ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE ENGINEERING AND TECHNOLOGY

CONTROLLER PARAMETER TUNING FOR AN ADAPTIVE CRUISE CONTROL SYSTEM BASED ON PARTICLE SWARM OPTIMIZATION APPROACH

M.Sc. THESIS

Erhan ÖZKAYA

Department of Mechanical Engineering

Automotive Programme

DECEMBER 2019

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE ENGINEERING AND TECHNOLOGY

CONTROLLER PARAMETER TUNING FOR AN ADAPTIVE CRUISE CONTROL SYSTEM BASED ON PARTICLE SWARM OPTIMIZATION APPROACH

M.Sc. THESIS

Erhan ÖZKAYA
(503171704)

Department of Mechanical Engineering

Automotive Programme

Thesis Advisor: Asst.Prof.Dr. Hikmet ARSLAN

DECEMBER 2019

ISTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

ADAPTİF SEYİR SİSTEMİ KONTROLCÜSÜ PARAMETRELERİNİN PARÇACIK SÜRÜ OPTİMİZASYONU YÖNTEMİ İLE BELİRLENMESİ

YÜKSEK LİSANS TEZİ

Erhan ÖZKAYA
(503171704)

Makine Mühendisliği Anabilim Dalı

Otomotiv Programı

Tez Danışmanı: Dr. Öğr. Üyesi Hikmet ARSLAN

ARALIK 2019

Erhan Özkaya, a M.Sc. student of İTU Graduate School of Science Engineering and Technology student ID 503171704, successfully defended the thesis/dissertation entitled "CONTROLLER PARAMETER TUNING FOR AN ADAPTIVE CRUISE CONTROL SYSTEM BASED ON PARTICLE SWARM OPTIMIZATION APPROACH", which he prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor : **Asst.Prof.Dr. Hikmet ARSLAN**
Istanbul Technical University

Jury Members : **Asst.Prof.Dr. Osman Taha ŞEN** ....
Istanbul Technical University

Assoc. Prof. Dr. Tarkan SANDALCI ....
Yıldız Technical University

Date of Submission : 15 November 2019
Date of Defense : 12 December 2019

FOREWORD

Throughout the history, transportation is one of the important fact for human being. With rapidly enhancing technology, automotive industry is changing day by day. Nowadays, all manufacturers and newly established start-up companies try to keep pace with this enormous change. They try to implement artificial intelligence into the automobiles and autonomous vehicles are being sighted on horizon. Adaptive and conventional cruise controller are very important start points for this development. This thesis mainly surmounts design and optimization of adaptive and conventional cruise controller with a crucial optimization method. Simulations are conducted and results are interpreted.

During the writing of this thesis and for my whole life, firstly I would like to thank my family, they always support and encourage me to finalise this thesis. Then, I have to thank my darling and friends for their all limitless support. Also for their great contributions, I would like to thank Asst.Prof.Dr. Hikmet ARSLAN, Asst.Prof.Dr. Osman Taha ŞEN and my colleagues in AVL Turkey.

November 2019 | Erhan ÖZKAYA
(Mechanical Engineer)

TABLE OF CONTENTS

Page

ABBREVIATIONS

ABS	: Anti-lock braking system
ACC	: Adaptive cruise control
ACO	: Ant colony optimization
ADAS	: Advanced driver assistance system
ALO	: Antlion optimization
CCC	: Conventional cruise control
DAS	: Driver assistance system
ESC	: Electronic stability control
ESP	: Electronic stability program
GA	: Genetic algorithm
GNSS	: Global Navigation Satellite System
HIL	: Hardware in the loop
HMI	: Human machine interface
IAE	: Integrated absolute error
MPC	: Model predictive control
MSE	: Mean square error
NoI	: Number of iterations
NoS	: Number of swarm particles
OEM	: Original equipment manufacturer
PD	: Proportional-derivative
PID	: Proportional-integral-derivative
PSO	: Particle swarm optimization
RMS	: Root mean square
RMSE	: Root mean square error
SAE	: Society of Automotive Engineers
TCS	: Traction control system
ZN	: Ziegler-Nichols

SYMBOLS

m	: Mass of vehicle
g	: Gravitational acceleration
θ	: Grading angle
$\mathbf{F_{Accel}}$	: Acceleration force
a	: Longitudinal acceleration
$\mathbf{P_{Cr}}$	: Critical period
$\mathbf{K_{Cr}}$	: Critical proportional gain
$\mathbf{T_i}$	: Time constant of integral controller
$\mathbf{T_d}$	: Time constant of derivative controller
$\mathbf{w_f}$	: Frequency
G(s)	: Transfer function
$\mathbf{x_i^s}$	: Position of each swarm particles at i^{th} iteration
$\mathbf{V_i^s}$	: Velocity of each swarm particles at i^{th} iteration
$\mathbf{n_{var}}$	: Number of variables
$\mathbf{n_{swarm}}$	: Number of swarms
$\mathbf{n_{iteration}}$	: Number of iterations
i	: Iteration indice
$\mathbf{Pbest_s^{pos}}$	: Personal best position for each swarm
$\mathbf{Gbest^{pos}}$	: Global best position
$\mathbf{c_{1,2}}$	: Acceleration constants
$\mathbf{w_h}$	: High inertia weight
$\mathbf{w_l}$	: Low inertia weight
$\mathbf{y_{act}}$	: Actual output of a system
$\mathbf{y_{des}}$	: Desired output of a system
$\mathbf{t_{tot}}$	: Total time
$\mathbf{T_{set}}$	: Time gap set by driver
$\mathbf{V_{Ego}}$	: Ego vehicle speed
$\mathbf{V_{set}}$	: Vehicle speed set by driver
$\mathbf{V_{Lead}}$	: Lead vehicle speed
$\mathbf{d_{des}}$	: Desired relative distance between ego and lead vehicle

$\mathbf{d_{act}}$	: Actual relative distance between ego and lead vehicle
$\mathbf{K_v, K_t}$	: Gains for speed and distance errors
$\mathbf{K_p, K_i, K_d}$	: Proportional, integral, derivative controller gains
$\mathbf{T_{wheel}}$	: Wheel torque
$\mathbf{T_{wheel,des}}$	: Desired wheel torque
$\mathbf{T_{Engine}}$	: Engine torque
$\mathbf{GR_i}$	: Gear ratio for corresponding gear
FDR	: Final drive ratio
$\boldsymbol{\eta_{Dr}}$	: Driveline efficiency
$\boldsymbol{\eta_{GB}}$	: Gearbox efficiency
$\mathbf{r}$	: Wheel radius
$\mathbf{F_{wheel}}$	: Wheel force
$\mathbf{s}$	: Longitudinal slip
$\mathbf{F_x}$	: Longitudinal tire force
$\mathbf{A_f}$	: Frontal projection area of vehicle
$\mathbf{C_d}$	: Air drag coefficient
$\mathbf{w_{wheel}}$	: Wheel speed
A, B, C, D	: Pajecka tire model constants

LIST OF TABLES

Page

LIST OF FIGURES

Page

CONTROLLER PARAMETER TUNING FOR AN ADAPTIVE CRUISE CONTROL SYSTEM BASED ON PARTICLE SWARM OPTIMIZATION APPROACH

SUMMARY

As the day goes on, there are some new trends which aim to product higher automated, safer and more comfortable vehicles in automotive industry. In order to achieve this aim, original equipment manufacturers (OEM) try to design higher automated, advanced driver assistance systems (ADAS) and implement to their vehicles. Implementation started with active safety features such as anti-lock braking system (ABS), electronic stability program (ESP) etc. which makes the vehicle safer in terms of vehicle dynamics and behaviours. Then this evolves to implementation of driver assistance systems which provide more driving comfort with safety nowadays. Moreover, developing ADAS features increase the vehicle safety, enhance driving convenience, increase fuel efficiency, reduce traffic accidents, improve pedestrian safety. But it is known that still there are many questions and discussions related with safety. Any sensor or algorithm failure can cause hazardous effect unless controller algorithms are robust and inclusive for all use cases in environment. Also with developing features and applications, legislations and standards are started to be established. Thus, in order to meet the requirements and provide safety, algorithms should be more comprehensive and well-designed. As a result of this, the calibration or optimization of this algorithms is becoming a problem. At this point, automated optimization algorithms is able to handle this problem more unique.

This thesis includes the design of adaptive cruise controller (ACC) algorithm and optimization method for that algorithm to achieve the objectives which are defined. Main scope of this thesis is implementation of optimization algorithm and its results. Firstly, in order to generate realistic results and behaviours, longitudinal vehicle model is built on Matlab/Simulink platform. Then, created ACC and vehicle model are connected. Various scenarios are generated on AVL VSM vehicle dynamics simulation software to simulate the following vehicle for each controllers which are conventional cruise control (CCC) and ACC respectively. Finally, particle swarm optimization (PSO) algorithm is connected to the complete model to optimize the defined variables and results are interpreted according to requirements. Apart from main flow of thesis; in the first chapter purpose of thesis is briefly given and literature is reviewed, the second chapter gives detailed information about concept and controller methodologies of ACC, then longitudinal vehicle dynamic equations and Simulink model integration are presented. Conventional and current controller optimization methods are briefly demonstrated in the fourth chapter and PSO is reviewed with relevant samples. Then, in the fifth chapter, the whole system is tackled, integration of vehicle model, controller and optimization tool are shown with using defined scenarios. As a last chapter, conclusions are explained and further studies are suggested.

**ADAPTİF SEYİR SİSTEMİ
KONTROLCÜSÜ PARAMETRELERİNİN
PARÇACIK SÜRÜ OPTİMİZASYONU YÖNTEMİ İLE BELİRLENMESİ**

ÖZET

Otomotiv piyasasında gün geçtikte ve teknolojinin yüksek hızda gelişimiyle birlikte daha otonom, güvenli ve daha konforlu araçların üretilmesi amacı yeni bir trend haline gelmiştir. Bu amaca ulaşmak için araç üreticileri; daha yüksek otonom özelliklere sahip, daha otomatikleşmiş gelişmiş sürücü yardım sistemleri geliştirmeye ve bu sistemleri araçlarına entegre etmeye çalışmaktadır. Sürücü yardım sistemlerinin geçmişine bakacak olursak, aktif güvenlik önlemleri olarak 1990'lı yılların başlarında anti-lock braking system (ABS) ile otomotiv endüstrisine entegre olmaya başlamıştır. ABS'den daha sonra aracın stabilizasyonunu sağlamaya çalışan electronic stability program (ESP) aracı daha güvenli bir noktaya götüren bir sistem olarak entegre edilmeye başlanmıştır. Yıllarca süren gelişmeler ışığında bu sistemler aktif güvenlik amaçlarını koruyarak, günümüzde sürücü konforuna da olumlu etki yapmaya başlamışlardır.
Seyir sistemleri de bu etkiyi yaratan teknolojik gelişimin ilk üyeleridir. Seyir sistemleri ilk olarak, sürücü tarafından belirlenen hızda sabit olarak hızı tutmaya çalışan cruise control ve belirlenen bir hızda aktif olarak hızı limitleyen hız limitleyicisi ile araçlarda sunulmuştur. Daha sonrasında ilerleyen teknoloji ile birlikte cruise control seyir sistemi kendisini adaptif seyir sistemlerine evirmiştir. Bu sistemler bir menzil sensörü ile öndeki aracın hızını algılayıp, belirlenen kontrol metoduyla iki araç arasındaki mesafeyi, zamanı veya hız farkını belirli bir düzeyde tutacak şekilde arkadaki aracın öndeki aracı takip etmesine olanak sağlar. Menzil sensörü için farklı sensörler kullanılabilir. Bu sensörlerin başında radar gelmektedir. Adaptif seyir sistemlerinde en fazla kullanılan sensör olan radar haricinde, sistemde kamera, ultrasonik sensörler veya lidar kullanılmaktadır. Adaptif seyir sistemleri sonrasında da sürücü destek sistemleri teknolojik gelişimine devam etmektedir. Son yıllarda otonom araç trendi bunun açıkça bir kanıtı olarak görülebilir.
Destek sistemlerinden daha çok sürücüsüz araçlara yönelik çalışamalar gün geçtikte artmaktadır. Artan teknolojik gelişmelerle, bu sistemlerin yazılımlarının çok daha karmaşık ve her durumu kapsayıcı şekilde tasarlanma gerekliliği aşikardır. Gün geçtikçe yazılımlar daha kapsamlı hale geliyor ve doğal bir sonuç olarak yazılımları doğru çalıştırmak için gerekli kalibrasyon aktiviteleri de büyük önem kazanmaktadır. Ayrıca yüksek hızda gelişen teknolojiyle paralel olarak, birçok cevaplanmamış soru ve tartışma hala var olmaktadır. Herhangi bir sensör ya da yazılım yanlışı veya hatası çok kötü sonuçlar doğurabilir. Bu sebepten ötürü de yazılımların doğru şekilde kalibre ve optimize edilmesi hayati değer taşımaktadır. Bu sistemler için gerekli yasalar ve regülasyonlar da oluşturulmaya başlamıştır. Bu kadar kapsamlı ve güvenlik kritik bir durum için yazılımların optimizasyonu otomatik optimizasyon yazılımları ile, çok daha güvenilir ve kolay şekilde çözülebilir.

Bu çalışmada adaptif seyir sistemleri algoritmalarının tasarımı ve tasarlanmış olan kontrolcünün modern optimizasyon tekniklerinden biriyle optimize edilmesi anlatılmaktadır. Modern optimizasyon teknikleri bu tarz problem için oldukça faydalı metotlardır. Bu teknikler ile istenen amaç fonksiyonu kullanıcının istediği şekilde minimize edebilir, maksimize edebilir ya da istenen herhangi bir değere yakınsatılabilir. Ayrıca belirlenen amaç fonksiyonu tek bir parametrenin sonucu değil, birden fazla parametrenin ortak bir fonksiyonu şeklinde belirlenerek, aynı anda birden fazla istenen sonuç olan problemlerde kullanılabilir.
Bu tezin odaklandığı ana nokta optimizasyon algoritmasının sisteme entegrasyonu ve simülasyonların doğru şekilde tamamlanarak sonuçlanmasıdır. İlk olarak bu tezde, gerçekçi sonuçlar ve davranışlar görebilmek için boylamsal eksende araç tek bir kütle olarak Matlab/Simulink platformunda modellenmiştir. Aynı zamanda adaptif seyir sistemi yazılımı tasarlanmış ve bu tasarım detaylı bir şekilde anlatılmıştır. Tasarlanan adaptif seyir sistemi yazılımı ile birlikte oluşturulan boylamsal araç modeli yine aynı platformda birbirlerine entegre edilmiştir. Sistemde sadece kontrol edilen, adaptif seyir sistemine sahip olan, aracın boylamsal modeli mevcuttur. Herhangi bir sensör modeli yoktur ve bu çalışmadaki odak noktası değildir. Öndeki aracı simüle etmek için öndeki araca ait bir hız profili ortaya çıkarılmış ve dışarıdan sisteme beslenmiştir. Bu hız profili çok kapsamlı bir paket program olan AVL VSM tarafından oluşturulmuştur.

AVL VSM; araç hakkında tekerleklerden süspansiyona, araç dinamiğinden motora kadar tüm bilgilerin girildiği araç dinamiği simulasyon programıdır. Ayrıca hız profili ve manevra da oluşturalabilmektedir. Lider araç için farklı hız profillerine sahip senaryolar bu programda oluşturulmuştur. Daha sonrasında parçacık sürüsü optimizasyonu sisteme uygun şekilde Matlab platformunda kodlanmış ve yazılım Simulink'teki entegre edilmiş araç ve seyir sistemi modeline bağlanmıştır. Optimize etmek istenilen parametreler simulasyon ortamında defalarca simulasyon koşularak istenen sonucu sağlayacak şekilde denenmiş ve kalibre edilmiştir.
Parçacık sürü optimizasyonunun ana mantığına bakacak olursak: bu metot "swarm intelligence" diye tanımlanan sürü zekası algoritmalarından biridir. Bu tarz algoritmalarda doğada sürü halinde yemek bulmaya veya avlanmaya çalışan canlı toplulukları birincil örnekler olarak düşünülmüştür. Örnek verecek olursak, bir kuş sürüsü sürü halinde bir noktadan başka bir noktaya hareket etmektedir. Sürü zekası algoritmalarının genelinde sürü içerisindeki bireylerin birbiri ile haberleştiği öngörülmektedir. Bu şekilde sürekli doğru noktaya doğru birbirlerini yönlendirirler. Bu çalışmada kullanılan PSO metotu da bu yöntemle her yapılan iterasyonun sonucunu birbiriyle paylaşan bir sürünün doğru noktaya ulaşması için uğraşır. Yazılımda tezin içerisinde detaylı bir şekilde anlatılacak olan parçacık ve iterasyon sayıları aslında bu sürünün kaç kuştan oluştuğu ve kaç defa hareket edeceğinin ölçütleridir. Doğru sonuca bizi doğru şekilde iletmesi açısından bu parametreler oldukça önemlidir. Tezin ana akışı ve amaçları dışında ana yapısına bakacak olursak, bu tez toplam altı bölümden oluşmaktadır.

İlk bölümde tezin amacı kısa ve öz bir şekilde anlatıldıktan sonra literatürdeki benzer çalışmalardan bahsedilmiştir. Bu bölümde literatür araştımaları daha çok PID kontrol metotlarının nasıl kalibre ya da optimize edildiği ile alakalıdır. Farklı kontrol algoritmaları ile aynı sistemin çalışma örnekleri bir sonraki bölümde verilmiştir. İkinci bölümde adaptif seyir sistemlerinin konsepti, ana çalışma mantığı ve yapısı ile birlikte literatürdeki kontrol metotları detaylı bir şekilde anlatılmıştır. PID dışından yeni metotlar da daha önce söylendiği gibi bu bölümün parçalarından biridir.

Yine aynı bölümde bu tezde kullanılmış olan adaptif seyir sistemi yazılımı detaylı bir şekilde gösterilmiş ve formülize edilmiştir.

Bir sonraki bölüm olan üçüncü bölümde, boylamsal olarak aracın nasıl modellendiği seçilen ve teknik özellikleri anlatılan bir araç üzerinden hesaplarla birlikte gösterilmiştir. Araç seçilip parameterler belirlendikten sonra; aracın tekerlek modeli kurulmuş, maruz kaldığı direnç kuvvetleri teker teker tanıtılmıştır.

Sistemin doğru bir şekilde simüle edilebilmesi için kritik olan noktalardan biri aracın limitlerinin modelde doğru şekilde oluşturulmasıdır. Bu yüzden gerekli vites ve motor bilgileri ile birlikte limitasyonları modellenmiştir. Bu tezin asıl odaklandığı nokta optimizasyon algoritmasının entegrasyonu olduğu için aracın vites değişimi mekanizması basit bir mantık ile kurulmuştur. Belirli bir motor hızında sürekli vites değişimi öngörülmekte ve bunun detayları anlatılmaktadır.

Dördüncü bölümde bu tezde kullanılan modern optimizasyon metodu, PSO, anlatılmıştır. Detaylı PSO anlatımından önce, PID kontrolcülerinin tarihi boyunca nasıl kalibre edildiği tartışılmış, literatürdeki çalışmalardan yararlanılmıştır. Ayrıca diğer modern optimizasyon metotlarından da bahsedilmiştir. Sonrasında detaylı bir şekilde PSO denklemleri tanımlanmış ve bu çalışmada nasıl kullanılacağından bahsedilmiştir.

Beşinci bölüm sistemin ana entegrasyonu ve simülasyonundan oluşmaktadır. Önce Matlab/Simulink ortamında modellerin ve yazılımın nasıl bir arada çalıştığı anlatılmış, belirlenmesi gereken parametreler belirlenmiş ve sebepleri açıklanmıştır. Burada parameterlerin belirlenmesindeki literatürdeki farklı metotlardan ayrıca bahsedilmiştir. Optimizasyon algoritmasının ana çalışmasını ve sonucunu direk etkileyen parametreler üzerinden çalışılan sistem ile ilişkiler kurularak neyin neden ve ne şekilde kullanıldığı açıklanmıştır. Daha sonrasında seyir sistemlerini ayrı ayrı ve birlikte değerlendiren farklı senaryolar ile birlikte simülasyon sonuçları paylaşılmıştır. Paylaşılan simülasyon sonuçları yorumlanmış ve beklenilen sonuçlara erişilmiştir.

Son bölüm olan sonuç bölümünde sonuçlar açık bir şekilde paylaşıldıktan sonra ileriki çalışmalar için öneriler sunulmuştur. Bu çalışmanın sonrasında yapılabilecek bir çok çalışmadan bahsedilmiştir. Bu çalışmaları kısaca tanımlayacak olursak, en başta kontrolcü algoritmanın başka durumlara entegre bir şekilde çalışabilmesi için daha komplex ve genel bir şekilde tanımlanması gelir. Sonrasında tek kütleli bir sistem olarak modellenen aracın tam olarak dört kütleli model şeklinde modellenerek, aracın tüm komponentlerinin neredeyse etkisi daha gerçekçi bir biçimde simule edilecektir.

Parçacık sürüsü optimizasyonunun parametreleri de geliştirilebilecek noktalardan biridir. Parametrelerin belirlenmesinde var olan farklı yaklaşımlar denenebilir. Son olarak bu sistem gerçek araç üzerinde yolda denenerek doğruluğu test edilmiş olacaktır. Bu bölümden sonra tezin içerisinde ayrıntılı olduğu için direkt olarak anlatılmamış ekler verilmiştir. Sonuçları gösterilmeyen senaryoların hepsinin sonucu ve bu sonuçların yorumları da ekler bölümünde mevcuttur.

1. INTRODUCTION

In automotive industry, usage of ADAS and autonomous functions are increasing day by day. As a brief history, driver assistance systems (DAS) started to take a role in industry at early 1980s with ABS technology. Then; ESP, Traction Control System (TCS) and Electronic Stability Control (ESC) features are introduced with using driving dynamics of vehicles. New sensors are included into the vehicle system architecture with enhancing assistance technologies such as Global Navigation Satellite System (GNSS), inertial measurement systems, radar, ultrasonic sensors etc. At 1990s thanks to these sensors, the second phase of DAS is implemented on vehicles and leads more detailed information flow for driver and improved comfort of driving [1]. Figure 1.1 shows how the DAS evolve in three decades in point of 2014 [1].

