bu ufuk için hesaplanan hız profili araç hız kontrol birimine giriş olarak iletilmektedir. Araç bu hız profilini takip etmektedir. Optimizasyon belli bir süre sonra tekrar güncellenerek bir sonraki ufuk için hesaplanan hız profilini araca iletmektedir. Bunun amacı gerçek sürüş esnasında aracın verilen hızı herhangi bir sebepten ötürü takip edememesi durumunda optimizasyon yeni koşulları baz alarak tekrardan gerçekleştirilir. Böylelikle gerçek zamanlı olarak sürekli global optimuma en yakın hız profilini kullanacak şekilde aracın ilerlemesi sağlanır.

Çalışmada, iki farklı eğime sahip rotada tam elektrikli kamyonun simulasyon ve analizleri gerçekleştirilmiştir. Her iki rota içinde farklı sabit hız değerleri ve hız yörünge optimizasyonunun ürettiği hız profili ile testler gerçekleştirilmiştir. Gerçekleştirilen simulasyonlar sonucunda enerji tüketiminden %4'e kadar, hedeflenen zamandan ise %2.5'e kadar tasarruf edildiği gözlemlenmiştir. Önerilen adaptif ağırlık faktörü sayesinde ise farklı rota, varış süreleri ve taşıt parametreleri için zaman-enerji dengesi korunduğu gözlemlenmiştir.

1. INTRODUCTION

In the course of the most recent years, the electrification of transportation has gotten increasingly significant. In the coming years, stricter emission regulations, lower battery costs, all the more broadly accessible charging infrastructure, and expanding purchaser acknowledgment will make a new and powerful driving force for the dissemination of electric vehicles (hybrid, plug-in, battery-electric, and fuel cell). The automotive consumer will decide the speed of this adoption (total cost of ownership will definitely have an effect) and laws and regulations will be a push, which will vary strongly at the regional and local level [1].

Thanks to technologies that have started to develop rapidly in recent years, such as electrification, connected vehicles, and autonomous driving, different and advanced optimization methods have become available. Speed trajectory optimization is part of these methods. The speed trajectory optimization function also uses the relevant road information along the determined horizon, the topographic map data, and the internal sensor data on the vehicle. Optimization is carried out periodically under the obtained information and variable boundary conditions. The calculated optimum speed trajectory is transmitted to the driver assistance system as a speed input. Thus, the vehicle can drive efficiently in terms of both time and energy consumption.

1.1 Literature Review

The earliest studies on vehicle speed trajectory optimization date back to 1977 [2]. In this study, velocity trajectory calculation was performed using the longitudinal vehicle model to minimize consumption. There are many different approaches to the velocity trajectory optimization problem in the literature. These approaches have been tried to be categorized in Figure 1.1.

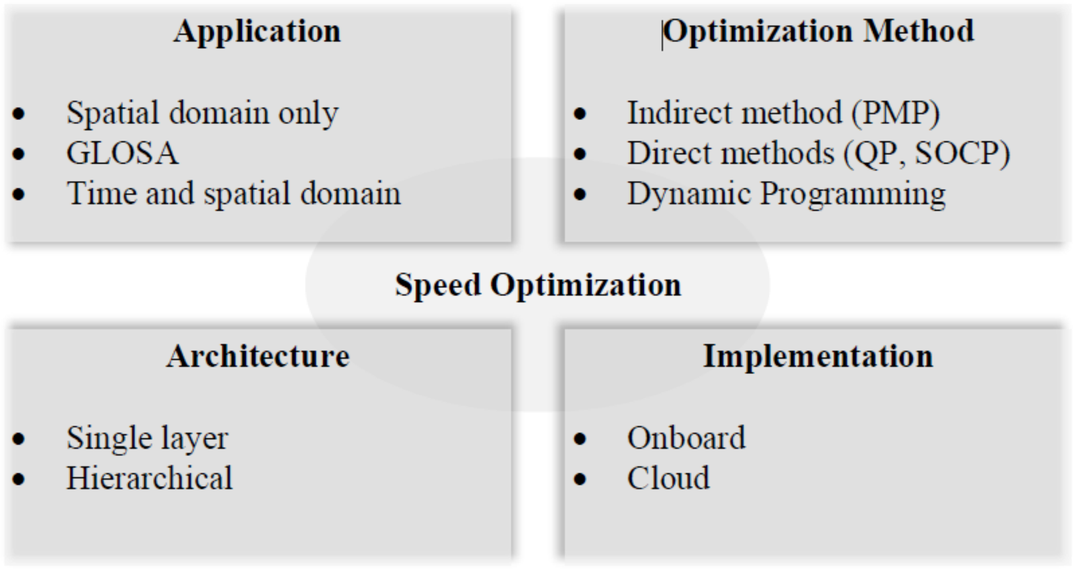

Figure 1.1 : Categories of speed optimization problem from different approaches **[3]**.

Reducing fuel consumption has often been the most important goal of optimizations [2-22]. Most of these studies have considered reducing travel time as a direct trade-off having minimum consumption. To achieve these purposes, various approaches are introduced. Using constraints under the spatial domain was observed such as road topography and speed limit information in order to optimize speed trajectory. Considering the potential to reduce energy consumption due to high inertia, especially in heavy commercial vehicles, and the average annual mileage, it is inevitable that it has received the largest share of studies in the literature [4], [5], [6], [7], [8], [9], [10], [11], [12]. Given that widespread use of heavy-duty trucks is normally on highways, traffic lights are not normally considered a restriction. The absence of constraints in the time domain also simplifies the problem. Figure 1.2 shows an illustration of such an application.

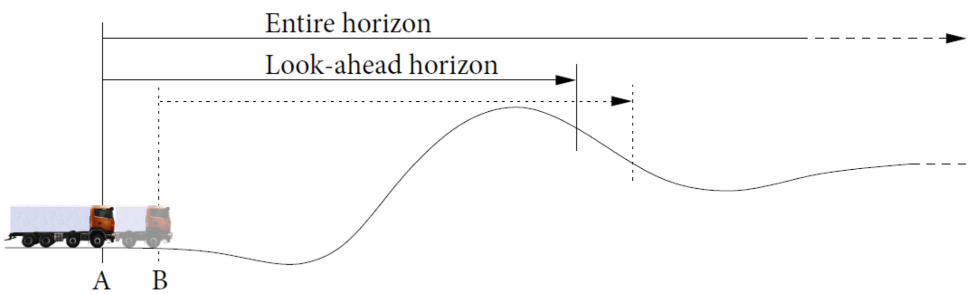

Figure 1.2 : Look ahead horizon of vehicle speed trajectory optimization in the spatial domain **[11].**

Optimization of velocity trajectory can be defined as an optimal control problem and there are different methods to obtain the solution. Generally, these methods fall into three groups: indirect methods, direct methods, and dynamics programming (DP) [13].

The first group, indirect methods, use Pontryagin's Principle of the Minimum (PMP). These methods solve the problem numerically and "indirectly" after transforming the problem into a multi-point Hamiltonian boundary value problem [14]. In the first study on this subject, the fuel flow rate of the internal combustion engine was modeled nonlinearly for constant road slope with an approximate polynomial equation as a function of engine speed and power, and PMP was used to solve the fuel consumption minimization problem of a vehicle by Schwarzkopf and Leipnik in 1977 [2]. In later studies, traffic light scenarios started to be included in the optimization. In this study, a solution to the optimization problem was sought by using a simple engine fuel consumption and linearized vehicle model [15].

Direct methods can be used as another group for solving the problem. Direct methods parametrize the state and control variables into the Nonlinear Programming (NLP) problem. For example, in hybrid electric vehicles for predictive energy management, battery energy, and vehicle speed are defined as state variables to use in Second Order Cone Program (SOCP) and Quadratic Programming (QP) [8]. It was seen that the problem was re-formulated convexly using approximations and calculated faster by SOCP. In the next step, the main purpose is to fit the requirement of QP solver, thus the problem is reformulated again. These two solvers provide approximately the same results for a given test scenario.

DP separates the issue into subproblems whose solution can be found by using “Bellman's Principle of Optimality”. It has been proven by many studies that the most used method in solving the vehicle speed optimization problem is the DP [5-20]. In one of these studies, for different scenarios to optimize speed trajectory, the time-based stage variable forward recursive DP is selected [16].

However, to eliminate time as a condition in the problem, developing a method that uses distance on behalf of time as a stage variable and introducing trip time into the cost function is proposed in [9] and also chosen for this thesis as well. [5] show that there is a possibility to reduce the computational cost by decreasing searching space in real-time implementations. Thanks to this approach, the eliminated constraints are not included search space and the computational cost is decreased.

In brief, problems that include a larger number of state variables can be solved faster by using indirect and direct methods instead of dynamics programming. On the contrary, the problem must be convex or linear form to be used by efficient solvers having a faster solution, and this affects the solution precision. Also, indirect and direct methods can not give the guarantee have the global optimal solution. However, dynamic programming separates into subproblems to reach the global optimal solution. DP does not need to specific form for any complex problem. This situation is the main reason for the common usage of DP in the literature, but the "curse of dimensionality" is a drawback of DP which increases computational cost and hence DP examples generally have used fewer states.

1.2 Outline of Thesis

The thesis content is organized into five chapters. Chapter 1 includes general introduction and literature review which covers different energy optimization methods. Chapter 2 includes the theory of dynamic programming and the detailed formulation of BELLMAN’s principle. Nonlinear longitudinal dynamic vehicle modeling and a detailed formulation and implementation of the DP algorithm, which covers the determination of optimization objective, stage and state variables, cost function, boundary conditions, functional architecture in Chapter 3. In Chapter 4, simulation results are presented for two route case studies. Finally, conclusions and future work possibilities are discussed in Chapter 5.

2. THEORY OF DYNAMIC PROGRAMMING

Dynamic programming is a method in which complex optimization problems can be broken down into a number of simpler problems. The combination of the solutions to these simpler problems leads to the solution of complex optimization problems. The greater the number of simpler problems, the greater the probability of a globally optimal solution for the optimization problem to be found under the given boundary conditions. In order to be able to apply the DP, the optimization problem must be related to each other subdivide building sub-problems. One speaks with such optimization problems also of multi-stage decision-making processes.

2.1 Multi-Stage Decision-Making Process

Due to the non-linear drive train characteristics and the changing driving resistances, when the system "vehicle" is operated under real ambient conditions, the resulting system behavior and thus the optimal control of parameters such as time and distance arise. In decision theory, such a process is generally referred to as a "multi-level decision-making process" [17]: In each process step, a decision must be made regarding process control, which in turn influences the possible decisions of subsequent process steps and thus the overall result. The principle of dynamic programming enables the optimization of this type of process and the derivation of optimal process control with regard to the underlying criteria. The basic features of dynamic programming go back to the American mathematician R. E. BELLMAN, who coined the term around 1950 [18]. Dynamic programming describes less a single explicit algorithm than a basic principle for solving multi-level decision problems [19], based on the well-known BELLMAN optimality principle [17] :

> "An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision".

Since it was first formulated, different variations of dynamic programming have been developed and applied in different forms depending on the problem. The classic deterministic dynamic programming (DDP), however, describes a numerical solution method that requires a time discretization of the process to be regulated as well as a complete value discretization of the state space. The originally time-continuous state-space model from equation (2.1) and (2.2) is time-discretized by putting system output $y(t)$, system state $x(t)$, control $u(t)$ and disturbance $w(t)$ in $k = 0, 1, \dots, N$ steps are sampled in discrete time in equation (2.3).

$$\dot{x}(t) = f\big(x(t), u(t), w(t)\big), \quad x(t_0) = x_0 \tag{2. 1}$$

$$y(t) = h(x(t), u(t), w(t)) \tag{2. 2}$$

$$y_k = y(t_k), \;\; x_k = x(t_k), \;\; u_k = u(t_k), \;\; w_k = w(t_k), \tag{2. 3}$$

The numerical integration according to the explicit EULER method generates the discrete-time, non-linear state difference equation with discrete-time system function ϕ as a calculation rule for the subsequent x_{k+1}, depending on the current state x_k, the control u_k used and the current disturbance variables w_k:

$$x_{k+1} = \phi(x_k, u_k, w_k), \quad x(0) = x_0, \;\; k = 0,1, \dots, N-1 \tag{2. 4}$$

2.2 BELLMAN's Principle of Optimality

Using dynamic programming, a complex dynamic optimization problem is broken down into a sequence of similar sub-problems and efficiently solving the overall problem by avoiding recursions can be put together from the individual partial solutions [20] [18]. Figure 2.1 shows an exemplary application of the principle [20].

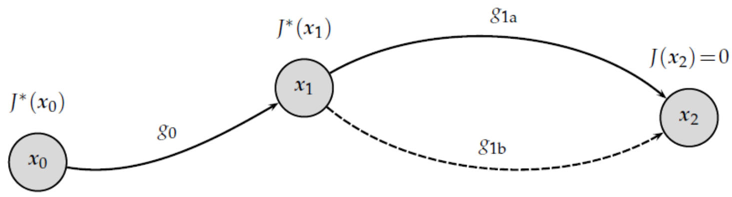

Figure 2.1 : Example for BELLMAN's principle of optimality.

A multi-stage decision-making process with the three system states is sketched $X = \{x_0, x_1, x_2\}$. The optimal transfer of the system from the initial state x_0 in final state x_2 causes minimal costs $J^* = J_{\pi^*} = \min\{J_\pi(x_0)\}$ in the state transitions and results from the application of the optimal control law π^*. By checking the permissible state transitions, the minimum cost J^* in this example to:

$$\begin{aligned} J^*(x_1) &= \min\,\{g_{1a}, g_{1b}\} + 0 \\ J^*(x_0) &= \; g_0 + J^*(x_1) \end{aligned} \tag{2. 5}$$

By generalizing the illustrated example, the BELLMAN's principle of optimality considering the remaining cost-to-go in any formulated transition state x_i [19]:

$$J^*(x_i) = J_{\pi_0^*}(x_i) = \min\left\{g_N(X_N) + \sum_{k=i}^{N-1} g_k(x_k, u_k, w_k)\right\}, \tag{2. 6}$$

2.3 BELLMAN's Recursion Equation

With the help of complete induction, BELLMAN's recursion equation of dynamic programming can be derived from the optimality principle formulated in equation (2.6), which solves the dynamic optimization problem backward recursively starting from the final state x_N [19]:

For each initial state x_0 the minimum costs $J^*(x_0)$ of the optimization problem correspond to the costs $J_0(x_0)$ resulting from the following algorithm, which goes backward from step $N - 1\ to\ 0$.

$$J_N(x_N) = g_N(x_N), \tag{2. 7}$$

$$J_k(x_k) = \min_{u_k \in U_k(x_k)} \{g_k(x_k, u_k, w_k) + J_{k+1}(\phi(x_k, u_k, w_k))\}, k = 0,1, \dots, N-1 \tag{2. 8}$$

If the control variable $u_k^*(x_k)$ minimizes the right-hand side of equation (2.7) and equation (2.8) for each x_k and, then the underlying control law π^* is optimal.

