Department of Mobility Systems Engineering
TUM School of Engineering and Design
Technical University of Munich

Calibration of Car Following Model with Genetic Algorithm and Particle Swarm Optimization methods

A thesis presented in part fulfilment of the requirements of the Degree of Bachelor of Science in Environmental Engineering at the Department of Civil and Environmental Engineering, Technical University of Munich.

Supervisor M.Sc. Qinglong Lu
Univ.-Prof. Dr. Constantinos Antoniou
Chair of Transportation Systems Engineering

Submitted by He, Yuting
03716058
Arcisstrasse 21
80333 München

Submitted on München, 16.10.2022

Abstract

Car following behaviour is one of the main behaviours in microscopic traffic flows and is therefore an important part of traffic flow modelling. Various car-following models have been proposed based on respective following principles, e.g., keeping desired space gap, resulting in different sets of model parameters. Moreover, traffic flows usually illustrate distinct characteristics under very different traffic scenarios. Car-following models thus should be specifically tuned to fit the traffic flow in concern, in particular the following principle and model parameters, such that they can be used to simulate the traffic flow with high accuracy and assist the analysis and evaluation of traffic management and control measures. Considering the complicated vehicle interactions around the highway entrances and exits areas, the thesis is to apply the genetic algorithm (GA) and particle swarm optimization (PSO) algorithm to calibrate car-following models for the highway entrances and exits traffic flows using a number of trajectory collections from Germany highway entrances and exists. Moreover, the heterogeneity in car-following behaviours of vehicle combination (i.e., car following car, car following truck, truck following car, truck following truck) is also specifically considered in the thesis. Wiedemann model is selected for calibration due to its superiority in theoretical logic and easy application in popular traffic simulators, such as VISSIM and SUMO. Furthermore, fundamental diagrams are extracted from the selected trajectories to validate the nonnegligible differences in the traffic flow at highway entrances and exits from urban traffic and freeway traffic. The experiment results show that PSO generally outperformed GA in terms of convergence speed and convergence results in the devised car-following calibration procedure. Importantly, the calibrated Wiedemann models can replicate the respective traffic with a relative percentage error of less than 10%.

Keywords: car-following model, Wiedemann model, genetic algorithm, particle swarm optimization, highway entrance/exit

Table of Contents

List of Figures

List of Tables

1. Introduction

1.1. Background and Motivation

The simulation of traffic flows presents an essential focus in the field of traffic control and planning. The traffic flow simulation reproduces the behaviour and characteristics of a natural traffic flow and provides a foundation for the implementation of traffic flow regulation and traffic planning.

Car-following models are an essential component of traffic flow simulation. It simulates the following behaviour of vehicles in their driving direction: the following vehicle adjusts its velocity and acceleration to the preceding vehicle's driving behaviour (Olstam & Tapani, 2004). The car-following model defines the interaction of drivers in the same lane of the traffic flow by defining the response of the target vehicle to the driving behaviour of the vehicle in front of it in the same lane with it (Brackstone & McDonald, 1999). Since the publication of the first car-following model (Pipes, 1953), various car-following models have been developed in accordance with different logical foundations, like stimulus-based models (Gazis et al., 1961), safety distance models (Gipps, 1981), optimal speed models (Bando et al., 1995), psycho-physical models (Wiedemann & Reiter, 1992), etc. Among these models, the Wiedemann model has the most well-established theoretical logic and has been proven as capable of simulating microscopic traffic flows under diverse traffic scenarios. Therefore, several mainstream traffic flow simulation software nowadays utilises the Wiedemann model as the foundation for car-following simulations.

The behavioural characteristics of microscopic traffic flows are susceptible to the influence of external factors. The typology of roadways, the composition type of microscopic traffic flows, the countries or regions situated, and time-of-day contribute to variations in traffic flow characteristics. By varying the parameter set settings of the model, the model is enabled to simulate traffic flows with different characteristics. To improve the simulation accuracy of the model, the parameter set of the model needs to be adjusted according to the traffic scenario being simulated.

The process of adjusting the parameter set of models by using observed data to improve the accuracy of the simulation is known as calibration. To reflect actual driver behaviour in the simulation, before a car-following model is used for microscopic traffic flow simulations, the parameters in the model should be calibrated. The calibration of car-following models usually uses real-world traffic trajectory data recorded in the traffic scenario with similar characteristics.

For the acquisition of real-world traffic trajectory data, several researchers have collected the data by installing sensors and cameras on vehicles (Pourabdollah et al., 2017; Zhu et al., 2018). However, as drivers differ individually in their driving styles and conventions, extrapolating the overall condition from the sample situation could easily be biased, especially with the insufficient sample. Therefore, some studies calibrated the car-following models by using datasets that record the overall traffic flow trajectory (He et al., 2015; Liu et al., 2021).

The interaction of vehicles surrounding highway entrances and exits is complicated. To ensure the safety of vehicles during travel, the simulation of traffic flow in this scenario is meaningful. The Wiedemann model was selected as the calibration model because of the complete logical basis and the possibility of direct application in mainstream microscopic traffic flow simulators such as SUMO and VISSIM. In this thesis, a recently published dataset recording traffic flow variations in the surroundings of entrances and exits in German highways was utilized for the calibration. Considering the influence of the heterogeneity of vehicle types on driving behaviour, in this study different types of following pairs are distinguished and calibrated separately. Because of the multi-parameter property of the Wiedemann model, the calibration was conducted by using genetic algorithm (GA) and particle swarm optimization (PSO) methods.

1.2. Research Questions and Objectives

The main objective of this study was to calibrate the car-following model of entrance/exit area in German highway and to obtain separate sets of parameters for different types of vehicles. Based on the above objectives, the following research questions are proposed:

1) What is the difference of highway entrance/exit traffic flow from the other traffic scenarios?

2) What is the difference between the following behaviour of different vehicle types?

3) Are the simulations using the calibrated model resembling real traffic situations?

1.3. Contribution

The main contributions of this thesis are threefold:

1) A newly published dataset is utilized to analyse the following behaviour of vehicles in a highway entrance/exit scenario and the car-following model is calibrated by using this dataset.

2) The behavioural differences between vehicle types were considered. The car-following data was filtered by different vehicle types and the respective parameter sets are calibrated separately.

3) The characteristics of the traffic trajectory data from the highway entrance/exit area were analysed utilizing the traffic flow fundamental diagram, which confirmed the non-negligible differences of traffic flow around the highway entrance/exit compared to the regular highway sections.

1.4. Thesis Structure

Chapter 1 of the thesis presents the background and motivation for the study, describes the study objectives, and provides an overview of the thesis in general. Chapter 2 reviews the previous literature in general and presents the current literature gaps in terms of the car-following model and the calibration methods. Chapter 3 focuses on the methodology used in this study. Chapter 4 introduces the dataset. In this chapter, traffic flow data is processed and analysed. Chapter 5 presents the results of the study as well as the analysis and discussion of the results. Chapter 6 provides an overall review and summary of the thesis and discusses the problems from the study as well as an outlook of possible further work. Figure 1 shows the framework of the thesis structure.

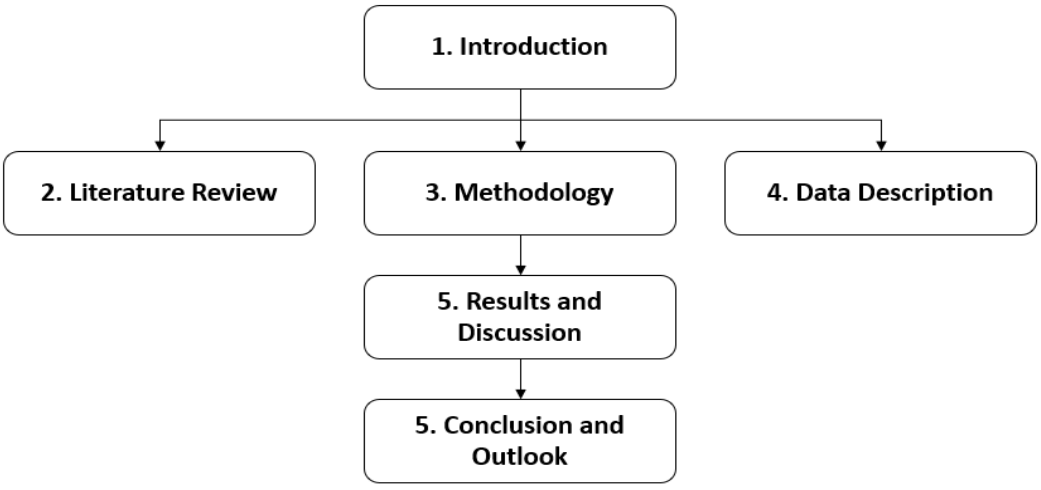


Figure 1 Thesis framework

2. Literature Review

In this chapter, the previous research relevant to the topic of this thesis is reviewed and summarised. Section 2.1 introduces the origins and development of the car-following model. In section 2.2 different car-following models are classified and briefly introduced. Sections 2.3 explains the reasons for calibrating the car-following model and the methods commonly used. These two methods are used in this thesis to calibrate the car-following model. Section 2.4 describes the current research gap.

2.1. Car-Following Models

Car-following models describe the following behaviour of drivers towards the vehicle in front of them in the traffic flow. Based on the rules of safe driving in traffic regulations, the theory of car-following was first proposed by Pipes in 1953 (Pipes, 1953). Using the car-following model, the driving behaviour of individual drivers can be simulated, which plays an important role in the study of microscopic traffic flows. The purpose of car-following model is to describe the influence of the leading vehicle on the driving behaviour of the vehicle behind during driving process from the individual level. The car-following process is one of the main processes in modern traffic flow theory. With the rise and development of autonomous driving, the importance of this type of model has increased further in recent years.

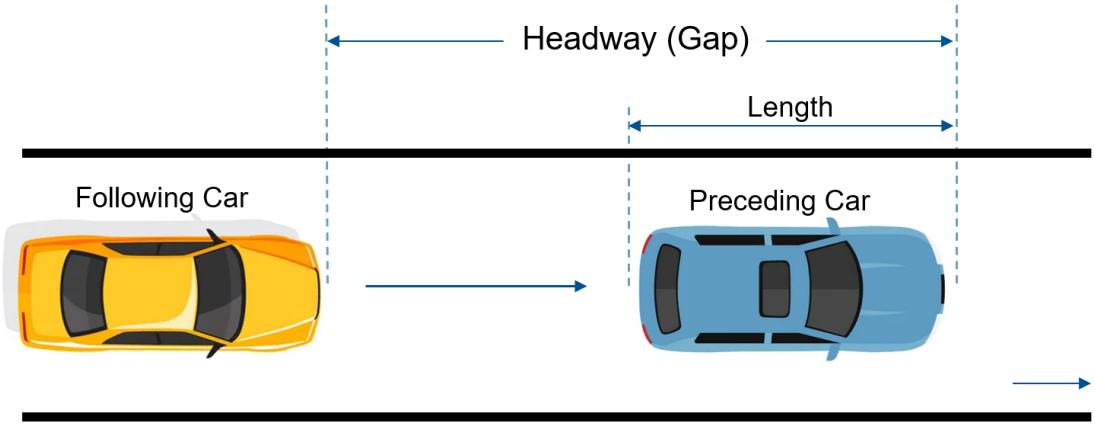


Figure 2.1 Typical illustration for car-following

Figure 2.1 illustrates a typical vehicle following scenario. The car-following behaviour studies the longitudinal travel process of the vehicle. As the following car, the driving behaviour of the yellow vehicle is modulated by the behaviour of the blue vehicle ahead in the same lane (Olstam & Tapani, 2004).

Depending on the regulation logic used, car-following models were classified into five categories. Table 1 shows a summary comparison of the categories. A brief introduction and comparison between these five types of models is given in the next subsection.

Table 2.1 Classification of car following models based on their logic (Ahmed et al., 2021)

Model Class	**Model logic and Assumption**	**Representative Model**
Stimulus based models	The stimulus-response function between the leader and follower vehicle: the acceleration of the follower is proportional to its speed, the speed difference between follower and leader, and the space headway.	Gazis-Herman-Rothery (GHR) model
Safety distance models	The follower always keeps a safe distance from the front vehicle.	Gipps model
Desired measure models	An ideal situation represented by certain measures (e.g., following distance and following speed) is assumed. The following driver is constantly attempting to eliminate the difference between the ideal and actual situation.	Intelligent driver model (IDM)
Optimal velocity models	The optimal safe velocity of following vehicle depends on the distance between follower and leader and the acceleration of the following vehicle is determined based on the difference between the actual speed and the optimal speed(Lazar et al., 2016).	Optimal velocity model (OVM)
Psycho-physical models	Thresholds are used to regulate the driving behaviour of following vehicle. The driver reacts to the vehicle when a threshold value for relative speed or spacing is reached.	Wiedemann model

2.2. Comparison of Different Car-Following Models

2.2.1. General Motors model and Gazis-Herman-Rothery model

The General Motors (GM) model is initial developed by General Motors research lab in 1958 and it was expanded and improved in 1961 (Gazis et al., 1961). The GM model might be the most widely known car-following model, and it provides the basis for all other car-following models with varying sensitivity parameters (Koutsopoulos & Farah, 2012). The original GM model is a linear model with a constant sensitive parameter and has a very simple structure, where the acceleration of the following vehicle is linearly related to the relative speed between it and the leading vehicles (Ahmed et al., 2021).

One of the reasons for the deviation between the basic GM model's simulation of the following vehicle's acceleration and the actual situation is that the impact of the spacing between the front and following vehicles on the stimulus is not considered. Based on this, an extended GM model was proposed by Gazis et al. (1961), which is also known as the Gazis-Herman-Rothery (GHR) model. In GHR model the acceleration equation is extended to incorporate traffic density as a sensitive term and proposes different sets of

parameters to decide the acceleration and deceleration. The GHR model also consider the non-linear behaviour in the sensitivity of the distance between the following and the leading vehicle and sensitivity of the relative speed of the following vehicle.

A large amount of work on the calibration of the GHR model was published between the 1960s and 1990s (Brackstone & McDonald, 1999). However, models based on GM model are currently used less frequently because of their own limitations in behaviour. Drivers respond to arbitrarily any small change in stimulus, e.g., the relative speed of the following vehicle to the leading vehicle is still influenced by its leading vehicle even at very large relative distances. When the relative speed is zero, the follower's response disappears. In case that the magnitude of fluctuations of acceleration increases, GHR model creates a larger headway than reality. Moreover, microscopic analysis from Rockwell and Treiterer confirms that the car-following behaviour is likely to vary with traffic and flow conditions (Rockwell et al., 1968), which is not considered in GM-based models.

2.2.2. Gipps model

Gipps model is the first general model based on safety distance logic. Unlike the GM-based models that use the stimulus-response equations, in the safety-distance model a safe following distance is always maintained between the following vehicle and the leading vehicle. Within a safe distance, a collision would be unavoidable if the driver of the lead vehicle acted “unpredictably” (Brackstone & McDonald, 1999). The basic rule of safety distance was quoted by Pipes in 1953 from the safe driving regulations of the California Motor Vehicle Code: "*A good rule for following another vehicle at a safe distance is to allow yourself the length of a car (about fifteen feet) for every ten miles per hour you are traveling*" (Pipes, 1953).

As an important development of safe-distance model, the Gipps model was proposed by Gipps in 1981. The Gipps model records the car-following behaviour when leading vehicle exists and the free-flowing behaviour of the vehicle when no vehicle is ahead. The following driver plans his or her speed for the following instant (i.e., after the time delay) to make sure he or she can stop safely even the leading vehicle break suddenly. In case that there is no vehicle in front of the driver, the driving speed can be obtained from the empirical formulations which combines two conditions: (a) the speed never exceeds the driver’s desired speed; and (b) the acceleration decreases as the speed increases and becomes zero until the desired speed is reached (Gipps, 1981).

Gipps model has been extended, modified, and calibrated by various researchers and is used in a variety of micro-simulation tools, such as AIMSUN, DRACULA, and SISTM for its simple calibration requirements (Barceló & Casas, 2005). However, one weakness of

the safe-distance model such as Gipps model is that the drivers' perception is not considered. Any minor changes might end in the reaction of the following driver (Lazar et al., 2016).

2.2.3. Intelligent Driver model

The intelligent driver model (IDM) is one of the most popular models using desired measures (Treiber et al., 2000). The IDM considers both the desired speed and the desired space headway. When the front vehicle is far away, the IDM performs as a free-flow model where the acceleration is controlled by the desired speed of the driver. The desired space headway in IDM is dependent on several factors: speed, speed difference, the maximum acceleration, a comfortable deceleration, the minimum spacing at the standstill situation, and the desired time headway. The IDM introduces a maximum acceleration and a comfortable deceleration rate, which prevents the model from producing unrealistically high acceleration or deceleration. This is a significant advantage of the IDM over earlier car-following models such as the GHR and Gipps models (Saifuzzaman & Zheng, 2014). The IDM was later extended by Treiber et al. in 2003. This extended model captures the effects of driver adaptation to the surrounding environment by using a memory function (Treiber & Helbing, 2003). This extension was based on the observation that after spending time in congested traffic, most drivers will adjust their driving style.

The main difficulty for models with desired measures such as IDM is that most parameters are inherently unobservable, which makes their estimation more challenging. As a result, many such models are not empirically estimated using real traffic data.

2.2.4. Optimal Velocity model

The instability of traffic flow can lead to congestion on highways and expressways. To describe this erratic behaviour, the optimal speed model is proposed (Bando et al., 1995). The OVM uses the equation of motion of each vehicle to describe the dynamic process of traffic congestion. Bando makes three assumptions about this model, including: (a) each driver responds to the stimulus of the vehicle in front of him with a legal speed; (b) each driver observes the motion of the vehicle in front of him and controls his acceleration/deceleration accordingly to maintain safe speed; (c) no response time delay is considered. In the OVM model, the driver does not overtake the vehicle in front of him under any circumstances. This model ensures that erratic traffic flows generate congestion rather than traffic accidents.

Like the GHR model, OVM has the advantage that the model structure is simple and easy to calibrate using numerical methods. Another advantage of the OVM is that it is

better able to simulate following traffic in congested conditions on expressways and highways. However, as noted in some studies (Lazar et al., 2016), in some cases OVM can produce unrealistically large accelerations.

