Simultaneous Plant/Controller Optimization of Traction Control for Electric Vehicle

by

Kuo-Feng Tong

A thesis
presented to the University of Waterloo
in fulfilment of the
thesis requirement for the degree of
Master of Applied Science
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2007

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Development of electric vehicles is motivated by global concerns over the need for environmental protection. In addition to its zero-emission characteristics, an electric propulsion system enables high performance torque control that may be used to maximize vehicle performance obtained from energy-efficient, low rolling resistance tires typically associated with degraded road-holding ability.

A simultaneous plant/controller optimization is performed on an electric vehicle traction control system with respect to conflicting energy use and performance objectives. Due to system nonlinearities, an iterative simulation-based optimization approach is proposed using a system model and a genetic algorithm (GA) to guide search space exploration.

The system model consists of: a drive cycle with a constant driver torque request and a step change in coefficient of friction, a single-wheel longitudinal vehicle model, a tire model described using the Magic Formula and a constant rolling resistance, and an adhesion gradient fuzzy logic traction controller.

Optimization is defined in terms of the all at once variable selection of: either a performance oriented or low rolling resistance tire, the shape of a fuzzy logic controller membership function, and a set of fuzzy logic controller rule base conclusions. A mixed encoding, multi-chromosomal GA is implemented to represent the variables, respectively, as a binary string, a real-valued number, and a novel rule base encoding based on the definition of a partially ordered set (poset) by delta inclusion.

Simultaneous optimization results indicate that, under straight-line acceleration and unless energy concerns are completely neglected, low rolling resistance tires should be incorporated in a traction control system design since the energy saving benefits outweigh the associated degradation in road-holding ability. The results also indicate that the proposed novel encoding enables the efficient representation of a fix-sized fuzzy logic rule base within a GA.

Acknowledgments

I would like to thank my two supervisors Professor David W.L. Wang and Professor Robert B. Gorbet for their vision, guidance, and consultation.

I would like to thank my readers Professor Mehrdad Kazerani and Professor John McPhee for providing valuable feedback on this work.

I would like to thank the Natural Sciences and Engineering Research Council of Canada (NSERC), and Earl Hughson, President of Solectron Invotronics, for supporting my studies through the Industrial Postgraduate Scholarship program.

I would like to thank Thomas Frommer, Yarko Matkiwsky, and the rest of the Research and Development team at Solectron Invotronics for a delightful and educational industrial work experience as part of their team.

I would like to thank the many sponsors and organizers of the Challenge X vehicle design competition and, in particular, the University of Waterloo Alternative Fuels Team (UWAFT) for the hands-on experience in developing an advanced fuel cell electric vehicle.

This work was made possible by the facilities of the Shared Hierarchical Academic Research Computing Network (SHARCNET:www.sharcnet.ca).

I would like to thank my parents and brothers for their continued love and support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Global concerns over climate change have created the need for environmental protection and action in sustainable development. Temperature rises within the past 35-50 years are attributed to anthropogenic (human) activity causing an increase in green house gas (GHG) concentration within the atmosphere with CO_2 emissions contributing to the majority (60%) of observed changes [1]. Measurements of worldwide GHG emissions indicate that the transportation sector is responsible for 21% of CO_2 emissions [2]. Within the transportation sector, CO_2 emissions can be accounted for by a strong dependence (94.4%) on non-renewable petroleum products [3] with primary energy use (73%) in land vehicles [4]. Combined with a projected 47% increase in world oil consumption from 2003 to 2030, with transportation accounting for approximately half the total [5], it is clear that development of low emission, energy efficient automotive technology is key to mitigating adverse effects of global warming [1]. From this standpoint, the conventional petroleum-based internal combustion engine (ICE) is non-ideal due to the release of GHG trapped within fossil fuels, and a narrow range of optimal operating efficiency [6]. Although innovations in adapting ICE technology to alternative and renewable fuels [7] have been partially successful in curbing emissions and attaining higher levels of fuel economy, further reductions and efficiency improvements are possible in an electric vehicle (EV) where an electric propulsion system is used either in conjunction with, or in place of, an ICE [8].

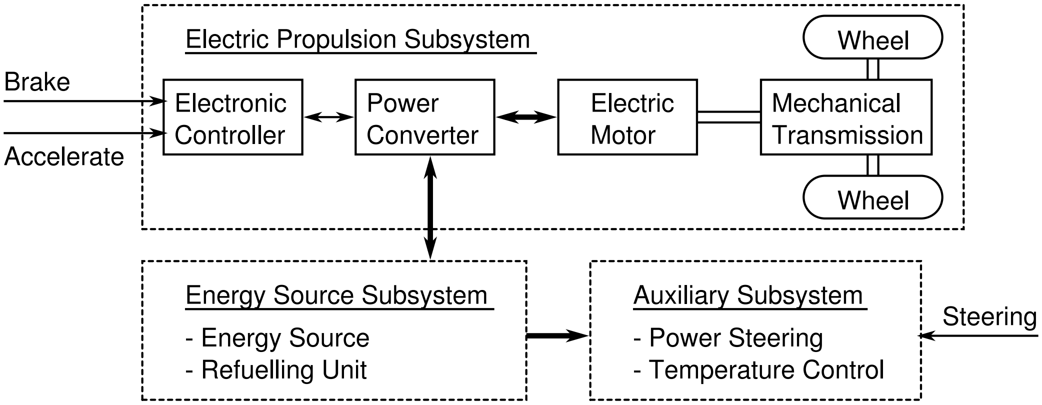


Figure 1.1: Major EV Subsystems [9]

As shown in Figure 1.1, the electric propulsion system is the core component of an electric vehicle (EV). Regardless of the electrical energy source, which may be provided by batteries, a fuel cell, or an ICE-powered generator, an EV relies on electronic control of a power converter and electric motor to efficiently deliver power to the wheels. Characteristics of an electric propulsion system include a high operating efficiency over a wide range of speed and torque demands, energy recovery during regenerative braking (assuming a power receptive energy source), and quick, accurate, torque response [9] [10].

Though high efficiency operation and regenerative braking directly benefit fuel economy, recent research has suggested that further energy savings may be obtained by the use of high performance traction control and energy efficient, low rolling resistance tires [11] [12]. Such a control system relies on the fast torque response of an electric propulsion system to compensate for degraded road adhesion associated with low rolling resistance tires [12]. Existing research has produced controllers that enable tire operation near the limits of performance [11] [12] [13]. However, the trade-offs between vehicle performance and energy economy resulting from the use of low rolling resistance tires with appropriate control has yet to be addressed. Since tire selection and controller tuning (plant and controller variables, respectively) both influence system level objectives in performance and energy consumption, the problem of balancing these high-level vehicle design requirements can be classified as a multi-objective, simultaneous optimization problem of plant and controller variables.

In automotive powertrain designs, a sequential plant then controller design process is typically taken to manage the complexity associated with each process. However, it has been recognized that the sequential powertrain design approach may not allow full optimization of the overall plant/controller system [14]. Simultaneous consideration of a combined plant/controller design problem enables further optimization with respect to system level design objectives. For example, the optimization of powertrain components selection with respect to cost [15] may be considered a separate problem from tuning a powertrain control strategy for optimal efficiency [16]. The final design may be non-optimal since the electric motor may be over-sized with respect to utilization by the control strategy. Thus independent optimization of component selection and control strategy may result in higher cost associated with an oversized motor as well as an unnecessary increase in vehicle mass.

Although simultaneous plant/controller optimization offers potential to improve overall system design, its application to large scale problems is limited by the increased complexity of the associated optimization problem and available solution techniques. In general, a simultaneous optimization must consider a "coupling loop" between plant and controller designs that is illustrated in Figure 1.2. As shown, a change in plant design affects the operation and design of a control strategy. This change, in turn, affects the physical requirements and design imposed on the plant by the controller. In an optimal design, it is desirable that the coupling loop be solved such that the capabilities of the plant are exactly fully utilized by the control strategy. For example, plant controller coupling has been observed in the simultaneous design of a motor and controller [17] where the power specification of the motor design affects plant properties such as armature resistance and inductance. Armature resistance and inductance, in turn, are parameters of the controller design and influence the maximum power used by the control strategy. For a feasible system, the minimum power specification should be greater than or equal to the maximum power required by the control strategy. Intuitively, an opti-

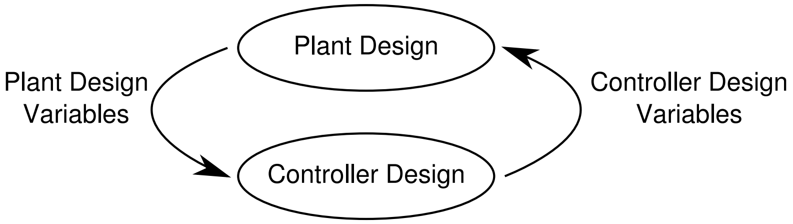


Figure 1.2: Plant/Controller Design Coupling

Table 1.1: Examples of Potential Vehicle Optimization Variables

Plant	Discrete	Selection among pre-manufactured parts Gear ratios (available in rational numbers) Battery voltage (integer number of cell voltage)
	Continuous	Geometry (i.e. suspension dimensions) Suspension spring-damper stiffness
Control	Discrete	Control strategy selection (i.e. PID vs Fuzzy Logic)
	Continuous	Control gain settings

mal system is expected when the maximum power required by the control strategy is equal to the maximum power available from the motor.

Additional complexity in a simultaneous optimization problem may also arise due to problem formulation as a mixed-integer problem involving both continuous and discrete optimization variables. Discrete variables create discontinuities in the problem search space and require search techniques that are independent of gradient information. Examples of continuous and discrete variables that may be considered in a vehicle design are listed in Table 1.1.

Due to the complexity of solving simultaneous optimization problems, application in automotive research is currently limited to individual subsystems such as an active suspension design [18], and an automatic transmission [19]. Further, these works, along with [17], omit application to discrete variables to focus on the optimization of continuous variables. Although this work is limited to a traction control subsystem, a mixed-integer simultaneous optimization problem is considered.

1.1 Research Objectives and Contributions

The goal of this research is to propose a strategy to allow the simultaneous optimization of continuous and discrete plant/controller design variables and to apply it to the design of a simplified electric vehicle traction control system. Research objectives include:

- Development of a vehicle model and traction controller to identify energy use to performance trade-offs associated with plant/controller design variable selection.
- Formulation of an optimization problem and design of a solution technique to provide quantitative assessment of the trade-offs.
- Application of the proposed strategy to a case study of a vehicle drive cycle and analysis of results.

The following contributions are made:

- Formulation of a traction control system as a simultaneous plant/controller optimization problem (Section 4.2).
- Application of a mixed encoding, multi-chromosomal genetic algorithm to the optimization of the resulting problem (Section 5.2).
- Definition of a partially ordered set by delta inclusion (Section 5.3.2) and its application to the encoding of a fuzzy logic controller rule base within a genetic algorithm (Section 5.3.3).

1.2 Overview

This work describes both the modeling of an electric vehicle traction control system and its optimization with respect to system-level objectives in performance and energy economy. Results are based on a simulation study using a mixed encoding genetic algorithm to perform a simultaneous optimization of plant and controller variables.

Chapter 2 details the development of a vehicle (plant) model based on the selection of a case study drive cycle. Vehicle performance and energy use are described using a single-wheel vehicle dynamics model with consideration of longitudinal tire force generation and rolling resistance. Tire performance degradation associated with selecting a low rolling resistance tire in vehicle design is identified.

Chapter 3 defines a fuzzy logic traction controller. First, the choice of controller is supported by an overview of traction control systems. Background information on fuzzy logic control is then presented and followed by application-specific details to complete the definition of the controller. Controller design variables are identified for optimization.

Chapter 4 formulates an simultaneous plant/controller optimization problem to select tire and controller parameters based on system performance and energy use objectives. A solution strategy based on a genetic algorithm and system simulation is proposed in consideration of the coupling between plant and controller, the mix of continuous and discrete optimization variables, and the presence of nonlinear system dynamics.

Chapter 5 describes the application of a mixed encoding, multi-chromosomal genetic algorithm to the optimization of the traction control system. Attention is given to a novel chromosomal encoding of a fuzzy logic controller rule base. Benefits of the encoding are described in terms of reductions in optimization search space and variable constraints.

Chapter 6 presents numerical results obtained from both unoptimized and optimizing simulation studies. Preliminary simulations of the unoptimized system allow verification of the implementation of system dynamics, establish a benchmark in system performance, and demonstrate the potential magnitude of performance to energy economy trade-offs for changes in tire selection and controller parameters. Simultaneous plant/controller optimization results are analyzed in terms the optimal design variable selection as well as on the evolutionary performance of the genetic algorithm.

Chapter 7 provides conclusions and direction for future work based on the development and results of the simultaneous plant/controller optimization.

Chapter 2

Plant Modeling

This chapter presents a simplified vehicle model and case study drive cycle. After a description of the drive cycle, modeling requirements are derived and components are presented to describe: vehicle dynamics, tire force generation mechanism, and energy loss due to rolling resistance. The final plant model is summarized in Section 2.4 and will be used in the formulation of a simultaneous plant/controller optimization problem in Chapter 4. Numeric values of parameters used in the model are summarized in Appendix A.

2.1 Case Study Drive Cycle

A typical drive cycle used to evaluate the performance of a traction control system is a step change in road characteristic under a constant driver torque request T_{ref} [11] [10] [13] [12]. The situation is illustrated in Figure 2.1 where a vehicle starting at initial position S_O ($s_x = 0$) on a level road surface experiences straight-line acceleration through a distance S_F with a road surface transition occurring after distance S_T.

The road characteristic of prime concern in a traction control system is the maximum coefficient of friction existing between tire and road. For a given tire, the maximum coefficient of friction available μ_r at position s_x in the drive cycle of

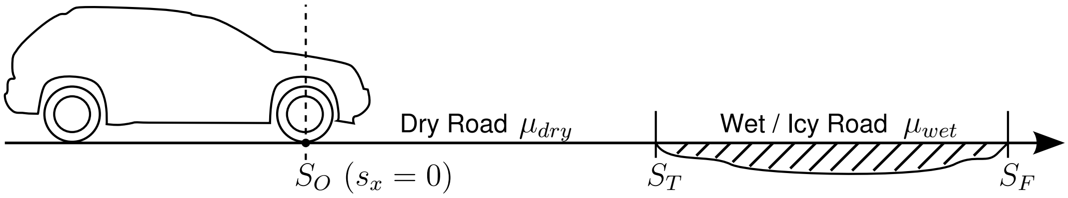


Figure 2.1: Case Study Drive Cycle

Figure 2.1 is:

$$\mu_r = \begin{cases} \mu_{dry} & s_x < S_T \\ \mu_{wet} & s_x \geq S_T \end{cases} \tag{2.1}$$

where μ_{dry} is the coefficient of friction on the dry road and μ_{wet} is the coefficient of friction on the wet or icy road such that $\mu_{dry} > \mu_{wet}$.

This drive cycle is used in this thesis to evaluate the basic response of a traction control system to changes in road surface. The drive cycle remains fixed throughout simulation studies to provide a basis for comparing different vehicle and controller configurations.

2.2 Vehicle Dynamics

Based on the case study drive cycle, assumptions are made to simplify vehicle dynamics to enable the use of a single-wheel longitudinal vehicle model. The model is further augmented to allow calculation of vehicle performance and energy use during the drive cycle.

2.2.1 Assumptions

In general, overall vehicle (chassis) motion may be described in terms of six degrees of freedom (DoF) that are defined relative to a coordinate system attached to the vehicle's center of gravity (CoG) [20]. As shown in Figure 2.2, these six DoF are:

- 3 DoF in rotation: roll, pitch, and yaw, and

- 3 DoF in translation: longitudinal, lateral, and vertical.

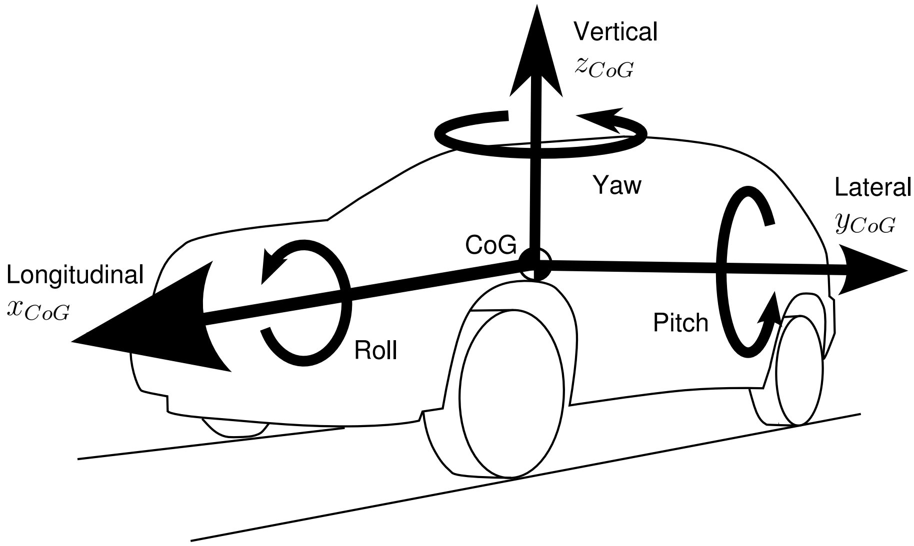


Figure 2.2: Vehicle Six Primary Degrees of Freedom

These six DoF are tightly coupled through the interaction of vehicle subsystems that introduce additional DoF. For example, to describe vehicle motion in a double lane change maneuver, a 94 DoF model is required to consider nonlinear dynamics produced by front and rear suspensions, steering system, and tire-road interaction [21].

For the straight-line acceleration drive cycle considered in this thesis, the following assumptions decouple and simplify the analysis of longitudinal vehicle dynamics to the CoG $x - z$ plane:

- Lateral forces, body yaw, and body roll are negligible and assumed to be zero.
- Body pitch and position of CoG relative to the wheels is not affected (assuming a stiff suspension with negligible motion).
- Longitudinal force is due solely to tire forces with negligible aerodynamic drag.

In general, a vertical dynamic weight transfer from the front to rear wheels is associated with vehicle acceleration since longitudinal tire forces cause a moment

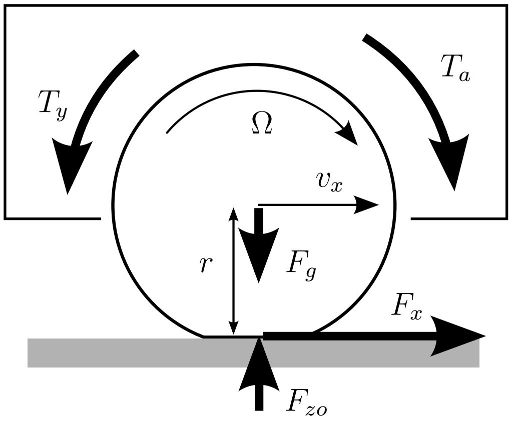


Figure 2.3: Longitudinal Single-Wheel Model [12]

about the vehicle CoG. However, as justified in Appendix B using the longitudinal bicycle model, vertical dynamics are neglected (vertical, or normal force is assumed constant) since it is assumed that the weight transfer is minimized due to:

- vehicle design with low CoG, and long wheel base (as common in a sports car), and
- case study drive cycle with low-to-moderate vehicle accelerations.

2.2.2 Single-Wheel Longitudinal Model

The preceding assumptions allow a vehicle to be represented by a **longitudinal single-wheel model** (Figure 2.3) that is used to focus analysis on the influence of tire-road interactions on longitudinal vehicle dynamics [12]. The vehicle is represented by a rigid body of mass m supported by a single wheel represented by a rigid body disc of radius r and rotational inertia J. The wheel is driven by an externally applied torque T_a. Wheel mass is unmodeled in the longitudinal dynamics since it is assumed negligible compared to vehicle mass. The vehicle experiences a gravitational force $F_g = mg$ that is partially countered by the reaction normal force F_{zo} at the tire-road contact point. The reaction force F_{zo} is constant due to the assumption of minimal vertical weight transfer.

Note. Special consideration is given in modeling the vehicle mass m as well as the normal force F_{zo} to allow the single-wheel model to capture the longitudinal dynamics of a full vehicle model. Assuming that the vehicle is driven by a single axle (either front wheel or rear wheel drive), half the physical vehicle mass is modeled in m since only one of two driven tires is modeled. Since a static weight distribution exists between front and rear wheels depending on the location of vehicle CoG relative to the wheels, the normal force F_{zo} acting on the modeled wheel is typically such that $F_{zo} < F_g$. The normal force F_{zo} is derived from the static weight distribution terms in either (B.7a) or (B.7b) depending on whether the front or rear tire is modeled, respectively. Note that F_g is balanced by both F_{zo} and a tire reaction force from the remaining undriven, unmodeled axle.

In general, the maximum longitudinal force $F_{x\,\max}$ is limited by the maximum tire-road friction μ_r such that $F_x \leq \mu_r F_{zo} = F_{x\,\max}$. However, the actual tire force generated F_x is a nonlinear function of wheel rotation, vehicle speed, and road condition as detailed in Section 2.3.1. In addition to providing longitudinal propulsion force, F_x also exerts a moment on the wheel through wheel radius r. The rolling resistance torque T_y opposes the rotation of the wheel and is further explained in Section 2.3.2.

The dynamics of the longitudinal single-wheel model is described by summing the forces and moments in Figure 2.3 to obtain the longitudinal acceleration $\dot{v}_x$ of the vehicle and the angular acceleration $\dot{\Omega}$ of the single wheel:

$$\dot{v}_x = \frac{F_x}{m} \tag{2.2}$$

$$\dot{\Omega} = \frac{T_a - T_y - rF_x}{J}. \tag{2.3}$$

where $F_x =$ tire longitudinal force (derived in Section 2.3.1);
$m =$ half the physical vehicle mass;
$v_x =$ vehicle speed;
$T_a =$ externally applied torque on wheel;
$T_y =$ tire rolling resistance torque (derived in Section 2.3.2);
$r =$ radius of wheel;
$J =$ rotational inertia of wheel (and effective motor inertia);

$\Omega =$ rotational speed of wheel.

2.2.3 Measuring Energy Use and Performance

The plant model is augmented with two additional state variables O_T and O_E to allow measurement of drive cycle completion time (representing vehicle performance) and energy use, respectively. These variables serve as optimization objectives in the formulation of the simultaneous plant/controller optimization problem in Chapter 4.

Drive cycle completion time is the time taken for the vehicle to traverse distance S_F (see Figure 2.1). Vehicle position s_x within the drive cycle is obtained by integrating vehicle speed:

$$\dot{s}_x = v_x. \tag{2.4}$$

The time O_T for the vehicle to complete the drive cycle is determined by integrating (with respect to time t):

$$O_T = \int \dot{O}_T \, dt \tag{2.5}$$

where

$$\dot{O}_T = \begin{cases} 1 & s_x \leq S_F \\ 0 & s_x > S_F. \end{cases} \tag{2.6}$$

Hence, (2.5) has the effect of setting $O_T = t$ while $s_x \leq S_F$.

In an EV, T_a is supplied by an electric propulsion system incorporating an electric motor. Electric motors exhibit rapid torque response that may be accurately controlled due to the proportional relationship between torque and motor current [13]. In this thesis, it is assumed that the torque response of the electric propulsion system is faster than other system dynamics and therefore may be instantaneously generated according to controller demands. It is also assumed that the electric propulsion system is 100% efficient in converting electrical energy into mechanical energy such that:

$$P_{in} = P_{elec} = P_{mech} = T_a \, \Omega \tag{2.7}$$

where P_{in}, P_{elec}, and P_{mech} represent instantaneous input, electrical, and mechanical power, respectively. Therefore, overall energy use O_E may be obtained by integrating P_{in} while $s_x \leq S_F$:

$$O_E = \int \dot{O}_E \, dt \tag{2.8}$$

where

$$\dot{O}_E = \begin{cases} T_a \, \Omega & s_x \leq S_F \\ 0 & s_x > S_F. \end{cases} \tag{2.9}$$

2.3 Tire Model

The pneumatic tire, an essential vehicle component, affects handling, traction, ride comfort, appearance, and fuel economy [22]. The pneumatic tire plays a major role in vehicle dynamics since vehicle motion is derived from forces generated by tire deformations as a result of interacting wheel motion and road conditions [23]. This section presents models to describe the longitudinal force generation mechanism of a tire as well as tire rolling resistance since the phenomena influence, respectively, vehicle performance and energy loss under the straight-line acceleration considered in this thesis. Extension of the tire model to situations involving maneuvers such as cornering requires the consideration of the lateral tire force that is beyond the current scope of study. The forces and moments associated with such a generalized tire model are discussed in Appendix C for future modeling extensions.

2.3.1 Longitudinal Tire Force Generation

A variety of approaches are available to model the force generation mechanism of a tire as shown in Table 2.1. The approaches range from purely empirical to purely theoretical from left to right across the table. Approaches in the middle columns trade accuracy and detail for reduced computational complexity. Proceeding from the left across the table, tire models may be derived empirically by curve fitting to a full set of tire tests including combined in-plane and out-of-plane operating

Table 2.1: Approaches to Tire Modeling [23][24]

Empirical	**Semi-Empirical**	**Simple Physical**	**Complex Physical**
Curve fit to extensive experimental data	Extrapolate from basic nominal experimental data	Simple representation & physical insight	Finite-element analysis to simulate details

conditions. The Magic Formula tire model is an example of a widely accepted empirical tire model [23].

A semi-empirical approach applies simple physical understanding to distort and rescale experimental data. This approach extrapolates operating characteristics from a reduced experimental data set covering nominal operating conditions.

In specific applications, a simple physical model may provide sufficient accuracy and insight into tire behavior. For example, the brush model describes force and moment generation in terms of a row of elastic bristles in contact with the road surface [23].

Theoretical finite-element models allow modeling of complex physical effects such as pressure distribution within the contact patch [23] and large deformation in tire abuse simulations [24]. Beyond vehicle simulation, finite-element models are used to establish relationships between tire performance and construction [23] and in the high-frequency modeling of tire noise emission [24].

This thesis adopts a semi-empirical approach to model longitudinal tire force generation. Specifically, the **similarity method** [23] is used to distort a Magic Formula curve fit for tire data obtained under nominal loading F_{zo} and on a road with (maximum) reference coefficient of friction μ_o. This approach yields a relatively simple, yet accurate, model that is suitable for the simulation-based optimization procedure developed in Chapter 4.

Model development is presented progressively and begins with a discussion of tire slip which serves as a basis for defining the Magic Formula. Next, the similarity method is defined along with its application to the Magic Formula tire model. The

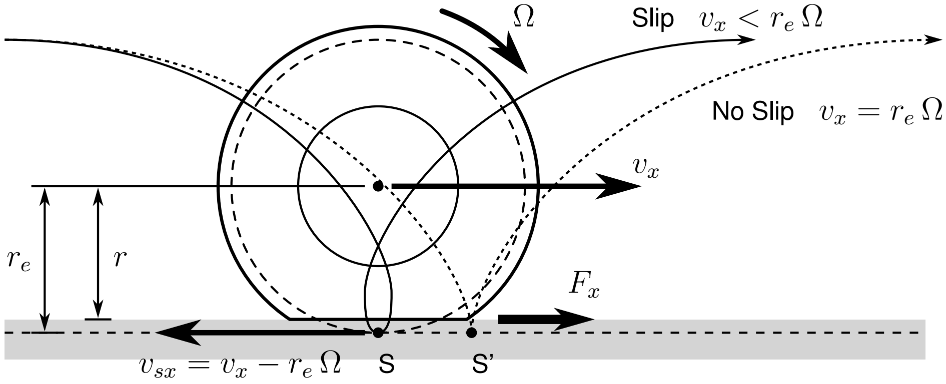


Figure 2.4: Velocity of Slip Points S (Driven) and S' (Free Rolling)

adhesion gradient is defined in terms of the derivative of the model to reveal tire characteristics suitable for use in developing a traction control strategy in Chapter 3.

Tire Slip

Under conditions of straight-line acceleration, tire force generation can be described using the planar tire model shown in Figure 2.4. The linear and rotational speeds of the wheel, v_x and Ω, are as defined by the single-wheel model in (2.2) and (2.3), respectively.

The **effective rolling radius** r_e of the wheel is empirically defined under free rolling conditions (with longitudinal force $F_x = 0$), such that the forward speed of the wheel v_x is related to the angular speed of rotation Ω by [23]:

$$v_x = r_e\,\Omega. \tag{2.10}$$

As shown in Figure 2.4, the effective rolling radius r_e is typically greater than the **loaded radius** r that describes the physical tire deformation [23]. Therefore physical tire deformation may be neglected by assuming an equivalent circular tire rolling on an imaginary surface that is parallel to the ground plane at radial distance r_e from the wheel center.

The relative speed of wheel rotation Ω to forward speed v_x is of particular interest in modeling tire force generation and is considered by defining a **slip point**

S at radius r_e on the tire that is instantaneously in contact with the imaginary surface.

Under free rolling conditions, (2.10) implies a wheel that is rolling, without slipping, over the imaginary surface. As shown by the dotted trajectory in Figure 2.4, the instantaneous velocity of a slip point S' on a free rolling wheel is zero due to the point's instantaneous reversal in vertical direction of travel.

In contrast, under driven conditions (with $F_x \neq 0$), the velocity of slip point S is non-zero due to slip between the tire and the imaginary surface. The tangent to the solid trajectory shown in Figure 2.4 shows that split point S on a driven tire exhibits a non-zero longitudinal speed v_{sx} relative to the road given by:

$$v_{sx} = v_x - r_e \, \Omega. \tag{2.11}$$

For convenience, v_{sx} is normalized with $r_e \, \Omega$ [23] and the resulting ratio is termed **longitudinal slip** κ:

$$\kappa = -\frac{v_{sx}}{r \, \Omega} = -\frac{v_x - r \, \Omega}{r \, \Omega}, \quad v_x \leq r \, \Omega \tag{2.12}$$

where tire deformation is neglected and it is assumed that $r = r_e$. In general, both the effective radius and loaded radius are nonlinear functions of tire wear, vertical loading, and speed [23]. The sign of κ is chosen such that $\kappa \geq 0$ for acceleration. Also, a maximum of $\kappa = 1$ occurs for spinning the wheels on a stationary vehicle ($\Omega > 0$, $v_x = 0$).

Note. The longitudinal slip as defined in (2.12) is not suitable for describing a situation involving starting from a standstill (Ω=0) since the denominator would be zero. To handle such cases, the longitudinal slip may be defined by a differential equation without singularities as done in [23]. To avoid the singularity in this thesis, initial conditions are defined in Appendix A such that the wheel is free rolling ($\kappa = 0$) with $v_{xo} > 0$ and $\Omega_o = v_{xo}/r$.

The Magic Formula

Full scale tire tests have shown that longitudinal force F_x is nonlinearly dependent on longitudinal slip κ [23]. Typically, as illustrated in Figure 2.5, F_x increases

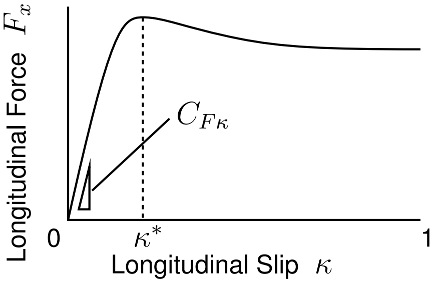


Figure 2.5: Longitudinal Tire Force Characteristic [12]

with initial slope $C_{F\kappa}$ up to a maximum at κ^* before decreasing to a fraction of its maximum at 100% slip.

The **Magic Formula** is widely accepted in fitting the longitudinal force generation characteristics of a tire [23]. It is based on a sine wave distortion as follows:

$$F_{xo} = D_{xo}\,\sin[C_x\,\tan^{-1}\{B_{xo}\,\kappa - E_x(B_{xo}\,\kappa - \tan^{-1}(B_{xo}\,\kappa))\}] \tag{2.13}$$

where $F_{xo} =$ nominal output longitudinal tire force
$\kappa =$ input longitudinal tire slip
$B_{xo} =$ nominal stiffness factor
$C_x =$ shape factor
$D_{xo} =$ nominal peak value
$E_x =$ curvature factor.

Note. Subscript o indicates values that describe nominal characteristics and serve to distinguish between rescaled values. This distinction will be clarified shortly during discussion of the similarity method.

Coefficients of the Magic Formula are named since D_{xo} determines the maximum amplitude of the sine function, C_x controls the overall shape by limiting the range of the sine function, and E_x determines the curvature and horizontal position of the peak. Since D_{xo} describes the maximum longitudinal force $F_{x\,\max}$, it may be defined as:

$$F_{x\,\max} = D_{xo} = \mu_o F_{zo} \tag{2.14}$$

where the tire loading force is equal to the constant normal force F_{zo} from the single-wheel model (Section 2.2.2) and μ_o is the maximum coefficient of the road surface associated with the fitted data.

The influence of B_{xo} is revealed by taking the derivative of the Magic Formula (2.13) with respect to κ (recalling $\frac{d}{dx}\tan^{-1}x = \frac{1}{1+x^2}$):

$$\begin{aligned}\frac{dF_{xo}}{d\kappa} &= D_{xo}\cos[C_x \tan^{-1}\{B_{xo}\,\kappa - E(B_{xo}\,\kappa - \tan^{-1}(B_{xo}\,\kappa))\}] \\ &\qquad C_x \times \left[\frac{B_{xo} - E\left(B_{xo} - \dfrac{B_{xo}}{1+B_{xo}^2\kappa^2}\right)}{1+\{B_{xo}\,\kappa - E(B_{xo}\,\kappa - \tan^{-1}(B_{xo}\,\kappa))\}^2}\right].\end{aligned} \tag{2.15}$$

By setting $\kappa = 0$ in (2.15), the initial slope of the Magic Formula is $C_{F\kappa} = B_{xo}C_xD_{xo}$. Therefore the remaining coefficient B_{xo} is free to control the initial "stiffness" of the Magic Formula to fit the initial slope $C_{F\kappa}$ indicated in Figure 2.5.

The Similarity Method

Although the Magic Formula allows an accurate description of longitudinal force in terms of tire slip, it is necessary to extend the tire model to account for the transition in road surface coefficient of friction as described in the drive cycle of Section 2.1. When the coefficient of friction of a road deviates from a nominal (reference) value μ_o, the maximum longitudinal tire force changes while the initial slope $C_{F\kappa}$ remains constant [23]. This observation enables the similarity method to extrapolate (by scaling) tire force characteristics from an experimental data set obtained at reference condition μ_o to the new condition μ_n [23].

Using the similarity method, the tire force generation characteristic F_x associated with the new μ_n is described by scaling the Magic Formula (2.13) by μ_n/μ_o in both horizontal and vertical directions [23]:

$$\begin{aligned}F_x &= \frac{\mu_n}{\mu_o}F_{xo}(\kappa_{eq}) \\ &= \frac{\mu_n}{\mu_o}D_{xo}\sin[C_x \tan^{-1}\{B_{xo}\,\kappa_{eq} - E_x(B_{xo}\,\kappa_{eq} - \tan^{-1}(B_{xo}\,\kappa_{eq}))\}]\end{aligned} \tag{2.16}$$

where the **equivalent longitudinal slip** κ_{eq} is given by:

$$\kappa_{eq} = \frac{\mu_o}{\mu_n}\,\kappa. \tag{2.17}$$

The vertical scaling establishes a new maximum in tire force generation according to the percentage change in road friction from nominal. The horizontal scaling on κ is required to preserve the initial slope $C_{F\kappa}$ as seen by taking the derivative of (2.16) with respect to κ and evaluating at zero [23]:

$$\left.\frac{dF_x}{d\kappa}\right|_0 = \left.\left(\frac{\mu_n}{\mu_o}\frac{dF_{xo}}{d\kappa_{eq}}\frac{d\kappa_{eq}}{d\kappa}\right)\right|_0 = \left.\frac{dF_{xo}}{d\kappa_{eq}}\right|_0 = C_{F\kappa}. \tag{2.18}$$

In this thesis, it is assumed that tire data is obtained on a dry reference road surface. It is also assumed that an 80% reduction in friction and peak force generation occurs when operating a tire on a wet road surface. Therefore, the road friction μ_r defined in (2.1) is substituted for the ratio μ_n/μ_o in (2.16) and (2.17) such that:

$$\mu_r = \frac{\mu_n}{\mu_o} = \begin{cases} \mu_{dry}(=1) & s_x < S_T \\ \mu_{wet}(=0.2) & s_x \geq S_T \end{cases}. \tag{2.19}$$

For convenience in future analysis of tire force generation, longitudinal tire force F_x is normalized by the normal tire force F_{zo} to define the **driving friction coefficient** $\mu(\kappa)$ (with implied dependence on road surface condition):

$$\mu(\kappa) = \frac{F_x}{F_{zo}}. \tag{2.20}$$

Analysis of F_x may be performed in terms of $\mu(\kappa)$ since F_{zo} is constant and $\mu(\kappa)$ varies directly with F_x. The convenience in using $\mu(\kappa)$ is that it emphasizes that actual friction between tire and road is dependent on wheel slip and is limited to a maximum μ_n as determined by substituting $F_{zo} = D_{xo}/\mu_o$, obtained by rearranging (2.14), into (2.20).

Adhesion Gradient

Additional longitudinal tire force characteristics may be deduced from analysis of the **adhesion gradient** A defined as the derivative of F_x (2.16) with respect to κ

that is normalized by stiffness $C_{F\kappa}$:

$$A = \frac{\dot{F}_x}{C_{F\kappa}} = \frac{\dot{F}_x}{B_{xo} C_x D_{xo}} \leq 1 \tag{2.21}$$

where $\dot{F}_x$ is termed the **force derivative**:

$$\dot{F}_x = \frac{\partial F_x}{\partial \kappa} = D_{xo} \cos[C_x \tan^{-1}\{B_{xo}\,\kappa_{eq} - E(B_{xo}\,\kappa_{eq} - \tan^{-1}(B_{xo}\,\kappa_{eq}))\}]$$
$$C_x \times \left[\frac{B_{xo} - E\left(B_{xo} - \dfrac{B_{xo}}{1 + B_{xo}^2 \kappa_{eq}^2}\right)}{1 + \{B_{xo}\,\kappa_{eq} - E(B_{xo}\,\kappa_{eq} - \tan^{-1}(B_{xo}\,\kappa_{eq}))\}^2} \right]. \tag{2.22}$$

The optimal slip κ^* corresponding to peak F_x may be obtained by setting (2.21) to zero. Further simplification to (2.21) is possible given that $E_x = 0$ in the tire data obtained from [23] (as listed in Appendix A):

$$A = \cos[C_x \tan^{-1}\{B_{xo}\,\kappa_{eq}\}] \times \left[\frac{1}{1 + \{B_{xo}\,\kappa_{eq}\}^2} \right]. \tag{2.23}$$

The solution of κ^* by setting $A = 0$ proceeds by first noting that as κ increases from zero, the argument of the cosine in (2.23) increases and the value of the cosine decreases from unity until it changes sign when the argument is $\pi/2$. Since the cosine is decreasing at this point, solution of κ in $C_x \tan^{-1}\{B_{xo}\,\kappa_{eq}\} = \pi/2$ corresponds to a maximum in F_x at optimal slip:

$$\kappa^* = \frac{\tan \dfrac{\pi}{2\,C_x}}{B_{xo}}\,\mu_r. \tag{2.24}$$

As κ continues to increase past κ^*, the cosine and A remain negative while the argument of the cosine remains less than $3\pi/2$:

$$C_x \tan^{-1}\{B_{xo}\kappa_{eq}\} < \frac{3\pi}{2} \quad \Rightarrow \quad C_x < 3. \tag{2.25}$$

Since the shape factor $C_x = 1.6$ given by the tire data from [23] (as listed in Appendix A) is less than 3, the adhesion gradient remains negative ($A < 0$) for $\kappa > \kappa^*$. Therefore, adhesion gradient A is positive for $\kappa < \kappa^*$, zero for $\kappa = \kappa^*$, and negative for $\kappa > \kappa^*$. This characteristic is further examined in the traction controller development of Chapter 3.3.

2.3.2 Rolling Resistance

A review of the impact of rolling resistance on fuel economy is followed by a discussion on design factors influencing rolling resistance and selection of a rolling resistance model.

Rolling Resistance and Fuel Economy

Rolling resistance is defined as the amount of energy expended by a tire per unit distance traveled [25]. The definition allows rolling resistance to be interpreted as a force F_r that opposes the direction of travel of a tire (with units [N]=[J]/[m]). Alternatively, rolling resistance may be interpreted as a torque T_y that opposes the rotation of a tire (with units [Nm]=[J]/[rad]).

As a tire rolls, portions of the tire entering and leaving the contact patch are repeatedly subject to cycles of deformation and recovery. Some of the energy used to deform the tire is not returned during the recovery period. Rather, this energy is lost as heat and decreases the available energy to turn the tire. Hence, a rolling tire incurs hysteresis loss characterized by its rolling resistance [22].

The biggest influence on a vehicle's fuel economy remains its powertrain design. The ICE powertrain of a midsize vehicle may account for 80% (in highway driving) to 87% (in urban driving, due to idling engine) of the total power losses (Figure 2.6) [22]. However, once the powertrain has converted the vehicle's primary energy supply into mechanical energy and delivered it to the wheels, only three other types of losses remain: braking, aerodynamic drag, and rolling resistance. The pie charts in Figure 2.6 show the relative proportion of energy loss once the remaining 13% - 20% of energy makes its way to the vehicle's wheels. The proportion of energy loss is different between urban and highway driving. In low-speed, stop-and-go urban driving, braking consumes the most energy while aerodynamic effects are minimal. In steady, high-speed highway driving, aerodynamic drag dominates due to its increase with vehicle speed. In both urban and highway driving, Figure 2.6 shows that rolling resistance accounts for at least 30% of the energy losses among aerodynamic drag, braking, and rolling. The minimization of rolling resistance is thus important to the realization of a fuel-efficient vehicle design.

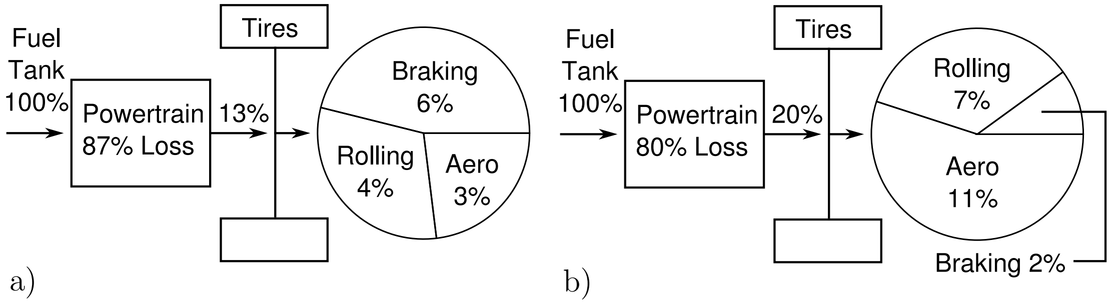


Figure 2.6: Energy Flow for a Typical ICE Midsize Passenger Car a) Urban Driving b) Highway Driving [22]

Factors Affecting Rolling Resistance

Rolling resistance is influenced by both tire design and operating conditions. Tire tread design is of particular interest since it contributes to the majority of energy loss (65%-75%) associated with rolling resistance [26]. Advances in tread block design and tread rubber compounds continue to be key in enabling cost-effective, low rolling resistance tires with enhanced tread-wear resistance and wet traction while maintaining standards in dry traction and noise, vibration and harshness (NVH) [26]. A reduction in tire width is also an option for reducing rolling resistance although it is typically associated with degraded road adhesion and handling characteristics [26].

In operation, tire rolling resistance is influenced by vertical loading F_z, inflation pressure P, speed of travel V, and temperature T. Under steady state conditions (straight rolling with zero slip), a standard [27] semi-empirical model for rolling resistance is:

$$F_r = P^{\alpha} F_z^{\beta}(a + bV + cV^2) \tag{2.26}$$

where α, β, a, b, c are exponents and coefficients chosen by regression techniques to fit experimental data. Generally, $\alpha \approx -0.4 < 0$ and $\beta \approx 1$ [27] [28] since rolling resistance decreases with increasing inflation pressure [29] and is proportional to normal load [23], respectively. Rolling resistance increases approximately linearly ($a \approx 0.05, b \approx 1\text{e} - 4, c \approx 1\text{e} - 6$ [27]) with speed as shown in Figure 2.7 [30]. The figure also shows that the model (2.26) exhibits the appropriate trends with respect

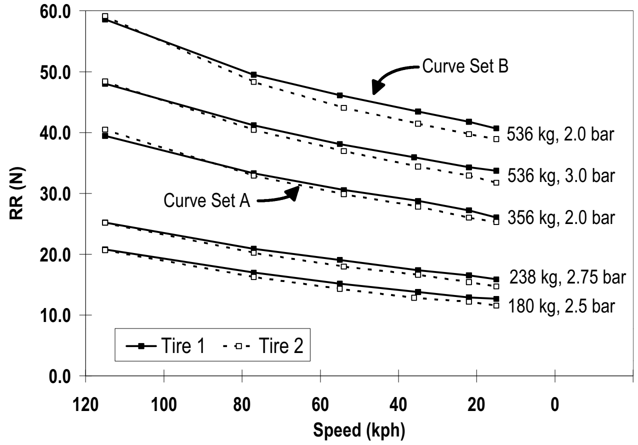


Figure 2.7: Speed, Pressure, and Loading vs Rolling Resistance : Direct from [27]

to rolling resistance versus normal loading and inflation pressure.

Due to its strong linearity with normal loading, rolling resistance is readily described as a **rolling resistance coefficient** C_r defined as the ratio of rolling resistance force to normal loading [31]:

$$C_r = F_r/F_z. \tag{2.27}$$

When measured at the same pressure and speed, C_r is a convenient way to compare rolling resistance of tires [31]. In a recent study, C_r was found to range from 0.007 to 0.014 on new tire designs [22].

Furthermore, rolling resistance decreases with increasing temperature when measured on road vehicles (Figure 2.8 [32]) due to a corresponding pressure increase from heating [29]. Figure 2.8 therefore indicates a conflicting trend in rolling resistance between a change in speed and temperature. For example, assume steady state operation at 55 km/h and 45°C (point A) in Figure 2.8. A quick transient speed increase to 75 km/h (point B) causes an initial increase in rolling resistance while prolonged operation at 75 km/h causes temperature to stabilize to 55°C (point C) with associated inflation pressure increase and rolling resistance decrease. The dependence of rolling resistance on temperature and speed is further considered in [29].

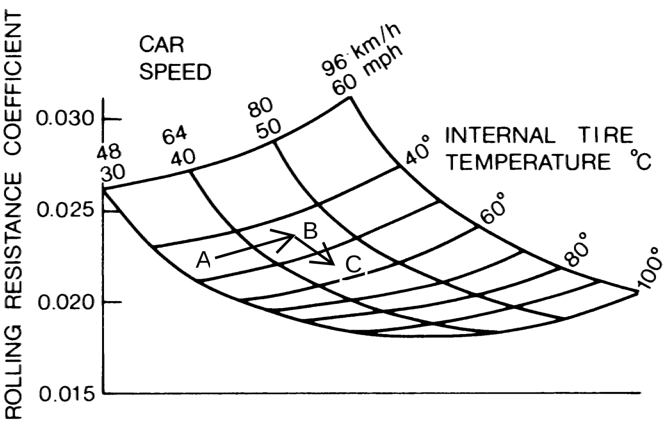


Figure 2.8: Speed and Temperature vs Rolling Resistance : Direct from [32]

In this thesis, the normal load F_z is assumed to be a constant F_{zo} and the effect of temperature and speed changes are assumed to cancel each other. Hence, rolling resistance F_r and its coefficient C_r are constant as related by (2.27) and the rolling resistance moment T_y is modeled as a constant [23]:

$$T_y = rC_rF_{zo} \tag{2.28}$$

where r and F_{zo} are the wheel radius and normal force from the single-wheel model, and C_r is the rolling resistance coefficient.