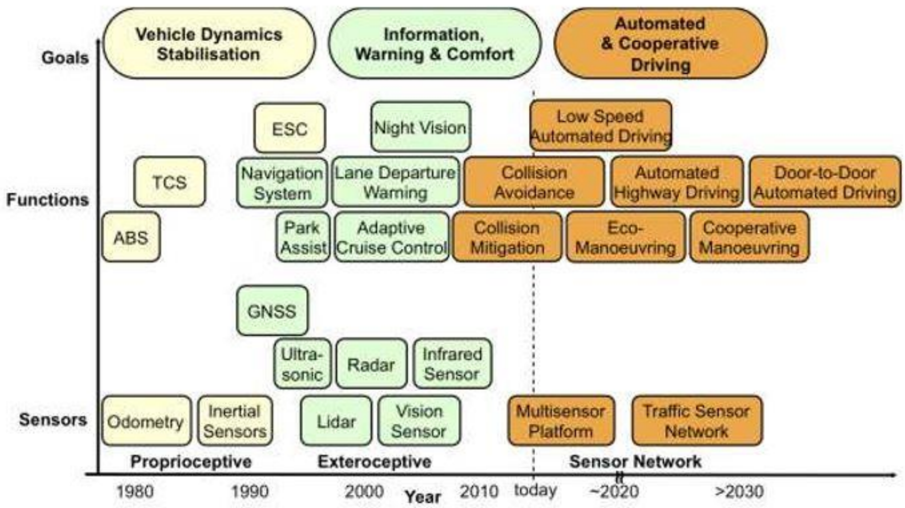


Figure 1.1 : Past and future trends on DAS [1].

Apart from the active safety features, conventional cruise control is one of the milestones on second phase of DAS as an intelligent transportation system. In this CCC systems, vehicle speed is set by the driver, and the algorithm automatically adjust the throttle and brake pedals in order to provide desired vehicle speed. Any driver intervention such as applied pedal or pressed button can overwrite the CCC systems [2]. Adaptive cruise control systems are based on CCC systems in a basic principle. It detects the preceding vehicle with a range sensor, and uses its velocity and the relative distance between two vehicles in order to set vehicle speed automatically [3]. Future trends go towards highly automated and cooperative systems with no driver intervention. The main disadvantage of this systems is the more complicated algorithms development.

1.1 Purpose of Thesis

As described before, with enhancing ADAS features, all the safety critic decisions are up to quality of sensors and algorithms. For higher complexity softwares, optimization is becoming more significant factor in order to meet safety and performance requirements. Therefore, the main purpose of this thesis is to design an ACC system and optimize the ACC and CCC's controller parameters and observe the contribution of the optimization algorithm on a vehicle simulation.

In order to get realistic results, a diesel vehicle is modelled on Matlab/Simulink platform. ACC algorithm, longitudinal vehicle model and the optimization and objective codes are connected each other to create a basic tool chain for providing iterative optimization. Then, for each ACC and CCC, ten main scenarios are defined to simulate the system. As an optimization algorithm, particle swarm optimization method is chosen.

1.2 Literature Survey

There are many studies related with this problem specifically and also controller design, optimization processes and vehicle modelling are tackled separately for different cases which are discussed in details on relevant chapters. As a start point, optimization of CCC is a massive problem. In 1999, Holzmann et al. stated that there are many different vehicles with equipped ACC, and optimization procedure of high level controller should be unique [4]. Controller optimization process is introduced with two main steps which are realistic simulation and vehicle test. Simulation phase leads to gain more time and decreases costs. In the simulation phase, Matlab's Optimization Toolbox are used to optimize defined proportional-integral-derivative (PID) controller parameters after integrating system specification and limits into model. Moon et al. indicated an optimization platform with using real world data from a vehicle with experienced drivers in 2008 [5]. With using a confusion matrix integrated with real driving data, controller is tuned on hardware in the loop (HIL) system. Main architecture of this workflow which conducted by Moon et al. and Figure 1.2 shows that.

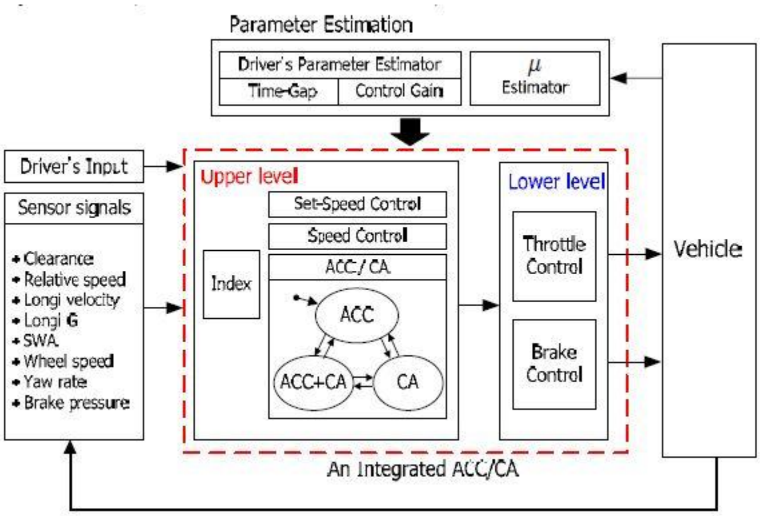


Figure 1.2 : System architecture [5].

Optimization techniques are enhanced day by day and implementation of this new methods to real use cases and applications. Rout et al. introduced an optimization of CCC PID parameters with using genetic algorithm (GA) in 2016 [6]. This study includes a transfer function which represents the longitudinal vehicle model and standard PID block. An objective function is described which is intented to minimize in this study. This function is a combination of maximum overshoot, peak time, settling time etc. In this article, genetic algorithm is used for minimizing the defined objective function with different PID parameters and it is declared that genetic algorithm demonstrates the its dominance in the comparisons with other techniques.

In 2017, Abdulnabi stated that researchers applied various techniques to optimize the PID controller in CCC algorithms [7]. He emphasized that PSO methods has not been implemented into this automotive applications yet. In the article, a longitudinal vehicle model is created and implemented with a m-file which includes main PSO algorithm. According to results of optimized parameters of PSO, it is compared with conventional methods. In contrary to other methods, better performance is demonstrated by Abdulnabi. Pradhan et al. suggested that many of the optimization algorithms are struggling with “premature convergence” [8]. Thus, to avoid it, antlion optimization (ALO) technique is used in the study for tuning PID parameters of an automobile CCC system. The significant element for using ALO is its massive search area. Pradhan et al. demonstrated the basic vehicle model which includes engine map, resistive forces with integrated PID based CCC. It is stated that the proposed optimization method is evaluated with the others and results are better.

In general, automobile speed control algorithms such as ACC and CCC uses the PID controller to follow the speed trace which is calculated by the controller. Nowadays, various control algorithms are also used for speed control problems. Model predictive control (MPC) is one of the most common used technique for it.

In 2010, Luo et al. proposed a new ACC algorithm which uses MPC [9]. An objective function is fused with using the parameters of driving comfort, safety and fuel economy. Different use cases are simulated and how ACC covers the requirements is highlighted in this study. Optimization algorithms are also developed for model predictive based speed control applications. Zhang et al. introduced a new tuning method which is called “weight coefficient self tuning strategy” in 2018 [10]. In the article, it is stated that conventional MPCs are not sufficient for some transient driving use cases. A proposed method is described as an adaptively adjusted dynamic weight matrices. According to this method, once the one objective parameter is fully satisfied, its weight is directed to the other objective parameters which are not achieved yet. Zhang et al. compared the results with conventional methods and emphasize the enhancing overall performance of the vehicle.

2. ADAPTIVE CRUISE CONTROL SYSTEMS

ACC settles the vehicle speed regarding to traffic environment, it is reproduced from CCC systems. In contrary to CCC systems, ACC provides increasing or decreasing vehicle speed automatically [11]. It requires a range sensor which is able to measure the vehicle speed of lead vehicle. This range sensor can be a radar, lidar, camera etc. Lead vehicle detection and selection are determined by fusing the information from specified range sensor and road curvature information [11].

2.1 Overview of ACC

ACC can be described as the automatically speed and distance control regarding the lead or target vehicle. As shown in the Figure 2.1, main terms of ACC system are clearance, time gap, lead vehicle speed, following vehicle speed. Clearence is the distance between lead vehicle's back surface and the following vehicle's frontal surface. Time gap is defined as the time when following vehicle can reach the lead vehicle according to current speeds [12]. These parameters have some limitations according to the ISO-15622 [13].

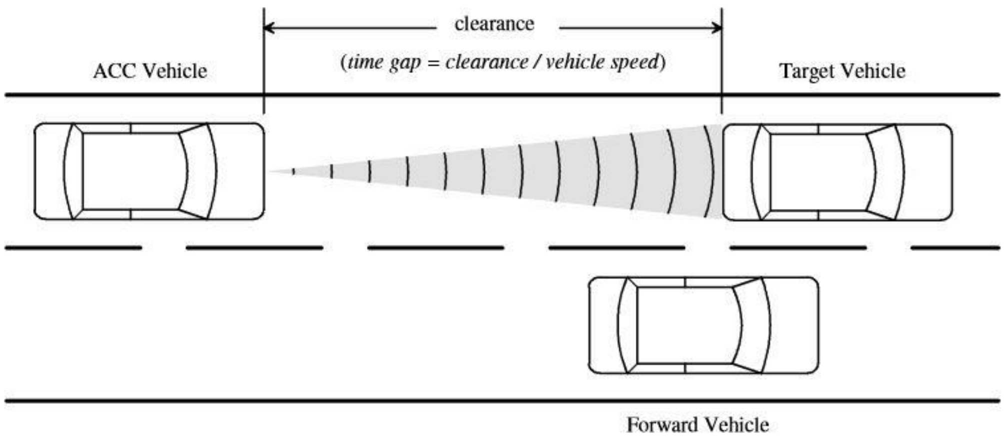


Figure 2.1 : ACC overview [12].

2.1.1. Concept and operation of ACC

If we shed light on systems for longitudinal control of the vehicle, whole system can be thought as position and velocitiy control. In ACC systems, if there is no detected vehicle in front of the following vehicle, control system adjusts vehicle speed as same as desired vehicle speed which is determined by the driver. Once the lead vehicle is detected, according to relative distance and time, controller adjusts the vehicle speed to protect desired time gap or distance [2]. If the lead vehicle leaves the lane or increases its vehicle speed higher than determined speed limit by the following vehicle's driver, controller system deactivates the ACC state, and turns into cruising state. This state transitions are also determined by the ISO-15622 standards [13], as shown in the figure 2.2.

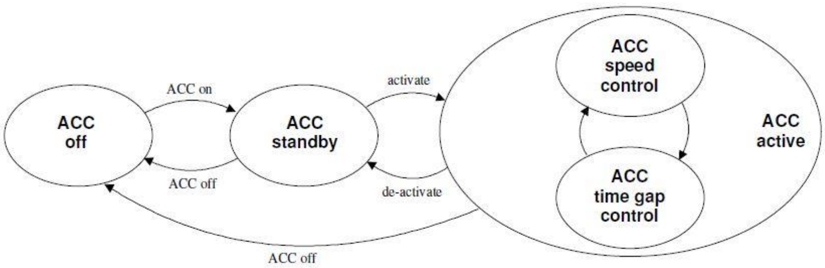


Figure 2.2 : ACC state machine [12].

ACC off state is deactivation phase of ACC. In the standby state, ACC waits for the activation consent by the driver. Speed control phase is the CCC phase, set speed is applied and algorithm provides it. Time gap control is the main ACC phase which adjusts brake or throttle inputs regarding to lead vehicle [11].

Two main inputs are defined by the driver in the ACC systems. One of the inputs is the desired set speed when the ACC is off, and the other one is how much time is desired between two vehicles. This time gap also has limitations according to the standards.

ACC system needs to be aware of the traffic flow. There are some downside scenarios for ACC and also these scenarios are mentioned in the standards and literature. In order to be aware of the traffic flow, controller and sensors have to make right decision for detecting the lead vehicle, pedestrian or any obstacle. As demonstrated in the figure 2.3, when the ACC is active while passing a curved road, range sensor detects lead vehicle as wrong. To prevent this, range sensor should also detect the environment as much as possible [2]. The other solution is to use more information which are capable to detect downside situations.

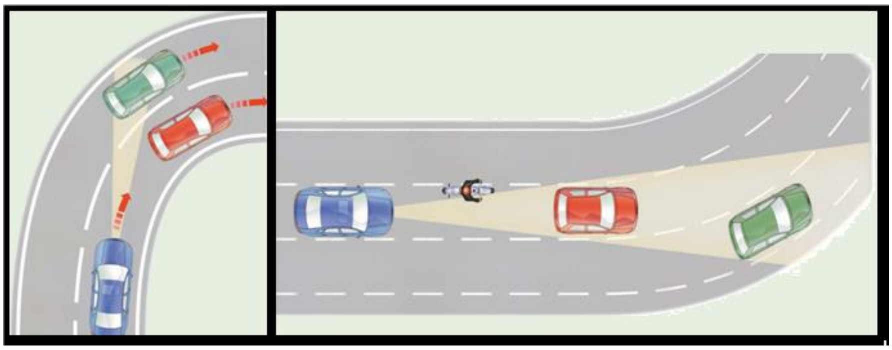


Figure 2.3 : Road curvature effect on ACC [2].

ACC has an extension usage which is called stop & go systems. In normal ACC systems, ACC can be activated above 30 kph. With this extensive feature of ACC, system can be activated in low speeds. Stop & go works in a wide range of acceleration in contrary to ACC systems. Thus, system needs to know more information about the conditions of traffic environment. It leads usage of alternative additional sensors [2].

2.1.2. Components of ACC systems

In general applications, ACC systems can use laser, ultrasonic, lidar, radar, camera, wheel speed, corner sensors and also accelerometer [14]. With integration of user desired info and sensor info, ACC algorithm decides what actions should be conducted.

Wheel speed is the one of the important information for electronically controlled vehicles. Vehicle speed, acceleration or longitudinal slip are converted from wheel speed [11]. It is generally a kind of tachometer. Acceleration information can be calculated from vehicle speed or wheel speed directly but for getting more accurate data, gyroscope or inertial measurement sensors are generally used.

In various applications ultrasonic sensors are extensively employed. The main principle behind the ultrasonic sensors are speed of sound which is sent through the obstacle. Distance between sensor and obstacle is calculated from speed of sound [11]. Thus, the location of sensor is the most important parameter for ACC systems. Ultrasonic sensors are working in low distance ranges, due to this fact, it is generally used for stop & go applications.

Radar is generally a main component of ACC systems. It works based on a principle that is to emit the signals which bounce back from an obstacle. Radar reports the data such as distance range, angle and doppler velocity. It is not affected by weather conditions. One of the other significant range sensor is Lidar. It employs an infrared laser wawes to appoint the relative distance gap between the obstacle and sensor [15].

A combination of all these sensors are used as a general layout for ACC application. Combination type determines capability and usage of the ACC system. Also with various combinations lead to create an usage area for rest of the ADAS applications. Figure 2.4 shows that an example for these combinations.

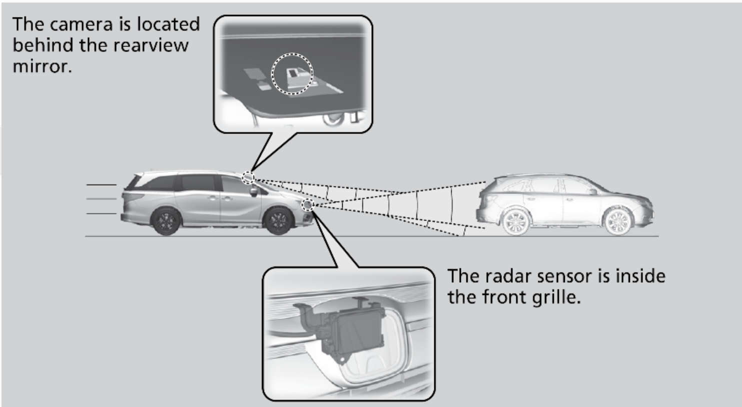


Figure 2.4 : ACC sensor combination example.

2.2. Controller Methods of ACC

ACC was integrated onto the market approximately two decades ago and controller algorithms are still evolving with the enhancing technology. Various types of controller strategies have been examined since the first introduction of ACC onto the automotive market [16]. Strategies can be classified into three main types which are PID controller, gain scheduling, MPC controller. Nowadays, newly developed control strategies started to get integrated with ACC systems such as fuzzy logic controller, neural network etc.

2.2.1. PID controller

PID is a widely employed system for industrial applications. It has three main elements which are proportional, integral and derivative elements [16]. In general, ACC control algorithms include two main loops which are outer and inner. Outer loop calculates the desired acceleration or torque etc. This also decides the switching conditions and what strategy is used for required action. Inner loop uses the information given from outer loop. It includes low level actuator control algorithms such as desired throttle or brake. Outer loop and inner loop can be also called as high level and low level controller respectively. PID basically calculates magnitude of required action with using the error signal. Error depends on the system usage and algorithm. In many ACC application, error is the relative distance between vehicles.

In 1993, Ioannau et al. described an ACC system which is based on time gap control [17]. According to these method, regarding the vehicle speed, desired relative distance differs due to desired constant time gap. Algorithm provides a constant gap of time and sets its speed in order to ensure desired time gap. Output of the outer loop is the acceleration reference and switching conditions between throttle and brake pedal. Inner loop uses this informations and controls throttle and brake behaviours to substantiate the reference. Figure 2.5 shows one of the ACC algorithm which is built with outer and inner loop integrated.

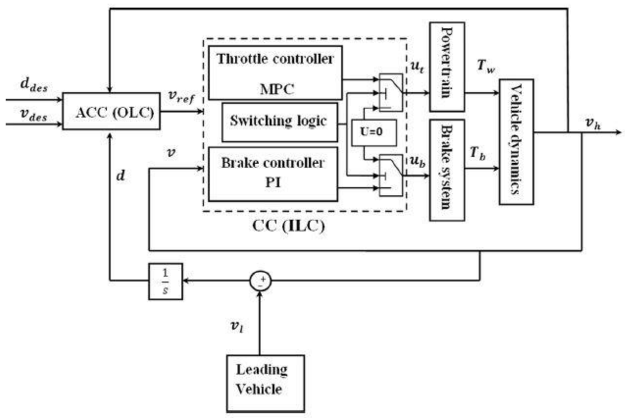


Figure 2.5 : ACC algorithm structure [17].

PD control structure is also generally implemented on ACC systems [2]. In this system commonly, relative vehicle speed and relative distance are controlled together. P gain is producted with relative distance error and D gain is producted with relative speed error. Desired relative distance is calculated with using desired time gap which is defined by the driver with human machine interaction (HMI) systems. Summation of these two result can be assigned as acceleration or torque reference. Apart from time gap strategy, there are different types of strategies. Constant clearance is one of the common strategies which can be described as the fixed spacing generated between the vehicles. It is an ideal strategy for close following [18]. Safety distance method is also used for ACC strategies. It is derived from a scenario. Scenario is based on the emergency braking or hard braking of lead vehicle. When hard braking is conducted on lead vehicle, following vehicle reacts it with some delayed distance which is a function of reaction time. The distance between the braking distance of the lead vehicle and the stopping distance of the following vehicle is the safety distance. Nowadays, variable time gap usage is increased due to specified scenarios. According to use case, time gap varies as intented. Finally, if we shed light on effects of the PID parameters, proportional gain has an influence on the error signal as increased proportionally. It generally does not eliminate if there is steady state error. Derivative gain can be thought as damping factor. It prevents overshoot but as proportional gain, it has not influence on steady state error. Integral gain tends to decrease the error at steady state phase. Too large integral gains cause oscillation on system.

2.2.2. Gain scheduling PID controller

Vehicles are highly dynamic systems, there are many system equipments which have nonlinear behaviours. They work on different operating conditions such as load, engine speed, acceleration etc. Behaviour of various operation conditions varies. Fixed gain PID controllers can not undertake the required actions on all operation points as intented or designed. Thus, gain scheduling is a specialized method that undertakes the desired control behaviours with different gains which vary regarding defined operation conditions. Optimal control can be provided on a large range operations with gain scheduling. In CCC algorithms, reference acceleration P gain is calculated for large range of vehicle speed and vehicle speed error combinations as an instance for gain scheduling. This method can be also implemented to other types of controllers such as MPC.

2.2.3. Model predictive controller

MPC does not indicate completely a control algorithm. It employs the system model to create control signal which aims to minimize an objective function [19]. As shown in Figure 2.6, MPC steers the predicted output to the reference trajectory with generating the ideal solutions for optimization process. It uses data from the past to predict the future. In 2010, Luo et al. introduced an ACC algorithm which cooperatively works with MPC controller [9].

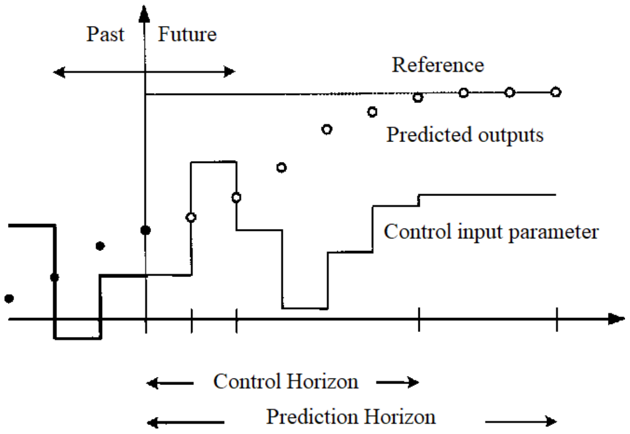


Figure 2.6 : MPC overview.

In the study, inner loop controller is assumed as well-designed, outer loop is controlled by MPC with multi-objective function. ACC algorithm is based on constant time gap. By using MPC, constant time gap is controlled considering comfort, safety, fuel efficiency and following the lead vehicle. Emphasised optimization function is a combination of objectives which is desired to be satisfied.