Dynamic programming is used in many different disciplines such as decision theory, control engineering, graph theory, or operations research. The optimization of the driving strategy can be formulated as a problem for each of these disciplines under different conditions. With regard to real-time implementation, the interpretation of the problem as the Shortest-Path problem of graph theory is particularly suitable.

3. OPTIMIZATION IMPLEMENTATION BASED ON DYNAMIC PROGRAMMING

3.1 Vehicle Modelling

The total resistance of a wheel consists of four main components.

Aerodynamic friction loss F_A can be written as equation (3.1).

$$F_A = \frac{1}{2} * \rho_{air} * A_{front} * C_d * \ v^2 \tag{3. 1}$$

where ρ_{air} denotes the air density, A_{front} denotes the vehicle's cross-frame area, c_d denotes the aerodynamic resistance coefficient.

Rolling friction loss F_R can be written as equation (3.2).

$$F_R = f_r * \ m_{veh} * g * \cos(\alpha)\,, \ \ v > 0 \tag{3. 2}$$

where f_r denotes the road friction coefficient as a constant, m_{veh} denotes the vehicle mass including passenger and payload, g denotes the gravity of earth, α denotes the average road inclination.

Slope driving force F_G can be written as equation (3.3).

$$F_G = m_{veh} * g * \sin(\alpha) \tag{3. 3}$$

Acceleration force F_{Iner} can be written as equation (3.4).

$$F_{Iner} = (f_{iner} * \ m_{veh} + \ I_R\ ) * a \tag{3. 4}$$

where f_{iner} denotes the inertia equivalent factor for vehicle mass and I_R rotational inertia value of all rotating components reduced to the wheel.

Total wheel level force is the sum of the above elements, provided in equation (3.5).

$$F_{Total} = F_{Iner} + F_A + F_R + F_G \tag{3. 5}$$

3.2 Optimization Objective

The objective is to optimize the vehicle speed trajectory of a BEV with a single gear, in order to minimize the energy demand without sacrificing travel time by considering road slope, road speed limitations. There are indeed various other factors influencing the energy demand apart from vehicle speed trajectories, such as power split between the combustion engine and electric motor in case of hybrid powertrains or gear position in case of a multi-gear transmission, thermal management as well. These factors will not be the focus of this work but can be considered in low-level optimization layers.

3.3 Algorithm Development DDP

Before formulating the problem, the basic elements of DP are visualized in Figure 3.1. Stages are displayed as discrete points along the horizontal axis. In the automotive industry, the stages of a DP problem are often defined as the time or distance within the problem range.

States show the data which can sufficiently assess the outcomes of various choices. Along these lines, the state should be characterized in such a manner, that the outcome of various choices can be reflected by various states. Another significant property that the state ought to have is to pass on sufficient data to settle on future choices regardless of how the cycle arrived at the present status. Figure 3.1 shows a single-dimensional state DP issue, where the states are addressed by discretized circles. For multi-objective issues, multi-dimensional states frequently should be characterized.

In the wake of describing stages and states, transitions between states can be described. Since the issue is partitioned into sub-issues by discrete stages, only transitions between states in adjoining stages are fundamental for solving the issue.

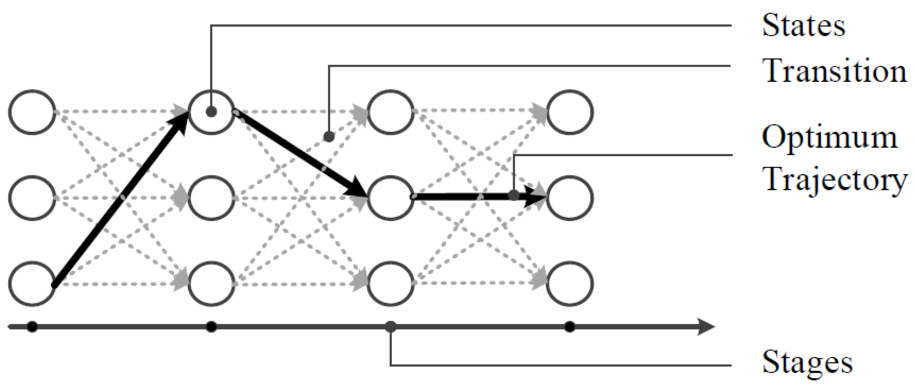

Figure 3.1 : Definition of Stage, State, and Transition of DP.

For dynamic programming problems in automotive implementations, time is often used as the stage variable especially for issues with time-varying inputs driving cycles, which is based on speed over time [21] [22] [23]. But it is more useful to use distance as a stage variable in problems where data related to distance such as speed limit change, traffic lights, curvatures are used. Although the actual position can be determined using the time and the vehicle speed due to effects of lane changes and road gradients calculation will not have good precision. For this reason, navigation systems are used in real applications.

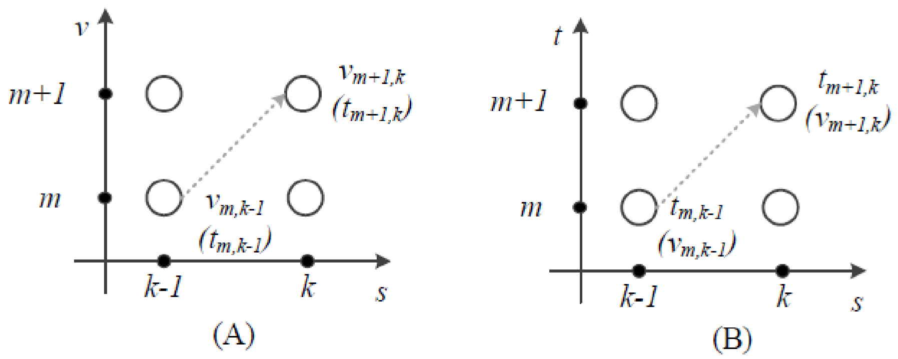

Figure 3.2 : State variable definition.

The vehicle speed can be determined as a state that has the obvious advantage of simplicity because chosen optimization variable is also speed. The other advantage of choosing speed as an optimization variable is the calculation of "cost-to-go" exclusion recursive terms.

Equation 3.17 shows the "cost-to-go" calculation. In that equation, $\bar{v}$ refers to average vehicle speed, $\bar{a}$ refers to average acceleration, and dt refers to transition time from one state to another. The i, j and k are the transition from state i to state j at later stage k.

After determining vehicle speed as state, average vehicle speed $\bar{v}_{i,j,k}$ can be written as:

$$\bar{v}_{i,j,k} = \frac{v_{j,k} + v_{i,k-1}}{2} \tag{3. 6}$$

Average acceleration $\bar{a}_{i,j,k}$ can be written as:

$$\bar{a}_{i,j,k} = \frac{v_{j,k}^2 - v_{i,k-1}^2}{2 * (s_k - s_{k-1})} \tag{3. 7}$$

Transition time $dt_{i,j,k}$ can be written as:

$$dt_{i,j,k} = 2 * \frac{s_k - s_{k-1}}{v_{j,k} + v_{i,k-1}} \tag{3. 8}$$

j^* means the index of the state which gives minimum cost from state j at later stage $k + 1$ to state i at the previous recursion state.

$$j^* = \min_{j \,\epsilon\, 1,2,\ldots,M} J_{i,k}(j) \tag{3. 9}$$

M refers to the number of states. The following equations represent different recursion directions.

Forward recursion,

$$t_{j,k}^* = t_{i^*,k-1}^* + \ dt_{j^*,i,k} \tag{3. 10}$$

Backward recursion,

$$t_{j,k}^{*} = t_{i^{*},k+1}^{*} - dt_{i^{*},j,k} \tag{3. 11}$$

In equation 3.8, if both state $v_{j,k}$ and $v_{i,k-1}$ have zero value, zero division problem can occur. For this reason, zero speed should be removed from the state space or can be defined very small number for the minimum speed state instead of zero.

The zero division problem can be handled by selecting the time as the state variable, as the time calculation for transition does not have division.

Average acceleration $\bar{a}_{i,j,k}$ can be written as:

$$\bar{a}_{i,j,k} = 2 * \frac{\bar{v}_{i,j,k} - v_{i,k-1}^{*}}{t_{j,k} - t_{i,k-1}} \tag{3. 12}$$

Transition time $dt_{i,j,k}$ can be written as:

$$dt_{i,j,k} = t_{j,k} - t_{i,k-1} \tag{3. 13}$$

Average vehicle speed $\bar{v}_{i,j,k}$ can be written as:

$$\bar{v}_{i,j,k} = \frac{s_k - s_{k-1}}{t_{j,k} - t_{i,k-1}} \tag{3. 14}$$

When the eliminated state is the optimal vehicle speed, speed equations can be written for forward recursion as:

$$v_{j,k}^{*} = 2 * \bar{v}_{i^{*},j,k} - v_{i^{*},k-1}^{*} \tag{3. 15}$$

For backward recursion as:

$$v_{j,k}^{*} = 2 * \bar{v}_{i^{*},j,k} - v_{i^{*},k+1}^{*} \tag{3. 16}$$

However, usage of time has also disadvantage which is detailed mentioned in [3]. To conclude, the disadvantage of determining time as the state is compelling to be used in real-time application. Thus, for the state variable definition, the one-state formulation with vehicle speed is selected.

3.3.1 Cost function

After disregarding the time from state variable determination, the cost function of the optimization includes the travel time penalty term. Since the required wheel force for the vehicle increases quadratically with the vehicle speed, trip time and required energy for the vehicle have a nonlinear trade-off. In addition to that, based on the selected step size for the state variable discretization, some uncomfortable speed change trajectory profile maybe occur, to eliminate that comfort penalty term is added using acceleration value.

$$J_{i,j,k} = dE_{i,j,k} + \beta * dt_{i,j,k} + \gamma * \bar{a}_{i,j,k} \tag{3. 17}$$

J refers to the "cost-to-go"; dE refers to the required energy for state transition; dt refers to the trip time for state transition; a refers to the average acceleration while state transition. β is an adaptive weighting factor for energy consumption and trip time trade-off adjustment. γ is also a weighting factor for comfort adjustment.

In 3.1, total wheel force calculation is introduced. Based on that the required energy for each discrete state transition can be calculated as (3.18).

$$\begin{aligned} dE_{i,j,k} = \Big\{\Big[&\frac{1}{2} * \rho_{air} * A_{front} * C_d * \bar{v}_{i,j,k}{}^2 + f_r * m_{veh} * g \\ &* \cos(\bar{\alpha}_k) + m_{veh} * g * \sin(\bar{\alpha}_k) \\ &+ (f_{iner} * m_{veh} + I_R) * \bar{a}_{i,j,k}\Big] * \bar{v}_{i,j,k} + P_{loss_{i,j,k}}\Big\} \\ &* dt_{i,j,k} \end{aligned} \tag{3. 18}$$

$P_{loss_{i,j,k}}$ refers to the combined power loss of electric motor, inverter. It can be determined based on required torque and electric machine speed at state i using look-up tables.

3.4 Implementation

The implementation of the optimization algorithm and other subsystems is performed by using Simulink®. Other related systems and task management become necessary for realization. Figure 3.3 shows the velocity trajectory optimization functional architecture.

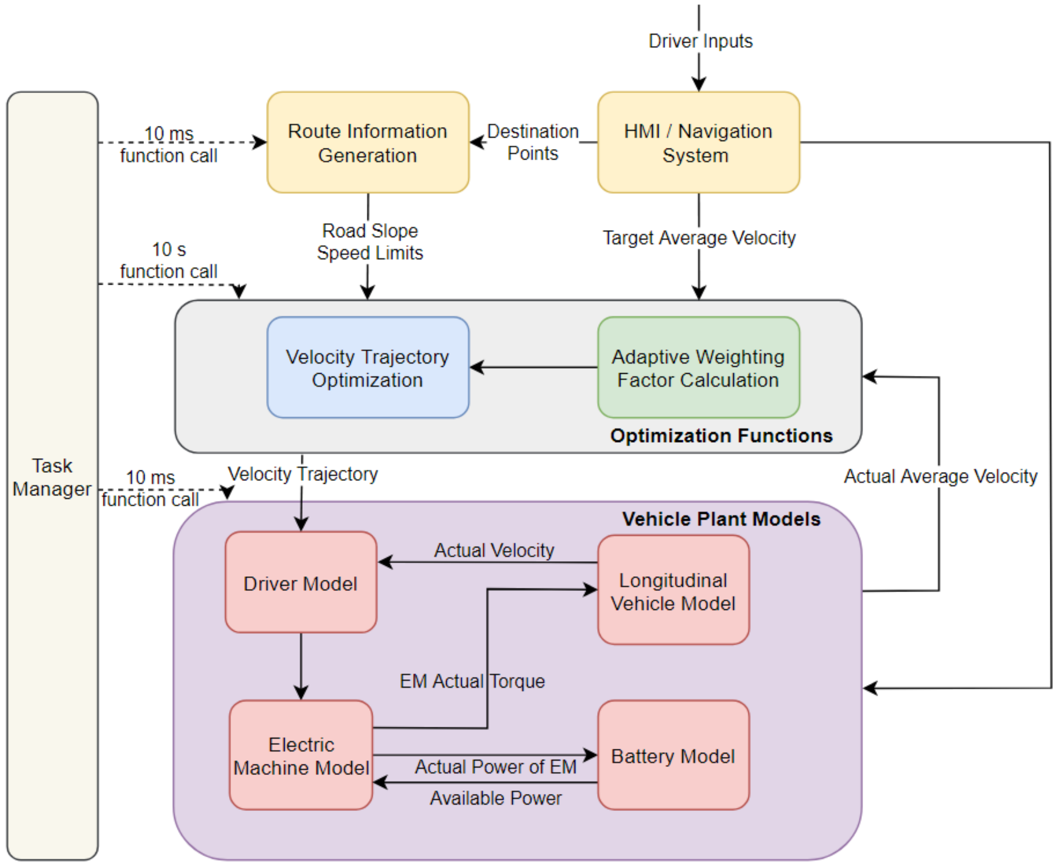

Figure 3.3 : Velocity trajectory optimization functional architecture.

Deciding on a prediction horizon is critical to velocity trajectory optimization since it affects the memory requirement, computation effort, and update time of route information generation function directly in real-time applications. Another important parameter is the update interval of the optimization task due to their close relationship with each other, the update interval and prediction horizon should be determined together. Figure 3.4 shows the distance-time plot contains the relation among prediction horizon S_h and update interval of the speed optimization function.