2.2.5. Wiedemann model

The Wiedemann model is one of the most studied psycho-physical models. This type of model considers the driver's behaviour as varying according to the traffic state he/she is in. Psycho-physical models use a “perceptual threshold” to define the boundary conditions for different states, which is a function of relative speed and relative distance to the leading vehicle (Wiedemann, 1974). The driver only responds to changes in spacing or relative speed when the threshold is reached. In this way, it also records the driver's attention to small spacing and lack of following behaviour for large spacing. In 1992, Wiedemann and Reiter applied the psycho-physical method to a simulation system named MISSION (Fellendorf & Vortisch, 2010). The Wiedemann model describes the interaction of follower and leader vehicle pairs using varying thresholds in four different regimes (Wiedemann & Reiter, 1992), including (1) free-flowing (the subject vehicle is not influenced by the leading vehicle), (2) approaching (the subject vehicle is influenced consciously by the slower leading vehicle), (3) following (The subject vehicle unconsciously follows the preceding vehicle and the following behaviour approaches a steady state equilibrium), and (4) emergency (critical situations requiring braking action).

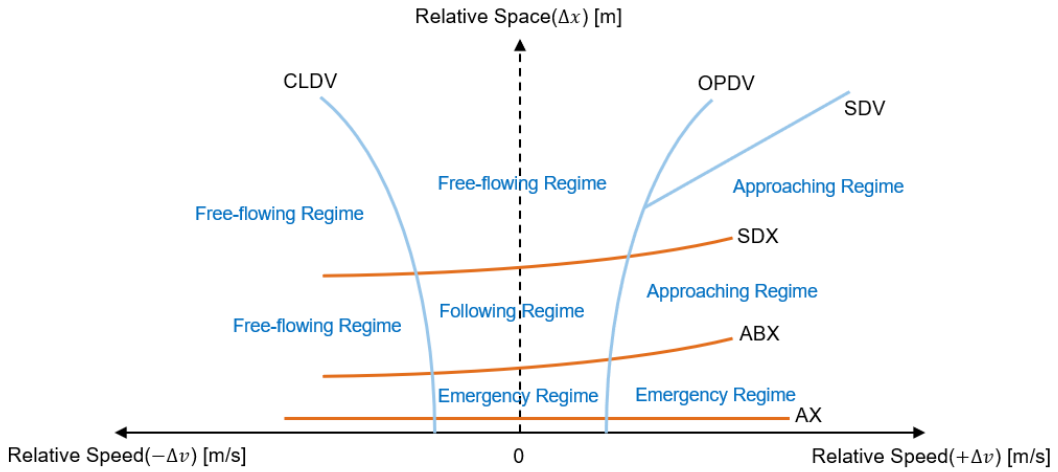


Figure 2.2 Schematic representation of thresholds and regimes in Wiedemann model

Six thresholds are defined in the relative speed and space diagram for the following pair, including:

- AX: the desired distance,
- ABX: the desired minimum following distance (the lower limit of the following regime),
- SDX: the maximum following distance (the upper limit of the following regime),

- SDV: the approaching point where the following driver perceives that he or she is approaching a slower leading vehicle,
- CLDV: the decreasing speed difference, and
- OPDV: the increasing speed difference.

Figure 2.2 depicts the boundary limits of the regime for different thresholds.

The concepts involved in the Wiedemann model have rarely been studied. While the whole system is able to simulate driving behaviour very well, the calibration for individual thresholds does not always seem to be accurate. However, many simulations have shown that the basis on which it is built is undoubtedly the most consistent and best describes most of the characteristics we see in daily driving behaviour (Brackstone & McDonald, 1999). Therefore, it is currently used by most micro-simulation systems, such as VISSIM and SUMO. Since its introduction, the Wiedemann model has been continuously enhanced. In 1999, Wiedemann updated his model (Wiedemann 99, W-99) from the original one (Wiedemann 74, W-74) and defined some of the thresholds in a different way to simulate freeway traffic better. For this reason, the driving behaviour in PTV VISSIM is set to W-99 for freeway driving and W-74 for urban driving.

After combing through the existing mainstream car-following models, it can be seen that the early models (such as the GHR and Gipps models) have a simple structure and fewer parameters to be calibrated, which makes them easy to apply. Desired measure models, such as the IDM model, are difficult to calibrate using real traffic data as their parameters are not directly observable. Optimal velocity models are good at simulating the dynamics of traffic flows, such as congestion on highways, but can still produce unrealistically large accelerations in specific cases. Psychophysiological models, such as the Wiedemann model, are the most complete car-following models available to describe the logic of driver behaviour and can describe most of the observed features of the daily driving behaviour well, and therefore it is the dominant car-following model used in most microsimulation systems today. However, the calibration of individual thresholds in it is not always accurate. For these reasons, we choose the Wiedemann model to calibrate and simulate the following behaviour.

2.3. Calibration of Car-Following Model

The car-following model describes the behaviour of the vehicle in the longitudinal direction, which includes not only the driving behaviour of the driver, but also the local traffic rules and possible restrictions. Drivers in different countries have different driving styles, drive different types of vehicles, and are subject to different traffic regulations. Drivers' driving behaviour and habits also respond differently on different road types. Therefore,

a car-following model could not simply use the default set of parameters directly to handle a particular task. Parameters must be modified and adapted to fit the problem under consideration. This process is known as calibration. Calibration are defined as (Treiber & Kesting, 2013): "*Calibration is the estimation of parameters to maximize the model's descriptive power to reproduce local driver behaviour and/or collective traffic-flow characteristics. The descriptive power is specified by an objective function to be applied to the test data.*" For linear car-following models, such as the GM model, the solution can be obtained directly by linear regression. However, for most non-linear car-following models, after constructing the objective function using mathematical methods, the approximate minimum value of the objective function needs to be obtained by iteration. The following discussion focuses on the mathematical principles of constructing the objective function and the commonly used non-linear optimisation methods.

2.3.1. Mathematical Principles

There are two main mathematical methods to formulate the calibration problem, the least squared errors, and the maximum likelihood.

1) The Ordinary Least Square (OLS) method

OLS is an intuitive method in regression analysis to approximate the solution of overdetermined systems. The objective function is defined directly in terms of a sum of squared errors (SSE), or the mean squared error (MSE) between the test data and the model prediction. The MSE is treated as a function of the parameters while the data are fixed.

For models which comprise a linear combination of the parameters, the regression model is linear, which is called linear least squares. For non-linear models, generally could not find a closed-form solution. In this case, numerical algorithms are used to find the value of the parameters that minimize the objective. By setting initial values and iteratively refining the parameters, an approximation to the minimum value can be obtained.

2) Maximum likelihood estimation (MLE)

The MLE is a method of estimating the parameters of an assumed probability distribution. As the MLE is explicitly based on probability, random elements with specific properties are necessary. If the model to be calibrated is deterministic, random noise needs to be added to it. The probability or probability density $p\left(\boldsymbol{y}_i^{sim}|\boldsymbol{\beta}\right)$ is defined that, at time t_i, the model makes the predictions $\boldsymbol{y}_i^{sim}$ to a given parameter vector $\boldsymbol{\beta}$ and suitable data driven conditions for the prior stem t_{i-1} or the initial state. The state to be predicted could be the speed and the gap between the following-pair, but also the detector counts, travel times, or propagation velocities of traffic waves. The likelihood function is defined as the joint probability that the model predicts all data points. The MLE of a parameter is

obtained by maximizing the likelihood function. The MLE reduces the calibration problem to a multi-variate non-linear optimization problem.

2.3.2. Non-linear Optimizations

For the problem that minimizing non-linear functions of several variables, is usually a complex task with no unique “one size fits all” solution. In terms of speed and robustness, the “best” solution for finding the global minimum depends on the complexity of the objective function.

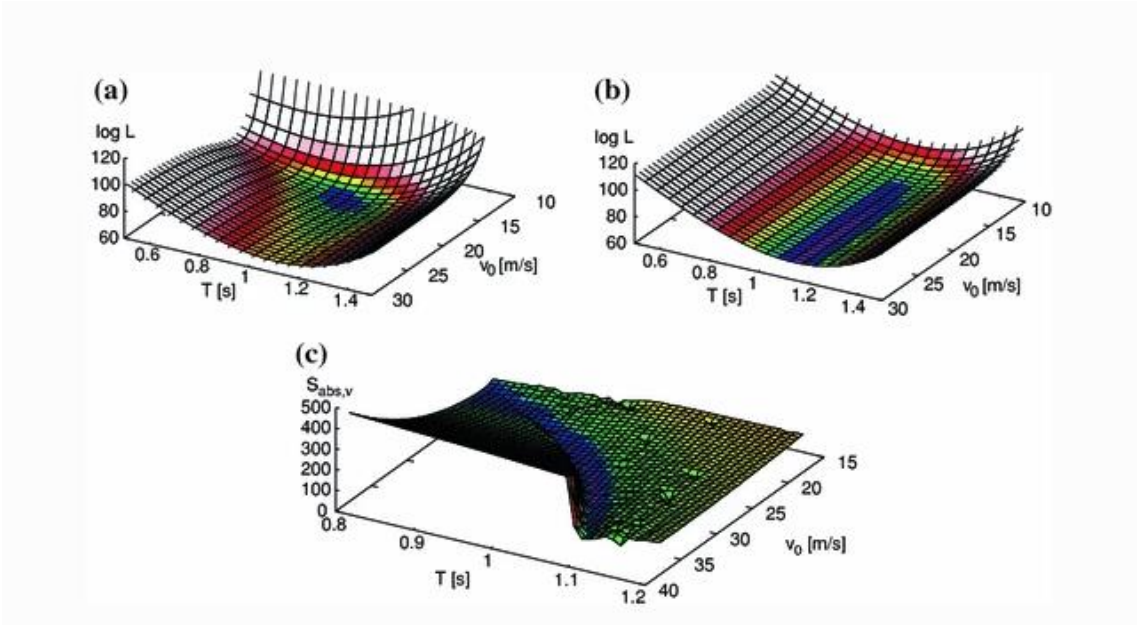


Figure 2.3 Examples of objectie function: (a) smooth function with a single global minimum, (b) smooth function with no definie minimum, and (c) objective function with a rugged fitting landscape (Treiber & Kesting, 2013).

Figure 2.3 shows three forms of objective functions. Figure 2.3 (a) and (b) show the local trajectory ML calibrations, and (c) shows a global OLS calibration to stationary detector data.

The calibration methods for car-following models can be grouped into two categories. The first category revolves around the concept of local fit. At each simulation step, the empirical vehicle state is fed into the formulated model to predict its acceleration. A cost function is then constructed to capture the error between the simulated acceleration trajectory and the corresponding experimental data, and the parameters are adjusted through iterations to minimize the error. This type of problem generally has a smooth objective function.

For smooth and unimodal objective functions (Figure 2.3 (a)), which are differentiable and have a global minimum, the global optimum solution can be found using simple deterministic methods. Commonly used methods are:

- Newton’s method: This method assumes that the objective function is quadratically differentiable and approximately quadratic. At each iteration, the objective function is approximated by a quadratic function and the extreme point of this function is the next iteration point. The advantage of Newton’s method is fast convergence.
- Gauss-Newton algorithm: This algorithm is a variation of Newton's method specifically for minimization of the sums of squares. The second derivatives are no longer required; however, this method is less robust compared to the Newton’s method.
- Gradient descent method: the search path proceeds always along the gradient at the last iteration point in this method. Along this direction we can easily determinate the minimum by a linear search, which becomes the new iteration point. This method is slower but more robust than the Newton and Gauss-Newton methods.
- Levenberg-Marquardt algorithm: This method makes a smooth transition between the gradient descent method and the Newton’s method and therefore combines the advantages of them. The transition is controlled by an adaptive “trust region”, which prevents premature convergence of the Gauss-Newton method. Levenberg-Marquardt algorithm is the most popular method for objective functions in form of a differentiable sum of squares.

The second category of calibration is global-fit. Compared with local-fit, the global-fit calibration results better in the minimization of fitting error than a local-fit calibration (Punzo & Simonelli, 2005). However, global fit often results in highly non-linear, non-smooth objective functions. In this situation, the four deterministic methods described above are no longer viable because of the difficulties of derivatives calculations (Li et al., 2016). Therefore, stochastic optimization methods need to be used to escape such minimum. Stochastic optimization methods are optimization methods that generate and use random variables. Introducing randomness into an optimization method, can, on the one hand speed up progress, and on the other hand, allow the method to escape from the local optimum and eventually approach the global (approximate) optimum. In addition, algorithms using stochastic optimization methods can often achieve almost uniformly good performance on many types of problems. In practice, a variety of algorithms based on stochastic optimization have been widely used in calibration problems for global fitting. Genetic algorithms (GA) and particle swarm algorithms (PSO) are popular metaheuristics that have been successfully applied in recent years to calibrate car-following models.

- Genetic algorithm (GA)

Genetic Algorithms (GA) is a metaheuristic inspired by Darwinian evolutionary theory, in which the survival of fitter creatures and their genes were simulated. In the 1960s and 1970s, John Holland and his students invented and extended genetic algorithms (Holland, 1992). Most of the models of GA that exist today are based on Holland's theory.

GA is an algorithm based on population. In GA, each possible solution corresponds to a chromosome and each parameter to be optimized represents a gene on the chromosome. A fitness function is used to assess the fitness of each individual in the population. GA simulates natural selection mechanisms, with a greater probability of selecting individuals with higher fitness values. After selection, chromosomes crossover to improve solutions. To prevent the solution from being trapped in a local optimum, in each iteration the operator randomly changes the genes of a portion of the individuals in the population. This part is called variation, which increases the exploration behaviour of the GA. Through selection, crossover, and mutation, the more suitable solution is maximally retained and perpetuated, while the diversity of the population is maintained.

GA is widely used in multi-objective optimization problems due to its global, parallel, and robust characteristics. Moreover, GA is simple in terms of search improvement and does not suffer from the restrictive assumption of search space. GA has been used to solve problems in a variety of fields, including engineering, medicine, and finance. In the transportation field, GA has now been successfully applied to many aspects, such as dynamic traffic signal control (Sun et al., 2006), air traffic control (Hu & Di Paolo, 2009), and traffic model calibration (Yu & Fan, 2017).

- Particle Swarm Optimization (PSO)

The Particle Swarm Optimization (PSO) algorithm of continuous nonlinear functions was proposed by R. Eberhart and J. Kennedy in 1995 (Eberhart & Kennedy, 1995). The concept was inspired on the one hand by the behaviour of social animals, especially the foraging behaviour of flocks of birds and fish (Marini & Walczak, 2015). On the other hand, the PSO algorithm is also related to genetic algorithms and evolutionary strategies.

PSO can be described imaginatively as a flock of birds searching for food in a region. The flock's task is to find the largest food source (the global optimal solution). Throughout the search process, the flock lets other birds know the location of the food source by passing information about their respective positions to each other. Eventually, the whole flock will gather around the food source, i.e., what we call the optimal solution is found and the problem converges.

An advantage of PSO is that it does not require the optimization function to be differentiable, divisible, and continuous. In addition, PSO has a fast convergence rate and a simple algorithm structure that is easy to program and implement. However, a disadvantage of the PSO algorithm is that it tends to cause premature convergence due to the rapid disappearance of particle diversity. For a search space with multiple local optima, the PSO algorithm can easily fall into a local optimum and fail to obtain a global optimum.

As a result, the PSO algorithm is generally suitable for optimization problems with a high-dimensional search space, where have multiple local optimum points, and for problem which is not necessary to obtain a very exact solution. Several researchers have used PSO to solve the problems in traffic field, such as calibrating the car-following model in congested traffic conditions (Aghabayk et al., 2016), the simulation of large-scale traffic flow (Liu & Yang, 2022), and developing a lane change model for autonomous based on PSO (Sheppard & Haberman, 2021).

2.3.3. The current state of calibration of Wiedemann models

In terms of car-following model calibration, in terms of three dimensions, including the type of car-following model, the calibration methods, and the road scenarios, numerous researchers have published a large number of studies in more than half a century since the introduction of car-following models.

On the calibration of the Wiedemann model, Chaudhari et al. calibrated the W-99 model by using trajectory data based on Root-mean-square deviation (Chaudhari et al., 2022). Motion cameras are used to calibrate driving behaviour parameters of different types of vehicle-following pairs using parameter estimation methods on a main section of a motorway in Los Angeles, California, U.S.A. (Durrani et al., 2016). The GA is also used in the calibration of the Wiedemann model, such as the calibration using real-world driving data in Sweden (Pourabdollah et al., 2017), the real-world data collected in the Shanghai Naturalistic Driving Study (SH-NDS) of urban expressway in China (Zhu et al., 2018), and the trajectory data at the High accident-prone roads in USA (Hamdar et al., 2015).

In 2015, PSO is used to calibration the parameters in VISSIM (Rrecaj & MBombol, 2015). Aghabayk et al. used PSO to calibrate the Wiedemann model for a mixed traffic flow including heavy vehicles (Aghabayk et al., 2016).

Focusing on different traffic scenarios, the GA is used to calibrate the Wiedemann model for an actuated signalized intersection (Park & Qi, 2005), in mixed traffic (Manjunatha et al., 2013), and the traffic environment with connected and autonomous vehicles (Liu & Fan, 2020). Leyn and Vortisch performed manual simulations of theGerman motorway following models using the Vissim tool (Leyn & Vortisch, 2015).

2.4. Literature Gap

As mentioned in the previous sections, a great deal of research has been published on the car-following model in the nearly 70 years since its introduction, but at the same time there are still a large number of gaps to be explored.

1) In the current mainstream research on the calibration of car-following models using real-world data, the data is mainly collected through sensors and cameras mounted on the vehicle. This approach has the following drawbacks: (a) only information on the trajectory of a particular vehicle can be collected, and the collection of information grows exponentially with the number of vehicles. As a result, traffic trajectory data collected based on sensor information is typically on the order of 10 squared; (b) individual samples are susceptible to traffic conditions (e.g., congestion, accidents, etc.) which can lead to biased estimates of the entirety, and the resulting outcome does not accurately represent the full picture of this traffic scenario in general.
2) Car-following model calibration for German highway ramp areas using real-world data has not yet been reported. German motorways are unique in that they (a) have many sections with no speed limits and (b) German drivers have driving habits that place a high premium on the experience of driving at high speeds. Therefore, car-following models calibrated by previous researchers using traffic data from other countries and other road types cannot be directly applied to the traffic scenario of German motorway ramps and need to be calibrated using data from the same traffic scenario.