2.3. Controller Design

In this thesis, ACC structure has two main layers which are outer and inner. Outer loop calculates the desired wheel torque to keep vehicle at desired conditions. Inner loop is assumed as a first order system [9]. Outer loop is controlled by PD controller with time gap main strategy. Outer loop has also switching algorithm between CCC and ACC states according to inter-vehicle distance.

Time gap that is set by the driver is employed to calculate the desired distance as shown in equation 2.1 and set time gap is represented as τ_{set}.

$$\tau_{set}.V_{Ego} = d_{des} \tag{2.1}$$

Desired distance error and vehicle speed error are the main control inputs of the system. Equation 2.2 shows that the main equation of the ACC controller:

$$\mathrm{T}_{wheel,des} = K_v.(V_{Lead} - V_{Ego}) + K_t.(\,d_{act} - d_{des}) \tag{2.2}$$

where K_v and K_t are the main controller parameters and V_{Lead} refers to lead vehicle speed and d_{act} is actual relative distance between the lead and following vehicle and it is conducted by the equation 2.3.

$$d_{act} = \int V_{Lead}\,dt - \int V_{Ego}\,dt \tag{2.3}$$

Initial values of integrators are 40 meter and 0 meter for lead and ego vehicles respectively. In this thesis, it is assumed that in start phase of the system 40 meter relative distance is existed between the vehicles. Main switching algorithm aims to activate ACC when the actual relative distance is lower than 30 meter. Unless this condition is satisfied, ego vehicle moves forward with only CCC. Moreover, if this condition is satisfied, to switch back to CCC control relative distance must be greater than 40 meter. This hystereses is considered for avoiding switching problem between the ACC and CCC around 30 meter.

CCC controller is activated when CCC is active. There are three parameters to satisfy set vehicle speed by the driver. Equation 2.4 demonstrates the motivation behind the CCC controller where K_p, K_i and K_d are the PID gains and V_{Set} is the set vehicle speed determined by the driver.

$$\mathrm{T}_{wheel,des} = K_p\left(V_{Set} - V_{Ego}\right) + K_i \int (V_{Set} - V_{Ego})dt + K_d \frac{d(V_{Set} - V_{Ego})}{dt} \tag{2.4}$$

These equations are combined and integrated into Simulink model which can be shown in the Figure 2.7.

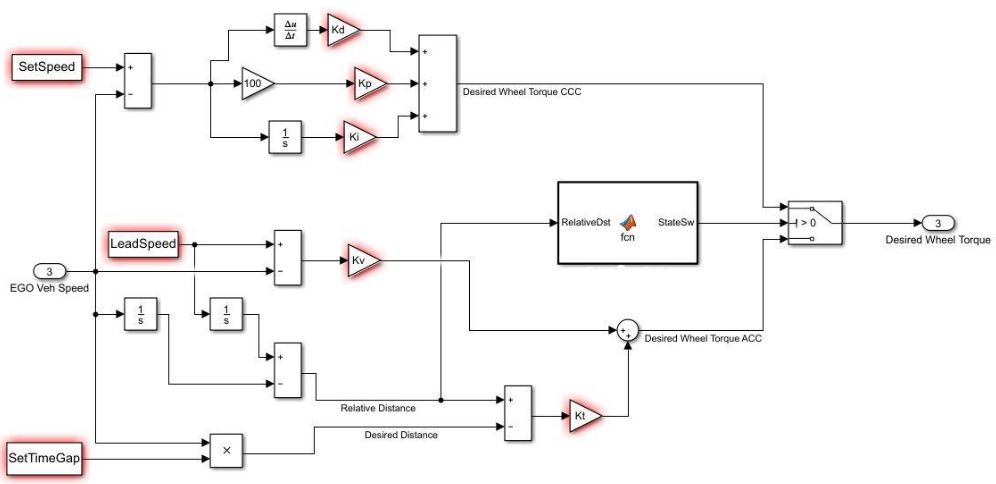


Figure 2.7 : Outer control algorithm.

3. VEHICLE MODELLING

To implement and simulate the ACC and CCC system, longitudinal vehicle model is needed. In this thesis, one mass longitudinal model is built with using vehicle dynamics, tire equations and a simple engine-gearbox model. First, ego vehicle specifications are introduced.

3.1 Ego Vehicle Specifications

As an ego vehicle, it is selected a 1.4 liter diesel engine with 6 level of gear vehicle. All specification relevant with the ego vehicle is given at Table 3.1.

Table 3.1 : Ego vehicle technical specifications.

Specifications	Values	Units
Power @4000 rpm	90	PS
Max Torque @1750-2750 rpm	220	Nm
Valves	16	Unitless
Capacity	1.4	Liter
1st Gear Ratio	3.8	Unitless
2nd Gear Ratio	2	Unitless
3rd Gear Ratio	1.37	Unitless
4th Gear Ratio	1	Unitless
5th Gear Ratio	0.84	Unitless
6th Gear Ratio	0.7	Unitless
Final Drive Ratio	4.3	Unitless
Curb Weight	1175	Kg
Test Weight	1250	Kg
Tyre Radius 205/55 R16	0.316	Meter
Driveline Efficiency	0.97	%
Transmission Efficiency	0.97	%

Engine torque curve is assumed as shown in the Figure 3.1.

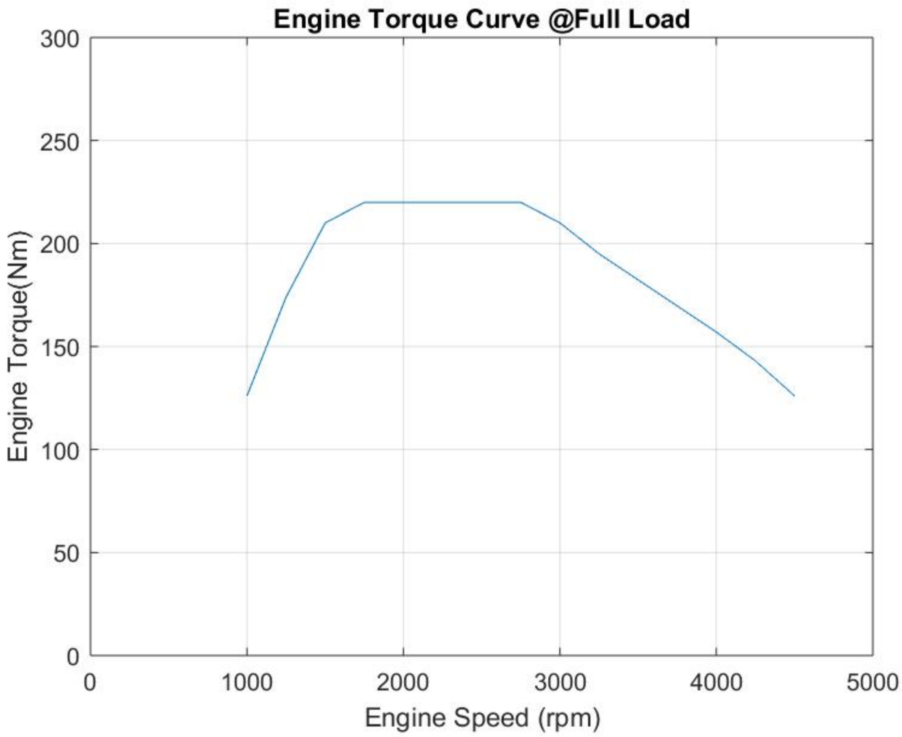


Figure 3.1 : Ego vehicle engine torque curve.

According to vehicle specifications wheel torque values are calculated as per one tire for corresponding engaged gear. Full load engine torque at a specific speed which is referred as T_{Engine} products with corresponding gear ratio (GR_i) and final drive ratio (FDR) with including driveline efficieny (η_{Dr}) and gearbox efficiency (η_{GB}) production. To calculate the wheel torque as per one tire, this calculated torque should be divided by two for two tires. Equation 3.1 represents the calculation of wheel torque for one tire that is indicated as T_{Wheel}.

$$\mathrm{T}_{wheel} = \frac{\mathrm{T}_{Engine}.GR_i.FDR.\eta_{Dr}.\eta_{GB}}{2} \tag{3.1}$$

Wheel torque values for each gear can be shown in the Figure 3.2.
Torque is converted to wheel domain. In this thesis, desired wheel torque is employed for controlling vehicle speed and inter-vehicle distance. At wheels torque can be converted into wheel force with dividing by wheel radius that is demonstrated by (r). Equation 3.2 points out to tractive force per tire F_{wheel}.

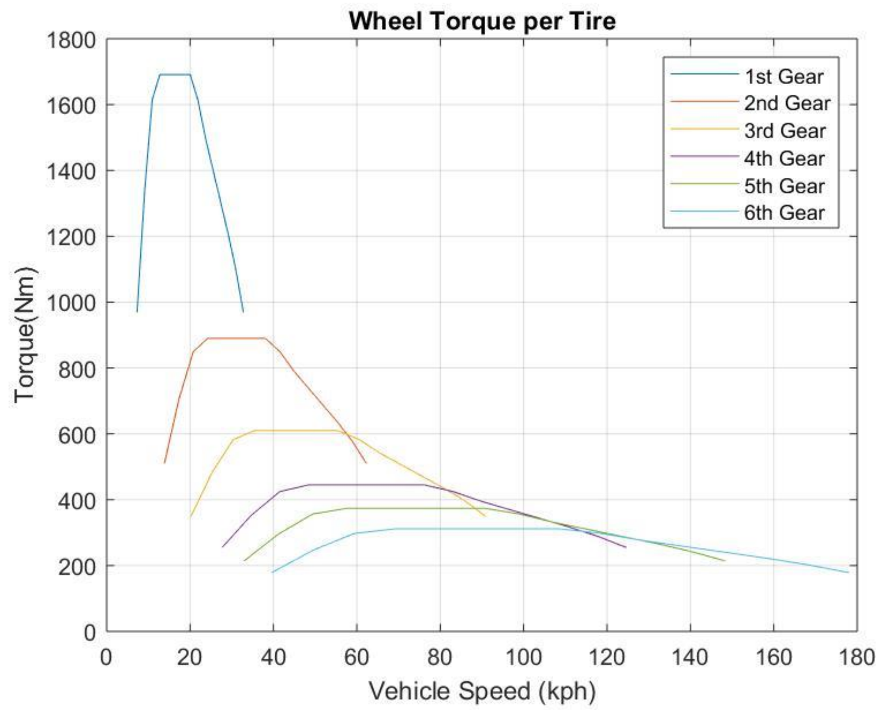


Figure 3.2 : Wheel torque curves.

Tractive force line is shown in the Figure 3.3.

$$F_{wheel} = \frac{T_{Wheel}}{r} \tag{3.2}$$

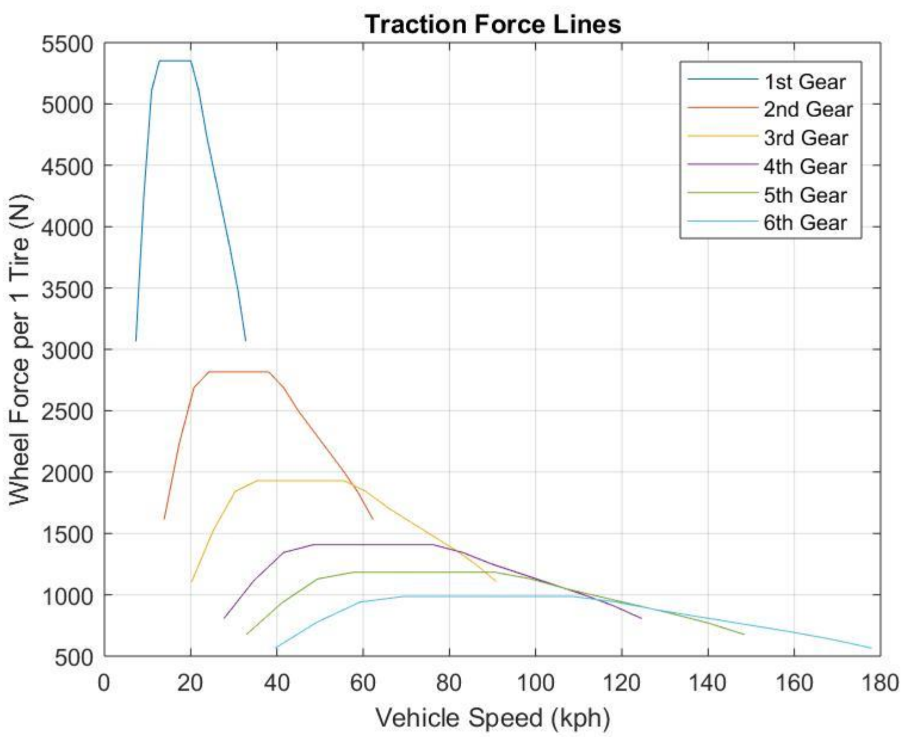


Figure 3.3 : Traction force lines for each gear.

Torque transfer from engine to wheel can be observed in Figure 3.4. This also represents how produced torque can tranform into wheel acceleration and torque flow on powertrain components.

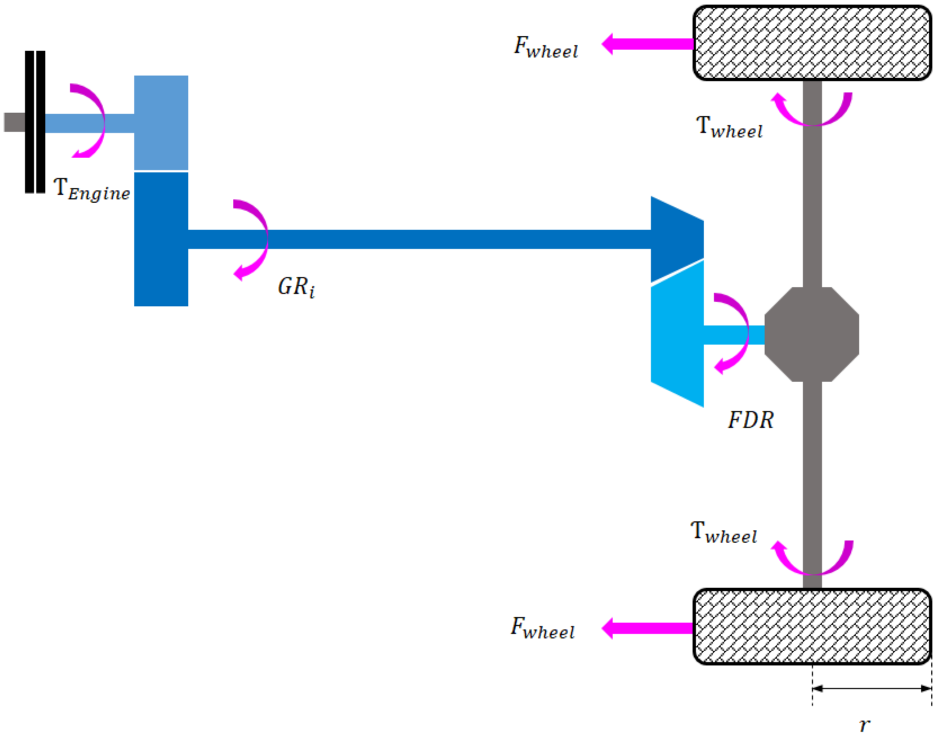


Figure 3.4 : Torque transfer scheme on powertrain.

3.2. Longitudinal Model of Ego Vehicle

Before starting of the vehicle model process, it is needed to determine a coordinate system which vehicle works. Society of Automotive Engineers (SAE) introduced a coordinate system as demonstrated in Figure 3.5 [20]. According to this coordinate system, in this thesis, only -x direction is used. Vehicle is modelled longitudinal -x direction with resistive forces equations and tire model.

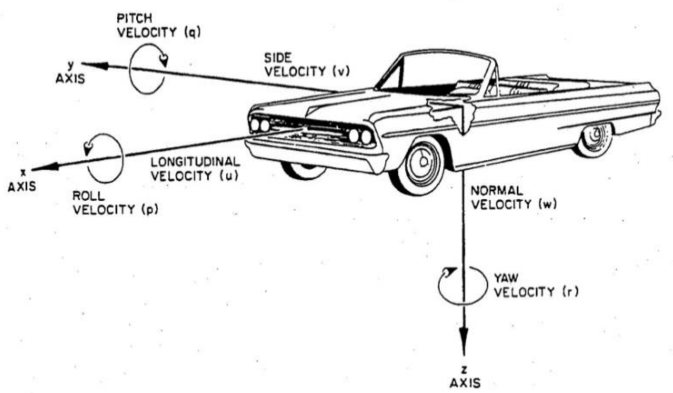


Figure 3.5 : SAE vehicle coordinate system [20].

3.2.1. Tire model

Tires are most significant elements which have an effect on vehicle dynamics. It is too complicated to easily build a tire model, thus there are some emprical methods to create a model of tire. Pacejka model is commonly employed for modelling a tire. This is also called as magic formula. In this method, longitudinal tire force F_X is computed with using longitudinal slip (s). Equation 3.3 points out to longitudinal slip calculation while braking.

$$s = 1 - \frac{R_0 . w_{wheel}}{V} \tag{3.3}$$

Equation 3.4 shows that slip while accelerating where R_0 is dynamic radius of wheel and w_{wheel} is wheel speed [21].

$$s = 1 - \frac{V}{R_0 . w_{wheel}} \tag{3.4}$$

Equation 3.5 shows the calculation of tire force [21].

$$F_X = A\sin(Btan^{-1}\left(Cs - D\left(Cs - tan^{-1}(Cs)\right)\right)) \tag{3.5}$$

where B, C and D are shape factors and are functions of slip angle, slip ratio, angle of camber and wheel force. A is the maximum value of experimental curve. Method is based on tire experimental curve and the principle behind that is the curve fitting strategy [21].

Pacejka formulations are implemented onto the Simulink model as a Matlab function, rest of the formulation are given in Appendix A.

3.2.2. Resistive forces

Total resistive force is a summation of aerodynamic resistance, rolling resistance and grading resistance forces as shown in Figure 3.6.

Aerodynamic force F_{Aero} is a resistance due to air friction during movement of vehicle. It is related with air density (ρ), frontal projection area of vehicle (A_f), vehicle speed and air drag coefficient of vehicle which is referred as C_d.

If the vehicle speed is higher, aerodynamic force increases. Calculation is indicated in equation 3.6.

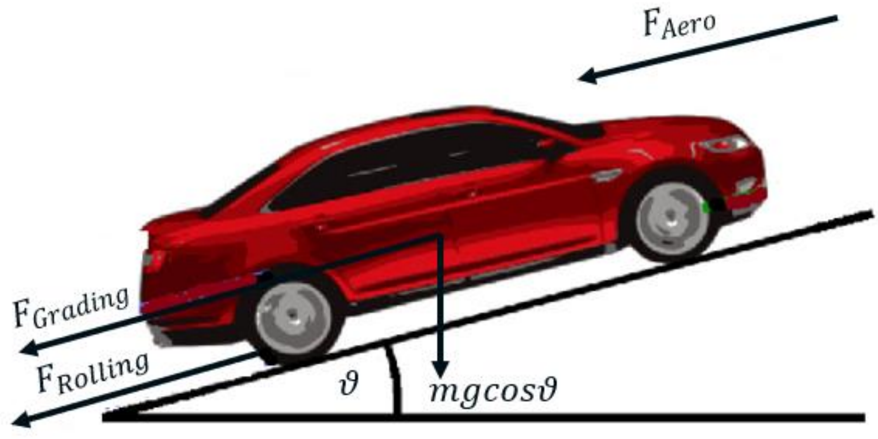


Figure 3.6 : Existing resistive forces on a vehicle.

$$F_{Aero} = \frac{1}{2}\rho.\mathrm{A}_f.V^2.C_d \tag{3.6}$$

The other resistive force is rolling resistance force. The transferred force value from engine to tire must exceed the required force resistance to maintain acceleration of the vehicle. This force is calculated with multiplying vehicle mass (m), gravity (g) and rolling resistance factor that is assumed as 0.0015 on dry road for an automobile. If the vehicle is on an inclined road, projection of vehicle mass force on longitudinal direction also resists to movement on -x direction. Multiplying mass, gravity and sinus of θ that is inclination angle gives the force value of grading. Equation 3.7 represents the rolling resistive force.

$$F_{Rolling} = m.g.f_r \tag{3.7}$$

Equation 3.8 shows the grading resistive force.

$$F_{Grading} = m.g.sin\theta \tag{3.8}$$

Apart from the resistive forces, at center of gravity of the vehicle during movement, there is an existing inertial force which is calculated by multiplying vehicle mass and longitudinal acceleration value of the vehicle. Equation 3.9 shows the inertial force.

$$F_{Accel} = m.a \tag{3.9}$$

All the calculations are combined at longitudinal -x direction of the vehicle in equation 3.10 and implemented on Simulink model as shown in Figure 3.7.

$$F_X - (F_{Aero} + F_{Rolling} + F_{Grading}) = F_{Accel} \tag{3.10}$$

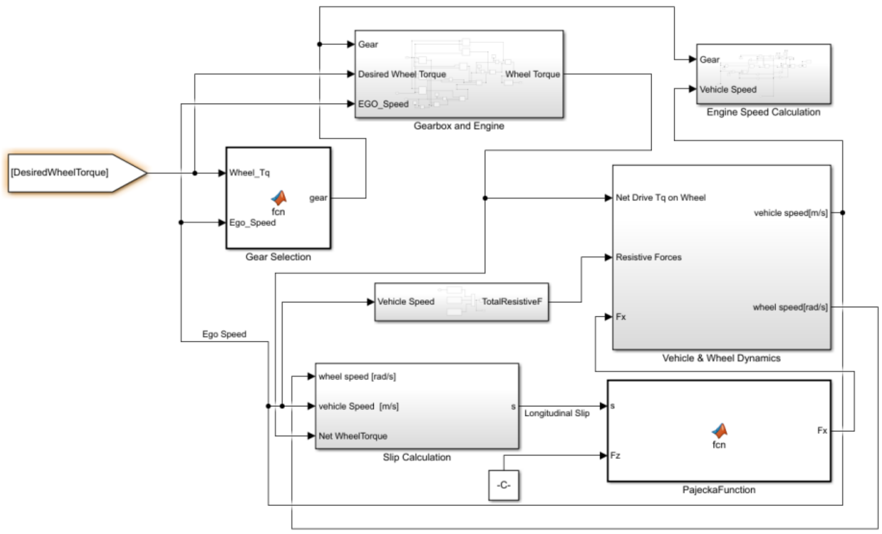


Figure 3.7 : Vehicle model integration on Simulink.