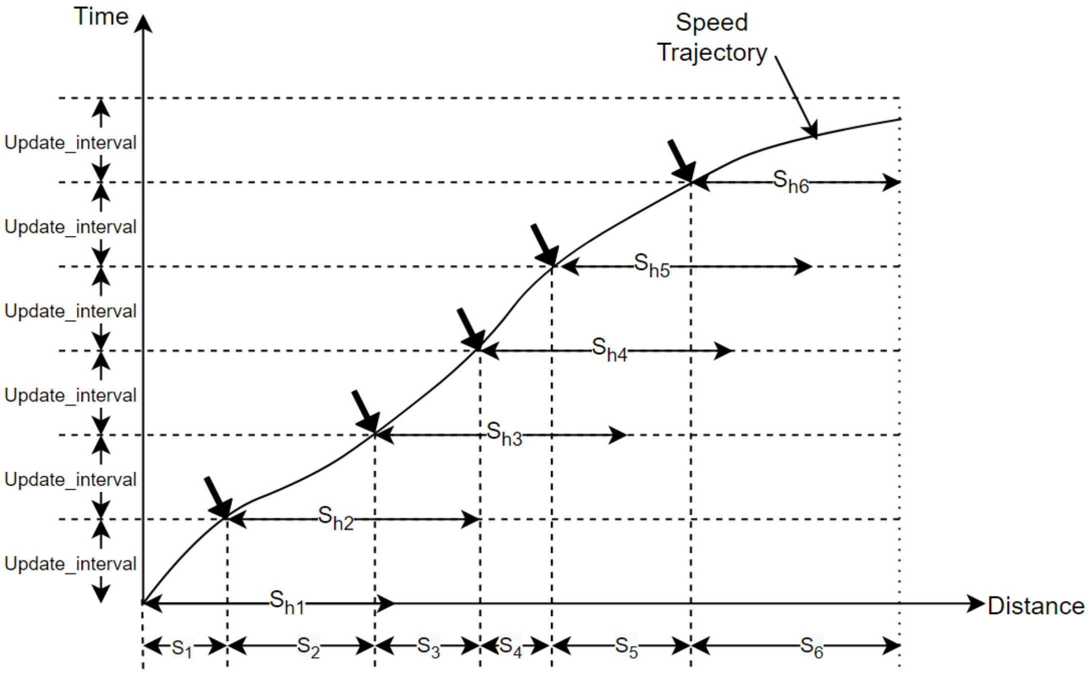

Figure 3.4 : Update interval and prediction horizon.

The distance traveled approaches the prediction horizon with the long update interval. Since the optimization function will use up-to-date route information with short intervals, the effect of the deviation may be more significant than the shorter update interval. On the other side, a very shorter update interval will not be reasonable when the vehicle dynamic response is considered. To solve this problem, 10 s is chosen for the update interval of the optimization functions, and with this, the optimum velocity trajectory relative to the position is applied until the next function call.

3.4.1 Route information generation

First of all, as a basis, the information of the intended route and the target arrival time are taken from the driver. After receiving this information from the HMI (Human Machine Interface), the target average speed for the relevant route is calculated. Along with the targeted speed, the slope and speed limit information of the targeted route is also given as input to the velocity trajectory optimization algorithms.

3.4.2 Optimization functions

Velocity trajectory optimization consists of multiple sub-functions. These sub-functions and their relations with each other are indicated in the diagram in Figure 3.5.

All functions in this diagram are written in C-code and used in Simulink as S-functions.

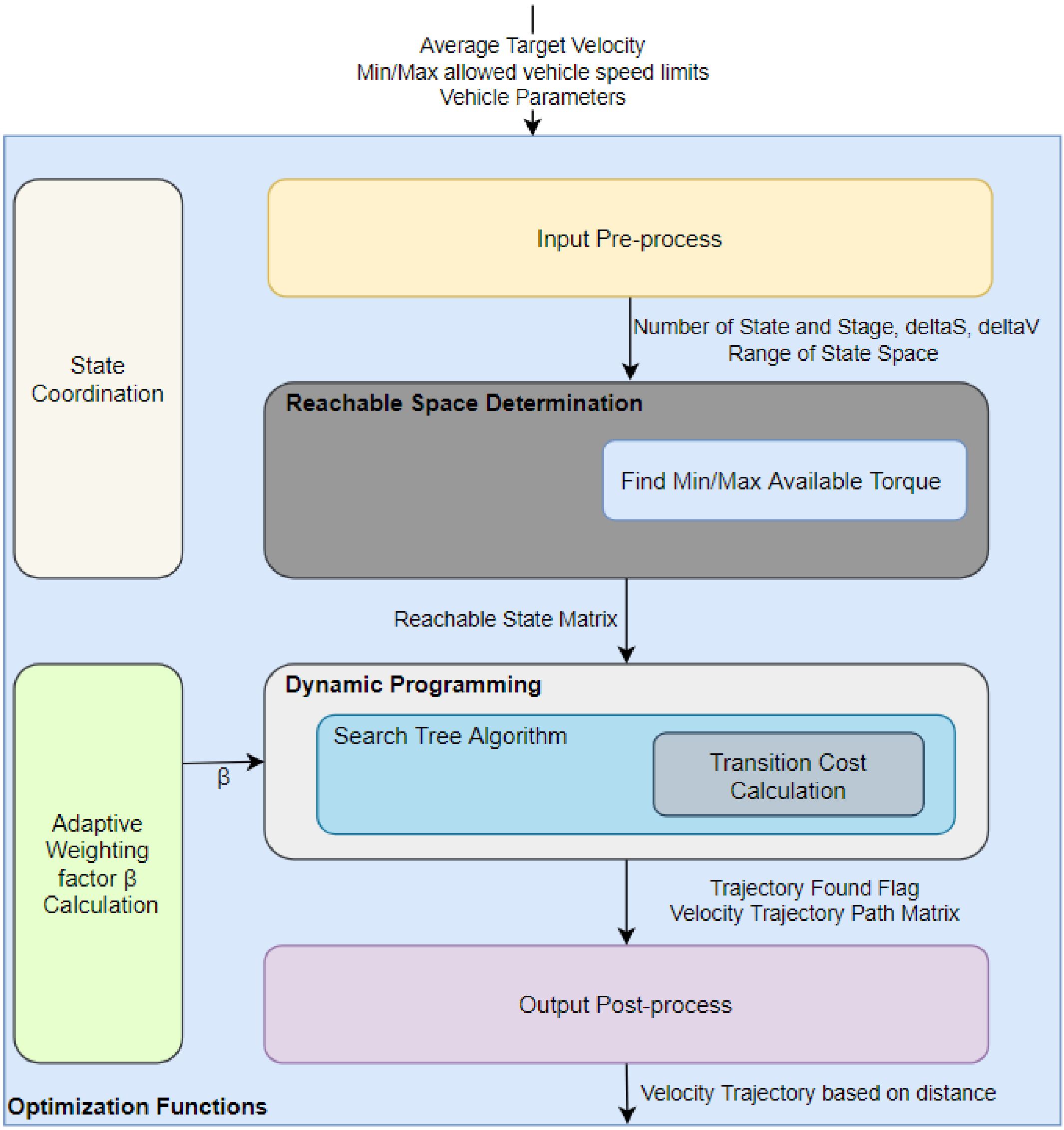

Figure 3.5 : Functional architecture of optimization functions

3.4.2.1 Input pre-process

In this partition of the software, the number of state and stage, step size of distance, step size of velocity state, and overall range of state space are provided to optimization functions by using average target velocity, speed limits, and vehicle parameters.

3.4.2.2 State coordination

The state coordination function is mainly responsible for enabling the optimization function. If the actual speed of the vehicle is not in the working range of the optimization function or the driver disenables the usage of this feature, state coordination will stop the optimization and the vehicle will be used constant average velocity target for cruise control.

3.4.2.3 Reachable state determination

"Curse of dimensionality" is a well-recognized downside of DP since it is the main reason of uprise computational cost. The simple DP formulation $O(M^2.N)$ where N means the number of nodes for distance stage variable, and M means the number of nodes for velocity as state variable gives results for velocity trajectory optimization computational complexity. For this reason, the amount of computation can be reduced by determining not reachable states according to available torques and not calculating for those states. Hereby, the computational complexity can be rewritten into $(M.N.T)$, where T represents the maximum number of states which can be reachable.

The achievable speed states are calculated for each state, taking into account the available torque of the electric machine at that speed, the resistive force to be overcome, and the powertrain efficiency, as shown in Figure 3.6.

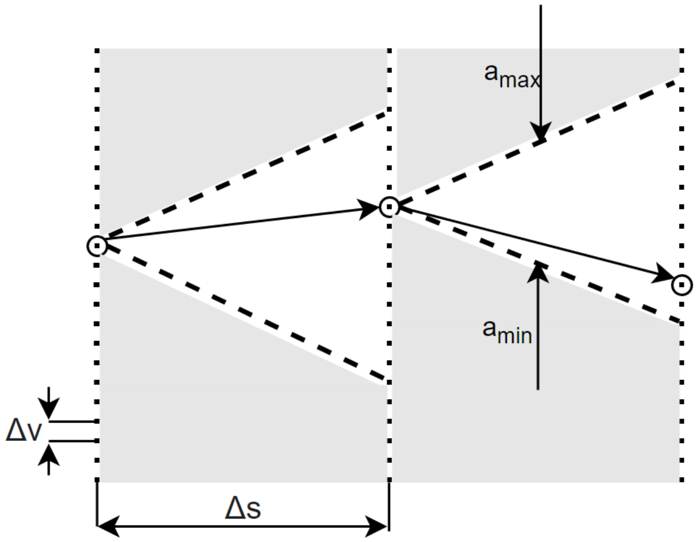

Figure 3.6 : Reachable state determination based on system limits

3.4.2.4 Adaptive weighting factor calculation

As described in 3.3.1, it is aimed to overcome the energy and time trade-off problem with the adaptive weighting factor β. When the literature is examined, it has been observed that these coefficients are usually taken as a parameter to be adjusted manually or calculated iteratively in a segment for each path, each powertrain.

However, in this study, a novel approach was used which is an adaptive parameter in run-time to calculate β. The initial value of the beta is determined with the following equation 3.19 via the time required to complete the run with target average speed [10], assuming no braking, zero slopes, and neglecting of the losses. The cost function can be written as;

$$J = \int dE + \beta . dT = S.(F_A + F_R) + \beta . \frac{S}{\vartheta_{avg}} \tag{3. 19}$$

$$\frac{dJ}{d\vartheta}(\vartheta_{avg}) = 0 \rightarrow \beta = 2 * F_A * \vartheta_{avg} = \rho_{air} * A_{front} * C_d * \vartheta_{avg}^3 \tag{3. 20}$$

After the first value is calculated according to equation 3.20, the beta factor is adapted to reach the target average velocity during the trip by using deviation between the target average velocity and the actual average velocity, as shown in Figure 3.7.

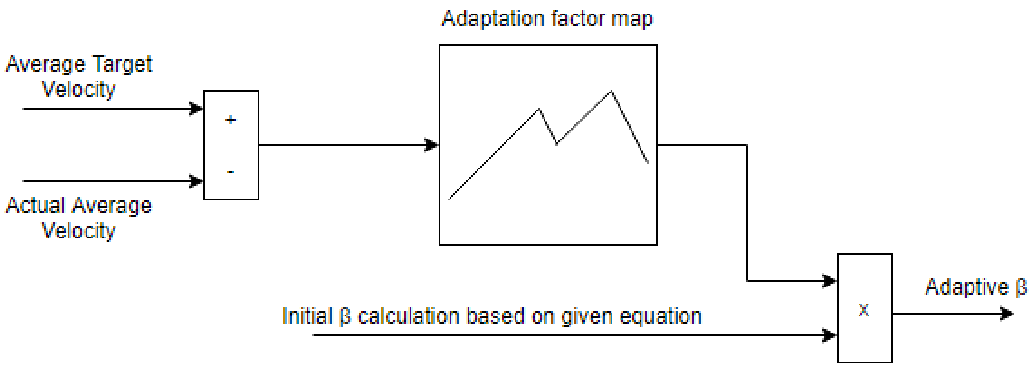

Figure 3.7 : Adaptive β calculation block diagram

3.4.2.5 Dynamic programming

In this function, search tree algorithms are developed and dynamic programming formulas are applied together with Bellman Recursion equations described in 2.3.

Transition cost calculation is performed for all reachable states which are provided over the reachable matrix from the previous function. During this calculation cost-to-go and path index matrix of optimum velocity trajectory are stored.

3.4.2.6 Output post-process

Output post-process converts the optimum velocity path index to the velocity trajectory array and provides the other optimization state, total cost, and beta values to the vehicle plant models.

3.4.3 Vehicle plant models

The vehicle plant model consists of 4 main parts as shown in Figure 3.3. The driver model basically performs the required torque demand with the PI controller, following the given set speed value. This torque request is transmitted to the electric motor model. In the electric motor model, the torque requested from the driver model is limited according to the maximum or minimum torque it can provide at that speed value and the maximum and minimum power values from the battery. While the actual requested power value calculated over the limited torque demand is transmitted to the battery model, the actual torque value is transmitted to the vehicle model.

In the vehicle model, how much this torque value will accelerate the vehicle is calculated by using the inertial parameters over the backward vehicle model.

In the battery model, the total consumed energy value with losses is calculated over requested the actual power value from the electric motor model by using instantaneous battery parameters based on actual SOC (state of charge), temperature values.

4. SIMULATION RESULTS

In this section, two different route case study analyses are performed to assess the energy reduction via optimal trajectory. Full electric trucks are used for both routes. Thanks to the high inertia of heavy-duty vehicles have a higher potential to decrease energy consumption. Since heavy-duty vehicles are generally used on highways, thus following the optimal velocity trajectory will be much easier than urban driving.

Vehicle and final selected function parameters are listed in Table 4.1 and Table 4.2.

Table 4.1 : Vehicle specifications.

Specification	Values
Battery capacity	190 kWh
Peak power of EM	350 kW
Continuous power of EM	240 kW
Vehicle mass	25000 kg
Air density	1.1839 kg/m^3
Frontal Area	9.5 m^2
Cd coefficient	0.415
Tire radius	0.5143
Final drive ratio	5.125
Maximum speed	95 km/h
Road friction coefficient	0.0055

Table 4.2 : Velocity optimization function parameters.

Specification	Values
Step size for velocity state	0.33 km/h
Step size for distance	20 m
Prediction horizon	1000 m
Task update interval for optimization functions	10 s
Task update interval for other functions	10 ms
State-space offset for set speed	[-25 km/h, 25 km/h]

4.1 Route Profile 1 Results

Three different average speed set values were created with the assumption of different arrival times for the route planned by the driver. These determined average velocity sets and velocity trajectory results calculated as a result of optimization were examined separately in terms of energy consumption and arrival time.