Based on the two gaps mentioned above, the contributions of this thesis are: (a) the use of overall traffic trajectory data taken by drones for calibration, which has a large data sample size and can reduce the impact of bias caused by individual differences on the overall estimation; (b) the sampling location is the area near the entrance/exit of a German highway, which is calibrated separately by distinction the type of vehicle in following pairs (truck or car) to obtain specific car-following model parameters for the following behaviour of different vehicle types in this specific traffic scenario.

3. Methodology

In this chapter, the methodology used in this thesis will be introduced. The introduction of the methodology consists of four parts. Section 3.1 introduces briefly the software and tools used in this study. Section 3.3 defines the Wiedemann 99 model. In section 3.4 the processes of the GA and PSO methods are explained separately.

3.1. Software and Tools

The programming language used in this thesis is Python 3.8. Python is a widely used programming language in scientific research because of its large number of libraries on data processing, optimization algorithms, and visualization. The list of python libraries used in this thesis is as follows:

- Data processing and computation: Pandas, NumPy
- Visualization and plotting: Matplotlib
- GA and PSO: Scikit-opt
- Other operations: Black, Sys, Pathlib

The hardware used in this thesis is a HP laptop with i7-1065G7 CPU and 16 GB RAM.

3.2. Operating procedures

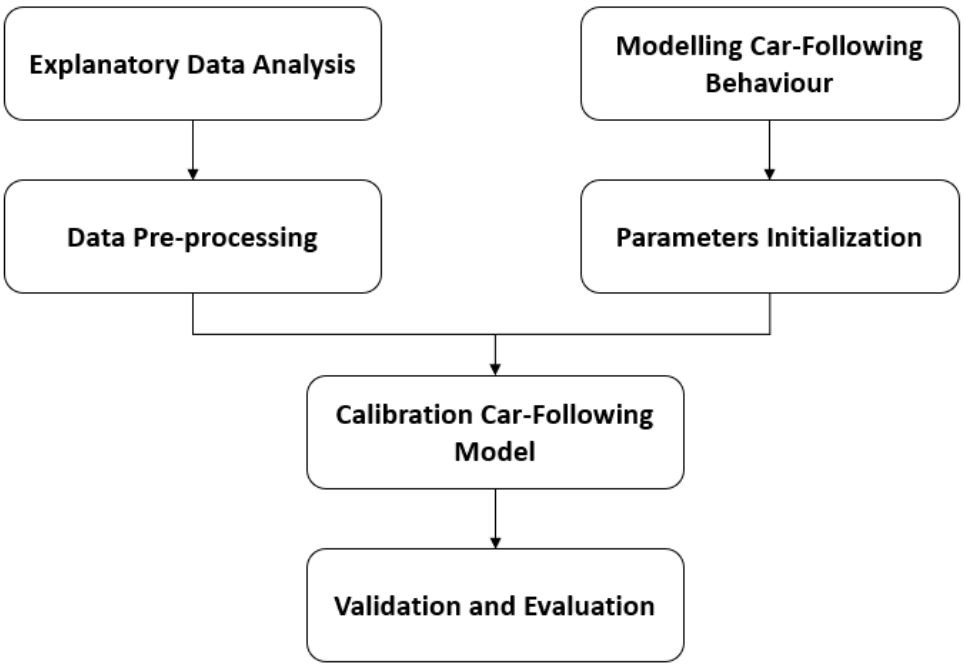


Figure 3.1 Thesis methodology

3.3. Car-following model

As we use data from the Highway, the Wiedemann 99 model (W-99) is better suited to this road scenario. Figure 3.2 describes the thresholds and the driving regimes. The complete flow chart for calculating the acceleration$a_n(t+1)$ by using W-99 model is shown in Figure 3.3 below (Zhu et al., 2018).

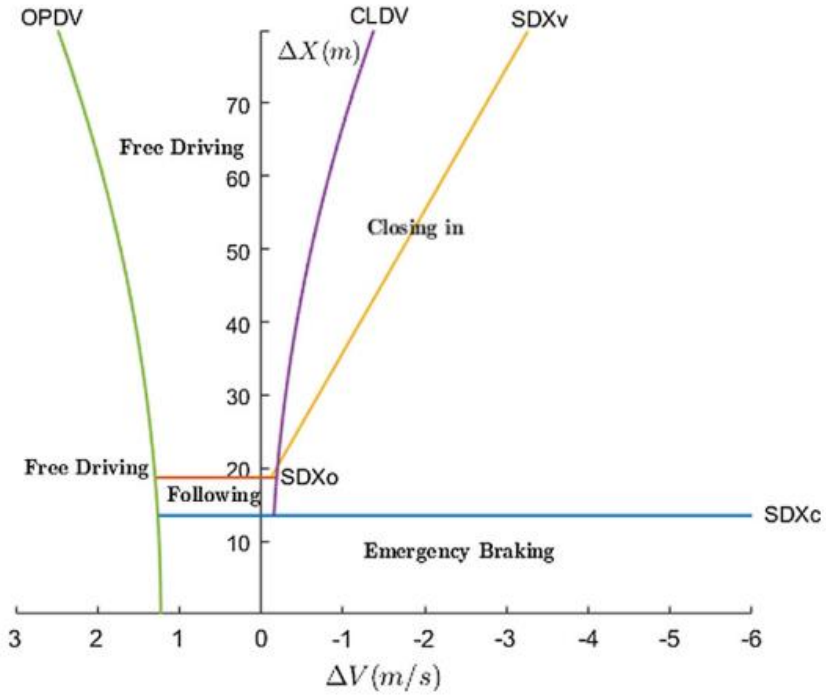


Figure 3.2 Thresholds and driving regimes in W-99 model (Zhu et al., 2018)

The corresponding thresholds in the flow chart are represented as:

$$\Delta X(t) = X_{n-1}(t) - X_n(t) - L_{n-1}$$

$$\Delta V(t) = V_{n-1}(t) - V_n(t)$$

$$SDX_c = CC_0 + CC_1 \cdot V_{slower}$$

$$V_{slower} = \begin{cases} V_n(t), & if\ \Delta V(t) > 0\ \ or\ \ LV_{acc}(t) < -1\ m/s^2 \\ V_n(t) - \Delta V(t) \cdot RND, & otherwise \end{cases}$$

$$RND = Uniform\ Random\ [-0.5, 0.5]$$

$$SDV = CC_6 \cdot (\Delta X(t) - L_{n-1})^2$$

$$SDX_0 = SDX_c + CC_2$$

$$SDX_v = SDX_0 + CC_3(\Delta V(t) - CC_4)$$

$$CLDV = \begin{cases} -SDV + CC_4, & if\ V_{n-1}(t) > 0 \\ 0, & otherwise \end{cases}$$

$$OPDV = \begin{cases} SDV + CC_5, & if\ V_n(t) > CC_5 \\ SDV, & otherwise \end{cases}$$

where:

X_n, X_{n-1} = position of the following and leading vehicle respectively,

ΔX = following gap (m),

V_n, V_{n-1} = velocity of the following and leading vehicle respectively (m/s),

L_{n-1} = length of the leading vehicle (m),

ΔV = speed difference between the following and leading vehicle (m/s),

SDX_c = minimum safe following distance (m),

LV_{acc} = acceleration of the lead vehicle (m/s^2),

SDV = perception threshold of speed difference (m/s),

SDX_0 = maximum following distance (m),

SDX_v = distance threshold of following vehicle perceiving its approach to a slower leader (m),

$CLDV$ = perception threshold of speed difference at short decreasing distances (m/s),

$OPDV$ = perception threshold of speed difference at short but increasing distances (m/s),

$VDES$ = desired speed of following vehicle (km/h),

a_{max} = maximum acceleration of vehicle (m/s^2), and

CC_0 to CC_9 = model parameters.

Table 3.3 shows the constraints applied to the Wiedemann 99 parameters according to Zhu et al. in 2018. The following data and boundaries are used as a reference for the initial value setting when initialising the following model.

Table 3.3 Wiedemann parameters value (Zhu et al., 2018)

Parameter	description	Bounds	Mean	Std. dev.	5%	95%
CC_0 [m]	Standstill gap	[0, 20]	1.0192	0.9696	0.1056	2.5673
CC_1 [m]	Headway time	[0, 5]	1.4242	0.4326	0.8200	2.1980
CC_2 [m]	“Following” variation	[0, 10]	5.8715	3.9500	0.5208	9.9986
CC_3 [s]	Threshold for entering “following”	[-20, 0]	-17.1579	4.7313	-19.9700	-4.5030
CC_4 [m/s]	Negative “following” threshold	[-5, 0]	-0.2312	0.2604	-0.9616	-0.1006
CC_5 [m/s]	Positive “following” threshold	[0.1, 5]	1.7946	1.5706	0.1391	4.7846
CC_6 [10^{-4} rad/s]	Speed dependency of oscillation	[0.1, 20]	3.5519	3.8974	0.2664	11.0473
CC_7 [m/s^2]	Oscillation acceleration	[−1, 1]	0.5350	0.3343	0.0974	0.9729
CC_8 [m/s^2]	Standstill acceleration	[0, 8]	4.0101	3.1826	0.1087	7.9984
CC_9 [m/s^2]	Acceleration at 80 km/h	[0, 8]	2.6549	3.0612	0.1095	7.8852
$VDES$ [km/h]	Desired speed of following vehicle	[1, 150]	86.0492	17.8998	67.2759	124.1050

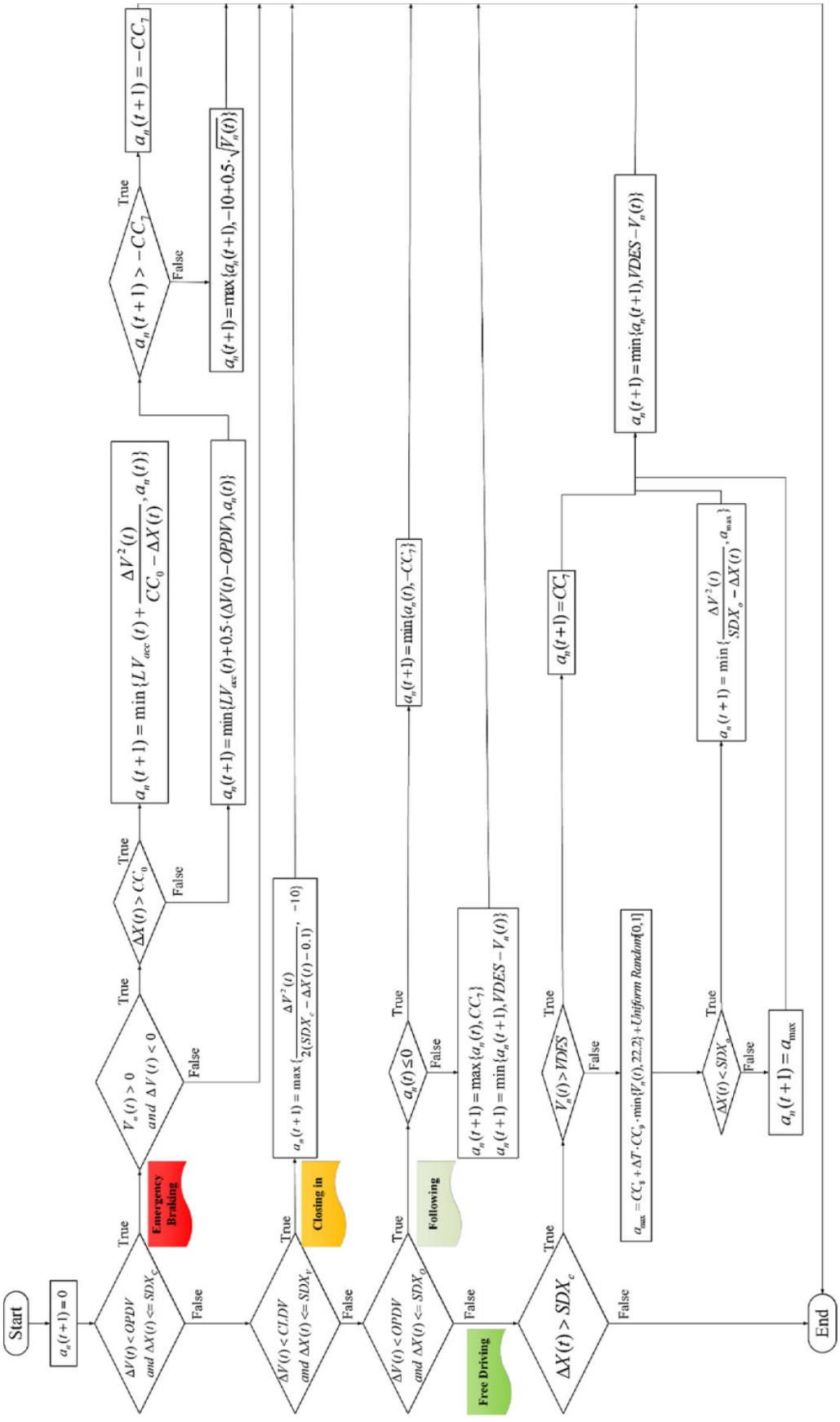


Figure 3.2 Calculation process in Wiedemann 99 model for acceleraton of following vehicle (Zhu et al., 2018)

3.4. Calibration Methods

Calibration refers to the adjustment of model parameters to improve the ability of the model to replicate driving behaviour and traffic characteristics. Figure 3.4 shows a general framework of calibration. According to this framework, calibration is an iterative process by which the error between the simulated and real values is reduced until a stopping criterion is achieved.

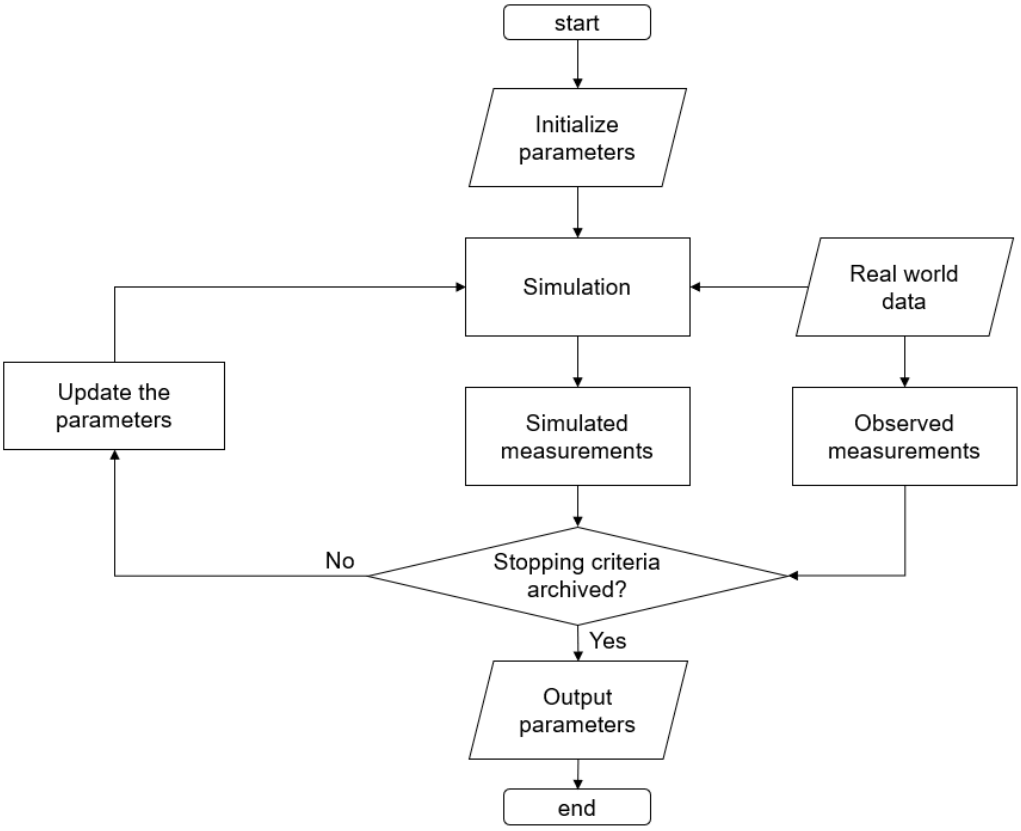


Figure 3.3 General framework of calibration

As mentioned before, in this study, the calibration methods are GA and PSO. There are several reasons for choosing these two optimisation methods. Firstly, as the Wiedemann model has multiple parameters and thresholds and there is no direct correlation between the thresholds, it is difficult to calibrate it directly using numerical methods. Using metaheuristic algorithms such as GA and PSO, by introducing randomness, largely reduces the amount of computation involved in the search process and can avoid the problem of difficult solutions using numerical methods. Secondly, the GA and PSO process is simple, the results are intuitive and the meaning of the parameters in the algorithm is easy to understand and adjust. The algorithms make few or no assumptions about the problem being optimised and, because of the stochastic nature, can search a very large space of candidate solutions. And by adjusting the parameters and repeating the experiments, convergence to the optimal solution can be achieved.

3.4.1. GA

The genetic algorithm consists of three main steps in each iteration: selection, crossover, and mutation. Figure 3.4 illustrates the basic processes of the GA.