According to vehicle technical specification, maximum torque values and force values are calculated and shown before. In the gearbox and engine model, these torque limitations with regarding actual vehicle speed are applied to desired wheel torque as a saturation limit. This structure also provides the maximum capability of vehicle and even if desired wheel torque value is greater than what engine and gearbox can provide, torque is saturated and controller acts according to this saturations. Gearbox and engine limitations are shown in Figure 3.8.

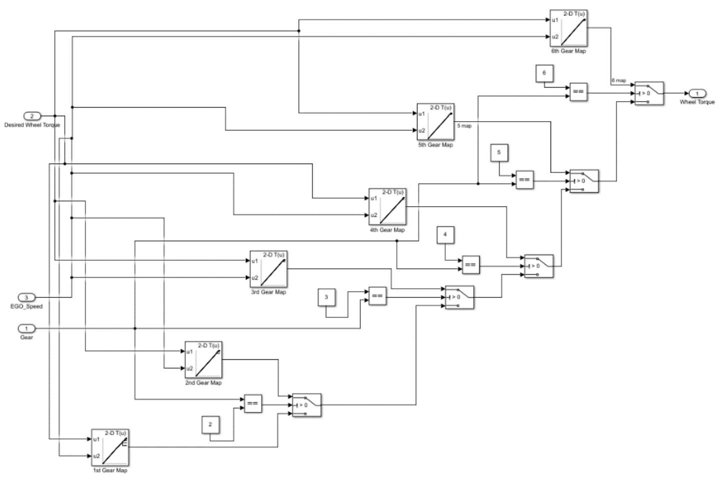


Figure 3.8 : Gearbox and engine torque saturation.

3.2.3. Shifting strategy

In the gearbox, it is assumed that engine speed upshift point is 2500 rpm. Vehicle tends to upshift once the engine speed reachs 2500 rpm. Thus, this assumption leads to create vehicle speed range for each gear level. This situation also has an impact on acceleration capability which is discussed in Chapter 5. Vehicle speed ranges for each gear are represented in Table 3.2. Maximum engine speed value is assumed as 4500 rpm.

Table 3.2 : Vehicle speed ranges for shifting.

	$V_{min}(kph)$	$V_{max}(kph)$
Gear 1	0	18
Gear 2	18	34
Gear 3	34	50
Gear 4	50	69
Gear 5	69	82
Gear 6	82	178

4. PID OPTIMIZATION TECHNIQUES

PID controller is a common used control methods applicable for almost all heavy and light industries. Throughout the history of PID controllers, there are too many developments conducted to enhance controllability and tuning of paramaters process. A question mark that is how to optimize the PID controller is still existed, too many research are conducted for this since the PID was invented. In 2013, Jaen-Cuellar et al. stated that thanks to acknowledged useful features of PID, almost all industry employed PID controllers [22]. Controlled variables performance is directly effected from tuning of the PID parameters. Thus, the optimization methods play a vital role in tuning hence, also performance of the plant.

4.1 History of PID Optimization Techniques

Optimization techniques can be classified as emprical, analytical and modern optimizations methods [22]. Ziegler-Nichols (ZN) is an emprical method which performs some premises about the plant model and aims to tune main parameters. It is advocated that step response of plant model is generally employed by almost all application [23]. There are two proposed method for ZN, only the second one is introduced in this thesis. In the second ZN method, firstly it is suggested that K_p should be increase to K_{Cr} which is referred as critical gain.

Critical gain can be calculated with using Routh arrays and also critical period P_{Cr} is calculated as demonstrated with equation 4.1 where w_f is the frequency of output oscillation [24].

$$P_{Cr} = \frac{2\pi}{w_f} \tag{4.1}$$

Equation 4.2 shows that the general transformation of transfer function calculation where $G(s)$ is the output, T_i is integrator time constant and T_d is derivative element time constant. Then, to tune the system, T_d should be changed for adjusting the settling time and T_i should be changed for eliminating the steady state error (SSE) according to ZN tuning rules which can shown in Table 4.1.

$$G(s) = K_p(1 + \frac{1}{T_i s} + T_d s) \tag{4.2}$$

Table 4.1 : Ziegler-Nichols PID tuning second rules [24].

Type of Controller	K_p	T_i	T_d
P	$0.5K_{cr}$	∞	0
PI	$0.45K_{cr}$	$\frac{1}{1.2}P_{cr}$	0
PID	$0.6K_{cr}$	$0.5P_{cr}$	$0.125P_{cr}$

Root locus and frequency response are analytical methods that aim to tune the parameters of PID. It is perceived that emprical and analytical techniques are sluggish processes [22]. With enhancing technology, more convenient modern optimization techniques are implemented into PID tuning problem. GA, PSO, ant colony optimization (ACO) are some of the modern optimization methods.

In 2010, Nagaraj and Murugananth compared GA, PSO, ACO and evolutionary programming with conventional methods [25]. In that study, objective function is built regarding the parameters which are overshoot, rising time and settling time. Firstly GA is a global search algorithm which aims to minimize or maximize objective function with using natural selection. In Figure 4.1 indicates the process flow of the GA.

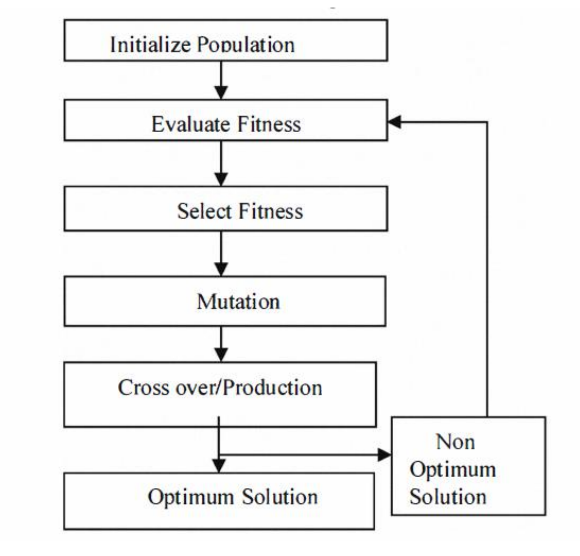


Figure 4.1 : Process flow of genetic algorithm [25].

Nagaraj and Murugananth implemented same objective function to ACO and rest of the optimization algorithms. As presented in Figure 4.2, ACO has similar process flow with the GA [25]. In the article, it is illustrated that greater performance is proved with modern optimization methods in contrary to conventional methods.

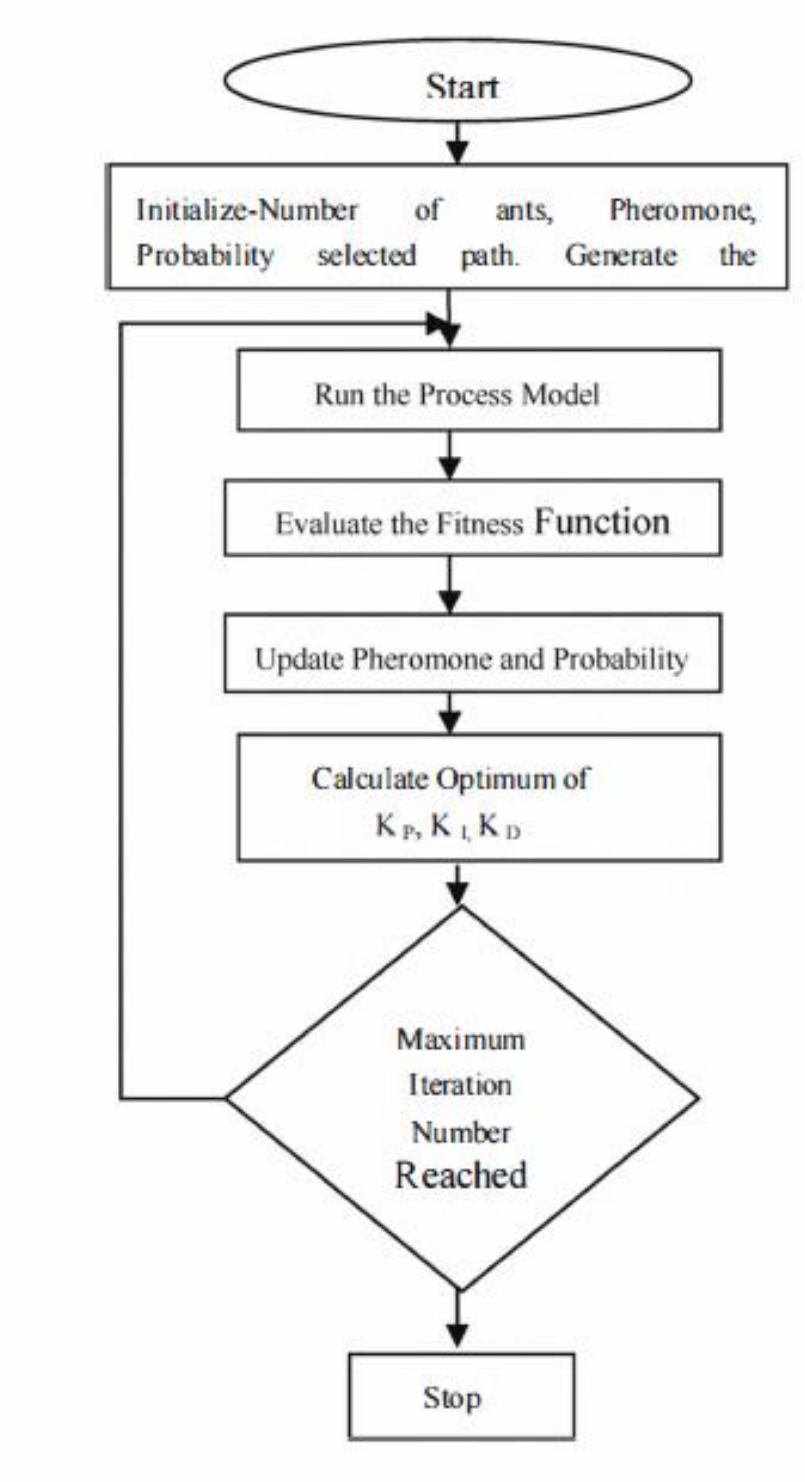


Figure 4.2 : ACO process flow [25].

4.2. Particle Swarm Optimization

Optimization methods generally are inspired from a natural event such as foraging. Insects, birds and similar species have to forage to keep alive themselves. During the foraging behaviour, each member of swarm need to be in communication each other. The basic objective is to reach the food instantly as much as possible [26]. It means in optimization process, the objective is to reach the minimum or maximum or desired value fast as much as possible as demonstrated in Figure 4.3.

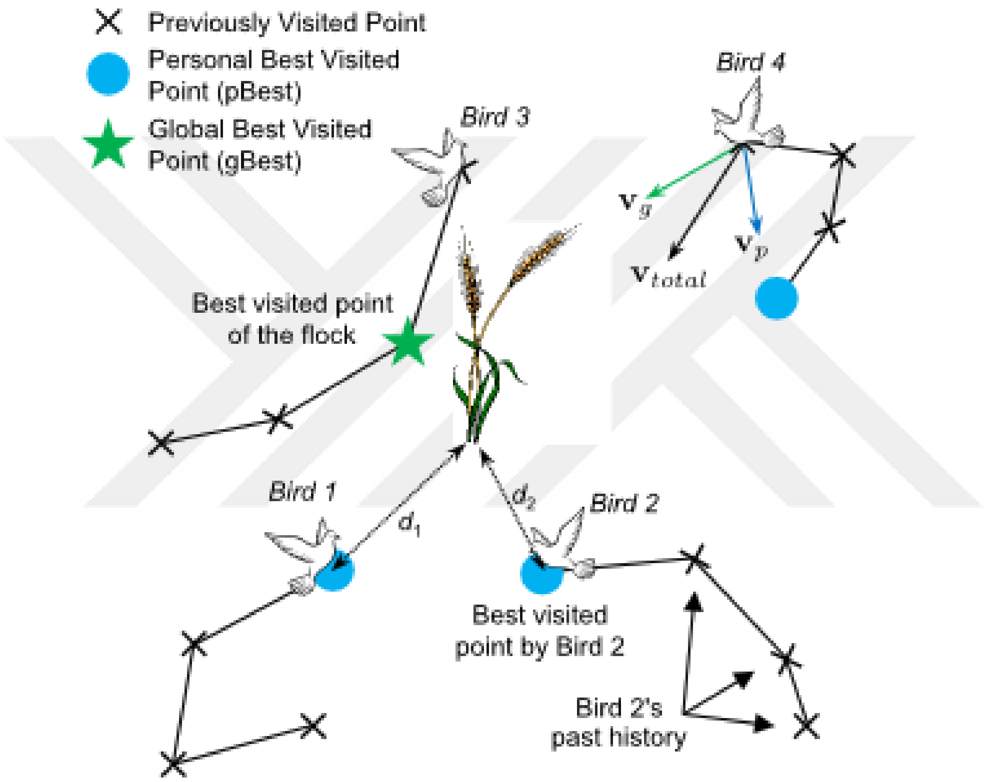


Figure 4.3 : PSO demonstration with birds [27].

The disipline which employs the collective behaviour of a social group is called as swarm intelligence. PSO is the member of swarm intelligence based optimization methods. This method is developed by Kennedy and Eberhart in 1995 [28]. The main principle behind the PSO is that every member of swarm sets its location according to best location in the group.

During this process, there are two significant elements of PSO which are location and velocity. Each particles have a location and according to difference between its location and best location of the swarm, they tend to be closer to the best location of swarm. Flowchart of the PSO is shown in the Figure 4.4.

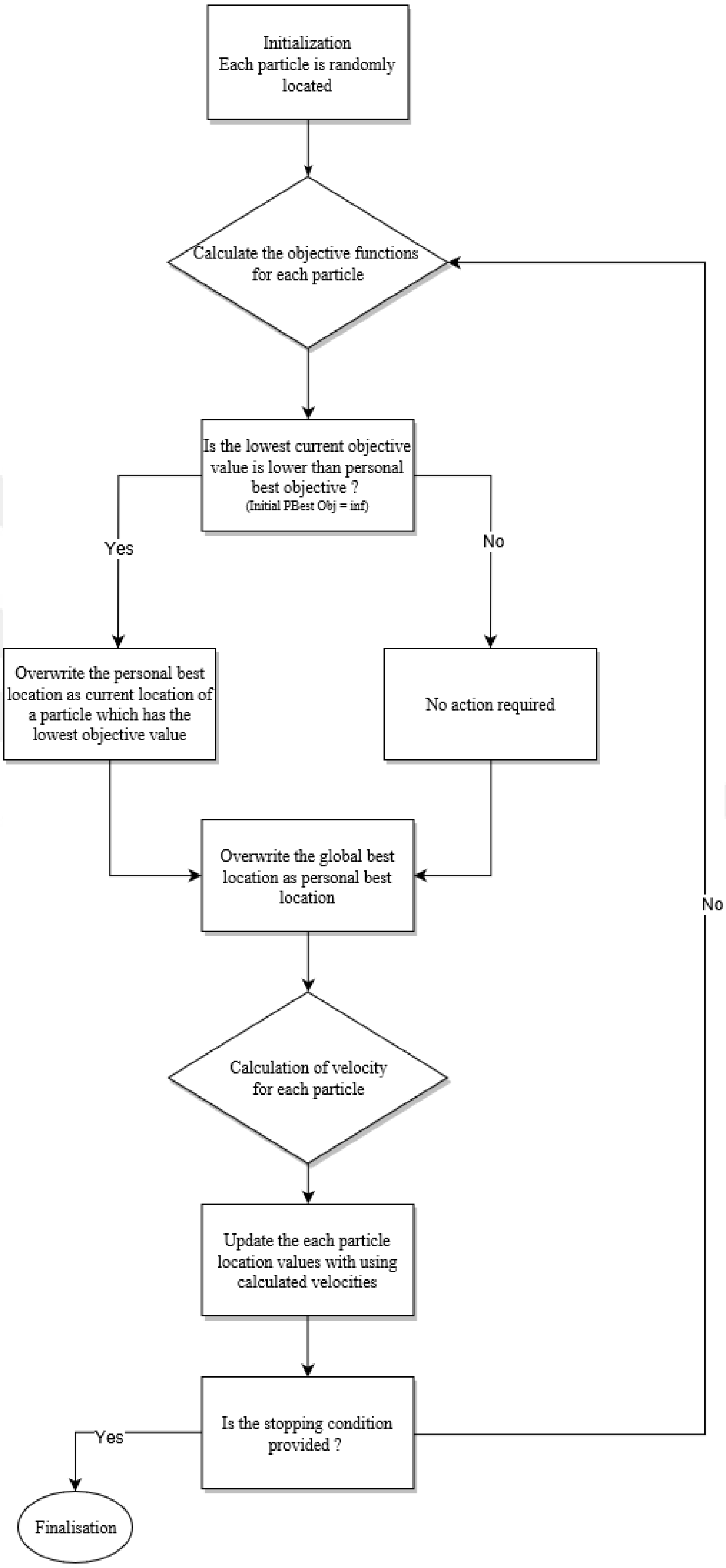


Figure 4.4 : PSO process flow.

During the implementation of PSO algorithm through ACC PID controller, firstly, number of particles of swarm is determined and it is referred as $\boldsymbol{n_{swarm}}$. Maximum iteration number is shown as $\boldsymbol{n_{iteration}}$. Also it is needed to know that how many variables are desired to tune. Variable numbers are also referred as $\boldsymbol{n_{var}}$. The values of tuneable variables are representation of position of each swarm particles that is pointed out as $\boldsymbol{x_s^i}$. Indice of s means which swarm particle is and i indice is referred iteration step. $\boldsymbol{V_s^i}$ shows the velocity of a particle where the iteration number is $\boldsymbol{i}$. Equation 4.3 indicates the first positions of the particles where the $\boldsymbol{rand}$ is a Matlab function that generates random numbers within desired lengths. Here the length is an 1 x $\boldsymbol{n_{var}}$ array.

$$x_s^i = rand(1, n_{var}) \tag{4.3}$$

Initially velocity of each particle is set to zero. It is calculated with increasing of iterations. General calculation of velocity of the each swarm particles can be shown in equation 4.4. In this equation $Pbest_s^{pos}$ is the best position of each swarm particle at an iteration and $Gbest^{pos}$ is the global best position. There are some coefficients which determines the character of algorithm, they are acceleration constants (c_1), (c_2) and inertia factor (β).

$$\begin{aligned} V_s^{i+1} = \beta . V_s^i + \ c_1 . rand(1, n_{var}) . (Pbest_s^{pos} - x_s^i) \\ + \ c_2 . rand(1, n_{var}) . (Gbest^{pos} - \ x_s^i) \end{aligned} \tag{4.4}$$

At each iteration, each swarm particle update their positions with using the calculated velocities as indicated in equation 4.5.

$$x_s^{i+1} = \ x_s^i + V_s^{i+1} \tag{4.5}$$

Personal best and global best values are determined with comparing the objective values of each swarm particles. With the initilization of the algorithm, for each particle has its best objective value as personal best. Then it is calculated for all iteration process.

Also global best value is calculated for each iteration and it means the best objective value of all particles at an iteration procedure. It is updated at all iterations until the iterations are completed. Global best is also called as social best and the other name of the personal best is individual best.

Inertia factor and acceleration constants are heavily important for PSO algorithm performance. There are some studies to investigate the effect of them on optimization performance. Shorbagy and Hassanien emphasized the consequences of these parameters change on PSO in 2018 [29]. It is stated that firstly PSO algorithms did not have the inertia weight, they had only acceleration constants. c_1 and c_2 indicate individual and communal parameters which are used for regulation. These parameters lead to jump the swarm particles to desired point with halfly reduced computing time. Then, parameters are selected as 2 both in this thesis. If this variables are set to a higher parameters, the swarm particles alter their positions fast [29], it can be cause of not to find the global minimum or maximum point. With the evolution of PSO, inertia weight concept got involved into PSO algorithms as a constant parameter which provides a momentum to swarm particles. Large inertia parameters consequence in straight particle paths with considerable passing over at the target. This leads to greater attribute global search. Whilst also low inertia values cause the particle trajectories sluggish with decreasing overshooting, thus, this conducts a good local search feature [29]. Then, inertia weights began to diversify over time. Linear inertia reduction and dynamic inertia methods were introduced in many researchs. In this study, position of each swarm particles which represent the PID parameters is limited with upper and lower boundaries and reduction of linear inertia method is used. Equation 4.6 shows the calculation of inertia weight in this thesis where w_h is higher inertia and w_l is lower.

$$\beta = (w_h - w_l)\left(\frac{n_{iteration} - i}{n_{iteration}}\right) + w_l \tag{4.6}$$

The principle behind this equation is to start with high inertia values to search better global minimum point. When the iteration is near to finalise, inertia weight is smaller and local searching is with around the global minimum point which is already found in beginning of the optimization. However it is stated that the downside of this linear decreasing weight method is to decrease capability of the global search with increasing iteration number [30].

5. SYSTEM APPLICATION

Informations of ACC control algorithm, vehicle model and PSO are given at previous chapters. In this chapter, it can be observed that all informations are fused and degraded to system application level. To see the application instance and results, vehicle model and control algorithm are connected and PSO algorithm is employed as a trigger condition for this connected model. With PSO algorithm, connected model is simulated $n_{iteration}\ x\ n_{swarm}$ times to find the parameters which provides the desired objective values. While simulating the model, some assumptions should be performed, parameters are determined due to corresponding reasons which will be introduced later. System application can be classified two main section which are integration of system and simulation scenarios.