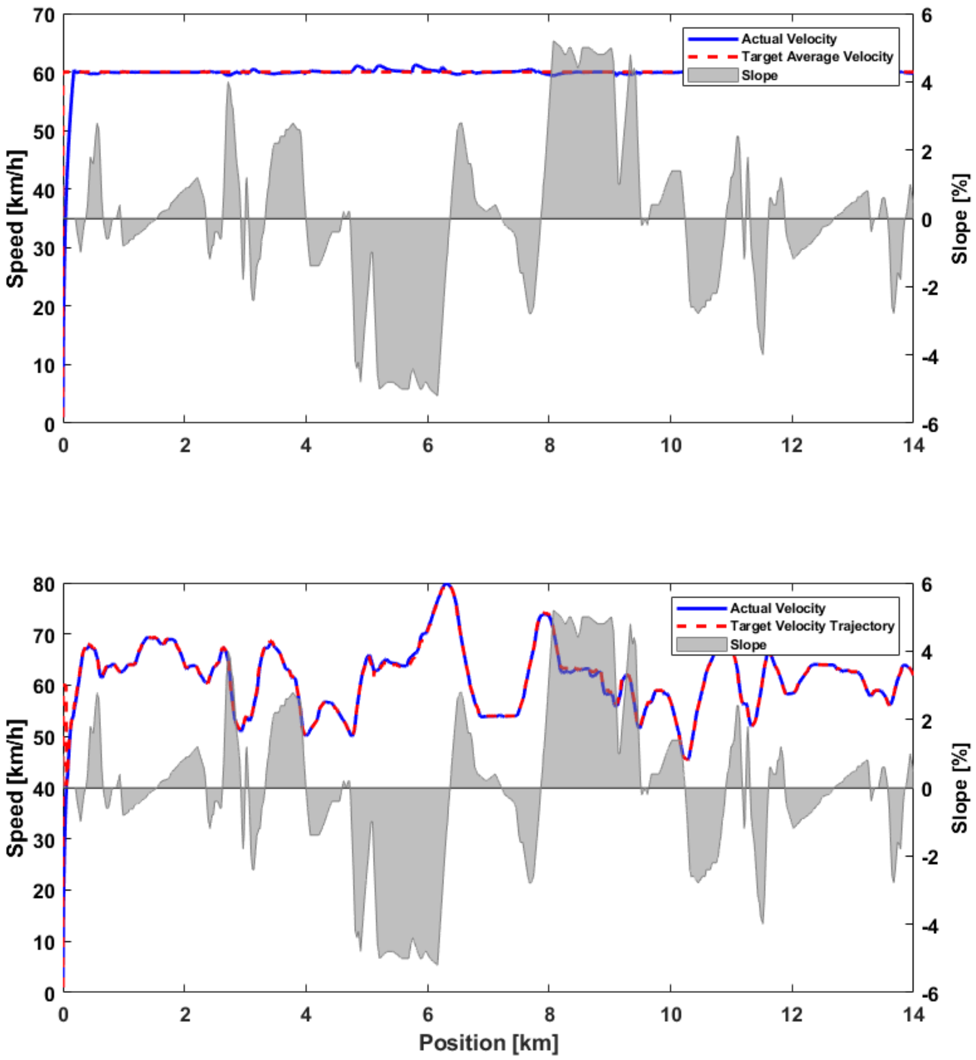

Figure 4.1 : Comparison of velocity trajectories for optimization active and inactive cases @60km/h target average speed in route profile 1

In the literature, using the authors' own or their own test drives as a base is not a fair comparison method. In this study, a comparison method with base scenarios that will follow the fixed set speed, which is a situation where acceleration is almost non-existent and only the road and aerodynamic force are affected, is preferred.

In Figure 4.1, the speed profile results obtained for road profile 1 based on the first scenario, 60 km/h average speed, are shared. The road slope profile is shown as a gray area in the graphs. It has been observed that the target velocity trajectory accelerates before uphill starts and slows down before downhill. Thus, it consumes less energy by accelerating before the start of the slope. With the acceleration brought by the descent by slowing down even before the descent starts, it does not consume energy and even in some cases, it can be recovered by regenerative braking.

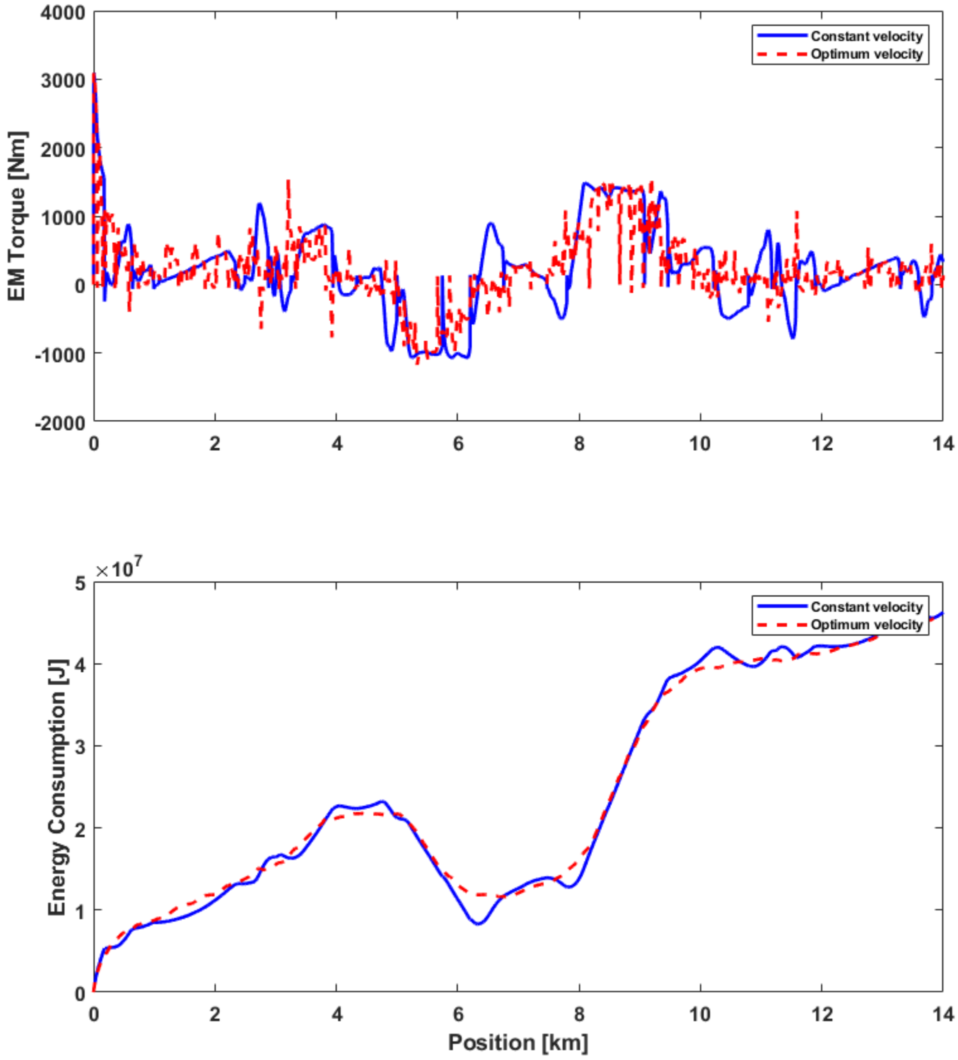

Figure 4.2 : Comparison of EM torque and energy consumption for optimization active and inactive cases @60km/h target average speed in route profile 1

In Figure 4.2, the electric motor torque and the total energy consumed from the battery for the scenario of 60 km/h average speed are shown with a blue solid line according to the position. In this case, the total energy consumed by the vehicle is 4.9068 x 10^7J.

The arrival time is 847,93 s. On the other hand, the dashed red line presented in Figure 4.2 describes the optimal velocity trajectory obtained with the optimization algorithm. The total energy consumed by the vehicle under this strategy is 4.7952 x 10^7 J. Arrival time is 833,47 s. The results show that the speed profile calculated by the optimization is 2.27% lower than the energy consumed 1.71% faster than by the vehicle driving at constant velocity.

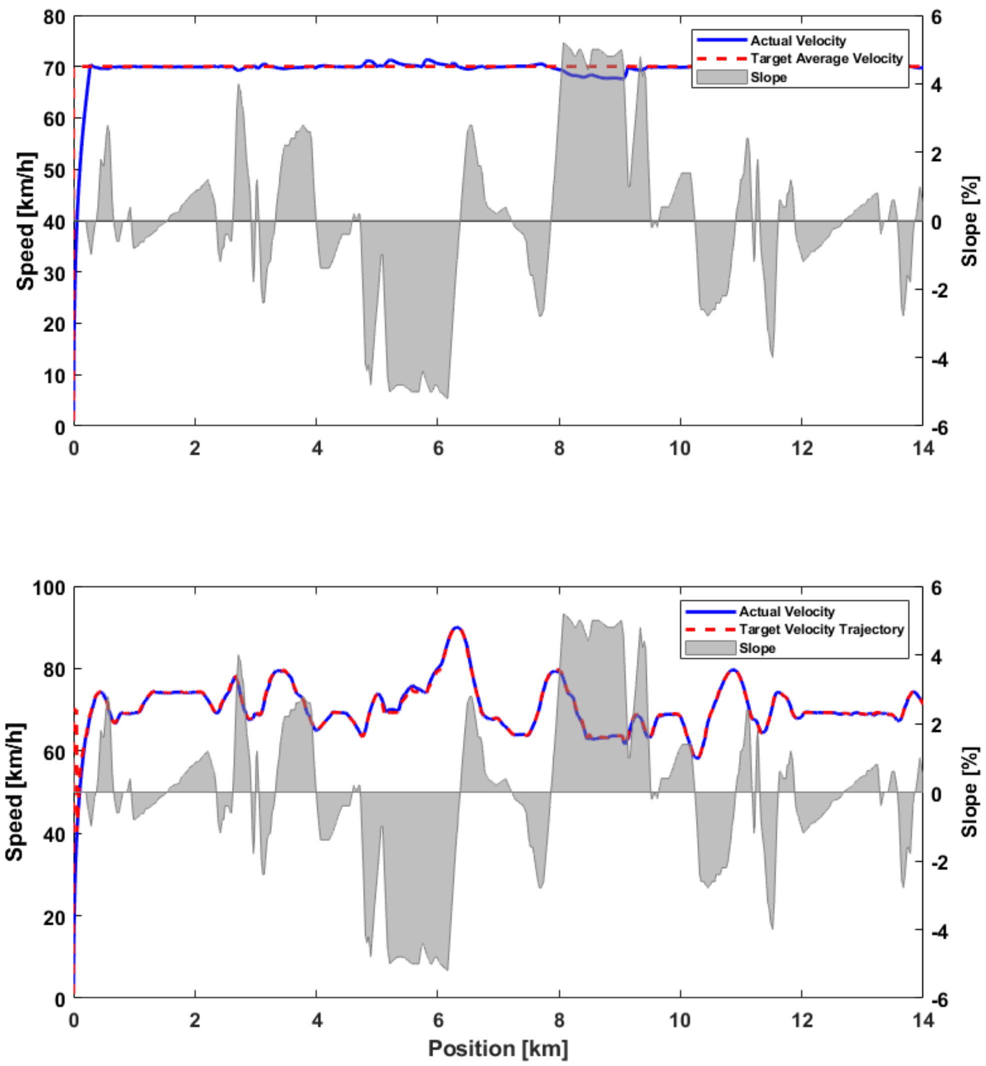

Figure 4.3 : Comparison of velocity trajectories for optimization active and inactive cases @70km/h target average speed in route profile 1

Figure 4.3 shows the velocity trajectory results for the 70 km/h average velocity target. The velocity trajectory started to accelerate between 6-7 km before the positive slope started. The velocity profile decreases at the point where the elevation starts and decreases to the minimum where it is maximum. Although there is a negative slope between 5-6 km, the reason why the vehicle does not accelerate more is that energy can be recovered thanks to regenerative braking in electric vehicles, it is prevented from accelerating by braking to the calculated speed profile.

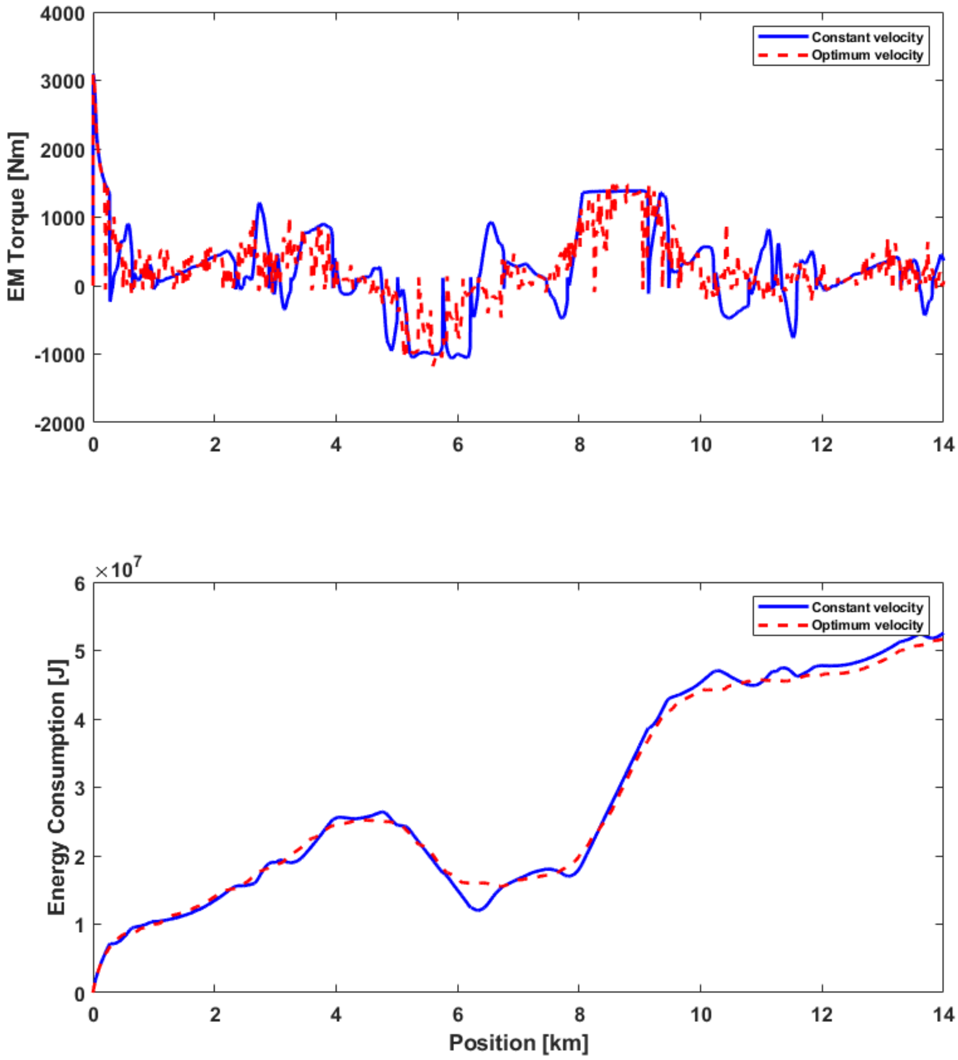

Figure 4.4 : Comparison of EM torque and energy consumption for optimization active and inactive cases @70km/h target average speed in route profile 1

In Figure 4.4, the electric motor torque and the total energy consumed from the battery are shown for the scenario of 70 km/h average speed this time. In the constant speed driving case, the total energy consumed by the vehicle is 5.5872 x 10^7 J. Arrival time is 731,36 s. On the other hand, following the speed profile calculated by the optimization, the total energy consumed by the vehicle is 5.3964 x 10^7 J. Arrival time is 724,94 s. The results show that the speed profile calculated by optimization is 3.41% lower than the energy consumed 0.88% faster than by the vehicle driving at constant velocity.