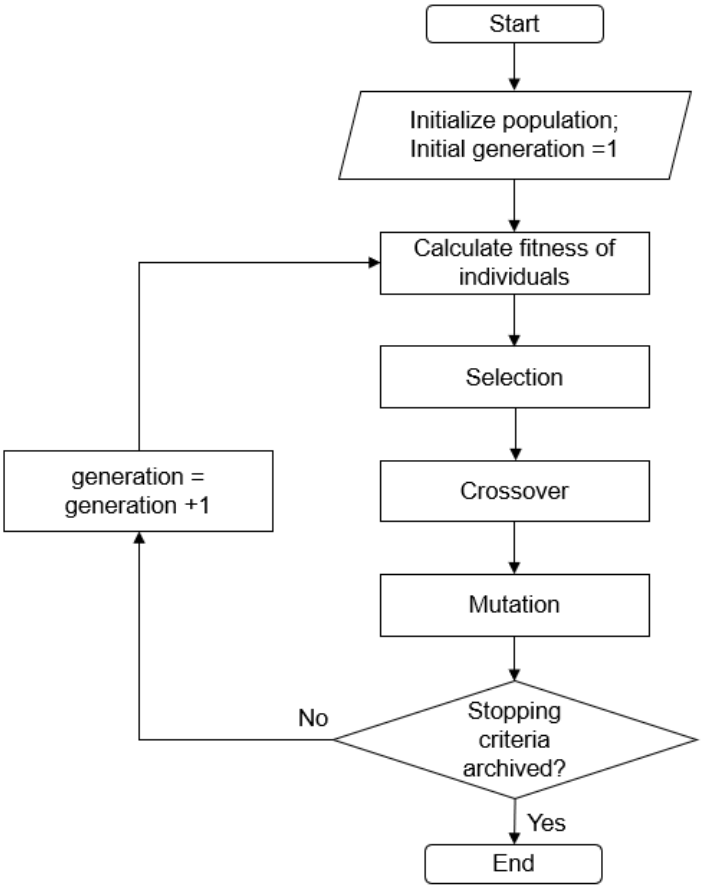


Figure 3.4 Flow chart of GA

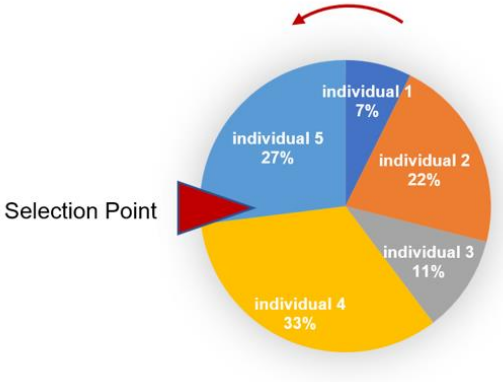
1) Selection

According to Darwin's theory of evolution, in the natural world, the most adapted individuals in nature have a greater chance of surviving and carrying on their genes to the next generation through mating. This causes their gene frequencies to increase in the next generation. To simulate natural selection, the GA used in this thesis selects individuals within a population by fitness proportionate selection. This selection method uses the fitness level to associate a probability of selection. For individual i in the population with N individuals, f_i is its fitness, and its probability of being selected is expressed as:

$$p_i = \frac{f_i}{\sum_{j=1}^{N} f_j}$$

This can be imagined as similar to a roulette wheel in a casino. Figure 3.4 shows an example of the roulette wheel. A proportion of the wheel is allocated to each of the possible selections based on their fitness value. It can be observed that the best individual

(with the highest Fitness value), i.e., individual 4, had the largest proportion of the roulette wheel, while the worst individual (individual 1) had the lowest proportion. In each selection phase of GA, two individual is selected by the roulette wheel.



Individual	Fitness Value	Percentage of Fitness [%]
Individual 1	15	7
Individual 2	44	22
Individual 3	22	11
Individual 4	68	33
Individual 5	55	27
Total	204	100

Figure 3.5 An example of the mechanism of fitness proportionate selection (roulette wheel selection)

2) Crossover

The two solutions selected by the selection process (the parent solutions) will be used to create the next generation. The two parent solutions will be crossed over to create two new child solutions. GA in this thesis uses the double-point crossover method. Figure 3.5 shows the concept of double-point crossover. Two crossover points are picked randomly from the parent chromosomes and the bits in between these two points are swapped between the parent chromosomes.

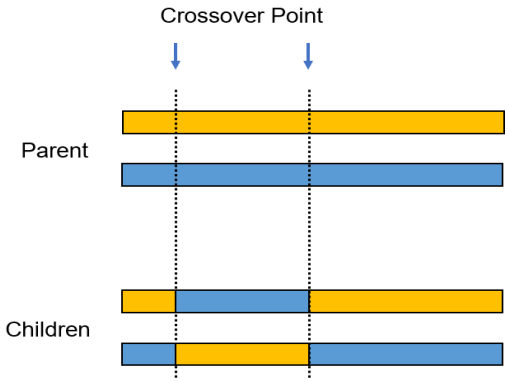


Figure 3.6 Concept of double-point crossover

3) Mutation

After the child solutions have been created, one or more of their genes will be randomly mutated. Mutations introduce another layer of randomness to maintain the diversity of the population. This operator reduces the probability that the GA will fall into a local optimum solution. In this thesis bit string mutation is the strategy of mutation, which gene at random locations was flipped.

3.4.2. PSO

The PSO algorithm for tracking the global optimum $GBEST$ published by R. Eberhart and J. Kennedy in 1995 has a simple form.

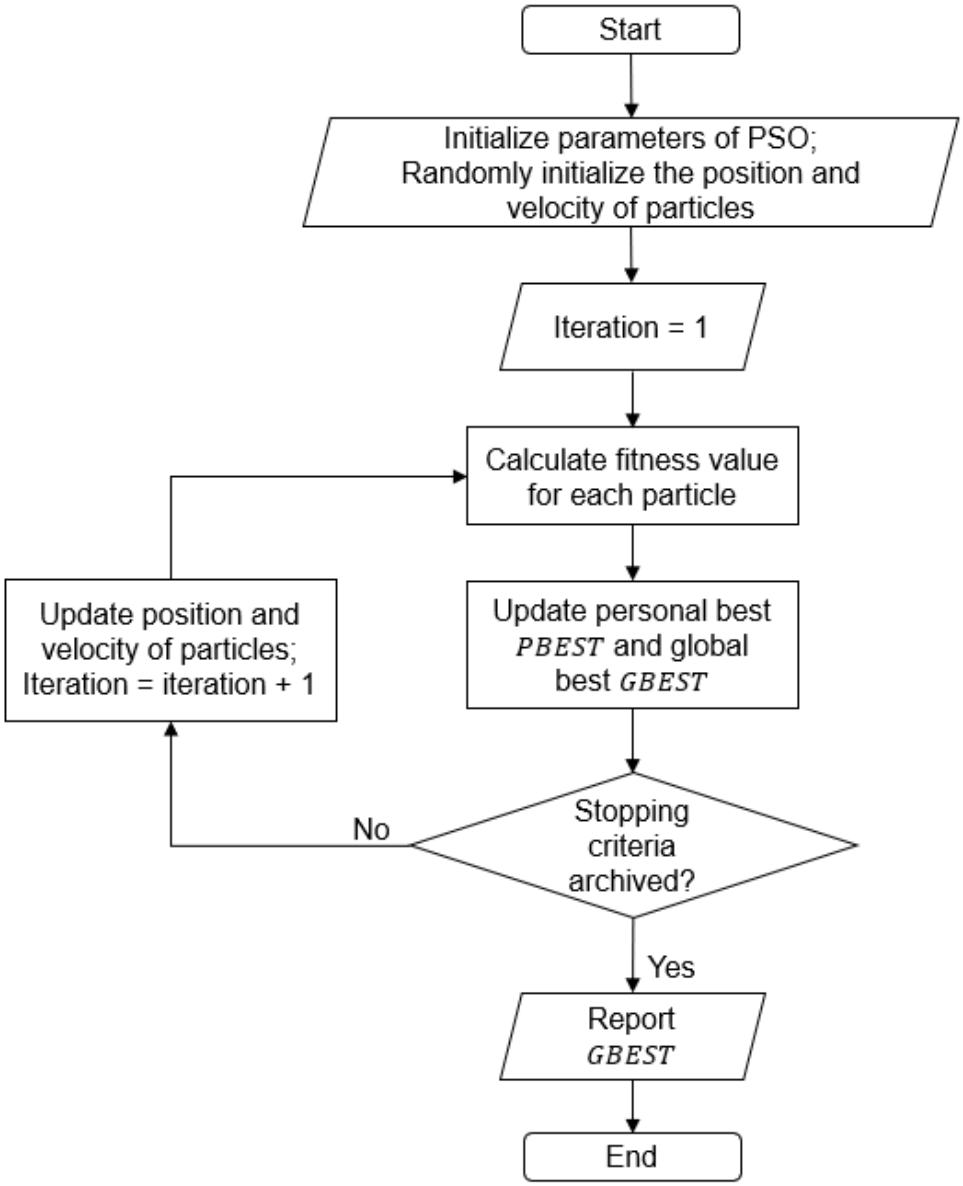


Figure 3.5 Flow chart: Steps of PSO algorithm.

The position of the i^{th} particle in D-dimensional hyperspace is described by a vector $\boldsymbol{x}_i$:

$$\boldsymbol{x}_i = [x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD}]$$

And the particle swarm is formed by the population of *N* candidate solutions:

$$\boldsymbol{X} = \{\boldsymbol{x_1}, \boldsymbol{x_2}, \boldsymbol{x_3}, \ldots, \boldsymbol{x_N}\}$$

In each time iteration, the position of particle changes by:

$$\boldsymbol{x_i}(t+1) = \boldsymbol{x_i}(t) + \boldsymbol{v_i}(t+1)$$

The mathematical formula of changing the velocity is:

$$\boldsymbol{v_i}(t+1) = w \cdot \boldsymbol{v_i}(t) + c_1 \cdot \boldsymbol{R}_1(t) \cdot \big(PBEST_i - \boldsymbol{x_i}(t)\big) + c_2 \cdot \boldsymbol{R}_2(t) \cdot \big(GBEST - \boldsymbol{x_i}(t)\big),$$

where w is inertia factor of particle swarm, c_1 and c_2 are two acceleration constants representing "cognitive coefficient" and "social coefficient" respectively. In general, c_1 and c_2 take values in the interval $[0, 4]$. They weight the two control the behaviour of particles. $\boldsymbol{R}_1$ and $\boldsymbol{R_2}$ are two matrices of random numbers generated from a uniform distribution in $[0, 1)$. For the tendency of the particles to move, they give a random effect. $PBEST_i$ is the personal optimum recorded of the i^{th} particle.

The stopping criterion can usually be one of the following three conditions:

- Achieve a specified number of iterations,
- $GBEST$ remains unchanged after a specified number of iterations since the last update, or
- attain the prespecified target of fitness value (Marini & Walczak, 2015).

4. Data Description

This chapter describes the processing and filtering of data. Section 4.1 describes how and where the data were collected and section 4.2 explains pre-processing operations on the data, including outlier removal and aggregation. Section 4.3 performs exploratory data analysis on the data. Section 4.4 describes the fundamental diagram of the road segment. In section 4.5, the following pairs are filtered out. The process is illustrated in Figure 4.1.

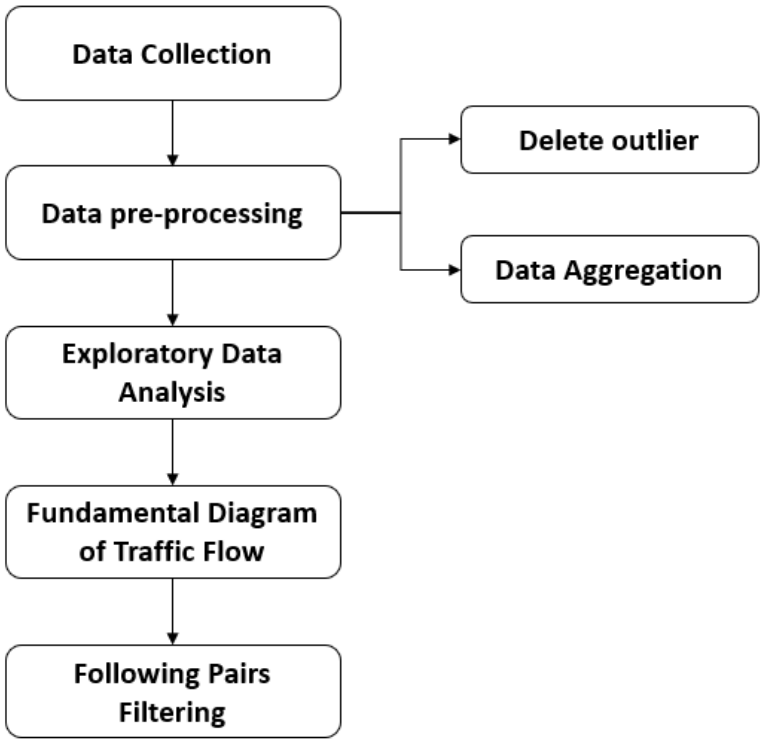


Figure 4.1 Flow chart of data processing and analysis

4.1. Data Collection

The following dataset used in this thesis is The Exits and Entries Drone Dataset (exiD) from fka GmbH (Moers et al., 2022). The exiD dataset is a new dataset that records the trajectories of natural road users at the exits and entrances of German highways. Using drones, the dataset has 92 individual records at seven different measurement locations at highway entrances and exits between Aachen and Cologne, Germany, where the trajectories of each road user and their type are extracted. In total, the traffic trajectories of 69,172 vehicles were extracted. The dataset uses state-of-the-art computer vision algorithms, the positional error is typically less than 10 cm.

The data containing information on traffic trajectories are recorded by camera-equipped drones. All videos were recorded in fine weather conditions with high visibility and little wind. At the flight altitude of the drone, approximately 420 meters of the highway are captured, which can cover a large part of the entrance and exit ramps. The video resolution is 4K and 25 frames per second. A deep neural network is used to detect road users in records and the detections from each frame are assigned to each other using a tracking-by-detection approach based on location and size to generate trajectories. The dataset was smoothed for the position, direction, velocity, and acceleration by using a Rauch-Tung-Striebel (RTS) smoother, which is a linear-Gaussian smoothing algorithm.

The dataset provides the position, heading, lateral and longitudinal speed, and acceleration of the recorded vehicles at each time step. In addition, the type of vehicle and the ID of the surrounding vehicles are also provided.

An overview of one recording from the exiD dataset is shown in Figure 4.1. This image visualizes the traffic trajectories of an entrance and exit on a German highway.

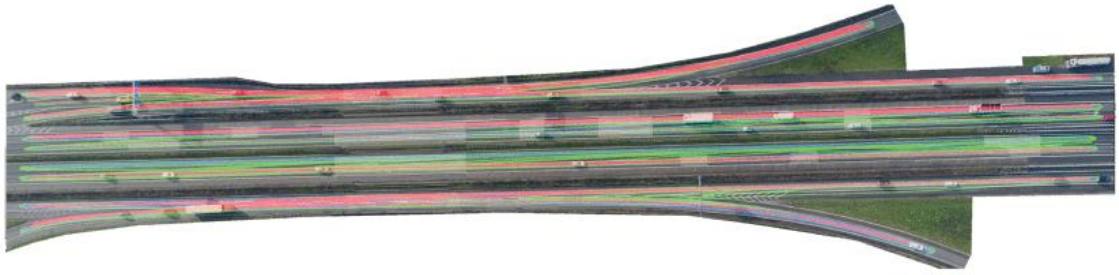


Figure 4.1 Overview image of one recording from the exiD dataset (Moers et al., 2022)

4.2. Data pre-processing

Data pre-processing refers to the manipulation or removal of data before it is used to ensure or improve the quality of the data. Data pre-processing is crucial before the start of an experiment as the quality of the data is directly related to the quality of the final results. Real data may inevitably contain a large number of missing values, and noise at the time of collection, or there may be outliers present as a result of manual entry errors. Therefore, pre-processing the data can reduce the impact of "dirty" data on the results.

The exiD dataset was post-processed after data collection. ExiD using an RTS smoother to obtain the smoothed position, direction, velocity, and acceleration of the traffic trajectory at each timestamp. Most missing values and data errors were automatically detected and resolved in this step. For errors that could not be resolved automatically, the developers of this dataset performed manual checks and processing. As a result, in the

datasets that we obtained, the missing values and most of the outliers during the data collection process have been processed and replaced. For the calibration of the car-following model, the data needs to be further filtered and aggregated based on conventional data processing.

4.2.1. Detecting and treating outlier

The main basis for further filtering is the reasonableness of the range of values of the parameters. Considering the reality of German highway traffic, in this study, the longitudinal velocity range of vehicles was set to be greater than 0 and less than 150 km/h. Observations outside this range are considered outliers. Manual inspection of the outliers revealed that the outliers did not appear in discrete timeframes, but always in the form of a continuous sequence in the data of a certain vehicle. One explanation for this phenomenon might be that the driving style of some vehicles differs too much from the totality, which could be caused by special circumstances or the personal style of the driver. As individual data deviate from the overall distribution, calibration using a dataset containing these data can also lead to a decrease in the accuracy of the results. Therefore, vehicle information containing outliers was removed from the training dataset.

4.2.2. Data aggregation

In the exiD dataset, the frame rate of the recordings is 25 Hz, which means that the time interval between two adjacent frames is 0.04 seconds. The calibrated car-following model aims to simulate as accurately as possible the behaviour of each vehicle in a realistic traffic scenario. The average human driver's reaction time is around one second, so a data accuracy of 0.04 seconds is clearly too high for this study. Aggregation of the time series data is required to match the realistic situation. In this study, data points within one second are aggregated. This means that every twenty-five consecutive data points in the time series will be aggregated into one data point. After aggregation, we obtain time series data with a time interval of one second.

4.3. Exploratory data analysis

Exploratory data analysis of the acquired data helps to gain the understanding of the data and discovering the patterns.

In this thesis, we conducted an exploratory analysis of the car and truck data separately to observe the distribution of the variables to be used in the calibration part of the experiment. The four variables observed were longitudinal velocity, longitudinal acceleration, time headway and distance headway. In discussing time headway and distance headway, we focused on vehicles with possible following behaviour by limiting the range of

distance headway to those with a distance headway of less than 120 meters from the vehicle in front.

As the lateral movement of the vehicle is not considered in the car-following behaviour, all references to velocity and acceleration in this thesis are in the longitudinal direction unless otherwise stated.