5.1. System Integration

Before starting simulation activities, system must be integrated and unknown parameters must be criticized. First, lead vehicle speed trace must be determined. Then, how objective functions are defined is introduced and objective function for the PSO algorithm is explained. After objective functions explanation, some parameters which are coefficient values of PSO, iteration number, swarm particle number are determined. Finally, introduced model is integrated on Matlab/Simulink platform.

5.1.1 Lead vehicle speed trace

In this study, there is only one vehicle model for ego vehicle, and lead vehicle speed trace is given as external information. While generating this speed data, AVL VSM which is a packaged vehicle dynamics simulation platform is used [31].

AVL VSM is a comprehensive tool for simulating activities of complex vehicle dynamics. It is generally used for simulation of vehicle dynamics and driveability.

For completion of simulation, tool requires a extended information about the vehicle such as tyre size, coastdown parameters, suspensions, driveline and engine gearbox characteristics as demonstrated in Figure 5.1.

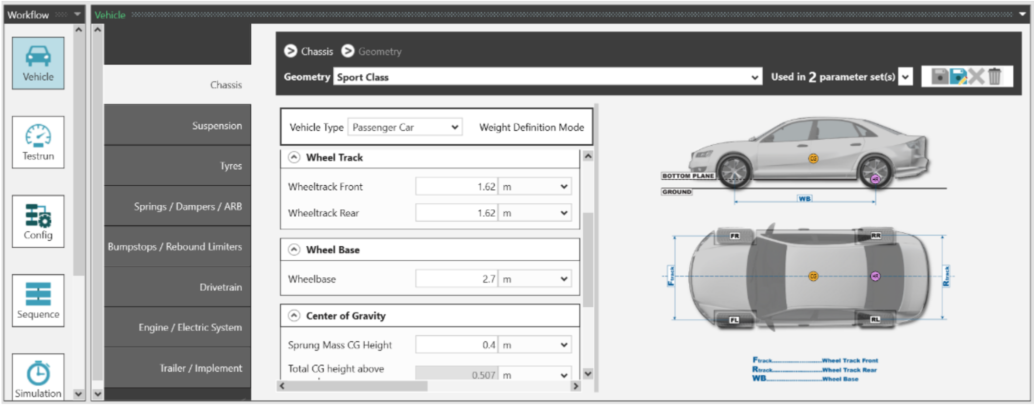


Figure 5.1: AVL VSM configuration screen.

Moreover, tool has also some default vehicle models inside it, models are on different types according to classes. AVL VSM includes a manouvre generation package also. Speed data for lead vehicle speed is generated using this package. Scenarios are generated with defined desired trace due to it contains acceleration and deceleration both. As shown in Figure 5.2. vehicle accelerates to 60 and 90 kph from standstill respectively, and after cruising some time, vehicle starts to decelerate.

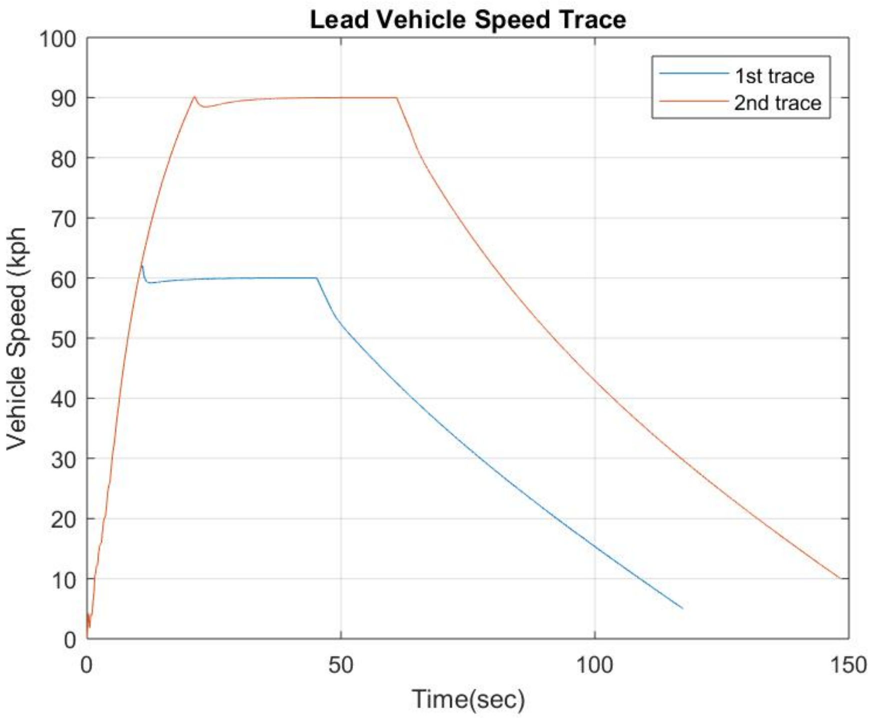


Figure 5.2: Lead vehicle speed traces.

With respect to different scenarios, these two vehicle speed traces are used for the lead vehicle behaviour demonstration.

Objective functions

Determination of objective functions on optimization problem is the most significant milestone in applications. There are various general objective functions in literature. It differs regarding the number of desired criterias. Once this number is only one, single-objective function is determined with corresponding methods which are root mean square (RMS), mean absolute error (MSE), average absolute deviation, mean squared deviation, squared deviations, integrated absolute error (IAE) etc.

Root mean square error ($RMSE$) is found as symbolised on equation 5.1 where the y_{act} is actual output value, y_{des} is desired output value and t_{tot} is the total time.

$$RMSE = \sqrt{\frac{\sum_0^{t_{tot}} (y_{act} - y_{des})^2}{t_{tot}}} \tag{5.1}$$

MSE is the mean of absolute errors in simulation with changing time. IAE is the summation of absolute errors as shown in equation 5.2.

$$IAE = \int_0^{t_{tot}} (y_{act} - y_{des}) dt \tag{5.2}$$

When number of objective values are greater than one, optimization problem turns into multi-objective. In multi-objective problems, desired values are weighted according to their priority. Prioritisation is performed with different methods in literature.

In the thesis, objective function of PSO algorithm is defined as single-objective. Area under the absolute error curve is optimised with IAE method which also contains the conventional PID error terms which are settling time, overshooting, rising time etc. Due to this fact, IAE is chosen as objective function. For CCC algorithm, error equation is defined in equation 5.3.

$$IAE_{CCC} = \int_0^{t_{tot}} \left|(V_{Set} - V_{Ego})\right| dt \tag{5.3}$$

For ACC algorithm, error equation is described in equation 5.4 where t_{Act} is actual relative time gap between ego and lead vehicle.

$$IAE_{ACC} = \int_0^{t_{tot}} |(t_{Set} - t_{Act})| dt \tag{5.4}$$

This objective error functions are optimized using the integrated model. Objective error functions determine the behaviour of optimization algorithm, in this thesis, it is observed and demonstrated that IAE method finds the parameters which satisfy the desired behaviour.

Parameters determination

Before simulation of scenarios, some parameters must be determined. In this integrated model, distance between ego and lead vehicle is determined as 40 meter. The threshold for vehicle detection is defined as 30 meter with a hysterisis between the ACC and CCC switching which is also explained before.

For the optimization algorithm; number of iterations (NoI), number of swarm particles (NoS), inertia weight and acceleration constants and upper and lower boundaries for PID parameters must be indicated. Iteration number are very significant parameter for optimization problems, because it has a impact directly on processing time. In this kind of problems, high processing time is an unintented circumstance.

NoI are determined with observing the convergency behaviour of the result. For this work, processing time is not main concern, due to this fact, NoI generally changes between 100 and 150 in this process. However, various stopping conditions should be applied the application, for an instance, if the objective value stays same with defined interval of NoI such as 20 iterations, optimizations should be stopped.

NoS are the other crucial parameter for PSO, once the NoS increases, simulation size raises and this leads more accurate optimization but also more processing time. Since the defined objective function is single-objective and constrained with a specific problem, NoS are chosen as 3. Especially for ACC system, actual time gap has not be below 0.8 seconds. With chosen parameters NoS and NoI, time gap never undershoots 0.9 seconds which can be interpreted as highly safe operation.

Inertia weight was described in previous chapter, while determining the inertia weight, linearly decreasing inertia weight method is used. And upper value and lower value are selected as 0.9 and 0.4 respectively. Acceleration constants are also defined in previous chapter and chosen as 2 both.

Finally, upper and lower boundaries of ACC and CCC control parameters are selected with using the global best values curve. Once the optimization process is completed, if the parameters are saturated with upper and lower boundaries, boundaries are expanded. Upper values of K_v and K_t are selected as 1000 and 5000 respectively for ACC. In the CCC simulations, upper boundaries of K_p, K_i and K_d parameters are defined as 7000, 1000 and 3000. Lower boundaries are 0 for all parameters. These boundaries leads avoiding jump through unexpected positions of each swarm particles. With using these defined constraints, optimization algorithm is employed.

Matlab/Simulink integration

In order to finalise simulation platform; vehicle model, controllers and PSO algorithm are integrated on Matlab/Simulink platform. As pointed out in Figure 5.3 and 5.4, CCC and ACC integrated model flows use the PSO algorithm to find the desired parameters with regarding objective function.

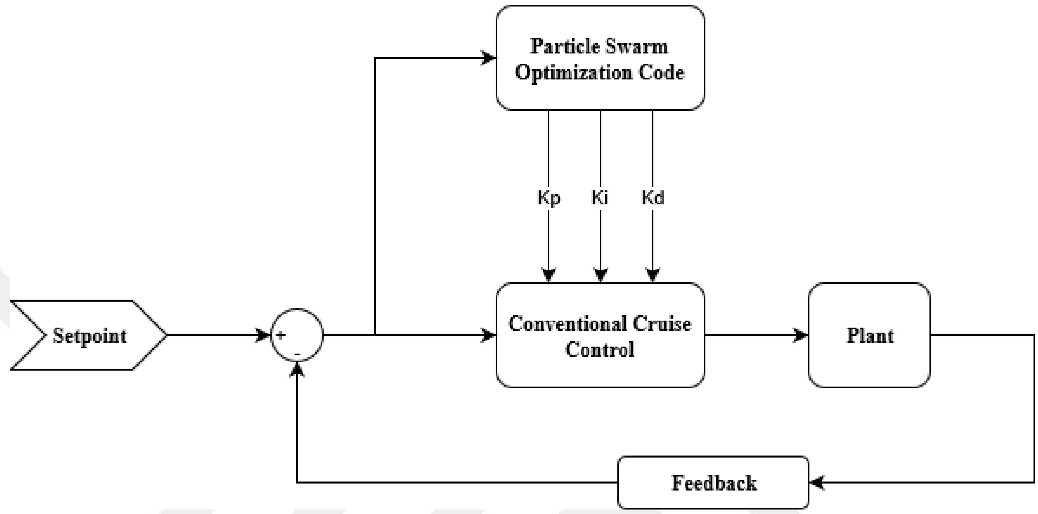


Figure 5.3 : CCC optimization process flow.

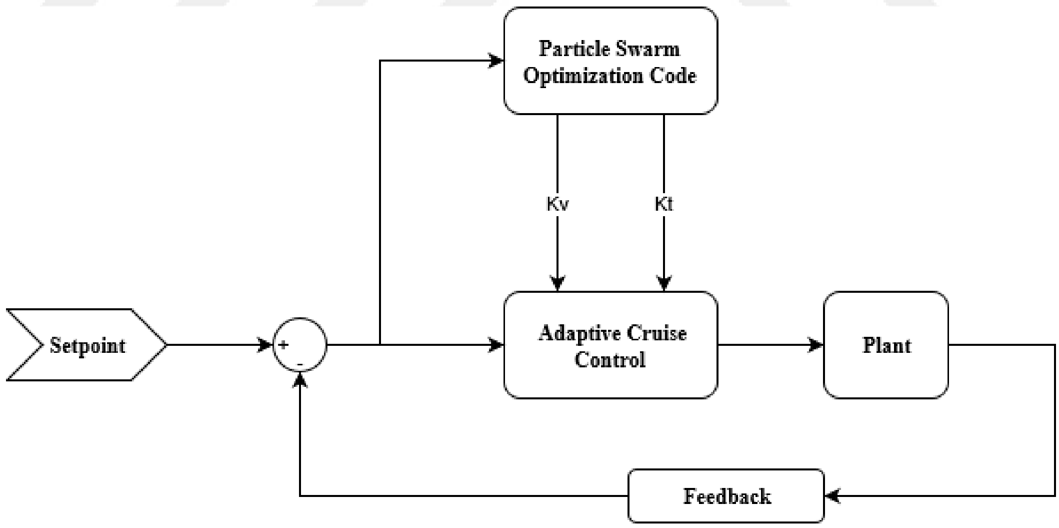


Figure 5.4 : ACC optimization process flow.

Vehicle model and controllers are designed in Simulink platform. PSO algorithm is created as Matlab function and in every iteration for every swarm particles, code calls the integrated model for the simulation and gets the results.

According to results PSO calculates the next movement of every swarm particles until iterations are completed. Integrated model also can be shown in Figure 5.5.

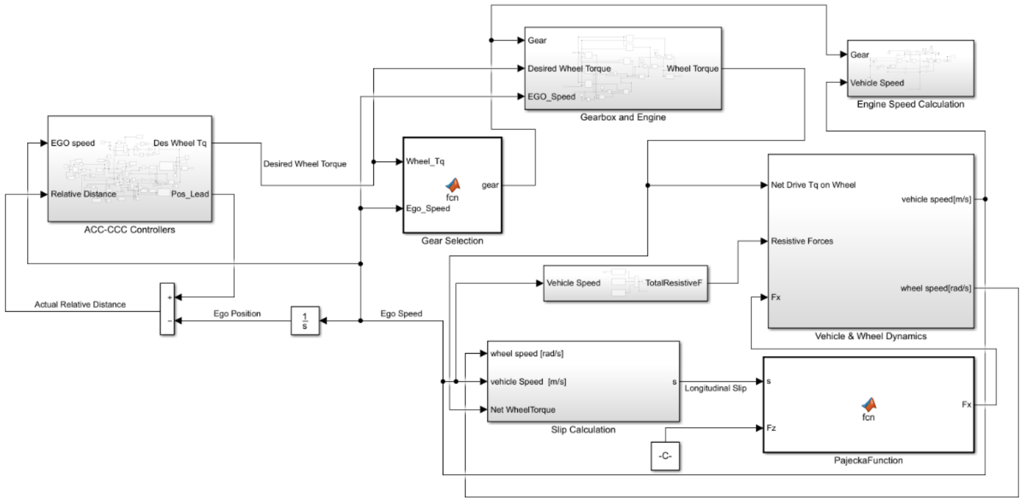


Figure 5.5: Complete integrated simulation model.

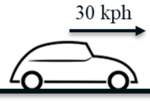
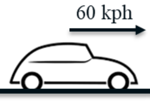
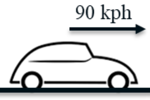
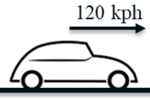
5.2. Scenarios

In this section, all scenarios for each control mode ACC and CCC are described. ACC and CCC also can work combined, scenarios related with combined mode are also introduced. ACC and CCC performances are interpreted with related scenarios.

5.2.1. Cruise control scenarios

CCC is simulated with four different scenarios which are represented in Table 5.1. These scenarios are generated with various speeds. The main reason behind this is to understand effect of vehicle speed on controller parameters.

Table 5.1 : CCC scenarios.

NO	SCENARIOS	DESCRIPTION
1	30 kph 	Active CCC with 30 kph set speed. No lead vehicle detected.
2	60 kph 	Active CCC with 60 kph set speed. No lead vehicle detected.
3	90 kph 	Active CCC with 90 kph set speed. No lead vehicle detected.
4	120 kph 	Active CCC with 120 kph set speed. No lead vehicle detected.

Results of only second scenario is shown in this section, rest of the results are demonstrated in Appendix B.1. In the beginning of second scenario, ego vehicle is in standstill. CCC set speed is 60 kph and vehicle accelerates through 60 kph and continues cruising with same speed. Due to shifting point as 2500 rpm, vehicle acceleration is limited. Usage of engine map is also limited due to this reason. Vehicle speed trace can be shown in Figure 5.6. No vehicle speed overshoot is observed. Rising time is saturated with vehicle dynamics. Settling time is almost zero.

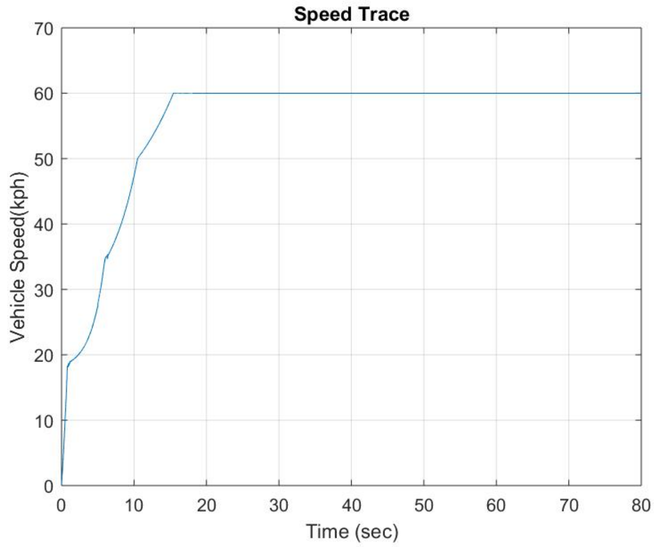


Figure 5.6 : Speed trace of CCC second scenario.

Figure 5.7 represents the iteration results. Axises of this graph are global best objective values and NoI. With increasing NoI, objective values decreases as expected. Optimization algorithm tries to provide convergence to zero.

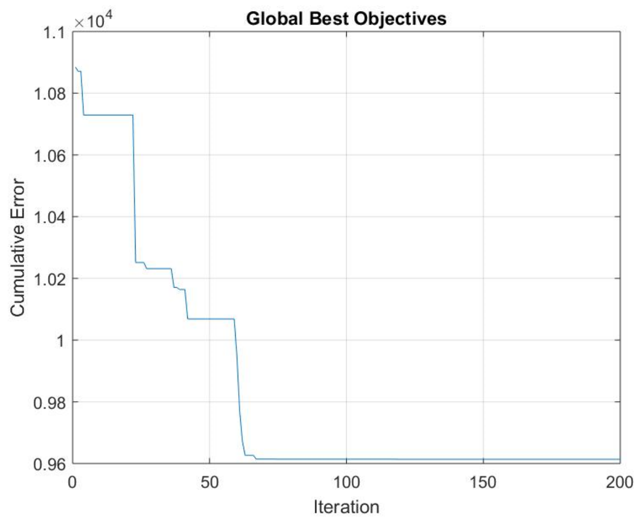


Figure 5.7 : Global best objectives of CCC second scenario.

Figure 5.8 demonstrates the engine speed behaviour. Once the engine speed reachs 2500 rpm, upshift is performed. Then, engine speed starts to increase through 2500 rpm again.

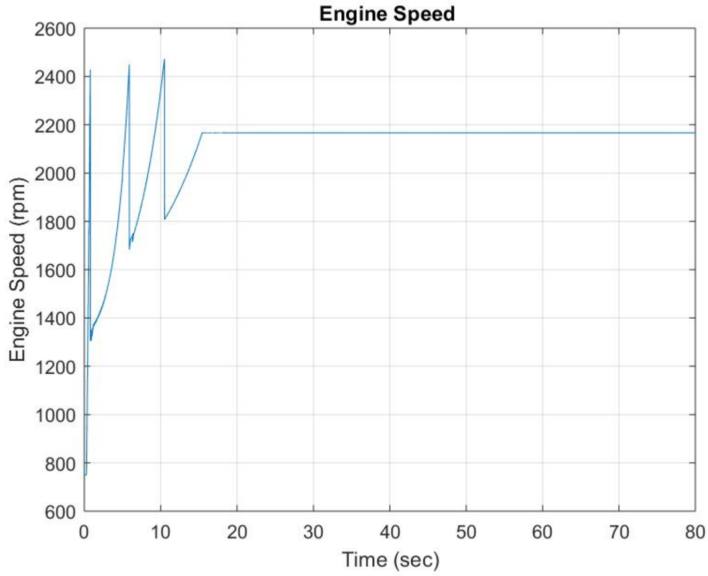


Figure 5.8 : Engine speed behaviour of CCC second scenario.

Figure 5.9 and Figure 5.10 represents the gear positions and wheel torque respectively. As mentioned before, gear positions change according to determined vehicle speed ranges. In the wheel speed graph, there is an observed oscillation at 15th second. It is the point where the vehicle speed settles to 60 kph. To prevent overshoot and oscillation, wheel torque output oscillates with controller inputs. But it is not observed on vehicle speed and engine speed both.

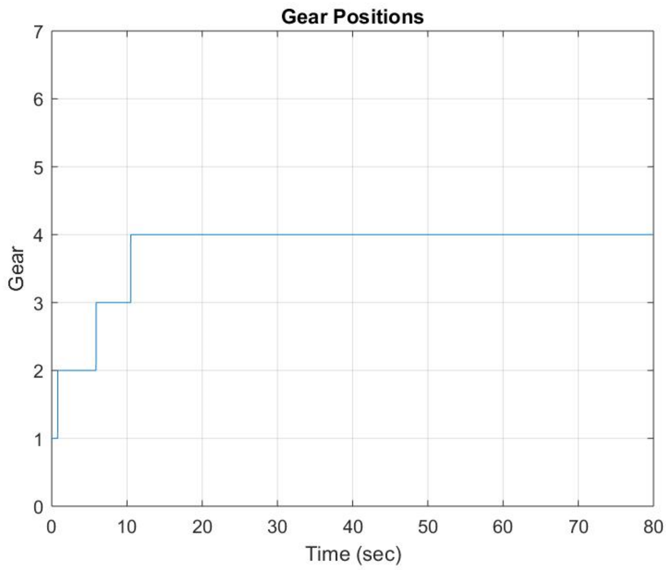


Figure 5.9 : Gear positions of CCC second scenario.