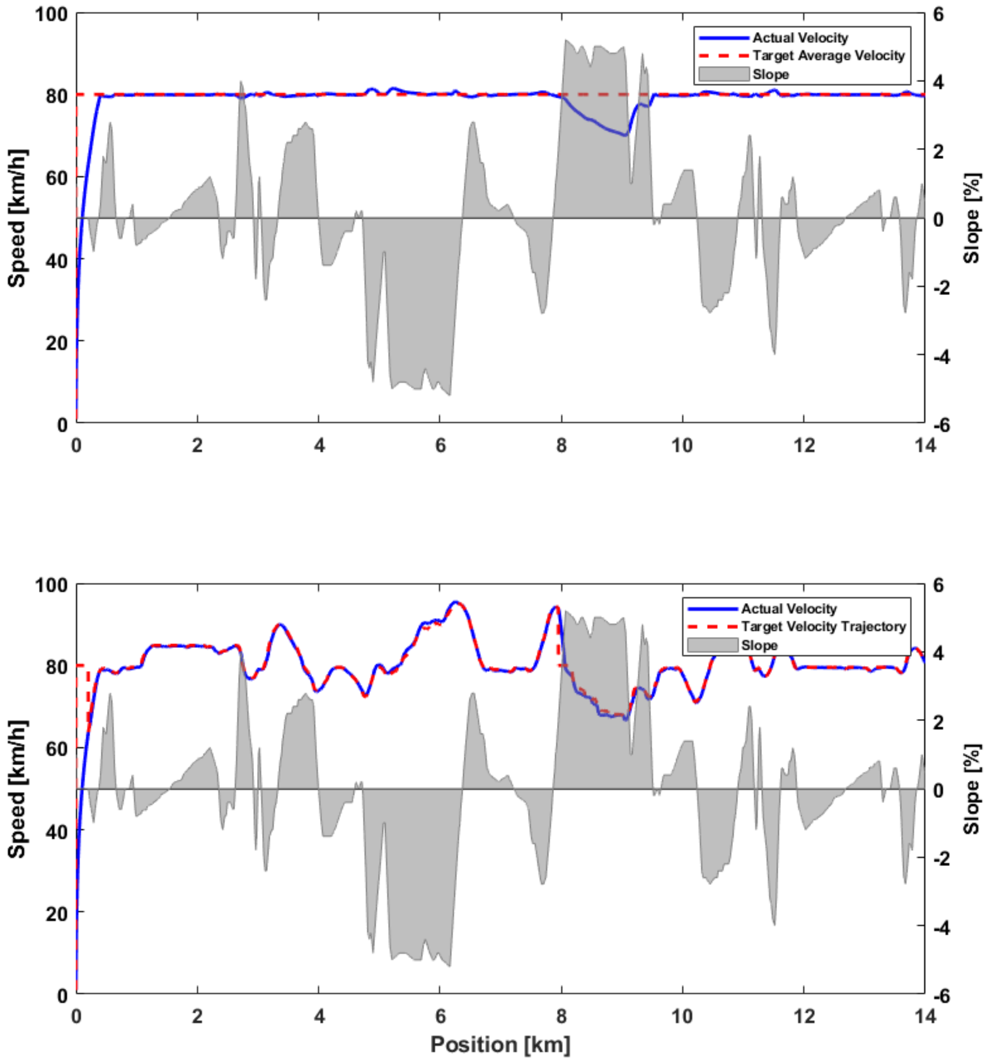

Figure 4.5 : Comparison of velocity trajectories for optimization active and inactive cases @80km/h target average speed in route profile 1

In Figure 4.5, testing was performed for an average velocity target of 80 km/h. In this test, the optimization produced the most efficient speed profile within the determined limits. In this test, unlike the others, in the constant speed case where the optimization is not active, the target speed cannot be reached due to the slope, although all available power is used between 8-10 km. In the case where the optimization is active, such a situation did not occur because both these speed states are inefficient and thanks to the reachable state generation function, those speed states are excluded from the optimization.

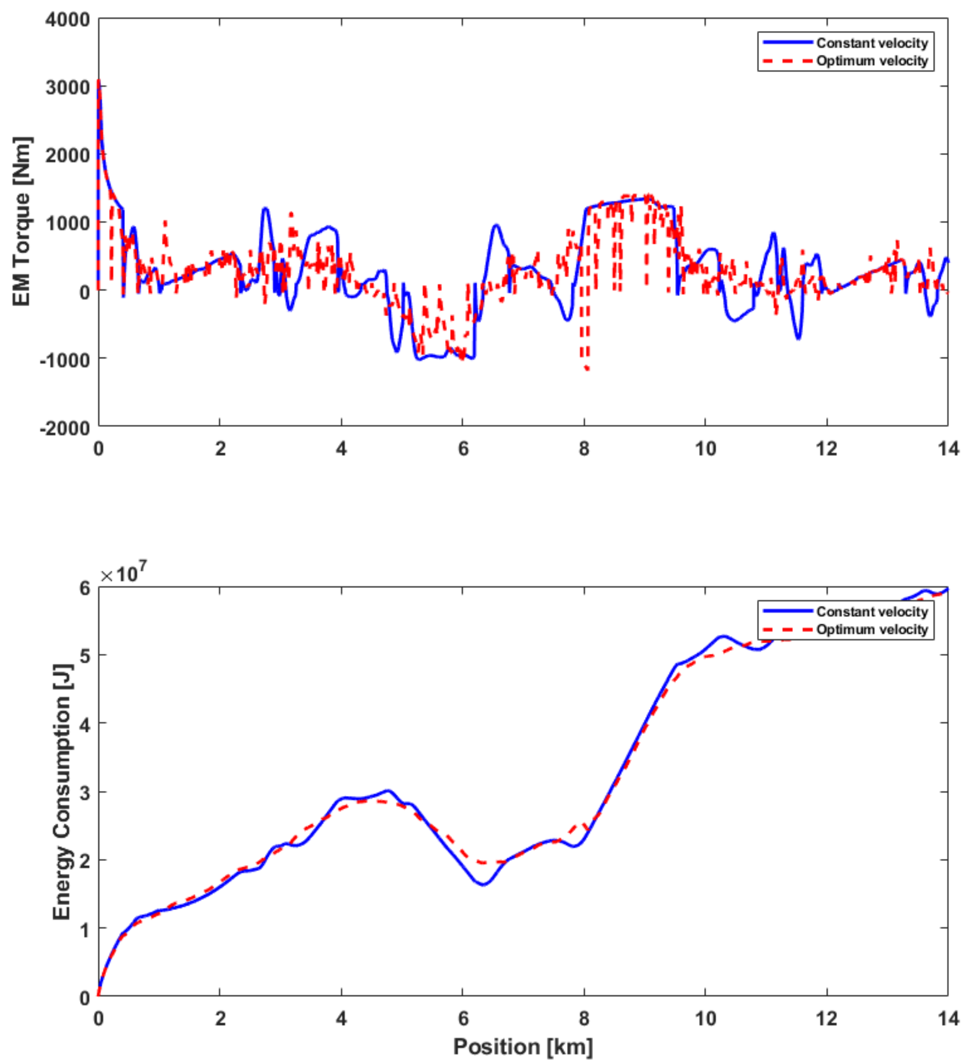

Figure 4.6 : Comparison of EM torque and energy consumption for optimization active and inactive cases @80km/h target average speed in route profile 1

In Figure 4.6, the electric motor torque and the total energy consumed from the battery are shown for the scenario of 80 km/h average speed this time. In the 80 km/h constant speed driving case, the total energy consumed by the vehicle is 6.3432 x 10^7 J. Arrival time is 647,38 s. On the other hand, following the speed profile calculated by the optimization, the total energy consumed by the vehicle is 6.1812 x 10^7 J. Arrival time is 639,59 s. The results show that the speed profile calculated by the optimization is

2.55% lower than the energy consumed 1.20% faster than by the vehicle driving at constant velocity.

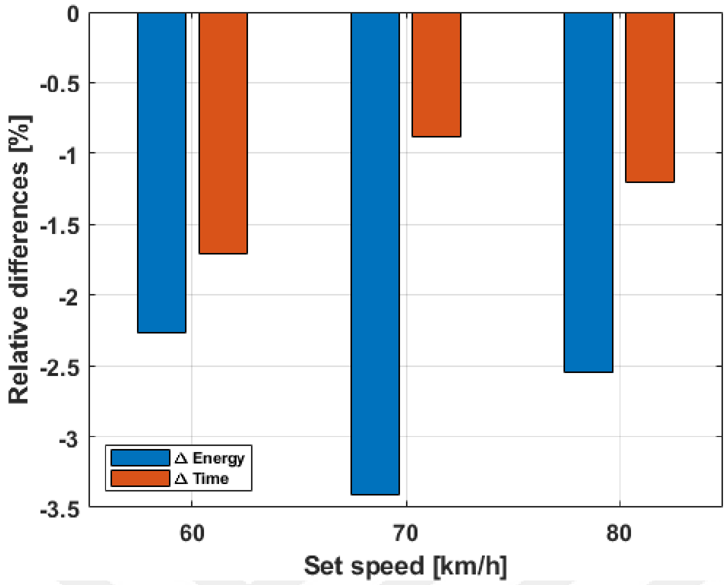

Figure 4.7 : Results for different target average speed in route profile 1

In Figure 4.7, the results obtained for 3 different average speed values, whose results are given separately, are shown together. In all three cases, it was observed that both energy and time were saved.

In Figure 4.8, 70 km/h average speed target was tested with different weights this time. It has been observed that the optimization gives more efficient results with different weights. In addition, thanks to adaptive beta calculation, different beta values required for different weights are calculated without the need for offline simulation / pre-calculated repeatedly.

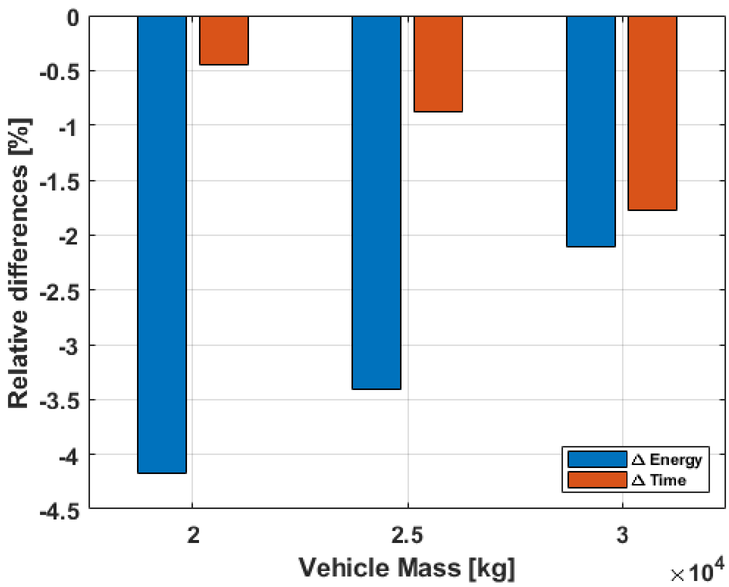

Figure 4.8 : Results for different weights of truck @70km/h target average speed in route profile 1

As defined in 3.3.1, the trade-off between energy demand and travel time is tuned by a weighting factor β. The biggest motivation for making β adaptive is the need for different beta values for different road profiles, different weights, and different target speeds. These values can normally be found with offline trials, but this approach is not suitable for real-time solutions and real use-cases.

In this study, it is calculated by using the average speed calculated according to the target arrival time taken from the driver and the instantaneous average speed. Figure 4.9 shows the test results for an average speed of 70 km/h. The velocity profile shown here with the blue solid line is the result produced by the optimization when using the constant beta. The constant β value has been determined by offline simulations until it coincides with the constant speed driving time of 70 km/h shown in Figure 4.3.

As seen in Figure 4.9, the calculated velocity profile is higher where beta is high. However, beta decreases as the average speed approach the target value. When the average speed exceeds the target, the beta value falls below the fixed value. At these times, the calculated velocity profile is lower. The energy value consumed with adaptive β is the same as in Figure 4.4. The energy consumed with constant β is 5.3208 x 10^7 J. The arrival time with fixed β time is 732,05 s.

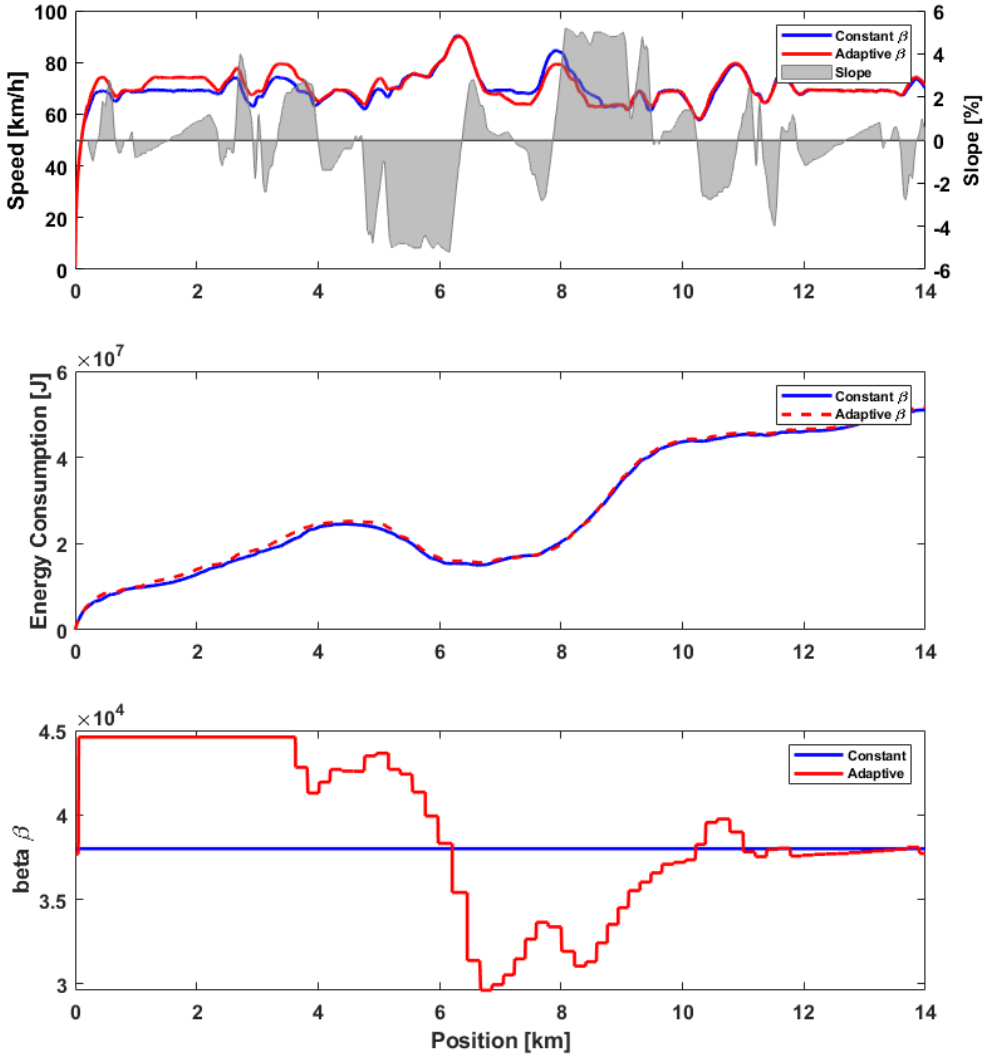

Figure 4.9 : Comparison of velocity trajectory and energy consumption for best time based tuned constant beta and adaptive beta @70kph in route profile 1

4.2 Route Profile 2 Results

All tests performed for road profile 1 were repeated in road profile 2. The main motivation for performing retests for the second route is to show that both the optimization algorithm and the adaptive β factor calculation yield successful results in different path profiles.