4.3.1. Variables distributions of car

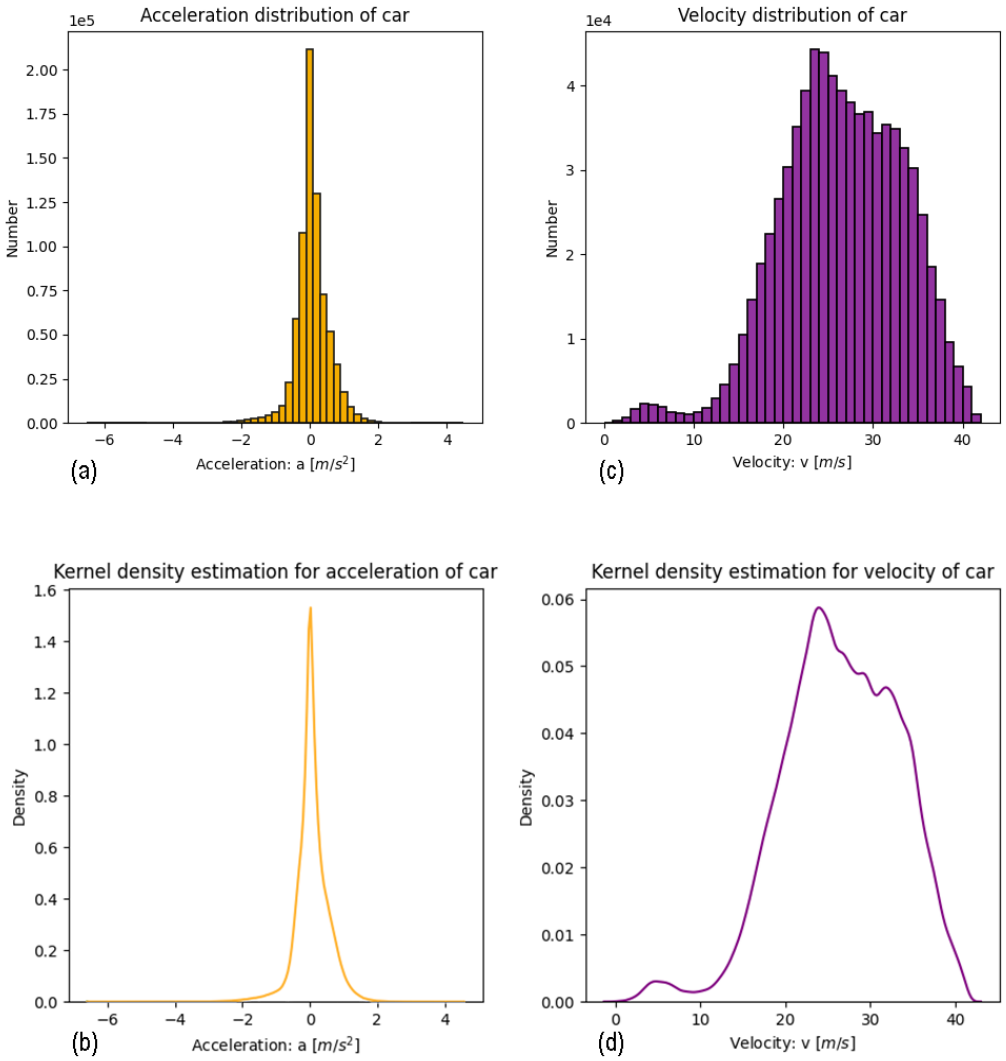


Figure 4.2 Distribution and kernel density estimation of velocity and acceleration of car: (a) acceleration distribution, (b) kernel density estimation for acceleration, (c) velocity distribution, and (d) kernel density estimation for velocity.

Figure 4.2 shows the velocity distribution and the acceleration distribution of car. Kernel density estimation helps us to observe the probability density distribution of the data in a more intuitive way. As illustrated, the acceleration follows an almost symmetrical uni-modal distribution with a very sharp peak. The peak value is approximately 0 m/s^2. Most acceleration values are in the range of -2 m/s^2 and 2 m/s^2.

The situation differs in terms of the distribution of velocity. There are obviously bimodal in the probability density curve about the velocity, with a first small peak occurring at about 4 m/s and a second large peak at about 24 m/s. A possible explanation for this is that there was unusual congestion on certain sections of the road during certain sampling periods, resulting in a lower speed distribution for this part of vehicles. This type of congestion is not common and therefore the volume of data is small in the overall picture. The majority of vehicles had speeds in the range of 10 to 40 m/s.

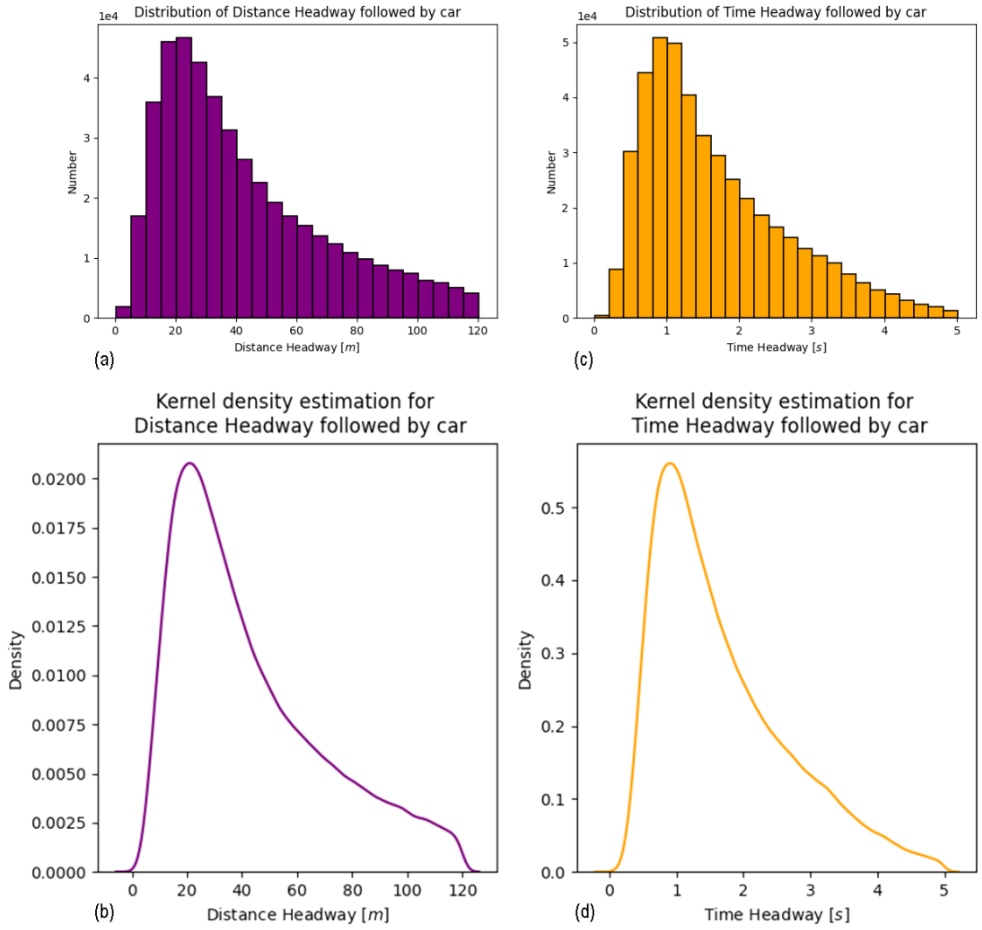


Figure 4.3 Distribution and kernel density estimation of distance headway and time headway of the following car: (a) distance headway distribution, (b) kernel density estimation for distance headway, (c) time headway distribution, and (d) kernel density estimation for time headway.

As mentioned above, in the analysis of the distance headway and time headway, the observations are restricted to vehicles with possible following behaviour, i.e., the distance headway falls within the range $(0, 120)$ meters. The distributions and the kernel density estimations are shown in Figure 4.3. It is obviously observed that both the time headway and distance headway show a unimodal and right-skewed distribution, which approximates the Rayleigh distribution in terms of shape. The kernel density estimation of time headway is relatively smooth. The distance headway peaks about 20 m and the time headway peaks at about 0.8 m/s.

The feature values for the probability density distributions are listed in Table 4.3 below. For the distribution of velocity and acceleration, a total of 755,793 observations were selected. For the time headway and distance headway, a total of 450,780 observations were selected. As the table shown, the mean value of car velocity is 26.5 m/s, the mean acceleration is 0.07 m/s^2, the mean time headway is 1.68 s, and the mean distance headway is 42.68 m.

Table 4.2 Description of the features of cars

	v [m/s]	a [m/s^2]	h [s]	s [m]
count	755793	755793	450780	450780
mean	26.5014	0.0699	1.6795	42.6810
std	6.7774	0.4912	0.9852	27.2075
min	0.0139	-6.5093	0.01	0.23
25%	22.0163	-0.1543	0.92	21.48
50%	26.5283	0.039	1.42	35.0049
75%	31.6847	0.3026	2.24	58.6125
max	41.6498	4.4765	4.99	119.99

Where $\boldsymbol{v}$ is the velocity, $\boldsymbol{a}$ is the acceleration, $\boldsymbol{h}$ is the time headway to lead vehicle, and $\boldsymbol{s}$ is the distance headway to lead vehicle. 25%, 50%, and 75% represents the first, seconds and the third quartiles respectively.

4.3.2. Variables distributions of truck

Figure 4.4 illustrates the velocity and acceleration distribution of the truck. It is noticeable that the velocity of the truck distributes more concentrated than the velocity of the car, which with a steep and slightly left-skewed unimodal distribution. The kernel density estimation curve of the truck's velocity is unsmooth. Most of the trucks have a concentrated distribution of velocities between 20 and 26 m/s.

The acceleration distribution pattern for trucks is similar to that of cars, also with a sharp, approximately symmetrical, and unimodal distribution, which the concentration of values around 0 m/s^2. The distribution of trucks is more concentrated than that of cars, most acceleration values of trucks are in the range of -1 m/s^2 and 1 m/s^2.

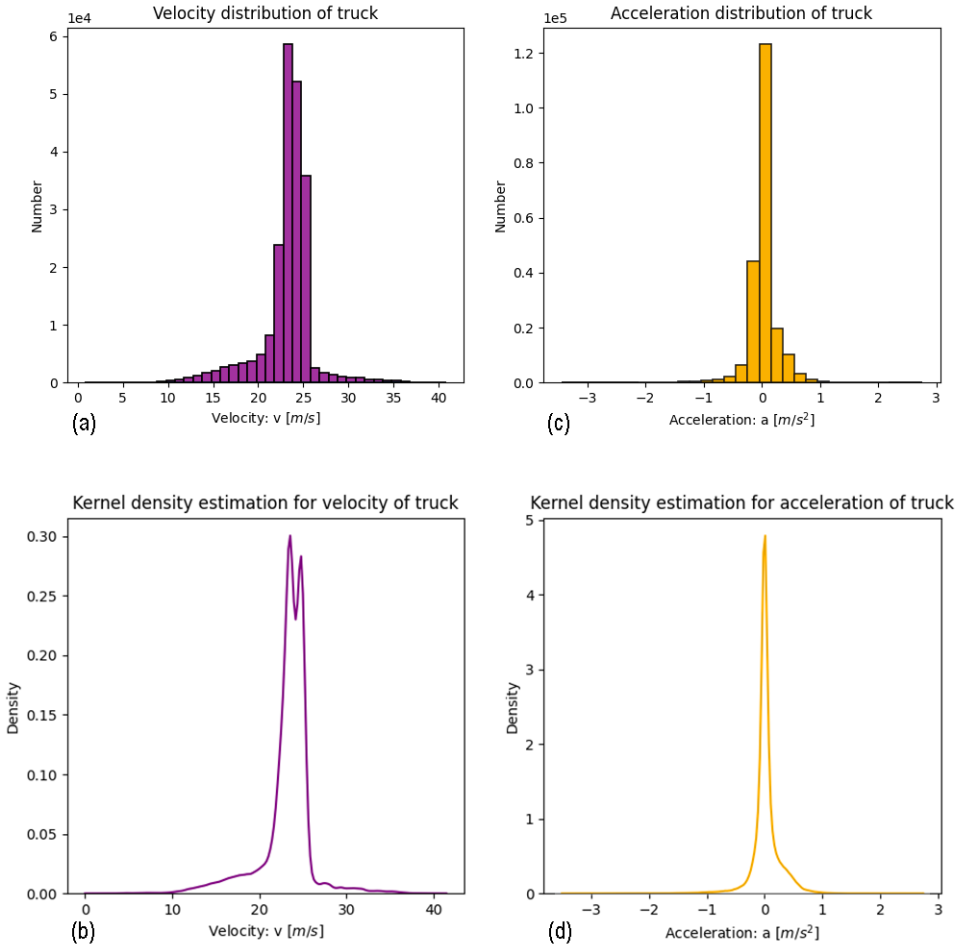


Figure 4.4 Distribution and kernel density estimation of velocity and acceleration of truck: (a) velocity distribution, (b) kernel density estimation for velocity, (c) acceleration distribution, and (d) kernel density estimation for accelration.

The distribution of time headway and distance headway between a truck, as a vehicle with following behaviour, and the vehicle in front of it is shown in Figure 4.5. The distribution of the distance headway of the truck in the interval $(0, 120)$ meter can be approximated as a right-skewed single-peaked distribution. The kernel density estimation plot presents a more intuitive visualisation of this. With the help of the quantile values listed in Table 4.3, it can be observed that most of the time headway observations are between 1.4 s and 3.15 s, and most of the distance headway values are between 32.08 m and 73.74 m.

In graphical terms, the distance headway and the time headway have coherence. This is obviously visible in the kernel density estimation graph. The absence of smooth edges on the right-hand side of the graphics indicates that the distance headway and time headway of some of the trucks between the front vehicle are too large and beyond the range of values taken in this thesis. This situation is also consistent with the observation in real world that trucks are usually further away from the vehicle in front than cars.

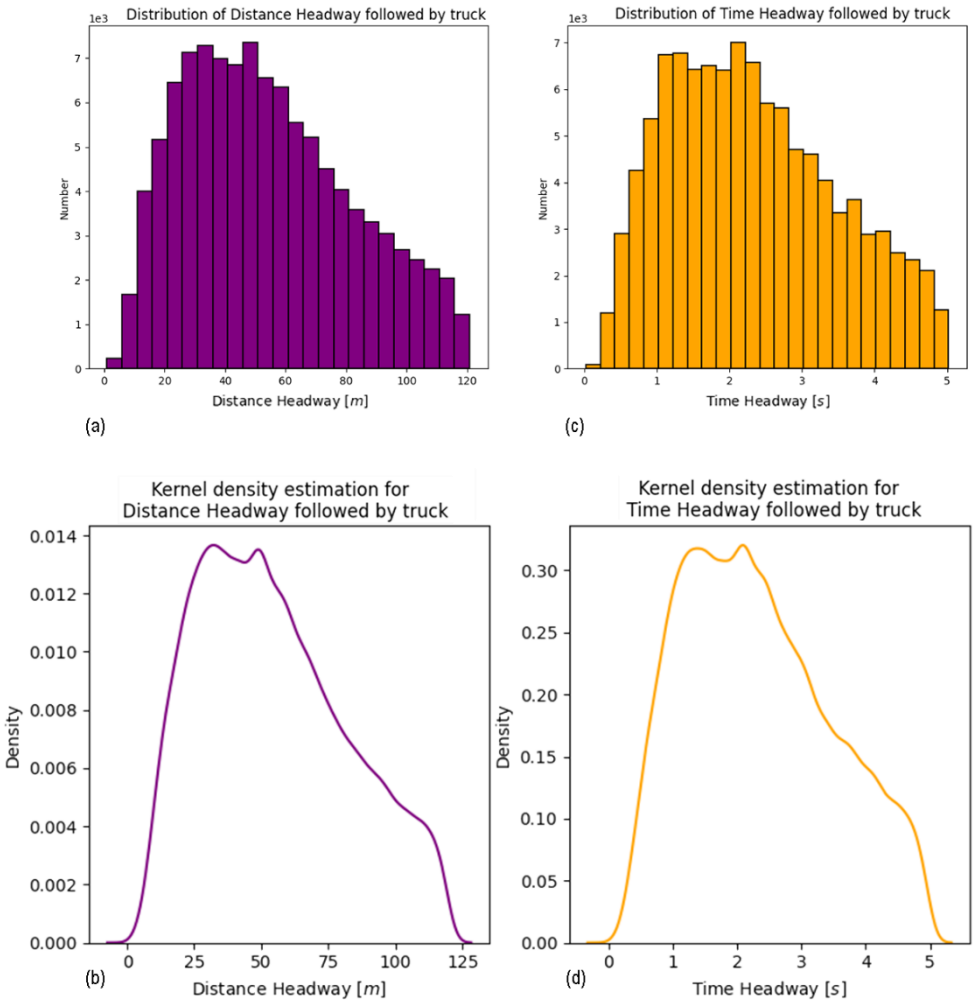


Figure 4.5 Distribution and kernel density estimation of distance headway and time headway of the following truck: (a) distance headway distribution, (b) kernel density estimation for distance headway, (c) time headway distribution, and (d) kernel density estimation for time headway.

Table 4.3 shows the distribution features of trucks. In total, there were 214,165 observations included in the velocity and acceleration distributions. For the time headway and distance headway, 105,940 observations were extracted. The mean value of truck velocity is 23.26 m/s, the mean acceleration is 0.02 m/s^2, the mean time headway is 2.33 s, and the mean distance headway is 54.43 m.

Table 4.3 Discription of the features of trucks

	v [m/s]	a [m/s^2]	h [s]	s [m]
count	214165	214165	105940	105940
mean	23.2561	0.0232	2.3319	54.4347
std	2.9994	0.2395	1.1507	27.6877
min	0.7971	-3.4437	0.02	0.86

	v [m/s]	a [m/s^2]	h [s]	s [m]
25%	22.6953	-0.0474	1.4	32.08
50%	23.6591	0.0066	2.19	50.73
75%	24.6941	0.0761	3.15	73.74
max	40.7015	2.6831	4.99	119.99

Where v is the velocity, a is the acceleration, h is the time headway to lead vehicle, and s is the distance headway to lead vehicle. 25%, 50%, and 75% represents the first, seconds and the third quartiles respectively.

A comparison of the distribution of the variables for trucks and cars follows. It can be noticed that cars have a wider distribution interval than trucks for both the velocity and acceleration variables. The distribution for trucks is more concentrated. This demonstrates that truck drivers are less likely to exhibit acceleration/deceleration behaviour of large magnitude compared to car drivers. In addition, driving styles and habits of individual truck drivers differ less from each other than from the group of car drivers.

The situation becomes different in the variables time headway and distance headway. Trucks show a wider distribution, while cars have a more concentrated peak. The explanation for this phenomenon is that car drivers tend to keep a smaller following distance from the vehicle in front, whereas truck drivers usually keep a larger following distance because of the longer braking distance trucks required compared to cars.

4.4. Fundamental Diagram of Traffic Flow

The fundamental diagram gives the relationship between traffic flow speed, density, and volume in a section of road. It is generally used to represent the traffic capability of the road segment and to simulate changes in traffic flow under regulatory measures such as speed limits.