2.3.3 Performance versus Energy Efficient Tire

Previous discussion on tire force and rolling resistance modeling show that tires influence vehicle performance and fuel economy. In general, tire design considerations beyond the current scope of study result in a trade-off between high performance tires and energy efficient, low rolling resistance tires [12] [26].

In this thesis, it is assumed that vehicle design freedom is restricted to selecting among pre-manufactured tires. Specifically, the impact of tire selection on vehicle performance and energy economy is studied using models of a performance tire (Tire 0), and a low rolling resistance tire (Tire 1). The naming of the tires is chosen to facilitate the definition of the optimization variable TIRE associated with tire selection in the overall traction control system optimization defined in Chapter 4. Since the impact of low rolling resistance tire design on dynamic performance remains an open area of research [33], potential impact is deduced from the observation that rolling resistance decreases as inflation pressure increases (2.26). Research on the impact of tire inflation pressure changes on dynamic performance [34] shows that the impact is highly tire dependent. However, simulation results within the work predict a decrease in both stiffness $C_{F\kappa o}$ and peak force D_{xo} since a higher inflation pressure decreases the tire-road contact patch area responsible for generating tire forces. Parameters describing Tire 0 and Tire 1 in Table A.3 have been selected in accordance with predicted trends.

The longitudinal tire force characteristics and its derivative for both Tire 0 (T0) and Tire 1 (T1), on dry and wet surfaces, are plotted in Figure 2.9 using the parameters presented in Table A.3. Corresponding normalized characteristics $\mu(\kappa)$ and A are plotted in Figure 2.10. The peak of the F_x plot demonstrates that the Tire 0 has higher force generation capability than Tire 1 and shows the trade-off in tire performance associated with the selection of a low rolling resistance tire. The force derivative plot shows that Tire 0 has a greater initial stiffness $C_{F\kappa}$ than Tire 1 and that $C_{F\kappa}$ remains constant for each tire despite a change from dry to wet road surface. The locations of optimal slip on dry κ^*_{dry} and wet κ^*_{wet} surfaces, corresponding to the peaks of the F_x and $\mu(\kappa)$ plots, occur at the κ-intercept of the force derivative and adhesion gradient plots (according to (2.24)) and differ between tire types and road conditions. The adhesion gradient plot shows that, regardless of tire type and road condition, zero tire slip corresponds with unity adhesion gradient.

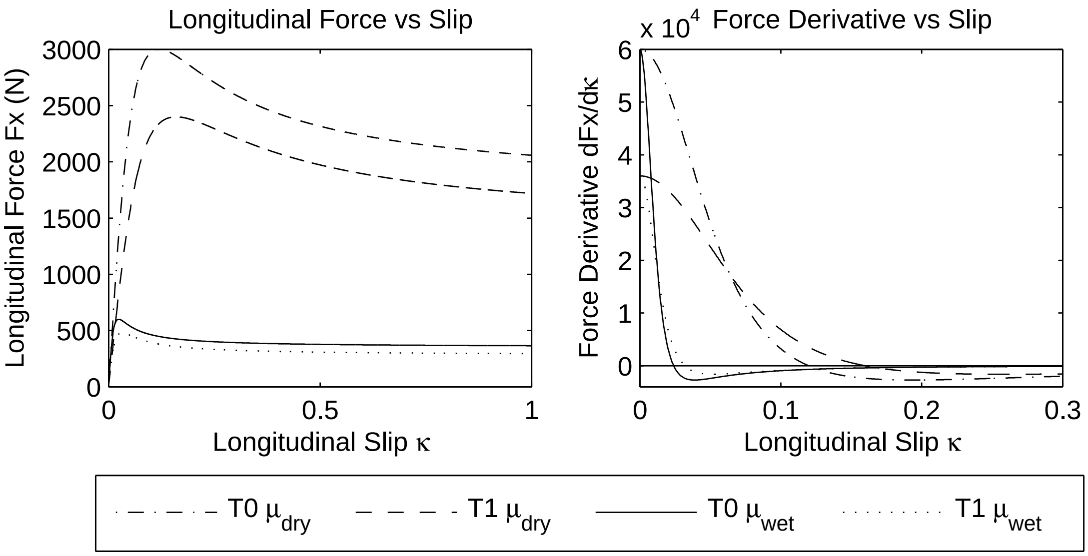


Figure 2.9: Longitudinal Force and Force Derivative

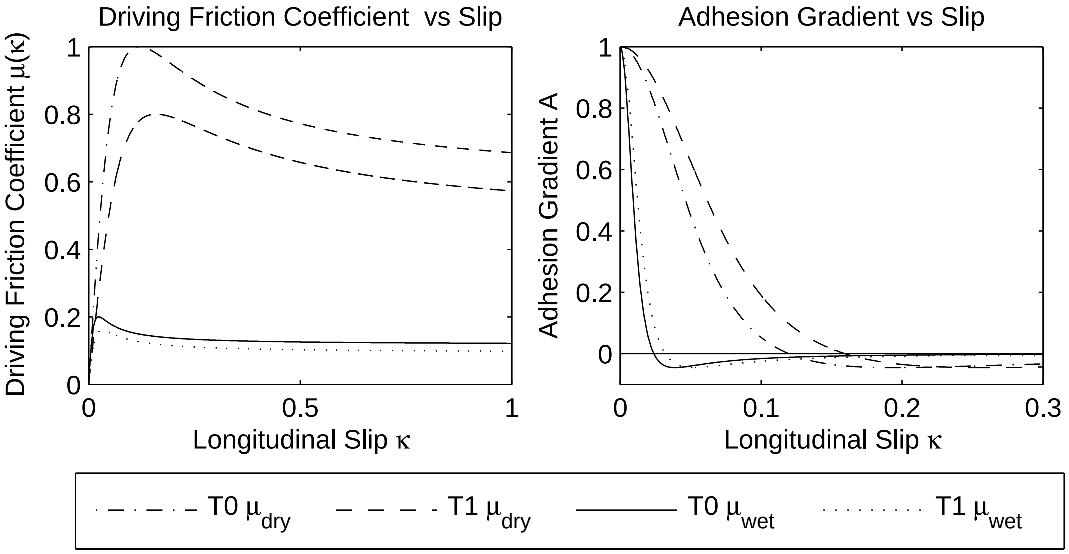


Figure 2.10: Driving Friction Coefficient and Adhesion Gradient

2.4 Plant Model Summary

The traction control system model assumed throughout the remainder of this thesis is summarized in equations (2.29) to (2.40). The model has been augmented with state variables O_T and O_E to measure the duration of and energy usage during the drive cycle. Reference is made to Section 3.3 for the definition of the **fuzzy**() function in (2.40). Also, T_a is assigned T_{ref} during open loop testing of the system.

State equations:

$$\dot{\Omega} = \frac{T_a - T_y - r\,F_x}{J} \tag{2.29}$$

$$\dot{v}_x = \frac{F_x}{m} \tag{2.30}$$

$$\dot{s}_x = v_x \tag{2.31}$$

$$\dot{O}_T = \begin{cases} 1 & s_x \leq S_F \\ 0 & s_x > S_F \end{cases} \tag{2.32}$$

$$\dot{O}_E = \begin{cases} T_a\,\Omega & s_x \leq S_F \\ 0 & s_x > S_F \end{cases} \tag{2.33}$$

Constituent equations:

$$\mu_r = \begin{cases} \mu_{dry} & s_x < S_T \\ \mu_{wet} & s_x \geq S_T \end{cases} \tag{2.34}$$

$$\kappa = -\frac{v_x - r\,\Omega}{r\,\Omega} \tag{2.35}$$

$$\kappa_{eq} = \frac{1}{\mu_r}\,\kappa \tag{2.36}$$

$$F_x = \mu_r D_{xo}\,\sin[C_x\,\tan^{-1}\{B_{xo}\,\kappa_{eq} - E_x(B_{xo}\,\kappa_{eq} - \tan^{-1}(B_{xo}\,\kappa_{eq}))\}] \tag{2.37}$$

$$\begin{aligned} A = \frac{\dot{F}_x}{C_{F\kappa o}} &= \cos[C_x\,\tan^{-1}\{B_{xo}\,\kappa_{eq} - E(B_{xo}\,\kappa_{eq} - \tan^{-1}(B_{xo}\,\kappa_{eq}))\}] \\ &\quad \times \left[\frac{1 - E\left(1 - \dfrac{1}{1 + B_{xo}^2\kappa_{eq}^2}\right)}{1 + \{B_{xo}\,\kappa_{eq} - E(B_{xo}\,\kappa_{eq} - \tan^{-1}(B_{xo}\,\kappa_{eq}))\}^2} \right] \end{aligned} \tag{2.38}$$

$$T_y = r\,C_r\,F_{zo} \tag{2.39}$$

$$T_a = \begin{cases} \text{fuzzy}\,(T_{ref}/T_{max}, A) \times T_{max} & \text{Traction Control (Closed Loop)} \\ T_{ref} & \text{No Traction Control (Open Loop)} \end{cases} \tag{2.40}$$

Chapter 3

Control

A traction control strategy, taken from literature [12], is reviewed and analyzed with the goal of identifying controller design variables that impact vehicle (plant) performance and energy use. An overview of the operating principles and control techniques of an electric vehicle traction control system provides justification for selecting a fuzzy logic control strategy. After a review of the operation of a fuzzy logic controller, application-specific details are presented to describe a fuzzy logic traction controller.

3.1 Traction Control Systems

Traction control improves safety during bad weather driving and reduces wear on tires by preventing excessive wheel slip. Compared to a conventional ICE vehicle, the electric motor and drive circuitry of an EV allows quick and accurate torque generation and measurement. Thus, the potential exists for high performance EV motion control [11] with applications in antilock braking [13], traction control [12], and electronic stability control [35]. An EV traction control system benefits from the possibility of operating tires closer to limits of performance while rapidly responding to changes in road conditions. Thus, it has been suggested that such a system may allow the use of energy efficient low rolling resistance tires that would otherwise be impractical due to an associated trade-off in tire performance [12].

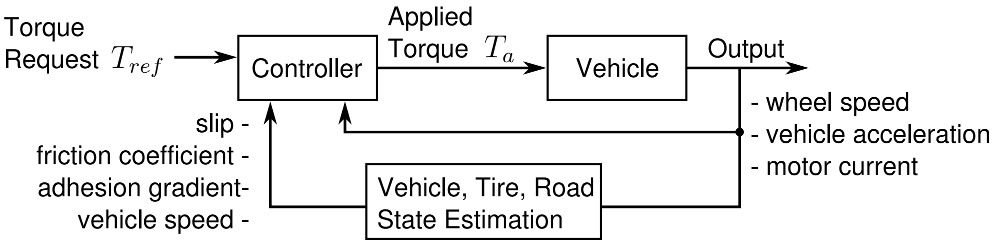


Figure 3.1: General Structure of a Traction Control System

The general form of a traction control system is shown in Figure 3.1 where a traction controller limits torque application to the wheels to satisfy driver torque request without exceeding tire performance limitations (as determined through sensor measurements and state estimation). The limit of tire performance is defined as the maximum force that may be generated between tire and road. As discussed in Section 2.3.1, tire performance may be described in terms of longitudinal slip κ, driving friction coefficient $\mu(\kappa)$ (directly related to tire force), and adhesion gradient A (directly related to the derivative of tire force). Typical performance characteristics for a tire on a dry and wet road are illustrated in Figure 3.2. Ideally, torque is controlled to limit wheel slip to an optimal and maximum value κ^* that corresponds to the peak in $\mu(\kappa)$. For $\kappa < \kappa^*$, the longitudinal force produced by the tire exerts a moment about the wheel that serves to stabilize wheel rotational speed (recall (2.29)). If the wheel is accelerated to higher values of slip, a decrease in longitudinal force causes a decrease in the moment opposing tire rotation and leads to further wheel acceleration and increase in slip. Thus, κ^* defines regions of stable and unstable tire operation as shown in Figure 3.2. Operation in the unstable region leads to high wheel speeds and decrease in vehicle performance. To return to the stable region, a traction control system reduces applied torque to the affected wheel.

In practice, the position of the peak and optimal slip κ^* are unknown since they vary for different road conditions as illustrated in Figure 3.2. Available vehicle measurements are typically limited to those provided by vehicle accelerometers, wheel speed sensors, and motor current sensors. Therefore, the estimation of vehicle

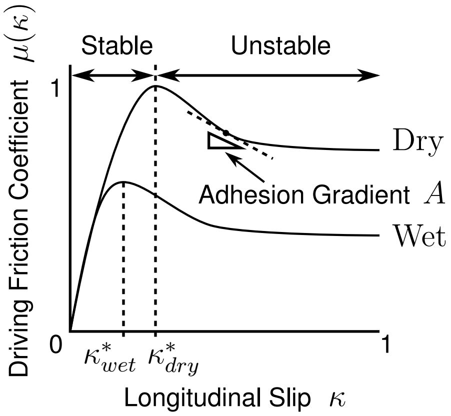


Figure 3.2: Driving Friction Coefficient [36]

states and tire/road force generation characteristics including vehicle speed, wheel slip, coefficient of friction, and adhesion gradient is an important aspect of traction control systems. The following discussion provides a brief overview of estimation techniques with references to further discussion:

- Determination of vehicle speed is important to enable the calculation of wheel slip (2.12). Vehicle speed may be obtained by an average of non-driven wheel speeds using (2.10). In the case of four-wheel drive vehicles, vehicle speed may be derived by integrating accelerometer output [35].

- Estimation of the coefficient of friction requires knowledge of the driving force between tire and road. In an EV, a driving-force observer [37] may be used to obtain this information due to the ability to easily deduce motor torque from measured current. With an ICE, in-vehicle torque measurement or estimation is impractical [37] and contributes to difficulty in accurately controlling torque output. As a result, commercially available traction control systems for ICE vehicles limit slip to a suboptimal prefixed value that accommodates worst case scenarios such as operation on ice [13].

- Adhesion gradient may be obtained by observing the changes in slip and coefficient of friction over time:

$$A = \frac{d\mu/dt}{d\kappa/dt} = \frac{d\mu}{d\kappa}.$$

An identification algorithm may be used to extract A from a set of μ - κ samples [37] or an approximation may be obtained by considering the ratio of changes in sequentially observed samples (with appropriate consideration of the singularity at steady state conditions when $\triangle\kappa = 0$) [12]:

$$A \approx \frac{\triangle\mu}{\triangle\kappa}.$$

Extensions have also been made to estimate the optimal κ^* resulting in peak μ [37].

Proposed traction control strategies include the control of wheel speed, wheel slip, and adhesion gradient. The advantages and disadvantages of each are summarized in Table 3.1. Fuzzy logic control of adhesion gradient offers potential as it offers simplistic practical implementation with little modeling and computational requirements. By tracking a small positive adhesion gradient, the traction control problem becomes independent of location of optimal slip and does not require prior knowledge of road conditions [12]. An adhesion gradient fuzzy logic controller is defined in Section 3.3 after a review of fuzzy logic control in Section 3.2. Fuzzy logic controllers typically require a tuning procedure based on designer intuition and experimental results [12]. In this work, the fuzzy controller design forms a part of the overall system optimization in Chapter 4.

Table 3.1: Traction Control Strategies

Strategy	**Control Technique**	**Advantage**	**Disadvantage**	**Ref.**
Maintain wheel speed in accordance with predicted wheel speed	Model-Following	Effective.	Requires vehicle model in control design.	[11]
Limit high wheel speeds using natural torque-speed characteristics of a DC motor	Feed-Forward	Fast acting. No computation.	Design must include DC motor. Speed limitation not enough to regain stability.	[10]
Slip Control	Proportional-Integral	Effective.	Requires complex estimation of optimal slip.	[11]
Adhesion Gradient Control	Sliding-Mode	Robust against disturbances.	High (infinite) switching frequencies.	[12]
Adhesion Gradient Control	Fuzzy Logic	Model independent. Practical implementation.	Tuning based on experience and experimental results.	[12]

3.2 Fuzzy Logic Control

The development of fuzzy logic control has been inspired by the capacity of humans to reason with uncertainty, yet reliably act and react in complex environments. By mimicking the human capacity to reason approximately, fuzzy logic control has been successful in controlling a broad range of nonlinear systems where classical control has not been effective or efficient [38]. In particular, fuzzy logic control is being considered in emerging automotive control systems to address the uncertainties and nonlinearities present in active suspension [39], regenerative braking [40], traction control [12], vehicle stability control [41], and hybrid powertrain control [6].

The structure of a fuzzy logic control system is shown in Figure 3.3. A fuzzy logic controller is a nonlinear mapping between its inputs x and outputs y [42]. The mapping is accomplished by operations on fuzzy sets. Therefore, this section begins with an overview of fuzzy sets and set membership. This is followed by a detailed explanation of the components of the fuzzy logic controller divided into: controller input, output, and (de)normalization; input fuzzification; rule base and inferencing; and output defuzzification. Finally, active research areas in fuzzy logic control are discussed.

3.2.1 Fuzzy Sets

Define the **universe of discourse**, or **universe**, as the collection of all available information on a given problem [38]. This universe may be represented mathematically as a set X containing information as elements x. The elements, x, may therefore be grouped according to properties associated with the underlying information. When these properties are distinct and readily perceived, the groupings may be defined with “crisp” boundaries [38]. In such cases, a crisp boundary defines a **crisp set** that is analogous to a classical set as described in Section 5.3.1. For example, each element $x \in X$ may therefore be mapped to a membership set containing elements 1 or 0, depending, respectively, on whether or not it is a member of A. The mapping, called $\chi_A(x)$ therefore expresses the “membership” of an

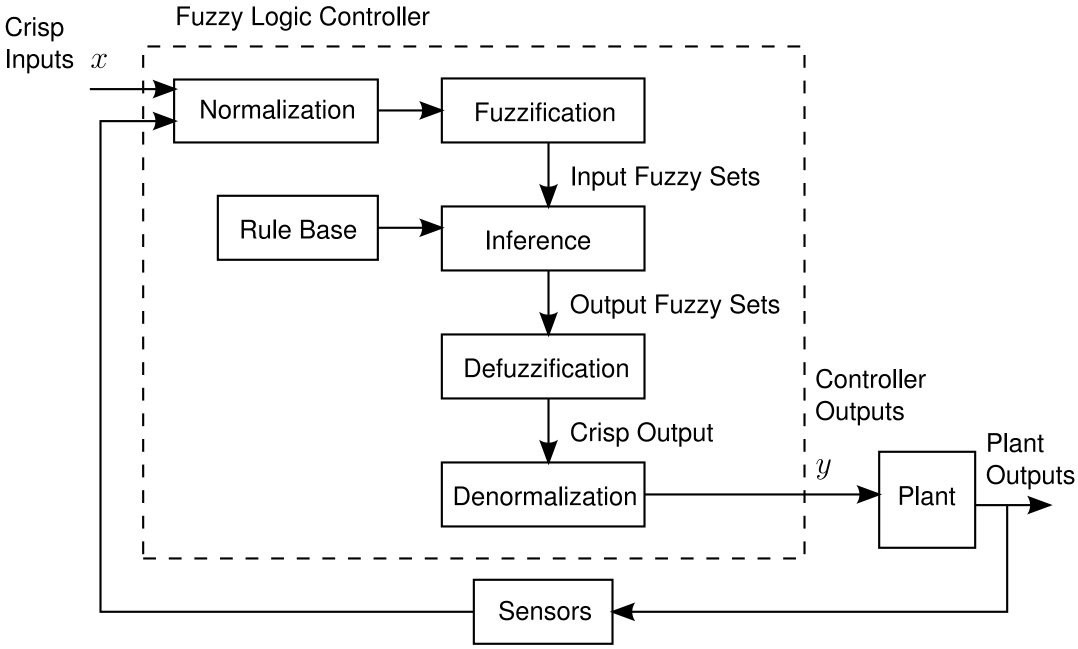


Figure 3.3: Fuzzy Logic Control System [38]

element x in the crisp set A as [38]:

$$\chi_A(x) = \begin{cases} 1, & x \in A \\ 0, & x \notin A. \end{cases} \tag{3.1}$$

Example 3.2.1. Figure 3.4a shows an abstraction of the universe as a set X, along with a crisp boundary defining a crisp set A. The solid line surrounding elements of the crisp set A indicates a clear distinction between members of A and the rest of the elements in X. Point a is a member of the crisp set A with $\chi_A(a) = 1$ while point b is not a member with $\chi_A(b) = 0$.

Often, the properties of information in the universe are not easily distinguished due to uncertainty, ambiguity, or vagueness. Information, again treated as elements x, may still be grouped, but by an equally uncertain, ambiguous, or vague boundary that defines a **fuzzy set** [38]. Like its crisp counterpart, a fuzzy set may include or exclude an element x. However, fuzzy sets can be considered as an extension to

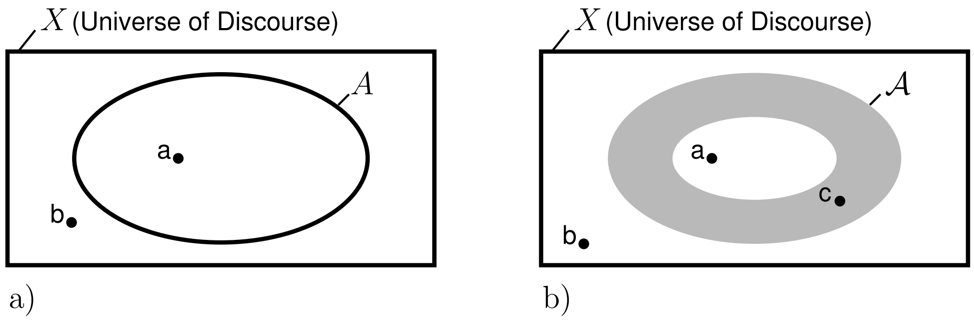


Figure 3.4: Comparison of a) Crisp Set Boundary and b) Fuzzy Set Boundary [38]

crisp sets since they also allow an element to be partially included in their group. As a result, there exists a gradual transition between membership and nonmembership of elements in a fuzzy set.

The membership of an element in a particular fuzzy set is described by mapping each element to a set of **membership values** on the interval 0 to 1, where 0 corresponds to nonmembership, 1 to complete membership, and values in between to partial membership. To avoid confusion between crisp and fuzzy sets, a fuzzy set is denoted in calligraphic type. The fuzzy set "A" is therefore denoted $\mathcal{A}$. The membership of an element x in a fuzzy set $\mathcal{A}$ is defined by the **membership function** $\mu_{\mathcal{A}}(x)$ as [38]:

$$\mu_{\mathcal{A}}(x) \in [0, 1]. \tag{3.2}$$

Example 3.2.2. The fuzzy set $\mathcal{A}$ shown in Figure 3.4b lacks a clear boundary but is defined by a shaded membership transition region. Point a is a full member of $\mathcal{A}$, with $\mu_{\mathcal{A}}(a) = 1$, as it resides in the central (unshaded) region. Point b is not a member, with $\mu_{\mathcal{A}}(b) = 0$, since it is completely outside the transition region. Point c has partial membership, $0 < \mu_{\mathcal{A}}(c) < 1$, since it resides in the transition region.

3.2.2 Properties of Membership Functions

Although the example fuzzy set $\mathcal{A}$ in Figure 3.4b clearly shows that points a and b are full members and nonmembers, respectively, the membership value of point

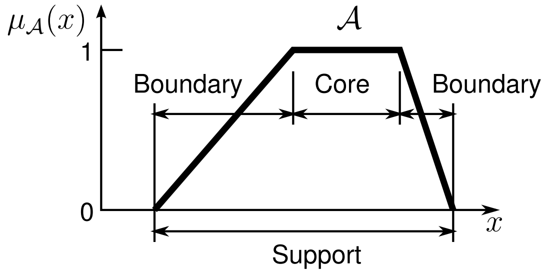


Figure 3.5: Core, Support, and Boundaries of a Fuzzy Set [38]

c is not clear from the figure. The membership values can be shown explicitly for elements x by directly plotting $\mu_{\mathcal{A}}(x)$ as a "height" against x [38]. For the usual case where x is one dimensional, the horizontal axis represents the elements x while the membership values are indicated by the height along the vertical axis (Figure 3.5). From here on, membership functions will be presented in this height-based manner.

For a membership function $\mu_{\mathcal{A}}(x)$, three regions can be distinguished (Figure 3.5) depending on the membership of elements x [38]:

- The **support** of a membership function is the region of the universe that is associated with nonzero membership in the fuzzy set: $\mu_{\mathcal{A}}(x) > 0$.
- The **core** of a membership function is the region of the universe that is associated with complete membership in the fuzzy set: $\mu_{\mathcal{A}}(x) = 1$.
- The **boundary** of a membership function is the region of the universe that is associated with partial (that is nonzero, but not complete), membership in the fuzzy set: $0 < \mu_{\mathcal{A}}(x) < 1$.

The regions of a membership function may be used to characterize a fuzzy set. A **normal fuzzy set** is associated with a membership function whose core comprises at least one element. When the core of a normal fuzzy set contains exactly one element, the element is called the **prototype** of the set.

A **convex fuzzy set** is defined as having a membership function that is either: a) monotonically increasing; b) monotonically decreasing; or c) monotonically increasing then monotonically decreasing. This is equivalently stated as [38]:

$$\forall\, x, y, z \in \mathcal{A},\; x < y < z \Rightarrow \mu_{\mathcal{A}}(y) \geq \min[\mu_{\mathcal{A}}(x), \mu_{\mathcal{A}}(z)].$$

Examples of common membership functions, named according to their shape, include: trapezoidal, gaussian, triangular, and single-point [38]. These membership functions are all normal and convex. These characteristics are useful since the universe is simply divided between a core region of complete certainty and, optionally, a surrounding fuzzy boundary of increasing uncertainty.

The triangular and single-point membership functions are used in this thesis and are further defined [38]:

- A **triangular fuzzy set**, $\mathcal{T}$ (Figure 3.6a), is a normal and convex fuzzy set with a prototype element t and, potentially asymmetric, spans σ^- and σ^+:

$$\mu(x) = \begin{cases} \max\left\{0, 1 + \dfrac{x-t}{\sigma^-}\right\} & \text{if } x \leq t \\ \max\left\{0, 1 + \dfrac{t-x}{\sigma^+}\right\} & \text{otherwise.} \end{cases} \tag{3.3}$$

- A **single-point fuzzy set**, $\mathcal{P}$ (Figure 3.6b), can be considered as a triangular fuzzy set with span approaching zero and prototype element p. A single-point fuzzy set is also called a **fuzzy number**.

  Note. Although a single-point fuzzy set and a single-element crisp set may both contain the same element, the two sets differ since the single-point fuzzy set allows partial membership while the crisp set does not (recall (3.2) and (3.1), respectively). Partial membership is possible when element x is unknown. This is the case prior to defuzzification (Section 3.2.6).

Since an element x can have simultaneous membership in multiple fuzzy sets in the same universe, it is convenient to visualize all membership functions at once by plotting them on the same axis.

Example 3.2.3. In Figure 3.7, the element $x = t$ has simultaneous memberships in fuzzy sets $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$ of 0, 0.2, and 0.8, respectively.

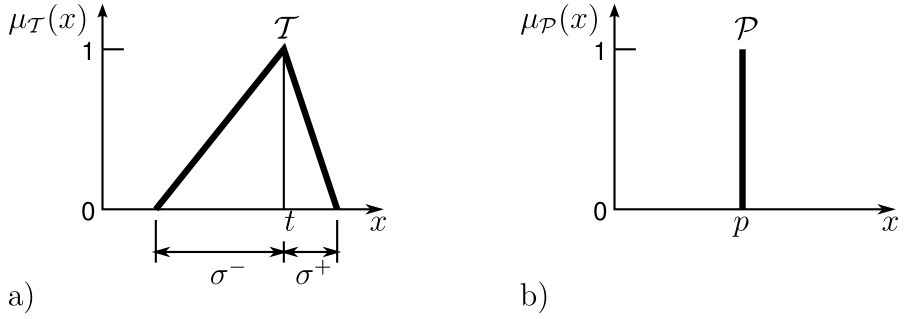


Figure 3.6: Membership Functions: a) Triangular, and b) Single-Point [38]

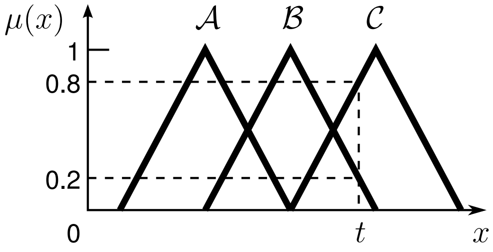


Figure 3.7: Membership in Multiple Fuzzy Sets

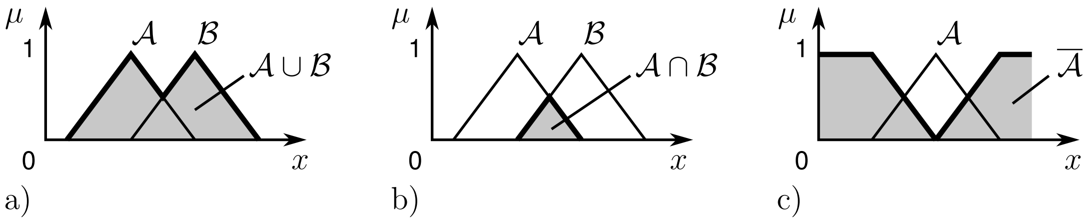


Figure 3.8: Fuzzy Set a) Union, b) Intersection, c) Complement [38]

Given fuzzy sets $\mathcal{A}$ and $\mathcal{B}$ in a universe X, the following standard operations on fuzzy sets are defined and illustrated in Figure 3.8 [38]:

$$\textbf{Union}\ \mu_{\mathcal{A}\cup\mathcal{B}}(x) = \max[\mu_{\mathcal{A}}(x), \mu_{\mathcal{B}}(x)] \tag{3.4}$$

$$\textbf{Intersection}\ \mu_{\mathcal{A}\cap\mathcal{B}}(x) = \min[\mu_{\mathcal{A}}(x), \mu_{\mathcal{B}}(x)] \tag{3.5}$$

$$\textbf{Complement}\ \mu_{\overline{\mathcal{A}}}(x) = 1 - \mu_{\mathcal{A}}(x). \tag{3.6}$$

The intersection operation will be given further consideration in Section 3.2.5 for its role in the inferencing process.

3.2.3 Controller Input and Output Space

As shown in Figure 3.3, the fuzzy logic control system under consideration consists of a plant, sensors, and a fuzzy logic controller (delimited by a dashed boundary). Although the fuzzy logic controller reasons using fuzzy sets, it communicates with the outside world through crisp, or single-valued (real-valued), inputs and outputs. Feedback signals (such as voltage) obtained from sensors are crisp since they do not explicitly contain information about uncertainty, though uncertainty is implied due to sensor performance limitations. The control action issued by the controller is also crisp, and produces physical forcing signals that are unambiguous to the plant. External reference signals are typically crisp as they specify numerical set-points. Therefore, a fuzzy logic controller operates on crisp inputs and produces crisp outputs.

Given a controller with n independent inputs and m outputs:

- Each input x_i is associated with a universe X_i, $i = 1, \ldots, n$, where X_i represents all possible crisp values for input x_i.

- Each output y_j is associated with a universe $Y_j, j = 1, \ldots, m$, where Y_j represents all possible crisp values for output y_j.

It is assumed that the elements of X_i and Y_j are real numbers since they represent possible physical quantities of crisp control signals.

Also, it is optional, but often convenient, to consider normalized control signals. This, for example, allows the controller to work with percentages instead of awkward full-scale values. Therefore, the elements of X_i are often normalized between $[0, 1]$, or some other convenient range. Correspondingly, elements of Y_j are denormalized to full-scale values suitable for application on the plant. The normalization and denormalization operations are shown in Figure 3.3 and are optionally the first and last stages, respectively, in a fuzzy logic controller.

3.2.4 Fuzzification

Fuzzification is defined as the process of making a crisp quantity fuzzy [38]. In a fuzzy logic controller, inputs are crisp quantities that require fuzzification prior to the inferencing stage. For each input x_i, its universe X_i is divided into k_i fuzzy sets $\mathcal{A}_i^l$, for $i = 1, \ldots, n$, and $l = 1, \ldots, k_i$. Fuzzification is then simply the process of determining the membership of x_i in each of the k_i fuzzy sets of X_i.

It is common to take $\mathcal{A}_i^l = \mathcal{T}_i^l$ to divide each universe X_i into k_i triangular fuzzy sets, $\mathcal{T}_i^l$, $l = 1, \ldots, k_i$. The triangular fuzzy sets are defined such that prototype elements t_i^l are distributed over the elements $x_i \in X_i$ with spans σ^-, σ^+ to adjacent prototype elements. The resulting overlapping sets ensure that a continuous, varying, control input gradually changes in membership from one fuzzy set to another.

Example 3.2.4. An example of dividing a universe X_i into 4 triangular fuzzy sets $\mathcal{T}_i^k$, $k = 1, \ldots, 4$, over all values of input $x_i \in [0, 1]$ is shown in Figure 3.9. The prototype elements t_i^k are evenly spaced at $0, 1/3, 2/3$, and 1 and all spans

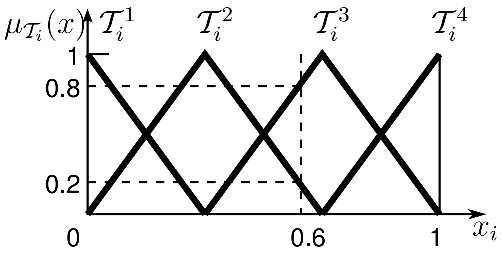


Figure 3.9: Division of Universe X_i into Triangular Fuzzy Sets

$\sigma_i^{k-} = \sigma_i^{k+} = 1/3$. Further, input $x_i = 0.6$ has been fuzzified and the resulting membership values, $\mu_{\mathcal{T}_i^k}(x_i)$, as calculated by (3.3) are 0, 0.2, 0.8, and 0, for $k = 1, \ldots, 4$, respectively.

3.2.5 Inferencing

The inferencing block in a fuzzy logic controller, as shown in Figure 3.3, determines appropriate control actions in response to controller inputs. The input-output relationship is stored in a rule base, also shown in Figure 3.3, that is accessed by the fuzzy inferencing system. The rule base represents human knowledge about control procedures encoded as a series of IF-THEN rules. Each rule takes the form of a natural language expression of the type [38]:

IF premise (antecedent), THEN conclusion (consequent).

The premise consists of a series of statements about the inputs x_i in terms of their membership in fuzzy sets $\mathcal{A}_i$. The membership for a particular input i in a fuzzy set $\mathcal{A}_i$ is described by an expression of the type:

x_i is $\mathcal{A}_i$

where the notation $\mathcal{A}_i$ is used to designate an arbitrary fuzzy set $\mathcal{A}_i^l$, $l \in \{1, \ldots, k_i\}$, in the universe X_i. The certainty, or truth, of the statement is determined by the membership of x_i in $\mathcal{A}_i$.

Using the method of **multiple conjunctive antecedents** [38], the complete state of n inputs may be described in the premise as:

$$x_1 \text{ is } \mathcal{A}_1 \text{ and } x_2 \text{ is } \mathcal{A}_2 \ldots \text{ and } x_n \text{ is } \mathcal{A}_n$$

where the "and" connectives invoke the notion of fuzzy set intersections as defined in (3.5). The entire premise is evaluated by means of the multidimensional membership function of the fuzzy set $\mathcal{A}_S = \mathcal{A}_1 \cap \mathcal{A}_2 \cap \ldots \cap \mathcal{A}_n$ given by:

$$\mu_{\mathcal{A}_S}(x_1, x_2, \ldots, x_n) = \prod_{i=1}^{n} \mu_{\mathcal{A}_i} = \min[\mu_{\mathcal{A}_1}(x_1), \mu_{\mathcal{A}_2}(x_2), \ldots, \mu_{\mathcal{A}_n}(x_n)]. \qquad (3.7)$$

Note. The $\prod$ operator, as applied to n membership functions, denotes taking the minimum of the membership functions as defined in (3.7).

Note. The "or" connective is used in the method of **multiple disjunctive antecedents** [38] to formulate a premise of the form:

$$x_1 \text{ is } \mathcal{A}_1 \text{ or } x_2 \text{ is } \mathcal{A}_2 \ldots \text{ or } x_n \text{ is } \mathcal{A}_n.$$

The "or" connectives invoke the notion of fuzzy set unions as defined in (3.4). The entire premise is evaluated in this case by means of the membership function of the fuzzy set $\mathcal{A}_S = \mathcal{A}_1 \cup \mathcal{A}_2 \cup \ldots \cup \mathcal{A}_n$ given by:

$$\mu_{\mathcal{A}_S}(x_1, x_2, \ldots, x_n) = \max[\mu_{\mathcal{A}_1}(x_1), \mu_{\mathcal{A}_2}(x_2), \ldots, \mu_{\mathcal{A}_n}(x_n)].$$

The conclusion of an IF-THEN rule describes the outputs y_j in terms of membership in output fuzzy sets. This may be done by dividing each output universe Y_j into k_j fuzzy sets $\mathcal{C}_j^l$, for $j = 1, \ldots, m$, and $l = 1, ..., k_j$. The membership for a particular output j in a fuzzy set $\mathcal{C}_j$ (denoting an arbitrary fuzzy set in Y_j) is described by an expression of the type:

$$y_j \text{ is } \mathcal{C}_j.$$

The truth, or certainty of the conclusion is "inferred" from, and therefore identical to, the truth of the premise. For a premise with multiple conjunctive antecedents, the certainty of the conclusion is therefore given by (3.7).

Hence, for a controller with n inputs and m outputs, the IF-THEN rules are of the form:

$$\text{IF } x_1 \text{ is } \mathcal{A}_1 \text{ and } x_2 \text{ is } \mathcal{A}_2 \ldots \text{ and } x_n \text{ is } \mathcal{A}_n,$$
$$\text{THEN } y_1 \text{ is } \mathcal{C}_1,\ y_2 \text{ is } \mathcal{C}_2, \ldots, y_m \text{ is } \mathcal{C}_m.$$

Note that the inference process is individually applied to each output y_j listed in the rule. This allows a fuzzy logic controller with m outputs to be designed as m single-output controllers. Therefore, multi-input single-output (MISO) fuzzy logic controllers may be considered without loss of generality.

For a MISO fuzzy logic controller with n inputs where each universe X_i has been divided into k_i fuzzy sets, the rule base may contain a maximum number of R rules [38]:

$$R = k_1 k_2 k_3 \ldots k_n. \tag{3.8}$$

For a controller with a small number of inputs ($n = 1, \ldots, 3$), a **fuzzy associative memory table**, or FAM table, allows a compact graphical representation of the rule base [38]. A controller with a small number of inputs and m outputs may be described by m number of FAM tables.

Example 3.2.5. An example of a FAM table, for two inputs ($n = 2$), is shown in Figure 3.10. The universe X_1 of the first input x_1 has been divided into 3 fuzzy sets $\mathcal{A}^1, \mathcal{A}^2$, and $\mathcal{A}^3$. The universe X_2 of the second input x_2 has been divided into 2 fuzzy sets $\mathcal{B}^1$, and $\mathcal{B}^2$. The universe Y of the output y has been divided into 4 fuzzy sets $\mathcal{C}^j$, for $j = 1, \ldots, 4$. The FAM table represents the following rule base:

$$\text{IF } x_1 \text{ is } \mathcal{A}^1 \text{ and } x_2 \text{ is } \mathcal{B}^1, \text{ THEN } y \text{ is } \mathcal{C}^1.$$
$$\text{IF } x_1 \text{ is } \mathcal{A}^1 \text{ and } x_2 \text{ is } \mathcal{B}^2, \text{ THEN } y \text{ is } \mathcal{C}^2.$$
$$\text{IF } x_1 \text{ is } \mathcal{A}^2 \text{ and } x_2 \text{ is } \mathcal{B}^1, \text{ THEN } y \text{ is } \mathcal{C}^2.$$
$$\text{IF } x_1 \text{ is } \mathcal{A}^2 \text{ and } x_2 \text{ is } \mathcal{B}^2, \text{ THEN } y \text{ is } \mathcal{C}^3.$$
$$\text{IF } x_1 \text{ is } \mathcal{A}^3 \text{ and } x_2 \text{ is } \mathcal{B}^1, \text{ THEN } y \text{ is } \mathcal{C}^3.$$
$$\text{IF } x_1 \text{ is } \mathcal{A}^3 \text{ and } x_2 \text{ is } \mathcal{B}^2, \text{ THEN } y \text{ is } \mathcal{C}^4.$$

	Input x_1		
Input x_2	$\mathcal{A}^1$	$\mathcal{A}^2$	$\mathcal{A}^3$
$\mathcal{B}^1$	$\mathcal{C}^1$	$\mathcal{C}^2$	$\mathcal{C}^3$
$\mathcal{B}^2$	$\mathcal{C}^2$	$\mathcal{C}^3$	$\mathcal{C}^4$

Figure 3.10: Double-Input, Single-Output FAM Table

3.2.6 Defuzzification

Defuzzification is the process of making a fuzzy quantity crisp. In a fuzzy logic controller, the output of the inferencing stage is a fuzzy quantity that is described as an element with membership in various output fuzzy sets. These output fuzzy sets are precisely the ones that appear in the conclusions of the IF-THEN rule base. For an output universe Y divided into k fuzzy sets C^l, $l = 1, \ldots, k$, the inferencing process assigns membership to the output y in C^l by evaluating the rule base. The result after the inferencing stage, in general, is that y is a member of multiple output fuzzy sets. However, only the membership information of y is known. The actual value of y is uncertain and requires a defuzzification stage (as shown in Figure 3.3) to produce a crisp controller output.

Note. To avoid confusion, the uncertain value of the output prior to defuzzification is denoted y. The resulting crisp output after defuzzification is denoted y^*.

Defuzzification occurs by weighting the membership information of y and many weighting methods have been proposed due to the freedom in deriving weights from the shapes and locations of membership functions [38]. In general, the choice of defuzzification method is context or problem dependent. However, real-time operation requirements of control systems emphasize the importance of computational simplicity for practical implementations. To this extent, the **zero-order Takagi-Sugeno method**, or **Sugeno method**, makes use of single-point output fuzzy sets and a weighted average method to avoid computational complexity.

The rules in a zero-order Sugeno method are of the form:

$$\text{IF } x_1 \text{ is } \mathcal{A}_1 \text{ and } x_2 \text{ is } \mathcal{A}_2 \ldots \text{ and } x_n \text{ is } \mathcal{A}_n \text{ THEN } y = P^l$$

where P^l is a constant representing the prototype element of the single-point fuzzy set and $l = 1, \ldots, k$ for k divisions of Y [38].

Note. A single output controller is assumed without loss of generality as discussed in Section 3.2.5.

Since single-point output fuzzy sets contain a single element, the output of each rule is a crisp value instead of a range of possible values. The uncertainty in the output of the inferencing process is therefore specified in terms of membership in k possible crisp values. This allows the final crisp output y^* to be determined using the computationally efficient weighted average method [38]:

$$y^* = \frac{\sum_{i=1}^{R} y^{(i)} w^{(i)}}{\sum_{i=1}^{R} w^{(i)}} \tag{3.9}$$

where $R =$ the number of rules in the rule base;
$y^{(i)} =$ the value of y assigned by the i^{th} rule;
$w^{(i)} =$ the membership value assigned by the i^{th} rule (as a result of evaluating the premise).

The inferencing and defuzzification process in the zero-order Sugeno method can be compactly represented by substituting $w^{(i)}$ in (3.9) for the computation by method of multiple conjunctive antecedents [38]:

$$y^* = \frac{\sum_{i=1}^{R} y^{(i)} \prod_{j=1}^{n} \mu_{j,i}}{\sum_{i=1}^{R} \prod_{j=1}^{n} \mu_{j,i}} \tag{3.10}$$

where $\mu_{j,i} =$ the membership value assigned to input j by rule number i;
$\prod_{j=1}^{n} \mu_{j,i} =$ the method of multiple conjunctive antecedents expressed as (3.7):

$$\prod_{j=1}^{n} \mu_{j,i} = \min[\mu_{1,i}\,,\ \mu_{2,i}\,, \ldots,\ \mu_{n,i}] = w_i.$$

Example 3.2.6. Consider a double-input single-output fuzzy logic controller using the zero-order Sugeno method and a rule base as described by the FAM table of Example 3.2.5. Assume that the first input universe X_1 is divided into 3 triangular fuzzy sets $\mathcal{A}^1$, $\mathcal{A}^2$, and $\mathcal{A}^3$ with prototype elements at 0, 0.5, and 1, respectively. Assume that the second input universe X_2 is divided into 2 triangular fuzzy sets $\mathcal{B}^1$, and $\mathcal{B}^2$ with prototype elements at 0 and 1, respectively. Further, assume that the output universe is represented by 4 single-valued fuzzy sets $\mathcal{C}^l = \mathcal{P}^l$, $l = 1, \ldots, 4$ with prototype elements at 0, 1/3, 2/3, and 1, respectively.

The entire process from fuzzification, through inferencing, to defuzzification can be graphically shown in a **fuzzy inference diagram** (Figure 3.11) [43]. Crisp inputs arrive into the controller at the bottom left corner of the diagram and are fuzzified according to the membership functions shown in the two left-most columns. For $x_1 = 0.7$, fuzzification gives $\mu_{\mathcal{A}^l}(x_1) = 0, 0.6, 0.4$ for $l = 1, 2, 3$, respectively. For $x_2 = 0.2$, fuzzification gives $\mu_{\mathcal{B}^l}(x_2) = 0.8, 0.2$ for $l = 1, 2$, respectively.