In Figure 4.10, the speed profile results obtained for road profile 2 based on 60 km/h average target speed are shared. The slope of route profile 2 is also shown as gray area in the graphics. Due to the uphill starting at approximately 1.5 km and the downhill starting after 2 km, the optimization target speed first decreased and then started to increase again with the start of the downhill. A similar situation is observed between 11.3 km and 12.5 km.

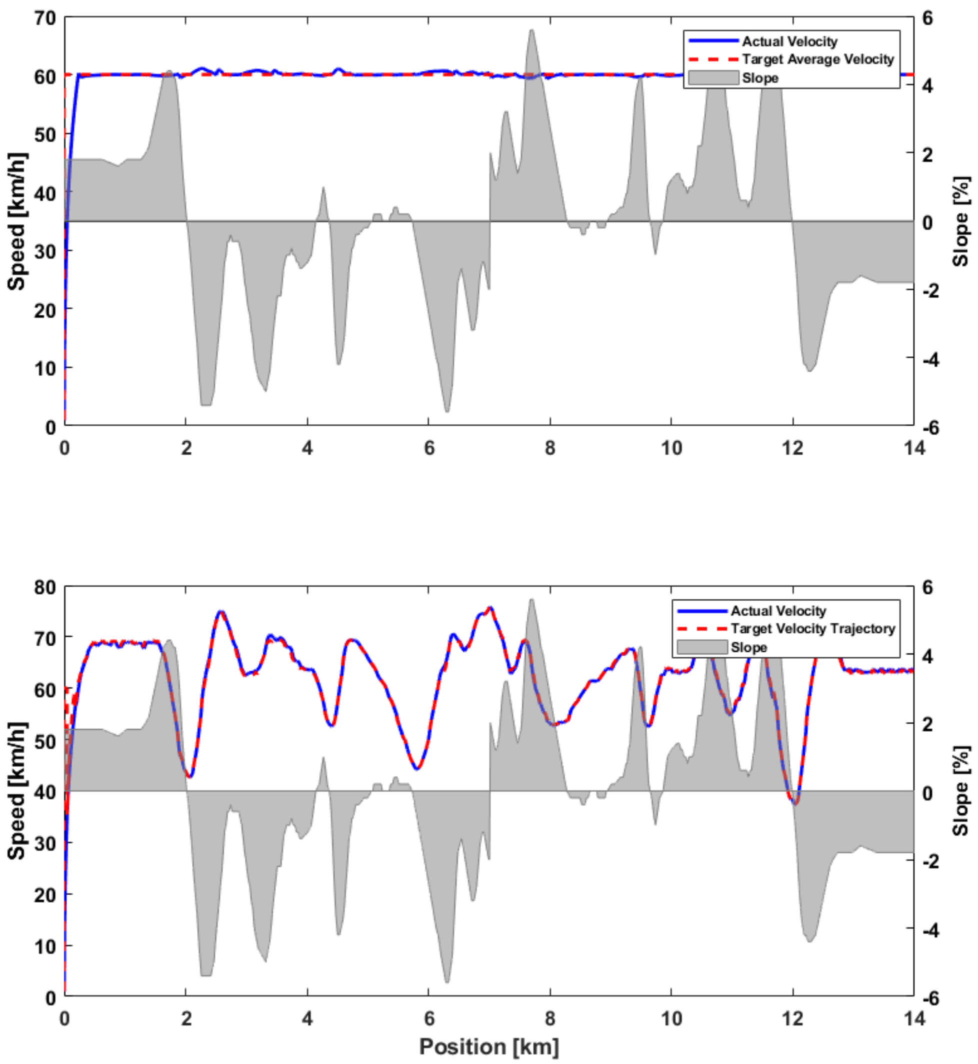

Figure 4.10 : Comparison of velocity trajectories for optimization active and inactive cases @60km/h target average speed in route profile 2

Electric motor torque and energy consumption values of 60 km/h average target speed performed in route profile 2 are shown in Figure 4.11. In the constant speed driving case, the total energy consumed by the vehicle is 4.9788 x 10^7 J. Arrival time is 849,35 s. On the other hand, following the speed profile calculated by the optimization, the total energy consumed by the vehicle is 4.9032 x 10^7 J. Arrival time is 832,27 s. The results show that the speed profile calculated by optimization is 1.52% lower than the energy consumed 2.01% faster than by the vehicle driving at constant velocity.

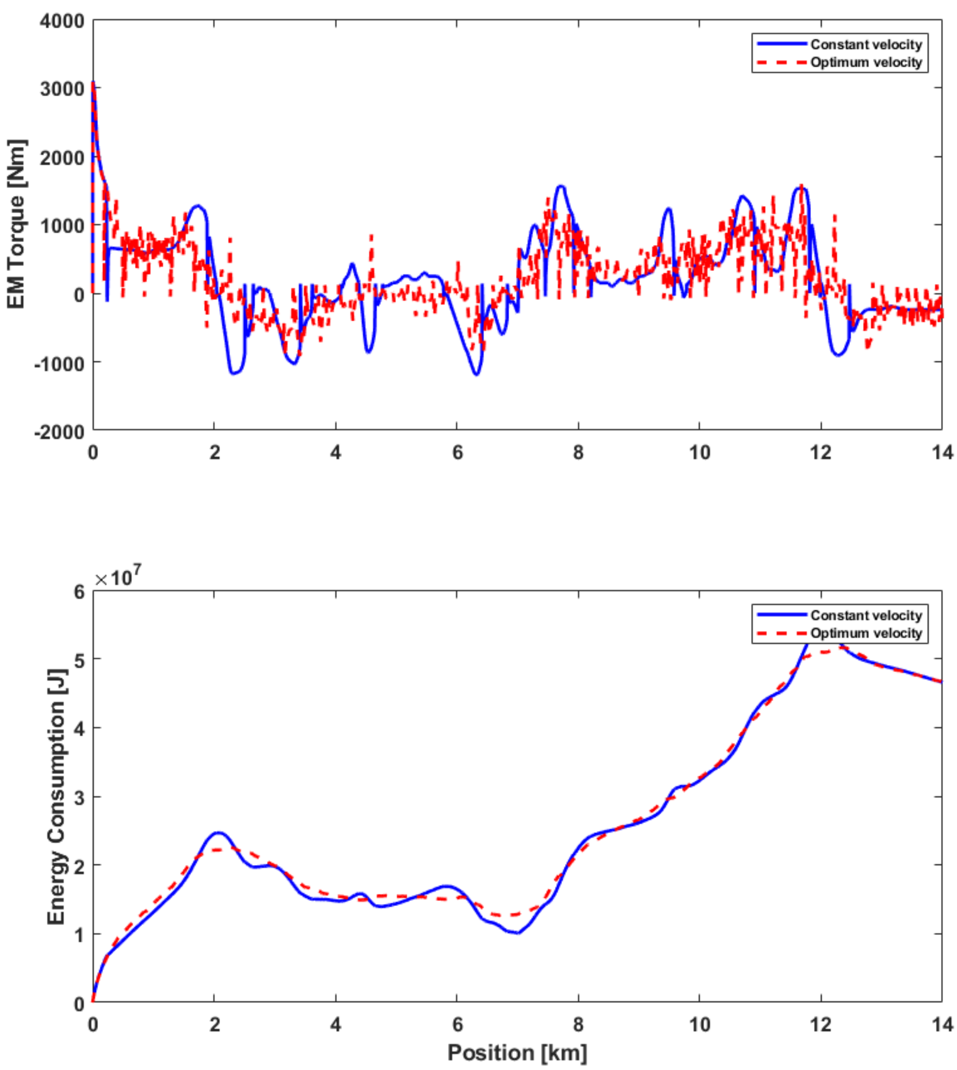

Figure 4.11 : Comparison of EM torque and energy consumption for optimization active and inactive cases @60km/h target average speed in route profile 2

The simulation results of 70 km/h average target velocity are also shown in Figure 4.12. The vehicle slows down before the negative slope starts at 5.5 km. It accelerates again with the start of the slope. Despite the acceleration of the vehicle, energy recovery is also provided with regenerative braking.

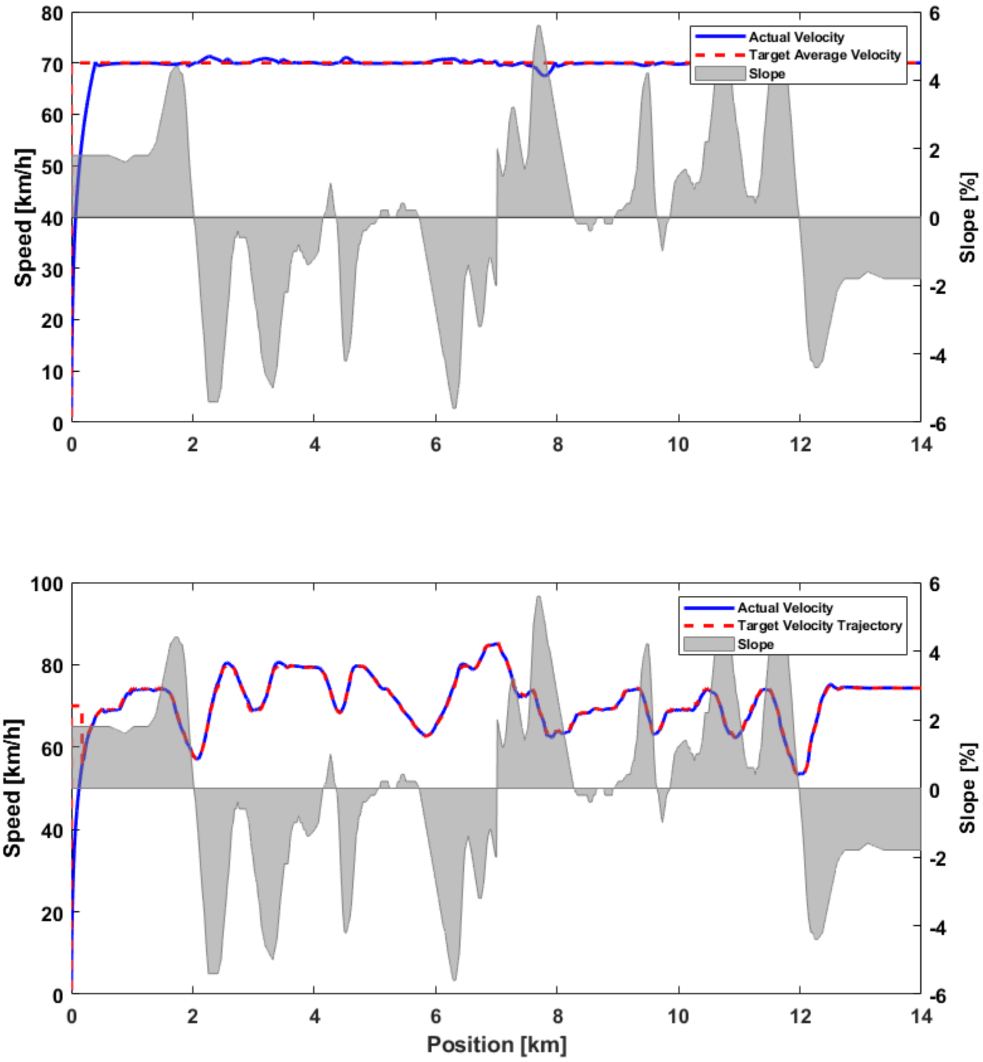

Figure 4.12 : Comparison of velocity trajectories for optimization active and inactive cases @70km/h target average speed in route profile 2

Electric motor torque and energy consumption values realized in route profile 2 of 70 km/h average target speed are shown in Figure 4.13. In the constant speed driving case, the total energy consumed by the vehicle is 5.6700 x 10^7 J. Arrival time is 732,87 s. On the other hand, following the speed profile calculated by the optimization, the total energy consumed by the vehicle is 5.5008 x 10^7 J. Arrival time is 723,53 s. The results show that the speed profile calculated by optimization is 2.98% lower than the energy consumed 1.27% faster than by the vehicle driving at constant velocity.

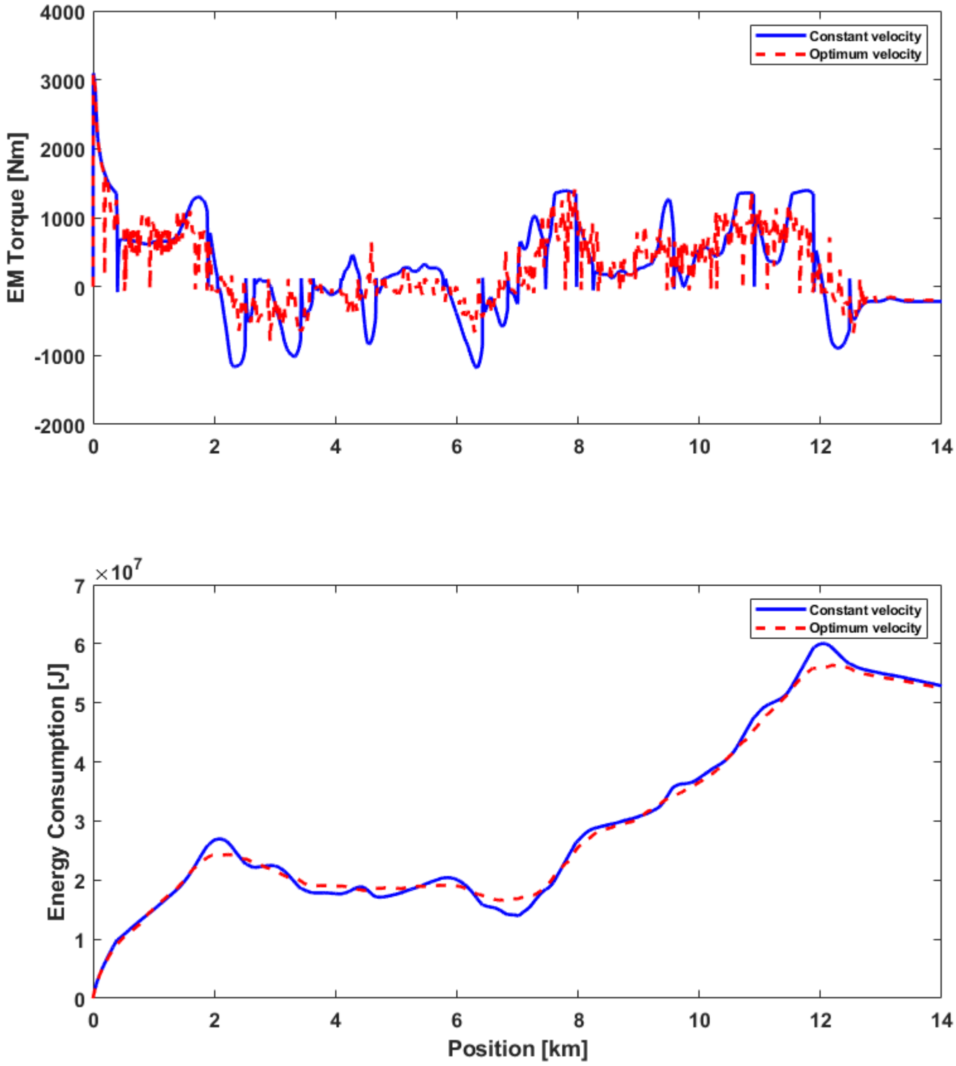

Figure 4.13 : Comparison of EM torque and energy consumption for optimization active and inactive cases @70km/h target average speed in route profile 2

The latest results for route profile 2 with 80 km/h target average velocity are shown in Figure 4.14. As in route profile 1, in the case of 80 km/h constant speed driving, speed tracking cannot be done at some slope values because sufficient electric motor torque cannot be produced. In velocity trajectory produced by optimization, this is not the case since reachable velocity states are calculated as mentioned before.