Figure 4.6 shows the fundamental diagram for the data collection road section numbered 01 in exiD dataset. In this figure q repesents traffic volume, v is the speed of the traffic flow, and k is the traffic density in the road segment. As the seven road sections contained in this dataset have similar fundamental diagrams, this subsection uses the road segment number 01 as an example to analyze the traffic flow characteristics in this dataset.

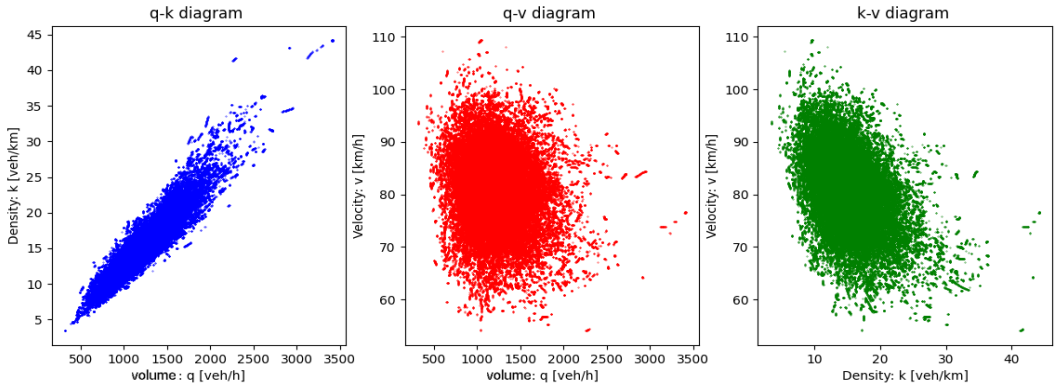


Figure 4.6 Fundamental diagram of Data collection site 01 in exiD dataset

It can be clearly observed from the $q-k$ diagram that traffic density and traffic flux show a significant positive correlation. In a more typical k-q diagram, the traffic density reaches a maximum when the traffic flux increases to a certain value and then gradually decreases as the traffic flux increases. However, in Figure 4.6, this change is not observed. A possible explanation for this situation is the low density of traffic flow on the observed sections of this dataset. As the traffic flux is below the capacity of this road segment, the maximum traffic ability of the road is not reached and therefore the traffic density does not reach the maximum value of the capacity of the road section.

Negative correlations between variables can be observed in both the $q-v$ and $k-v$ plots. Compared to the $q-k$ plots, the $q-v$ and $k-v$ plots present a less pronounced correlation. The explanation for this phenomenon is also because of the low density of traffic. In this case, the speed of the traffic flow depends mainly on the desired speed of each individual vehicle, which in turn is related to the driving habits of the driver and the characteristics of the vehicle type. Traffic flow and density have some influence on speed but are not the main factors.

4.5. Selection of Following Pairs

To begin with, we need to define the constraints of car-following behaviour. The constraints generally consist of three components, namely the lateral distance between the two vehicles (to determine that the lead vehicle is in the same lane as the following vehicle), the longitudinal distance (when the front vehicle is beyond a certain distance from the vehicle behind, the behind vehicle is in free driving and the behaviour of the front vehicle has no effect on the behind vehicle), and the length of time that the following state is maintained (to determine that the car-following behaviour between two vehicles does exist and do not briefly exist in a front-to-back relationship). Table 4.4 summarize the parameter ranges of the car-following periods in several studies and the final criteria for car-following extraction used in this thesis based on previous studies.

Table 4.4 Summary of extraction criteria for car-following behaviour

	Lateral distance [m]	Longitudinal distance [m]	Duration [s]
(LeBlanc et al., 2013)			> 15
(Chong et al., 2013)	< 1.9	< 120	> 30
(Fernandez, 2011)		< 100	
(Zhu et al., 2018)	< 2.5	< 120	> 15
This study	At the same lane (Lane ID known)	< 120	> 15

According to most of the literature listed, the longitudinal distance between the lead vehicle and the following vehicle was set to less than 120 meters and the following time was longer than 15 seconds in this study. By using these conditions, the following pairs between different types of vehicles in the dataset are filtered out. The flow chart of the following-pair selection, in which the following vehicle is a truck and the leading vehicle is a car, is illustrated in Figure 4.7.

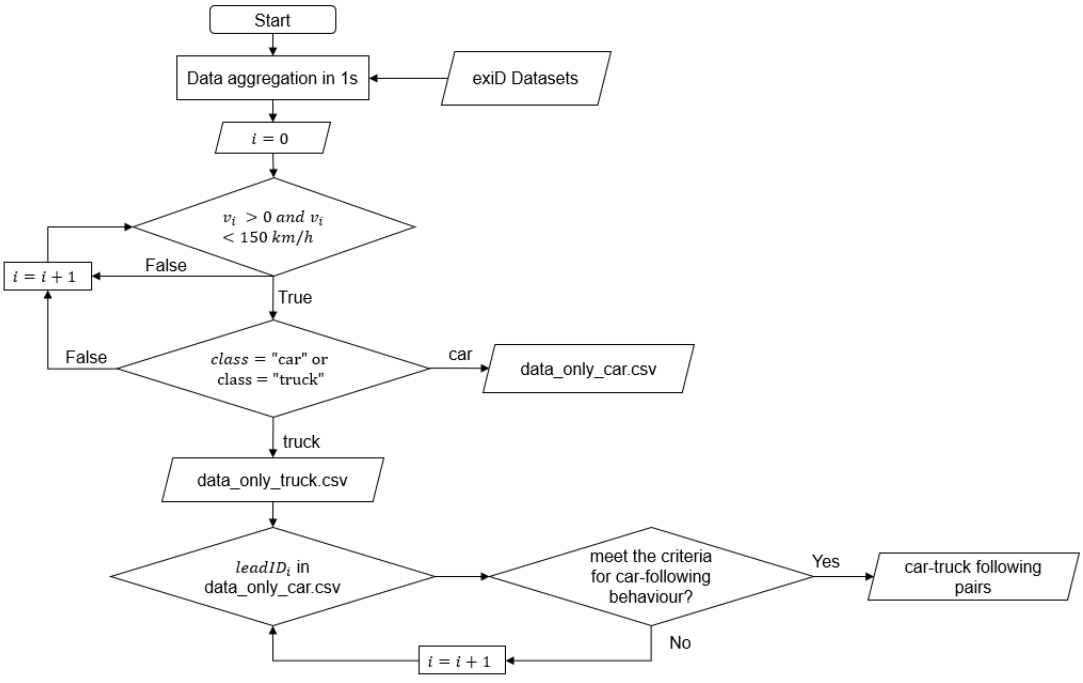


Figure 4.7 Flowchart of following-pair selection: by truck follows car

Table 4.5 shows the number of car-following pairs. As can be seen in Table 4.5, in the dataset used in this thesis, the largest quantity of following pairs was of type car-car, with a total of 4,466 pairs, and the smallest quantity of following pairs was of type car-truck,

with only 224 pairs in 92 datasets across seven observed locations. Trucks followed cars the least, which might be explained by the more frequent lane changing behaviour of cars or by the driving habits of truck drivers who prefer to follow trucks.

Table 4.5 Amount of different car-following type in dataset

		Type of following vehicle	
		Car	Truck
Type of leading vehicle	Car	4466	224
	Truck	715	625

5. Results and Discussion

This chapter introduces the experimental procedures and results. Section 5.1 explains the experimental design. In sections 5.2 and 5.3, the convergence and the performance of the optimization algorithms are evaluated. In section 5.4, the velocities estimated by the calibrated model and the observed values are compared.

5.1. Experimental Design

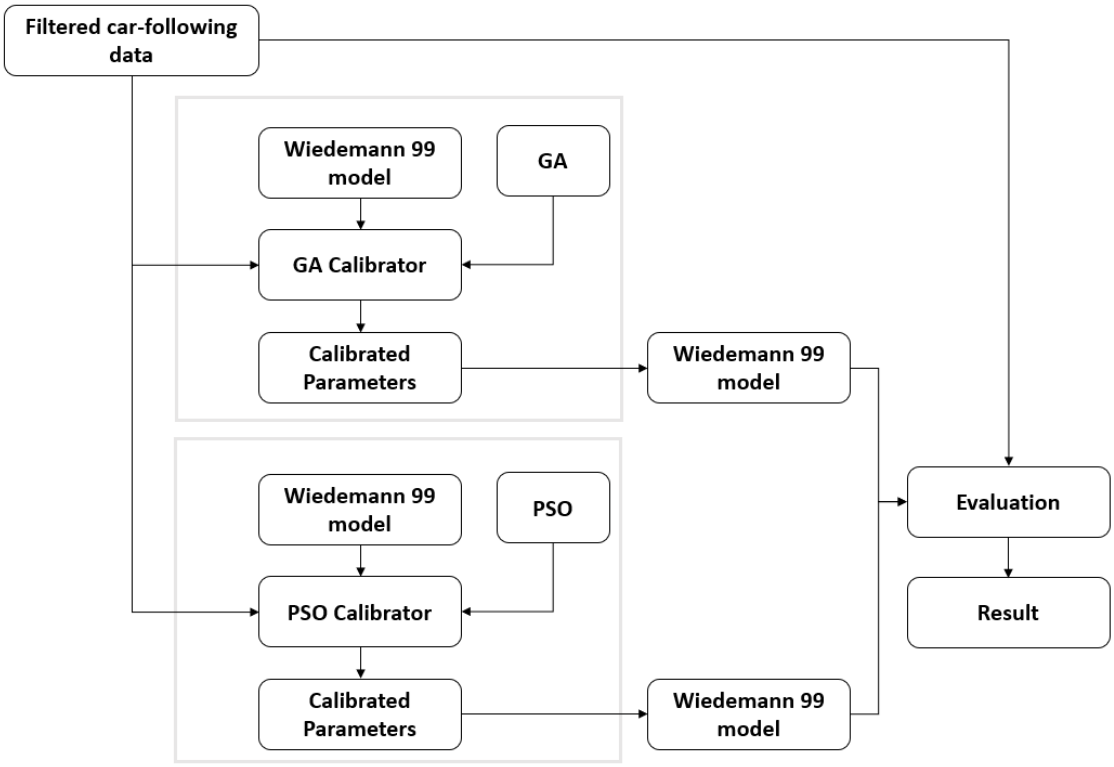


Figure 5.1 Flowchart of Wiedemann 99 model calibration by GA and PSO

As shown in Figure 5.1, the experimental procedure consists of two parts, which are the calibration of the W-99 model and the evaluation of the calibration results. In the calibration section, the filtered car-following data is fed into the GA and PSO calibrators respectively. The calibrator uses the chosen optimization algorithm (GA or PSO) to adjust the parameters of the model with the goal of making the difference between the model simulated output and the real observations as minimal as possible.

5.1.1. Initialization of models

Before calibration, the car-following model and calibrator need to be initialized. The parameters to be calibrated in the car-following model and the parameters in calibrators are initialized with the values recommended by the existing literature. The setting of the

calibrator parameters has a direct impact on the calibration results, so it is important to choose the appropriate parameters.

- Wiedemann 99 model

The initial values and bounds for the parameters to be calibrated in the W-99 model are given in Table 5.1. The bounds for the values are referenced from Vortisch and Zhu's study (Leyn & Vortisch, 2015; Zhu et al., 2018). Compared to the references, the desired speed boundaries $VDES$ are more narrowly defined at 72 to 108 km/h, which is set based on the high-speed characteristics of the traffic flow on the highway.

Table 5.1 The default value and bound of parameters in W-99 model

Parameter	**description**	**Bounds**	**Default**
CC_0 [m]	Standstill gap	[0, 20]	1.0192
CC_1 [m]	Headway time	[0, 5]	1.4242
CC_2 [m]	“Following” variation	[0, 10]	5.8715
CC_3 [s]	Threshold for entering “following”	[-20, 0]	-17.1579
CC_4 [m/s]	Negative “following” threshold	[-5, 0]	-0.2312
CC_5 [m/s]	Positive “following” threshold	[0.1, 5]	1.7946
CC_6 [10^{-4} rad/s]	Speed dependency of oscillation	[0.1, 20]	3.5519
CC_7 [m/s^2]	Oscillation acceleration	[−1, 1]	0.5350
CC_8 [m/s^2]	Standstill acceleration	[0, 8]	4.0101
CC_9 [m/s^2]	Acceleration at 80 km/h	[0, 8]	2.6549
$VDES$ [km/h]	Desired speed of following vehicle	[72, 108]	86.0492

- GA and PSO Calibrator

Table 5.2 and 5.3 shows the initial parameter values for GA and PSO calibrator respectively. In this thesis, GA and PSO algorithms used in the calibrator are called from an open-source Python library named scikit-opt, which is a module built specifically for heuristic algorithms (Guo, 2020).

The initial values are set with reference to published literature on the calibration of car-following models (Dadashzadeh et al., 2019). As PSO typically has a faster convergence speed than GA, a lesser maximum iteration is set for PSO in the experiments.

Table 5.2 Initial parameter values of PSO calibrator

Parameter	description	Default
$maxIter$	Maximum number of iterations	50
$nPop$	Population size	10
$nDim$	Dimension of objective function	11
$pMutation$	Mutation rate	0.001
$precision$	Precision: int/float or list	1e-4

Table 5.3 Initial parameter values of PSO calibrator

Parameter	description	Default
$maxIter$	Maximum number of iterations	40
$nPop$	Swarm size	10
$nDim$	Dimension of objective function	11
w	Inertia weight	0.5
c_1	Personal learning coefficient	1
c_2	Global learning coefficient	1

5.1.2. Goodness-of-Fit Function Definition

Goodness-of-Fit (GoF) function measures the accuracy of the calibrated model, which serves as the objective function of GA and PSO. The fitness function plays an important role in the selection process of an individual solution for the next generation in GA and determines the tendency of the particle swarm to move in PSO. In this thesis, the root-mean-square error (RMSE) is used to define the fitness function. The RMSE is defined as:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \hat{x}_i)^2}{N}}$$

where x_i is the actual observation from real-world data in frame i, $\hat{x}_i$ is the simulated value, and N is the total number of frames input.

Since the output of the Wiedemann model is the predicted acceleration of the vehicle at the next time stop, the fitness function is defined as the RMSE of acceleration:

$$Fitness = RMSE_{acc} = \sqrt{\frac{\sum_{i=1}^{N}(a_i - \hat{a}_i)^2}{N}}$$

where a_i is the observed acceleration in frame i, $\hat{x}_i$ is the simulated acceleration from W-99 model, and N is the total number of frames input.

5.2. Optimization Convergence Analysis

In an optimization algorithm, convergence represents a steady state at the end of the process, when more iterations will not lead to further changes or improvements in the results. During optimization, premature convergence might be observed, where the algorithm finds a stable point (usually referred to as a sub-optimal solution) too early and the evaluation results are worse than expected.

5.2.1. Convergence of GA

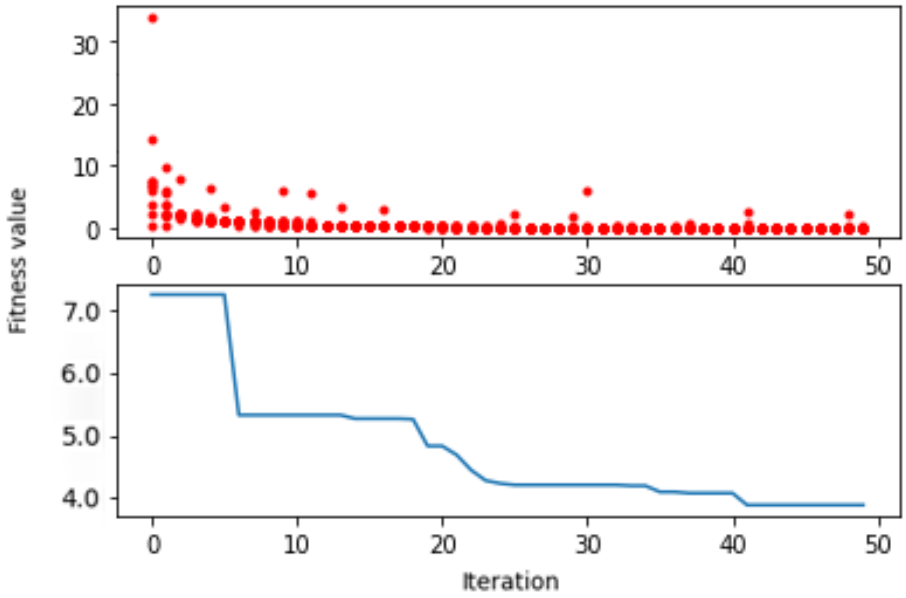


Figure 5.2 Convergence of GA calibrator: by following-pair type with car-truck, $nPop = 10$, $maxIter = 50$, and $pMutation = 0.004$

Figure 5.2 illustrates the convergence process for calibration of the W-99 model by using the GA calibrator. The input following-pairs data is in type car-truck, in which the lead vehicle is a car, and the following vehicle is a truck. The scatter plot above shows the fitness value of each individual in every generation, and the line plot below presents the lowest fitness value achieved in each iteration. In the scatter plot we can observe that, in general, the distribution of fitness values for the population tends to converge as the

iterations increase. During the iterations, some outliers appear occasionally. This is attributed to the randomness of the mutation in the algorithm. Since the direction of variation is uncertain, from the scatter plot we can observe the individuals with high fitness values after mutation, which are usually dropped out of the selection and do not survive into the next generation of inheritance. In contrast, when individuals with lower fitness values than their parents appear, there is a reduction in the minimum fitness value recorded by the line plot. This mutation is usually passed on to the next generation. As can be noticed from the line plot, there is a significant decrease in the lowest fitness value of the population as the number of iterations increases.