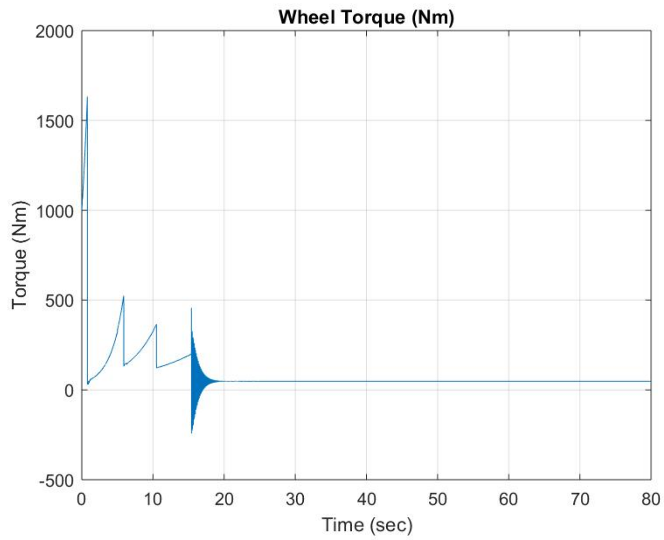


Figure 5.10 : Wheel torque behaviour of CCC second scenario.

All CCC scenarios are similar. The optimization algorithm results can be found in Table 5.2.

Table 5.2 : Calibrated PID values for CCC scenarios.

	K_p	K_i	K_d
Scenario 1	32.1	0	3.7
Scenario 2	246.2	0	63
Scenario 3	549.7	0	132
Scenario 4	544.8	0	758.7

With regarding this graph, it can be interpreted that integral controller is not required for this control problem. For further studies, gain scheduled controller can be employed for this problem with respect to vehicle speed error change.

5.2.2. Adaptive cruise control scenarios

Two different scenarios for ACC are described in this section. These scenarios are generated with using two different speed trace of lead vehicle. When the inter-vehicle distance is 40 meter, lead vehicle accelerates through 60 kph and 90 kph respectively for fifth and sixth scenarios. Ego vehicle tries to follow lead vehicle. In this scenarios, only ACC is evaluated.

CCC intervention is not active for these scenarios. Only the results of fifth scenarios are demonstrated in here, the sixth one is included into Appendix B.2. Table 5.3 shows the scenarios.

Table 5.3 : ACC scenarios.

NO	SCENARIOS	DESCRIPTION
5		ACC active. Lead vehicle accelerates to 60 kph.
6		ACC active. Lead vehicle accelerates to 90 kph.

Figure 5.11 represents the vehicle speed traces of lead and ego vehicle. Ego vehicle has two different speed trace which are calibrated with PSO and calibrated with base parameters. Base parameters are 10 both K_v and K_t. It is a proof of what PSO contributes to calibration or tuning of controller parameters.

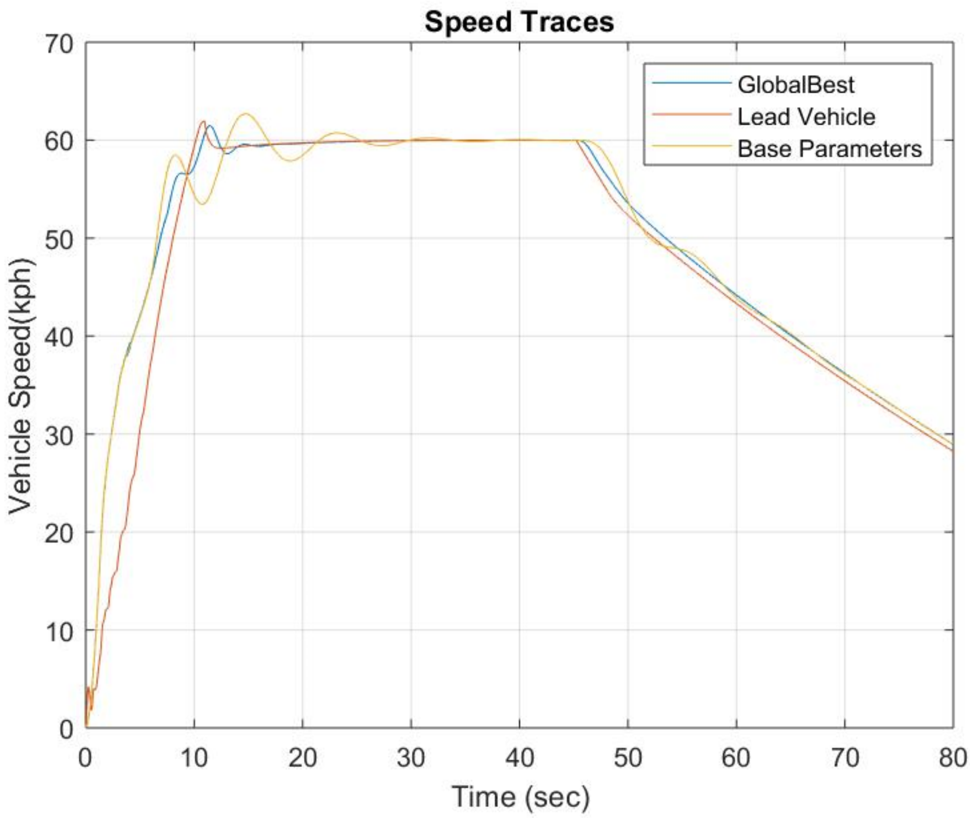


Figure 5.11 : Speed traces for ACC fifth scenario.

Figure 5.12 demonstrates the convergency of iteration results. The main control objective is to provide the set time gap which is 1 second. Figure 5.13 shows the actual relative time gap between two vehicles. It never undershoots 0.8 second which is critical limit.

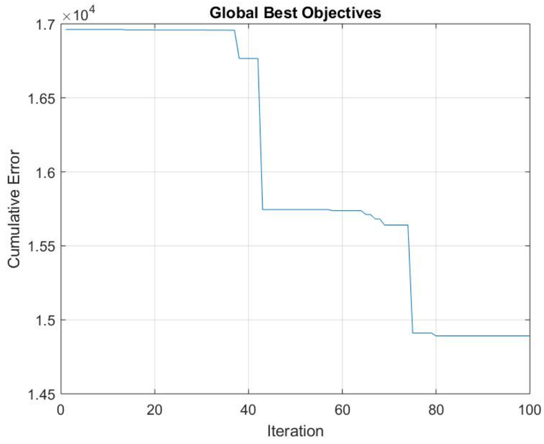


Figure 5.12 : Global best objectives of ACC fifth scenario.

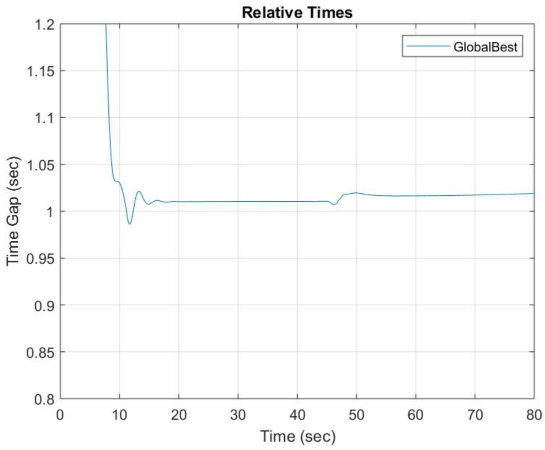


Figure 5.13 : Actual relative time gap of ACC fifth scenario.

Also with parallel to time gap behaviour, relative distance and actual distance comparison can be shown in Figure 5.14. Inter-vehicle distance decreases through desired relative distance from 40 meter which is initial distance between vehicles.

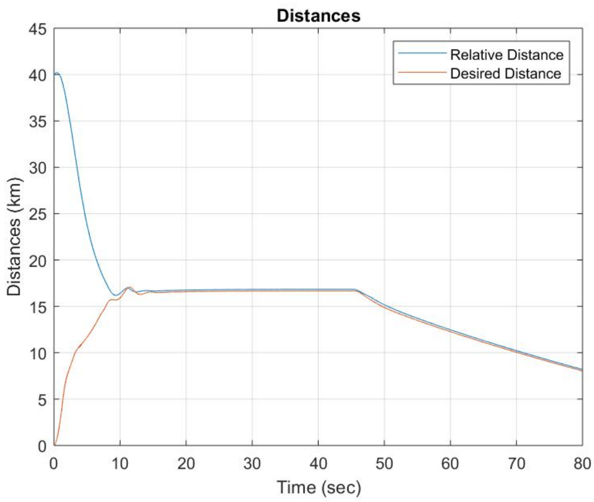


Figure 5.14 : Relative distances of ACC fifth scenario.

Figure 5.15 and 5.16 show behaviour of engine speed and gear positions respectively. Ego vehicle can reach fast to the desired time gap but limited vehicle capability limits acceleration in this manouvre also.

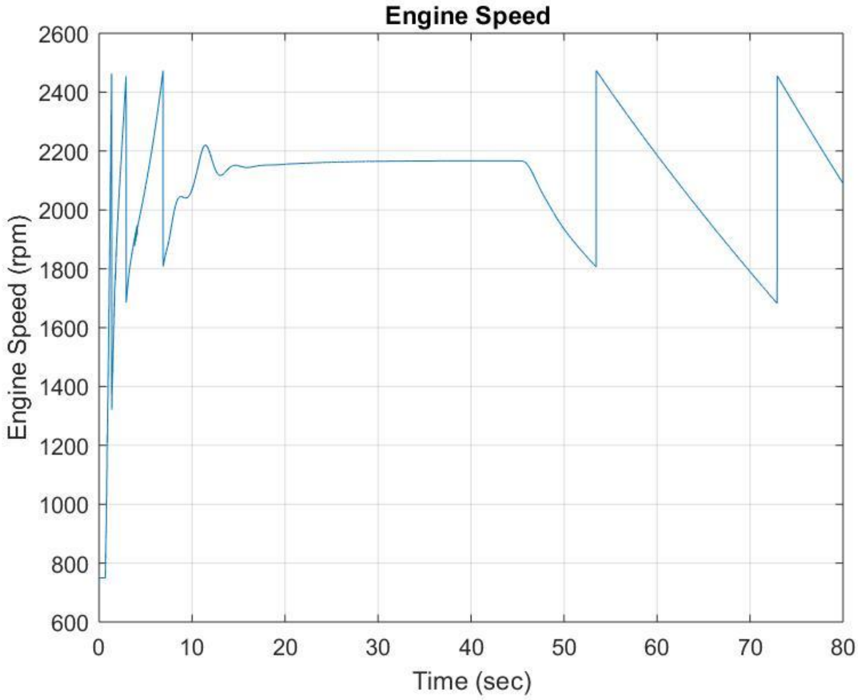


Figure 5.15 : Engine speed behaviour of ACC fifth scenario.

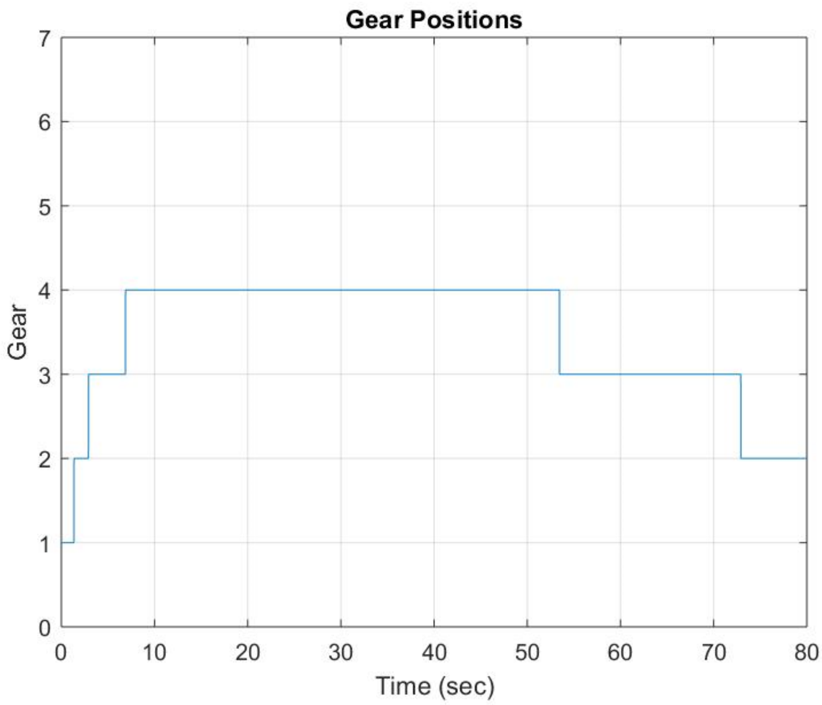


Figure 5.16 : Gear positions of ACC fifth scenario.

Figure 5.17 and 5.18 also show wheel torque and forces on longitudinal axis.

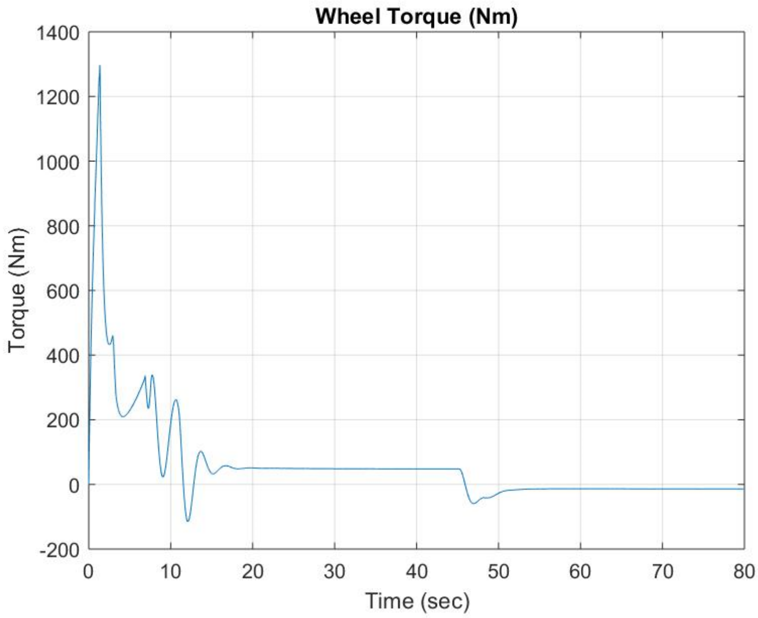


Figure 5.17 : Wheel torque behaviour of ACC fifth scenario.

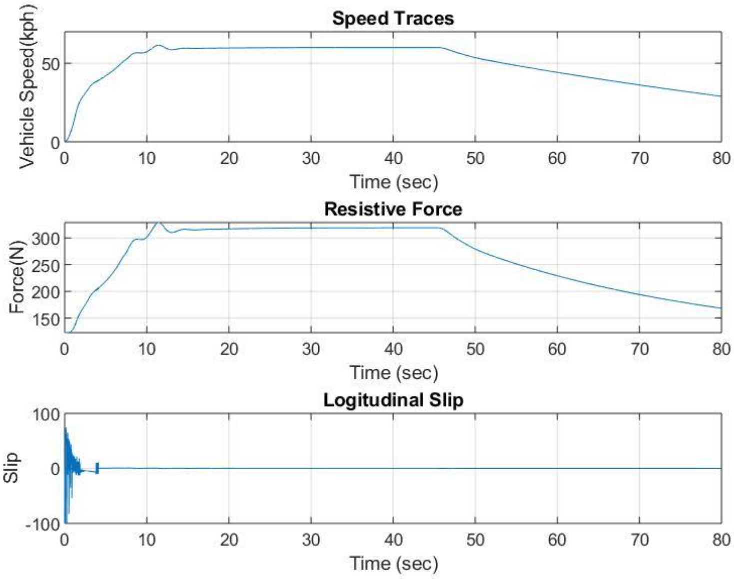


Figure 5.18 : Longitudinal speed, resistive force and slip of ACC fifth scenario.

Parameter tuning results of these two ACC scenarios are described in Table 5.4.

Table 5.4 : Calibrated PD values for ACC scenarios.

	K_v	K_t
Scenario 5	286.3	27.2
Scenario 6	926.6	27.3

This table shows that K_t almost does not change with different scenarios. But for K_v varies regarding speed target change. For CCC and ACC, tuning parameters are used in combined scenarios.

5.2.3. ACC-CCC scenarios

In the real world, ACC and CCC works combined. If there is no detected lead vehicle, CCC works and once the lead vehicle is detected, ACC activates. Table 5.5 shows that whole scenarios for combined mode.

Table 5.5 : ACC-CCC combined scenarios.

NO	SCENARIOS	DESCRIPTION
7		Active CCC with 30 kph set speed. Lead accelerates to 60 kph.
8		Active CCC with 90 kph set speed. Lead accelerates to 60 kph.
9		Active CCC with 60 kph set speed. Lead accelerates to 90 kph.
10		Active CCC with 120 kph set speed. Lead accelerates to 90 kph.

Main results of seventh and eighth scenarios are introduced in this section. Remaining results and graphs are shown in Appendix B.3.

In remaining scenarios, lead vehicle accelerates through 90 kph and CCC set speeds are 60 and 120 kph for nineth and tenth scenarios respectively.

In the seventh and eighth scenarios both, lead vehicle accelerates through 60 kph, different is the initial set CCC speeds. In the seventh scenario, set speed of CCC is 30 kph and ego vehicle does not reach and detect the lead vehicle.

However, set speed of CCC is 90 kph in the eighth scenario, ego vehicle tries to accelerate through 60 kph. With detection of lead vehicle, CCC is deactivated and ACC is activated. Ego vehicle follows the lead vehicle with the objective which is to control actual time gap as 1 second. Figure 5.19 shows the vehicle speed graphs for seventh scenario.

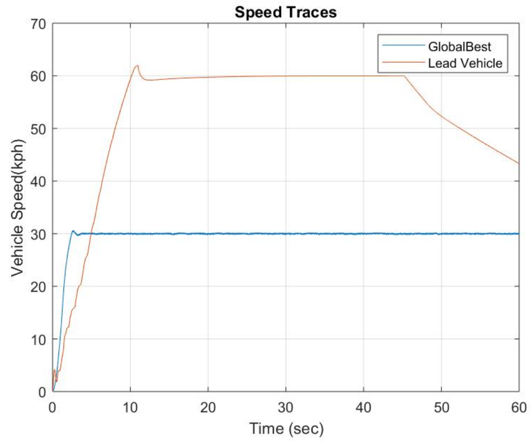


Figure 5.19 : Speed traces of seventh scenario.

CCC switch is always active because following vehicle does not detect to lead vehicle. Figure 5.20 represents the switch and positions of vehicles.

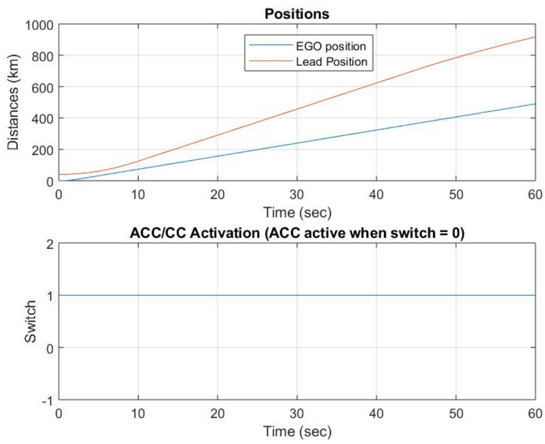


Figure 5.20 : ACC/CCC switching of seventh scenario.

Time gap increases when the lead vehicle accelerates. Figure 5.21 shows increase of time gap.

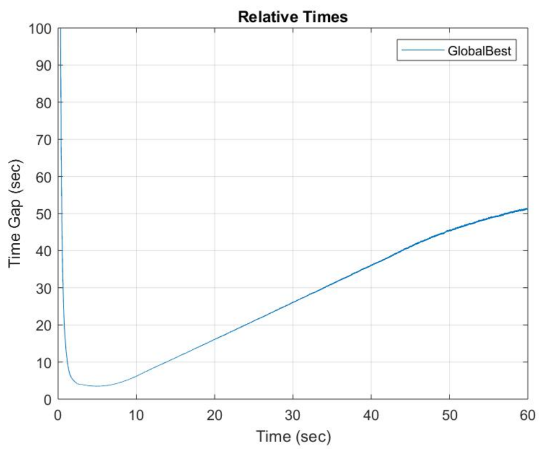


Figure 5.21 : Actual relative time gap of seventh scenario.

For the eighth scenario, Figure 5.22 shows the vehicle speed traces. ACC algorithm provides to reach the desired distance between the vehicles.

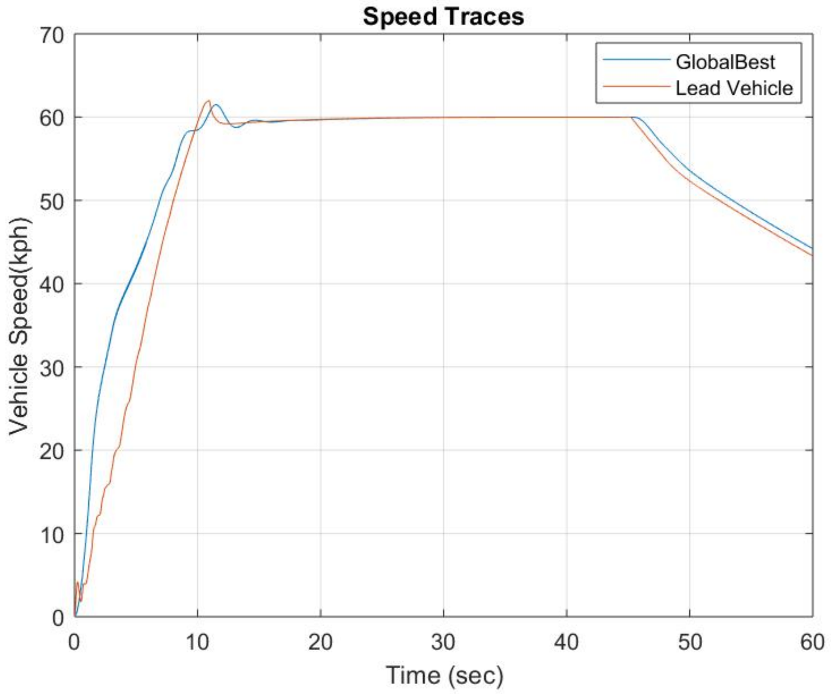


Figure 5.22 : Speed traces of eighth scenario.