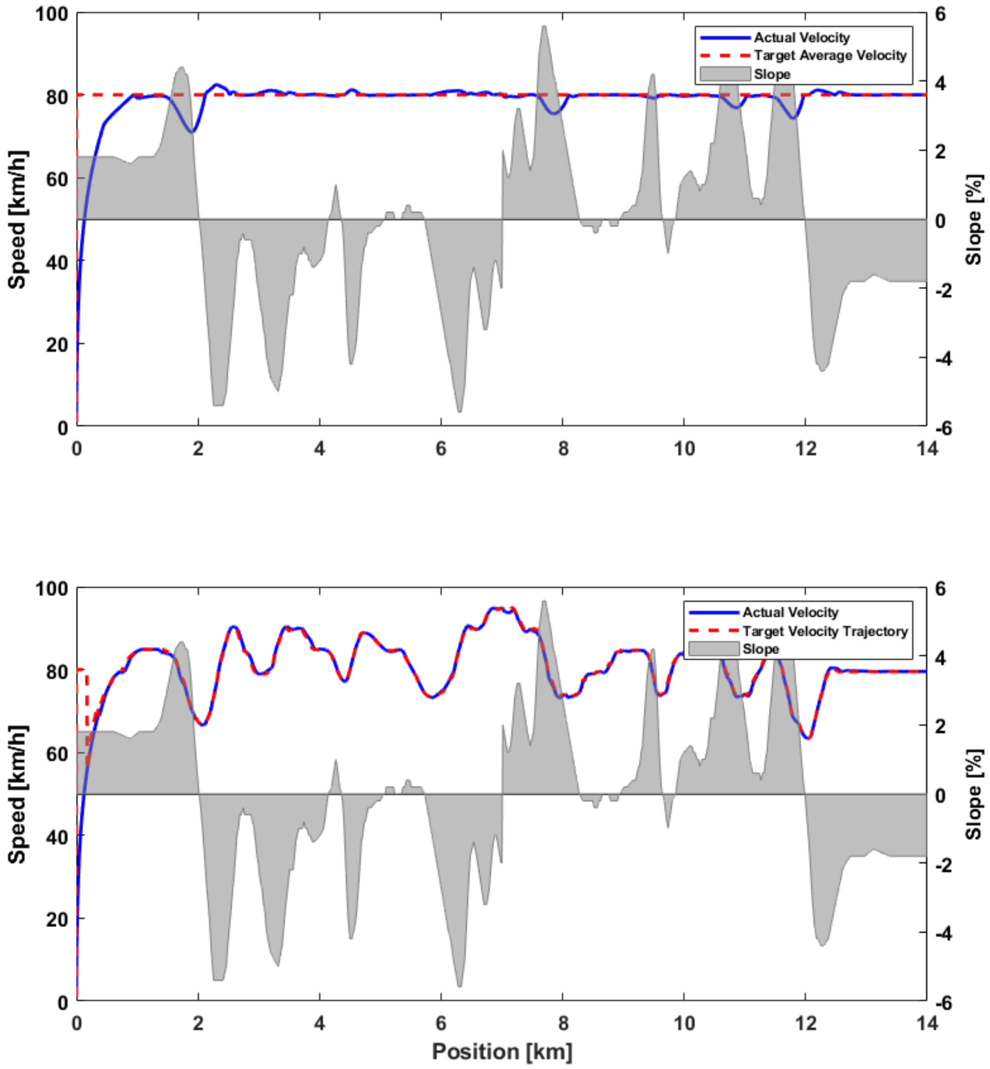

Figure 4.14 : Comparison of velocity trajectories for optimization active and inactive cases @80km/h target average speed in route profile 2

Electric motor torque and energy consumption values realized in route profile 2 of 80 km/h average target speed are shown in Figure 4.15. In the constant speed driving case, the total energy consumed by the vehicle is 6.4152 x 10^7 J. Arrival time is 649,77 s. On the other hand, following the speed profile calculated by the optimization, the total energy consumed by the vehicle is 6.1920 x 10^7 J. Arrival time is 638,92 s. The results show that the speed profile calculated by optimization is 3.48% lower than the energy consumed 1.67% faster than by the vehicle driving at constant velocity.

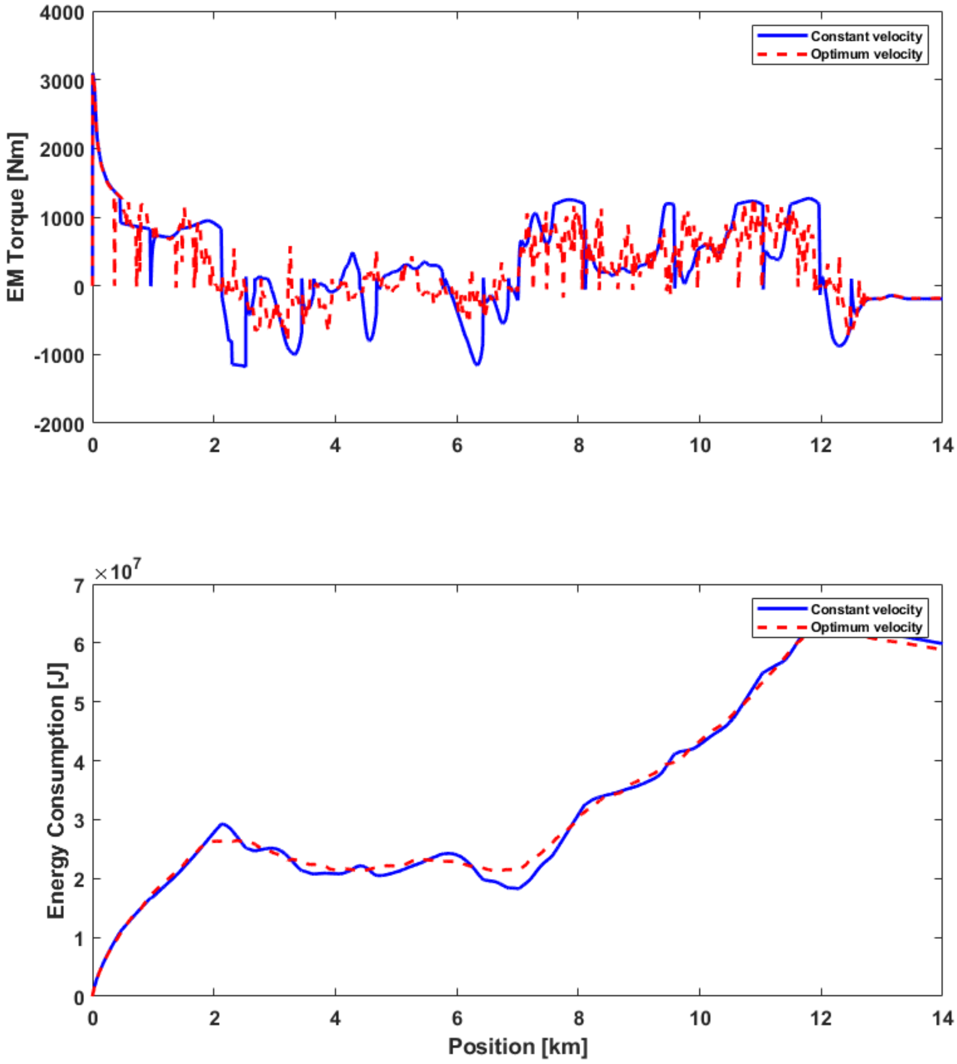

Figure 4.15 : Comparison of EM torque and energy consumption for optimization active and inactive cases @80km/h target average speed in route profile 2

In Figure 4.16, the results obtained for 3 different average speed values, which were tested separately for route profile 2, are shown together. As given in Figure 4.7, it was observed that both energy and time were saved in all three cases.

In Figure 4.17, as in Figure 4.8, 70 km/h average speed target was tested with different weights for route profile 2 this time. It has been observed that the optimization gives successful results in different weights in this route.

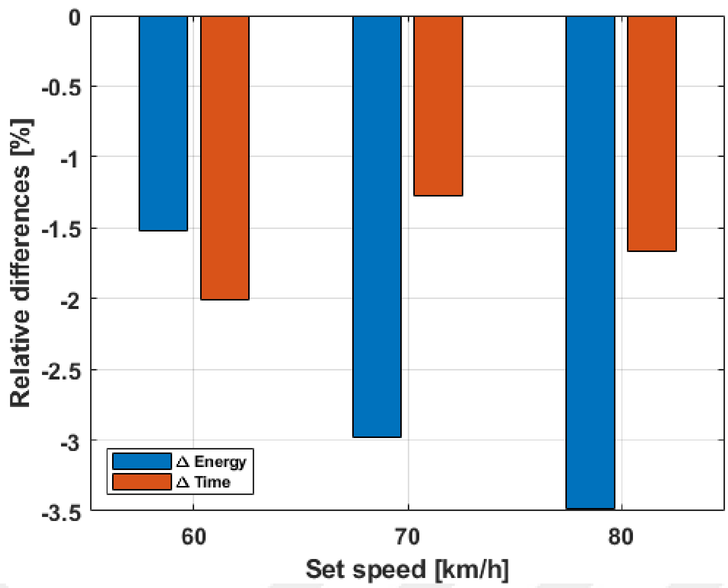

Figure 4.16 : Results for different target average speed in route profile 2

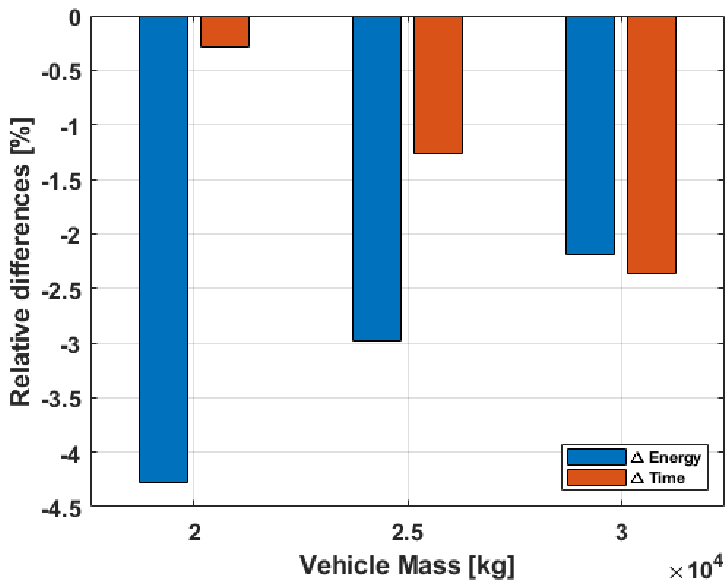

Figure 4.17 : Results for different weights of truck @70km/h target average speed in route profile 2

For route profile 2, as in Figure 4.9, the results of the constant beta of 70 km/h and the use of adaptive beta are given. The constant β value here has been determined by offline simulations until it coincides with the constant speed driving time of 70 km/h shown in Figure 4.12.

As seen in Figure 4.18, beta starts from a high value in the first place to reach the average speed. Where beta is high, the calculated velocity profile is higher. However, beta decreases as the average speed approach the target value. When the average speed exceeds the target, the beta value falls below the fixed value. At these times, the speed profile calculated with adaptive beta is lower. The energy value consumed with adaptive β is the same as in Figure 4.13. The energy consumed with constant β is 5.4468 x 10^7 J. The arrival time with fixed beta is 733,34 s.

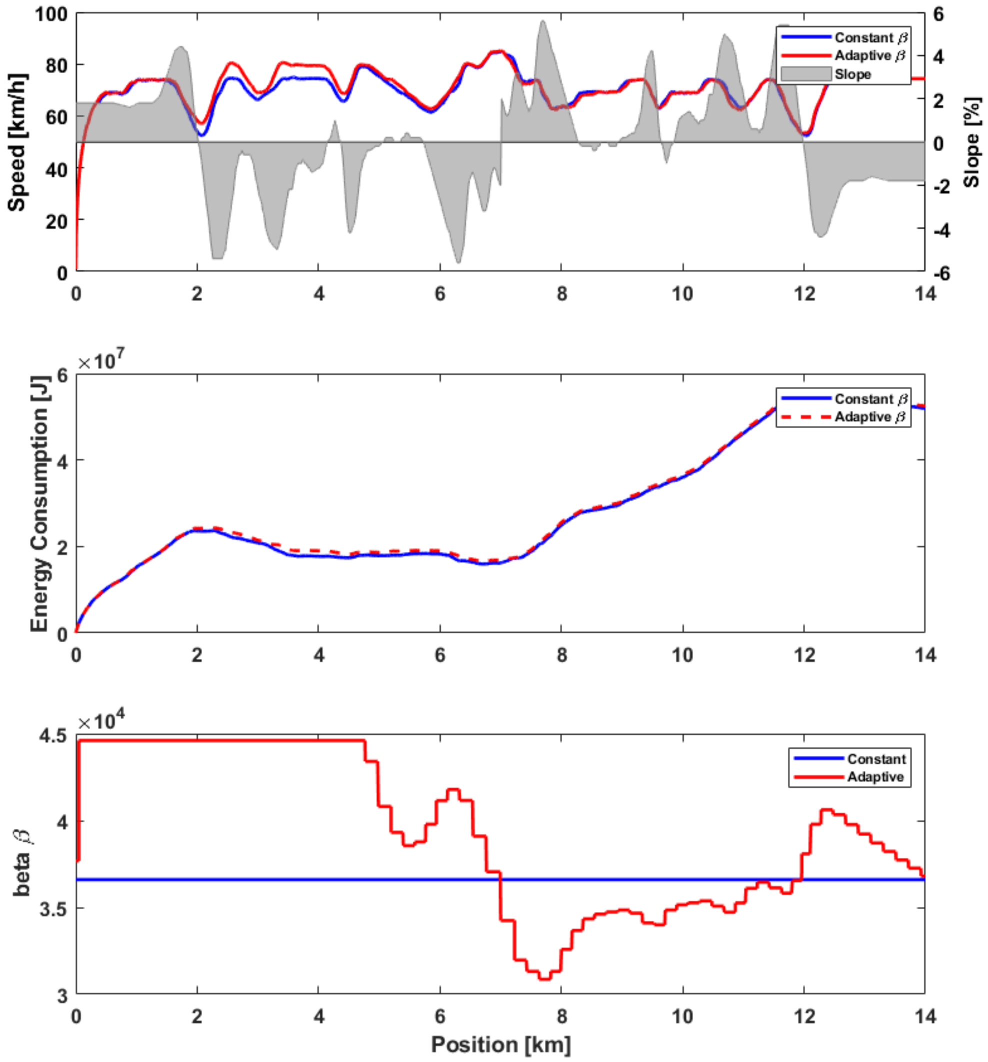

Figure 4.18 : Comparison of velocity trajectory and energy consumption for best time based tuned constant beta and adaptive beta @70kph in route profile 2

5. CONCLUSIONS AND FUTURE WORKS

In this thesis, a predictive vehicle speed trajectory optimization function has been developed for the truck with an all-electric powertrain topology. The main purpose of this function is to calculate the speed trajectory that will enable us to consume the least energy within the determined route and the targeted time. This study includes literature research, problem formulation, function development and implementation in Simulink and C-code environment, and evaluation of results for different routes.