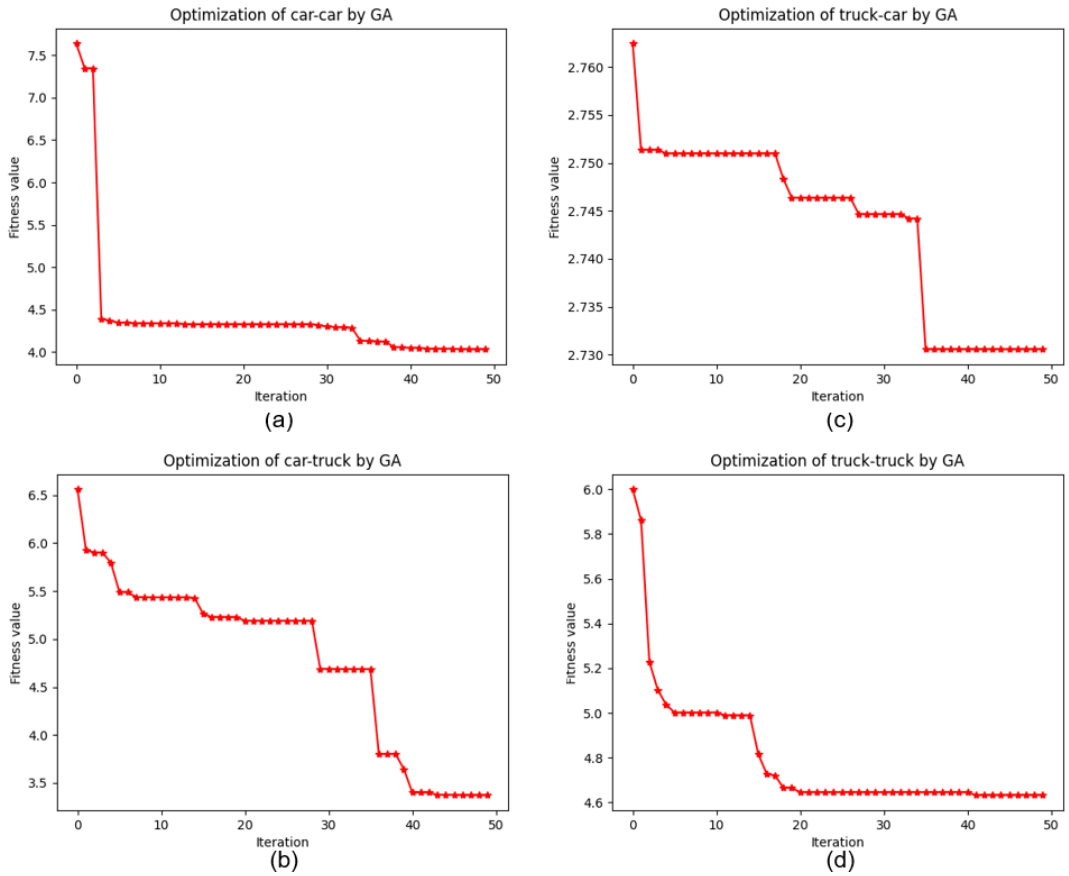


Figure 5.3 The optimization process by GA-calibrator in different following-pair type: (a) car-car, (b) car-truck, (c) truck-car, and (d) truck-truck.

As can be seen in Figure 5.3 that for the four different types of following pairs, the final fitness value (i.e., the RMSE between predicted acceleration and observed acceleration) that the GA calibrator converges to differs. In comparison, the GA calibrator has a better convergence for the calibration of the following pair of types of truck-cars. For data of type car-car and truck-truck, convergence is significantly worse. Possible reasons for this result are as follows:

- Algorithm features. Because of the random characteristics of both mutation and selection in GA, it is probable that during this optimization process, no better mutations

occurred or that genes with low fitness values were mutated but not selected by roulette wheel selection. Therefore, the good mutations were not passed on to the next generation.

- Dataset features. As the traffic flow conditions in the entrance/exit area of the highway are complex, vehicles would accelerate, decelerate, and change lanes more frequently in the entrance/exit area than on a normal highway. Consequently, car-following behaviour might also be impacted and exhibits instability.

As mentioned earlier, mutation has a significant impact on the convergence of the GA. Therefore, mutation rate is a key parameter in changing the GA convergence. The appropriate mutation rate varies for different optimisation problems. A mutation rate that is too low makes "profitable" mutations less likely to occur and will reduce the speed of convergence, while a mutation rate that is too high may increase the probability of a “good” gene being changed, which might also impact GA convergence. Therefore, it is important to find the appropriate mutation rate for a specific optimization problem. Figure 5.4 shows the effect of varying the mutation rate on the convergence of the GA calibrator, with all other parameters kept constant.

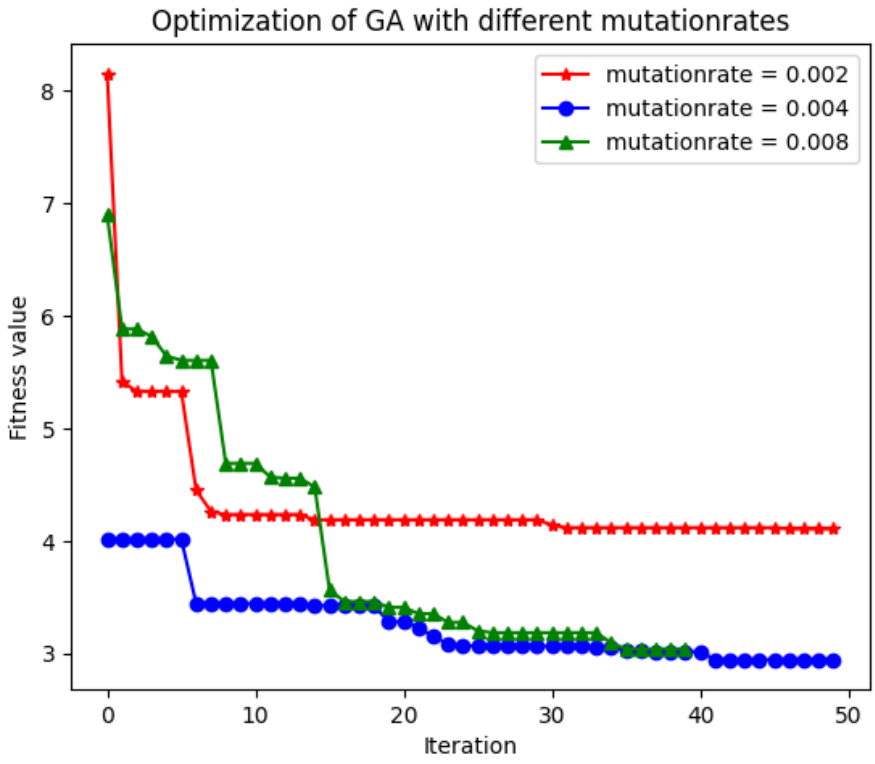


Figure 5.4 The influence of different mutation rates on the convergence of the GA calibrator under identical conditions: following-pair type car-truck, $nPop = 10$, $maxIter = 50$.

After comparison it is observable that, under the same conditions, the optimization process with a mutation rate of 0.002 converges significantly worse than the other two mutation rates. This is because the lower mutation rate in this scenario makes the probability

of obtaining a good gene lower. In the initial iterations, the optimization with a mutation rate of 0.008 has a higher adaptation value, but after about 30 iterations, it shows similar convergence to the optimization with a mutation rate of 0.004.

5.2.2. Convergence of PSO

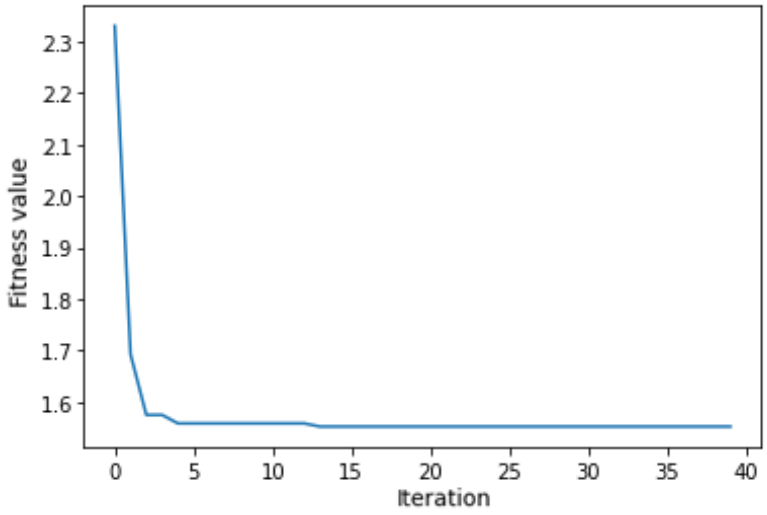


Figure 5.5 Convergence of PSO calibrator: by following-pair type with truck-car, $nPop = 10$, $maxIter = 40$, $w = 0.8$, $c_1 = 1$, and $c_2 = 1$

The calibration convergence process by the PSO calibrator using car-truck data is shown in Figure 5.5. The line graph records the change in the global optimum $GBEST$ as the number of iterations increases. From the plot, we can observe that the fitness value of $GBEST$ decreases rapidly and converges in the first few iterations. After about 15 iterations, the particles converge on a single point and do not move to other positions anymore. Fast convergence is a property of the PSO algorithm, which is consistent with our observations.

Figure 5.6 below illustrates the convergence of the optimisation using the PSO calibrator for different types of follow-up data. It can be seen that, in general, the PSO calibrator converges rapidly. The calibration of type car-car data converged the slowest of the four types, completing convergence at the 28th iteration. For the other three types, convergence was over before the 15th generation.

From the convergence results, despite the slowest convergence rate, the lowest fitness value was obtained for the calibration of the car-car data. For calibrations of type truck-truck, despite having the fastest convergence, the convergence is the worst in comparison in terms of convergence values. Possible explanations for this situation are as follows:

- Premature convergence of the PSO. This is the biggest drawback of the original PSO algorithm. Since the particles converge too quickly and always tend to move towards

the GBEST obtained in this iteration, PSO could easily fall into one of the local optima before it has even searched for a global optimum in space. Although PSO can reduce the probability of falling into a local optimum by assigning random terms to the velocity and direction of particle motion, the standard PSO is weakly able to escape from local optima compared to algorithms such as GA.

- Dataset characteristics. Since the GA calibrator also converged the worst for truck-truck data, it is also possible that there is relatively large variability in individual following behaviour in this combination of following-pairs.

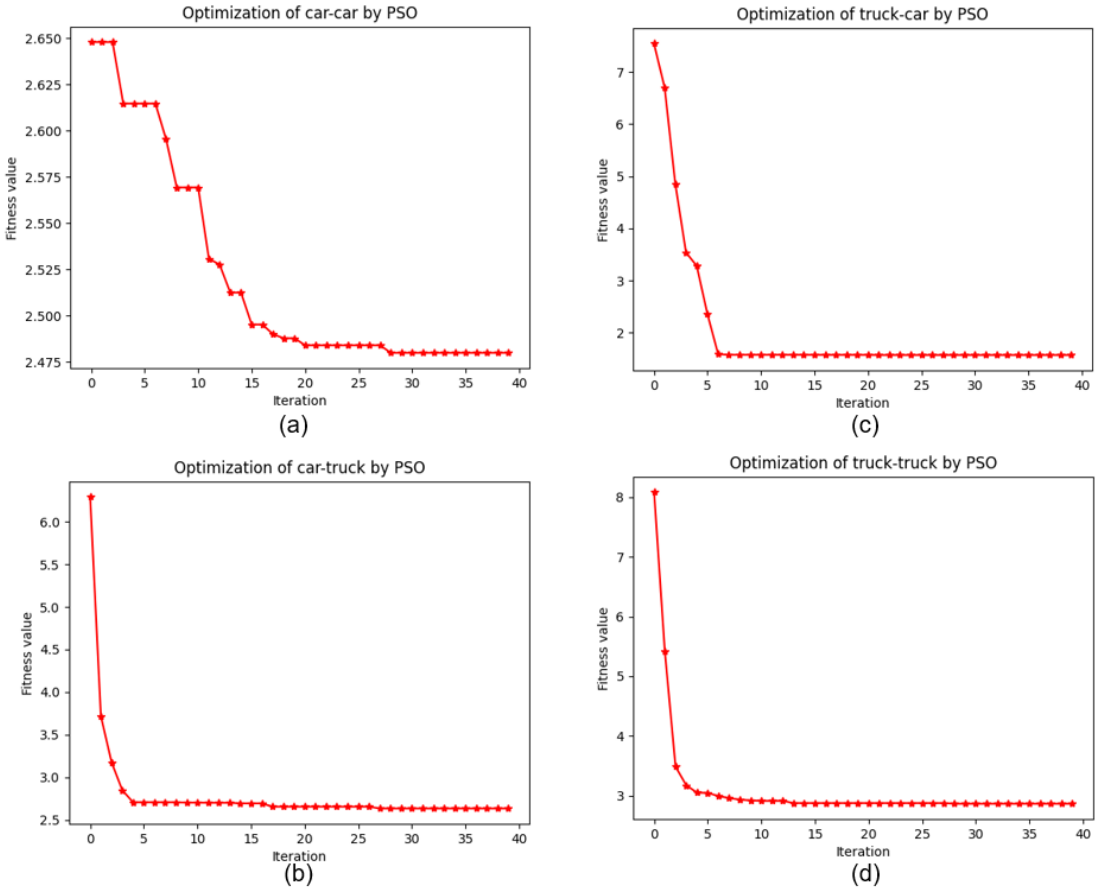


Figure 5.6 The optimization process by PSO-calibrator in different following-pair type: (a) car-car, (b) car-truck, (c) truck-car, and (d) truck-truck.

For PSO, three parameters have a relatively large impact on the convergence of the algorithm, namely the inertia weight parameter w, the cognitive coefficient c_1, and the social coefficient c_2. These three parameters affect the inertia and tendency of the particles when they move. Figure 5.7 and Figure 5.8 show the effect of the parameter w and parameters c on the convergence process of the PSO calibrator respectively, under the identical conditions.

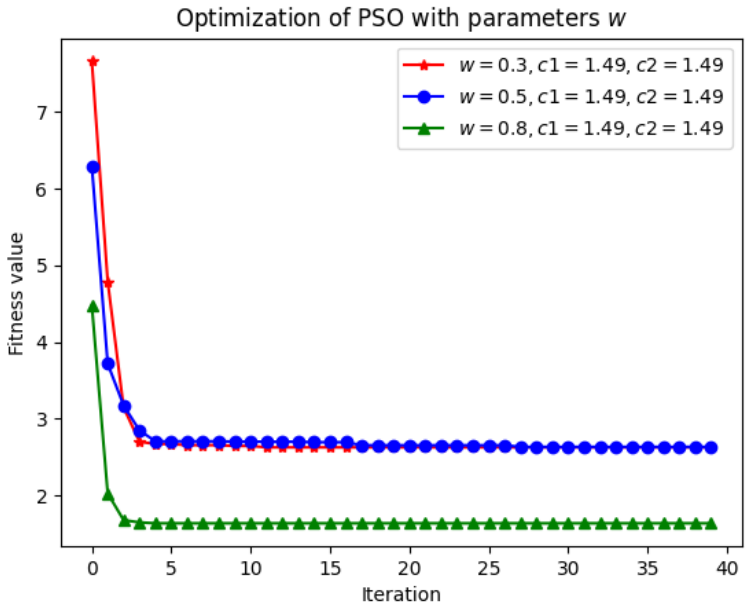


Figure 5.7 The influence of different inertia weight parameter on the convergence of the PSO calibrator under identical conditions: following-pair type car-truck, $nPop = 10$, $maxIter = 40$, $c_1 = c_2 = 1.49$

As shown in the figure, the PSO calibrator converges to a minimum at $w = 0.8$ for the inertia weight parameter. For values of 0.3 and 0.5, there is no significant difference in this optimisation problem.

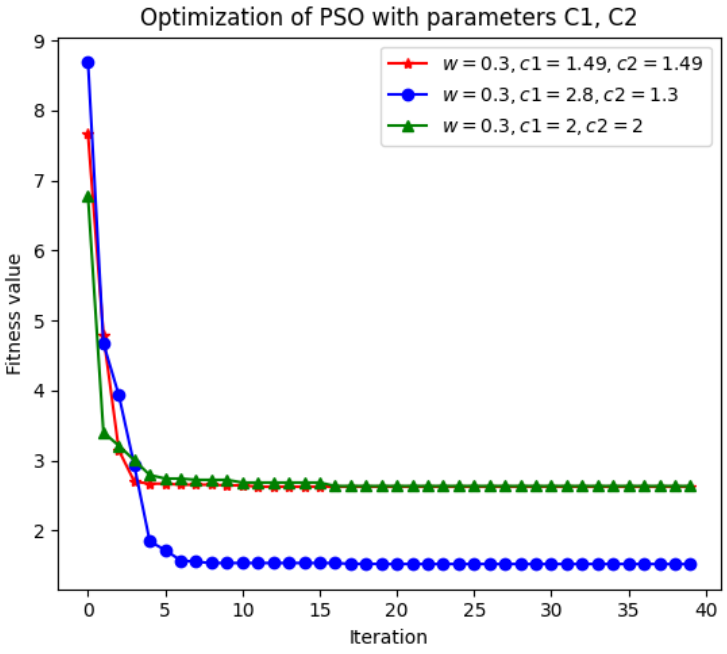


Figure 5.8 The influence of different cognitive and social coefficient on the convergence of the PSO calibrator under identical conditions: following-pair type car-truck, $nPop = 10$, $maxIter = 40$, $W = 0.3$

The cognition term represents tendency for individuals to replicate past behaviours that have proved successful, while the social term represents the tendency to follow the successful experiences of others. In mainstream research, c_1 and c_2 are frequently set to

2.0, as this allows the search to cover areas which are centred on PBEST and GBEST. Another commonly used value is 1.49445, which ensures convergence of the PSO algorithm. Carlisle has experimented extensively and has proposed a better set of parameter settings, setting $c_1 = 2.8$ and $c_2 = 1.3$ (Carlisle & Dozier, 2001). In the present optimisation problem, as shown in Figure 5.8, the set of parameters from Carlisle has a lower convergence value. For the other two options, there is no significant difference in convergence.

5.2.3. Comparison of GA and PSO

Comparing the convergence of the GA and PSO calibrators, it can be seen that PSO significantly outperforms GA in terms of the convergence speed of the algorithm. PSO requires fewer iterations to converge rapidly, while the undirected nature of mutation sacrifices some of the efficiency of the GA algorithm as the convergence of GA relies mainly on 'effective' mutation (i.e., individual in child generation obtains lower fitness values by mutation). The undirected nature of mutation sacrifices some of the efficiency of the GA algorithm. Therefore, in general, GA requires more iterations to converge than PSO.

In terms of convergence performance, the PSO algorithm usually achieves a lower fitness value than the GA as well. This situation is probably because the GA is still not completely converged after the maximum number of iterations. Therefore, in future improvements, a better convergence could be considered by increasing the number of iterations of the GA.