The right-most column shows the single-point fuzzy sets associated with the output. Together, the three columns represent the rule base encoded in the FAM table. For example, the third rule can be extracted from the figure as:

$$\text{Rule 3)}\quad \text{IF } x_1 \text{ is } \mathcal{A}^2 \text{ and } x_2 \text{ is } \mathcal{B}^1\text{, THEN } y = \mathcal{P}^2 = 2/3.$$

The premise of each rule is evaluated according to the method of multiple conjunctive antecedents (3.7) and the resulting uncertainty is assigned to the conclusion in the form of a membership value in the output fuzzy set. Evaluating Rule 3) using the method of multiple conjunctive antecedents gives a certainty of $w_3 = 0.6$ to the conclusion $y = 2/3$:

$$\prod_{j=1}^{2} \mu_{j,3} = \min[\mu_{1,3}\,,\ \mu_{2,3}] = \min[0.6\,,\ 0.8] = 0.6 = w_3.$$

Finally, defuzzification takes place at the bottom right corner of the diagram where the weighted average is calculated to produce a crisp output. In the example, after all rules have been evaluated, the crisp output y^* is determined by:

$$y^* = \frac{\sum_{i=1}^{6} y^{(i)} \prod_{j=1}^{n} \mu_{j,i}}{\sum_{i=1}^{R} \prod_{j=1}^{n} \mu_{j,i}} = \frac{\frac{1}{3}0.6 + \frac{2}{3}(0.2 + 0.4) + 1(0.2)}{0.6 + (0.2 + 0.4) + 0.2} = 0.57.$$

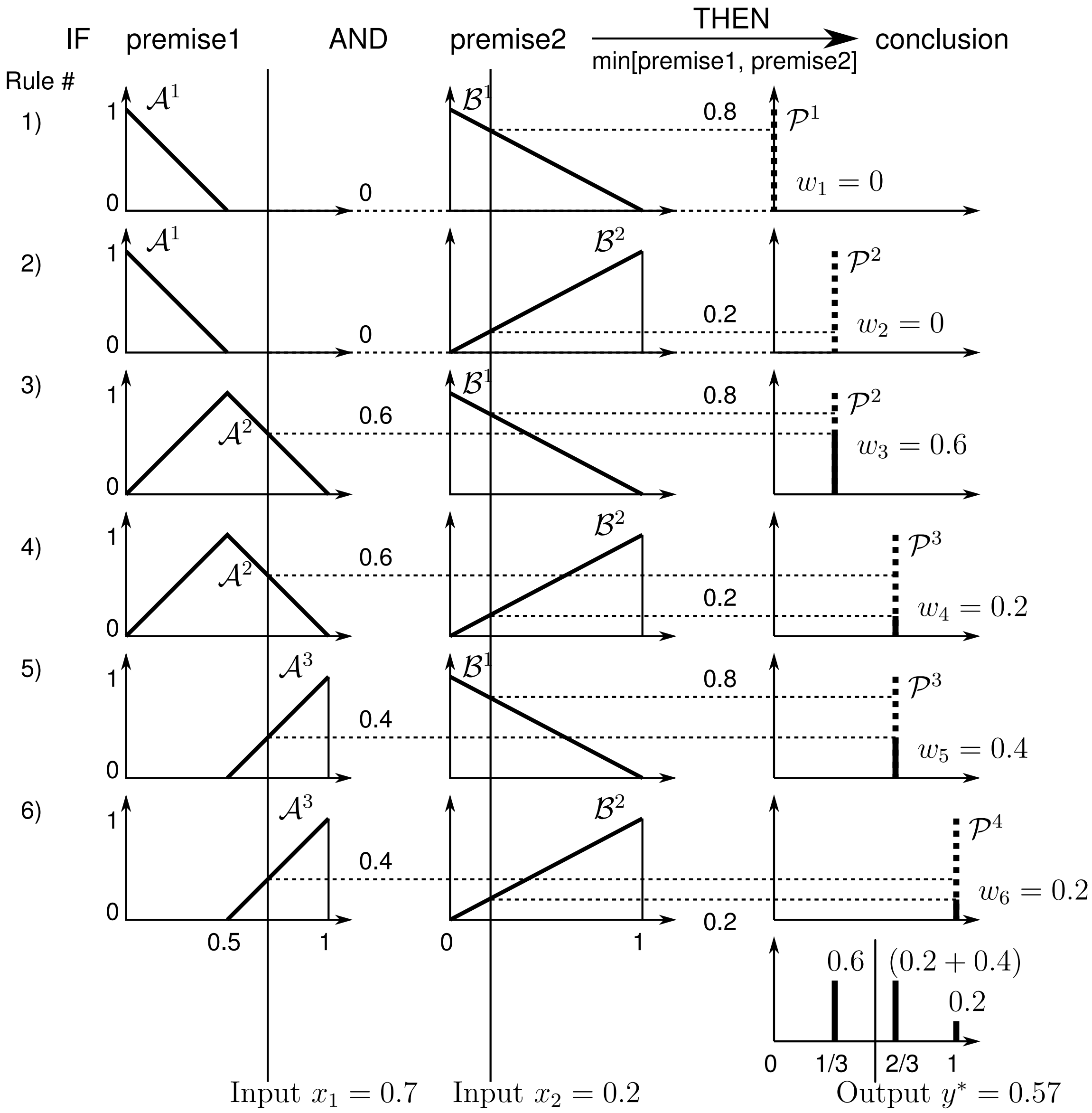


Figure 3.11: Fuzzy Inference Diagram

3.2.7 Research in Fuzzy Logic Control

As a result of on-going research [38], the following aspects of fuzzy logic controller design are discussed separately in this section:

1. choosing the number, shape, and relative placement of membership functions to divide the controller input and output space;
2. choosing appropriate IF-THEN rules for the rule base;
3. stability of fuzzy logic controllers;
4. optimality of fuzzy logic controllers.

The first two aspects, dealing with the choice of controller parameters, are often addressed by engineering intuition where human understanding about the system allows acceptable, but sub-optimal, decisions to be made [38]. For example, the fuzzy logic controllers in [12], [40], and [41] were designed in this manner. In the vehicle control system of [41], a simplified model of nonlinear vehicle dynamics was derived "only to organize [their] imagination in order to extract the fuzzy rules correctly." In all three cases, additional parameter tuning occurred through trial and error.

For even simple systems with few inputs, manual tuning can be time-consuming due to the numerous parameters for describing the membership functions that divide the input and output space. The manual tuning process becomes infeasible for a large number of inputs where the large number of membership function parameters are compounded by an exponential increase in the number of rules (recall equation (3.8)). This problem can be partly alleviated by using methods to automate the creation of membership functions and rule sets based on data sets of input-output data points [38]. For example, instead of using an explicit formula, such as (3.3), to calculate the membership of a data point in a fuzzy set, an artificial neural network (ANN) may be trained to determine the membership of data points in a fuzzy set [38]. This frees the designer from choosing membership function parameters such as the location of prototype elements and the span of triangular membership functions. Other methods such as the "clustering method" and the "modifed learning from example method" prescribe systematic algorithms for calculating both membership functions and rules from input-output data [38]. In these methods, data points from an input-output data set are considered in turn and an algorithm calculates whether the existing rule base can infer the input-output relationship, or, if a new rule is to be added to the rule base. The resulting rule base tends to cluster the data points into fuzzy sets and hence the methods are effective at deriving rules from numerical data, as opposed to human intuition.

Stability and optimality, the last two items from the list at the beginning of this section, are mostly open problems in fuzzy logic controller design [38]. Stability, a fundamental control performance criteria, is not addressed explicitly in fuzzy logic control system design. Although stability criteria have been developed for certain classes of fuzzy logic controllers (such as in [44]), human intuition combined with extensive system simulation and testing is typically relied upon to determine parameters for an acceptable range of stable operation.

The optimality of fuzzy logic controller design parameters is also difficult to ascertain due to the nonlinear nature of both plant and controller in typical fuzzy logic control systems. Fuzzy logic controller parameter optimization can therefore be considered to be a nonlinear optimization problem with a multidimensional search space involving both discrete variables (in the number of rules and fuzzy sets) and continuous variables (in the distribution of membership functions). Genetic algorithms, with their ability to work with such an irregular search space, are an effective and widely used means to optimize fuzzy logic controllers [38]. After genetic algorithms are introduced in Section 5.1, their application to the optimization of fuzzy logic controllers is considered separately in Section 5.1.5.

3.3 Adhesion Gradient Traction Controller

Among proposed traction control strategies (Table 3.1) the fuzzy logic control of adhesion gradient is chosen for further study as it allows simplistic implementation with minimal modeling and computational requirements (Section 3.1). The adhesion gradient fuzzy logic controller is specified through discussion on traction control-specific inputs and outputs, fuzzification and defuzzification membership functions, and rule base.

When possible, numerical parameters involved in defining the fuzzy logic controller are specified intuitively to minimize the number of controller design variables. Otherwise, the discussion highlights the remaining parameters as variables for further optimization in Chapter 4. The numerical values of all control parameters (including nominal values assigned to identified optimization variables) are listed in Table A.4 of Appendix A.

3.3.1 Controller Inputs and Output

The double-input, single-output adhesion gradient-based fuzzy logic controller assumed in this thesis is of the same structure as presented in [12]. The inputs are the driver torque request T_{ref}, derived from accelerator pedal position, and the tire-road adhesion gradient A as determined, in practice, by a vehicle state estimator. Controller output is an applied torque T_a to the driven wheel.

Note. It is assumed that the adherence gradient A is directly measurable in this thesis to reduce the number of design variables. In practice, additional variables are associated with estimator design and it is noted that simultaneous optimization of plant, controller, and estimator is an open area of research [45].

For convenience, the fuzzy logic controller is designed using normalized inputs and outputs. The torque request input T_{ref} is normalized with respect to a maximum motor torque T_{max} such that $T_{ref} \leq T_{max}$. Normalized adhesion gradient A is calculated from the plant model by (2.38). A gain of T_{max} is applied to the controller to denormalize the output torque T_a. Hence, the nonlinear mapping between input and output can be denoted by a function **fuzzy**() such that:

$$T_a = \text{fuzzy}\left(T_{ref}/T_{max}, A\right) \times T_{max}. \tag{3.11}$$

3.3.2 Membership Functions

The normalized torque request input is fuzzified using the membership functions of Figure 3.12 which have been given linguistic labels: Small (S), Medium-Small (MS), Medium-High (MH), and High (H). As indicated in the figure, the membership functions are evenly spaced to represent a linear transition from zero to maximum torque request.

The normalized adhesion gradient is fuzzified using the membership functions of Figure 3.13 which have been given linguistic labels: Negative (N), Zero (Z), Positive-Small (PS), and Positive-High (PH). As noted in [12] these membership functions are not equally spaced and the work recommends a trial procedure for tuning the location of membership functions. An automated tuning procedure is also possible using the method of genetic-fuzzy control presented in Section 5.1.5 and is taken in this thesis. Although the location of all membership functions may be subjected to the optimization procedure, the following intuition and assumptions constrain all but the location of the Positive-Small membership function:

- Negative values of A indicate tire operation within the unstable region and a reduction in output torque is required. This can be ensured by setting the location of the negative membership function to 0. This choice allows the fuzzy logic controller to identify all negative values of adhesion gradient with 100% certainty.

- Mainly Zero values of A indicate that the system is near the optimum and that wheel speed should not be allowed to increase further. The location of the Zero membership function can be interpreted as the optimal performance set-point of the traction controller. Ideally, the traction controller should limit A to zero, the point of marginal stability. However, to provide a factor of safety against entering the unstable region, the location of the Zero membership function is set to 0.1 to limit the adhesion gradient to 10% of its maximum.

- Positive-Small values of A indicate that wheel acceleration is possible without entering the unstable region. The location of the membership function, A-PS $\in [0.2, 0.8]$, is allowed to vary between the locations of adjacent Zero and Positive High fuzzy sets. The range of variation is restricted to allow a transition boundary of uncertainty between adjacent fuzzy sets.

- Positive-High values of A indicate that the wheel is free to accelerate. The location of the positive high membership function is set to 0.9 to allow a linear input-output torque response while adhesion gradient remains more than 90% of the maximum. This effectively disables traction control for high adhesion gradients corresponding to low values of slip.

The membership functions of the output torque (Figure 3.14) have been given linguistic names: Small (S), Medium-Small (MS), Medium-High (MH), and High (H). They are evenly spaced to allow a linear response to torque request input at Positive-High adhesion gradients.

3.3.3 Rule Base

Inferencing and defuzzification is done with the Sugeno method of Section 3.2.6, due to its computational simplicity, using the output torque membership functions of Figure 3.14 and the rule base shown in Figure 3.15a. The rule base shown in Figure 3.15a is as proposed in [12] and is based on a translation of the above considerations where specifically:

- Small torque requests result in Small output torques regardless of adhesion gradient.

- Negative adhesion gradients indicate system instability and result in Small output torques regardless of torque request to regain stability.

- Positive-High adhesion gradients result in a linear response to torque request ranging from Small to High with matching torque output.

The remaining six rules, identified by $\mathcal{C}_1$ to $\mathcal{C}_6$ in Figure 3.15b, represent a gradual reduction in output torque for decreasing torque request and adhesion gradient. Alternative definition of rule conclusions $\mathcal{C}_1$ to $\mathcal{C}_6$ influence the "aggressiveness" of the output torque reduction in response to decreasing adhesion gradient and approach of tire performance limits. Selection of these rule conclusions is part of the simultaneous plant/controller optimization in Chapter 4.

The nonlinear input to output mapping of the fuzzy logic traction controller, as determined by the **fuzzy**() function of (3.11), is shown in Figure 3.16 for the controller defined by Figures 3.12 - 3.14, 3.15a and assuming the location of Positive-Small A at 0.5. These nominal controller parameters are obtained from [12] and also listed in Table A.4. The surface plot is generated by fuzzification of the range of inputs of T_{req} and A and applying the Sugeno method in (3.10) to obtain output T_a. The figure shows that the traction controller exhibits a linear response in torque request to output torque for large adhesion gradients (small values of slip) and that a nonlinear torque reduction occurs as A approaches negative values.

Since the case study drive cycle adopts a constant input torque request T_{ref}, only a subset of the rule base is stimulated. Torque request T_{ref} and maximum torque T_{max} parameters of Table A.2 have been chosen in the ratio 5/6 to equally stimulate rules incorporating the Medium-High and High torque request input membership functions of Figure 3.12. Thus it is expected that controller tuning efforts using the case study drive cycle will only meaningfully affect rules with conclusions $\mathcal{C}_3$ to $\mathcal{C}_6$.

In addition to the performance and efficiency benefits produced by limiting excessive tire slip, a trade-off in performance and energy consumption exists depending on the amount of torque reduction exerted by the traction controller. Optimal tuning of the controller is required in conjunction with tire selection to realize overall system objectives in performance and fuel economy.

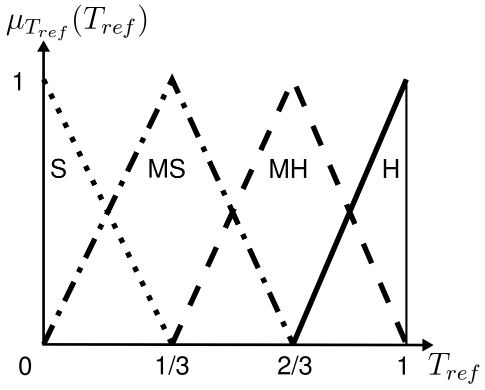


Figure 3.12: Torque Request Input Membership Functions [12]

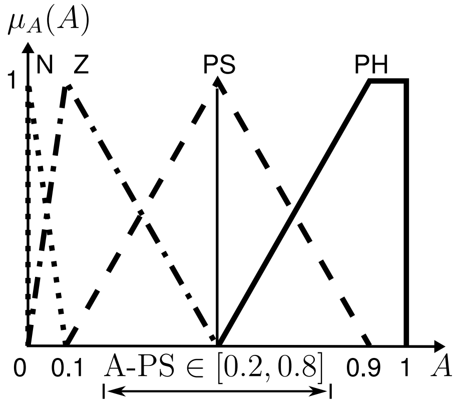


Figure 3.13: Adhesion Gradient Input Membership Functions [12]

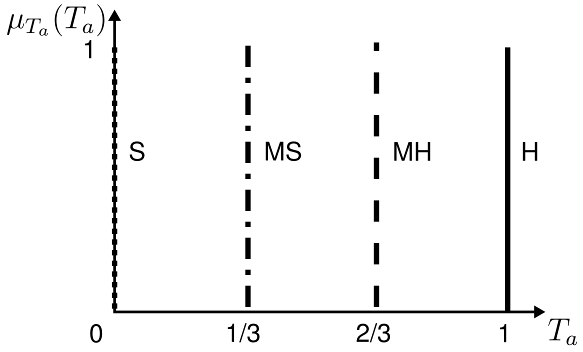
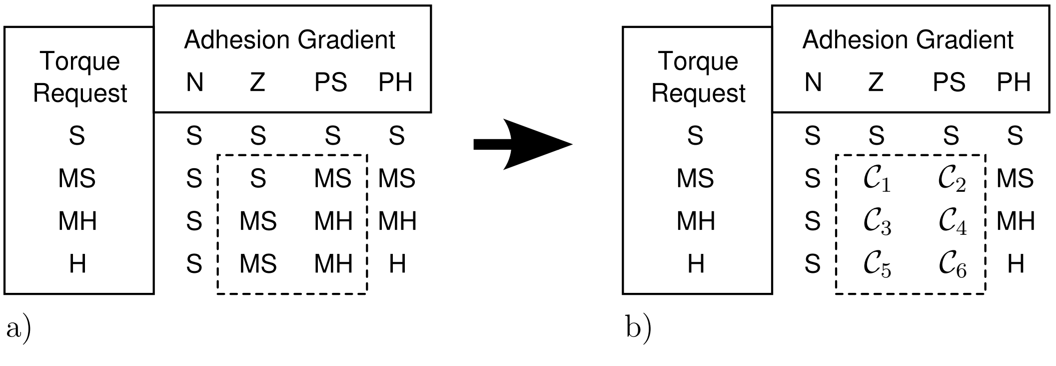


Figure 3.14: Torque Output Membership Function [12]



a)

Torque Request	Adhesion Gradient N	Z	PS	PH
S	S	S	S	S
MS	S	S	MS	MS
MH	S	MS	MH	MH
H	S	MS	MH	H

b)

Torque Request	Adhesion Gradient N	Z	PS	PH
S	S	S	S	S
MS	S	$\mathcal{C}_1$	$\mathcal{C}_2$	MS
MH	S	$\mathcal{C}_3$	$\mathcal{C}_4$	MH
H	S	$\mathcal{C}_5$	$\mathcal{C}_6$	H

Figure 3.15: Rule Base a) Original Rules [12], and b) Rules for Optimization

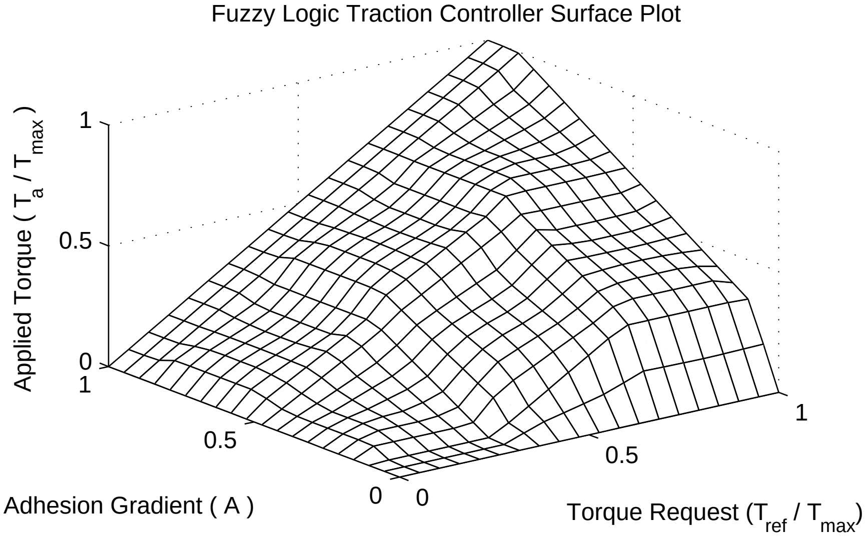


Figure 3.16: Fuzzy Logic Traction Controller Surface Plot

Chapter 4

Optimization Problem Formulation and Solution Strategy

The vehicle model and traction controller developed in Chapter 2 and Chapter 3, respectively, are combined into a system model and a system-level optimization problem is defined with respect to minimizing energy use and drive cycle completion time. A solution strategy based on a genetic algorithm is proposed since the resulting problem includes a mix of continuous and discrete optimization variables and requires simulation of system dynamics to evaluate optimization objectives.

A review of the general coupling between plant and controller design variables and associated plant/controller optimization strategies is followed by development of the traction control system model, the optimization problem formulation (summarized in Section 4.2.4), and the proposed solution strategy.

Design details of the genetic algorithm are discussed separately in Chapter 5. Optimization results using the proposed solution strategy and genetic algorithm are presented in Section 6.4.

4.1 Simultaneous Plant/Controller Optimization

In the design of a system consisting of a plant and controller, it is natural to adopt a sequential strategy where the plant is designed first, followed by the controller [17].

However, potential coupling between plant and controller design requires a concurrent (or simultaneous) strategy to system-level optimization. This section provides a review of the coupling between plant and controller, describes sequential and concurrent strategies for plant/controller optimization, and concludes with a description of the optimization process involving a simulation model.

4.1.1 Plant/Controller Coupling

Plant/controller coupling can be observed by considering the formulation of a combined plant/controller optimization problem as described in [45] and refined in [17]. The following definitions will facilitate further discussion [17]:

- In the notation for a function $f(d; c)$, quantities before the semicolon are variables and quantities after the semicolon are parameters of the function.
- In the optimization of a function f, **parameters** are quantities that are fixed while **variables** are quantities that permit exploration of the search space.

Begin by assuming no coupling between plant and controller optimization. Plant optimization is typically a static optimization problem of the form:

$$\min_{d} O_p(d; c) \quad \textbf{subject to: } h(d) = 0,\ g(d) \leq 0 \tag{4.1}$$

where $d =$ plant design variables
$c =$ plant parameters
$O_p =$ plant objective function
$h, g =$ equality and inequality constraints, respectively, on plant design variables.

In contrast, the controller optimization problem is a dynamic optimization problem of the form:

$$\min_{x(t),z(t)} O_c(x(t), z(t); w) \quad \textbf{subject to: } \dot{x}(t) = f(x(t), z(t), t),\ x(t_o) = x_o \tag{4.2}$$

where $t =$ time

$f =$ system dynamic equations
$x =$ system states as governed by f
$z =$ variable control input (as determined by a control law)
$w =$ control parameters
$O_c =$ controller objective function
$t_o,\ x_o =$ initial time and conditions, respectively.

In general, a tight coupling may exist between plant and controller optimization. In other words, variables and parameters chosen during plant design may influence controller performance and vice versa. Accounting for such a coupling, the individual plant and controller optimization problems (4.1) (4.2) are combined into a system-level optimization problem (Figure 4.1) [17]:

$$\begin{aligned} \min_{d,x(t),z(t)} \quad & O_s\{O_p(d;c), O_c(x(t),z(t);w)\} \\ \textbf{subject to:} \quad & h(d;c) = 0,\ g(d;c) \leq 0, \\ & c = \{a,b\},\ v = V(d;c), \\ & \dot{x}(t) = f(x(t),z(t),t,d;w),\ x(t_o) = x_o, \\ & w = \{u,v\},\ b = B(x(t),z(t),t;w) \end{aligned} \tag{4.3}$$

where, additionally, $O_s =$ system-level optimization objective
$c = \{a,b\} =$ set of plant parameters
$a =$ simple plant parameters
$b =$ plant parameters affected by control requirements
$B =$ function to calculate control requirements
$w = \{u,v\} =$ set of controller parameters
$u =$ simple control parameters
$v =$ control parameters affected by plant design
$V =$ function to calculate plant design requirements.

Note. The system-level optimization problem stated in (4.3) may also be considered as a multidisciplinary optimization (MDO) problem since the overall system may involve mechanical, hydraulic, control, and other subsystem designs [18].

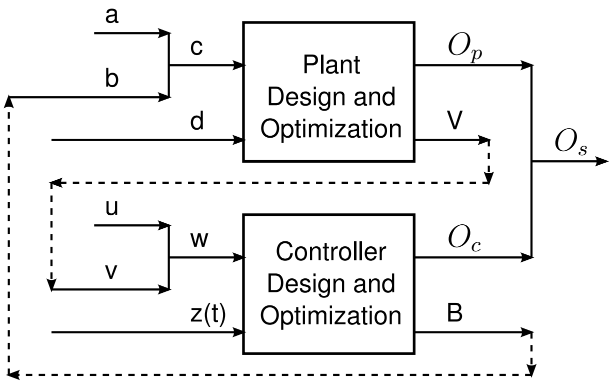


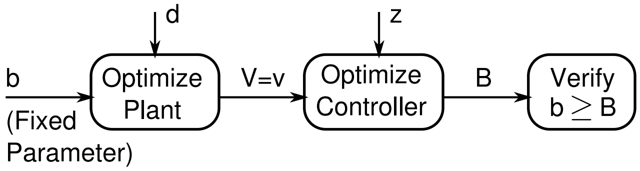
Figure 4.1: Plant/Controller Optimization and Coupling Loop (dashed lines) [17]

In (4.3), the optimization is dependent upon plant and controller parameters b, v in addition to variables $d, z(t)$. This dependence is illustrated in Figure 4.1 and the coupling is shown by dashed lines.

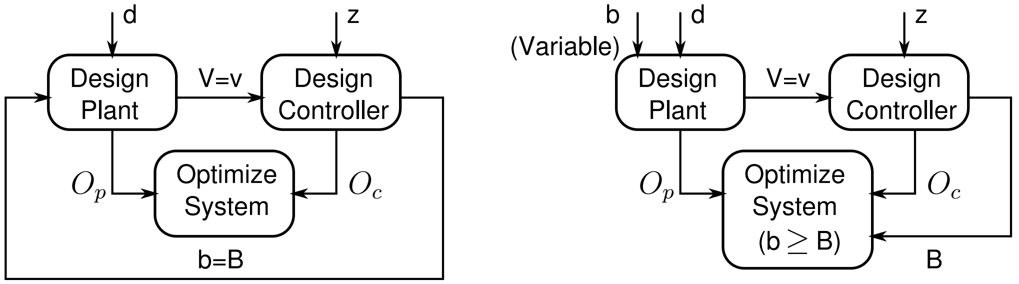
4.1.2 Optimization Strategies

A common approach to optimizing a plant and controller system is the **sequential single pass strategy** where the plant is first optimized followed by optimizing the controller [17]. In this strategy, as shown in Figure 4.2a, the coupling between plant and controller is ignored. A fixed set of plant parameters b is assumed and the plant design variables d are varied to optimize the plant. Parameters required for controller design v are extracted from the plant design by V (such that $v = V$) and controller variables z are determined in the control optimization stage. As b and v are constant with respect to the individual plant and controller optimizations, this approach is equivalent to solving (4.1) followed by (4.2). Though simplistic, this approach requires the verification that controller requirements B have been satisfied by the assumed plant parameters b (such that $b \geq B$) to ensure a feasible system. System level optimality is not guaranteed when B and b differ [17]. A similar argument holds for the converse controller-then-plant sequence.

Deficiencies of the single pass strategy are resolved by the introduction of a system level objective function O_s and by deriving plant parameters b from con-



a) Sequential Single Pass Strategy



b) Ideal Concurrent Strategy

c) All At Once Strategy

Figure 4.2: Plant/Controller Optimization Strategies [17]

troller requirements B such that $b = B$. The result is the **ideal concurrent strategy** shown in Figure 4.2b (a simplified diagram of Figure 4.1). The ideal concurrent strategy seeks to directly solve the plant/controller optimization problem as described by (4.3) under the full coupling constraints $v = V$ and $b = B$. Unfortunately, determination of b and v involves the challenging solution of a set of simultaneous, differential-algebraic equations presented by the coupling constraints. Further, the full coupling constraint equations are often implicitly defined within complex simulations [17] and not readily available for analysis.

The **all at once strategy** shown in Figure 4.2c is a practical approach to system level optimization and is obtained by relaxing the coupling constraints. Either plant or controller parameters b or v, respectively, are allowed to deviate from system requirements as long as minimum requirements are satisfied. That is, the search space is enlarged to include feasible but non-optimal systems by replacing either the constraint $b = B$ with $b \geq B$ or $v = V$ with $v \geq V$ in (4.3). The constraint relaxation simplifies determination of b and v since it removes the requirement of simultaneous solution of plant/controller coupling constraints and allows b to deviate from B (assuming $b \geq B$). Therefore, plant parameters b are treated as plant variables and specified explicitly along with d in the optimization process while controller parameters v are derived under the constraint that $b \geq B$.

Note. The above strategies, and other optimization strategies, namely the sequential iterative strategy, the concurrent decoupled system strategy, and the concurrent bilevel strategy are presented in [17]. The strategies were compared in [17] in an optimal design of an electric DC motor and controller, with the conclusion that the all at once strategy be recommended for solving system optimization problems.

4.2 Problem Formulation

The desire for a traction control system design that minimizes both energy use and drive cycle completion time creates a multi-objective optimization problem whose solution involves the selection of plant and controller design variables. First, the plant and controller are combined to form a complete traction control system. Formulation of the optimization problem then requires the definition of: plant and controller design variables (optimization decision variables), variable constraints, and objective function. The complete optimization problem is summarized in Section 4.2.4 and a solution strategy is proposed in Section 4.3.

4.2.1 Combined Plant and Controller System

The traction control system model to be optimized consists of the vehicle (plant) model summarized in Section 2.4 and the adhesion gradient fuzzy logic traction controller described in Section 3.3. The system model is illustrated in Figure 4.3 where the drive cycle energy use O_E and completion time O_T are plant outputs of particular interest as minimization objectives in the optimization process. The traction controller considers a driver torque input request T_{ref} and the tire adhesion gradient A obtained from the plant and outputs an applied torque T_a on the vehicle tire.

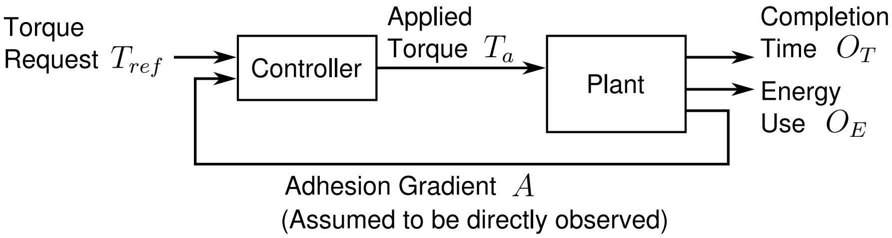


Figure 4.3: Traction Control System Model

4.2.2 Decision Variables and Constraints

A minimal number of decision variables are considered to minimize optimization search space and focus on a simultaneous plant/controller optimization with a mixture of plant and controller variables as well as a mixture of continuous and discrete variables. The chosen variables are defined as:

- Discrete selection of vehicle (plant) tire through variable TIRE:

  $$\text{TIRE} = \begin{cases} 0, & \text{Select Performance Tire 0} \\ 1, & \text{Select Low Rolling Resistance Tire 1} \end{cases} \tag{4.4}$$

  where TIRE is constrained to a binary selection TIRE $\in \{0,1\}$.

  The value of TIRE impacts the plant design by modifying parameters of the tire model through the constraint:

  $$(B_{xo}, D_{xo}, C_r) = \begin{cases} \text{Tire 0 Parameters}, & \text{TIRE} = 0 \\ \text{Tire 1 Parameters}, & \text{TIRE} = 1 \end{cases} \tag{4.5}$$

  where B_{xo}, D_{xo} are, respectively, the stiffness factor and peak value of the tire force generation model and C_r is the tire rolling resistance as discussed in Section 2.3. The numerical tire parameters are listed in Table A.3.

- Continuous controller variable A-PS representing the location of the Positive-Small membership function of the adhesion gradient. The lower and upper limits of A-PS are intuitively defined based on previous discussion in Section 3.3.2:

  $$\text{A-PS} \in [0.2, 0.8]. \tag{4.6}$$

- Discrete selection of fuzzy logic controller rule base conclusions $\mathcal{C}_i$ for $i = 1, \ldots, 6$ as defined in Section 3.3.3. For convenience, the rule base conclusions $\mathcal{C}_i$ are written as a matrix RULE to facilitate interpretation of the rules in tabular format in conjunction with Figure 3.15b:

$$\text{RULE} = \begin{bmatrix} \mathcal{C}_1 & \mathcal{C}_2 \\ \mathcal{C}_3 & \mathcal{C}_4 \\ \mathcal{C}_5 & \mathcal{C}_6 \end{bmatrix}. \tag{4.7}$$

To facilitate numerical optimization of the rule base conclusions, the mapping Ψ_{RULE} is used to assign a numerical value to represent the output membership functions such that:

$$\Psi_{RULE} : \text{S} \rightarrow 1,\ \text{MS} \rightarrow 2,\ \text{MH} \rightarrow 3,\ \text{H} \rightarrow 4. \tag{4.8}$$

The mapping Ψ_{RULE} therefore constrains the selection of rule base conclusions $\mathcal{C}_i$ such that:

$$\mathcal{C}_i \in \{\text{S, MS, MH, H}\} \Leftrightarrow \mathcal{C}_i \in \{1, 2, 3, 4\}. \tag{4.9}$$

However, additional constraints may be placed on $\mathcal{C}_i$ since it is intuitive that a practical rule base design for the adhesion gradient traction controller will tend to prescribe an increase in torque output T_a as input torque request T_{ref} and adhesion gradient A increase. In other words, with reference to Figure 3.15, rule base conclusions tend to increase from a minimum at the top-left of the rule base towards a maximum at the bottom-right of the rule base. This intuition allows the following constraints to be imposed on $\mathcal{C}_i$:

$$\begin{aligned} &\mathcal{C}_1 \leq \mathcal{C}_2,\ \mathcal{C}_3 \leq \mathcal{C}_4,\ \mathcal{C}_5 \leq \mathcal{C}_6, \\ &\mathcal{C}_1 \leq \mathcal{C}_3,\ \mathcal{C}_3 \leq \mathcal{C}_5,\ \mathcal{C}_2 \leq \mathcal{C}_4,\ \mathcal{C}_4 \leq \mathcal{C}_6, \\ &\mathcal{C}_2 \leq 2,\ \mathcal{C}_4 \leq 3. \end{aligned} \tag{4.10}$$

The last two constraints $\mathcal{C}_2 \leq 2, \mathcal{C}_4 \leq 3$ are a direct consequence of considering the prefixed and immediately adjacent (to the right) rule base conclusions of Figure 3.15b associated with Positive-High (PH) inputs of adhesion gradient.

Note. Although the inclusion of inequality constraints (4.10) in the optimization problem enables a reduction in optimization search space, the requirement to satisfy the group of constraints increases the complexity of the solution technique as variables $\mathcal{C}_i$ may no longer be chosen independently. A novel fuzzy logic rule base encoding is presented in Section 5.3 to simultaneously address constraint satisfaction and search space reduction.

Example 4.2.1. The following variable assignments define a traction control system configured with the performance tire (Tire 0), Positive-Small adhesion gradient membership function located at 0.5 (Figure 3.13), and rule base corresponding to Figure 3.15a:

$$\text{TIRE} = 0,\ \text{A-PS} = 0.5,\ \text{RULE} = \begin{bmatrix} 1 & 2 \\ 2 & 3 \\ 2 & 3 \end{bmatrix}. \tag{4.11}$$

4.2.3 Objective Function

The minimization of both drive cycle energy use O_E and completion time O_T presents a multi-objective optimization problem. In general, the optimal solution of a multi-objective optimization problem is not unique. Rather, the solution is represented by a **Pareto set** defined as a set of designs for which no individual objective may be improved without compromising any other objective [45]. Various approaches to multi-objective optimization exist and can be broadly divided into aggregating and non-aggregating approaches [46]. In aggregating approaches, the individual objectives (i.e. O_E and O_T) are combined into a single function. Non-aggregating approaches seek to improve a vector of objective functions.

Among aggregating approaches, the **weighted sum approach** simply takes the weighted sum of the individual objective functions [46]. Further, it is assumed that the weighting coefficients add to 1. Due to its simplicity, the weighted sum approach is assumed in the analysis of plant/controller optimization strategies found in [17] [45], and as well as in this thesis. The system-level objective function O_s of the traction control system is therefore the weighted sum of the drive cycle energy use

O_E and completion time O_T:

$$O_s = w_E\,O_E + (1 - w_E)\,O_T \tag{4.12}$$

where $w_E \in [0, 1]$ is the weighting coefficient on energy use. The choice of w_E controls the relative importance between the two objectives and its impact on optimization results is further considered in Section 6.4.2.

4.2.4 Problem Summary

The simultaneous plant/controller optimization problem considered in this thesis is summarized in (4.13). The weighted sum of energy use O_E and completion time O_T for the drive cycle described in Section 2.1 is minimized by selection of vehicle tire TIRE, the location of the Positive-Small adhesion gradient membership function A-PS, and controller rule base RULE. Due to the inclusion of a controller optimization problem, additional constraints exist on the dynamic state $\dot{x}$ of the system.

$$\begin{aligned}
&\min_{\text{A-PS, TIRE, RULE}} w_E\, O_E + (1 - w_E)\, O_T \\
&\textbf{subject to:} \\
&\qquad (B_{xo}, D_{xo}, C_r) = \begin{cases} \text{Tire 0 Parameters}, & \text{TIRE} = 0 \\ \text{Tire 1 Parameters}, & \text{TIRE} = 1 \end{cases} \\
&\qquad \dot{x} = \begin{pmatrix} \dot{\Omega} \\ \dot{v}_x \\ \dot{s}_x \\ \dot{O}_T \\ \dot{O}_E \end{pmatrix} = f(x, T_a) \quad \begin{array}{r}\text{(Closed loop dynamics} \\ \text{See (2.29) to (2.40))}\end{array} \\
&\qquad x_o = \left(\frac{v_{xo}}{r}, v_{xo}, 0, 0, 0\right)^T, \; v_{xo} > 0 \quad \begin{array}{r}\text{(Initial} \\ \text{Conditions)}\end{array} \qquad\qquad (4.13) \\
&\qquad \text{RULE} = \begin{bmatrix} \mathcal{C}_1 & \mathcal{C}_2 \\ \mathcal{C}_3 & \mathcal{C}_4 \\ \mathcal{C}_5 & \mathcal{C}_6 \end{bmatrix} \\
&\qquad \mathcal{C}_1 \leq \mathcal{C}_2,\; \mathcal{C}_3 \leq \mathcal{C}_4,\; \mathcal{C}_5 \leq \mathcal{C}_6, \\
&\qquad \mathcal{C}_1 \leq \mathcal{C}_3,\; \mathcal{C}_3 \leq \mathcal{C}_5,\; \mathcal{C}_2 \leq \mathcal{C}_4,\; \mathcal{C}_4 \leq \mathcal{C}_6, \\
&\qquad \mathcal{C}_2 \leq 2,\; \mathcal{C}_4 \leq 3 \\
&\qquad \text{TIRE} \in \{0, 1\} \\
&\qquad \mathcal{C}_i \in \{1, 2, 3, 4\}, \quad i = 1, \ldots, 6 \\
&\qquad \text{A-PS} \in [0.2, 0.8]
\end{aligned}$$

4.3 Solution Strategy

A solution strategy for the traction control optimization problem summarized in Section 4.2.4 is proposed and consists of the application of the all at once optimization variable selection strategy within a simulation-based optimization loop. This section further considers the plant/controller coupling and the application of the all at once strategy with respect to the traction control system under study. The requirement for a simulation-based optimization is examined and the resulting optimization loop is described.

4.3.1 All At Once Variable Selection

As previously discussed in Section 4.1.2, the all at once optimization variable selection strategy is a practical approach to simultaneous plant/controller optimization and is therefore applied to the traction control optimization problem.

With respect to the plant/controller coupling loop shown in Figure 4.1, the coupling within the traction control optimization problem can be described as follows:

- Tire selection represents the plant design variable $d = \text{TIRE}$.
- The applied torque is the control signal to be optimized $z(t) = T_a$. From a design perspective, T_a is a function of the selection of controller design variables A-PS and RULE: $z = T_a(\text{A-PS}, \text{RULE})$.
- Intuitively, control action is a function of tire selection since the tire adhesion gradient characteristic A is an input to the traction controller. Thus tire model parameters (B_{xo}, D_{xo}, C_r) impact control design and may be considered as control design parameters $v = (B_{xo}, D_{xo}, C_r)$ that are affected by plant design variable TIRE.
- Coupling between plant and controller is described by function V that determines controller parameters v depending on plant design variable TIRE. Plant/controller coupling is present in the optimization problem due to inclusion of constraint (4.5) which may be rewritten to explicitly show the

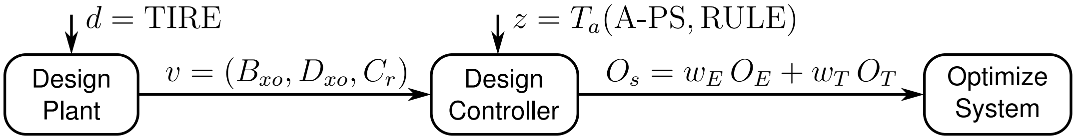


Figure 4.4: Traction Control Optimization Variables and Objective Function

dependence of control design parameters on plant design:

$$v = (B_{xo}, D_{xo}, C_r) = V(\text{TIRE}) = \begin{cases} \text{Tire 0 Parameters}, & \text{TIRE} = 0 \\ \text{Tire 1 Parameters}, & \text{TIRE} = 1. \end{cases} \tag{4.14}$$

The traction control optimization problem under consideration does not include plant parameters b that are affected by controller design. Hence the simultaneous plant/controller optimization considered in this thesis is simplified since a full coupling loop between plant and controller design does not exist. The complete (with full coupling loop) all at once strategy shown in Figure 4.2c therefore simplifies to a strategy incorporating the partial plant/controller coupling shown in Figure 4.4. Although the resulting strategy in Figure 4.4 shows some graphical resemblance to the sequential single pass strategy shown in Figure 4.2a, the two differ since the former retains the simultaneous selection of plant and controller design variables with the goal of optimizing a system objective O_s while the latter seeks to optimize plant and controller individually.

Note. A possible extension to the traction control optimization problem is to include the optimal design of the electric motor driving the vehicle. As described in [17], armature resistance and inductance vary depending on motor power specification and impact the maximum power used by a motor controller design. Such an extension would therefore require consideration of the full plant/controller coupling loop to ensure that the minimum motor power specification b is greater than or equal to the power requirements of the control strategy B.

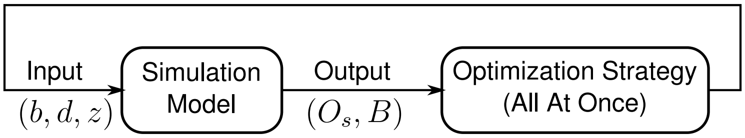


Figure 4.5: Simulation Optimization Model [47]

4.3.2 Simulation Optimization

Although the all at once strategy prescribes a simultaneous selection of plant and controller variables, it does not suggest a mechanism for search space exploration. The diagrams in Figure 4.2 and Figure 4.4 do not show how optimization variables should be altered during the optimization process to converge towards an optimal solution. Ideally, for a given optimization objective, the optimization variables would be determined through closed form solution. However, system complexity often requires a simulation model to evaluate the system-level objective function O_s as well as plant requirements B. An iterative approach is therefore required to converge towards an optimal solution. Further, high computational cost associated with system simulation may prevent the exploration of the entire search space.

A **simulation optimization** is therefore desirable where the best (but perhaps non-optimal) input variable values are obtained in a limited iterative exploration of the search space [47]. A simulation optimization model applied to the all at once strategy is shown in Figure 4.5 and consists of a system simulation model and an optimization strategy within an optimization loop. The optimization strategy, in this case the all at once strategy, determines the values of the variables d, z, and b for the next simulation based on the previous simulation results O_s, and B. Among methods of implementing the optimization strategy, heuristic direct search methods (requiring only function values) such as genetic algorithms are frequently used for simulation optimization [47].

In the optimization of the traction control system, a simulation model is required to determine system dynamics due to nonlinearities in the tire model, road surface transition, and fuzzy logic controller. The nonlinearities preclude closed form evaluation of objectives O_E and O_T and therefore the all at once optimization

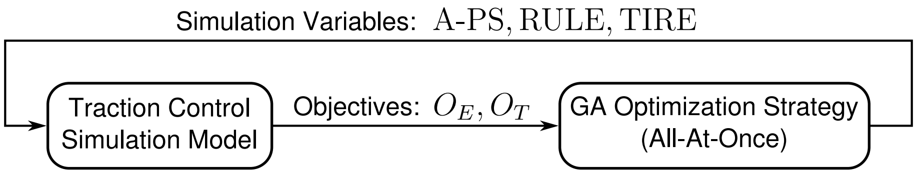


Figure 4.6: Optimization Loop Using Traction Control Simulation Model

strategy is applied within a simulation optimization.

The proposed solution strategy of the traction control optimization problem is summarized in Figure 4.6 where the traction control simulation model is as previously described in Section 4.2.1 and the all at once variable selction strategy is implemented using a genetic algorithm to be described in Chapter 5. The figure shows that the genetic algorithm affects changes in the optimization variables A-PS, RULE, and TIRE according to an evaluation of objectives O_E and O_T obtained from the simulation model.

Chapter 5

Genetic Algorithm Design

Design details of the genetic algorithm used in the simultaneous plant/controller optimization of the traction control system are described. First, background on genetic algorithms is presented, including mixed encoding of continuous and discrete variables and application in fuzzy logic control optimization. Application-specific design details regarding optimization variable encoding and definition of associated genetic operations are then provided to define a mixed encoding, multi-chromosomal genetic algorithm suitable for optimizing the traction control problem summarized in Section 4.2.4. A novel fuzzy logic rule base encoding is presented to address challenges in rule base constraint satisfaction associated with search space reduction.

5.1 Genetic Algorithms

Genetic Algorithms (GA) are heuristic optimization methods that were inspired by Charles Darwin's principle of natural selection [48]. This principle stems from the observation that, as a population of species evolves, individuals with beneficial traits are more likely to survive to reproduce, and hence propagate their traits to their offspring. GA numerically emulate this process to evolve good solutions from a population of candidate solutions. As a numerical optimizer, with generally problem-independent mathematical formulation, a GA possesses intrinsic flexibility and can equally be applied to problems exhibiting a combination of nonlinear,

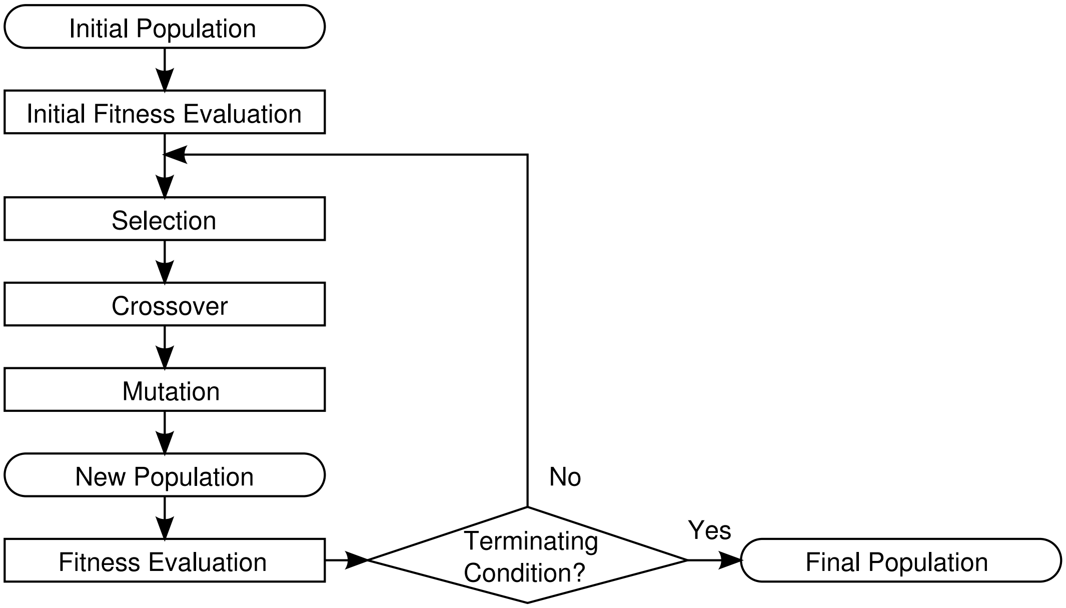


Figure 5.1: Genetic Algorithm Structure [48]

constrained, discontinuous, and multi-modal characteristics [48].

The basic operation of a GA (as presented in Figure 5.1) entails refining a population of candidate solutions through repeated application of a series of genetic operations consisting of solution selection, crossover, mutation, and fitness evaluation. In the following subsections, the classical operation of a GA on binary strings is discussed, followed by extensions allowing alternatives to binary encoding, mixed encoding with multi-chromosomal representation, as well as constrained optimization. Parallel implementation strategies that allow high-speed execution of a GA are also reviewed. This section concludes with a discussion of the application of genetic algorithms to fuzzy logic controller optimization in the field of genetic-fuzzy control.

5.1.1 Unconstrained Binary GA

The following sub-sections detail the unconstrained optimization process of a binary GA with reference to Figure 5.1.

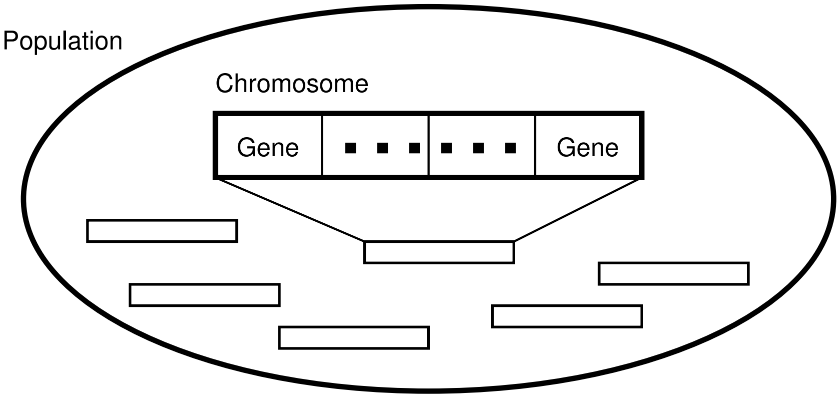


Figure 5.2: Genes, Chromosomes, and Population

Representation

Prior to the application of a GA, it is assumed that the solution to a given problem can be represented by a set of variables. Following biological inspiration, each variable is referred to as a **gene** and the set of genes forms a **chromosome**. In a binary GA, a single chromosome corresponds to a **candidate solution** to the problem. A set of candidate solutions forms a **population**. The relationship between gene, chromosome, and population is shown in Figure 5.2 where a single chromosome has been enlarged to show the gene structure. For convenience, a candidate solution may be referred to as a **member** of a population. A particular population may be referred to as a **generation** to reflect the fact that a new population is derived from a previous population after every iteration of the GA.