The dynamic programming method, which gives a global optimum based on the Bellman optimality principle, has been chosen as a solution to this problem. Numerically, the method, which is solved iteratively from reverse to beginning, solves the problem in time and state space dimension by decomposing, while system and control constraints are defined through penalty functions added to the cost function.

The one-state formulation of the DP problem is still selected because of its significant utility in real-time application versus the two-state formulation, although global optimality is not guaranteed.

The selection of some optimization parameters directly affects the optimization performance and computational cost. In the literature, especially for time and energy consumption trade-offs, parameterization is made in a route-based fixed or adaptive way over the traffic light durations for city driving. In this study, a different adaptive beta calculation has been developed that enables the targeted route to be reached within the targeted time by consuming less energy.

The developed optimization function was investigated for two different road profiles for different fixed target speed values and different weights. In the simulations, it has been observed that the energy consumption is saved up to 4% compared to driving with constant speed, and the completion time of the route is reduced up to 2.5%. It is obvious that the energy savings will be even more if driving time at the constant speed is equalized with the driving time at the optimum speed by adapting to the constant speed value to match the time.

In future studies, it is aimed to realize this function on a vehicle. Solutions will also be sought for problems that will directly affect the optimization, such as precision and communication latency, which will occur during implementation. In addition, its contribution to energy consumption for different vehicle groups such as passenger cars and electric race cars will be examined.

REFERENCES

[1] McKinsey, "Disruptive trends that will transform the auto industry," January 2016. [Online].

[2] A. B. Schwarzkopf and R. B. Leipnik, "Control of highway vehicles for minimum fuel consumption over varying terrain," 1977.

[3] Z. Ye, *Predictive Vehicle Speed Trajectory Optimization,* Aachen, 2019.

[4] F. Lattemann, "The Predictive Cruise Control – A System to Reduce Fuel Consumption of Heavy Duty Trucks," *SAE transactions,* pp. vol. 113, no. 724, pp. 139–146, 2004.

[5] E. Hellström, "A Real-Time Fuel-Optimal Cruise Controller for Heavy Trucks Using Road Topography Information," *SAE International,* pp. vol. 2006, no. 724, 2006.

[6] W. Huang, "Using 3D road geometry to optimize heavy truck fuel efficiency," in *11th International IEEE Conference on Intelligent Transportation Systems,* 2008.

[7] A. Fröberg and L. Nielsen, "OPTIMAL FUEL AND GEAR RATIO CONTROL FOR HEAVY TRUCKS WITH PIECE WISE AFFINE ENGINE CHARACTERISTICS," *IFAC Proc. Vol.,* pp. vol. 40, no. 10, pp. 335–342, 2007.

[8] L. Johannesson, "Predictive energy management of hybrid long-haul trucks," *Control Eng. Pract,* pp. vol. 41, pp. 83–97, 2015.

[9] T. v. Keulen, "Optimal Energy Management in Hybrid Electric Trucks Using Route Information," *Oil Gas Sci. Technol. – Rev. l'Institut Français du Pétrole,* pp. vol. 65, no. 1, pp. 103–113, 2010.

[10] E. Hellström, *Look-ahead Control of Heavy Vehicles,* Linköping: Dept. Elect. Eng., Linköping University, 2010.

[11] E. Hellström, "Look-ahead control for heavy trucks to minimize trip time and fuel consumption," *IFAC Proc. Vol.,* pp. vol. 5, no. PART 1, pp. 439–446, 2007.

[12] E. Hellström, "Design of an efficient algorithm for fuel-optimal look-ahead control," *Control Eng. Pract.,* pp. vol. 18, no. 11, pp. 1318–1327, 2010.

[13] M. Frego, *Numerical Methods for Optimal Control Problems with application to autonomous vehicles,* University of Trento, 2014.

[14] H. Waschl, "Optimization and Optimal Control in Automotive Systems," 2014.

[15] E. Ozatay, "Analytical Solution to the Minimum Fuel Consumption Optimization Problem With the Existence of a Traffic Light," in *2012 5th Annual Dynamic Systems and Control Conference joint with the JSME 2012 11th Motion and Vibration Conference,* 2012.

[16] J. N. Hooker, "Optimal driving for single-vehicle fuel economy," *Transp. Res. Part A Gen,* pp. vol. 22, no. 3, pp. 183–201, 1988.

[17] R. E. Bellman, Dynamic Programming, Princeton: Princeton University, 1957.

[18] S. Rao, Engineering Optimization, Hoboken, NJ: John Wiley & Sons, 2009.

[19] D. Bertsekas, Dynamic Programming and Optimal Control, Nashua, NH: Athena Scientific , 2005.

[20] M. Papageorgiou, Optimierung, Berlin: Springer Vieweg, 2015.

[21] Chan-Chiao Lin and Jun-Mo Kang and Grizzle, J.W. and Huei Peng, "Energy management strategy for a parallel hybrid electric truck," in *Proceedings of the 2001 American Control Conference*, 2001.

[22] Yang Bin, Y. Li, Q. Gong and Z. Peng, "Multi-information integrated trip specific optimal power management for plug-in hybrid electric vehicles," in *2009 American Control Conference*, 2009.

[23] Qiuming Gong, Yaoyu Li and Zhong-Ren Peng, "Computationally efficient optimal power management for plug-in hybrid electric vehicles based on spatial-domain two-scale dynamic programming," in *2008 IEEE International Conference on Vehicular Electronics and Safety*, 2008.

[24] L. Johannesson, "Predictive energy management of hybrid long-haul trucks," *Control Eng. Pract*, pp. vol. 41, pp. 83–97, 2015.

[25] Wahl, Hans-Georg and Bauer, Kai-Lukas and Gauterin, Frank and Holzäpfel, Marc, "A real-time capable enhanced dynamic programming approach for predictive optimal cruise control in hybrid electric vehicles," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, 2013.

[26] Padilla, G. P. and Weiland, S. and Donkers, M. C. F., "A Global Optimal Solution to the Eco-Driving Problem," *IEEE Control Systems Letters*, vol. 2, no. 4, pp. 599-604, 2018.

[27] Thorsten Plum and Marius Wegener and Markus Eisenbarth and Ziqi Ye and Konstantin Etzold and Stefan Pischinger and Jakob Andert, "A simulation-based case study for powertrain efficiency improvement by automated driving functions," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 233, no. 5, pp. 1320-1330, 2019.

[28] Kutter, S. and Bäker, B., "An iterative algorithm for the global optimal predictive control of hybrid electric vehicles," in *2011 IEEE Vehicle Power and Propulsion Conference*, 2011.

[29] Wahl, Hans-Georg and Gauterin, Frank, "An iterative dynamic programming approach for the global optimal control of hybrid electric vehicles under real-time constraints," in *2013 IEEE Intelligent Vehicles Symposium (IV)*, 2013.

[30] Wahl, Hans-Georg and Holzäpfel, Marc and Gauterin, Frank, "Approximate dynamic programming methods applied to far trajectory planning in optimal control," in *2014 IEEE Intelligent Vehicles Symposium Proceedings*, 2014.

[31] E. Ozatay, *Driving Profile Optimization for Everyday Driving*, Ohio, 2014.

[32] M. Stapelbroek, J. Pfluger, J. Abel, J. Ogrzewalla, Z. Ye, and T. Plum, "Dynamic Speed Trajectory Optimization for Achieving Real World Optimum Energy Consumption," in *FISITA 2016 World Automotive Congress*, 2016.

[33] Lin, Xiaohai and Görges, Daniel and Liu, Steven, "Eco-driving assistance system for electric vehicles based on speed profile optimization," in *2014 IEEE Conference on Control Applications (CCA)*, 2014.

[34] Biswas, Atriya and Emadi, Ali, "Energy Management Systems for Electrified Powertrains: State-of-the-Art Review and Future Trends," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 7, pp. 6453-6467, 2019.

[35] Dib, Wissam and Serrao, Lorenzo and Sciarretta, Antonio, "Optimal control to minimize trip time and energy consumption in electric vehicles," in *2011 IEEE Vehicle Power and Propulsion Conference*, 2011.

[36] Uebel, Stephan and Murgovski, Nikolce and Tempelhahn, Conny and Bäker, Bernard, "Optimal Energy Management and Velocity Control of Hybrid Electric Vehicles," *IEEE Transactions on Vehicular Technology,* vol. 67, no. 1, pp. 327-337, 2018.

[37] Plianos, A., Jokela, T., and Hancock, M., "Predictive Energy Optimization for Connected and Automated HEVs," *SAE Technical Paper 2018-01-1179,* p. 15, 2018.

[38] Ye, Ziqi and Li, Kailai and Stapelbroek, Michael and Savelsberg, Rene and Günther, Marco and Pischinger, Stefan, "Variable Step-Size Discrete Dynamic Programming for Vehicle Speed Trajectory Optimization," *IEEE Transactions on Intelligent Transportation Systems,* vol. 20, no. 2, pp. 476-484, 2019.

[39] Khalik, Z. and Padilla, G.P. and Romijn, T.C.J. and Donkers, M.C.F., "Vehicle Energy Management with Ecodriving: A Sequential Quadratic Programming Approach with Dual Decomposition," in *2018 Annual American Control Conference (ACC),* 2018.

[40] Engin Ozatay and Umit Ozguner and Dimitar Filev, "Velocity profile optimization of on road vehicles: Pontryagin's Maximum Principle based approach," *Control Engineering Practice,* vol. 61, pp. 244-254, 2017.

[41] Uebel, Stephan and Murgovski, Nikolce and Bäker, Bernard and Sjöberg, Jonas, "A Two-Level MPC for Energy Management Including Velocity Control of Hybrid Electric Vehicles," *IEEE Transactions on Vehicular Technology,* vol. 68, no. 6, pp. 5494-5505, 2019.

[42] Guo, L., Gao, B., Li, Y, "A fast algorithm for nonlinear model predictive control applied to HEV energy management systems," *Science China Information Sciences*, vol. 60, no. 9, 2017.

[43] T.Schwickart,H.Voos,J.Hadji-Minaglou, and M. Darouach, "A Novel model-predictive cruise controller for electric vehicles and energy-efficient driving," in *2014 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, 2014.

[44] Schwickart, Tim and Voos, Holger and Darouach, Mohamed, "A real-time implementable model-predictive cruise controller for electric vehicles and energy-efficient driving," in *2014 IEEE Conference on Control Applications (CCA)*, 2014.

[45] T. Ohtsuka, "A tutorial on C/GMRES and automatic code generation for nonlinear model predictive control," in *2015 European Control Conference (ECC)*, 2015.

[46] Schwickart, Tim and Voos, Holger and Hadji-Minaglou, Jean-Régis and Darouach, Mohamed, "An efficient nonlinear model-predictive eco-cruise control for electric vehicles," in *2013 11th IEEE International Conference on Industrial Informatics (INDIN)*, 2013.

[47] Nikolce Murgovski and Lars Johannesson and Jonas Sjöberg and Bo Egardt, "Component sizing of a plug-in hybrid electric powertrain via convex optimization," *Mechatronics*, vol. 22, no. 1, pp. 106-120, 2012.

[48] Gunter Heppeler and Marcus Sonntag and Oliver Sawodny, "Fuel Efficiency Analysis for Simultaneous Optimization of the Velocity Trajectory and the Energy Management in Hybrid Electric Vehicles," *IFAC Proceedings Volumes*, vol. 47, no. 6612-6617, 2014.

[49] Frank Lattemann, Konstantin Neiss, Stephan Terwen, Thomas Connolly, "The Predictive Cruise Control – A System to Reduce Fuel Consumption of Heavy Duty Trucks," in *SAE Commercial Vehicle Engineering Congress & Exhibition*, 2004.

[50] Sciarretta, Antonio and De Nunzio, Giovanni and Ojeda, Luis Leon, "Optimal Ecodriving Control: Energy-Efficient Driving of Road Vehicles as an Optimal Control Problem," *IEEE Control Systems Magazine*, vol. 35, pp. 71-90, 2015.

[51] T. Radke, *Energieoptimale Längsführung von Kraftfahrzeugen durch Einsatz vorausschauender Fahrstrategien*, 2013.

[52] H. G. Wahl, *Optimale Regelung eines prädiktiven Energiemanagements von Hybridfahrzeugen*, 2015.

CURRICULUM VITAE

Name Surname **:** Abdullah KIZIL

EDUCATION **:**

- **B.Sc.** **:** 2016, Yıldız Technical University, Machine Engineering Faculty, Mechatronics Engineering Department
- **B.Sc.** **:** 2017, Yıldız Technical University, Electrical and Electronics Engineering Faculty, Electronics and Communication Engineering Department(Double Major)

PROFESSIONAL EXPERIENCE AND REWARDS:

- 2016 - 2017 AVL Turkey – System Engineer.
- 2017 - 2019 FEV Turkey – Electric Vehicle System Engineering
- 2019 - FEV Turkey – Team Leader xCU & Power Electronics

OTHER PUBLICATIONS, PRESENTATIONS AND PATENTS:

- **A.Kızıl,** S.Koçak, A.C.Kutlucan, C.Doğan, V.Sezer, "Design and Implementation of Autonomous Parallel and Vertical Parking Mobile Vehicle", 6th International Conference on Control Engineering & Information Technology 2018.

- **A.Kızıl,** B.Vural, "Yakıt Hücreli Araçlarda Enerji Yönetimi İçin Bulanık Mantık Tabanlı Kontrol", Otomatik Kontrol Tük Milli Komitesi Ulusal Toplantısı 2016. Eskişehir Osmangazi Üniversitesi, (pp. 261-266).