As shown in the Figure 5.23, when the distance between the vehicles reach the threshold value for CCC to ACC switching, ACC activates.

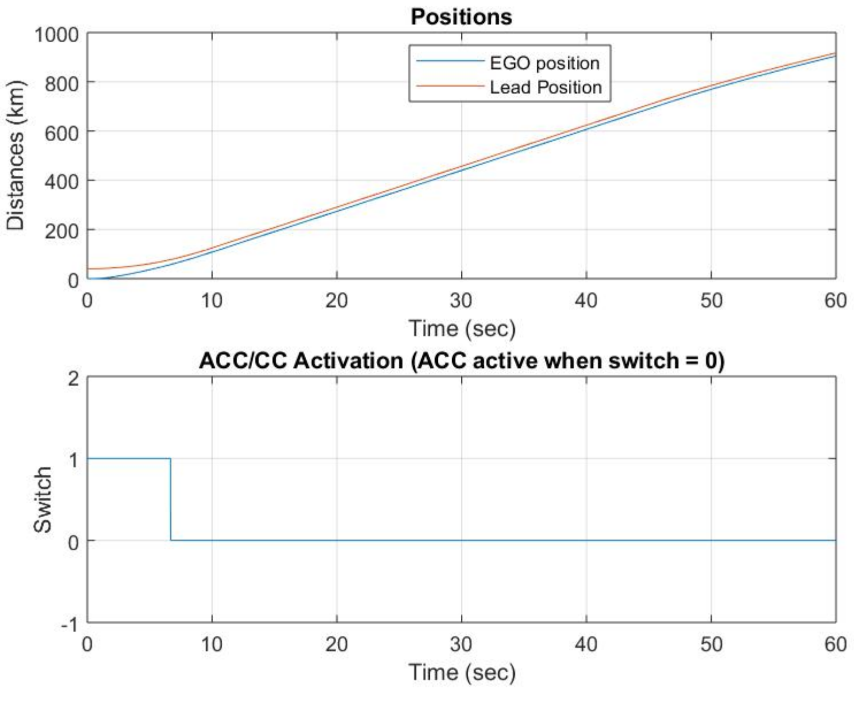


Figure 5.23 : ACC/CCC switching of eighth scenario.

Figure 5.24 represents the time gap behaviour. Time gap is controlled around 1 second as intented.

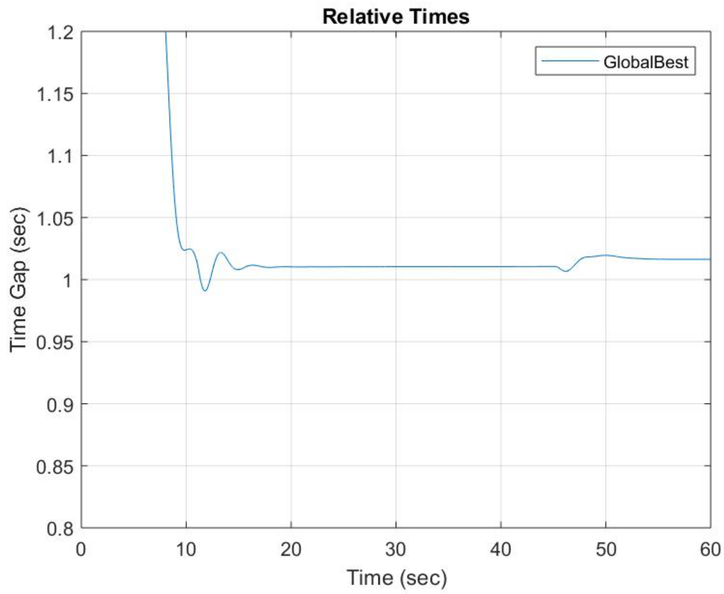


Figure 5.24 : Actual relative time gap of eighth scenario.

In these combined scenarios, parameters which are found by PSO algorithm are used for relevant scenarios. For an instance, for 30 kph set speed of CCC, relevant optimized parameters are used, and also for 60 kph speed trace of lead vehicle, ego vehicle uses the relevan tuned paramaters.

As a brief result, PSO algorithm gives desired behaviours of objective function with using the vehicle and controller model on Matlab/Simulink. This reduces the tuning time and leads getting more stabilized results.

6. CONCLUSION AND RECOMMENDATIONS

In this thesis, ACC system is designed and connected with one-mass longitudinal vehicle model which includes resistive forces, tire model, engine and gearbox models. PSO algorithm is briefly explained and the scope of thesis is to optimize the designed ACC controller parameters with using connected model and PSO algorithm. Results are introduced and interpreted on Chapter 5. According to results, PSO method leads to substantiate the intented behaviours of ACC and CCC controller on vehicle level since it is one of the global search optimization methods and swarm intelligence optimization. With increasing robustness on controller system, calibration activities gain more importance. Especially, this increase of robustness causes to raise the number of desired intention subject. Moreover, optimization problems turn into multi-objective problems more. PSO is an important algorithm to achieve better results in multi-objective problems. ACC and CCC can be also called as multi-objective, however in this thesis, they are assumed as single-objective. The reason behind this is that simulations are conducted in limited scenarios, and IAE almost covers the cost functions of PID controllers.

In this thesis, specific scenarios are generated according to real world scenarios. Various conditions are simulated. ADAS features are being developed day by day. Then, simulation platforms also gain more importance for this kind of optimization problems.

For further activities, it is suggested that ACC and CCC algorithms should be optimized with a multi-objective function whose weights are determined according to desired behaviour. In contrary to one-mass vehicle model, it should be more complicated and realistic with adding suspension and driveline characteristics. Then, this algorithm can be employed also for real vehicle testing.

In real world, results can differ from simulation platforms due to sensor noises, road types and curvatures, vehicle dynamics etc. The other suggestion is to design more complicated controller which can respond every conditions. Gain scheduling can be used for this. Finally, to improve the performance of PSO, dynamic inertia weight can be used.

REFERENCES

[1] **Bengler, K., Dietmayer, K., Färber, B., Maurer, M., Stiller, C., Winner, H.** (2014). Three Decades of Driver Assistance Systems – Review and Future Perspectives, *IEEE Intelligent Transportation Systems Magazine,* Volume 6, No: 4, pp. 6-22.

[2] **Kural, E.** (2006). Adaptif Seyir Sistemleri (Master's Thesis). Istanbul Technical Univesity, Graduate School of Science Engineering and Technology.

[3] **Xiao, L., Gao, F.** (2010). A Comprehensive Review of the Development of Adaptive Cruise Control Systems, *International Journal of Vehicle Mechanics and Mobility,* Volume 48, No: 10, pp. 1168-1170.

[4] **Holzman, H., Halfmann, C., Isermann, R.** (1999). Model-based Optimization of Vehicle Longitudinal Motion Controllers, *14th Triennial World Congress of The International Federation of Automatic Control (IFAC)*.

[5] **Moon, S., Yi, K., Moon, I.** (2008). Design, Tuning and Evaluation of Integrated ACC/CA Systems, *17th World Congress of IFAC.*

[6] **Root, M.K., Sain, D., Swain, S. K., Mishra, S.K.** (2016). PID Controller Design for Cruise Control System using Genetic Algorithm, *International Conference on Electrical, Electronics and Optimization Techniques (ICEEOT)*.

[7] **Abdulnabi, A.R.** (2017). PID Controller Design for Cruise Control System using Particle Swarm Optimization, *Iraqi Journal for Computers and Informatics (IJCI),* Volume 43, Issue 2, pp. 30-33.

[8] **Pradhan, R., Majhi, S. K., Pradhan, J. K., Pati, B. B.** (2018). Antlion Optimizer tuned PID Controller Based on Bode Ideal Transfer Function for Automobile Cruise Control Systems, *Journal of Industrial Information Integration,* Volume 9, pp. 45-52.

[9] **Luo, L., Liu, H., Wang, H., Li, P.** (2010). Model Predictive Control for Adaptive Cruise Control with Multi-Objectives: Comfort, Fuel-Economy, Safety and Car-Following, *Journal of Zhejlang University,* pp. 191-194.

[10] **Zhang, J., Li, Q., Chen, D.** (2018). Integrated Adaptive Cruise Control with Weight Coefficient Self-Tuning Strategy, *Applied Science* 2018; 8: 978.

[11] **Winner, H., Hakuli, S., Lotz, F., Singer, C.** (2016). *Handbook of Driver Assistance Systems: Basic Information*, *Components and Systems for Active Safety and Comfort,* Germany.

[12] **5th Meeting of the U.S. Software System Safety Working Group.** (2005, 12-15 April). *Adaptive Cruise Control System Overview.*

[13] **BS ISO.** (2018). *Intelligent Transport Systems – Adaptive Cruise Control Systems – Performance Requirements and Test Procedures (BS ISO 15622:2018).*

[14] **Zhang, G., Wang, Z., Fan, B., Zhao, L., Qi, Y.** (2018). Adaptive Cruise Control System with Traffic Jam Tracking Function Based on Multi-sensors and the Driving Behaviour of Skiller Drivers, *Advances in Mechanical Engineering.*

[15] **Kocic, J., Jovicic, N., Drndarevic, V.** (2018). Sensors and Sensor Fusion in Autonomous Vehicle, *6th Telecommunications Forum (TELFOR)*, pp. 420-425.

[16] **He, Y. et al.** (2019). Adaptive Cruise Control Strategies Implemented on Experimental Vehicles: A Review, *IFAC,* Volume 52, Issue 5, pp. 21-27.

[17] **Ioannou, P. and Xu, Z., Eckert, S, Clemons, D. and Sieja, T.** (1993). Intelligent Cruise Control: Theory and Experiment, *Proceedings of the 32nd Conference on Decision and Control,* pp. 1885-1890.

[18] **Flores, C., Milanes, V., Nashashibi, F.** (2017). A Time Gap-Based Spacing Policy for Full-Range Car-Following, *Intelligent Transportation Systems Conference 2017*, Japan.

[19] **Özdemir, S.** (2015). Model Öngörülü Kontrol ve Bir Endüstriyel Model Öngörülü Kontrol Uygulaması (Master's Thesis). Istanbul Technical Univesity, Graduate School of Science Engineering and Technology.

[20] **SAE.** (2008). *Vehicle Dynamics Terminology (J670-200801).*

[21] **Jazar, R. N.** (2017). *Vehicle Dynamics: Theory and Application.*

[22] **Cuellar, A. Y. et al.** (2013). PID-Controller Tuning Optimization with Genetic Algorithms in Servo Systems, *Internation Journal of Advanced Robotic Systems,* Volume 10.

[23] **Bansal, H. O., Sharma, R., Shreeraman, P. R.** (2012). PID Controller Tuning Techniques, *Journal of Control Engineering and Technology (JCET),* Volume 2, Issue 4, pp. 168-176.

[24] **Ogata, K.** (2004). *System Dynamics, Upper Saddle River, NJ :Pearson/Prentice Hall.*

[25] **Nagaraj, B., Murugananth, N.** (2010). A Comparative Study of PID Controller Tuning using GA, EP, PSO and ACO, *International Conference on Communication Control and Computing Technologies 2010,* pp. 305-313.

[26] **Erdoğmuş, P.** (2018). *Particle Swarm Optimization with Applications*.

[27] **Url-1** <*joshkovitz.com/research/projects/optimization-in-electromagnetics/*>, date retrieved 10.11.2019.

[28] **Özsağlam, M. Y., Çunkaş, M.** (2008). Optimizasyon Problemlerinin Çözümü için Parçacık Sürü Optimizasyonu Algoritması, *Journal of Polytechnic,* Volume 11, No:4, pp. 299-305.

[29] **El-Shorbagy, M. A., Hassanien, A. E.** (2018). Particle Swarm Optimization from Theory to Applications, *Internation Journal of Rough Sets and Data Analysis,* Volume 5, Issue 2.

[30] **Arasomwan, M. A., Adewumi, A. O.** (2013). On the Performance of Linear Decreasing Inertia Weight Particle Swarm Optimization for Global Optimization, *The Scientific World Journal 2013*.

[31] **Url-2** *<https://www.avl.com/tr/-/avl-vsm-4->*, date retrieved 10.11.2019.

[32] **Url-3** *< http://www.racer.nl/reference/pacejka.htm>*, date retrieved 04.11.2019.

APPENDICES

APPENDIX A : Magic tire formula calculation code

Calculation formula of longitudinal tire force with Pajecka tire model is given with Figure A.1 in below as a code [32].

```
function Fx = fcn(s,Fz)

b0=1.65;
b1=0;
b2=1688;
b3=0;
b4=229;
b5=0;
b6=0;
b7=0;
b8=-10;
b9=0;
b10=0;
b11=0;
b12=0;


C=b0;
mb=b1*Fz+b2;
D=mb*Fz;
BCD=(b3*Fz^2+b4*Fz)*exp(-b5*Fz);
B=BCD/(C*D);
E=b6*Fz^2+b7*Fz+b8;
sh=b9*Fz+b10;
sv=0;

Fx=D*sin(C*atan(B*(s+sh)-E*(B*(s+sh)-atan(B*(s+sh)))))+sv;
```

Figure A.1 : Matlab code of Magic Tire Formula.

APPENDIX B : Scenario simulation results

Some of the scenarios' results are not given in Chapter 5. Remaining results are given in Appendix B. CCC, ACC and combined scenarios are demonstrated in different sections.

B.1 : CCC scenarios

In the below for CCC; first, third and fourth scenarios' results are shown respectively. Figure B.1 shows the speed traces for first CCC scenario which can be described as 30 kph set speed manouvre. Base parameters are 10 for all elements of PID controller.

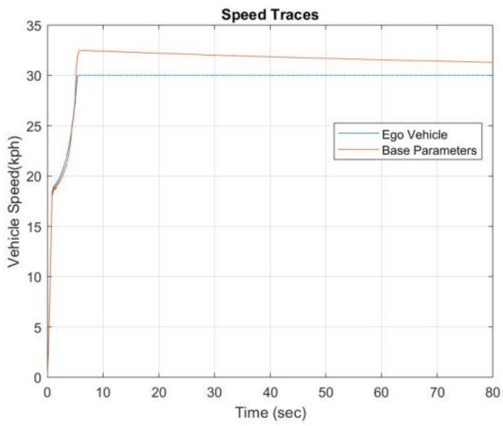


Figure B.1 : Speed traces of CCC first scenario.

Figure B.2 and B.3 demonstrates the global best objective values and wheel torque respectively.

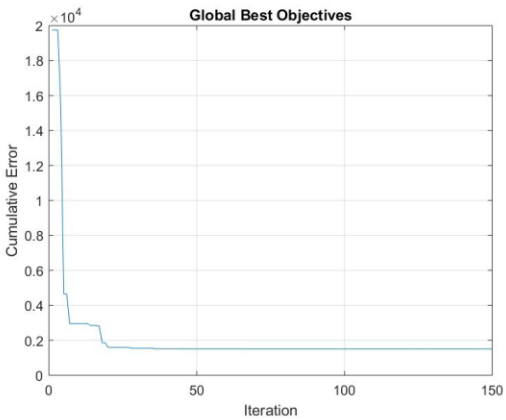


Figure B.2 : Global best objectives of CCC first scenario.

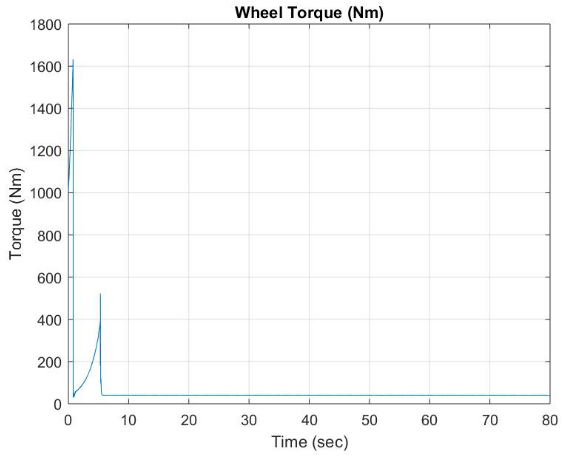


Figure B.3 : Wheel torque behavior of CCC first scenario.

Figure B.4 and B.5 shows the engine speed and gear positions respectively.

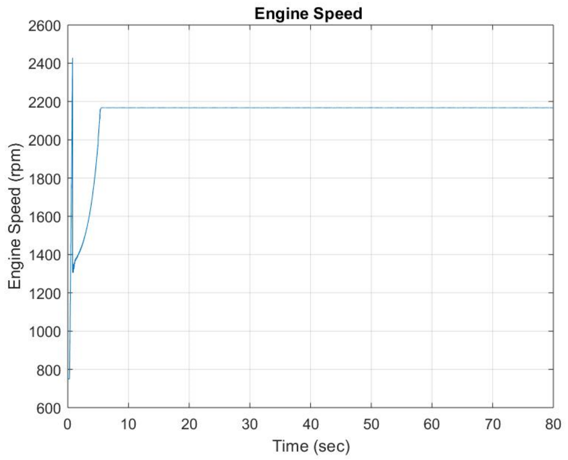


Figure B.4 : Engine speed behaviour of CCC first scenario.

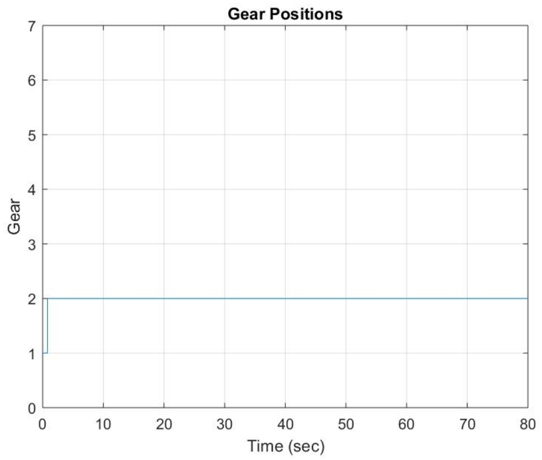


Figure B.5 : Gear positions of CCC first scenario

Figure B.6 and B.7 indicates the vehicle speed trace and global best objectives of CCC third scenario where CCC set speed is 90 kph.

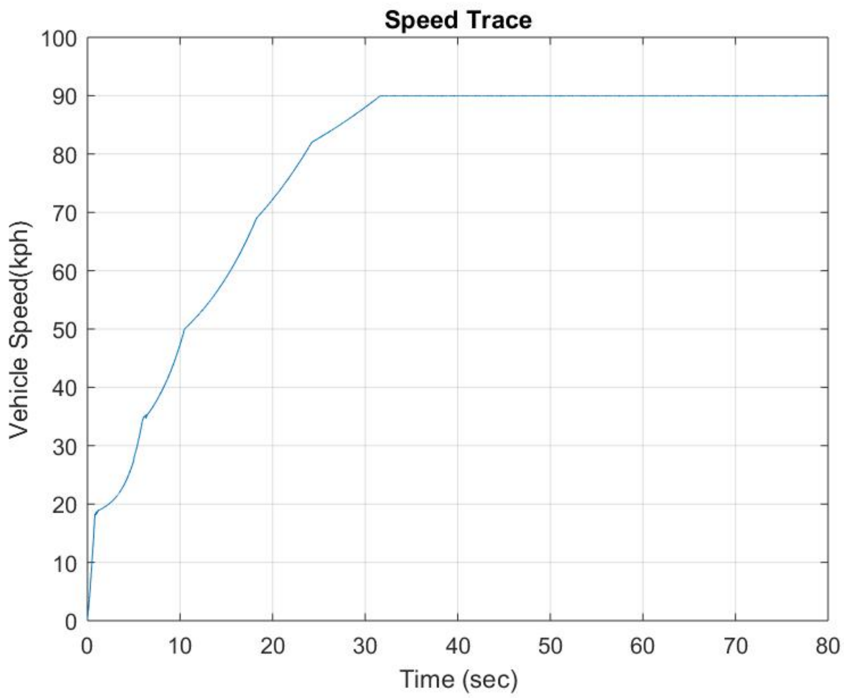


Figure B.6 : Vehicle speed trace of CCC third scenario.

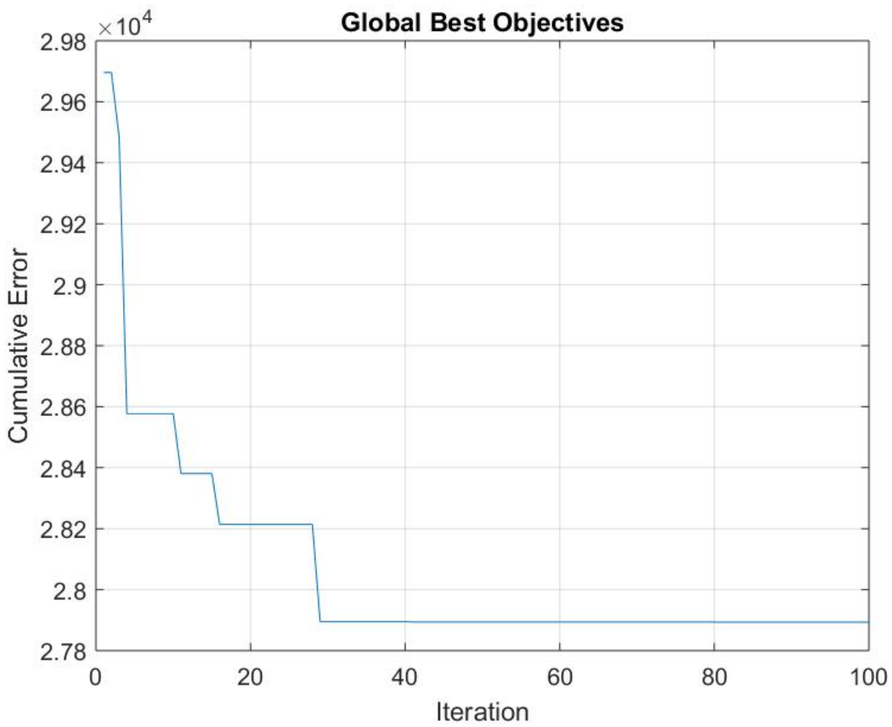


Figure B.7 : Global best objectives of CCC third scenario.