In general, the representation and encoding of variables is problem-dependent. In a binary GA, it is assumed that all variables are encoded as a string of binary numbers. For a problem with n variables encoded as genes x_i, $i = 1 \ldots n$, a chromosome X is a binary string of length $L(X) = \sum_{i=1}^{n} L(x_i)$, where function L is the number of bits required to represent its argument. The binary representation allows the crossover and mutation genetic operators to derive new candidate solutions, or **offspring**, from existing candidate solutions, or **parents**, in a problem-independent manner.

Initial Population

An initial population is instantiated at the beginning of a GA. The initial population consists of randomly generated candidate solutions [48]. In the case of a binary GA, a candidate solution is a random binary string of length $L(X)$. The population's size is typically fixed during the application of GA. A larger population size increases the number and diversity of solutions explored but incurs more processing time per iteration of the GA.

Objective Function and Fitness Evaluation

Every member in the initial population, as well as in subsequent populations, is subject to a fitness evaluation resulting in a **fitness value** F used to evaluate the "goodness" or optimality of the candidate solution for the problem [48]. A chromosome X with a higher fitness value than another chromosome Y means that chromosome X represents a solution closer to the optimum than chromosome Y.

Fitness evaluation begins with the use of an **objective function** to evaluate the optimality of a chromosome and produce an objective value O. The objective function is problem-dependent and the variables encoded in the chromosome may be evaluated by various means ranging from a simple algebraic expression to a large numerical simulation. In the case of a simulation optimization (Section 4.3.2), and in this thesis, a simulation model is assumed to evaluate the encoded variables.

Being problem-dependent, the objective value may assume a wide range of numbers requiring problem-specific knowledge to interpret. For example, smaller numbers represent desirable solutions in minimization problems whereas the converse is true in maximization problems. Therefore, a mapping $\Psi : O \rightarrow F$ is introduced to convert the problem-dependent objective value O to a problem-independent fitness value F [48].

A common mapping is the **ranking scheme** used to assign fitness according to the position ranks of the sorted objective values of a given population [48]. Population members are sorted according to their objective values and the fitness value is taken as the rank of each member. Higher ranks correspond to higher fitness and the ascending or descending sort order is chosen depending on whether the optimization is maximizing or minimizing, respectively. The ranking scheme is affected only by the relative difference in ranking and not by difference in magnitude of objective values. This mapping characteristic is desirable as discussed in the following section on the selection process.

Selection

The population is evolved by selectively applying genetic operations that produce a new generation from members of the current population. Following biological inspiration, new members are preferably derived, or reproduced, from fit members with the goal of improving the average population fitness towards an optimal solution.

Members are commonly chosen using **roulette wheel selection** [48] as follows:

1. Sum the fitness values of all population members: $F_{sum} = \sum F$.
2. Generate a random number r between 0 and F_{sum}.
3. Return the first population member whose fitness value, when added to the fitness value of all proceeding population members, is greater than or equal to r.

This is equivalent to assigning each population member a segment of a roulette wheel whose size is proportional to the member's fitness value. Members are selected for reproduction by uniformly selecting a random point along the circumference of the roulette wheel and choosing the indicated member.

Example 5.1.1. Assuming a maximization problem and objective values O of 5, 10, 20, 60, 200 for a population of 5 members $X_i,\ i = 1 \ldots 5$, respectively, the probability of selecting a given member is shown in Figure 5.3 where Ψ is: a) a direct mapping $F = O$ ($F_{sum} = 5 + 10 + 20 + 60 + 200$), and b) according to the ranking scheme ($F_{sum} = 1+2+3+4+5$). The figure shows that the ranking scheme is insensitive to differences in magnitude of the objective function and ensures that one very good solution X_5 will not disproportionally dominate the selection process by occupying a large portion of the roulette wheel [48].

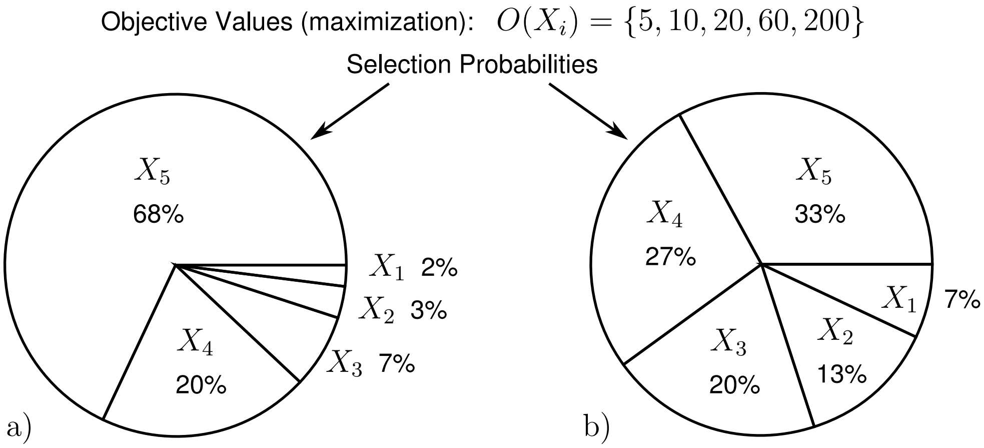


Figure 5.3: Roulette Wheel Selection a) Direct Mapping, b) Ranking Scheme [48]

As roulette wheel selection relies on strictly positive fitness values with higher fitness values corresponding to fitter individuals, it is often paired with an application of the ranking scheme during fitness evaluation. The combination allows the consideration of minimization problems where a smaller objective value corresponds to higher fitness.

Example 5.1.2. Assume a minimization problem and objective values O of 5, 20, 10, 200, 60 for a population of 5 members $X_i, i = 1 \dots 5$, respectively. As shown in Table 5.1, application of a direct mapping $F = O$ leads to an invalid fitness assignment since the member with the highest objective value is incorrectly assigned the highest fitness (highest selection probability). In contrast, a valid mapping for the minimization problem results after sorting objective values in descending order and applying the ranking scheme such that $F = rank$ and the lowest objective value is assigned the highest fitness.

Crossover

Members of a new population are derived from existing members using a crossover process followed by a mutation process. Two parent members A, B are selected from the current generation, which may include elite members, and a **crossover**

Table 5.1: Example Roulette Wheel Selection on Minimization Problem

			Fitness Value F	
Rank	Member	Objective O	Direct Map	Ranking Scheme
1	X_4	200	200	1
2	X_5	60	60	2
3	X_2	20	20	3
4	X_3	10	10	4
5	X_1	5	5	5

operation f_{cross} is performed resulting in an offspring C containing genes from both parents:

$$C = f_{cross}(A, B). \tag{5.1}$$

A common crossover operation for binary strings is the **one-point crossover** shown in Figure 5.4 [48]. For parent chromosomes $A = \{a_1, a_2, \ldots, a_L\}$ and $B = \{b_1, b_2, \ldots, b_L\}$ with $L = L(A) = L(B)$ bits each, a random crossover point j is chosen such that the resulting offspring is $C = \{a_1, \ldots, a_j, b_{j+1}, \ldots, b_L\}$. Although two offspring are generated from each crossover (the other being $C' = \{b_1, \ldots, b_j, a_{j+1}, \ldots, a_L\}$), only one is kept such that m crossover operations are needed to produce a population of size m.

Note. Either offspring C or C' is discarded depending on the implementation of the GA. Either offspring may be discarded since the two parents to the crossover operation are randomly chosen in the selection process.

Mutation

A **mutation** operation f_{mutate} is applied to each offspring C after crossover to produce a randomly perturbed offspring D:

$$D = f_{mutate}(C) \tag{5.2}$$

Mutation randomly selects and alters a gene with small probability (p_m), typically less than 10% [48]. **Bit mutation** f_{mutate} is applied to binary strings by randomly

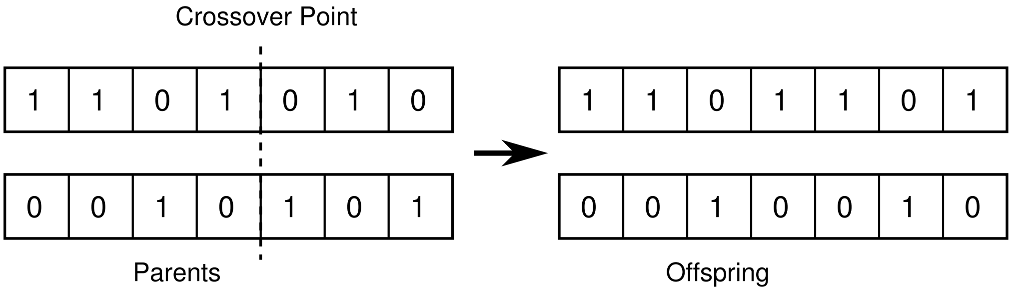


Figure 5.4: Example of One-Point Crossover [48]

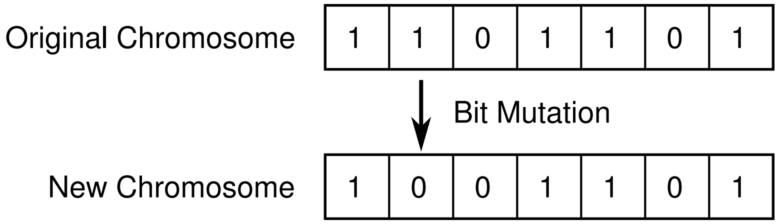


Figure 5.5: Example of Bit Mutation [48]

changing a single bit (Figure 5.5).

Terminating Condition

The cycle of genetic operations is repeated until a terminating condition is reached. Common choices of termination criteria are a specified number of generations, predefined value of fitness, rate of change of fitness, or a combination of these factors [48]. Upon termination of a successful GA optimization, the average population fitness should have improved with fit candidate solutions approximating an optimal solution.

Hypothesis Of Operation

The **building block hypothesis** (BBH) has been proposed to describe the optimizing behavior of a GA [48]. The hypothesis assumes the existence of short sequences of genetic information (or short bit strings in a binary GA) called **build-**

ing blocks that, when present in the encoding of a candidate solution, increases the fitness of the associated population member. Crossover, as it operates on long sequences of genetic material, tends to preserve short building blocks while shuffling them between population members. Combined with selection, crossover creates population members consisting of sequences of short building blocks. Crossover is likely to destroy long building blocks, which negates its benefits. Mutation tends to generate new building blocks as any portion of the genetic sequence is equally affected by its action. Hence, BBH offers guidance for the problem-dependent representation of optimization variables with encodings that lead to short building blocks for a successful application of a GA. BBH also motivates modification of the binary genetic algorithm to accommodate alternative representations as described next in Section 5.1.2.

5.1.2 Mixed Encodings and Multi-Chromosomal GA

The binary GA of Section 5.1.1 can be classified as a **homogeneous** GA since all optimization variables are encoded using the same data type, namely binary. Further, the binary GA is considered a **single-chromosome** GA since a candidate solution is represented on one chromosome. This section reviews the use of other data types for encoding, followed by the extension to a mixed-encoding, or **heterogeneous**, GA as implemented in multi-chromosomal GA.

Alternative Encodings

The BBH provides motivation to adapt the GA to other data types such as integer, and floating-point (real-valued) encodings [48]. A motivational example is presented in Figure 5.6 that illustrates the advantage gained by using an alternative encoding on the simple optimization problem:

$$\min\,(x_1^2 + x_2^2) \quad \textbf{subject to: } x_1, x_2 \in \{0, 1, \ldots, 15\}. \tag{5.3}$$

The two parameters x_1, x_2 of this integer optimization problem, with known minimum $x_1 = x_2 = 0$, can either be encoded as two 4-bit strings or as two integers.

$$\min\,(x_1^2 + x_2^2) \quad \textbf{subject to: } x_1, x_2 \in \{0, 1, \ldots, 15\}$$

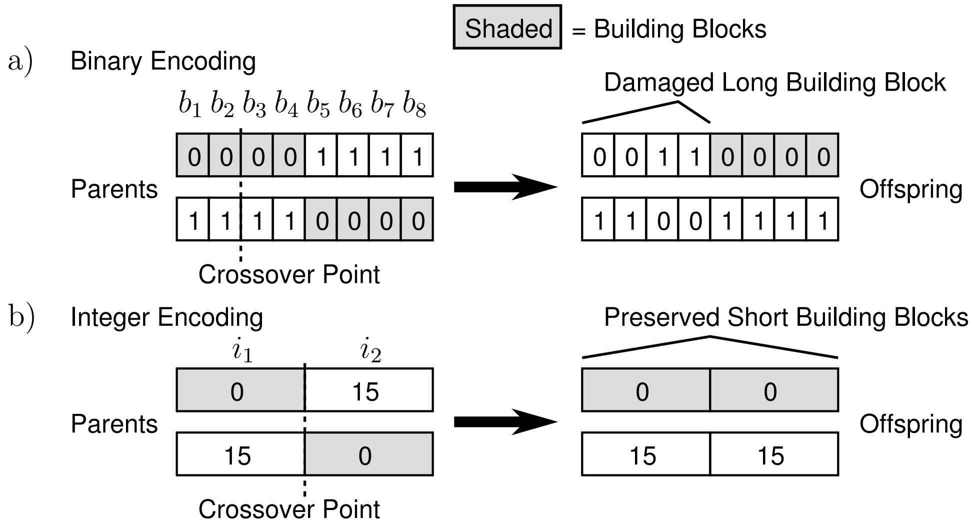


Figure 5.6: Effect of Crossover in a) Binary GA and b) Integer GA

The binary encoding results in longer building blocks corresponding to bit strings $b_1b_2b_3b_4 = 0000$ and $b_5b_6b_7b_8 = 0000$ for $x_1 = 0$ and $x_2 = 0$, respectively. In contrast, integer encoding directly represents the building blocks $i_1 = 1$ and $i_2 = 0$. For the encoding of two candidate solutions $(15, 0)$ and $(0, 15)$ shown in Figure 5.6, BBH suggests that a single-point crossover operation in a binary GA is likely to destroy either building blocks. In contrast, a single-point crossover operation in an integer GA may lead directly to an optimal solution. In this case, problem-specific knowledge has allowed a problem decomposition resulting in an integer encoding leading to smaller building blocks which are not destroyed in crossover. A similar case applies when comparing a binary GA to a real-valued GA in the continuous version of (5.3).

In addition to chromosome encoding, it is also required to define appropriate crossover and mutation operations to manipulate the underlying data type. For example, the real-valued equivalent to binary crossover is commonly the **arithmetical crossover** where a new offspring C is generated by taking the average of the real-valued genes of parents A and B such that [49]:

$$C = \left\{ \frac{a_1 + b_1}{2}, \frac{a_2 + b_2}{2}, \dots, \frac{a_n + b_n}{2} \right\}. \tag{5.4}$$

Further, **Gaussian mutation** may be applied to create a real-valued chromosome D by the addition of a random distribution to a random gene c_i of chromosome C [48]:

$$D = \{c_1, \dots, c_i + \psi(\mu, \sigma), \dots, c_n\} \tag{5.5}$$

where c_i = randomly selected real-valued gene
ψ = Gaussian (normal) random function with mean μ and standard deviation σ.

Mixed Encodings

In the solution of a large problem, a divide-and-conquer approach relies on the solution of sub-problems leading to the solution of the entire problem. It is natural to solve each sub-problem using the most efficient technique. For example, the optimization problem in (5.3) may be viewed as a linear combination of two sub-problems: $\min x_1^2$ and $\min x_2^2$. Assuming the use of a GA, it was shown that an integer encoding was preferable for both variables x_1 and x_2 (due to the identical nature of the sub-problems). In general, a problem may be decomposed into sub-problems that are best represented by different encodings. Due to coupling between sub-problems, as demonstrated in Section 4.1, a simultaneous optimization is required on all sub-problems to find a global optimum [50].

One methodology for breaking down a mixed-encoding problem is to divide optimization variables into homogeneous groups consisting of the same type [50]. Each group of variables is considered as a separate chromosome X_i in a **multi-chromosomal** representation $\widehat{X} = \{X_1, X_2, \dots, X_m\}$ where m is the number of groups and $i \in \{1 \dots m\}$. In a multi-chromosomal GA, a single candidate solution, or population member, $\widehat{X}$ is therefore represented by a collection of m homogeneous chromosomes X_i as shown in Figure 5.7.

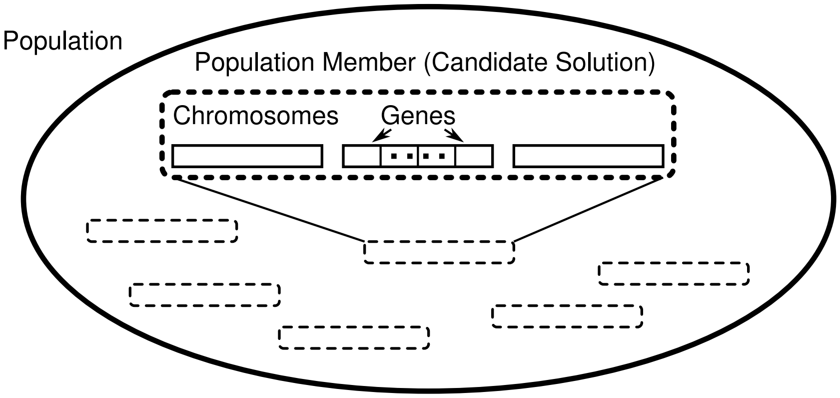


Figure 5.7: Candidate Solution Representation in Multi-Chromosomal GA

The basic operation of a multi-chromosomal GA remains as shown in Figure 5.1. Crossover and mutation operations defined for existing homogeneous GAs may be reused in the **matched chromosome mating** process of a multi-chromosomal GA [50]. Population members are considered as ordered sets of chromosomes with chromosome A_i of $\widehat{A}$ encoding the same variables (with similar encoding type) as chromosome B_i of $\widehat{B}$. Hence, the creation of $\widehat{C}$ from $\widehat{A}$ and $\widehat{B}$ can be defined in terms of (5.1) as:

$$C_i = f_{cross}(A_i, B_i) \tag{5.6}$$

where f_{cross} is an appropriate crossover operator for the chromosome type of A_i and B_i. Similarly, the mutation of a member $\widehat{C}$ to a perturbed $\widehat{D}$ can be defined in terms of (5.2) as:

$$D_i = f_{mutate}(C_i) \tag{5.7}$$

where f_{mutate} is an appropriate mutation operator for the chromosome type of C_i.

An extension to the crossover process is obtained by considering the individual chromosomes of a multi-chromosomal GA as big building blocks, or equivalently, as solutions to sub-problems. Thus, **chromosome shuffling** has been proposed in order to directly exchange, and preserve, solution information at the chromosome level [51]. Chromosome shuffling creates a new candidate solution C by selecting chromosomes A_i or B_i from either parent $\widehat{A}$ and $\widehat{B}$, respectively, with a function

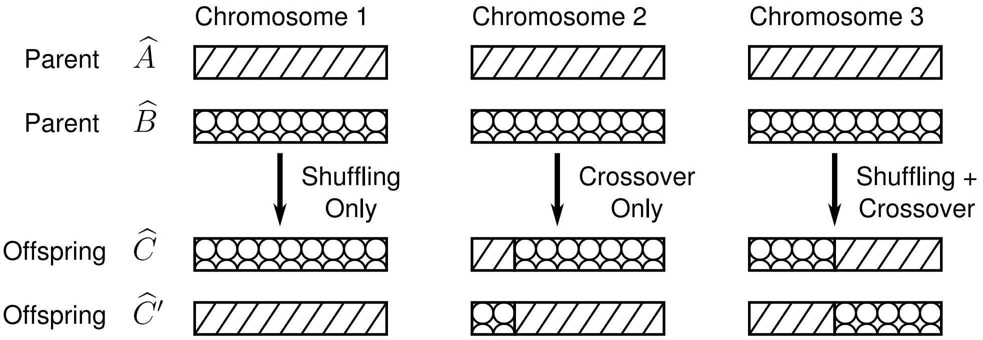


Figure 5.8: Chromosome Shuffling and Crossover [51]

$f_{shuffle}$:

$$C_i = f_{shuffle}(A_i, B_i). \tag{5.8}$$

An example definition of $f_{shuffle}$ is to randomly assign A_i or B_i to C_i:

$$C_i = \begin{cases} A_i, & rand(\ ) < 0.5 \\ B_i, & \text{otherwise.} \end{cases} \tag{5.9}$$

where $rand(\ )$ returns a uniform random number between 0 and 1. Since chromosome shuffling and crossover both operate on two parents, the two may be performed simultaneously in a single step as shown in Figure 5.8. Two offspring $\widehat{C}$ and $\widehat{C}'$ are generated by chromosome shuffling and crossover, but, as in the case of one-point crossover, only one offspring $\widehat{C}$ is kept.

5.1.3 Constraint Handling

A GA relies upon the generation of candidate solutions by combining existing solutions through crossover and randomly perturbing existing solutions through mutation. The inclusion of constraints in an optimization problem may lead to the generation of infeasible solutions that require additional consideration as discussed in this section.

The Feasible Search Space

Consider a general optimization problem (to be solved by GA) of the form:

$$\text{optimize } O(X), \tag{5.10}$$

where O is an objective function and X (or $\widehat{X}$ for multi-chromosomal GA) is a candidate solution encoding n optimization variables $x_i, i = 1 \ldots n$. As currently stated, (5.10) is an unconstrained optimization problem where the domain of the variables defines the **search space** $\mathcal{S}$ of the problem. For example, if X is a homogeneous encoding of n real-valued variables, then the search space of (5.10) is the n-dimensional space of real numbers $\mathcal{S} = \mathbb{R}^n$.

Definition of genetic operations, thus far, has allowed the GA to generate candidate solutions within the entire search space. This is desired for unconstrained optimization where it is assumed that all $X \in \mathcal{S}$ are feasible candidate solutions. The addition of inequality and equality constraints, for example as present in (4.1) to (4.3), may restrict the set of feasible solutions, or **feasible search space** $\mathcal{F}$, to a subset of the search space such that $X \in \mathcal{F} \subset \mathcal{S}$ [52]. In Figure 5.9, candidate solutions A and B are within the feasible search space with an optimum solution of X. The figure illustrates that constraints may produce a disjoint, nonconvex feasible search space. It is then possible that genetic operations may create infeasible candidate solutions C within the **infeasible search space** $\mathcal{U}$ which is defined as the complement of $\mathcal{F}$. The remaining discussion is focused on proposed methods [52] for dealing with infeasible candidate solutions generated by constraints. The methods may be broadly classified as either rejecting, avoiding, or incorporating infeasible solutions into the GA.

Rejection

Perhaps the simplest approach is to discard infeasible candidate solutions from the population and reapply the selection process to generate an alternative candidate. This method works but is disadvantageous when the feasible search space $\mathcal{F}$ is small compared to the search space $\mathcal{S}$. In addition to the computational overhead of generating alternatives to an infeasible solution, the infeasible candidate solutions may

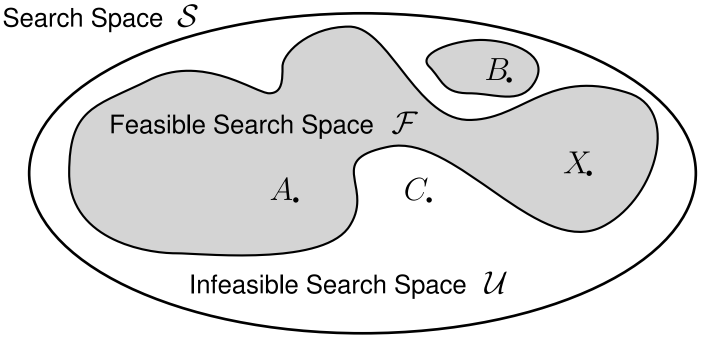


Figure 5.9: Search Space with Feasible and Infeasible Parts [52]

contain building block information that should be not be discarded. For example, in Figure 5.9, infeasible candidate solution C may allow the feasible candidate A to "cross" an infeasible region to reach the solution optimum X [52].

Searching Domain

The method of searching domain requires no modification to an unconstrained GA as it involves restating the optimization itself such that the feasible search space becomes equivalent to the search space $\mathcal{F} = \mathcal{S}$ [48]. Careful selection, or novel introduction, of variable encoding techniques may remove constraints required in alternative representations.

Example 5.1.3. Suppose a variable x is constrained to the natural numbers $0, 1, \ldots, 7$. An integer encoding with the constraint that $0 \leq x \leq 7$ satisfies optimization requirements. However, the constraint can be eliminated from the solution process if the variable is alternatively encoded as a 3-bit binary string whose search space is equal to the feasible search space (binary numbers 000 to 111 map to integers 0 to 7).

Repair Mechanism

Repair algorithms are implemented when it is relatively easy to change an infeasible candidate solution C into a feasible candidate solution A [52]. The repaired candidate solution A may replace the original C in the population, or the repair may be done only for fitness evaluation and leave C within the population. The repair mechanism is typically inserted in-between the mutation and fitness evaluation steps of Figure 5.1. Since fitness evaluation is typically the most computationally intensive process of a GA [48], the computational overhead associated with repair is acceptable as it allows the GA to focus on exploring the feasible search space.

Example 5.1.4. Assume a variable is constrained to the natural numbers $0, 1, \ldots, 5$ and represented by a 3-bit binary encoding. A 3-bit binary representation is necessary to include the feasible search space $\mathcal{F} = \{000, 001, 010, 011, 100, 101\}$. However, candidate solutions "110", "111" violate the constraint and are therefore infeasible. A repair mechanism may be implemented to map the infeasible candidates into the feasible search space by taking their complements such that $110 \rightarrow 001$ and $111 \rightarrow 000$.

Specialized Genetic Operations

Under the assumption that the initial population is within the feasible search space, the crossover and mutation operations may be designed as a **closed set** of operations to maintain the feasibility of solutions. Such specialized genetic operations have been demonstrated for convex feasible search space however the method is currently unable to deal with nonconvex search spaces (and hence nonlinear constraints in general) [52].

Example 5.1.5. Assuming candidate solutions A and B, with real-valued encoding and within a convex feasible search space, arithmetical crossover (5.4) $\frac{A+B}{2}$ always produces feasible offspring C. A closed set definition of mutation on C is to randomly choose a point on a line with random slope passing through C. The choice of the point is restricted to within the segment defined by two points of intersection of the line with the boundary of the convex search space [52].

Penalty Scheme

Various penalty schemes have been proposed to allow the direct inclusion and evaluation of infeasible candidate solutions within a population [52]. Consideration in penalty scheme design involve methods to compare two infeasible members as well as an infeasible member to a feasible member. For a problem with m constraints, a corresponding number of **constraint violation measures** $f_j, j = 1 \dots m$ are introduced to allow such comparisons. Constraint violation measures are associated with penalties that decrease the fitness of a candidate solution depending on a combination of how many or by how much constraints are violated.

The penalty scheme does not guarantee that the final solution found will be within the feasible search space, although it is the intent of penalties to minimize constraint violation through the selection process. This method may permit a small violation of a constraint to allow a large gain towards an optimal solution. However, the penalty scheme can incur much computational overhead in evaluating infeasible candidates when the feasible search space is much smaller than the search space [48].

Example 5.1.6. Assume a minimization problem with m constraints and objective function $O(X)$ that evaluates both feasible and infeasible candidate solutions X. A simple penalty scheme is obtained through a weighted combination of the objective function and the number of violated constraints to produce a modified objective function O' [48]:

$$O'(X) = O(X) + \delta \sum_{i=1}^{m} f_j(X)$$

where $m =$ number of constraints

$\delta =$ penalty coefficient (positive for minimization problems)

$$f_j = \begin{cases} 1, & \text{constraint } m \text{ violated by X} \\ 0, & \text{otherwise.} \end{cases}$$

5.1.4 Parallel Architecture

The structure of a GA (Figure 5.1) allows various methods of parallelization [48]. In general, a parallel implementation of a GA utilizes multiple computational nodes connected by a communication network to perform the time consuming, yet independent, crossover, mutation, and fitness evaluation operations.

The **Global GA** implementation is based on a farmer-worker architecture (Figure 5.10). The farmer node selects pairs of population members to send to worker nodes for crossover, mutation and fitness evaluation, and collects the results as well as the new offspring. This information allows the farmer to create the next generation and repeat the selection process. A disadvantage of the Global GA is that workers are idle while the farmer is doing the selection process which may involve sorting and ranking the results. A global GA implementation is implemented in this thesis.

Other parallelization schemes require some modification to the basic operation of the GA. In **migration GA**, also called coarse grained parallel GA, the population is divided into sub-populations. Each sub-population is then evolved separately using a conventional GA on a dedicated node. A **migration** operation is introduced to allow fit candidate solutions to move between sub-populations and ultimately spread within the whole population. Migration GA apply to small multi-processor clusters (i.e. a small number of high performance nodes) since a small number of sub-populations are evolved on dedicated nodes [48].

In contrast, on large clusters with many nodes, a **diffusion GA**, also called a fine grained parallel GA, assigns each population member to a single dedicated node [48]. The node performs a modified selection process in addition to regular crossover, mutation, and fitness evaluation for a single member. Unlike a global GA, selection is distributed within the cluster. Each node, however, is limited to selecting between a restricted number of immediately adjacent nodes (or members) on the cluster's communication network. The limitation on adjacent nodes allows the practical implementation of communication networks on massively parallel computers. For example, nodes of a diffusion GA may be connected by a high-speed, rectangular grid communication topology that limits adjacent nodes to four.

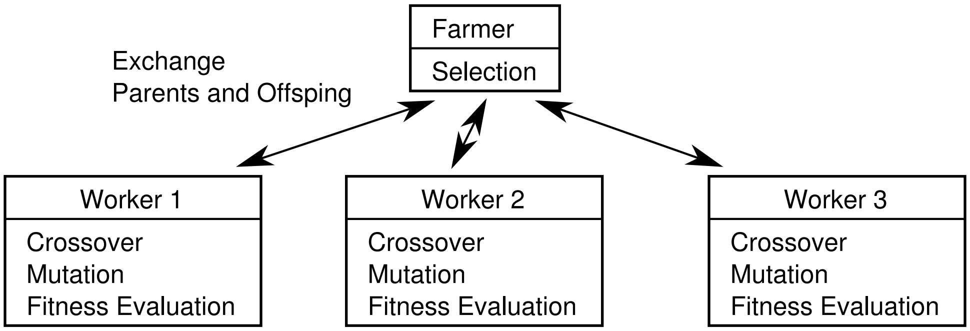


Figure 5.10: Global GA [48]

5.1.5 Genetic-Fuzzy Control

Previous discussion of fuzzy logic control in Section 3.2 and the current discussion of GA have established their individual flexibility and broad-applicability in control and optimization problems, respectively. A fuzzy logic controller presents a general control structure with many parameters required in the definition of membership functions and rule base. Fuzzy logic controller parameters are typically chosen by intuition derived from real-world experience and therefore not optimal. As briefly mentioned in Section 3.2.7, the optimization of fuzzy logic controllers is difficult due to the nonlinear control action of fuzzy logic, the large search spaces involved (as a result of numerous parameters), and the mix of continuous and discrete parameters. Researchers have therefore relied upon genetic algorithms as a highly problem-independent, general framework for the nonlinear optimization of fuzzy logic controllers termed **genetic-fuzzy control** [6].

Automotive applications of genetic-fuzzy control have focused on the use of a single-chromosome binary GA (Section 5.1.1) to optimize the shape of membership functions while assuming a fixed, potentially non-optimal rule base [6][39]. In both works, continuous triangular membership function parameters, including prototype location and span, are discretized and encoded as binary strings. The controller parameters are treated as optimization variables in a simulation optimization involving the system under consideration (being a hybrid electric vehicle powertrain [6], and an active vehicle suspension [39]). Both works conclude that genetic-fuzzy control is effective at determining the shape of membership functions.

The flexibility of a GA design permits both membership functions and rule base to be optimized simultaneously [48]. In this approach, traditional control objectives are supplemented by objectives to minimize the number of membership functions and rules in the optimized controller.

Continuous variables are required to define the shape of membership functions in addition to the following discrete variables for the definition of the rule base:

- Number of membership functions associated with the universe of each input.
- Number of membership functions associated with the universe of each output.
- Conclusion associated with the premise in each rule.

These variables allow the optimization of a rule base to be decomposed into the optimization of the conclusion of existing rules as well as the optimization of the number of rules as clarified in Example 5.1.7.

Example 5.1.7. Assume a double-input, single-output fuzzy logic controller with a rule base represented by a FAM table. Let n_{x1}, n_{x2}, n_y be the number of membership functions associated with the universe of inputs x_1, x_2 and output y, respectively. For $n_{x1} = 3$, input x_1 may be fuzzified using membership functions representing various degree of "cold", "warm", and "hot". For $n_{x2} = 2$, input x_2 may be fuzzified using membership functions representing various degree of "slow" and "fast". For $n_y = 4$, output y may be defuzzified using membership functions representing various degree of "tiny", "small", "med(ium)", and "large". An initial set of 6 (arbitrary) rules is given by the FAM table shown in Figure 5.11a.

The conclusions of existing rules may be modified by assigning new entries within the FAM table as in Figure 5.11b. Note that the conclusion "huge" in response to premise "x_1 is hot and x_2 is slow" is made possible by the addition of membership function "huge" to the output universe such that $n_y = 5$.

The number of rules is directly affected by the number of membership functions associated with the universe of each input. In Figure 5.11c, the FAM table describes 9 rules since n_{x2} has been increased to 3 with the addition of membership function "faster" to the input universe of x_2.

x_1 x_2	cold	warm	hot
slow	tiny	small	med
fast	med	large	large

a)

x_1 x_2	cold	warm	hot
slow	med	tiny	huge
fast	small	med	large

b)

x_1 x_2	cold	warm	hot
slow	med	tiny	huge
fast	small	med	large
faster	tiny	small	med

c)

Figure 5.11: Rule Base Optimization Variables a) Initial Rule Base, b) Rule Conclusions, c) Number of Rules

Due to difficulties in encoding a rule base of varying size, the minimization of the number of rules requires a non-standard extension to the GA as shown in Figure 5.12 [48]. The figure shows that the minimization of the number of rules used by a genetic-fuzzy controller requires a separate population containing rule sets of all possible sizes. For example, the rule base population of Figure 5.11a would contain $(n_{x1} - 1) \times (n_{x2} - 1) \times (n_y - 1) = 6$ members where each member corresponds to a rule base with unique n_{x1}, n_{x2}, n_y and assuming $n_{x1}, n_{x2}, n_y > 1$ to ensure inputs and outputs are associated with more than one membership function. A particular rule base is selected according to rule base parameters encoded within members of a population encoding the shape and number of membership functions. The GA structure presented in Figure 5.12 ensures that, prior to fitness evaluation, a controller is synthesized with a rule base sized in agreement with the encoded definition of membership functions. Applications of the proposed optimization scheme to a water pump system and a solar heating plant have demonstrated that desired system performance can be maintained while the number of rules is minimized through appropriate selection of rule conclusions and membership functions [48].

Although a worthwhile consideration, the minimization of rule base size is not considered in this thesis since the proposed extension to the GA of Figure 5.11 is problem-specific to the design of genetic-fuzzy controllers. This is in conflict with the current desire to develop a generalized simultaneous plant/controller optimization framework which may incorporate other forms of controllers. However, without modification to the basic GA structure of Figure 5.1, this thesis does consider an optimization over the numerous permutations of the possible conclusions associated

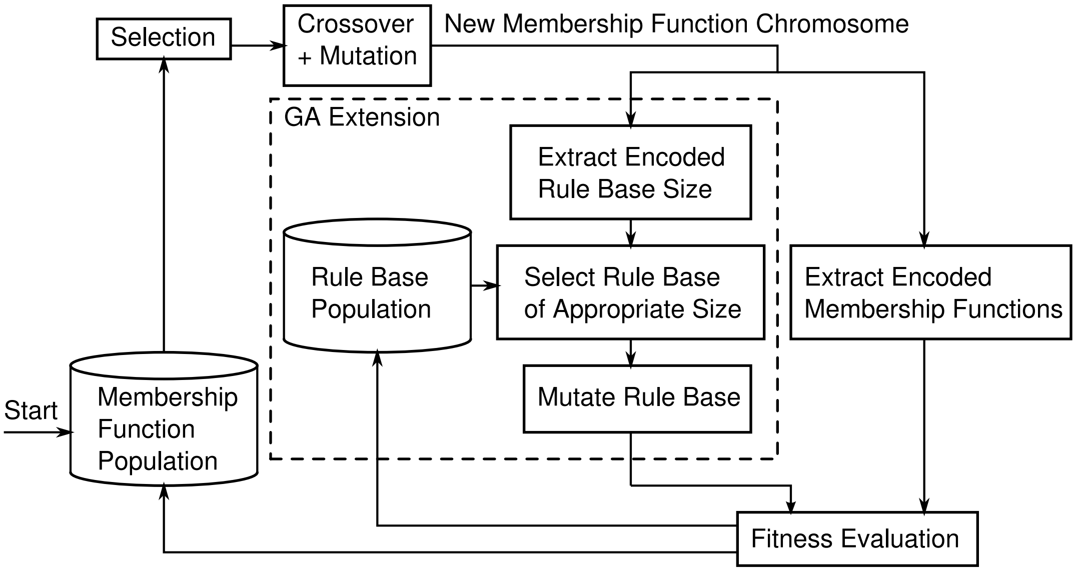


Figure 5.12: GA for Membership Function & Rule Base Optimization [48]

with a given premise in a rule base of fixed size as detailed in Section 5.3.3.

5.2 GA in Traction Control System Optimization

A mixed encoding, multi-chromosomal GA is selected to optimize the traction control system since the problem includes a mixture of continuous and discrete variables. This section describes design details that adapt the general description of the mixed encoding GA of Section 5.1.2 to the traction control optimization problem summarized in Section 4.2.4. With reference to the structure of the mixed encoding, multi-chromosomal GA shown in Figure 5.13, the following discussion describes: the encoding of optimization decision variables within population members, the fitness evaluation and selection process based on candidate solutions associated with low drive cycle energy use O_E and completion time O_T, the combined chromosome shuffling and crossover operation, and the multi-chromosomal mutation operation. To simplify the GA implementation, the optimization process terminates after a fixed number of generations are produced.

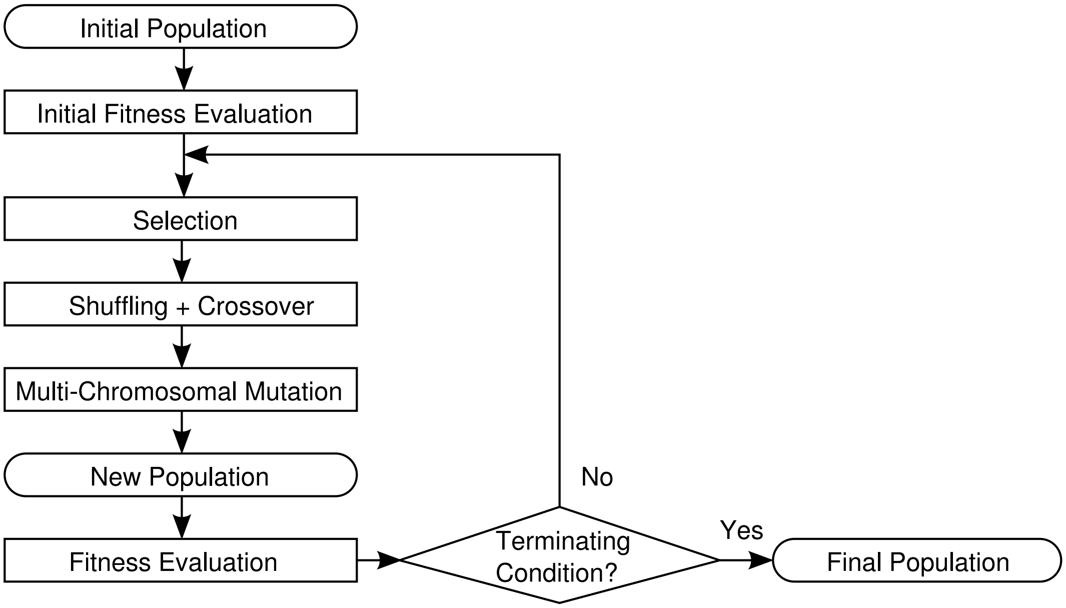


Figure 5.13: Structure of Mixed Encoding, Multi-Chromosomal GA

5.2.1 Decision Variable Encoding

The multi-chromosomal representation $\widehat{X}$ of optimization decision variables RULE, A-PS, and TIRE is shown in Figure 5.14. Since the variables are of different data types, they are encoded in three single-gene chromosomes labeled X_{NODE}, $X_{\text{A-PS}}$, and X_{TIRE}, respectively.

The following discussion provides encoding details of TIRE and A-PS. As shown in Figure 5.14, the fuzzy logic rule base RULE is indirectly encoded by a variable NODE due to considerations in search space reduction and variable constraint reduction. Derivation of the alternative rule base encoding NODE from the RULE encoding is considered separately in Section 5.3 along with associated background in set theory.

Note. The mixed encoding, multi-chromosomal GA operates in a data type independent manner assuming that suitable crossover and mutation operations are defined for each data type in use. This data type independence is established in definitions of the combined chromosome shuffling and crossover operation (Section 5.2.3), and

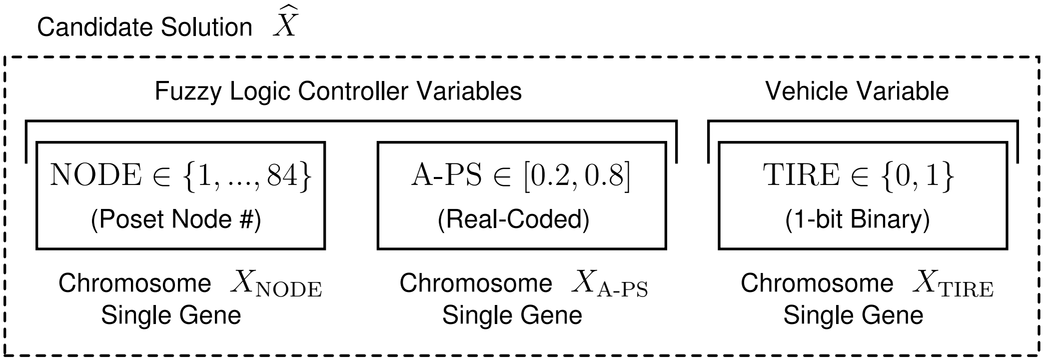


Figure 5.14: Encoding of Traction Control System Design Variables

the multi-chromosomal mutation operation (Section 5.2.4).

TIRE Encoding

Since TIRE is constrained between the selection of either a performance tire or a low rolling resistance tire, it is encoded as a single bit binary gene in chromosome X_{TIRE}. Binary one-point crossover and bit mutation operations are defined on X_{TIRE} as described in Section 5.1.1 although they exhibit degenerate operation on TIRE since X_{TIRE} consists of a single bit.

Given single-gene binary chromosome parents A_{TIRE} and B_{TIRE}, the crossover operation degenerates to the simple assignment $C_{\mathrm{TIRE}} = A_{\mathrm{TIRE}}$ (assuming that the implementation discards the alternative offspring $C'_{\mathrm{TIRE}} = B_{\mathrm{TIRE}}$). An implementation may consistently discard either offspring since the two parents to the crossover operation are randomly chosen in the selection process.

Mutation of the single-gene binary chromosome C_{TIRE} causes TIRE to invert and therefore toggles tire selection between the performance tire and the low rolling resistance tire.

A-PS Location Encoding

Since A-PS represents the location of the Positive-Small adhesion gradient membership function over a continuous range of values, it is real-coded as a floating point number.

Arithmetic crossover on two single-gene parents $A_{\text{A-PS}}$ and $B_{\text{A-PS}}$ is defined as in (5.4) but degenerates to the average of A-PS stored in both parents such that offspring $C_{\text{A-PS}} = \dfrac{A_{\text{A-PS}} + B_{\text{A-PS}}}{2}$.

Gaussian mutation on offspring $C_{\text{A-PS}}$ is defined in Equation (5.5) and, in the case of a single-gene chromosome, produces a perturbed chromosome $D_{\text{A-PS}} = C_{\text{A-PS}} + \psi(0, \sigma)$ such that a random number is added to A-PS with a distribution defined by Gaussian function ψ with zero mean and standard deviation σ.

Due to the constraint A-PS $\in [0.2, 0.8]$ and the fact that ψ has an unbounded range and can return large values with small probability, a repair mechanism is required to prevent an out-of-bounds assignment to A-PS. Simply saturating a mutation to the lower and upper bounds allowed for A-PS would greatly increase the occurrences of mutations resulting in A-PS being assigned a minimum or maximum limit. To avoid this, a repair on an out-of-bounds mutation is done by "bouncing" [53] the mutation back within the allowable range by first saturating the sum of $C_{\text{A-PS}} + \psi(0, \sigma)$ to the bounds of A-PS while retaining the remainder exceeding the violated bound. The negative of the remainder is then added back to the sum and the process is repeated until the remainder is zero.

Example 5.2.1. Assume A-PS $\in [0.2, 0.8]$. If $C_{\text{A-PS}} = 0.3$ and a Gaussian mutation results in a change of $\psi(0, \sigma) = -0.9$, then the sum $0.3 + (-0.9)$ is saturated to the lower bound of A-PS to produce the intermediate result $C'_{\text{A-PS}} = 0.2$ and the remainder exceeding the lower bound is -0.8. The negative of the remainder is then added to $C'_{\text{A-PS}}$. The sum $(C'_{\text{A-PS}} + 0.8)$ is saturated to the upper bound to produce the intermediate result $C''_{\text{A-PS}} = 0.8$ with a remainder of 0.2. The final addition of the negative of the remainder to $C''_{\text{A-PS}}$ produces the final mutation assignment $D_{\text{A-PS}} = 0.8 - 0.2 = 0.6$ that satisfies the constraint on A-PS.

5.2.2 Fitness Evaluation and Selection

The fitness of each member $\widehat{X}$ within the population is evaluated by first configuring the traction control system model according to the design variables encoded in $\widehat{X}$ and simulating system dynamics to evaluate the drive cycle energy use O_E and completion time O_T objectives. The system-level objective value O_s is then evaluated using the weighted sum of O_E and O_T as defined in (4.12). After O_s is evaluated for all members within the population, the ranking scheme is applied by sorting members in descending order according to O_s such that members with lower O_s are assigned a higher rank and fitness. Therefore, members are assigned fitness values appropriate for the minimization of O_s. For each member to be generated in the new population, roulette wheel selection is applied twice to select the parent members for the subsequent combined shuffling and crossover operation.

5.2.3 Chromosome Shuffling and Crossover

Chromosome shuffling is implemented as a **one-point shuffle** that is analogous to the one-point crossover operation. For two parent members with a multi-chromosomal representation $\widehat{A} = \{A_1, A_2, \ldots, A_m\}$ and $\widehat{B} = \{B_1, B_2, \ldots, B_m\}$ of length m, a random crossover point $j \in \{1, \ldots, m\}$ is chosen such that a one-point shuffle about j results in child $\widehat{C} = \{A_1, \ldots, A_j, B_{j+1}, \ldots, B_m\}$.

Although shuffling and crossover can be done simultaneously as illustrated in Figure 5.8, in this thesis only one is randomly performed when generating an offspring from two parents $\widehat{A}$ and $\widehat{B}$. As shown in Figure 5.15, a chromosome j is randomly selected and, with equal probability, either a one-point shuffle is performed about chromosome j or a one-point crossover is performed on chromosome j to produce offspring $\widehat{C}$. This definition of the shuffling and crossover operation allows the GA to consider: sequences of chromosomes as building blocks when shuffling is performed, or sequences of genes within a chromosome as building blocks when crossover is performed.

Note. Recall from the building block hypothesis (Section 5.1.1) that building blocks are short sequences of genetic information that, when present in an encoding, increase the fitness of the associate population member.

As defined, the shuffling and crossover operation is independent from the underlying chromosome data types of the parent members. Either an entire chromosome

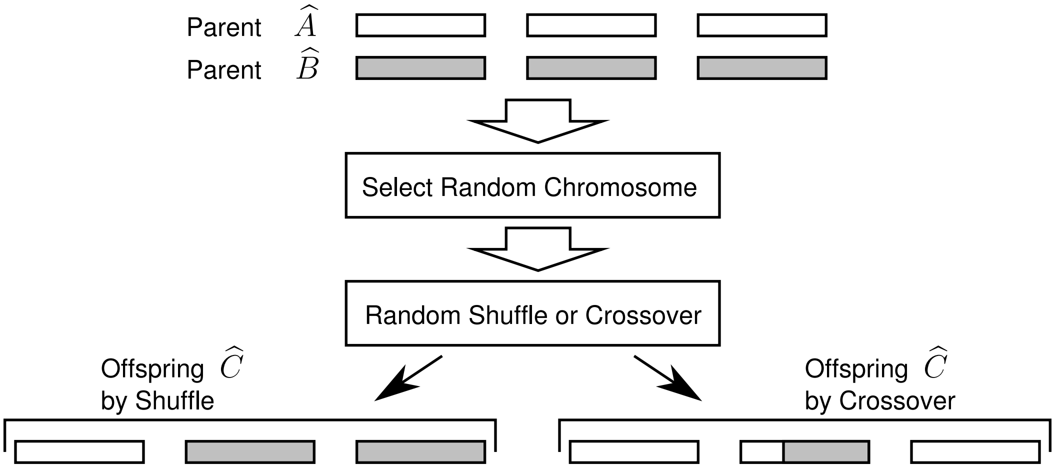


Figure 5.15: Random Chromosome Shuffle or Crossover Operation

is exchanged in the shuffling operation, or the crossover operation of the chromosome is invoked by the shuffling and crossover operation of the mixed encoding, multi-chromosomal GA. Hence, the shuffling and crossover operation can be extended to support new chromosome data types provided that a crossover operation is defined on the data type.

5.2.4 Multi-Chromosomal Mutation

The mutation of a member consisting of a multi-chromosomal representation proceeds by choosing a random chromosome within the member and invoking the appropriate mutation operation depending the chromosome's data type. Unlike the shuffling and crossover operation that must be performed to generate each offspring of a new population, the multi-chromosomal mutation operation is applied to an offspring with a small mutation probability p_m.

The mutation probability p_m is chosen large enough such that new building blocks may be produced through mutation. However, excessive mutation may also destroy existing building blocks. For this reason, the multi-chromosomal mutation is defined such that a change occurs in a single chromosome of a multi-chromosomal representation. This is in contrast to the definition of the mutation process defined

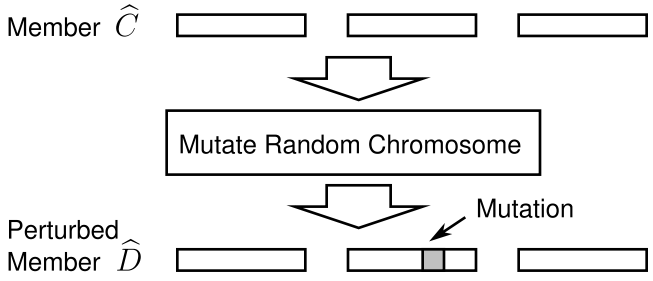


Figure 5.16: Multi-Chromosomal Mutation

in (5.7) where each chromosome in the multi-chromosomal representation is mutated.

Multi-chromosomal mutation is shown in Figure 5.16 where a single chromosome of member $\widehat{C}$ is mutated to produce a perturbed member $\widehat{D}$. As defined, the multi-chromosomal mutation operation is independent from the underlying chromosome data types of member $\widehat{C}$ since the operation invokes the mutation operation associated with the data type. The multi-chromosomal mutation operation can be extended to support new chromosome data types provided that a mutation operation is defined on the data type.

Since the shuffling and crossover operation and the multi-chromosomal mutation operation are responsible for generating new candidate solutions, their independence from chromosome data types allows the mixed encoding, multi-chromosomal GA to be extended to support a new data type provided that crossover and mutation operations can be defined on the data type. This property is exploited in Section 5.3 to define a novel encoding for the fuzzy logic rule base.

5.3 Novel Fuzzy Logic Rule Base Encoding

Motivation for a novel fuzzy logic controller rule base encoding is obtained by considering a simple and direct encoding of rule base conclusions as independent integers. The direct encoding leads to exponential growth of the search space with the number of rule base entries. For example, encoding the 6 rule base conclusion variables $\mathcal{C}_i$ from RULE of the traction controller results in a search space with $4096 = 4^6$ rule base configurations since each of the variables may assume any of 4 possible conclusions according to (4.9).

To reduce the search space, application-specific knowledge may be used to derive constraints on allowable rule base configurations. For the traction control system, the constraints in (4.10) are intuitively derived to admit only those that tend to prescribe an increase in torque output T_a as input torque request T_{ref} and adhesion gradient A increase. However, such constraints may create a large difference in size (large infeasible search space) between the search space that exists prior to constraint application and the smaller feasible search space defined by the constraints. As described in Section 5.1.3, a constraint handling mechanism must then be incorporated in the optimizing GA to deal with infeasible solutions that violate constraints.

In the current optimization of the traction control system, the rule base is heavily constrained by (4.10) (to be quantified in the encoding analysis of Section 5.3.3) to produce a small feasible search space and, consequently, a large infeasible search space. With the direct rule base encoding, constraint handling methods such as rejection and the penalty scheme are undesirable due to high computational overhead associated with simulating infeasible solutions. It is also currently not obvious how to repair an infeasible rule base definition. Thus the direct rule base encoding leads to challenges in either exploring a large search space, or reducing computational overhead associated with evaluating infeasible solutions.

An alternative and novel encoding of rule base conclusions is therefore proposed to evolve an optimal rule base configuration within a constrained search space while avoiding infeasible solutions. Minimal background in set theory is presented to enable the definition of a partial ordering of “monotonic increasing” matrices. The partial ordering, termed a partial ordering by delta inclusion, is then applied to encode the rule base for use in the mixed-encoding, multi-chromosomal GA previously described in Section 5.2.

Table 5.2: Preliminary Set Theory Notation [54]

Notation	Meaning
$a = b$	"a" and "b" denote the same thing
$a \neq b$	"a" and "b" denote different things
$a \in S$	"a" is an element of set "S"
$P \Rightarrow Q$	If "P" then "Q" ("P" implies "Q")
$\forall$	"for all"
$\exists$	"there exists"
$\ni$	"such that"
$P(x)$	A statement P in the variable x
$\Leftrightarrow$	"If and Only if"
$A \subseteq B$	"set A is a subset of set B"

5.3.1 Set Theory

This section reviews the set theory required to define a lattice. All definitions in this section are taken from [54] unless stated otherwise.

Definition 5.3.1. A set is a collection of objects. These objects may be physical or abstract. Most mathematical objects can be defined in terms of sets including points, numbers, and operations.

Set theory formalizes the expressive power of sets and, as such, is part of the foundation for mathematics [54]. Table 5.2 reviews set theory notation used in this section. Set theory notation is used to abbreviate statements in set theory. For example, the subset statement $A \subseteq B$ is an abbreviation for the statement:

$$\forall\, x, \quad x \in A \Rightarrow x \in B.$$

Definition 5.3.2. Given a set S, the **power set** of S, denoted by $X = [S]$, is a set where:

$$\forall\, x, \quad x \in [S] \Leftrightarrow x \subseteq S.$$

The power set of S is therefore a set whose elements $x \in X$ are all the possible subsets of S. In other words, the elements x of a power set are themselves sets. Therefore, in this case, x carries a double meaning of an element $x \in X$ and of a set $x \subseteq [S]$.

Definition 5.3.3. Given two sets A and B, an **ordered pair** is denoted as (a, b) where $a \in A$ and $b \in B$. Ordered pairs have the property that:

$$(a, b) = (c, d) \Leftrightarrow a = c,\ b = d, \quad a, c \in A, \quad b, d \in B.$$

Definition 5.3.4. The **cartesian product** of two sets A and B is defined as:

$$A \times B = \{(a, b)|\ a \in A, b \in B\}.$$

The cartesian product $A \times B$ is therefore a set containing all ordered pairs formed from elements of sets A and B.

Definition 5.3.5. Given two sets A and B, and the subset f of $A \times B$, then f is a **mapping** of A onto B if:

1. For each $a \in A$, $\exists\, b \in B \ni (a, b) \in f$.
2. $(a, b) \in f$ and $(a, c) \in f \Rightarrow b = c$.

In the mapping of A onto B, each element of A is assigned to exactly one element of B. The mapping f may also be called a **function** or **transformation**. Also, the notation afb means $(a, b) \in f$.

Definition 5.3.6. A **relation** F from set A to set B is any subset of $A \times B$.

If elements $a \in A$ and $b \in B$ are in a relation F, then $(a, b) \in F$. Also, $(a, b) \in F$ is abbreviated as aFb. A relation can be considered as a rule that assigns to each $a \in A$ to zero or more elements $b \in B$. A mapping can therefore be interpreted as a special case of a relation since each element of A is assigned to one element of B.

The sets A and B can be different sets or refer to the same set X. When A and B refer to the same set, the relation F from A to B is called a relation F on set X.

Definition 5.3.7. A **partial ordering** defined on a set X is a relation F on X satisfying the following three properties:

1. **Relexivity:** $aFa, \quad \forall\, a \in X$.
2. **Antisymmetry:** $(aFb \quad \text{and} \quad bFa) \Rightarrow a = b, \quad \forall\, a, b \in X$.
3. **Transitivity:** $(aFb \quad \text{and} \quad bFc) \Rightarrow aFc, \quad \forall\, a, b, c \in X$.

A set with a defined partial ordering is termed a partially ordered set or **poset** [55]. In general, $a \leq b$ can be written for aFb when (a, b) is a member of a partial ordering. If, additionally, $a \neq b$ then $a < b$ may be written [54]. Elements $a, b \in X$ are **comparable** if either $a \leq b$ or $b \leq a$. In a partial ordering, it may be the case that two arbitrary elements a, b are not comparable [55].

Note. In set theory, the symbol "$\leq$" denotes any relation F that can be used to partially order an abstract set X as defined above. Hence the use of "$\leq$" in set theory is not limited to the "usual" use in ordering sets of numbers. However, for the particular case where X represents a set of numbers (for example $X = \mathbb{R}$, the set of real numbers), the symbol "$\leq$" in "$a \leq b$" represents a particular partial ordering where the statement $a \leq b$ can be writen when $(a, b) \in$ "$\leq$" $\Leftrightarrow$ the numerical value of a is less than or equal to the numerical value of b, $a, b \in \mathbb{R}$. This leads to the usual interpretation of $a \leq b$ when applied to numbers. In this thesis, the meaning of "$\leq$" is inferred by context and will denote either an abstract relation when applied to general sets, or the usual numerical ordering operation when applied to numbers. Similar consideration should be given to the use of the symbol "$<$".

Example 5.3.1. Partial ordering by inclusion is a common example of partial ordering and is defined as a relation F on the power set of S as follows: Given the power set $X = [S]$, the relation F on X is such that $(A, B) \in F \Leftrightarrow A \subseteq B, \quad A, B \in X$. Therefore, if $A \subseteq B$, then (A, B) is a member of a partial ordering by inclusion and $A \leq B$ may be written [54]. Intuitively, set A is smaller or equal than set B since it contains fewer or equal elements and, in this way, the elements of a power set can be partially ordered.

The partial ordering between elements in a poset X can be visualized in a type of directed graph called a **Hasse diagram**. In a Hasse diagram, nodes correspond to elements $x \in X$ and edges to pairs (x, y) with $x < y, \quad x, y \in X$. Edges are drawn as rising lines from a "lower" node x to a "higher" node y to indicate $x < y$. To simplify the Hasse diagram, edges are only drawn for "neighboring" x, y [55].

Example 5.3.2. The partial ordering by inclusion of the power set of $S = \{a, b, c\}$ is shown in the Hasse diagram of Figure 5.17. The following steps were used to derive the graph:

1. The power set of S is $X{=}[S]{=}\{\{a, b, c\}, \{a, b\}, \{a, c\}, \{b, c\}, \{a\}, \{b\}, \{c\}, \emptyset\}$. There are 8 elements in X and each corresponds to a node in Figure 5.17.
2. A partial ordering is defined as a relation F on X where F is a subset of $X \times X$. Since X is partially ordered by inclusion, the elements of F are ordered pairs (A, B) where $A \subseteq B$ and $A, B \in X$. Specifically:

$$\begin{aligned} F = \{&(\{a,b,c\},\{a,b,c\}),(\{a,b\},\{a,b,c\}),(\{a,c\},\{a,b,c\}),(\{b,c\},\{a,b,c\}),\\ &(\{a\},\{a,b,c\}),(\{b\},\{a,b,c\}),(\{c\},\{a,b,c\}),(\emptyset,\{a,b,c\}),\\ &(\{a,b\},\{a,b\}),(\{a\},\{a,b\}),(\{b\},\{a,b\}),(\emptyset,\{a,b\}),\\ &(\{a,c\},\{a,c\}),(\{a\},\{a,c\}),(\{c\},\{a,c\}),(\emptyset,\{a,c\}),\\ &(\{b,c\},\{b,c\}),(\{b\},\{b,c\}),(\{c\},\{b,c\}),(\emptyset,\{a,b\}),\\ &(\{a\},\{a\}),(\emptyset,\{a\}),(\{b\},\{b\}),(\emptyset,\{b\}),(\{c\},\{c\}),(\emptyset,\{c\}),(\emptyset,\emptyset)\}. \end{aligned}$$

3. Edges are drawn between nodes A and B to indicate $A < B$. Only edges between "neighboring" A, B are drawn. For example, although node $\{a\}$ is a subset of each of nodes $\{a, b\}$, $\{a, c\}$, and $\{a, b, c\}$, edges are drawn only between it and its neighbors $\{a, b\}$ and $\{a, c\}$. The fact that $\{a\} \subseteq \{a, b, c\}$ is implied by the edges between $\{a, b\}$ - $\{a, b, c\}$ and $\{a, c\}$ - $\{a, b, c\}$.

Definition 5.3.8. Given a poset X and elements $x, y \in X$, an element $a \in X$ is a **lower bound** for x and y if $a \leq x$ and $a \leq y$. Multiple lower bounds may exist

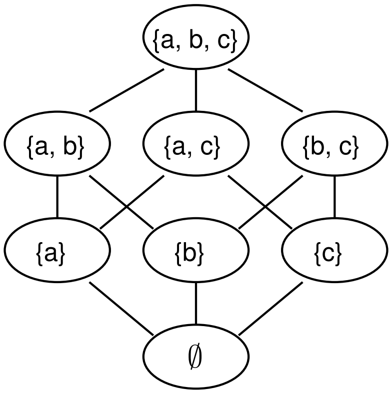


Figure 5.17: Example Hasse Diagram of a Partially Ordered Set by Inclusion [55]

for x and y. However, a **greatest lower bound**, or **meet**, $l \in X$ for x and y may exist, and is unique if so. An element $l \in X$ is defined as a meet when [55]:

$$\begin{aligned} &l \leq x \text{ and } l \leq y \quad (l \text{ is a lower bound for } x \text{ and } y),\\ &(\forall\, a \in X)\, a \leq x \text{ and } a \leq y \Rightarrow a \leq l\\ &\qquad\qquad\qquad\qquad (l \text{ is greater than other lower bounds for } x \text{ and } y). \end{aligned}$$

Definition 5.3.9. Similarly, an element $b \in X$ is an **upper bound** for x and y if $x \leq b$ and $y \leq b$. Multiple upper bounds may also exist for x and y. A **least upper bound**, or **join**, $u \in X$ may exist, and is unique if so. An element $u \in X$ is defined as a join when [55]:

$$\begin{aligned} &x \leq u \text{ and } y \leq u \quad (u \text{ is an upper bound for } x \text{ and } y),\\ &(\forall\, b \in X)\, x \leq b \text{ and } y \leq b \Rightarrow u \leq b\\ &\qquad\qquad\qquad\qquad (u \text{ is less than other upper bounds for } x \text{ and } y). \end{aligned}$$

Definition 5.3.10. A **lattice** is a special case of a poset and occurs when the meet and join of any two elements of the poset always exist. Given a lattice L, the meet of two elements $x, y \in L$ is written as $x \wedge y$, and the join of the elements as $x \vee y$ [55].

Example 5.3.3. The partial ordering by inclusion of $[S]$ is a lattice [55]. Intuitively, this is because the meet and join operators, $\wedge$ and $\vee$, are equivalent to the set union and intersection operators, $\cap$ and $\cup$, respectively. The intersection of two sets is the biggest set that contains elements common to both sets (meet). The union of two sets is the smallest set that contains all elements of both sets (join). For example, in Figure 5.17:

$$\{a, b\} \wedge \{a, c\} = \{a\} = \{a, b\} \cap \{a, c\}$$
$$\text{and}$$
$$\{a\} \vee \{b, c\} = \{a, b, c\} = \{a\} \cup \{b, c\}.$$

5.3.2 Novel Partial Ordering by Delta Inclusion

This section builds on Section 5.3.1 to define a novel partial ordering on a set termed "partial ordering by delta inclusion (poset by delta inclusion)." The mechanism of delta inclusion is similar to that of subset inclusion in that both rely on the notion that certain elements of X "contain" other elements of X. In contrast to subset inclusion where sets "contain" other sets, delta inclusion relies on the idea that matrices "contain" other matrices. The remainder of this section describes posets by delta inclusion. First, a mechanism to compare particular matrices is defined. Then the construction of X and the relation F on X are considered with respect to posets by delta inclusion.

Consider an $m \times n$ matrix, $E^{m \times n}$, with entries $e_{i,j}$ for all $1 \leq i \leq m$ and $1 \leq j \leq n$:

$$\begin{bmatrix} e_{1,1} & \dots & e_{1,n} \\ \vdots & \ddots & \vdots \\ e_{m,1} & \dots & e_{m,n} \end{bmatrix} = \begin{bmatrix} e_{i,j} \end{bmatrix}, \quad \forall\, i = 1..m, j = 1..n.$$

Definition 5.3.11. Define a **monotonic increasing matrix**, $E^{m \times n}$, as a matrix with non-decreasing components from the set of natural numbers:

$$\begin{array}{lll} e_{i,j} \in \{1, 2, 3, ...\}, & i = 1..m, & j = 1..n \\ e_{i,j} \le e_{i,j+1}, & i = 1..m, & j = 1..(n-1) \\ e_{i,j} \le e_{i+1,j}, & i = 1..(m-1), & j = 1..n. \end{array}$$

That is, each entry of a monotonic increasing matrix is a natural number and is less than or equal to entries immediately to the right, below, and diagonally to the right and below.

Example 5.3.4. Examples of monotonic increasing matrices are:

$$\begin{bmatrix} 2 & 2 \\ 3 & 3 \end{bmatrix}, \quad \begin{bmatrix} 1 & 2 \\ 3 & 3 \end{bmatrix}, \quad \text{and} \quad \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}.$$

Definition 5.3.12. Given two monotonic increasing matrices, $A^{m \times n}, B^{m \times n}$, define the **delta inclusion** operator "$\Subset$" as:

$$A^{m \times n} \Subset B^{m \times n} \Leftrightarrow (a_{i,j} \le b_{i,j}, \, \exists \, a_{i,j} < b_{i,j}) \quad \forall \, i = 1..m, j = 1..n.$$

Similar to the subset operator $\subseteq$, the delta inclusion operator $\Subset$ invokes the notion of "containment."

Example 5.3.5. By delta inclusion, it can be said that:

$$\begin{bmatrix} 1 & 2 \\ 3 & 3 \end{bmatrix} \Subset \begin{bmatrix} 2 & 2 \\ 3 & 3 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \Subset \begin{bmatrix} 1 & 2 \\ 3 & 3 \end{bmatrix}.$$

Definition 5.3.13. Define the **delta set** of a monotonic increasing matrix $T^{m \times n}$, written $\langle T^{m \times n} \rangle$, as a set that includes all monotonic increasing matrices $E^{m \times n}$ where:

$$\forall \, E^{m \times n}, \quad E^{m \times n} \in \langle T^{m \times n} \rangle \Leftrightarrow E^{m \times n} \Subset T^{m \times n}.$$

Example 5.3.6. For example, given $T^{2 \times 2} = \begin{bmatrix} 2 & 2 \\ 3 & 3 \end{bmatrix}$, then the delta set of $T^{2 \times 2}$ is the set of 14 possible monotonic increasing matrices:

$$X = \langle T^{2\times 2}\rangle = \left\langle \begin{bmatrix} 2 & 2 \\ 3 & 3 \end{bmatrix} \right\rangle = \left\{ \begin{bmatrix} 2 & 2 \\ 3 & 3 \end{bmatrix}, \begin{bmatrix} 1 & 2 \\ 3 & 3 \end{bmatrix}, \begin{bmatrix} 2 & 2 \\ 2 & 3 \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ 3 & 3 \end{bmatrix}, \begin{bmatrix} 1 & 2 \\ 2 & 3 \end{bmatrix}, \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix}, \right.$$
$$\left. \begin{bmatrix} 1 & 1 \\ 2 & 3 \end{bmatrix}, \begin{bmatrix} 1 & 2 \\ 1 & 3 \end{bmatrix}, \begin{bmatrix} 1 & 2 \\ 2 & 2 \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ 1 & 3 \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ 2 & 2 \end{bmatrix}, \begin{bmatrix} 1 & 2 \\ 1 & 2 \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \right\}.$$

Definition 5.3.14. Given a delta set $X = \langle T^{m\times n}\rangle$, a **partial ordering by delta inclusion** is defined as a relation F on X such that:

$$(A^{m\times n}, B^{m\times n}) \in F \Leftrightarrow A^{m\times n} \Subset B^{m\times n}, \quad A^{m\times n}, B^{m\times n} \in X.$$

Example 5.3.7. The Hasse diagram for the delta set $X = \left\langle \begin{bmatrix} 2 & 2 \\ 3 & 3 \end{bmatrix} \right\rangle$, as partially ordered by delta inclusion, is shown in Figure 5.18 (ignoring for the moment the "level" indications along the left margin). Each node of the Hasse diagram corresponds to an element of X, where the elements of X are the 14 monotonic increasing matrices described above. Note that "neighboring" matrices $A^{m\times n}, B^{m\times n}$ in the Hasse diagram differ in one entry. That difference, or delta, is ± 1. Specifically, given neighboring $A^{m\times n}, B^{m\times n}$ and $A^{m\times n} \Subset B^{m\times n}$, exactly one entry of $A^{m\times n}$ is 1 less than the corresponding entry of $B^{m\times n}$. Intuitively, $A^{m\times n}$ is smaller or equal than $B^{m\times n}$ since all entries of $A^{m\times n}$ are less than or equal to corresponding entries of $B^{m\times n}$. In this way, the elements of a delta set X can be partially ordered.

Certain elements in a delta set X may not be comparable. This is a general property of partial ordering (Definition 5.3.7).

Example 5.3.8. The elements $A = \begin{bmatrix} 1 & 2 \\ 2 & 3 \end{bmatrix}$ and $B = \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix}$ from Figure 5.18 are not comparable since:

$$\text{Neither } A \Subset B \text{ nor } B \Subset A.$$

Definition 5.3.15. Given a delta set $X = \langle T^{m\times n}\rangle$, the meet (or greatest lower bound) of $A^{m\times n}, B^{m\times n} \in X$ is a matrix whose entries are the minimums of corresponding entries from $A^{m\times n}$ and $B^{m\times n}$:

$$A^n \wedge B^n = \Big[\min(a_{i,j}, b_{i,j})\Big], \quad i = 1..m, j = 1..n.$$

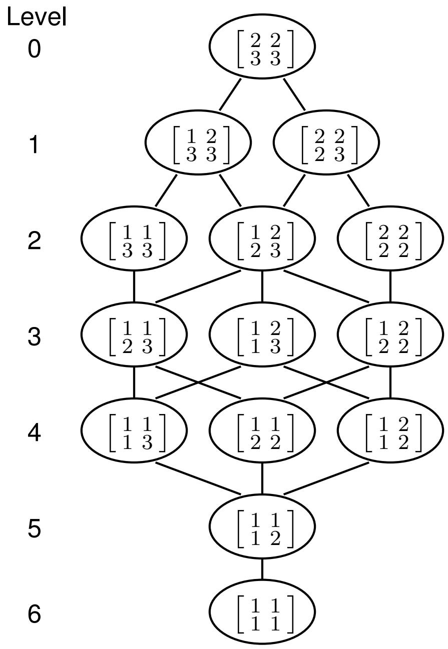


Figure 5.18: Hasse Diagram of Poset by Delta Inclusion with Level Indications

Definition 5.3.16. Dually, the join (or lowest upper bound) of $A^{m\times n}, B^{m\times n} \in X$ is a matrix whose entries are the maximums of corresponding entries from $A^{m\times n}$ and $B^{m\times n}$:

$$A^n \vee B^n = \Big[\max(a_{i,j}, b_{i,j})\Big], \quad i = 1..\,m, j = 1..\,n.$$

The existence and uniqueness of the above definitions of meet and join follow from the existence and uniqueness of the minimum and maximum of two numbers. Hence a poset by delta inclusion is also a lattice by Definition 5.3.10.

Example 5.3.9. In Figure 5.18:

$$\begin{bmatrix}1 & 2\\ 3 & 3\end{bmatrix} \wedge \begin{bmatrix}2 & 2\\ 2 & 2\end{bmatrix} = \begin{bmatrix}\min(1,2) & \min(2,2)\\ \min(3,2) & \min(3,2)\end{bmatrix} = \begin{bmatrix}1 & 2\\ 2 & 2\end{bmatrix}$$

and

$$\begin{bmatrix}1 & 2\\ 2 & 3\end{bmatrix} \vee \begin{bmatrix}2 & 2\\ 2 & 2\end{bmatrix} = \begin{bmatrix}\max(1,2) & \max(2,2)\\ \max(2,2) & \max(3,2)\end{bmatrix} = \begin{bmatrix}2 & 2\\ 2 & 3\end{bmatrix}.$$

As seen in the Hasse diagram of Figure 5.18, both elements $\begin{bmatrix}1 & 2\\2 & 3\end{bmatrix}, \begin{bmatrix}2 & 2\\2 & 2\end{bmatrix} \in \left\langle \begin{bmatrix}2 & 2\\3 & 3\end{bmatrix} \right\rangle$ appear in the same horizontal plane. This is not a coincidence since both elements are the same "distance" away from $T^{2\times 2} = \begin{bmatrix}2 & 2\\3 & 3\end{bmatrix}$. In other words, both elements are the result of decrementing entries of $T^{2\times 2}$ a total of 2 times. For example, the $t_{1,1}$ entry of $T^{2\times 2}$ is first decremented to give $\begin{bmatrix}1 & 2\\3 & 3\end{bmatrix}$, followed by a second decrement on entry $t_{2,1}$ to give $\begin{bmatrix}1 & 2\\2 & 3\end{bmatrix}$.

To generalize, let $\sum A$ denote the sum of all elements in the matrix A:

$$\sum A = \sum_{i=1}^{m} \sum_{j=1}^{n} a_{i,j}.$$

Definition 5.3.17. Define the **level** of an element $E^{m\times n}$ in delta set $X = \langle T^{m\times n} \rangle$ as:

$$level(E^{m\times n}) = \sum T^{m\times n} - \sum E^{m\times n}.$$

Intuitively, the level is the number of decrements necessary to produce $E^{m\times n}$ from $T^{m\times n}$.

Note. Although some pairs of elements in a partial ordering are not comparable (Example 5.3.8), their positions in a Hasse diagram may still be sepecified using the level operator.

Note. Given elements $A^{m\times n}, B^{m\times n} \in X$:

$$level(A^{m\times n} \wedge B^{m\times n}) \geq \begin{cases} level(A^{m\times n}) \\ level(B^{m\times n}) \end{cases} \tag{5.11}$$

and

$$level(A^{m\times n} \vee B^{m\times n}) \leq \begin{cases} level(A^{m\times n}) \\ level(B^{m\times n}). \end{cases} \tag{5.12}$$

Example 5.3.10. The level of each matrix is indicated along the left margin of the Hasse diagram for the example in Figure 5.18. The following statements demonstrate the use of the level operator:

$$\begin{aligned}
&\text{Given } A^{2\times 2} = \begin{bmatrix} 1 & 2 \\ 3 & 3 \end{bmatrix}, B^{2\times 2} = \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix} \in \left\langle \begin{bmatrix} 2 & 2 \\ 3 & 3 \end{bmatrix} \right\rangle, \text{then} \\
&\quad \mathit{level}\left(A^{2\times 2}\right) = 1, \text{ and } \mathit{level}\left(B^{2\times 2}\right) = 2, \\
&\quad \mathit{level}\left(A^{2\times 2} \wedge B^{2\times 2}\right) = \mathit{level}\left(\begin{bmatrix} 1 & 2 \\ 2 & 2 \end{bmatrix}\right) = 3 \geq \begin{cases} \mathit{level}\left(A^{2\times 2}\right) \\ \mathit{level}\left(B^{2\times 2}\right), \end{cases} \\
&\quad \mathit{level}\left(A^{2\times 2} \vee B^{2\times 2}\right) = \mathit{level}\left(\begin{bmatrix} 2 & 2 \\ 3 & 3 \end{bmatrix}\right) = 0 \leq \begin{cases} \mathit{level}\left(A^{2\times 2}\right) \\ \mathit{level}\left(B^{2\times 2}\right). \end{cases}
\end{aligned}$$

5.3.3 Application to Rule Base Encoding

The encoding of the fuzzy logic controller rule base is defined in terms of a monotonic increasing matrix to allow the definition of the delta set containing possible rule base configurations. The delta set of rule base configurations is partially ordered by delta inclusion and specialized crossover and mutation operations are defined based on the meet, join, and level of set members. Advantages and current weaknesses requiring future work of the novel encoding are presented.

Delta Set of Rule Base Configurations

To define the rule base in terms of a monotonic increasing matrix, first recall the definition of the rule base optimization variable RULE from (4.7) along with associated constraints in (4.9) and (4.10) as re-stated in (5.13) to (5.15), respectively:

$$\text{RULE} = \begin{bmatrix} \mathcal{C}_1 & \mathcal{C}_2 \\ \mathcal{C}_3 & \mathcal{C}_4 \\ \mathcal{C}_5 & \mathcal{C}_6 \end{bmatrix} \tag{5.13}$$

$$\mathcal{C}_i \in \{\text{S, MS, MH, H}\} \Leftrightarrow \mathcal{C}_i \in \{1, 2, 3, 4\} \tag{5.14}$$

$$\begin{aligned} &\mathcal{C}_1 \leq \mathcal{C}_2,\ \mathcal{C}_3 \leq \mathcal{C}_4,\ \mathcal{C}_5 \leq \mathcal{C}_6, \\ &\mathcal{C}_1 \leq \mathcal{C}_3,\ \mathcal{C}_3 \leq \mathcal{C}_5,\ \mathcal{C}_2 \leq \mathcal{C}_4,\ \mathcal{C}_4 \leq \mathcal{C}_6, \\ &\mathcal{C}_2 \leq 2,\ \mathcal{C}_4 \leq 3. \end{aligned} \tag{5.15}$$

By Definition 5.3.11 and the first seven constraints in (5.15), RULE is a monotonic increasing matrix and may be manipulated according to the definitions in Section 5.3.2. Consideration of the last two constraints in (5.15), along with (5.14), therefore allows the set of all possible rule base configurations that may be assigned to RULE to be defined by the delta set X:

$$X = \left\langle \begin{bmatrix} 2 & 2 \\ 3 & 3 \\ 4 & 4 \end{bmatrix} \right\rangle. \tag{5.16}$$

Thus it is possible to interpret RULE as encoding a member of delta set X:

$$\text{RULE} \in X \tag{5.17}$$

where existing constraints (5.14) and (5.15) on RULE are incorporated in the definition of X. Further justification for encoding the rule base as a member of delta set X is provided by considering associated properties of the delta inclusion operator and the poset by delta inclusion that enable an intuitive understanding of the impact of changes in rule base selection on optimization objectives.

Given two rule base configurations A, $B \in X$ such that $A \Subset B$, rule base B prescribes a higher output torque T_a then A in some rule base entries and equal rule base definition in the remaining entries by Definition 5.3.12 of the delta inclusion operator $\Subset$. A comparison of the **aggressiveness** between rule base configurations A and B is therefore defined where the higher torque output of aggressive rule base configuration B leads to lower drive cycle completion time O_T and higher associated energy use O_E. Thus, the delta inclusion operator provides an effective mechanism to compare the aggressiveness of rule base configurations and to assess the impact of a change in rule base selection in RULE on optimization objectives O_E and O_T where:

$$A \Subset B \Rightarrow (A \text{ is less aggressive than } B). \tag{5.18}$$

Unfortunately, the delta inclusion operator may not be exclusively used to guide the selection of rule base configuration since "sorting" delta set X according to delta inclusion results in a poset by delta inclusion (Definition 5.3.14). As a partial ordering, there exists rule base configurations in X that are not comparable by delta inclusion. Thus, the concept of comparing rule base configurations based on aggressiveness is extended by considering the level (Definition 5.3.17) of rule base configurations in X.

The Hasse diagram in Figure 5.19 shows the poset by delta inclusion of all possible rule base configurations in X. The level of each rule base configuration is indicated along the left margin of the diagram. Additionally, a unique **node number** is assigned to each member of X as indicated in bold to the top-left side of each node of the Hasse diagram. The node number serves as a compact reference of the associated rule base configuration and provides convenience in discussion as well as encoding implementation.

As shown in Figure 5.19, delta inclusion between two rule base configurations A, $B \in X$ implies the following relationship in their levels:

$$A \Subset B \Rightarrow level(A) > level(B). \tag{5.19}$$

Although (5.19) implies a unidirectional if-then relationship on comparable set members, it is assumed that the relationship on the level of A and B and their aggressiveness can be extended to rule base configurations that are not comparable using delta inclusion such that:

$$(A \text{ is less aggressive than } B) \Leftrightarrow level(A) > level(B) \tag{5.20a}$$

$$(A,\, B \text{ are of equal aggressiveness}) \Leftrightarrow level(A) = level(B). \tag{5.20b}$$

Therefore it is assumed that a change in RULE to a rule base configuration with a lower level results in lower drive cycle completion time O_T and higher energy use O_E. A change in RULE to a rule base configuration with identical level is assumed to produce unknown, but relatively small, changes in O_E and O_T.

Example 5.3.11. With reference to Figure 5.19, the following statements can be made about node numbers 1, 11, 12, 20, and 84. Node numbers 1, 11, 12, 20, and

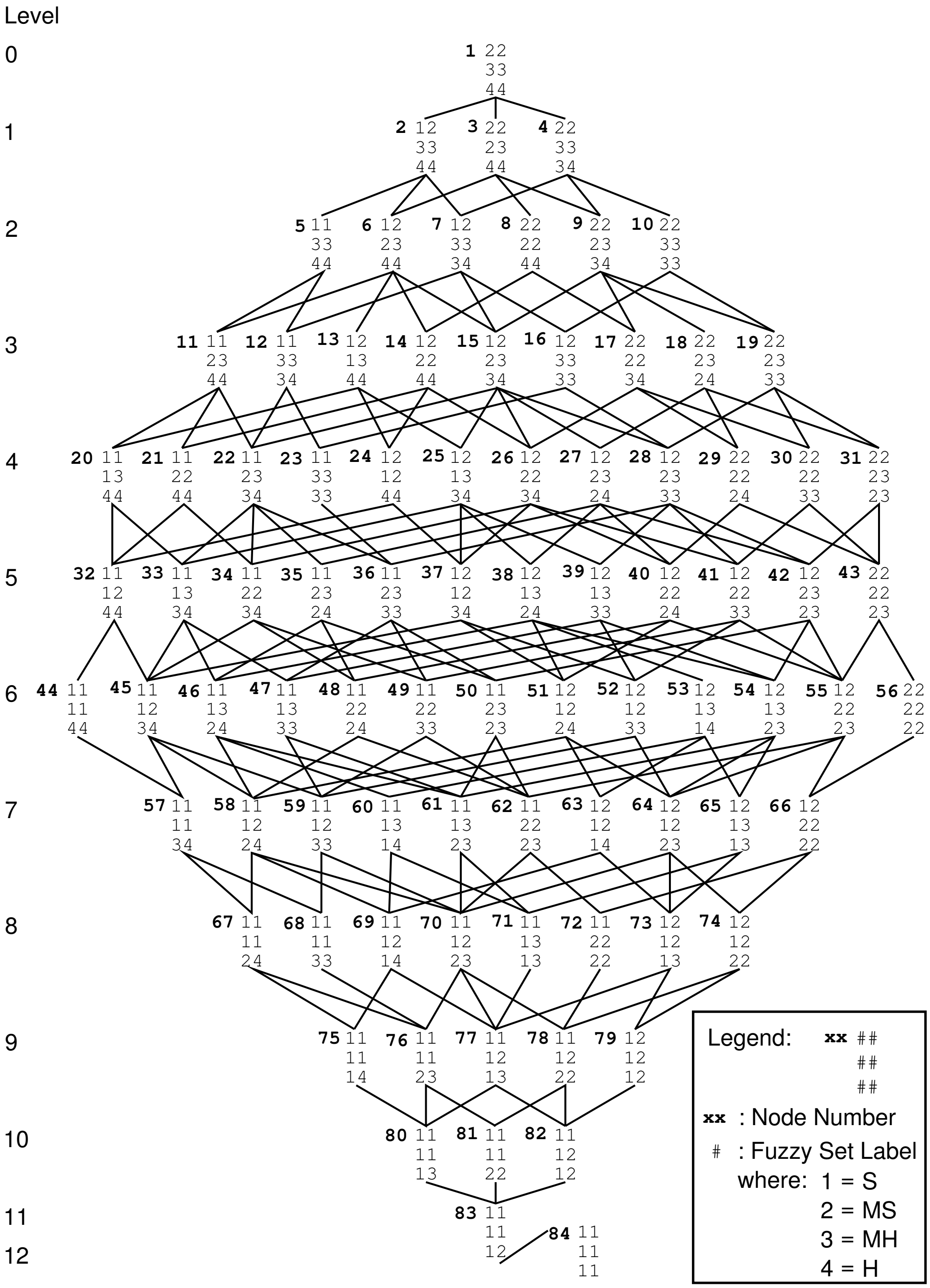


Figure 5.19: Poset by Delta Inclusion of Traction Control Rule Base

84 correspond, respectively, to rule base configurations:

$$\begin{bmatrix} 2 & 2 \\ 3 & 3 \\ 4 & 4 \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ 2 & 3 \\ 4 & 4 \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ 3 & 3 \\ 3 & 4 \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ 1 & 3 \\ 4 & 4 \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{bmatrix}.$$

Node number 1 (with lowest level 0) represents the most aggressive rule base configuration and contains all others by delta inclusion. Node number 84 (with highest level 12) represents the least aggressive rule base configuration and is contained by all others by delta inclusion. Node number 11 is a more aggressive rule base than node number 20 since node number 11 contains node number 20 by delta inclusion. Although node number 11 and node number 12 are not comparable by delta inclusion, it is assumed that they represent rule base configurations of equal aggressiveness since they are of equal level.

Consideration of the level of a rule base and its impact on optimization objectives motivates the definition of crossover and mutation operations that operate on the level of a rule base. Since the level of a rule base constrains the possible entries of $\mathcal{C}_i$ in variable RULE, the approach is taken to encode the rule base conclusions by a variable NODE that references a member of delta set X rather than by six independent integers. As shown in Figure 5.14, the variable NODE is represented by the single-gene chromosome X_{NODE} and constrained to poset node numbers that reference the 84 possible rule base configurations shown in the Hasse diagram of Figure 5.19 such that:

$$\text{NODE} \in \{1, \ldots, 84\}. \tag{5.21}$$

Although the node number reference in NODE is implemented using an integer, the actual data type of NODE is the delta set X and specialized crossover and mutation operations must be defined to affect changes on NODE based on rule base level.

Since a direct mapping between node numbers and members of delta set X exists, the definitions of the meet, join, and level operations over members of delta set X are extended, for convenience, to include node number references with the implicit understanding that node numbers are dereferenced prior to operator application.

Crossover

According to the building block hypothesis described in Section 5.1.1, an important consideration in the definition of a crossover operation is the preservation of building block information encoded within parent members of the operation. The presence of building blocks in a candidate solution increases candidate fitness and the influence confirms that building blocks have a fundamental impact on optimization objectives. Consequently, the level of the traction control rule base encoded in X_{NODE} can be considered as a building block because (5.20) establishes a relationship between the level of the rule base and its impact on optimization objectives O_E and O_T. Therefore, definition of the crossover operation focuses on the preservation of the level information encoded in parent rule base configurations.

Given two parent rule base configurations referenced by A_{NODE} and B_{NODE}, the crossover operation produces an offspring C_{NODE} using a two step process. First, it is randomly decided (with equal probability) to take either the meet $(A \wedge B)$ or join $(A \vee B)$ of the parents to produce an intermediate rule base configuration T_{NODE}. Additional processing on T_{NODE} is then performed to compensate for bias changes in level due to the application of the meet or join to produce C_{NODE}.

Taking the meet of the parents A_{NODE} and B_{NODE} tends to increase the level of the resulting rule base T_{NODE} by (5.11). This bias towards an increase in level is undesirable since repeated application of a rule base crossover operation will cause the average level of rule base configurations within the population to increase. To avoid this, additional processing is applied to restore the level of T_{NODE} to the minimum level of A_{NODE} and B_{NODE} to produce C_{NODE}. The additional processing involves traversing the Hasse diagram such that both (5.22a) and (5.22b) are satisfied by C_{NODE} as demonstrated in Example 5.3.12.

Similarly, taking the join of two rule base configurations causes an undesirable decrease in the average level of rule base definitions within the population by (5.12). Similar processing is applied to restore the level of T_{NODE} to the maximum level of A_{RULE} and B_{RULE}. The generated offspring C_{NODE}, as derived from a join operation, satisfies both (5.23a) and (5.23b). An example of a crossover operation derived from the join of parents is provided in Example 5.3.13. The meet and join crossover operations maintain the feasibility of generated rule base configurations since generated offspring are members of delta set X.

$$\begin{aligned}(A_{\text{NODE}} \wedge B_{\text{NODE}}) &\Subset C_{\text{NODE}} && (5.22a)\\ level(C_{\text{NODE}}) &= \min[level(A_{\text{NODE}}, level(B_{\text{NODE}})] && (5.22b)\end{aligned}$$

$$\begin{aligned}C_{\text{NODE}} &\Subset (A_{\text{NODE}} \vee B_{\text{NODE}}) && (5.23a)\\ level(C_{\text{NODE}}) &= \max[level(A_{\text{NODE}}), level(B_{\text{NODE}})] && (5.23b)\end{aligned}$$

Example 5.3.12. Consider the crossover operation shown in the rule base poset fragment of Figure 5.20 with parent chromosomes $A_{\text{NODE}} = 6$ and $B_{\text{NODE}} = 7$. It is assumed that the random decision is made to perform a crossover operation by taking the meet of the parents. The meet of the parent rule base configurations results in an intermediate rule base configuration $T_{\text{NODE}} = 15 = A_{\text{NODE}} \wedge B_{\text{NODE}}$. Additional processing is applied on T_{NODE} to return an offspring C_{NODE} of level 2 since $level(T_{\text{NODE}}) = 3$ is greater than the minimum of $level(A_{\text{NODE}}) = 2$ and $level(B_{\text{NODE}}) = 2$. The bold lines in Figure 5.20 indicate that possible rule base configurations of level 2 that contain T_{NODE} consist of node number 9 in addition to parent node numbers 6 and 7. It is assumed in this example that random selection among possible rule base configurations of level 2 that contain T_{NODE} results in $C_{\text{NODE}} = 9$.

Example 5.3.13. Consider the crossover operation shown in Figure 5.21 that is based on taking the join of parents $A_{\text{NODE}} = 6$ and $B_{\text{NODE}} = 12$. The join of the parents results in an intermediate rule base configuration $T_{\text{NODE}} = 2 = A_{\text{NODE}} \vee B_{\text{NODE}}$. Since $level(T_{\text{NODE}}) = 1$, additional processing is applied on T_{NODE} to return an offspring C_{NODE} of level 3 corresponding to the maximum of $level(A_{\text{NODE}}) = 2$ and $level(B_{\text{NODE}}) = 3$. The level of T_{NODE} is increased iteratively by first randomly selecting a rule base configuration T'_{NODE} of level 2 that is contained by T_{NODE} and then by randomly selecting C_{NODE} of level 3 that is contained by T'_{NODE}. The bold

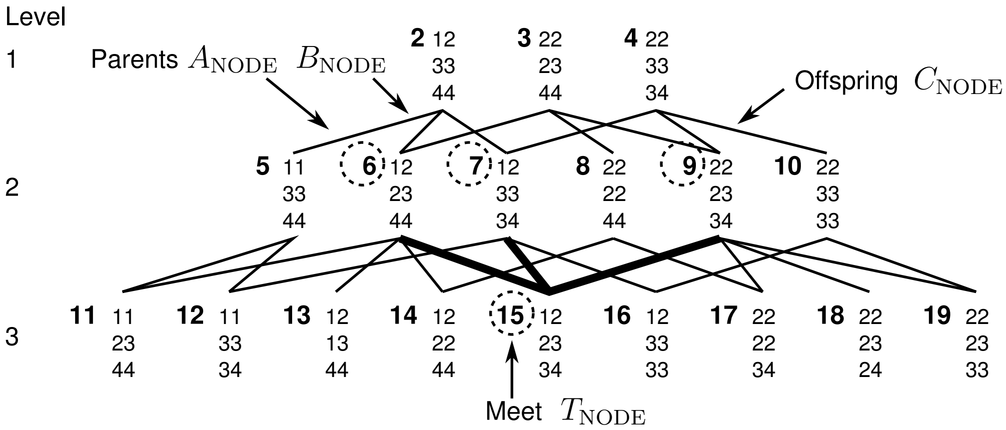


Figure 5.20: Example of Traction Control Rule Base Crossover by Meet

lines in Figure 5.21 indicate possible paths that may be taken from T_{NODE} to a rule base configuration of level 3. Possible assignments to C_{NODE} consist of node numbers 11 to 16. The example assumes that T'_{NODE} is assigned either node number 6 or 7 leading to $C_{\text{NODE}} = 15$.

Mutation

For a given chromosome C_{NODE}, mutation is defined to create a new chromosome D_{NODE} that is randomly chosen from a subset of the delta set X consisting of rule base configurations m with immediately higher or lower level than C_{NODE}:

$$D_{\text{NODE}} \in \{level(m) = level(C_{\text{NODE}}) + 1, level(m) = level(C_{\text{NODE}}) - 1, m \in X\}. \tag{5.24}$$

Mutation allows the level (and aggressiveness) of the candidate solution to be altered and permits diversity in exploring rule base configurations that are not derived from the meet or join of the parents of the crossover operation.