Figure B.8 and B.9 points out the engine speed and gear positions behaviour of CCC third scenario.

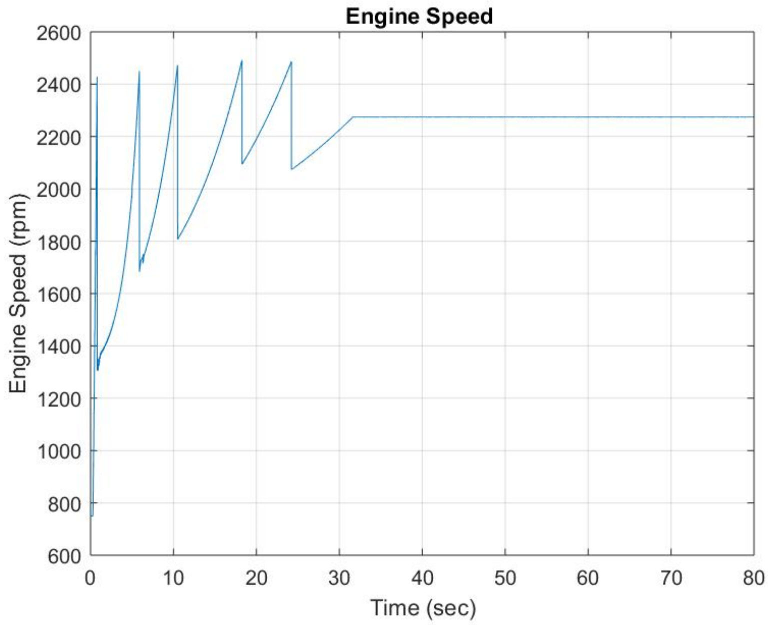


Figure B.8 : Engine speed behaviour of CCC third scenario.

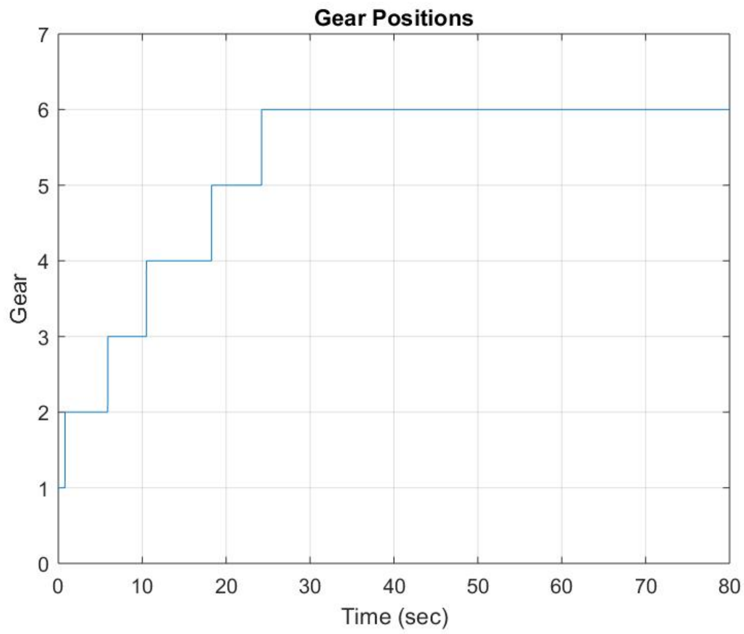


Figure B.9 : Gear positions of CCC third scenario.

Figure B.10 shows the vehicle speed trace for CCC fourth scenario where set speed is 120 kph. Respectively Figure B.11, B.12, B.13 represent the global best objectives, engine speed behaviour and gear positions of CCC fourth scenario.

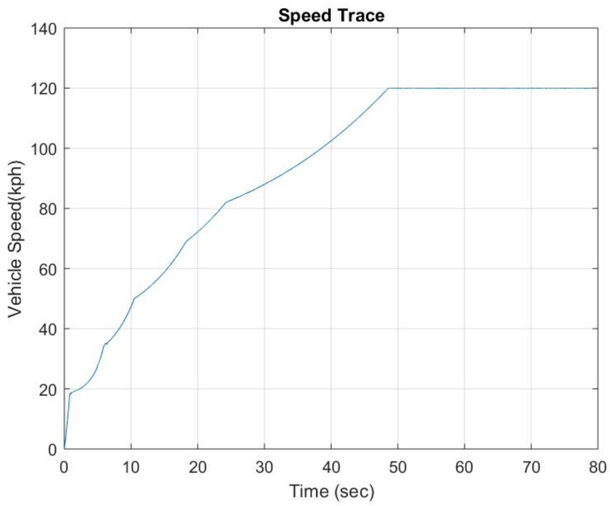


Figure B.10 : Vehicle speed trace of CCC fourth scenario.

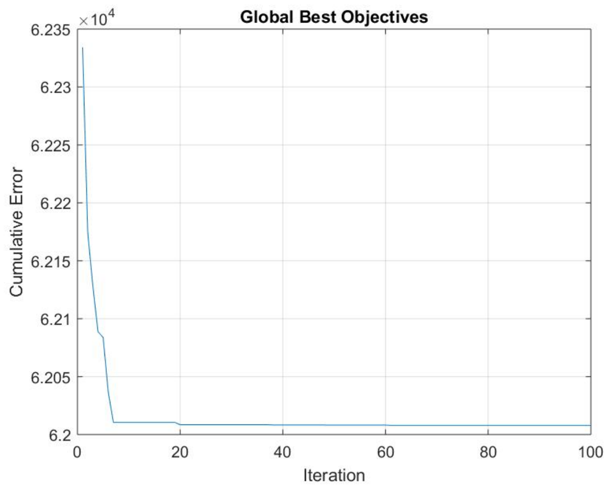


Figure B.11 : Global best objectives of CCC fourth scenario.

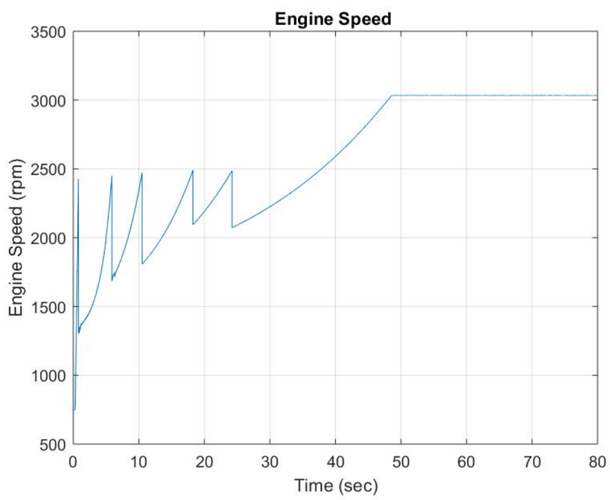


Figure B.12 : Engine speed behaviour of CCC fourth scenario.

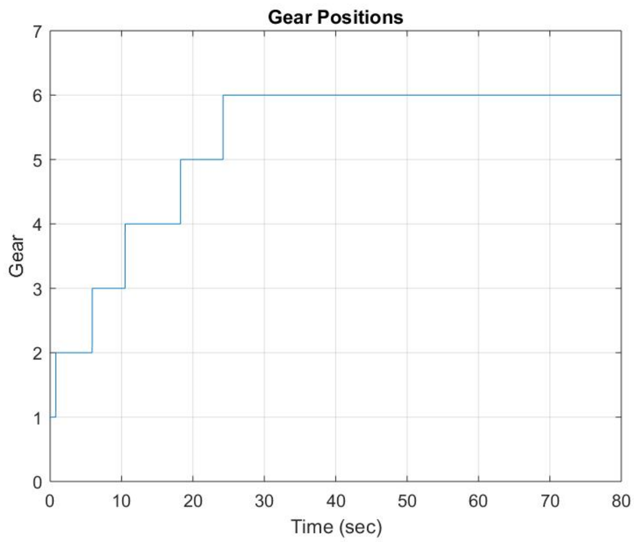


Figure B.13 : Gear positions of CCC fourth scenario.

B.2 : ACC scenarios

This section includes sixth scenario of simulations. Sixth scenario is described that lead vehicle accelerates through 90 kph and initial inter-vehicle distance is 40 meter. Ego vehicle tries to reach lead vehicle and saves the desired time gap as 1 second. Figure B.14 shows the vehicle speed trace of this scenario. Respectively, Figure B.15, B.16, B.17 represents the global best objectives, time gap behaviour, relative distances. Base parameters are 10 for P and D gains both.

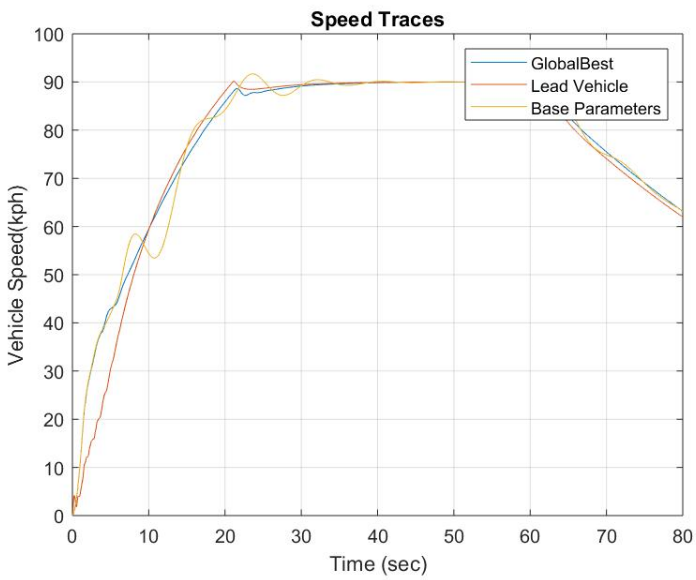


Figure B.14 : Vehicle speed trace of ACC sixth scenario.

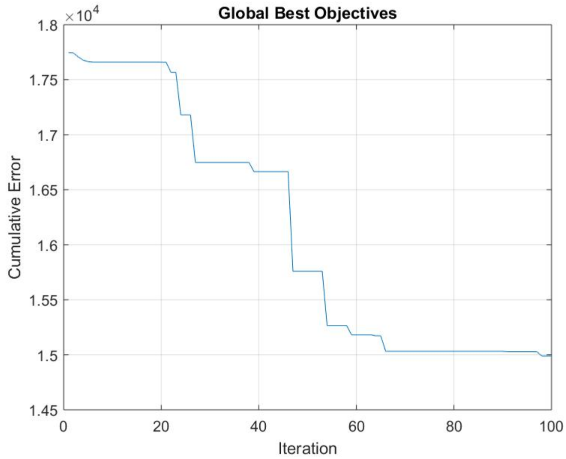


Figure B.15 : Global best objectives of ACC sixth scenario.

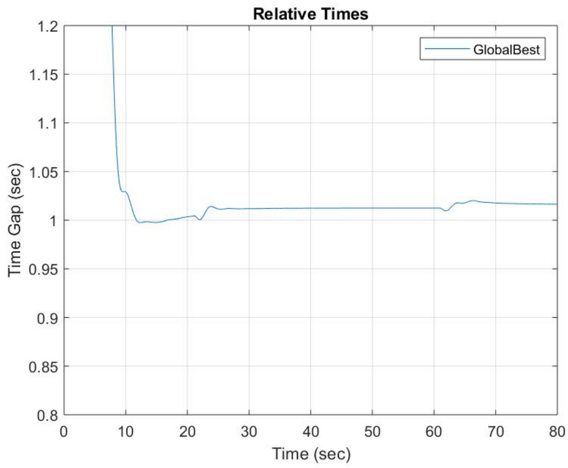


Figure B.16 : Time gap behaviour of ACC sixth scenario.

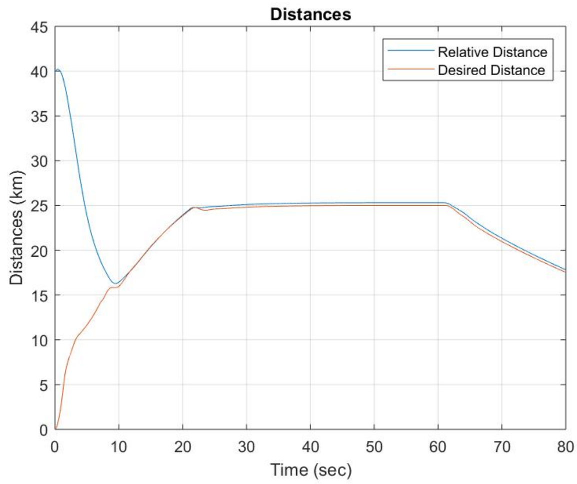


Figure B.17 : Relative distances of ACC sixth scenario.

B.3 : ACC scenarios

This section includes results of nineth and tenth scenarios. Figure B.18 shows the vehicle speed trace of nineth scenario.

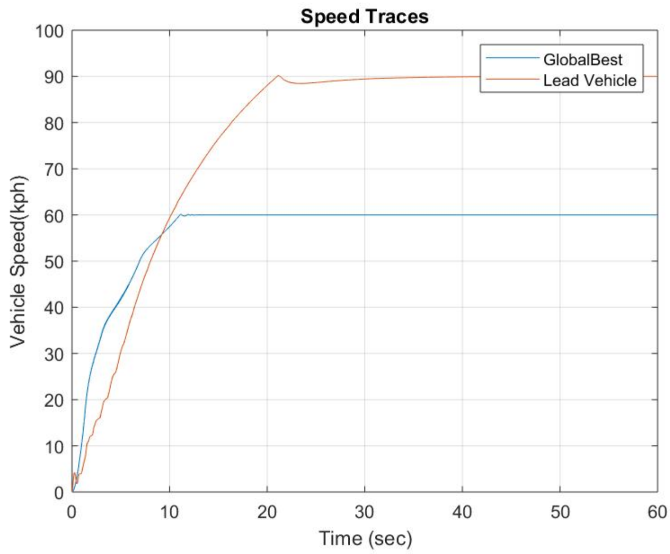


Figure B.18 : Vehicle speed trace of nineth scenario.

Figure B.19 demonstrates switching of ACC/CCC. CCC switch is always active due to high inter-vehicle distance.

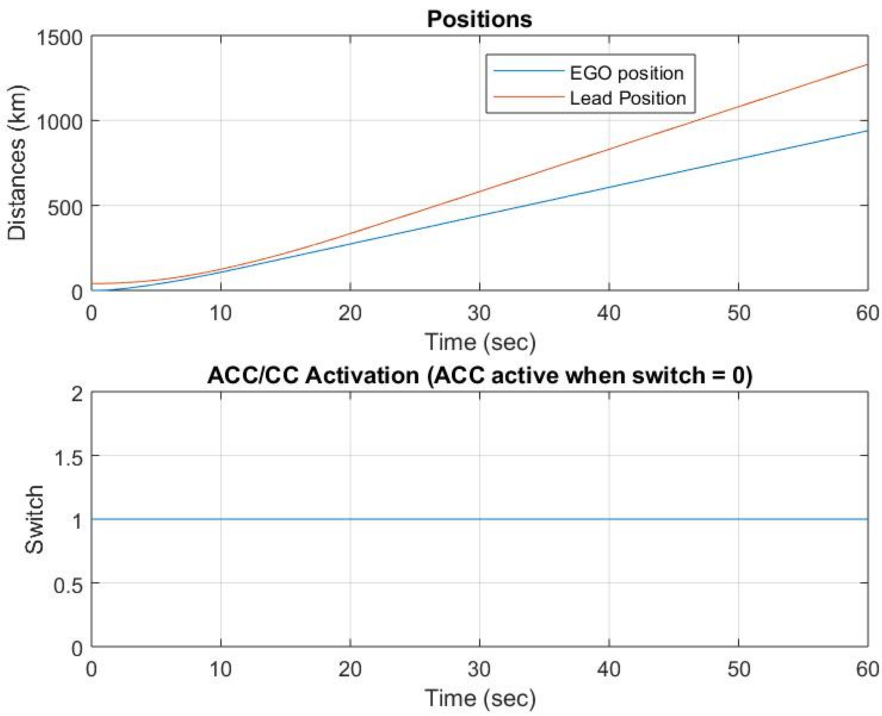


Figure B.19 : ACC/CCC switching of nineth scenario

In the tenth scenario, following vehicle set speed is 120 kph. Even if the inter-vehicle distance is 40 meter, ego vehicle reachs the lead vehicle and gets activated ACC. Figure B.20, B.21 and B.22 represent the vehicle speed traces, switching behaviour and time gap behaviour of tenth scenario.

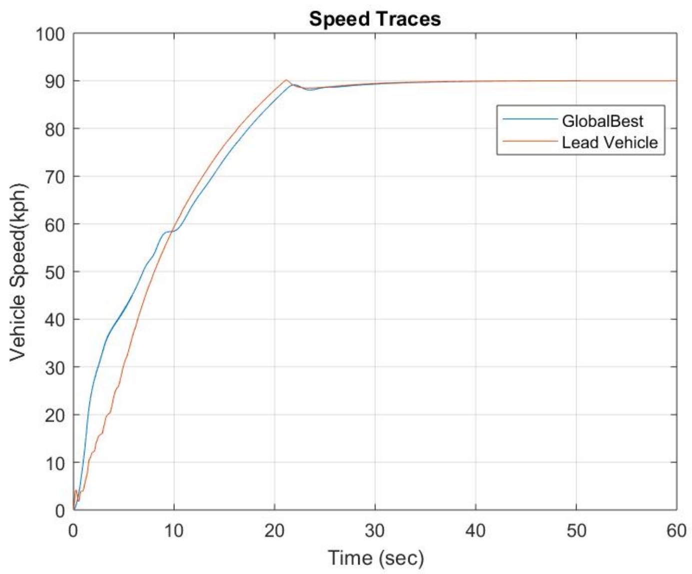


Figure B.20 : Vehicle speed trace of tenth scenario.

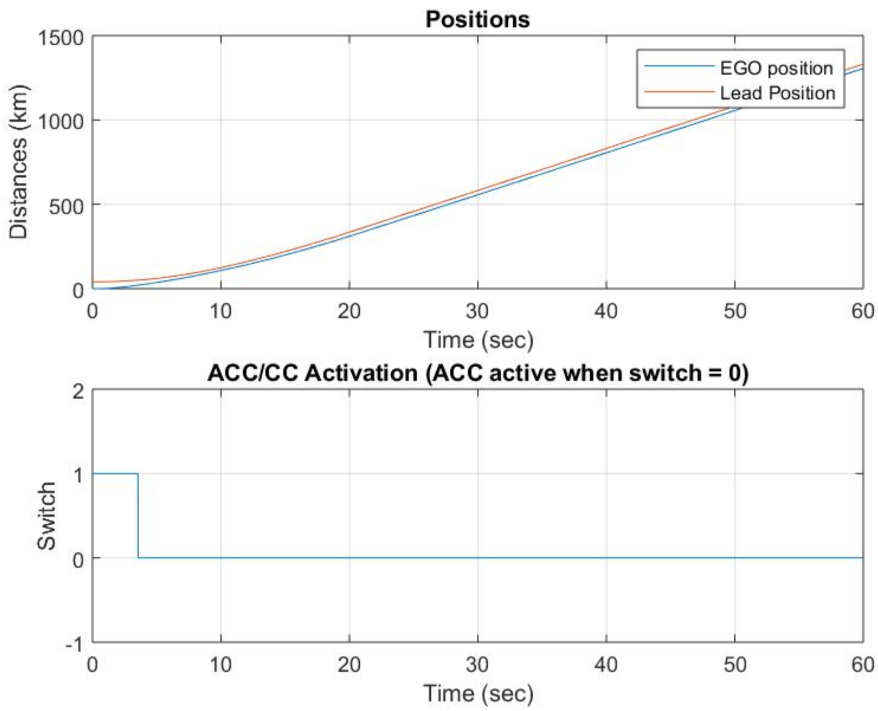


Figure B.21 : ACC/CCC switching of tenth scenario.

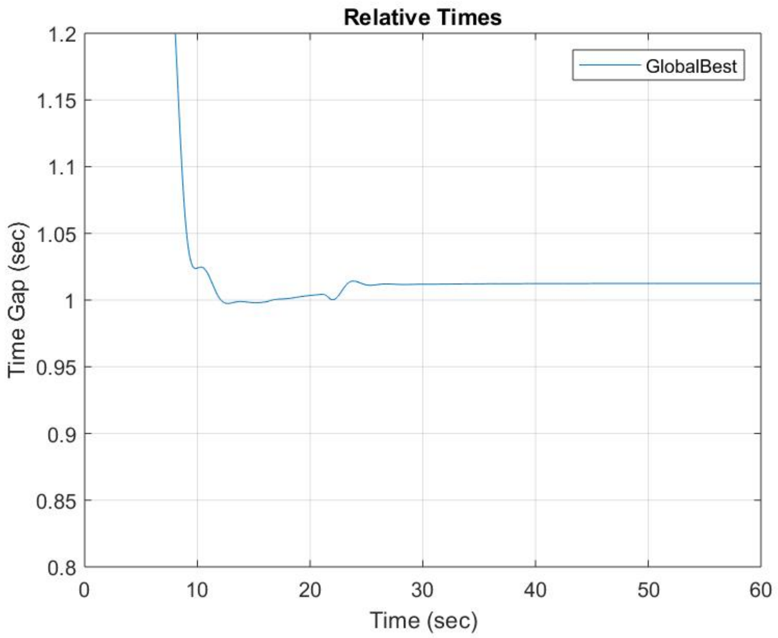


Figure B.22 : Time gap behaviour of tenth scenario.

CURRICULUM VITAE

Name Surname : **Erhan Özkaya**

Place and Date of Birth : **Istanbul/Turkey - 22.07.1995**

E-Mail : **ozkayaer@itu.edu.tr**

EDUCATION:

- **B.Sc.:** 2017, Istanbul Technical University, Mechanical Engineering Faculty, Mechanical Engineering Department

PROFESSIONAL EXPERIENCE

- 2017 - …, Powertrain calibration engineer