Example 5.3.14. Consider the mutation of chromosome $C_{\text{NODE}} = 10$ in the rule base poset fragment of Figure 5.22. Since $level(C_{\text{NODE}}) = 2$, mutation is restricted to rule base configurations with a level of 1 or 3. The mutated chromosome D_{NODE} is therefore assigned a random node number within the subset of node numbers {2,

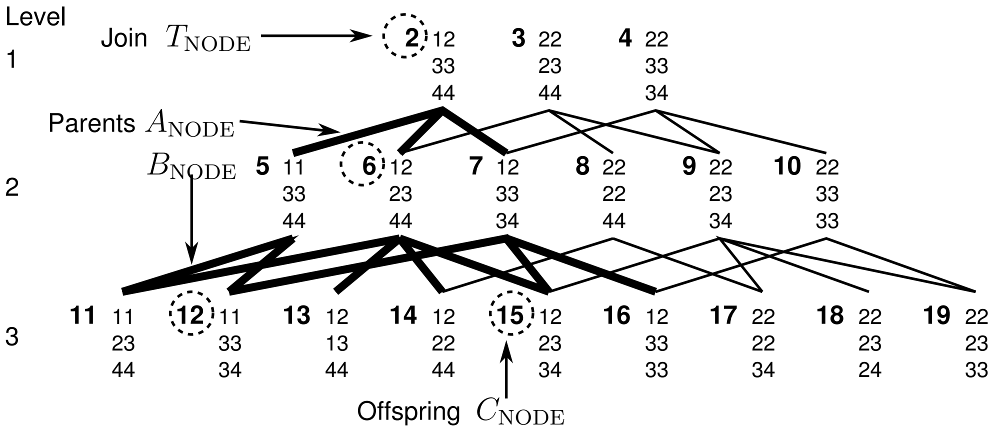


Figure 5.21: Example of Traction Control Rule Base Crossover by Join

3, 4, 11, 12, 13, 14, 15, 16, 17, 18, 19}. In Figure 5.22, mutation of C_{NODE} results in the random assignment $D_{\text{NODE}} = 11$.

The definition of the mutation operation maintains the feasibility of a perturbed rule base since all mutations results in rule base configurations within delta set X. Together, the definitions of the crossover and the mutation operations on the novel rule base encoding enables the use of the X_{NODE} chromosome within the mixed encoding, multi-chromosomal GA of Section 5.2.

Encoding Analysis

A discussion is provided on the search space reduction and optimization constraint elimination associated with the use of the novel encoding. The dependence of the specialized crossover and mutation operations on a precomputed Hasse diagram is examined and presents a potential weaknesses in the encoding.

The direct rule base encoding of the six rule base conclusions $\mathcal{C}_i$ from RULE as independent integers produces an unconstrained search space consisting of 4096 possible rule base configurations. The application of rule base constraints (5.14) and (5.15) reduces the feasible search space by approximately 98% to include 84 possible rule base configurations (as enumerated by the node numbers in Figure 5.19). The

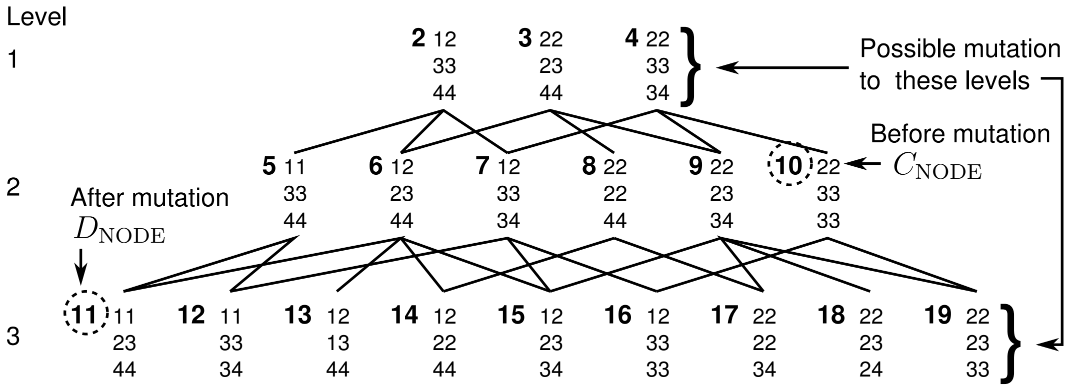


Figure 5.22: Example of Traction Control Rule Base Mutation

constraints introduce a large infeasible search space consisting of the remaining 4012 possible rule base configurations.

Under a direct encoding, the independent modification of the integer-based representation is likely to produce constraint violations resulting in infeasible solutions. A GA design that incorporates a direct rule base encoding must therefore consider the handling of infeasible rule base configurations. By incorporating the rule base constraints within the definition of delta set X and defining specialized crossover and mutation operations that ensure feasible candidate solutions, the novel encoding of the rule base effectively restates the constrained optimization of the rule base in terms of an unconstrained optimization over an identical feasible search space. Compared to the direct encoding, the use of the novel encoding avoids the computational overhead associated with handling infeasible rule base configurations while exploring an otherwise highly constrained feasible search space.

Although the numerical results in Section 6.4 demonstrate successful application of the novel rule base encoding to the current traction control optimization problem, the dependence of the crossover and mutation operations on a precomputed Hasse diagram may limit application of the encoding to larger problems. To improve computational efficiency, the crossover and mutation operations rely on a precomputed Hasse diagram (i.e. Figure 5.19) to determine the level of a rule base configuration as well as the relationship by delta inclusion between two given rule base configurations. Specifically, the crossover operation uses an iterative algorithm to traverse the precomputed Hasse diagram to compensate for changes in level associate with taking the meet and join of operation parents. The mutation operation also uses an iterative algorithm to search the precomputed Hasse diagram for a subset of rule base configurations with immediately higher or lower level than its input argument.

Dependence on a precomputed Hasse diagram creates a potential problem in computing and storing the diagram if the novel encoding is applied to larger optimization problems. Consider, for example, application of the encoding to the moderately sized, complete traction controller rule base (Figure 3.15a) consisting of 16 rule base entries with each rule base conclusion assuming any of 4 possible values. Exponential growth with respect to the number of rule base conclusions produces an unconstrained search space with approximately 4.2 billion ($= 4^{16}$) entries. Assuming that a similar 98% reduction in search space can be obtained by application of the encoding, the delta set of the full rule base may also exhibit exponential growth and contain 85 million possible rule base configurations.

Rather than relying on the extrapolation of results obtained through enumeration, an analytic expression should be derived to accurately calculate the number of members in a delta set. In the optimization of large problems, such an expression may also support the decision to redesign the crossover and mutation operations to dynamically compute required portions of the Hasse diagram. The analytic calculation of the size of a delta set and the design of crossover and mutation operations that do not rely on a precomputed Hasse diagram are open for future research.

Chapter 6

Numerical Results

Numerical results are presented and analyzed for both unoptimized and optimizing simulation studies of the traction control system. The numerical precision of all simulation results is discussed using a simulation convergence study to establish numerical solver settings. A set of preliminary open and closed loop simulations is performed to demonstrate: the performance limitations associated with open loop control of the vehicle model summarized in Section 2.4, and the benefits gained in vehicle performance and energy economy objectives made possible by the fuzzy logic traction controller defined in Section 3.3. Both plant and controller are then optimized with respect to these system-level objectives according to the optimization problem summarized in Section 4.2.4 by application of the genetic algorithm developed in Chapter 5. The numerical values of system parameters given in Appendix A are assumed in all simulations.

6.1 Numerical Solver Settings

The system model is implemented as a set of software m-files written for compatibility between the Matlab [56] and Octave [57] numerical simulation environments. Individual trial simulations such as those discussed in Section 6.2 and Section 6.3 are executed in the Matlab environment due to its powerful graphical user interface allowing interactive data analysis. The large scale simulations of Section 6.4 are executed non-interactively in Octave over the high performance distributed Shared Hierarchical Academic Research Computing Network (SHARCNET) [58] with data post analysis in Matlab.

Due to emphasis on results of the simulation study in this thesis and the dual computational platforms used to execute the source code, a simulation convergence study involving the analysis of the effect of solver parameters on numerical output was conducted and summarized in Appendix D. The closed loop traction control system configured with the performance tire (Tire 0) and nominal controller parameters from Table A.4 was simulated with a series of solver settings and the numerical outputs of the energy objective O_E and time objective O_T state variables were recorded. Solver settings, used throughout this thesis, were chosen for convergence to 5 significant figures of accuracy with numerical agreement between the Matlab and Octave solvers (Table D.1).

Additional information such as Matlab and Octave software revision numbers and detailed numerical results are provided in Appendix D.

6.2 Open Loop System Evaluation

Performance characteristics of the uncontrolled vehicle system with the performance tire (Tire 0) are investigated in open loop simulations where $T_a = T_{ref}$ (replacing the fuzzy logic controller according to (2.40)). Two open loop system configurations, as described in Table 6.1, are considered with different T_{ref} to demonstrate open loop system instability (system no-traction-control-system-unstable nTCSu) as well as marginal stability (system nTCS-stable nTCSs).

The following analysis of the drive cycle response of nTCSu and nTCSs demonstrate, respectively: the nature of the open loop instability that occurs when tire performance limits are exceeded, and the degradation in system performance associated with designing a prefixed open loop torque limitation to ensure stability during worst-case driving conditions (wet/icy road).

Table 6.1: Open Loop System Configurations

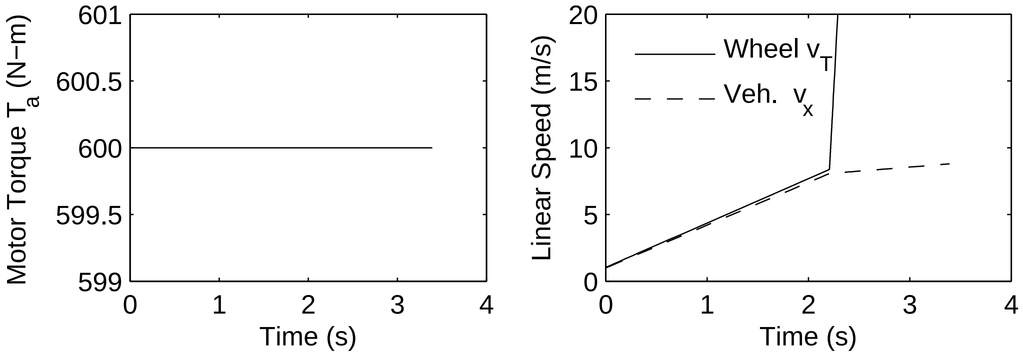
Config.	$T_a = T_{ref}$ (N m)
nTCSu	600
nTCSs	196.015

6.2.1 Open Loop Instability of nTCSu

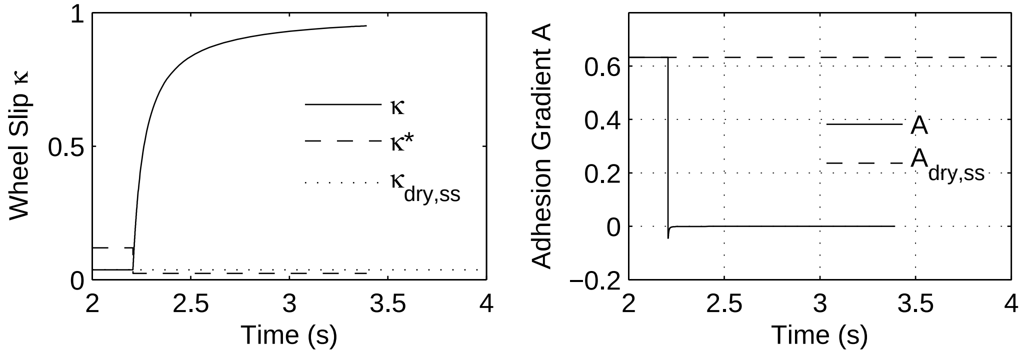
The drive cycle response of nTCSu is shown in Figure 6.1 where the applied torque T_a (Figure 6.1a) has been selected such that system performance is torque limited on the dry road surface and friction limited on the wet road surface. The evolution of the vehicle speed v_x and wheel speed Ω is plotted in Figure 6.1b where Ω is represented by the **linear wheel speed** $v_T = r\Omega$. The onset of instability occurs at the point of the road transition and can be observed by the rapid increase in wheel speed with respect to vehicle speed.

The block diagram of the tire and vehicle dynamics in Figure 6.2 shows that the state variables v_x and Ω are nonlinearly dependent on the feedback of tire force generation F_x. Hence an analysis of wheel slip κ is key to understanding the nature of the open loop system dynamics. Although system nonlinearity precludes a closed form description of state variable dynamics of v_x and Ω, their variation may be indirectly described by observing that κ attains a steady-state value on the dry road and approaches a limit of 1 due to rapid increase of Ω on the wet road after the onset of instability caused by the road surface transition.

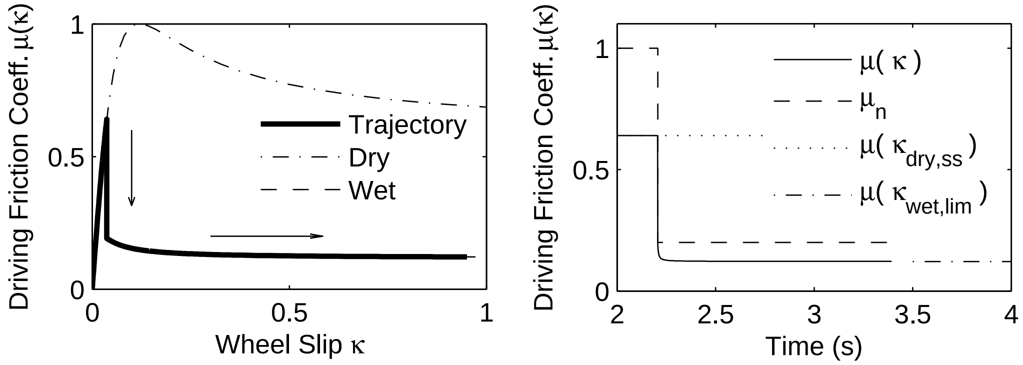
The following analytical derivation of the resulting steady-state and limit conditions provides insight into open loop system dynamics and allows verification of a correct implementation of the dynamic equations within the simulation. Prior to the road transition, application of torque causes the wheel to accelerate with associated generation of wheel slip κ. The wheel slip in turn produces force F_x within the tire-road contact patch to accelerate the vehicle. Assuming stable operation, vehicle acceleration tends to reduce slip and for a constant T_a, a steady-state operating point is reached with proportional changes in wheel and vehicle speed producing



(a) Applied Torque vs Time (b) Linear Velocity vs Time



(c) Wheel Slip vs Time (d) Adhesion Gradient vs Time



(e) $\mu - \kappa$ Trajectory (f) Driving Friction Coefficient vs Time

Figure 6.1: Unstable Open Loop System Response of nTCSu, with Comparison to Analytical Steady-State Solutions from Table 6.2

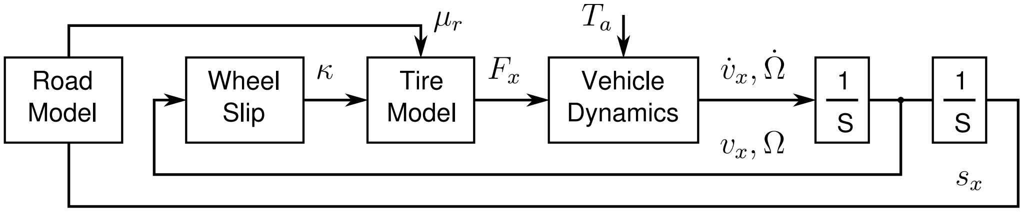


Figure 6.2: Tire and Vehicle Dynamics

constant κ_{ss} and F_{xss}. For a constant T_a, steady-state operation is reached when:

$$\frac{d\kappa}{dt} = \frac{d}{dt}\left(1 - \frac{v_x}{r\,\Omega}\right) = \frac{-\dot{v}_x\Omega + v_x\dot{\Omega}}{r\,\Omega^2} = 0. \tag{6.1}$$

Assuming $\Omega \neq 0$, steady-state operation implies:

$$v_x\dot{\Omega} = \dot{v}_x\Omega. \tag{6.2}$$

where v_x may by written in terms of κ_{ss} by rearranging (2.35):

$$v_x = r\,\Omega(1 - \kappa_{ss}). \tag{6.3}$$

Substitution of (6.3), (2.29), and (2.30) into (6.2) results in elimination of v_x and cancellation of Ω:

$$r(1 - \kappa_{ss})\frac{T_a - T_y - rF_{xss}}{J} = \frac{F_{xss}}{m}. \tag{6.4}$$

Since F_x is a nonlinear function of κ, the steady-state $\kappa_{dry,ss}$ of nTCSu was solved by numerical root finding using the **fsolve** function in the Maple [59] script in Appendix E. Having obtained $\kappa_{dry,ss}$, steady-state $A_{dry,ss}$ and $\mu(\kappa_{dry,ss})$ are available by application of (2.38) and (2.20) with known road condition $\mu_r = \mu_{dry}$.

After the road transition, the rapid increase in Ω with respect to v_x causes κ to approach unity. Substitution of $\kappa_{wet,lim} = 1$ into (2.38) and (2.20) with known road condition $\mu_r = \mu_{wet}$ gives $A_{wet,lim}$ and $\mu(\kappa_{wet,lim})$, respectively.

The results of the steady-state and limit analysis are summarized in Table 6.2 and plotted for comparison with simulation results in Figure 6.1d and Figure 6.1f. The agreement between results obtained through numerical integration (simulation)

Table 6.2: Steady State and Limit Conditions of nTCSu

Dry Road Steady States	Wet Road Limits
$\kappa_{dry,ss} = 0.0371114$ *	$\kappa_{wet,lim} = 1$
$A_{dry,ss} = 0.632064$ *	$A_{wet,lim} = -0.203137e - 3$ *
$\mu(\kappa_{dry,ss}) = 0.640351$ *	$\mu(\kappa_{wet,lim}) = 0.121659$ *

* Calculation Details in Appendix E

and algebraic steady-state analysis using independent software packages provides some confidence in the correctness of the implementation of tire and vehicle models.

In Figure 6.1c and Figure 6.1f, the time variation in κ and driving friction coefficient $\mu(\kappa)$ are compared, respectively, with the limits of tire performance determined by the road surface consisting of optimal slip κ^* producing peak driving friction coefficient μ_{peak}. The optimal slip κ^* is given by (2.24) and Figure 6.1c shows that the available torque limits tire performance on the dry road since $\kappa_{dry,ss} < \kappa^*$. On the wet road, a reduction in κ^* below $\kappa_{dry,ss}$ triggers entry into the region of unstable tire operation with $\kappa > \kappa^*$ and $\kappa \to 1$ (see Figure 3.2). With the assumption derived in (2.25), the onset of instability can be associated with a sign change from positive to negative adhesion gradient A as shown in Figure 6.1d. After the onset of instability, A decreases to a minimum negative value before increasing towards zero as $\kappa \to 1$ according to the adhesion characteristic shown in Figure 2.10.

The simulated driving friction coefficient $\mu(\kappa)$ is compared with its theoretical maximum $\mu_{peak} = \mu_n$ in Figure 6.1f. The maximization of driving friction coefficient across road surfaces is desirable since it is directly related to longitudinal tire force F_x and therefore affects vehicle performance. The peak driving friction coefficient μ_{peak} corresponding with optimal κ^* is defined as the amplitude of the sine function of (2.20):

$$\mu_{peak} = \frac{\mu_r D_{xo}}{F_{zo}} = \mu_r \mu_o = \mu_n \tag{6.5}$$

where identity with the coefficient of friction μ_n is shown by recalling $D_{xo} = \mu_o F_{zo}$ (2.14) and $\mu_r = \mu_n/\mu_o$ (2.19). As μ_{peak} represents a maximum, the identity therefore shows that tire-road force generation is limited by road conditions and

$$\mu(\kappa) \leq \mu_n. \tag{6.6}$$

The comparison of $\mu(\kappa)$ with μ_n in Figure 6.1f shows that (6.6) is satisfied. In conjunction with knowledge of wheel slip, Figure 6.1f indicates that the sub-maximal $\mu(\kappa_{dry,ss})$ on the dry road is due to a limitation in applied torque causing suboptimal wheel slip.

Upon transition to the wet road, a general reduction of $\mu(\kappa)$ and system performance is expected due to the physical limitations imposed by reduction in μ_r. However, in nTCSu, performance is further degraded due to excessive wheel slip beyond κ^* causing a reduction in $\mu(\kappa)$ towards $\mu(\kappa_{wet,lim})$. Both effects contribute to the reduction in vehicle acceleration observed in Figure 6.1b.

The $\kappa - \mu$ plot of Figure 6.1e summarizes the impact of a road transition on tire performance resulting in loss of traction and open loop instability. The normalized tire characteristics of Figure 2.10 for Tire 0 used by nTCSu have been overlaid and emphasize the impact of tire characteristics on performance. Initial application of torque on the dry road causes an increase in κ and $\mu(\kappa)$ following the dry road characteristics of the tire. Steady-state operation is attained at $(\kappa, \mu) = (\kappa_{dry,ss}, \mu(\kappa_{dry,ss}))$. The road transition causes a change from dry to wet tire characteristics with associated reduction in $\mu(\kappa)$ and vehicle performance. The reduction in $\mu(\kappa)$ corresponds to a proportional reduction in F_x which, for a constant T_a, causes an increase in $\dot{\Omega}$ and hence κ according to (2.29) and (2.35), respectively. Unstable tire operation results since the wet tire characteristics dictate that an increase in κ from $\kappa_{dry,ss}$ on the wet road causes further decrease of F_x. Therefore, the open loop system tends to unity wheel slip with corresponding performance decrease towards $\mu(\kappa_{wet,lim} = 1)$.

6.2.2 Limited-Torque Open Loop Control of nTCSs

In an open loop system without traction control, stability may be guaranteed by designing a prefixed torque limitation in anticipation of operation in worst case situations (wet/icy roads). In nTCSs, the design problem is to find the maximum torque permissible for stable operation on the wet road where system nTCSu failed.

As previously shown, constant torque application results in a steady-state condition given by (6.4). For marginal stability on the wet surface, it is desired that the steady-state conditions are set such that $\kappa_{wet,ss} = \kappa^*$ where κ^* is given by (2.24) with $\mu_r = \mu_{wet}$. Operation at κ^* implies that $F_{xss,wet}$ is equal to the peak of the tire force characteristics of (2.37):

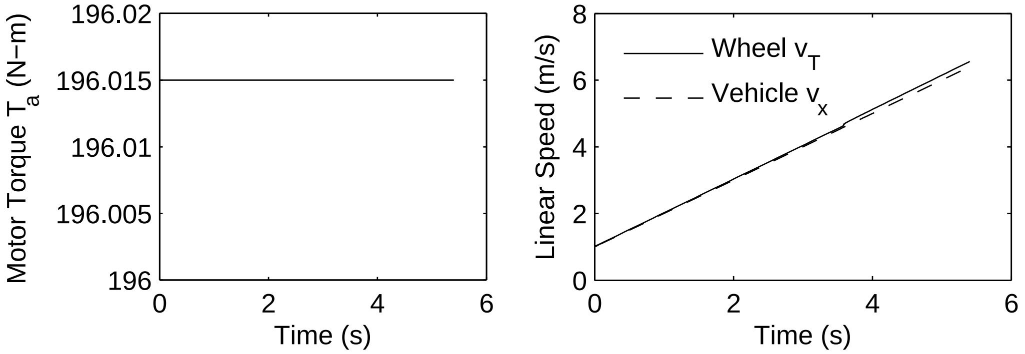
$$F_{xss,wet} = \mu_{wet} D_{xo}. \tag{6.7}$$

The maximum torque limit is calculated by substituting (2.24) and (6.7) into (6.4) and solving for T_a. The resulting torque limit is listed for nTCSs in Table 6.1.

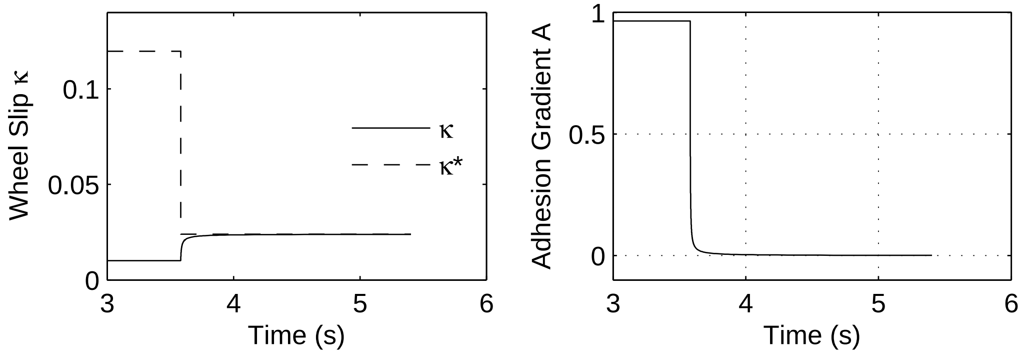
The simulated drive cycle response of nTCSs is shown in Figure 6.3. The value of the applied torque in Figure 6.3a is set smaller than the calculated maximum by truncation after the third decimal point to ensure marginal stability. System stability throughout the drive cycle is shown in Figure 6.3b by the close tracking of vehicle and wheel speeds. After the road transition, the figure shows that wheel speed exhibits a small but important offset from vehicle speed which translates into an increase in wheel slip.

The increase in slip when transitioning to the wet road is shown in Figure 6.3c. As designed, the prefixed torque limit allows κ to approach κ^* on the wet road. The marginal stability of the system on the wet road is further demonstrated by the vanishing adhesion gradient shown in Figure 6.3d. Any increase in T_a will trigger an unstable response similar to nTCSu. While on the dry road, the near unity adhesion gradient is due to small wheel slip resulting in tire operation near a region characterized by the initial stiffness $C_{F\kappa o}$ of the tire from which A is derived (2.38).

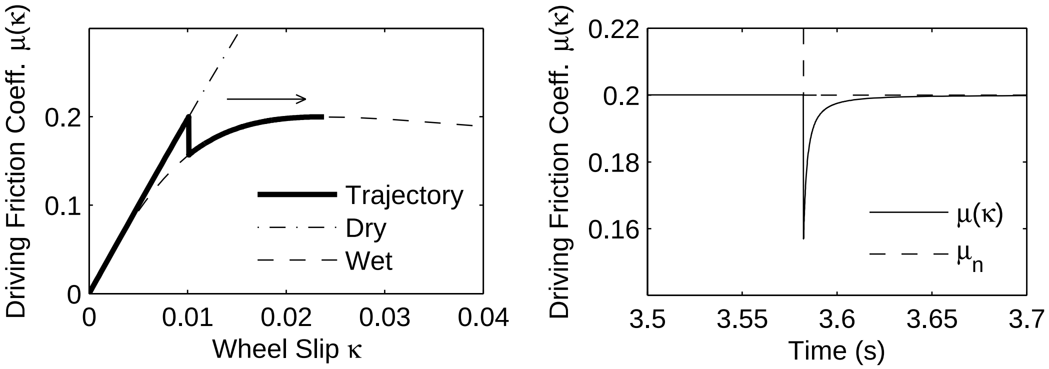
The impact on vehicle acceleration, due to limiting applied torque to worst case scenarios, is shown in Figure 6.3f where $\mu(\kappa)$ is restricted to $\mu(\kappa_{wet,ss})$ even on the dry road. Thus open loop torque limitation obtains maximum tire force generation on the wet road by incurring a large sacrifice in force generation on the dry road. The momentary decrease in $\mu(\kappa)$ at the road transition can be explained by considering the $\kappa{-}\mu$ trajectory shown in Figure 6.3e. Initial application of torque on the dry road causes an increase in wheel slip and force generation according to



(a) Applied Torque vs Time (b) Linear Velocity vs Time



(c) Wheel Slip vs Time (d) Adhesion Gradient vs Time



(e) $\mu - \kappa$ Trajectory (f) Driving Friction Coefficient vs Time

Figure 6.3: Marginally Stable Open Loop System Response of nTCSs

dry road characteristics of the tire until steady-state conditions are reached. A change in road condition causes an initial decrease in $\mu(\kappa)$ and an increase in slip. However, unlike nTCSu, $\kappa_{dry,ss}$ remains less that κ^* during the transition and therefore an increase in driving force and $\mu(\kappa)$ occurs for an increase in slip. The driving force causes vehicle acceleration which tends to reduce slip until the system reaches a new steady-state condition on the wet road. In nTCSs, the applied torque has been designed such that the steady-state condition on wet road occurs at the limits of tire performance where the system is marginally stable.

6.3 Closed Loop System Evaluation

To maximize vehicle performance, a traction control system actively monitors road conditions and dynamically adjusts torque application to maintain system stability despite changes in road conditions. Analysis of the drive cycle response of the adhesion gradient-based traction controller provides additional insight into the operating principles of traction control. Comparison of closed and open loop drive cycle response demonstrates the overall benefits of traction control in performance and energy economy. A parametric study is used to assess the impact of variations in traction control system design on performance and energy economy.

Closed loop systems considered in this section are named and described in Table 6.3 where, in particular, TCSo corresponds to a nominal traction control system with the performance tire (Tire 0) and nominal controller parameters from [12] as listed in Appendix A. The column A-PS indicates the location of the Positive-Small membership function of the adhesion gradient input to the fuzzy logic traction controller. Columns $\mathcal{C}_1$ through $\mathcal{C}_6$ describe alternative controller rule bases. The TIRE column indicates the use of either a high performance tire (Tire 0) or a low rolling resistance tire (Tire 1). Systems TCSA-TCSE are variations of TCSo used in the parametric study of Section 6.3.3.

Table 6.3: Closed Loop Traction Control System Configurations

	Fuzzy Logic Controller							Plant
Config.	A-PS	$\mathcal{C}_1$	$\mathcal{C}_2$	$\mathcal{C}_3$	$\mathcal{C}_4$	$\mathcal{C}_5$	$\mathcal{C}_6$	TIRE
TCSo	0.5	S	MS	MS	MH	MS	MH	0
TCSA	-	-	-	-	-	-	-	1
TCSB	0.2	-	-	-	-	-	-	-
TCSC	0.8	-	-	-	-	-	-	-
TCSD	-	MS	MS	MH	MH	H	H	-
TCSE	-	S	S	S	S	S	S	-

'-' Indicates same as nominal configuration TCSo

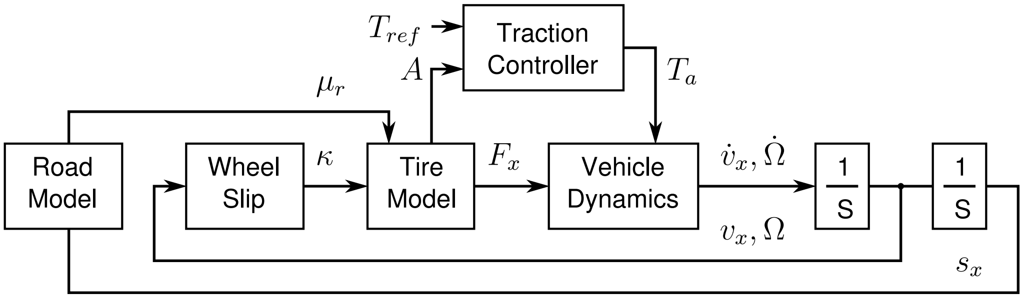


Figure 6.4: Closed Loop Traction Control System Dynamics

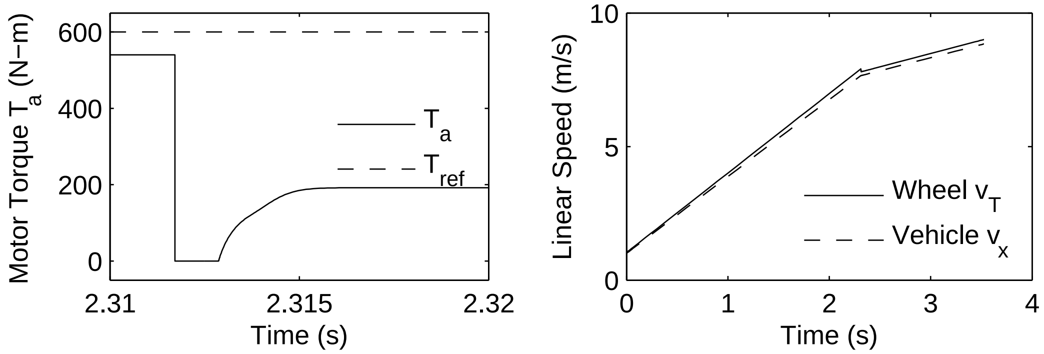
6.3.1 Closed Loop Response of TCSo

Of particular interest in the analysis of closed loop response is the behavior of T_a. As shown in Figure 6.4, T_a is generated by an adhesion gradient-based fuzzy logic traction controller that is capable of affecting tire and vehicle dynamics independently of driver torque request T_{ref}. Control action taken is key to maintaining system stability when direct application of the requested torque would produce instability.

The stabilizing effects of control are demonstrated in the drive cycle response of TCSo (Figure 6.5). Figure 6.5a shows that controller action causes T_a to vary despite a constant T_{ref}. The variation allows stability to be maintained on the wet road as indicated by the close tracking of vehicle and linear wheel speeds shown in Figure 6.5b. In contrast, the direct application of the same amount of torque in the open loop system nTCSu causes instability.

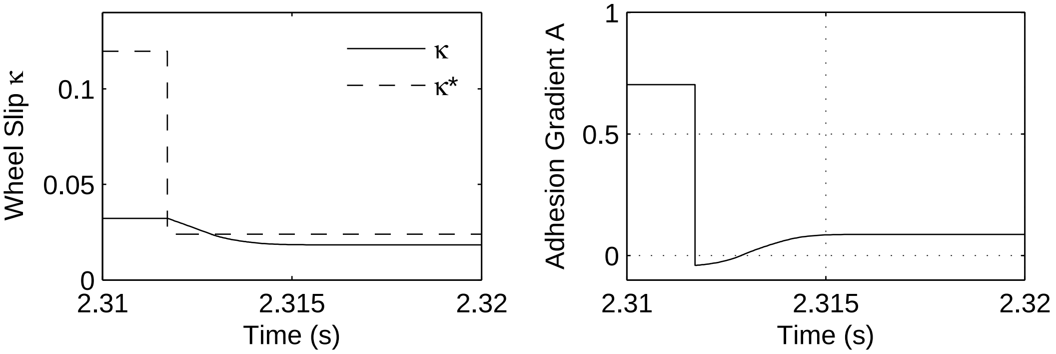
The wheel slip limiting effect of traction control is observed in Figure 6.5c where the road transition causes κ^* to drop below κ to produce an unstable situation similar to unstable nTCSu in Figure 6.1c. Reduction in T_a at this point allows Ω to decrease, and slip to decrease below κ^* to re-enter the stable region of tire operation. Once stability is regained, torque is reapplied in consideration of the new limits of performance.

In an adhesion gradient-based traction controller, tire-road performance limits are determined indirectly through observation of a vanishing adhesion gradient and the controller acts to avoid negative adhesion gradients. In Figure 6.5d, the adhesion gradient A remains negative while $\kappa > \kappa^*$. The plot of the trajectory of T_a against the surface defined by the nonlinear mapping of the fuzzy logic traction controller in Figure 6.6 provides additional understanding of controller action. Initial conditions of the system (Location 1 in the figure) are such that wheel slip is zero, $A = 1$, and therefore $T_a = T_{ref}$. Application of torque causes an increase in wheel slip and a decrease of adhesion gradient according to (2.38). This causes a reduction in T_a on the dry road, as dictated by the surface plot of the fuzzy logic controller, that tends to decrease wheel slip resulting in steady-state torque application on the dry road $T_{a,dry,ss}$ (Location 2). The torque reduction in response to wheel slip causes $T_a < T_{ref}$ while on the dry road as shown in Figure 6.5a. The road transition causes a jump to negative adhesion gradient (Location 3) where $\kappa > \kappa^*$ and the surface plot dictates an application of zero torque causing wheel deceleration. As the wheel decelerates, κ decreases towards κ^* and A increases towards zero. Stability is regained once $A > 0$ and the surface plot dictates a gradual increase in torque. The torque increase counters the deceleration of the wheel and results in steady-state torque application on the wet road $T_{a,wet,ss}$ (Location 4). The surface plot trajectory in Figure 6.6 also emphasizes that for the selected drive cycle, controller action is limited to a line defined by the intersection of the surface with the plane given by constant T_{ref}/T_{max}.



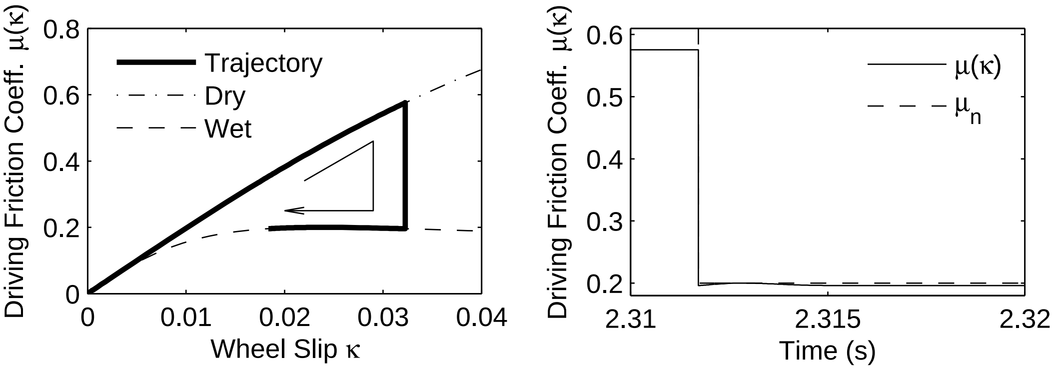
(a) Applied Torque vs Time

(b) Linear Velocity vs Time



(c) Wheel Slip vs Time

(d) Adhesion Gradient vs Time



(e) $\mu - \kappa$ Trajectory

(f) Driving Friction Coefficient vs Time

Figure 6.5: Closed Loop Fuzzy Logic Traction Control (Nominal Parameters)

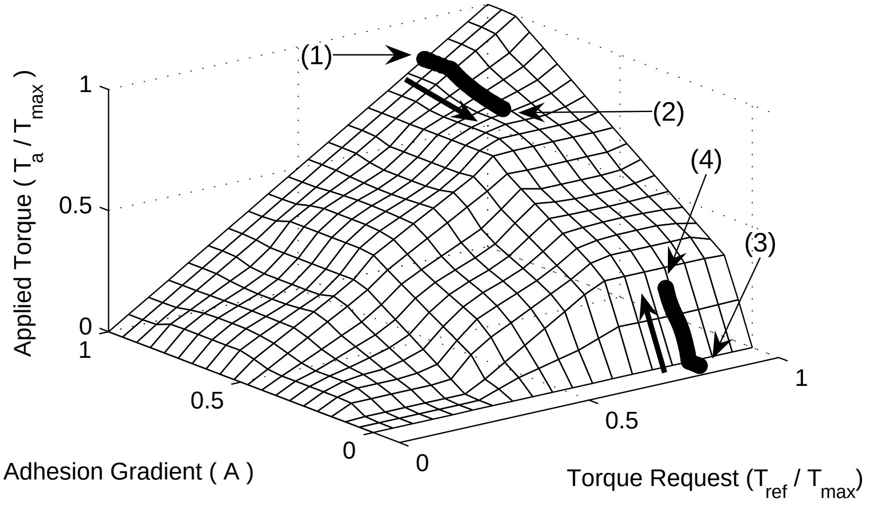


Figure 6.6: Surface Plot Trajectory of TCSo

Ideally, maximum tire performance is obtained if steady-state torque application occurs at $A = 0$ with corresponding operation at optimal slip $\kappa = \kappa^*$. This state is not achievable on the currently defined surface plot and hence the associated controller is non-optimal with respect to tire performance. In general, the class of adhesion gradient traction controllers considered in this thesis do not permit optimal tire performance at $\kappa = \kappa^*$ since the definition of the adhesion gradient controller assumes no knowledge of tire characteristics other than the current adhesion gradient operating point. Thus an adhesion gradient controller cannot be defined to apply a steady-state torque at $A = 0$ since this would require the steady-state calculation involving a tire force generation model as described in Section 6.2.2. For future work, it is proposed to investigate the use of an integral controller in conjunction with an adhesion gradient controller to attain optimal tire performance by allowing increases in T_a, while $A > 0$, beyond those prescribed by the adhesion gradient surface plot and up to the amount T_{ref} requested by the driver.

Note. The fuzzification process of the controller interprets any Negative adhesion gradient as Zero with 100% certainty according to the membership functions assumed in Figure 3.13. Therefore the domain of the nonlinear mapping of the fuzzy logic controller does not need to be explicitly extended to negative adhesion gradients.

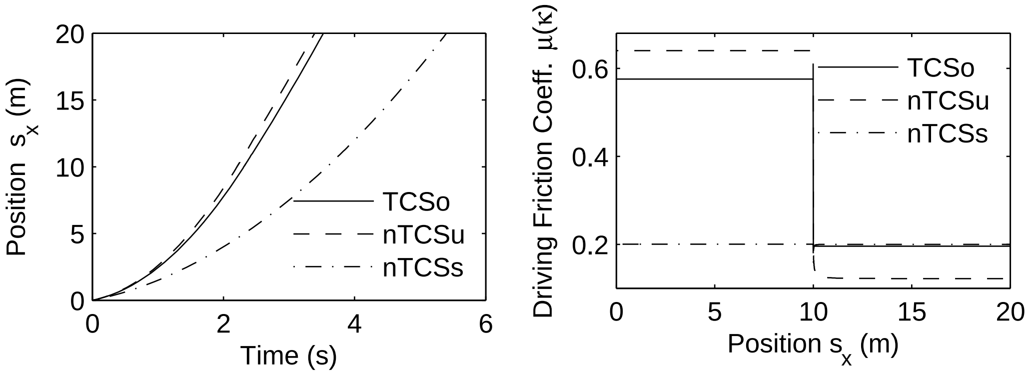
Figure 6.5f shows that traction control enables near optimal performance in the friction limited situation presented on the wet road while allowing high levels of performance in the torque limited situation on the dry road. The $\kappa - \mu$ trajectory in Figure 6.5e demonstrates that traction control fundamentally benefits $\mu(\kappa)$ on the wet road by reducing excessive wheel slip that would otherwise lead to instability with residual force generation.

6.3.2 Comparison with Open Loop Systems

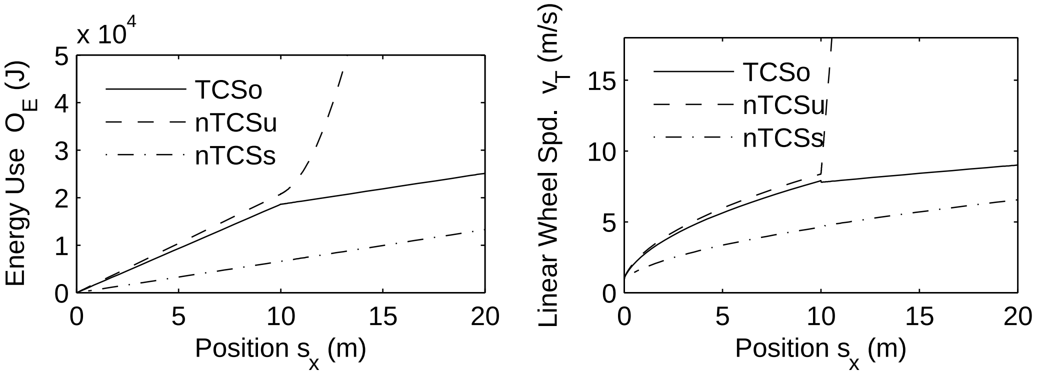
A comparison of the previous open loop and closed loop simulation results confirm the overall benefits of traction control in performance and energy economy over a variety of road surfaces. Drive cycle energy consumption O_E and completion time O_T, obtained by integration of (2.33) and (2.32), respectively, serve as universal benchmarks for system comparison across plant and controller configurations. The impact of system dynamics on these benchmarks is discussed followed by a comparison of numerical results.

The closed loop drive cycle response of TCSo and the open loop responses of nTCSu and nTCSs from Section 6.2 are compared in Figure 6.7. With the exception of Figure 6.7a, the plots of the remaining sub-figures are made with respect to s_x rather than time in order to allow a consistent analysis of system response to the road transition.

The progression in position s_x of each system through the drive cycle leads to different completion times O_T as shown in Figure 6.7a. Although the figure indicates that open loop application of torque leads to quickest O_T, the driving friction coefficients $\mu(\kappa)$ attained by each system (Figure 6.7b) indicate drive cycle choice affects the relative performance of open loop and closed loop systems. Since $\mu(\kappa)$ is directly proportional to force F_x (2.20), it dictates vehicle acceleration. Open loop application of torque results in the highest acceleration on dry road but offers the lowest on wet road due to loss of traction. In contrast, a prefixed open



(a) Vehicle Position vs Time (b) Driving Friction Coefficient vs Position



(c) Energy Use vs Position (d) Linear Wheel Speed vs Position

Figure 6.7: Closed Loop Traction Control vs Open Loop Control

loop torque limit, by design, restricts acceleration to worst case conditions on the dry road to obtain optimal acceleration on the wet road. Thus if the length of wet road in the drive cycle was to be moderately increased, the higher acceleration attained by TCSo on a wet road would allow quickest completion of the drive cycle. For a long stretch of wet road, the optimal application of torque by nTCSs would allow it to complete the drive cycle first. Thus, the selected drive cycle favors high performance on the dry road. Although dry road acceleration of TCSo is affected by control action as described in the discussion of Figure 6.6, it is able to achieve relatively high accelerations on both surfaces where either wet road acceleration is negatively affected by instability in nTCSu or dry road acceleration is limited for nTCSs.

The energy consumption of each system is compared in Figure 6.7c. Since energy use E is determined by the integration of instantaneous power to drive the wheel $E = \int \Omega\, T_a dt$, it is explained by considering the torque application in Figures 6.1a, 6.3a, and 6.5a. with the aid of the linear wheel speed ($v_T = r\,\Omega$) comparison in Figure 6.7d. On the dry road, energy use of nTCSu is slightly larger than TCSo due to higher T_a and Ω. On the wet road, energy use of nTCSu increases quickly due to high Ω resulting from instability. In contrast, energy use of nTCSs remains smaller than TCSo on both roads due to small T_a and Ω. The figure shows that traction control avoids the high energy use resulting from instability and that a low torque application reduces energy use.

Note. With the exception of the rapid energy increase observed for nTCSu after the road transition, energy use in Figure 6.7c exhibits an approximate piecewise linear increase with distance traveled. For a constant torque T_a, energy E is linear with respect to angular wheel rotation Θ since:

$$E = \int \Omega\, T_a dt = \int \frac{d\Theta}{dt} T_a dt = \Theta T_a. \tag{6.8}$$

Although E versus s_x is plotted in Figure 6.7c, it is noted that the linear wheel speed v_T closely tracks vehicle speed v_x while the system is stable as indicated in Figures 6.1b, 6.3b, and 6.5b. Hence $v_T = r\,\Omega \approx v_x$ or equivalently $\Omega \approx v_x/r$.

Table 6.4: Closed Loop vs Open Loop Performance and Energy Use

Config.	O_E (J)	% O_E Diff.	O_T (s)	% O_T Diff
TCSo	25134.2	0	3.52446	0
nTCSu	241572	861.1	3.39271	-3.7
nTCSs	13292.1	-47.1	5.40307	53.3

Substitution into (6.8) reveals the approximate linear relationship:

$$E \approx \int \frac{v_x}{r} T_a dt = \int \frac{1}{r} \frac{ds_x}{dt} T_a dt = \frac{s_x}{r} T_a. \tag{6.9}$$

Numerical results of O_E and O_T are compared in Table 6.4 for the open loop nTCSu and nTCSs systems of Section 6.2 and the closed loop TCSo system. Percentage difference is also listed with respect to TCSo. Ideally, low energy consumption and low (fast) completion times are desired. The table shows that direct open loop torque application of a torque request by nTCSu results in marginally quicker (-3.7%) performance over TCSo while incurring a large (861.1%) increase in energy consumption due to high wheel slip. In contrast, the prefixed open loop limitation of torque application to worst case road conditions results in lower (-47.1%) energy consumption than TCSo with a slower (53.3%) completion time. Hence the table summarizes that traction control enables a balance between high performance and low energy consumption.

6.3.3 Parametric Study on Performance and Energy Use

A power flow analysis is conducted to identify system components affecting energy use and performance. The impact of traction control system design on drive cycle energy use and performance is assessed using a parametric study.

Power Flow Analysis

The electric propulsion system provides a source of power input $P_{in} = \Omega T_a$ to the system. Input power is primarily used to accelerate the wheel P_{wh} and vehicle P_{veh}

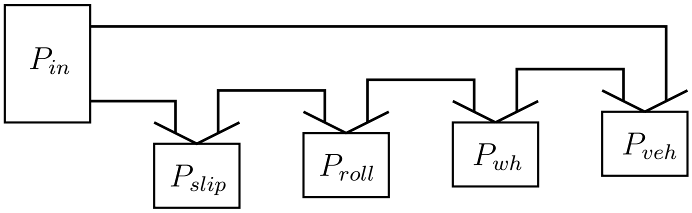


Figure 6.8: Electric Propulsion System Power Flow

and therefore the output power is $P_{out} = P_{wh} + P_{veh}$. Power losses P_{loss} are incurred due to rolling resistance P_{roll} and tire slip P_{slip} such that $P_{loss} = P_{roll} + P_{slip}$. Power flow within the traction control system is summarized in Figure 6.8 and obeys:

$$P_{out} = P_{in} - P_{loss}. \tag{6.10}$$

Expressions to calculate the components of power flow in (6.10) are derived by rearranging (2.29) to find the net torque on the driven vehicle wheel $T_{net,wh}$ (6.11) and multiplying by Ω to obtain the power output to the wheel P_{wh} (6.12). Without loss of generality, the output power to accelerate the vehicle $P_{veh} = v_x\, F_x$ is added to both sides of the equation in (6.13) to derive the final expression in (6.14) which is equivalent to (6.10).

$$\begin{aligned}
T_{net,wh} &= T_a - T_y - r\,F_x && (6.11)\\
P_{wh} &= \Omega\,T_a - \Omega\,T_y - r\,\Omega\,F_x && (6.12)\\
P_{wh} + v_x\,F_x &= \Omega\,T_a - \Omega\,T_y - (r\,\Omega - v_x)\,F_x && (6.13)\\
P_{wh} + P_{veh} &= P_{in} - P_{roll} - P_{slip} && (6.14)
\end{aligned}$$

where $P_{roll} = \Omega\,T_y$ is the power loss due to rolling resistance and $P_{slip} = (r\,\Omega - v_x)F_x$ is the power loss due to wheel slip.

An energy balance is obtained by integrating (6.14). The input energy is equivalent to the total drive cycle energy usage O_E. The change in kinetic energy in both the vehicle ΔE_{veh} and wheel ΔE_{wh} can be calculated directly by:

$$\Delta E_{veh} = \frac{1}{2}m(v_{xF}^2 - v_{xo}^2), \qquad \Delta E_{wh} = \frac{1}{2}J(\Omega_F^2 - \Omega_o^2) \tag{6.15}$$

where v_{xF} and Ω_F are the final linear and rotation speeds of the vehicle and wheel, respectively, at the end of the drive cycle, and v_{xo} and Ω_o are the initial linear and rotation speeds of the vehicle and wheel, respectively, at the beginning of the drive cycle. Therefore energy loss due to rolling resistance and wheel slip is deduced by:

$$E_{loss} = O_E - \Delta E_{veh} - \Delta E_{wh}. \tag{6.16}$$

Note. In this thesis, energy loss due to rolling and wheel slip are lumped together since they are not directly calculated during simulations. Augmenting the system model in Section 2.4 with $\dot{E}_{roll} = \Omega T_y$ would allow energy loss associated with rolling resistance to be calculated. Losses due to slip may then be determined by $E_{slip} = E_{loss} - E_{roll}$.

Overall energy use O_E of the drive cycle can be minimized by both an increase in system efficiency and a reduction in input energy. To maximize system efficiency, it is desirable to minimize losses P_{roll} and P_{slip}. Rolling resistance can be reduced by the use of low rolling resistance tires and therefore consists of a change in vehicle (plant) design. Losses due to wheel slip can be reduced by the use of a control strategy that operates a tire at low values of slip. Input energy can be reduced by a reduction in applied torque T_a.

Thus, trade-offs in energy use and performance are present depending on plant and controller design since: low rolling resistance tires are associated with degraded force generation characteristics, operation at low values of tire slip corresponds to low tire force generation and low vehicle acceleration, and a reduction in input energy limits the energy available to accelerate the vehicle and limits the final speed of the vehicle v_{xF} by (6.15).

Parametric Study

A parametric study is performed to demonstrate energy use and performance trade-offs with variation in plant and controller design of the traction control system. The design parameters under study correspond to the optimization decision variables selected in Section 4.2.2 and are varied from the nominal system TCSo according to Table 6.3 to produce systems TCSA-TCSE. The following analysis shows that a

Table 6.5: Impact of System Design on Performance and Energy Use

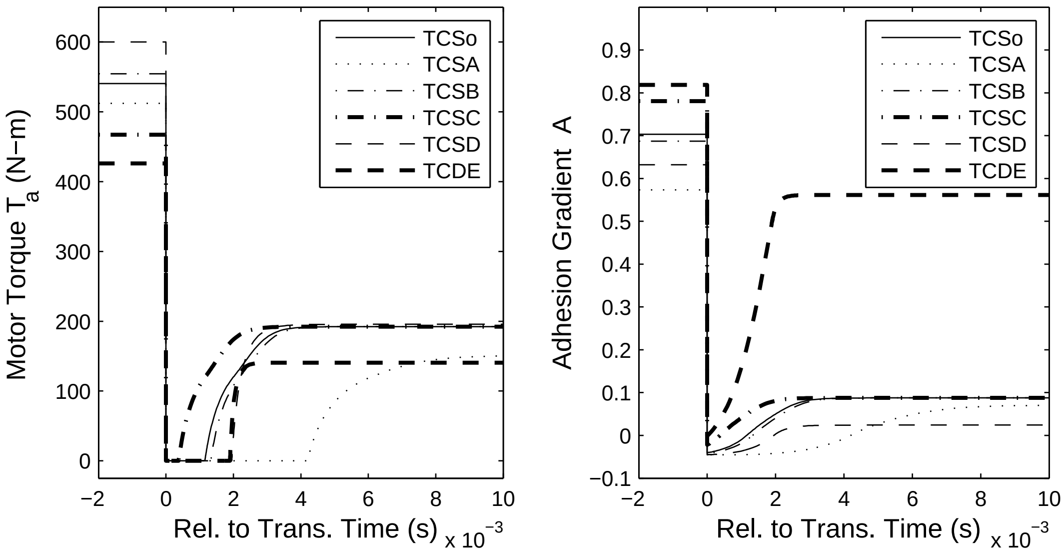
Param.	Config.	O_E (J)	% O_E Diff.	O_T (s)	% O_T Diff
Nominal	TCSo	25134.2	0	3.52446	0
Tire	TCSA	23202.0	-7.7	3.60793	2.4
A-PS	TCSB	25641.0	2.0	3.48454	-1.1
	TCSC	22526.2	-10.4	3.75711	6.6
Rule	TCSD	27423.2	9.1	3.36382	-4.6
Base	TCSE	19277.0	-23.3	3.94327	11.9

change in either plant or controller design impacts drive cycle energy use O_E and completion time O_T.

TCSA is a system using a low rolling resistance tire (Tire 1) as defined in Appendix A and whose characteristics are illustrated in comparison with the nominal performance oriented tire (Tire 0) in Figure 2.10. TCSB and TCSC are systems where the location of the Positive-Small membership function of the adhesion gradient A is set to a minimum of 0.2 and a maximum of 0.8, respectively, as described in Section 3.3.2. TCSD and TCSE are systems with an aggressive and non-aggressive fuzzy logic rule base, respectively, as obtained by changing rule conclusions $\mathcal{C}_1$ - $\mathcal{C}_6$ as defined in Section 3.3.3. Simulated drive cycle energy use O_E and completion time O_T for these systems are listed in Table 6.5 and compared to the nominal system TCSo. The results show that changes in plant and controller parameters lead to trade-offs in energy use and drive cycle completion time.

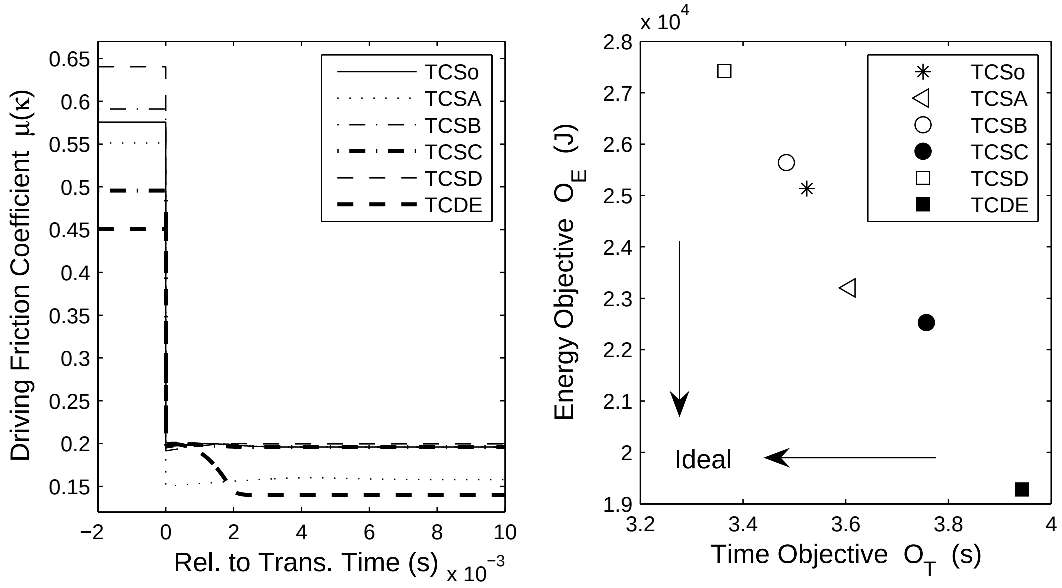
System response to the drive cycle road transition are compared in Figures 6.9a - 6.9c. The figures compare the response of the systems relative to the road surface transition time occuring at Rel. to Trans. Time = 0 s because the absolute transition times between systems vary due to differing plant and controller configurations. The following discussion details observed changes in system response to road transition for systems TCSA to TCSE.

A change in tire from a performance oriented tire to a low rolling resistance tire in TCSA changes the force generation characteristics of the tire and therefore affects



(a) Applied Torque Variation

(b) Adhesion Gradient Variation



(c) Driving Friction Coefficient Variation

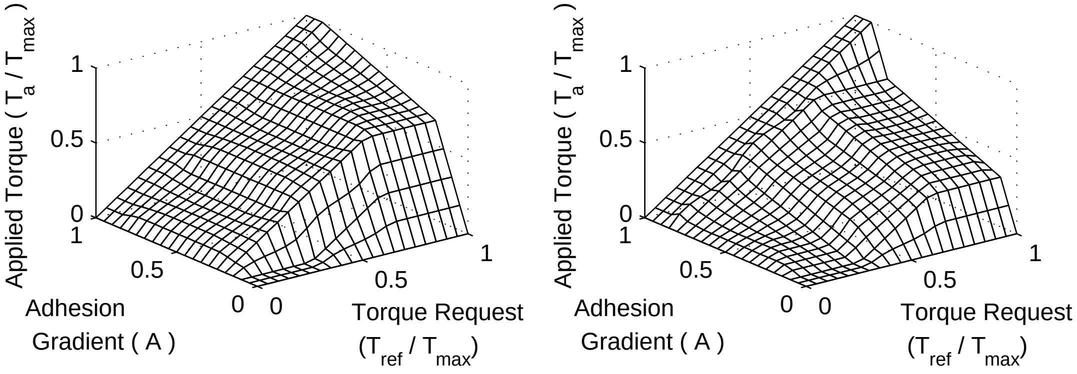
(d) Unoptimized Trade-Off in Energy and Time Objectives

Figure 6.9: Parametric Impact on Performance and Energy

the operation of the traction controller (whose parameters remain unchanged from TCSo). Torque application T_a in Figure 6.9a is reduced compared to TCSo on both dry and wet surfaces while the plot of adhesion gradient in Figure 6.9b shows that Tire 1 is operated closer to its performance limit. This is a result of Tire 1 having degraded performance compared to Tire 0 and the two figures indicate that Tire 1 operates closer to its limit even when less torque is applied as compared to TCSo. The performance on wet surface of TCSA is near optimal since adhesion gradient is small $A < 0.1$. Therefore the low $\mu(\kappa)$ observed on the wet surface in Figure 6.9c is due to a limitation in tire force generation characteristics. In contrast, the decrease in $\mu(\kappa)$ on the dry surface compared to TCSo can be countered by a change in controller to allow a higher T_a and a decrease in A to improve performance on the dry surface.

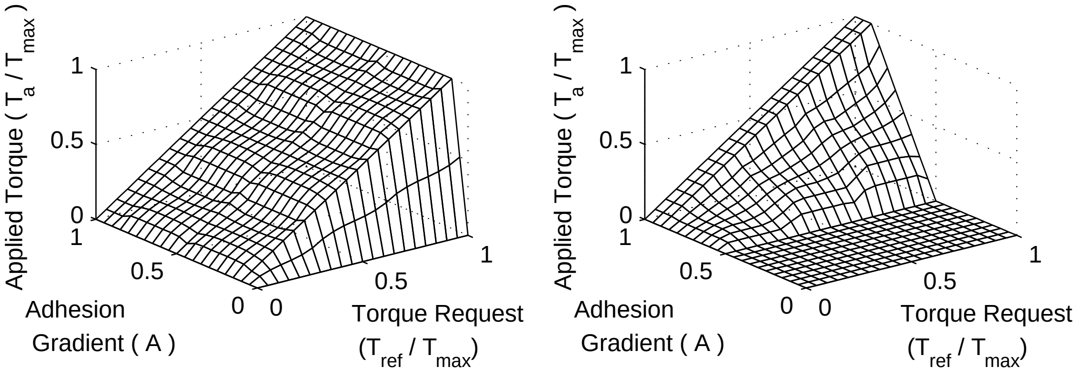
Torque re-application for TCSA is delayed relatively longer than TCSo since, as shown in Figure 6.9b, the adhesion gradient requires a longer amount of time to become positive. The delay is further explained by considering the differences of the adhesion gradient characteristics of the performance tire (Tire 0) and the low rolling resistance tire (Tire 1) shown in Figure 2.10. On the dry surface, Figure 6.9b shows that TSCo operates Tire 0 at $A \approx 0.7$ while TCSA operates Tire 1 at $A \approx 0.6$. Figure 2.10 therefore shows that, on the dry surface, TCSo operates Tire 0 at a lower tire slip κ than TCSA with Tire 1. The figure also shows that, on the wet surface, the optimal slip corresponding to $A = 0$ is similar for both tire. Therefore, after the road transition, more time is required for the tire slip of TCSA and Tire 1 to decrease such that $A = 0$.

Setting the location of A-PS to its lower limit in TCSB results in a system that considers smaller values of adhesion gradient as Positive-Small in comparison to TCSo. The impact on the shape of the fuzzy logic control surface is shown in Figure 6.10a where T_a decreases less in response to initial decreases in A as compared to the controller of TCSo as shown in Figure 3.16. The control surfaces differ mostly between adhesion gradients of 0.1 and 0.5. Therefore TCSB results in marginally less torque reduction on the dry surface compared to TCSo as shown in Figure 6.9a. The marginally higher T_a leads to marginally higher wheel slip and



(a) Small A-PS (TCSB)

(b) Large A-PS (TCSC)



(c) Aggressive Rule Base (TCSD)

(d) Non-Aggressive Rule Base (TCSE)

Figure 6.10: Parametric Variation in Fuzzy Logic Controller

a marginally more negative adhesion gradient is caused by the road transition. As a result, torque re-application is marginally delayed compared to TCSo after the road transition since the adhesion gradient only requires marginally more time to re-stabilize (regain positive A) as shown in Figure 6.9b.

Setting the location of A-PS to its upper limit in TCSC results in a system that considers higher values of adhesion gradient as Positive-Small in comparison to TCSo. The impact on the shape of the fuzzy logic control surface is shown in Figure 6.10b where T_a decreases more rapidly for initial decreases in A compared to TCSo. The reduced T_a on the dry road results in less wheel slip and less time for the wheel to regain stable operation after the road transition. Thus re-application of T_a occurs sooner than in TCSo and TCSB.

Note. Higher T_a in TCSB translates to higher $\mu(\kappa)$ on the dry road as shown in Figure 6.9c while the converse is true for lower T_a in TCSC. However, steady-state performance on the wet road is similar for TCSo, TCSB, and TCSC as shown in Figure 6.9b and Figure 6.9c. This is due to the choice of the location of the Zero membership function of the adhesion gradient Z=0.1 and the fact that steady-state operation of the systems on the wet road occur at $A < 0.1$. Since membership in A-PS is zero at adhesion gradients less than Zero, the variation of A-PS does not affect fuzzy logic controller output on the wet surface. In future research, to affect steady-state performance on the wet road surface through A-PS, the location of the Zero membership function could be changed to less than the steady-state condition to allow non-zero membership in the Positive-Small membership function of the adhesion gradient. The remaining discussion shows that controller design still affects operation on both road surfaces through changes in the fuzzy logic rule base.

The aggressive rule base of TCSD results in the least amount of torque reduction as shown in Figure 6.9a and operates the tire near its limit of performance on the wet road surface as shown by the low adhesion gradient attained in Figure 6.9b. Performance on the dry surface is at a maximum since $T_a = T_{ref}$. Although TCSD operates the tire at a lower adhesion gradient than TCSo, only a marginal performance gain in $\mu(\kappa)$ is obtained on the wet surface as shown in Figure 6.9c. The definition of $\mathcal{C}_1$ - $\mathcal{C}_6$ and resulting fuzzy logic control surface of TCSD shown in Figure 6.10c effectively disable traction control until the adhesion gradient becomes less than the location of the Zero membership function of the adhesion gradient.

The non-aggressive rule base of TCSE results in the most amount of torque reduction as shown in Figure 6.9a and does not operate the tire near its limit of performance as shown by the high adhesion gradient ($A > 0.5$) on both road surfaces in Figure 6.9b. As a result, $\mu(\kappa)$ is the lowest in Figure 6.9c. The definition of $\mathcal{C}_1$ - $\mathcal{C}_6$ and resulting fuzzy logic control surface of TCSE shown in Figure 6.10d limit T_a to zero for adhesion gradients less than the location of A-PS = 0.5. Therefore, the torque re-application after the road transition is delayed in Figure 6.9a until

Table 6.6: Percentage of Energy Stored in Vehicle Kinetic Energy

Config.	O_E (J)	v_{xF} (m/s)	ΔE_{veh} (J)	$(\Delta E_{veh}/O_E) \times 100$ (%)
TCSo	25134	8.8396	23142	92.07
TCSA	23202	8.4785	21265	91.65
TCSB	25641	8.9262	23603	92.05
TCSC	22526	8.3760	20747	92.10
TCSD	27423	9.2187	25196	91.88
TCSE	19277	7.7502	17720	91.92

$A > 0.5$.

The observed trade-offs between energy use and performance listed in Table 6.5 are graphically summarized in Figure 6.9d by plotting O_E against O_T. The parametric study therefore demonstrates that O_E and O_T are dependent on both plant and controller design decisions. Further, these system level objectives are dependent on one another and cannot be optimized independently. A necessary trade-off between O_E and O_T occurs for variations in system design. As indicated in the figure, an ideal system would have both low energy use and drive cycle completion time. Hence an optimization procedure is required to balance energy use and performance requirements.

Note. As shown in Table 6.6, a significant amount (91-92 %) of drive cycle energy use O_E is stored as kinetic energy ΔE_{veh} in the moving vehicle. Therefore, the trade-off between O_E and O_T is strongly influenced by the final vehicle speed v_{xF} as obtained by recording v_x at time O_T such that $v_{xF} = v_x(O_T)$. Also, Figure 6.9a indicates that controller transients persists for less then 10 ms for a drive cycle with a duration between 3 and 4 seconds (Table 6.5). Therefore, the trade-off between O_E and O_T is weakly affected by transient action taken by the controller.

Table 6.7: GA Parameters

Parameter	Value
Mutation Probability p_m	2%
Real-Coded Mutation Standard Deviation σ	0.1
Population Size	200
Terminating Condition	After 50 Generations
Number of Parallel CPUs	50
Members per CPU	4

6.4 Simultaneous Optimization Results

The traction control system is optimized with respect to the weighted sum of the drive cycle energy use O_E and completion time O_T using the mixed encoding, multi-chromosomal GA. Problem-independent parameters that impact the execution of the GA are first described. Results from the simultaneous plant/controller optimization of the traction control system are then analyzed with focus on the impact of the relative energy to performance weighting on system design. The evolutionary performance of the GA is examined to further describe its optimization process.

6.4.1 GA Parameters

The numerical values of parameters that affect the operation of the GA are summarized in Table 6.7. The selection of a GA's parameters is used to balance search space exploration with running time and influences the quality (optimality) of optimization results. In general, the selection of a GA's parameters is problem dependent and remains an active area of research with recent advances, for example, in on-line adaptive tuning of the mutation probability [60]. The following discussion outlines the parameter selection process applied in this thesis.

The mutation probability is set at $p_m = 2\%$ in accordance with the general recommendation that $p_m < 10\%$ [48] to balance its benefit in generating new building blocks with its tendency to destroy long building blocks.

The standard deviation associated with the Gaussian mutation of the real-coded variable A-PS is set at $\sigma = 0.1$. This allows 99.7% of mutations to produce changes within 3 standard deviations (± 0.3) of A-PS [61]. Therefore, assuming that A-PS is assigned a nominal value of 0.5, $\sigma = 0.1$ generates mutations mostly within the constrained range of A-PS $\in [0.2, 0.8]$ and reduces the probability of repairing an out-of-bounds mutation.

The population size of 200 and terminating condition of the GA after 50 generations were chosen based on a trial procedure that results in satisfactory convergence of the population to a single optimized candidate solution. The evolutionary performance of the GA is further considered in Section 6.4.3.

To achieve a higher fitness evaluation throughput rate, the GA is implemented using the parallel farmer-worker architecture described in Section 5.1.4. For each generation, the farmer node selects and distributes 4 sets of parent genetic information to 50 worker nodes. For each set of parents received, the worker nodes perform the crossover and mutation operations to generate a new candidate solution in addition to objective evaluation through system simulation. The resulting offspring's genetic information and simulation results are then sent back to the farmer node for fitness evaluation and selection for the next generation.

6.4.2 Impact of Energy Use to Performance Weighting

Simultaneous plant/controller optimization results of the traction control system are summarized in Table 6.8. Six optimization runs, labeled as TCS1 to TCS6, were performed under different energy weights w_E. For each optimization run, the table lists the individual drive cycle energy use O_E and completion time O_T objectives used in the calculation of the system-level objective O_s. In addition, the table shows the resulting optimized assignment of system design variables TIRE, A-PS, and NODE that specify the selection of vehicle tire, the location of the Positive-Small adhesion gradient membership function, and the fuzzy logic rule base configuration, respectively. Finally, the table indicates the level of the selected rule base (from Figure 5.19). The following discussion describes the selection of the energy weight w_E coefficient and its impact on optimization results. The optimized results are

Table 6.8: Optimization Results for Various Energy Weights w_E

Config.	w_E	O_E	O_T	O_s	TIRE	A-PS	NODE	*level*
TCS1	0	27423	3.3638	3.3638	0	0.38005	2	1
TCS2	1e-6	26748	3.3640	3.3908	1	0.26404	1	0
TCS3	8e-6	26718	3.3643	3.5780	1	0.37270	15	3
TCS4	10e-6	26654	3.3648	3.6314	1	0.36856	33	5
TCS5	12e-6	26474	3.3667	3.6844	1	0.40093	53	6
TCS6	100e-6	16621	4.1552	5.8168	1	0.60768	83	11

then compared with the unoptimized nominal system TCSo from Section 6.3.1.

The energy weight coefficient w_E controls the relative importance of the energy objective O_E and the time objective O_T in the minimization of their weighted sum O_s (4.12). To avoid having one objective dominate the other, a knowledge of the scalar range of the individual objectives is required to select an appropriate numerical value for w_E. Although Table 6.8 lists the optimization runs by ascending values of energy weight, the optimizations were performed in a different order beginning with an initial selection of w_E derived from the results of the parametric study summarized in Table 6.5. Consideration of Table 6.5 indicates that O_E changes on the order of 1e2 to 1e4 while O_T changes on the order of 1e-1. Hence, an initial energy weighting of $w_E =$ 1e-5 (corresponding to optimized system TCS4) was selected to approximately balance the optimization of O_E and O_T. To determine the impact of objective weight selection on the optimized system, simulation runs were performed where w_E is an order magnitude smaller (TCS2) and larger (TCS6) then used in TCS4. In comparing the resulting optimized selection of NODE in Table 6.8, it was noted that a change in an order of magnitude in w_E resulted either in the selection of a very aggressive rule base configuration in TCS2 ($level(\text{NODE} = 1) = 0$) or a very non-aggressive rule base configuration in TCS6 ($level(\text{NODE} = 83) = 11$). Therefore, additional optimization runs were performed with w_E deviating from TCS4 by -2e-6 in TCS3 and +2e-6 in TCS5. Since the tire selection TIRE, did not change in TCS2 to TCS6, an optimization run with $w_E = 0$ in TCS1 was performed to verify the operation of the GA.

Overall, the simulation results in Table 6.8 show the following trends:

- Energy use O_E decreases while drive cycle completion time O_T increases as the weighting on energy use is increased. These trends are expected and confirm the trade-off between energy economy and vehicle performance. Thus, the selection of w_E determines how the optimization process balances conflicting objectives in the minimization of O_E and O_T.

- The system-level objective O_s increases with w_E since O_E is four orders of magnitude larger than O_T and therefore increases in w_E tend to cause a net increase in O_s. The trend is purely a consequence of the scalar range of the individual objective functions O_E and O_T. In general, O_s is only used internally to determine an optimized solution within an optimization run and does not have significance when compared across optimization runs.

- Optimization runs TCS2 to TCS6 indicate that the use of a low rolling resistance tire (TIRE=1) leads to an optimal system configuration whenever energy use is considered. In other words, unless performance is the only criteria in the vehicle design with $w_E = 0$ in TCS1, the energy saving benefit resulting from the use of a low rolling resistance tire outweighs the incurred degradation in longitudinal tire performance.

- The location of the Positive-Small adhesion gradient membership function A-PS tends to be greater for larger weightings of energy use (TCS4 to TCS6). This is expected since a larger value of A-PS causes a quicker decrease in T_a for initial decrease in A as shown in the comparison of Figure 6.10a and Figure 6.10b. Therefore, systems with higher values of A-PS tend to output less torque T_a when operating the tire over similar ranges of A and are associated with lower O_E and higher O_T.

  The contrary trend observed between TCS3 and TCS4 may indicate that system objectives may be less sensitive to changes in A-PS than in NODE. This is partially supported by the results of the parametric study summarized

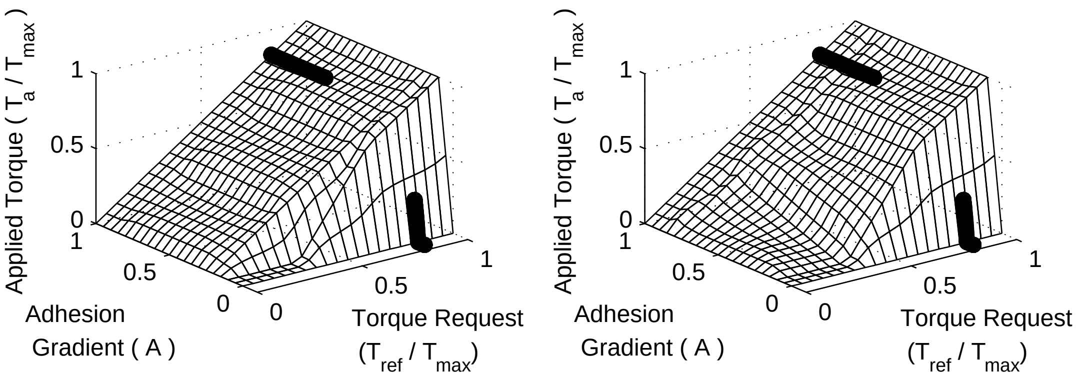
Table 6.9: Impact of A-PS Variation in TCS1 on Objectives

Config.	A-PS	O_E	O_T
TCS1a	0.2	27423.24	3.363832
TCS1b	0.8	27423.26	3.363834

in Table 6.5 where it is seen that the selection of rule base may exhibit a greater impact (-23.3%, 11.9%) on the objectives O_E and O_T, respectively, than the selection of A-PS (-10%, 6.6%).

The values of A-PS for TCS1 and TCS2 cannot be included in the trend analysis since their rule base configurations make the optimization insensitive to changes in A-PS. This is numerically demonstrated for TCS1 in Table 6.9 by taking the system variable selection of TCS1 (TIRE=0 and NODE=2) and running two additional individual simulations TCS1a, TCS1b with variation in A-PS. The table shows that the resulting objective values differ in the seventh significant digit. The numerical difference is insignificant since numerical solver settings were chosen for convergence to 5 significant figures of accuracy (Section 6.1). Therefore, for the rule base selection of TCS1, the optimization is insensitive to changes in A-PS and the TCS1 results in Table 6.8 can be considered identical to those of TCS1a and TCS1b in Table 6.9.

Detailed examination of the rule base selection in TCS1 and TCS2 reveal the reason for the insensitivity to variation in A-PS. For TCS1 and TCS2, the respective rule base selection of NODE = 2 and NODE = 1 is such that $\mathcal{C}_3 = \mathcal{C}_4 = 3 = MH$ and $\mathcal{C}_5 = \mathcal{C}_6 = 4 = H$ by mapping Ψ_{RULE} (4.8). In conjunction with Figure 3.15b, a traction controller is therefore defined that prescribes the same output conclusion over a range of Z to PH adhesion gradient for a given torque request input in the range MH to H. This is illustrated by a comparison of the fuzzy logic controller surface plot of TCS1a (Figure 6.11a) against TCS1b (Figure 6.11b). For torque request inputs above 0.5, the plots are nearly identical due to the definition of $\mathcal{C}_3$ to $\mathcal{C}_6$. For torque request inputs less than 0.5, the plots show the impact of changes in A-PS on



(a) TCS1a: Small A-PS=0.2

(b) TCS1b: Large A-PS=0.8

Figure 6.11: A-PS Variation in Traction Controller Rule Base of TCS1

the fuzzy logic controller since in this region, the location of A-PS determines the transition of torque output T_a from Medium-Small ($\mathcal{C}_2 = 2$) to Small ($\mathcal{C}_1 = 1$). To emphasize the independence of A-PS in the optimization of TCS1, the trajectory of T_a along the surface plot is shown in Figure 6.11 for both TCS1a and TCS1b. The two trajectories are nearly identical and result in similar objective values O_E and O_T as listed in Table 6.9. Since the trajectories taken are only influenced by the definitions of $\mathcal{C}_3$ to $\mathcal{C}_6$ as described in Section 3.3.3, the preceding analysis applies equally to explain the independence of A-PS in the optimization of TCS2.

- The *level* of the optimized rule base increases as w_E is increased from TCS2 to TCS6. Since an increase in level is observed along with a decrease in O_E and an increase in O_T, the optimization results support the assumption that a higher *level* corresponds to a less aggressive rule base as stated in (5.20). The observation also supports the assumption that the level of a fuzzy logic rule base is a building block in the encoding of the rule base. The novel encoding of a fuzzy logic rule base as a member of a poset by delta inclusion (Section 5.3.3) is therefore shown to be efficient since it manipulates the level of a rule base directly and acts to preserve level information encoded within

Table 6.10: Improvement in Optimized Systems over Benchmark TCSo

Config.	w_E	O_s	$O_{s,TCSo}$	% Improvement
TCS1	0	3.3638	3.5245	4.56
TCS2	1e-06	3.3908	3.5496	4.47
TCS3	8e-06	3.5780	3.7255	3.96
TCS4	10e-06	3.6314	3.7758	3.82
TCS5	12e-06	3.6844	3.8260	3.70
TCS6	100e-06	5.8168	6.0375	3.66

building blocks.

The observed trend can be extended to include the cursory discrepancy observed in TCS1 by noting that NODE=2 for TCS1 and NODE=1 for TCS2 represent rule base conclusions that differ only in $\mathcal{C}_1$. Since the choice of constant T_{ref} only stimulates rule base conclusions $\mathcal{C}_3$ to $\mathcal{C}_6$ as described in Section 3.3.3, the selection of either NODE=2 or NODE=1 produces equivalent simulation results for the chosen drive cycle and do not preferentially impact the optimization process.

A comparison between the optimization results TCS1 to TCS6 and the benchmark nominal system TCSo can be made by calculating the unoptimized system-level objective $O_{s,TCSo}$ consisting of the weighted average of the unoptimized objective values O_E and O_T for TCSo obtained from Table 6.5. In Table 6.10, the comparison of optimized objective values O_s for TCS1 to TCS6 against the unoptimized objective value $O_{s,TCSo}$ of TCSo indicates that the optimization process allows an objective value improvement of approximately 4% over the various choices of energy weights w_E.

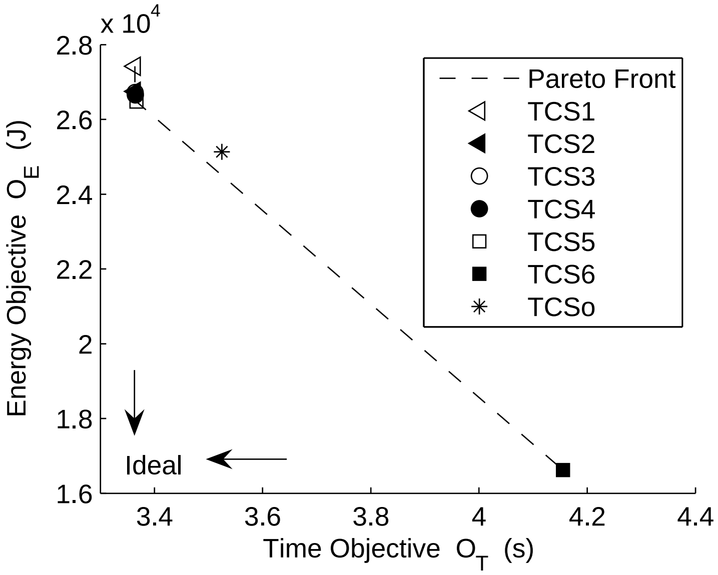
As stated in Section 4.2.3, the solution to a multi-objective optimization problem is not unique but is instead represented by a Pareto set of designs for which no individual objective may be improved without compromising any other objective. In the current traction control system design, the Pareto set is defined as the set of all globally-optimal designs over the allowable continuous range of the energy weight $w_E \in [0, 1]$. Systems TCS1 through TCS6 represent approximate members of the Pareto set since a GA optimization does not guarantee a global optimization. An approximation of the Pareto set is plotted in Figure 6.12a using the optimized objective values of TCS1 to TCS6. The figure also indicates an approximation of the **Pareto front** [62] that is defined by interpolating between members of a Pareto set. Since it is defined in terms of a Pareto set, the Pareto front in Figure 6.12a defines a boundary that indicates the limits in the simultaneous minimization of O_E and O_T over the variation of weight w_E. For example, the unoptimized nominal system TCSo is plotted in Figure 6.12a and is behind the Pareto front.

Further consideration of Figure 6.12a shows that the optimization results in Table 6.8 are not evenly distributed along the Pareto front. More optimization runs are necessary with energy weights $w_E \in$ [20e-6, 100e-6] to provide an accurate approximation of the overall Pareto front. The current simulation results do, however, allow a relatively accurate approximation of the Pareto front with $w_E \in$ [0, 12e-6] as shown by plotting TCS1 to TCS5 in Figure 6.12b.

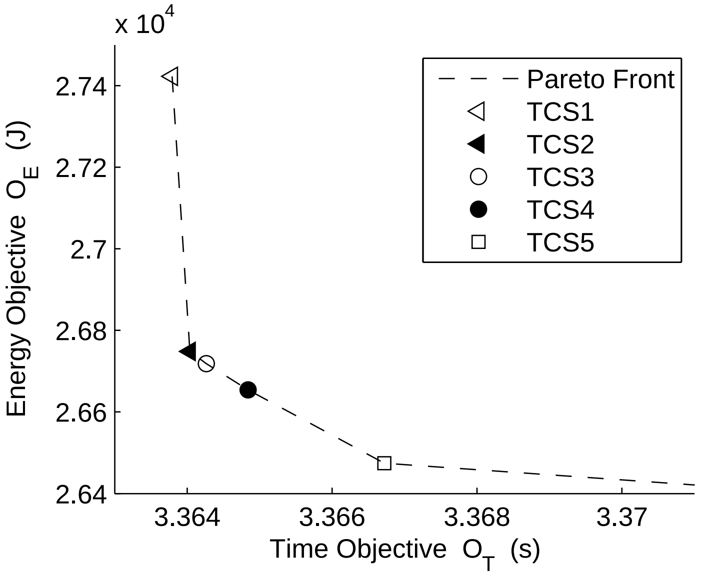
As a member of a Pareto set, a specific optimized traction control system design cannot be prescribed without further criteria for specifying the relative weighting of O_E and O_T. Such criteria is application specific and needs to be developed in a greater context, for example, as determined by a study on driver preferences. Without further guidance on the selection of w_E, systems TCS1 to TCS6 can be considered equally optimal in balancing the energy use to performance trade-off.

6.4.3 Evolutionary Performance of the GA

As a heuristic optimization method, a GA does not guarantee global optimality although it may be effective at finding near optimal solutions. In general, the fitness evaluation and selection process of a GA causes diversity in an initial population to converge towards a fit candidate solution that has been created by the use of genetic operations. As discussed in Section 6.4.1, the optimality of the final solution is affected by the selection of a GA's parameters. The following analysis of the evolutionary performance of the mixed encoding, multi-chromosomal GA provides further insight into the influence of the GA's parameter selection on its optimization

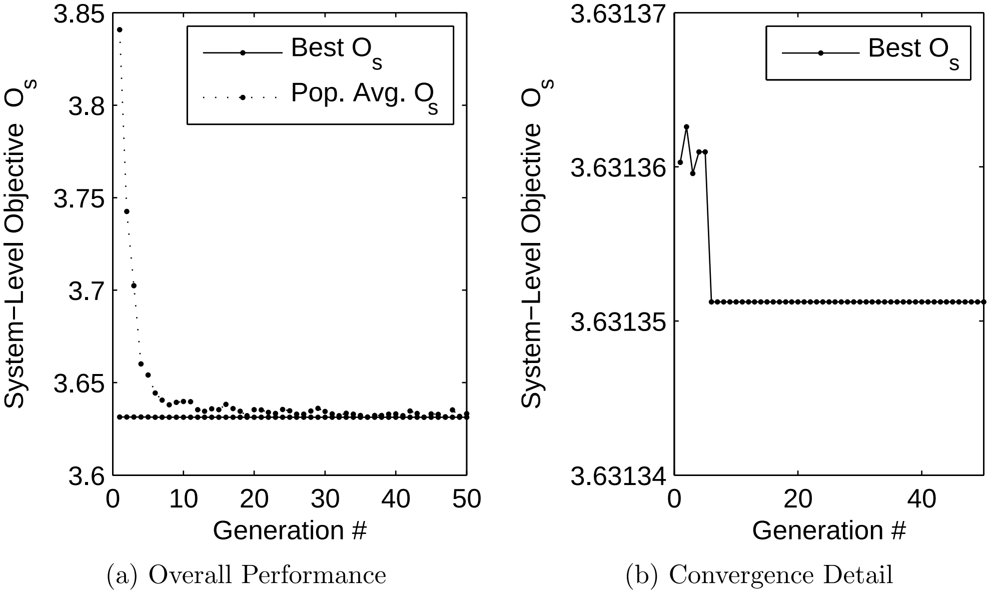


(a) Overall Pareto Set Approximation



(b) Pareto Set Approximation Detail

Figure 6.12: Approximation of Pareto Set



(a) Overall Performance

(b) Convergence Detail

Figure 6.13: Evolutionary Performance of the GA for TCS4

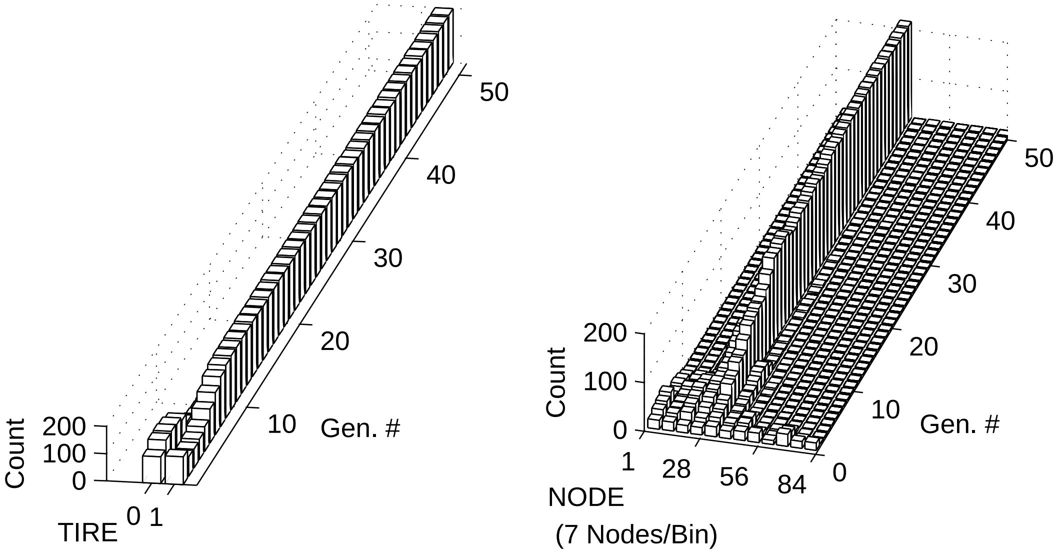
process.

For the GA's parameters listed in Table 6.7, Figure 6.13 provides a graphical representation of the GA performance for the TCS4 optimization run. Overall performance is summarized in Figure 6.13a where the best (minimum) system-level objective value O_s within each generation is plotted over the evolution of the population (Gen. #). The plot indicates that the GA finds a near optimal solution within the initial population (Gen. 0) with numerically insignificant improvement (Figure 6.13b) during the course of the optimization process. The average O_s over all members of each generation is plotted in Figure 6.13a and shows that population members tend to converge towards the fit candidate solution.

The presence of a near optimal solution within the initial population of a GA is generally not expected since population size typically limits initial search space coverage. However, consideration of the discrete portion of the search space specific to the current traction control system optimization problem reveals that 168 discrete combinations exists as a result of the limited selection from among 84 rule base possibilities and 2 tire selections. Due to the use of a constant T_{ref} that only stimulates a portion of the rule base as explained in Section 3.3.3, some of the 84 rule base configuration possibilities are equivalent (specifically those with identical $\mathcal{C}_3$ to $\mathcal{C}_6$). Since the population size is 200 (from Table 6.7), it is large enough to more than doubly cover the discrete search space and thus increases the possibility of a near optimal solution being present in the initial population. It is expected that a near optimal candidate solution would be absent from the initial population if the population size is decreased.

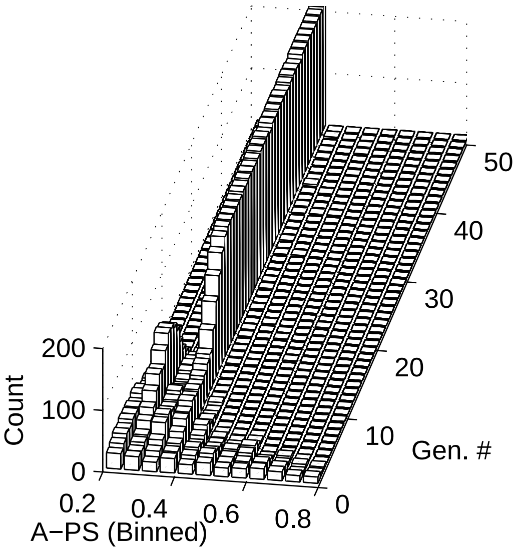
Further insight into the operation of the GA is provided by plotting a histogram (frequency count) of values assumed by optimization variables per generation over the evolution of the population. Such plots are shown, for the optimization of TCS4, in Figures 6.14a to 6.14c and depict the distribution of values assumed by optimization variables TIRE, NODE, and A-PS, respectively. To simplify graphical interpretation, consecutive nodes numbers are binned in groups of 7 in the distribution of NODE shown Figure 6.14b. Similarly, the distribution of the values of A-PS in Figure 6.14c is also binned such that each bin represents a range of 0.05. All three plots show an approximately uniform distribution in the values of the optimization variables at the initial generation (Gen. 0) caused by random population initialization. As new generations are created, the plots show convergence towards the optimized solution for TCS4 listed in Table 6.8 before the termination of the optimization process at Gen. 50. For example, Figure 6.14a indicates that approximately half (100) of the candidate solutions in Gen. 0 select the performance tire TIRE=0 while by Gen. 50, all 200 candidate solutions select the low rolling resistance tire TIRE=1.

Although, Figure 6.13a indicates the presence of the near optimal solution of TCS4 within the initial population, the plots of the evolution of TIRE and A-PS show an initial increase, and hence preference, in candidate solutions containing non-optimal TIRE=0 (up to Gen. 4) as well as A-PS $\in [0.25, 0.3]$ (up to Gen. 10). Intuitively, this indicates that a number of good solutions exist with these characteristics and initially results in a higher combined probability of selection than the final solution of TCS4. However, the high fitness characteristics associated with



(a) Evolution of Tire Selection

(b) Evolution of Rule Base Selection



(c) Evolution of A-PS Selection

Figure 6.14: Optimization Variable Distribution for TCS4

the variable selections in TCS4 allow them to spread through the entire population. Therefore, the reversal of the frequency trends of TIRE and A-PS demonstrate the expected survival of fit building blocks and supports the general building block hypothesis of the GA operation stated in Section 5.1.1.

Chapter 7

Conclusions and Future Work

A simultaneous plant/controller optimization is performed on an electric vehicle traction control system with respect to conflicting energy use and performance objectives. Compared to a typical sequential plant then controller design process, the formulation of a combined plant/controller design problem enables overall improvement of a system design. Although a highly simplified traction control system is considered, the resulting optimization problem exhibits characteristics in common with larger problems including: a coupling between plant and controller design variables that precludes their independent optimization, a mixture of both continuous and discrete design variables, and the presence of nonlinear system dynamics requiring simulation to evaluate optimization objectives. The optimization of the traction control system requires development of a system model, formulation of the optimization problem, and the proposal, implementation, and execution of a solution strategy.

The system model consists of a vehicle drive cycle, vehicle model, and traction controller. A simple drive cycle is used to evaluate the dynamic response, energy use O_E, and completion time O_T (representing performance) of a vehicle under a straight-line acceleration over 20 m. A road transition is located midway in the drive cycle and is represented by a step change from a dry road to a wet road with an 80% reduction in coefficient of friction. Vehicle dynamics are described using a single-wheel longitudinal model to focus analysis on longitudinal tire force generation and rolling resistance.

Longitudinal tire force F_x is a strong function of the tire slip $\kappa \in [0, 1]$ that is caused by differences in vehicle speed and linear wheel speed. Under free rolling conditions $F_x = 0$ and $\kappa = 0$. For increasing κ, F_x exhibits an initial increase to a peak force at an optimal κ^* before decreasing to a fraction of its maximum at $\kappa = 1$. In general, the position of the peak force, κ^*, and hence the tire performance limit, change with road conditions. Therefore, the Magic Formula is used to model F_x generation on the dry road, and the similarity method is used to extrapolate tire force characteristics on the wet road. Further, tire adhesion gradient A provides an alternative method of locating optimal κ^* since it is defined proportionally to the derivative of F_x and therefore exhibits a positive to negative sign change at κ^*.

Tire rolling resistance is caused by energy loss associated with the cyclic deformation experienced by portions of a rolling tire as they contact the ground. Rolling resistance is modeled as a constant moment opposing wheel rotation. The observation that rolling resistance decreases as inflation pressure increases allows the tire force generation characteristics of a low rolling resistance tire to be derived from those of a highly inflated tire. Compared to the tire force characteristics of a performance tire (Tire 0), a low rolling resistance tire (Tire 1) exhibits a lower peak force and its use incurs a trade-off in tire performance to energy economy. The selection of either Tire 0 or Tire 1 for use in a vehicle design is represented by the variable TIRE in the overall traction control system optimization.

The quick and accurate torque generation and measurement capabilities inherent in an electric vehicle propulsion enables the design of a high performance traction control system such as the adhesion gradient fuzzy logic controller proposed in [12]. The adhesion gradient traction controller monitors driver torque request T_{ref} and tire adhesion gradient A and decreases output torque application T_a to a vehicle wheel when adhesion gradient decreases towards zero. The torque reduction enables recovery from unstable tire operation where $\kappa \rightarrow 1$ causes excessive wheel speed, high energy consumption, and decreasing F_x. The adhesion gradient controller in [12] is adopted in this work. However, in place of a manual parameter tuning procedure, the non-intuitive specification of the location (shape) of the Positive-Small adhesion gradient membership function A-PS and a portion of the fuzzy logic rule base RULE is considered in the overall traction control system optimization.

A traction control system optimization problem is defined in terms of the minimization of the weighted sum of O_E and O_T over the selection of variables TIRE, A-PS, and RULE. The system model is incorporated within an optimization loop and the all at once variable selection strategy is implemented using a GA to iteratively explore the search space. A mixed-encoding, multi-chromosomal GA is designed to encode the mixture of continuous and discrete optimization variables within three single-gene chromosomes of different data types. The variables TIRE, and A-PS are represented, respectively, using standard encodings for a single bit binary string, and a real-valued number while a novel fuzzy logic rule base encoding is developed and applied to indirectly represent RULE.

Attention to rule base encoding is required since the direct encoding of rule base conclusions as independent integers results in exponential growth of the search space with the number of rule base entries. A 98% reduction in feasible search space may be obtained by imposing constraints on rule base entries derived from the intuitive observation that a rule base should prescribe an increase in T_a as controller inputs T_{ref} and A increase. Under a direct encoding, the constraints create a large infeasible search space and introduce computational overhead in handling infeasible candidate solutions. A novel rule base encoding is therefore proposed to simultaneously address the challenges in rule base constraint satisfaction and search space reduction. Possible rule base configurations are defined as nodes and assigned a level within the novel definition of a partially ordered set (poset) by delta inclusion. A relationship is assumed between the level of a rule base configuration and its impact on optimization objectives O_E and O_T where a high node level is associated with a low O_E and a high O_T. The relationship allows the level of a rule base to be considered as a building block of genetic information that influences the overall fitness of a candidate solution in a GA. Hence, the rule base RULE is indirectly represented by a variable NODE that references a corresponding rule base configuration within a poset by delta inclusion. Specialized crossover and mutation operations are defined on NODE to directly manipulate the level of the referenced rule base configuration.

Both unoptimized (open loop and closed loop) and optimizing (closed loop) simulation studies are performed, using the previously described drive cycle, to allow analysis of the traction control system design. A simulation convergence study is done to determine numerical solver settings and establish the numerical accuracy of results to 5 significant figures.

Open loop simulation of the uncontrolled vehicle model shows that constant torque application leads to unstable tire operation when the road surface transition causes a reduction in the road surface coefficient of friction. Imposing a prefixed open loop torque limitation to ensure stability under worst case wet road conditions enables optimal tire operation on the wet road but unnecessarily limits performance on the dry road. The simulations show the performance limitations of an open loop system. Further, analytical derivation of the steady-state and limit conditions attained during operation of the dry and wet road surfaces, respectively, enables the verification of a correct implementation of the vehicle and tire dynamic equations.

A benchmark drive cycle response of an unoptimized closed loop system is obtained using the adhesion gradient traction controller with nominal parameter definitions from [12]. Comparison of the open loop and closed loop simulation results show that traction control enables a high (but non-optimal) performance across both dry and wet road surfaces while avoiding the high energy use associated with unstable tire operation on the wet surface. A parametric study on the nominal closed loop system shows that changes in any of the system design variables TIRE, A-PS, and RULE (or NODE) produce conflicting changes in drive cycle energy use O_E and completion time O_T. The parametric study establishes the need for an optimization procedure to balance the trade-off in O_E and O_T.

In the simultaneous plant/controller optimization of the traction control system, the specification of the energy weight coefficient w_E controls the relative weighting of O_E and O_T. A range of optimized designs are obtained through the variation of w_E across multiple optimization runs. Optimization results indicate that, under straight-line acceleration and unless energy concerns are completely neglected

($w_E = 0$), low rolling resistance tires should be incorporated in a traction control system design since the energy saving benefits outweigh the associated degradation in longitudinal tire performance. A weak relationship between an increase in A-PS and an increase in w_E exists since the sensitivity of O_E and O_T to changes in A-PS is dependent on rule base configuration. Finally, the level of a selected rule base configuration is observed to increase with w_E. Since w_E controls the relative weighting between O_E and O_T, an increase in w_E causes a decrease in O_E and an increase in O_T. Therefore, optimization results support the assumption that a rule base of high level results in low O_E and high O_T. Consequently, the novel encoding of a fuzzy logic rule base as a member of a poset by delta inclusion is shown to be efficient since its crossover and mutation operators manipulate the level of a rule base directly.

7.1 Future Work

To facilitate the consideration of the simultaneous plant/controller optimization of the traction control system, various assumptions are made to simplify both the system model as well as the optimization process itself. Future work to be considered can be classified as improvements and extensions to either the traction control system model, or the optimization process.

The following improvments and extensions relating to the drive cycle, vehicle and tire model, and adhesion gradient controller should be considered in future studies of the high performance traction control of low rolling resistance tires:

- Although the simple drive cycle considered in this work enables analysis of the basic system response to a step change in road friction with constant driver torque resquest T_{ref}, it restricts controller operation to a single dimension along a two dimensional surface defined by controller inputs T_{ref} and adhesion gradient A. Also, the basic system response is dominated by steady state operation before and after the road transition with a relatively quick transient behavior at the road transition. A longer drive cycle design incorporating varying T_{ref} and more transients in the road surface coefficient of friction should be considered to approximate realistic driving conditions in the development of a practical traction controller.

- Although an ideal traction control system exhibits both low energy use O_E and completion time O_T, it may be more intuitive to formulate an optimization problem in terms of maximizing energy efficiency and minimizing completion time. Analysis of the simulation results from the parametric study shows that 91-92% of energy use is associated with the kinetic energy ΔE_{veh} used to accelerate a vehicle. Thus, the use of O_E as a minimization objective implicitly penalizes systems with a high final speed at the end of the drive cycle. The maximization of system efficiency, defined as $\Delta E_{veh}/O_E$ would allow the optimization of final vehicle speed to be defined separately if desired.

- The use of the single-wheel longitudinal model in the current work neglects the vertical dynamic weight transfer from front to rear wheels associated with vehicle acceleration. Since tire force generation and rolling resistance are influenced by normal loading, a longitudinal bicycle model should be implemented to consider dynamic weight transfer when significant amounts of acceleration are modeled. Aerodynamic resistance should also be modeled since the associated vehicle speed-dependent drag force contributes to dynamic weight transfer.

- The longitudinal tire force generation model may be refined by modeling the impact of normal loading on tire deformation and distinguishing between the effective rolling radius r_e and loaded tire radius r. In general, the two radii are nonlinear functions of tire wear, vertical loading, and speed [23]. The effective rolling radius is used in the calculation of wheel slip κ while the loaded radius determines the moment countering wheel rotation caused by tire force F_x.

  Also, a singularity exists in the definition of tire slip that prevents describing tire operation from a standstill. This may be remedied by defining tire slip using a differential equation as proposed in [23]; however, the definition introduces new tire dynamics without consideration of their impact on rolling resistance.

- The modeling of tire rolling resistance as a constant neglects its dependence on tire vertical loading, speed of travel, inflation pressure, and temperature. Rolling resistance increases proportionally with vertical loading while relatively recent work in [29] describes the dynamic changes in rolling resistance associated with wheel speed, temperature, and pressure. Studies should be conducted to determine the relevance of rolling resistance transients in the design of an energy efficient traction control system.

- The adhesion gradient of a tire cannot be directly measured as assumed in this work. Therefore, the design of a practical adhesion gradient controller should include the design of an adhesion gradient observer as proposed in [12].

- Since an adhesion gradient controller does not rely on a tire specific model, it cannot operate a tire at optimal κ^* where $A = 0$. The application of a corresponding torque T_a at $A = 0$ would require a calculation involving a tire force generation model. Future work may investigate the use of an integral controller in conjunction with an adhesion gradient controller to gradually increase T_a beyond those prescribed by the adhesion gradient controller and up to the amount T_{ref} requested by the driver while $A > 0$.

- The vehicle and tire models may be extended to allow system optimization with respect to cornering maneuvers. It is an open area of research to consider the modeling of tire rolling resistance during cornering maneuvers.

- Ultimately, experimental results should be used to validate simulations of the traction control system model. Initial experiments may involve the use of an external, rotational-loading motor to emulate tire force generation characteristics as proposed in [12] and [13]. Experiments may then progress towards in-vehicle testing similar to those in [10].

The following consideration of future work related to rule base encoding, Pareto front optimization, automated GA parameter control, and plant/controller design coupling pertain to the improvement and extension of the proposed simultaneous plant/controller optimization process:

- Currently, the specialized crossover and mutation operations of the novel encoding of the fuzzy logic rule base depend on a precomputed Hasse diagram that indicates the relationship of members within a poset by delta inclusion. A potential weakness exists in the storage requirements for the Hasse digram since the unconstrained search space size increases exponentially with the number of rule base entries. It is proposed to investigate the derivation of an analytical expression to calculate the number of members within a poset by delta inclusion to determine its space complexity. Such an expression may also support the decision to redesign the crossover and mutation operations to dynamically compute required portions of the Hasse diagram to minimize memory requirements.

- An open research problem exists in the encoding and definition of genetic operations that permit the minimization of the number of rules in a rule base. The challenge lies in determining suitable building blocks that may be preserved during the crossover between, and mutation to, matrices of different sizes.

- Multiple simulation runs are used in the current work to produce a variety of optimized traction control systems corresponding to different weightings in energy use and performance objectives. These systems approximate a Pareto set and their analysis reveals limitations in system design. Thus, determination of a Pareto set of a multi-objective optimization problem provides valuable design guidance. Rather than generating a Pareto set through repeated execution of the GA by manually choosing different weightings between objectives, it is proposed to consider a non-aggregating approach that directly considers the optimization of multiple objectives. This may be achieved, for example, by modifying the selection routine to create a Niched Pareto GA to spread population members along the Pareto frontier in a single optimization run as described in [63].

- Due to the relatively small size of the optimization problem considered in this work, the GA parameters including mutation rate, population size, and number of generation before termination are chosen using a trial procedure to allow a satisfactory exploration of the search space. In general, the computational requirements required to optimize a larger problem may preclude the convenience of executing trial optimization runs for the purpose of manually tuning the GA parameters. For such problems, the incorporation of automated, dynamic control of the GA parameters during the GA execution is recommended. An overview of parameter control mechanisms is provided in [64].

- Only a partial coupling between plant and controller design variables exists in this work since the traction controller design does not impose constraints on the vehicle design. In general, a full coupling loop exhibiting mutual dependence between plant and controller variables is expected in a simultaneous design problem. In contrast to the ideal concurrent variable selection strategy, the all at once strategy allows a relaxation in either the plant to controller constraints or the controller to plant constraints to facilitate system optimization. Implementation of a penalty scheme within the GA is recommended to minimize deviation from the optimal condition with coupling constraint equality. The traction control system optimization problem may be extended to introduce a full coupling loop by incorporating a motor design as demonstrated in [17] where the specification of minimum motor power is derived from dynamic controller performance and controller performance is in turn dependent on the armature resistance and inductance derived from a motor design.

Appendix A

Traction Control System Parameters

Tables A.1 to A.4 list the numerical values used in the model of the traction control system.

Table A.1: Single-Wheel Vehicle Parameters

Component	Parameter	Value	Source$^{(1)}$
Vehicle	Mass	$m = 600$ kg	[23]
	Wheel Inertia	$J_w = 1$ kg m^2	[23]
	Wheel Radius	$r = 0.3$ m	[23]
	Initial Velocity	$v_{xo} = 1$ m/s	-
	Initial Wheel Speed	$\Omega_o = v_{xo}/r$	-

Table A.2: Drive Cycle Parameters

Component	Parameter	Value	Source$^{(1)}$
Road	Transition Distance	$S_T = 10$ m	-
	Final Distance	$S_F = 20$ m	-
	Torque Reference	$T_{ref} = 600$ N m	-
	Maximum Torque	$T_{max} = 720$ N m	-
	Dry Friction	$\mu_{dry} = 1.0$	-
	Wet Friction	$\mu_{wet} = 0.2$	-

Table A.3: Tire Parameters

Component	Parameter	Value	Source$^{(1)}$
Tire (Common)	Shape Factor	$C_x = 1.6$	[23]
	Curvature Factor	$E_x = 0$	[23]
	Nominal Load	$F_{zo} = 3000$ N	-
Tire 0 (Performance)	Stiffness Factor	$B_{xo} = 12.5$	[23]
	Peak Factor	$D_{xo} = 3000$ N	[23]
	Rolling Resistance Coefficient	$C_r = 0.014$	[22]
Tire 1 (Low Rolling Resistance)	Stiffness Factor	$B_{xo} = 9.375$	-
	Peak Factor	$D_{xo} = 2400$ N	-
	Rolling Resistance Coefficient	$C_r = 0.007$	[22]

Table A.4: Fuzzy Logic Controller Parameters

Component	Parameter	Value	Source
Torque Request Fuzzy Set	Small	S = 0	[12]
	Medium-Small	MS = 1/3	[12]
	Medium-High	MH = 2/3	[12]
	High	H = 1	[12]
Adhesion Gradient Fuzzy Set	Negative	N = 0	-
	Zero	Z = 0.1	-
	Positive-Small	PS = 0.5, A-PS $\in [0.2, 0.8]$ *	-
	Positive-High	PH = 0.9	-
Torque Output Fuzzy Set	Small	S = 0	[12]
	Medium-Small	MS = 1/3	[12]
	Medium-High	MH = 2/3	[12]
	High	H = 1	[12]
Fuzzy Rule Base IF (T_{ref}, A) is (Parameter) THEN T_a is (Value)	(S, N)	S	[12]
	(S, Z)	S	[12]
	(S, PS)	S	[12]
	(S, PH)	S	[12]
	(MS, N)	S	[12]
	(MS, Z)	$\mathcal{C}_1$ =S *	[12]
	(MS, PS)	$\mathcal{C}_2$ =MS *	[12]
	(MS, PH)	MS	[12]
	(MH, N)	S	[12]
	(MH, Z)	$\mathcal{C}_3$ =MS *	[12]
	(MH, PS)	$\mathcal{C}_4$ =MH *	[12]
	(MH, PH)	MH	[12]
	(H, N)	S	[12]
	(H, Z)	$\mathcal{C}_5$ =MS *	[12]
	(H, PS)	$\mathcal{C}_6$ =MH *	[12]
	(H, PH)	H	[12]

* Nominal value for testing. Parameter is a variable during optimization.

Appendix B

Longitudinal Bicycle Model

Analysis of a longitudinal bicycle model provides insight into the dynamic weight transfer between front and rear tires associated with vehicle acceleration. The analysis is used to derive conditions on vehicle design parameters and operation that enable the assumption of negligible weight transfer between tires.

Longitudinal and vertical vehicle forces are considered in a planar **longitudinal bicycle model** (Figure B.1) to determine the impact of vehicle acceleration on vertical weight distribution between tires. The vehicle is modeled as a rigid body with mass m, center of gravity located a distance h above a horizontal road, front and rear tire-road contact points located at distances l_1 and l_2, respectively, from the center of gravity, and wheelbase $l = l_1 + l_2$. Half of the physical vehicle mass is

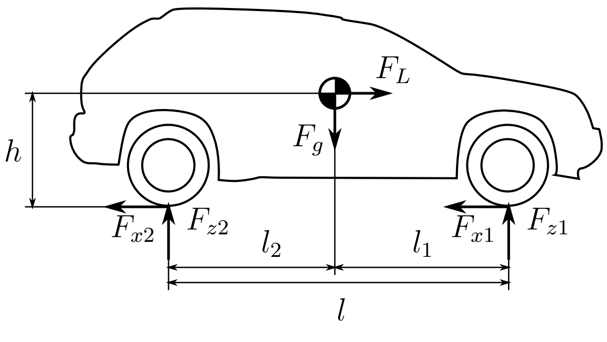


Figure B.1: Longitudinal Bicycle Model During Braking [23] modeled in m to account for the fact that half of the wheels are modeled.

Front and rear normal forces F_{z1} and F_{z2}, respectively, are present at the tire-road contact points in reaction to gravitational force:

$$F_g = mg = F_{z1} + F_{z2} \tag{B.1}$$

where $g \approx 9.81$ m/s^2 is the gravitational constant. Front and rear longitudinal tire forces F_{x1} and F_{x2}, respectively, arise due to longitudinal reaction forces produced by the road on the tires as a result of propulsion or braking action. The direction of the longitudinal forces is positive (points forward) during propulsion and negative (points backwards) during braking. In Figure B.1, the longitudinal forces have been indicated for braking. Though tire force generation mechanics (such as wheel rotation and tire deformation) are not explicitly modeled, the maximum longitudinal force available from each tire can still be determined if it is assumed that it is friction-limited by the amount of normal force exerted on the tire (as is the case with powerful propulsion and braking systems). For a road with uniform coefficient of friction μ, the longitudinal and normal tire forces are related as follows:

$$|F_{x1}| \leq |F_{x1\,\mathrm{max}}| = \mu F_{z1} \qquad\qquad |F_{x2}| \leq |F_{x2\,\mathrm{max}}| = \mu F_{z2} \tag{B.2}$$

Analysis of the longitudinal bicycle model provides the following insight on the limits of vehicle and tire longitudinal performance:

- Vehicle longitudinal acceleration is limited by surface friction. The vehicle acceleration a is determined by Newton's second law:

  $$a = \frac{F_x}{m} \tag{B.3}$$

  where $F_x = F_{x1} + F_{x2}$ is the net external longitudinal force applied by road on the front and rear tires.

  However, the longitudinal force F_x acting on the vehicle is limited by (B.2) since it is the sum of friction-limited longitudinal tire forces:

  $$|F_x| \leq |F_{x1\,\mathrm{max}}| + |F_{x2\,\mathrm{max}}| = \mu(F_{z1} + F_{z2}) = \mu F_g. \tag{B.4}$$

Therefore, by substitution of (B.1) and (B.3) into (B.4), vehicle acceleration is limited to:

$$|a| \leq \mu\, g. \tag{B.5}$$

Hence, assuming negligible aerodynamic drag and downforce [65], maximum vehicle acceleration is limited by the gravitational constant and the coefficient of friction μ between tire and road.

- Longitudinal tire forces F_{x1} and F_{x2} are limited by dynamic variation of normal forces as a function of vehicle longitudinal acceleration. Tire normal forces may be analyzed by applying d'Alembert's Principle [66] to convert the vehicle longitudinal dynamics in (B.3) into a static system by reformulating Newton's second law:

$$F_x = ma \;\leftrightarrow\; F_x + F_L = 0 \tag{B.6}$$

where $F_L = -ma$ is a fictitious **inertial force** acting longitudinally on the vehicle center of gravity in the opposite direction of vehicle acceleration as indicated in Figure B.1.

Tire normal forces are calculated by considering the sum of moments produced by gravitational, inertial, and normal forces about the rear and front tire-road contact points for the front and rear tires, respectively:

$$F_{z1} = F_g \frac{l - l_1}{l} + \frac{F_L \times h}{l} \tag{B.7a}$$

$$F_{z2} = \underbrace{F_g \frac{l_1}{l}}_{\text{static}} - \underbrace{\frac{F_L \times h}{l}}_{\text{dynamic}}. \tag{B.7b}$$

The first terms of both (B.7a) and (B.7b) represent the contribution to normal force as a result of static weight distribution determined by the longitudinal location of the vehicle center of gravity. The second terms represent the contribution to normal force caused by dynamic **weight transfer** resulting from vehicle acceleration and deceleration. Although the individual front and rear tire normal forces vary, the total normal force remains in balance with gravitational force (as seen by substituting (B.7) into (B.1)). Therefore the limits of longitudinal tire performance in (B.2) are dynamic and vary with vehicle acceleration as described by (B.7).

By assuming a low center of gravity (small h), long wheel base (large l), and low-to-moderate accelerations (small F_L), the dynamic weight transfer terms in (B.7) can be neglected. Therefore, the normal forces acting on a tire are determined solely by the static weight distribution terms in (B.7) and the assumptions allow longitudinal tire performance to be considered independently of vehicle acceleration.

Appendix C

Tire Operating Conditions

Additional background on general tire operating conditions and associated forces and moments is provided to establish a basis for future consideration of a vehicle model incorporating dynamics due to vehicle acceleration and cornering maneuvers.

In general, the operating conditions, forces, and moments of a tire may be described using the coordinate system and parameters shown in Figure C.1. The road surface is defined relative to a fixed inertial reference coordinate system at point O. The position of the center of the wheel is defined by point A relative to point O. At a radial distance r from A, the tire is in contact with an area of the road surface called the **contact patch**. All forces and moments required to move a vehicle are generated within the contact patch and assumed to act on the **contact center** at point C [23]. The road surface near the contact patch is approximated by a flat **ground plane** with normal vector n pointing upwards. The wheel rotates about a wheel axle s with angular speed Ω. The wheel rotation occurs within a **wheel center plane** passing through the wheel center A and the contact center C and perpendicular to the wheel axle s.

The orientation of the wheel is defined relative to a local coordinate system at point C on the road surface. The x-axis points forward along the line of intersection between the ground plane and the wheel center plane. The y-axis points right along the ground plane perpendicular to the x-axis. The z-axis points downwards normal to the ground plane ($-n$). The wheel velocity, and hence the velocity v_C of the

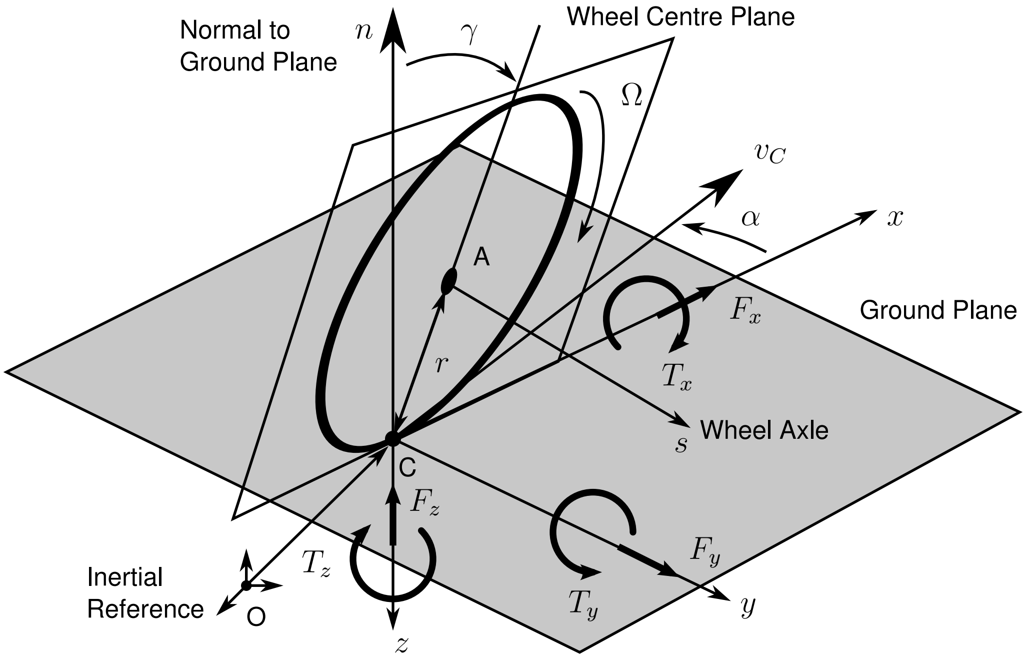


Figure C.1: Position, Motion, Forces, and Moments for Road and Tire [23]

contact center may differ from the direction in which the tire is pointed by a **slip angle** α from the x-axis. The wheel center plane may exhibit a **camber angle** γ, relative to the ground plane normal n. Longitudinal, lateral, and normal forces (F_x, F_y, F_z), and overturning, rolling resistance, and aligning moments (T_x, T_y, T_z) are defined relative the x, y, and z axes, respectively. The positive direction of the forces and moments are defined according to right hand convention with the exception of the vertical force F_z that acts in the same direction as the ground plane normal n.

Note. Sign conventions in the literature are not uniform and may be chosen, for example, to minimize the number of negative quantities in a specific application [23]. The sign convention presented in Figure C.1 is an adapted form of the Society of Automotive Engineers (SAE) convention and differs in the definition of positive side slip angle α. Other conventions, such as taken by the International Standards Organization (ISO), orient the wheel coordinate system such that the (x, y, z) axes

Table C.1: Tire Operating Modes [23]

	In-Plane Mode (Negligible α, γ, Variable Ω, v_x)	**Out-of-Plane Mode** (Constant Ω, v_x, Variable α, γ)
Primary function	Braking/Driving (F_x) Load Carrying (F_z)	Cornering Side Force (F_y)
Secondary effect	Rolling Resistance (T_y)	Overturning Moment (T_x) Aligning Moment (T_z)

point forward, left, and upwards, respectively [23].

The definition of the wheel center plane allows a distinction between in-plane, out-of-plane operating modes of a tire [23]. The two modes of operation are summarized in Table C.1. In the case of straight-line driving or braking, tire camber γ and slip angle α are assumed negligible and analysis may focus on the generation of in-plane longitudinal force F_x, normal force F_z, and rolling resistance moment T_y. In the case of simple cornering maneuvers, the wheel rotational speed Ω and the forward speed v_x (the x-component of the contact center velocity v_C) are assumed constant and analysis may focus on the generation of out-of-plane lateral force F_y, overturning moment T_x, and aligning moment T_z. For complex, and general, situations such as braking in a turn, a nonlinear combination of in-plane and out-of-plane tire operating modes results [23].

Table C.1 also shows that a distinction can be made between the primary functions and secondary (but often important) effects in the operation of a tire [23]. The primary functions of a tire are to transmit forces from the road to a vehicle in the three perpendicular directions (F_x, F_y, F_z) to affect vehicle speed and direction while absorbing road irregularities, respectively. Secondary moments (T_x, T_y, T_z) arise due to large slip angles inducing vehicle roll-over, energy lost in tire deformation, and uneven lateral force distribution within the contact patch area creating a moment about the contact center, respectively [23].

Appendix D

Simulation Convergence Data

Table D.1 lists the results of simulation convergence tests to determine solver settings in Matlab and Octave using a closed loop traction control system configured with a performance oriented tire and nominal traction controller parameters.

Solver settings are listed in the maximum step, relative tolerance, and absolute tolerance columns. Wall clock time is the actual time taken to run a 10 second simulation of the system on an AMD Sempron 1.60 GHz (Matlab) and on an AMD Opteron 2.20 GHz node available on SHARCNET (Octave). Data in the energy and time objective columns represent state variables of the system under consideration and are affected by solver settings.

Table D.1: Numerical Solver Convergence Data

Solver	**Max. Step**	**Rel. Tol.**	**Abs. Tol.**	**Wall Clock (s)**	**Energy Objective (J)**	**Time Objective (s)**
Matlab R2006a Stiff Solver						
ode15s	1e-1	1e-4	1e-4	0.56	25139.86	3.525445
ode15s	1e-1	1e-5	1e-5	0.69	25134.68	3.524545
ode15s*	1e-1	1e-6	1e-6	0.84	25134.17	3.524456
ode15s	1e-1	1e-7	1e-7	1.02	25134.08	3.524441
ode15s	1e-1	1e-8	1e-8	1.26	25134.09	3.524441
Octave 2.1.73 Stiff Solver						
lsode	1e-1	1e-4	1e-4	1.23	25135.63	3.524222
lsode	1e-1	1e-5	1e-5	1.67	25133.90	3.524351
lsode*	1e-1	1e-6	1e-6	2.17	25134.08	3.524431
lsode	1e-1	1e-7	1e-7	2.45	25134.09	3.524441
lsode	1e-1	1e-8	1e-8	3.24	25134.09	3.524441

*Selected solver settings

Appendix E

Steady State and Limit Condition Calculations

The following Maple 9.5 code listing is used to calculate steady-state and limit operating conditions of the open loop control of tire and vehicle dynamics. Numerical results are discussed in Section 6.2.

```
>  restart;
>  Common:=Cx=1.6, Fzo=3000, r=0.3, m=600, J=1;
>  TireA:= Bxo = 12.5, Cx = 1.6, Dxo = 3000, Cr=0.014, Common;
>  ur_uo_dry:=1; ur_uo_wet:=0.2;
>  Fx := ur_uo * Dxo * sin(Cx*arctan(Bxo*k/ur_uo));
>  muk := Fx/Fzo;
>  Ty := r*Cr*Fzo;
>  A := diff(Fx,k) / (Bxo*Cx*Dxo);
```

$$\mathit{Common} := \mathit{Cx} = 1.6,\ \mathit{Fzo} = 3000,\ r = 0.3,\ m = 600,\ J = 1$$

$$\begin{aligned}\mathit{TireA} := {} & \mathit{Bxo} = 12.5,\ \mathit{Cx} = 1.6,\ \mathit{Dxo} = 3000,\ \mathit{Cr} = 0.014,\\ & \mathit{Cx} = 1.6,\ \mathit{Fzo} = 3000,\ r = 0.3,\ m = 600,\ J = 1\end{aligned}$$

$$\mathit{ur\_uo\_dry} := 1$$

$$\mathit{ur\_uo\_wet} := 0.2$$

$$\mathit{Fx} := \mathit{ur\_uo}\,\mathit{Dxo}\,\sin\left(\mathit{Cx}\,\arctan\left(\frac{\mathit{Bxo}\,k}{\mathit{ur\_uo}}\right)\right)$$

$$\mathit{muk} := \mathit{ur\_uo}\,\mathit{Dxo}\,\sin\left(\mathit{Cx}\,\arctan\left(\frac{\mathit{Bxo}\,k}{\mathit{ur\_uo}}\right)\right)\mathit{Fzo}^{-1}$$

$$\mathit{Ty} := r\,\mathit{Cr}\,\mathit{Fzo}$$

$$A := \cos\left(\mathit{Cx}\,\arctan\left(\frac{\mathit{Bxo}\,k}{\mathit{ur\_uo}}\right)\right)\left(1 + \frac{\mathit{Bxo}^2 k^2}{\mathit{ur\_uo}^2}\right)^{-1}$$

```
> ss := r*(1-k)*(Ta-Ty-r*Fx)/J = Fx/m;
```

$$\begin{aligned}\mathit{ss} := {} & r\,(1-k)\left(\mathit{Ta} - r\,\mathit{Cr}\,\mathit{Fzo} - r\,\mathit{ur\_uo}\,\mathit{Dxo}\,\sin\left(\mathit{Cx}\,\arctan\left(\frac{\mathit{Bxo}\,k}{\mathit{ur\_uo}}\right)\right)\right)J^{-1}\\ & = \mathit{ur\_uo}\,\mathit{Dxo}\,\sin\left(\mathit{Cx}\,\arctan\left(\frac{\mathit{Bxo}\,k}{\mathit{ur\_uo}}\right)\right)m^{-1}\end{aligned}$$

```
> Ta_nTCSu:=600;
> k_dryss:=fsolve(subs(Ta=Ta_nTCSu, ur_uo=ur_uo_dry, TireA, ss),k);
```

$$\mathit{Ta\_nTCSu} := 600$$

$$\mathit{k\_dryss} := 0.03711144962$$

```
> Fx_dryss:=evalf(subs(k=k_dryss, ur_uo=ur_uo_dry, TireA, Fx));
> muk_dryss:=evalf(subs(k=k_dryss, ur_uo=ur_uo_dry, TireA, muk));
> A_dryss:=evalf(subs(k=k_dryss, ur_uo=ur_uo_dry, TireA, A));
```

$$\mathit{Fx\_dryss} := 1921.053802$$

$$\mathit{muk\_dryss} := 0.6403512675$$

$$\mathit{A\_dryss} := 0.6320640207$$

```
> muk_wetlim:=evalf(subs(k=1, ur_uo=ur_uo_wet, TireA, muk));
> A_wetss:=evalf(subs(k=1, ur_uo=ur_uo_wet, TireA, A));
```

$$\mathit{muk\_wetlim} := 0.1216598993$$

$$\mathit{A\_wetss} := -0.0002031371373$$

Bibliography

[1] Intergovernmental Panel on Climate Change. *Climate Change 2001: Synthesis Report*. Cambridge University Press, Cambridge, UK, 2001. [Online]. Available: `http://www.grida.no/climate/ipcc_tar`, Oct. 2006 [date accessed].

[2] *World Energy Outlook: 2004 Edition*. International Energy Agency, Paris Cedex, France, 2004. [Online]. Available: `http://www.iea.org/`, Oct. 2006 [date accessed].

[3] International Energy Agency. Key world energy statistics: 2006 edition, 2006. [Online]. Available: `http://www.iea.org`, Oct. 2006 [date accessed].

[4] Intergovernmental Panel on Climate Change. *Climate Change 2001: The Scientific Basis*. Cambridge University Press, Cambridge, UK, 2001. [Online]. Available: `http://www.grida.no/climate/ipcc_tar/`, Oct. 2006 [date accessed].

[5] *International Energy Outlook: 2006*. Energy Information Administration, U.S. DOE, Washington, DC, June 2006. [Online]. Available: `http://www.eia.doe.gov/oiaf/ieo/`, Oct. 2006 [date accessed].

[6] M. Montazeri-Gh and M. Asadi. 2006-01-3293: Influence of the road grade on the optimization of fuzzy-based hybrid electric vehicle control strategy. In *Powertrain & Fluid Systems Conference & Exhibition*, Oct. 16-19, 2006.

[7] O. Volpato, F. Theunissen, and R. Mazara. 2005-01-3777: Engine control module for flex fuel plus compressed natural gas vehicles. In *Powertrain & Fluid Systems Conference and Exhibition*. SAE, Oct. 24-27, 2005.

[8] N. Brinkman, M. Wang, T. Weber, and T. Darlington. *GM Study: Well-to-Wheels Analysis of Advanced Fuel/Vehicle Systems - A North American Study of Energy Use, Greenhouse Gas Emissions, and Criteria Pollutant Emissions.* Argonne National Laboratory, 2005. [Online]. Available: `http://www.transportation.anl.gov/pdfs/TA/339.pdf`, Oct. 2006 [date accessed].

[9] C. C. Chan and Y. S. Wong. The state of the art of electric vehicles technology. In *The 4th International Power Electronics and Motion Control Conference, 2004. IPEMC 2004.*, pages 46–57, 14-16 Aug. 2004.

[10] X. Liu, L. Li, Y. Hori, T. Akiba, and R. Shirato. Optimal traction control for EV utilizing fast torque response of electric motor. In *Industrial Electronics Society, 2005. IECON 2005. 32nd Annual Conference of IEEE*, pages 2614–2619, Nov. 6-10 2005.

[11] Y. Hori. Future vehicle driven by electricity and control-research on four-wheel-motored "UOT Electric March II". *Industrial Electronics, IEEE Transactions on*, 51(5):954–962, Oct. 2004.

[12] V. Delli Colli, G. Tomassi, and M. Scarano. "Single wheel" longitudinal traction control for electric vehicles. *Power Electronics, IEEE Transactions on*, 21(3):799–808, May 2006.

[13] P. Khatun, C. M. Bingham, N. Schofield, and P. H. Mellor. Application of fuzzy control algorithms for electric vehicle antilock braking/traction control systems. *Vehicular Technology, IEEE Transactions on*, 52(5):1356–1364, Sept. 2003.

[14] N. Shidore and M. Pasquier. 2005-01-3457: Interdependence of system control component sizing for a hydrogen-fueled hybrid vehicle. In *Future Transporta-*

tion Technology Conference, Warrendale, PA, Sept 7-9 2005. SAE International.

[15] S. Golbuff. 2007-01-1545: Design optimization of a plug-in hybrid electric vehicle. In *2007 World Congress*, Warrendale, PA, April 16-19, 2007. SAE International.

[16] G. L. Ohl, M. J. Prucka, F. S. Kanafani, M. J. Duty, E. DiValentin, and D. M. Kramer. 2004-01-0893: Development of powertrain coordination at DaimlerChrysler Corporation. In *2004 SAE World Congress*, Warrendale, PA, March 8-11, 2004. SAE International.

[17] J. A. Reyer and P. Y. Papalambros. Combined optimal design and control with application to an electric DC motor. *Journal of Mechanical Design, Transactions of the ASME*, 124(2):183–191, June 2002.

[18] Y. He and J. McPhee. A design methodology for mechatronic vehicles: Application of multidisciplinary optimization, multibody dynamics and genetic algorithms. *Vehicle System Dynamics*, 43(10):697–733, October, 2005.

[19] W. Wang and J. J. Moskwa. 1999-01-0977: A study on automatic transmission system optimization using a HMMWV dynamic powertrain system model. In *International Congress and Exposition*, Warrendale, PA, March 1-4, 1999. SAE International.

[20] U. Kiencke and L. Nielsen. *Automotive Control Systems: For Engine, Driveline, and Vehicle.* Springer-Verlag Berlin Heidelberg, New York, 2000.

[21] S. Hegazy, H. Rahnejat, and K. Hussain. Multi-body dynamics in full-vehicle handling analysis under transient manoeuvre. *Vehicle System Dynamics*, 34(1):1–24, July 2000.

[22] *Tires and Passenger Vehicle Fuel Economy: Informing Consumers, Improving Performance*, volume Special Report #286. Transportation Research Board,

National Research Council, Washington, D.C., July 11, 2006. [Online]. Available: `http://www.trb.org/publications/sr/sr286.pdf`, Sept. 2006 [date accessed].

[23] H. B. Pacejka. *Tire and Vehicle Dynamics*. SAE International, Warrendale, PA, U.S.A., 2006.

[24] A. Eichberger and M. Schittenhelm. Implementations, applications and limits of tyre models in multibody simulation. *Vehicle System Dynamics*, 43(Suppl):18–29, 2005.

[25] *Tire Performance Technology - SAE J2047 Feb98*, volume 2 of *2002 SAE Handbook*, pages 30.01–30.24. SAE, Warrendale, PA, U.S.A., 2002.

[26] M. Lebel. 1999-01-0788: Multiple applications, multiple roles: How carbon black technologies help tires roll better. In *SAE International Congress and Exposition*, Warrendale, Pennsylvania, USA, March 1-4, 1999. SAE International.

[27] *Stepwise Coastdown Methodology for Measuring Tire Rolling Resistance - SAE J2452 JUN99*, volume 2 of *2002 SAE Handbook*, pages 30.36–30.57. SAE, Warrendale, PA, U.S.A., 2002.

[28] P. S. Grover. 980251: Modeling of rolling resistance test data. In *SAE International Congress and Exposition*, Warrendale, Pennsylvania, USA, Feb. 23-26, 1998. SAE International.

[29] L. Nielsen and T. Sandberg. 2002-01-1200: A new model for rolling resistance of pneumatic tires. In *SAE 2002 World Congress*, Warrendale, Pennsylvania, USA, Mar. 4-7, 2002. SAE International.

[30] P. S. Grover and S. H. Bordelon. 1999-01-0787: New parameters for comparing tire rolling resistance. In *SAE International Congress and Exposition*, Warrendale, Pennsylvania, USA, March 1-4, 1999. SAE International.

[31] *Rolling Resistance Measurement Procedure for Passenger Car, Light Truck, and Highway Truck and Bus Tires - SAE J1269 SEP00*, volume 2 of *2002 SAE Handbook*, pages 30.57–30.61. SAE, Warrendale, PA, U.S.A., 2002.

[32] J. Y. Wong. *Mechanics of Pneumatic Tires*, chapter 1, pages 14–15. Theory of Ground Vehicles. John Wiley, New York, 3 edition, Mar. 2001.

[33] California Energy Commission. *California State Fuel-Efficient Tire Program: Volume I - Summary of Findings and Recommendations.* Number Sept 23,. Jan. 31, 2003. [Online]. Available: `http://www.energy.ca.gov/reports/2003-01-31_600-03-001F-VOL1.PDF`, Sept. 2006 [date accessed].

[34] A. J. C. Schmeitz, I. J. M. Besselink, J. De Hoogh, and H. Nijmeijer. Extending the Magic Formula and SWIFT tyre models for inflation pressure changes. *VDI Berichte*, (1912):201–225, 2005. [Online]. Available: `http://www.mate.tue.nl/mate/pdfs/5852.pdf`, Apr. 2007 [date accessed].

[35] R. Pusca, Y. Ait-Amirat, A. Berthon, and J. M Kauffmann. Fuzzy-logic-based control applied to a hybrid electric vehicle with four separate wheel drives. In *IEE Proceedings: Control Theory and Applications*, volume 151, pages 73–82, January, 2004.

[36] A. Mirzaei, M. Moallem, B. M. Dehkordi, and B. Fahimi. Design of an optimal fuzzy controller for antilock braking systems. *Vehicular Technology, IEEE Transactions on*, 55(6):1725–1730, Nov. 2006.

[37] K. Furukawa and Y. Hori. Recent development of road condition estimation techniques for electric vehicle and their experimental evaluation using the test EV "UOT March I and II". In *Industrial Electronics Society, 2003. IECON '03. The 29th Annual Conference of the IEEE*, volume 1, pages 925–930, Nov. 2-6 2003.

[38] T. J. Ross. *Fuzzy Logic: With Engineering Applications.* John Wiley & Sons, Ltd., West Sussex, England, 2004.

[39] F. Jinzhi, Y. Fan, and L. Jun. 2002-01-0983: An investigation to controller design for active vehicle suspension by using GA-based PID and fuzzy logic. In *SAE 2002 World Congress*, Mar. 4-7, 2002.

[40] D. Peng, C. L. Yin, and J. W. Zhang. 2006-01-3583: Advanced braking control system for hybrid electric vehicle using fuzzy control logic. In *Commercial Vehicle Engineering Congress and Exhibition*, Oct. 31-Nov. 2, 2006.

[41] M. Davoudi, M. B. Menhaj, and M. Davoudi. 2006-01-3483: A fuzzy based vehicle dynamic stability control (FDSC). In *Commercial Vehicle Engineering Congress and Exhibition*, Oct. 31-Nov. 2, 2006.

[42] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach.* Pearson Education, Inc., Upper Saddle River, New Jersey, 2003.

[43] The MathWorks. Fuzzy logic toolbox: The fuzzy inference diagram, 2006. [Online]. Available: `http://www.mathworks.com/access/helpdesk/help/toolbox/fuzzy/`, Nov. 2006 [date accessed].

[44] C. W. Chen. Stability conditions of fuzzy systems and its application to structural and mechanical systems. *Advances in engineering software*, 37(9):624–629, September 2006.

[45] H. K. Fathy, J. A. Reyer, P. Y. Papalambros, and A. G. Ulsoy. On the coupling between the plant and controller optimization problems. In *2001 American Control Conference*, volume 3, pages 1864–1869. Institute of Electrical and Electronics Engineers Inc., Jun 25-27 2001.

[46] C. A. C. Coello. An updated survey of GA-based multiobjective optimization techniques. *ACM Computing Surveys*, 32(2):109–143, June 2000.

[47] Y. Carson and A. Maria. Simulation optimization: Methods and applications. In *Winter Simulation Conference Proceedings*, pages 118–126, Dec 7-10 1997.

[48] K. F. Man, K. S. Tang, and S. Kwong. *Genetic Algorithms.* Springer-Verlag London Ltd., Great Britain, 1999.

[49] D. F. Nehab and M. A. C. Pacheco. Schemata theory for the real coding and arithmetical operators. In *Applied Computing 2004 - Proceedings of the 2004 ACM Symposium on Applied Computing*, pages 1006–1012, Mar 14-17 2004.

[50] S. Ronald, S. Kirkby, and P. Eklund. Multi-chromosome mixed encodings for heterogeneous problems. In *Proceedings of the 1997 IEEE International Conference on Evolutionary Computation, ICEC'97*, pages 37–42, Apr 13-16 1997.

[51] H. A. Mayer and M. Spitzlinger. Multi-chromosomal representations and chromosome shuffling in evolutionary algorithms. In *The 2003 Congress on Evolutionary Computation, 2003. CEC '03.*, volume 2, pages 1145–1149, Dec 8-12 2003.

[52] Z. Michalewicz. A survey of constraint handling techniques in evolutionary computation methods. In *Evolutionary Programming IV. Proceedings of the Fourth Annual Conference on Evolutionary Programming*, pages 135–155, Cambridge, MA, USA, March 1-3 1995. MIT Press.

[53] S. Ben Hamida and A. Petrowski. Need for improving the exploration operators for constrained optimization problems. In *Proceedings of the 2000 Congress on Evolutionary Computation*, volume 2, pages 1176–1183. IEEE, Jul 16-Jul 19 2000.

[54] S. Hayden and J. F. Kennison. *Zermelo-Fraenkel Set Theory.* Charles E. Merrill Publishing Company, Columbus, Ohio, 1968.

[55] W. M. Wonham. *Supervisory Control of Discrete-Event Systems: ECE 1636F/1637S 2005-06.* Jul. 1, 2005. [Online]. Available: `http://www.control.toronto.edu/people/profs/wonham/wonham.html`, Jan. 2006 [date accessed].

[56] The MathWorks, 2007. [Online]. Available: `http://www.mathworks.com/`, Apr. 2007 [date accessed].

[57] J. W. Eaton. GNU Octave, 2006. [Online]. Available: `http://www.gnu.org/software/octave/`, Apr. 2007 [date accessed].

[58] Shared Hierarchical Academic Research Computing Network, 2006. [Online]. Available: `http://www.sharcnet.ca/`, Apr. 2007 [date accessed].

[59] Maplesoft, 2007. [Online]. Available: `http://www.maplesoft.com/`, Apr. 2007 [date accessed].

[60] L. M. San Jose-Revuelta. A new adaptive genetic algorithm for fixed channel assignment. *Information Sciences*, 177(13):2655–2678, Jul 1, 2007.

[61] R. Hinterding. Gaussian mutation and self-adaption for numeric genetic algorithms. In *Proceedings of the 1995 IEEE International Conference on Evolutionary Computation*, pages 384–388. IEEE, Nov 29-Dec 1 1995.

[62] I. Y. Kim and O. L. De Weck. Adaptive weighted sum method for multiobjective optimization: A new method for pareto front generation. *Structural and Multidisciplinary Optimization*, 31(2):105–116, Feb., 2006.

[63] J. Horn, N. Nafpliotis, and D. E. Goldberg. A niched Pareto genetic algorithm for multiobjective optimization. In *Evolutionary Computation, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the First IEEE Conference on*, volume 1, pages 82–87, June 27-29 1994.

[64] A. E. Eiben, R. Hinterding, and Z. Michalewicz. Parameter control in evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 3(2):124–141, Jul 1999.

[65] N. J. Mckay and A. Gopalarathnam. 2002-01-3294: The effects of wing aerodynamics on race vehicle performance. In *Proceedings of the 2002 SAE Motorsports Engineering Conference and Exhibition (P-382)*, December 2-5, 2002.

[66] J. H. Ginsberg. *Advanced Engineering Dynamics*. Cambridge University Press, Cambridge, U.K., 1998.