5.3. Optimization Performance Analysis

In order to understand the simulation ability of the calibrated model to car-following behaviour, the performance of the calibrated model needs to be assessed by comparing the simulated results of the model with the real observations.

5.3.1. Calibrated parameter values of W-99 model

Table 5.1 and Table 5.2 present the parameters of the W-99 model after calibration using GA and PSO respectively. Different parameter values are obtained by using data of different following-pair types.

Table 5.1 Calibrated parameters of W-99 model by GA calibrator

Parameter	description	Car-Car	Car-Truck	Truck-Car	Truck-Truck
CC_0 [m]	Standstill gap	4.8726	5.8139	3.1064	8.0051

Parameter	description	Car-Car	Car-Truck	Truck-Car	Truck-Truck
CC_1 [m]	Headway time	4.5812	4.9331	4.7698	4.7660
CC_2 [m]	“Following” variation	7.0157	9.9659	9.3601	7.2280
CC_3 [s]	Threshold for entering “following”	-16.2409	-9.2694	-12.8016	-12.6292
CC_4 [m/s]	Negative “following” threshold	-0.9385	-2.8122	-0.8511	-2.0539
CC_5 [m/s]	Positive “following” threshold	3.6455	0.9240	2.7024	4.8331
CC_6 [10^{-4} rad/s]	Speed dependency of oscillation	17.5722	8.7354	13.6691	17.9360
CC_7 [m/s^2]	Oscillation acceleration	-0.3954	0.6699	0.2151	-0.7996
CC_8 [m/s^2]	Standstill acceleration	4.3382	0.2371	4.4260	7.8375
CC_9 [m/s^2]	Acceleration at 80 km/h	2.2306	5.6909	7.6523	6.9053
$VDES$ [km/h]	Desired speed of following vehicle	84.4924	72.234	72.0601	73.3952

From Table 5.1 we could observe that the calibration results for the same parameter differed using different types of car-following data. Some of these results can be explained with the help of the actual meaning of the parameters. For example, the parameter CC_0, which represents the standstill gap between the two vehicles, and looking at the value of CC_0, we can see that there is a maximum standstill gap between the truck and the truck, a result that corresponds to the real situation.

Table 5.2 Calibrated parameters of W-99 model by PSO calibrator

Parameter	description	Car-Car	Car-Truck	Truck-Car	Truck-Truck
CC_0 [m]	Standstill gap	3.5862	0	0	1.2947
CC_1 [m]	Headway time	5	5	5	5
CC_2 [m]	“Following” variation	10	0	7.8606	10
CC_3 [s]	Threshold for entering “following”	-20	-7.5925	-17.2691	-18.8389
CC_4 [m/s]	Negative “following” threshold	-3.6509	-3.4204	-5	0
CC_5 [m/s]	Positive “following” threshold	0.1	0.4526	0.1	3.8594
CC_6 [10^{-4} rad/s]	Speed dependency of oscillation	0.1	9.1922	20	0.1
CC_7 [m/s^2]	Oscillation acceleration	0.3198	0.9752	-0.7654	0.9885

Parameter	description	Car-Car	Car-Truck	Truck-Car	Truck-Truck
CC_8 [m/s^2]	Standstill acceleration	8	0	0	0
CC_9 [m/s^2]	Acceleration at 80 km/h	5.9378	0	0	0
$VDES$ [km/h]	Desired speed of following vehicle	72	94.3696	94.0597	72

Comparing the two tables above, we can see that there are significant differences between the calibrated parameters of GA and PSO calibrators. When calibrating with the PSO calibrator, it is observed that the particles often converge to the parameter boundaries of the W-99 model. This situation did not occur in the GA calibrator. For parameter CC_1, the calibration converged to the upper boundary for all four following-pair types. Whereas for parameter CC_8, the results for car-car converge to the upper boundary, while the other three types converge to the lower boundary. It is difficult to conclude the pattern in terms of the parameter values. One conjecture for this phenomenon is that the setting of the boundaries may not exactly match the traffic scenario being calibrated. Another conjecture is that the parameters that converge to the boundaries are not strongly correlated with the adaptation values. In such cases, due to the algorithmic nature of PSO, it is likely to be observed that the particles converge to the boundaries.

5.3.2. Estimation performance evaluation

The distribution of relative errors could be used to assess the ability of the model to simulate realistic situations. Because of the frequent and sudden instantaneous changes in acceleration, the relative error in velocity is used for error analysis in this thesis in the section discussing calibration errors. Figures 5.9 and 5.10 illustrate the relative error in velocity for calibration of the W-99 model using the GA and PSO calibrators respectively. The histograms of the error distribution are differentiated by the type of car-following pairs.

Figure 5.9 shows the relative error distribution between the predicted velocity and the actual observed values for the W-99 model after calibration by using the GA calibrator. It is observed that the relative error is generally distributed between the range from -0.1 to 0.1 for the four types of following pairs and the distribution peaks are around zero. This means that the relative error is less than 10%. The W-99 model calibrated using the data with the following-pair type of truck-truck has a more concentrated relative error distribution, indicating that it is better calibrated. One reason for this result is the velocity of trucks cannot change significantly suddenly.

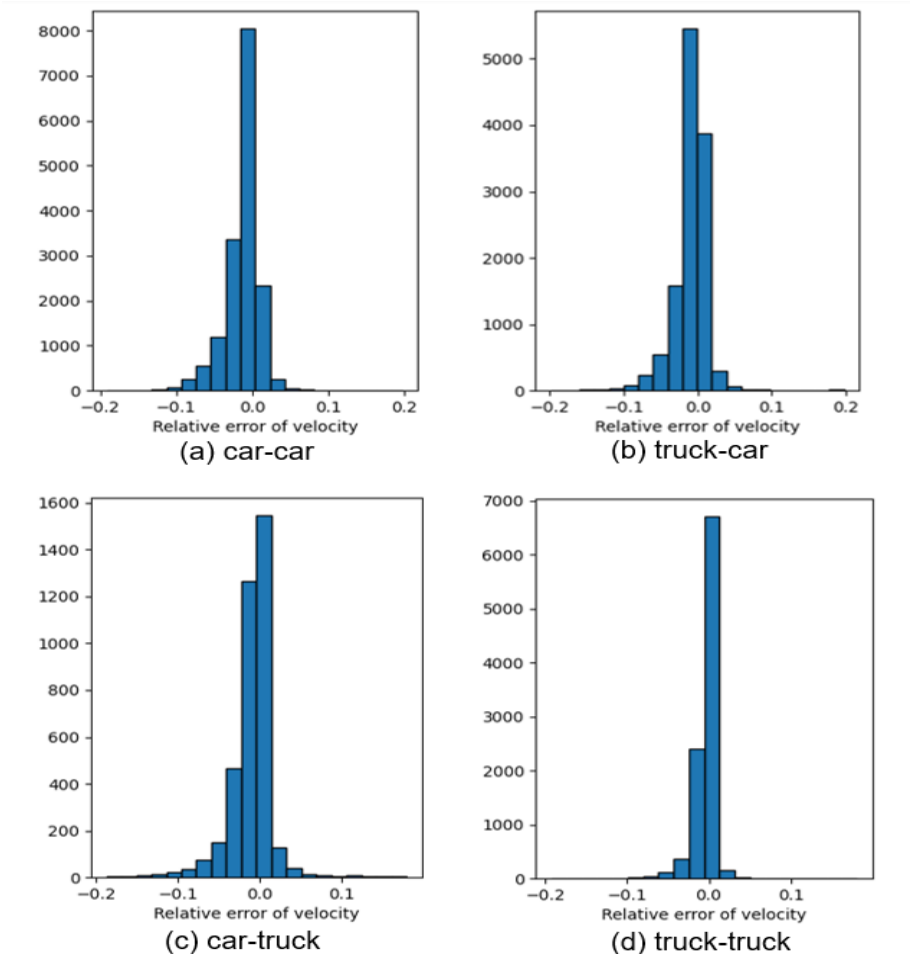


Figure 5.9 Relative error of velocity by GA calibrator in different following-pair types: (a) car-car, (b) car-truck, (c) truck-car, and (d) truck-truck.

For the W-99 model calibrated using the PSO calibrator, the distribution of relative errors is quite similar to that using the GA calibrator. Figure 5.10 demonstrates this intuitively. Similarly, the relative error of calibration using the PSO calibrator is less than 10% and the relative error distribution is more concentrated for the truck-truck type.

The previous analysis shows that the truck-truck group, which converged less well in the calibration process, obtained the most concentrated relative error distribution in the relative error analysis. That is, although the optimization process did not converge well, the calibrated parameters still simulated the actual situation relatively well. This phenomenon is perhaps an implication, pointing to the fact that the fitness function we have chosen for convergence in this study perhaps does not exactly correspond with the optimization objectives.

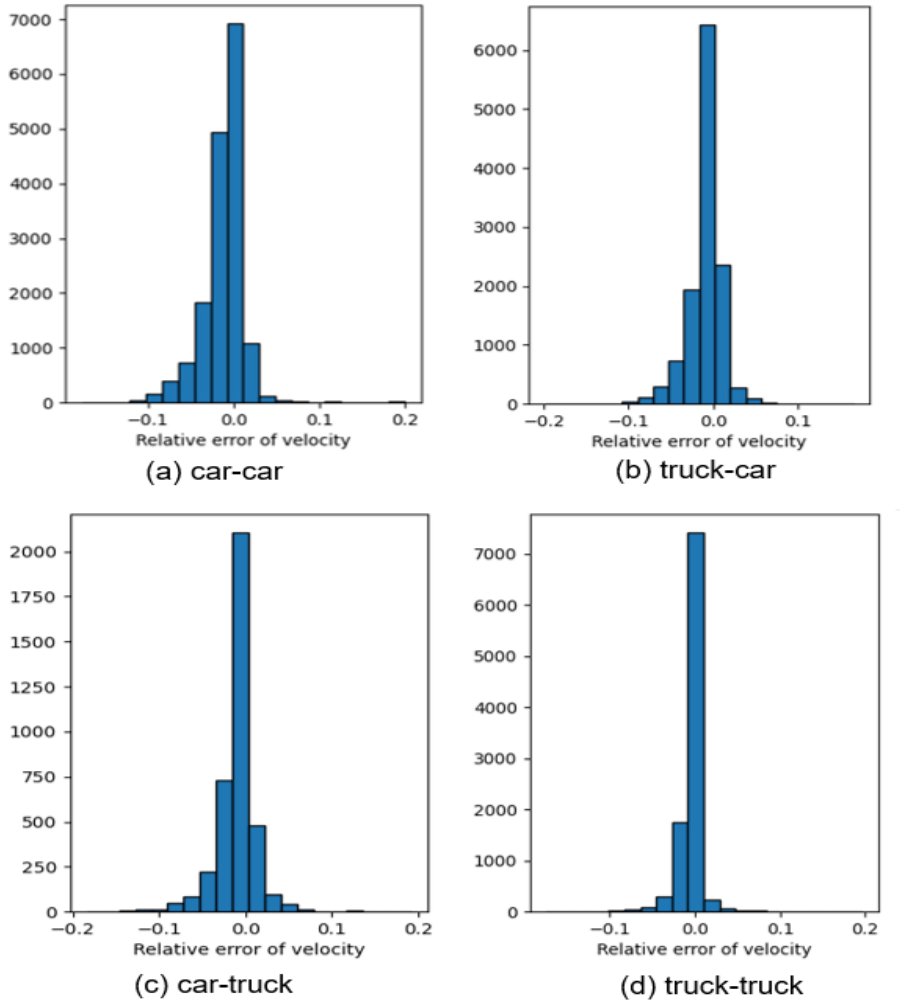


Figure 5.10 Relative error of velocity by PSO calibrator in different following-pair types: (a) car-car, (b) car-truck, (c) truck-car, and (d) truck-truck.

5.3.3. Simulation by calibrated model

The next section will focus on individuals in the traffic flow. An individual vehicle is selected and the acceleration at the next time frame is simulated by using the calibrated W-99 model from the first frame, and then the predicted velocity for the next time frame is calculated by using the predicted acceleration and the current velocity. the time series of the predicted velocity and the observed velocity for that individual are plotted in the same graph. By this graph, it is possible to obtain a more intuitive picture of the model's ability to simulate the trajectory of an individual vehicle.

After generating the simulation plots from the model calibrated by GA and PSO, it was found that the GA and PSO calibrated models presented very similar results in the individual traffic behaviour simulation plots, therefore here only the GA-calibrated W-99 model is analysed here as an example for the single vehicle simulation.

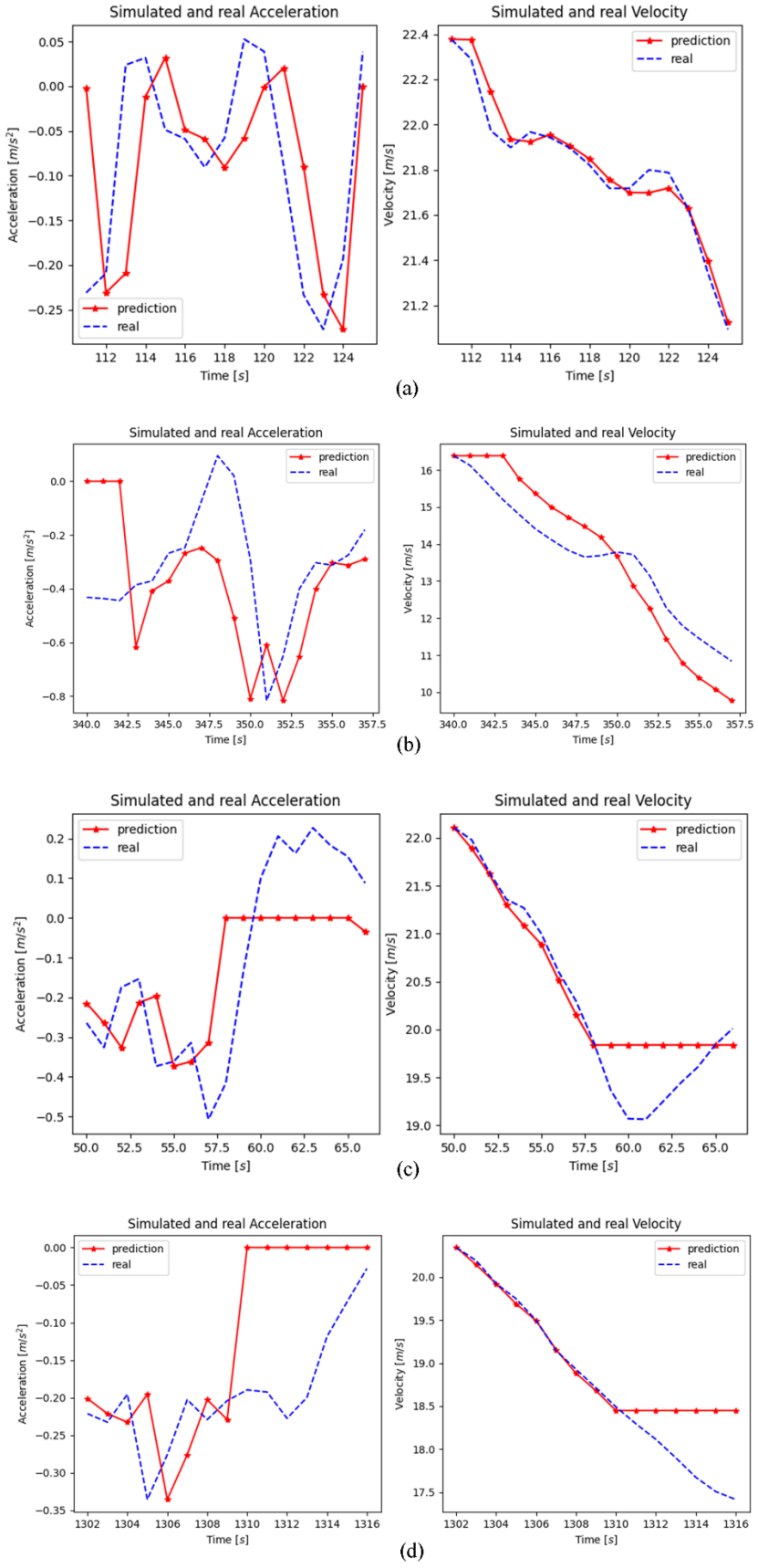


Figure 5.11 Simulation of single vehicle by GA-calibrated W-99 model of different following-pair in accelration and velocity: (a) car-car, (b) car-truck, (c) truck-car, and (d) truck-truck

As can be seen in Figure 5.11(a), the single-vehicle selected in the car-car following pair is better modelled. The time series of acceleration is plotted on the left and the velocity

is plotted on the right. From the acceleration plot, we can observe that the predicted and observed values have the same shaped time series curve, but there is a deviation of about a one-time frame delay between the predicted and observed values. When reflected in the velocities, the time series of predicted and observed values approximately overlap.

In figures 5.11 (b), (c), and (d), the predicted and observed accelerations maintain shape consistency in only a portion of the time frames, while large deviations occur in other parts. This large deviation of acceleration has an impact on the simulation results for velocity. In figure 5.11 (c), for example, the predicted acceleration remains at zero after 58 second, whereas in the actual observation the acceleration rises to greater than zero from about 60 seconds and remains positive. This is reflected in the velocity plot, where the predicted speed of the vehicle remains constant after 58 second, while the actual speed of the vehicle stays up after a small drop until it crosses the curve with the predicted speed value. One possible explanation for this phenomenon of successive anomalies in the acceleration predictions at some of the time frames is due to the limitations of the W-99 model. The fact that the predicted acceleration remains constant from a certain point onward may be because certain thresholds in the model settings have been reached. This suggests that for a particular traffic situation there may be further possibilities for the model to be refined from a design perspective.

The relationship between the predicted and observed values of velocity can be obtained by setting them as the x and y axes respectively. This result is illustrated in Figure 5.12. Again, as the GA and PSO results are very similar, therefore only the GA results is shown here as an example.

As shown in Figure 5.12, although there are some deviations, in general, the true speed variation is simulated by the model. the data points for the car-car following pair are closest to the 45-degree line, which means that the car-following model calibrated by using data of this type could simulates good the same car-following scenario in reality better. For the other three types of following pairs, the speed of the rear vehicle was able to coincide partially with the 45-degree line (as in figures (c) and (d)), or the speed points are still generally distributed around the 45-degree line despite the error between them (as in figure (b)).

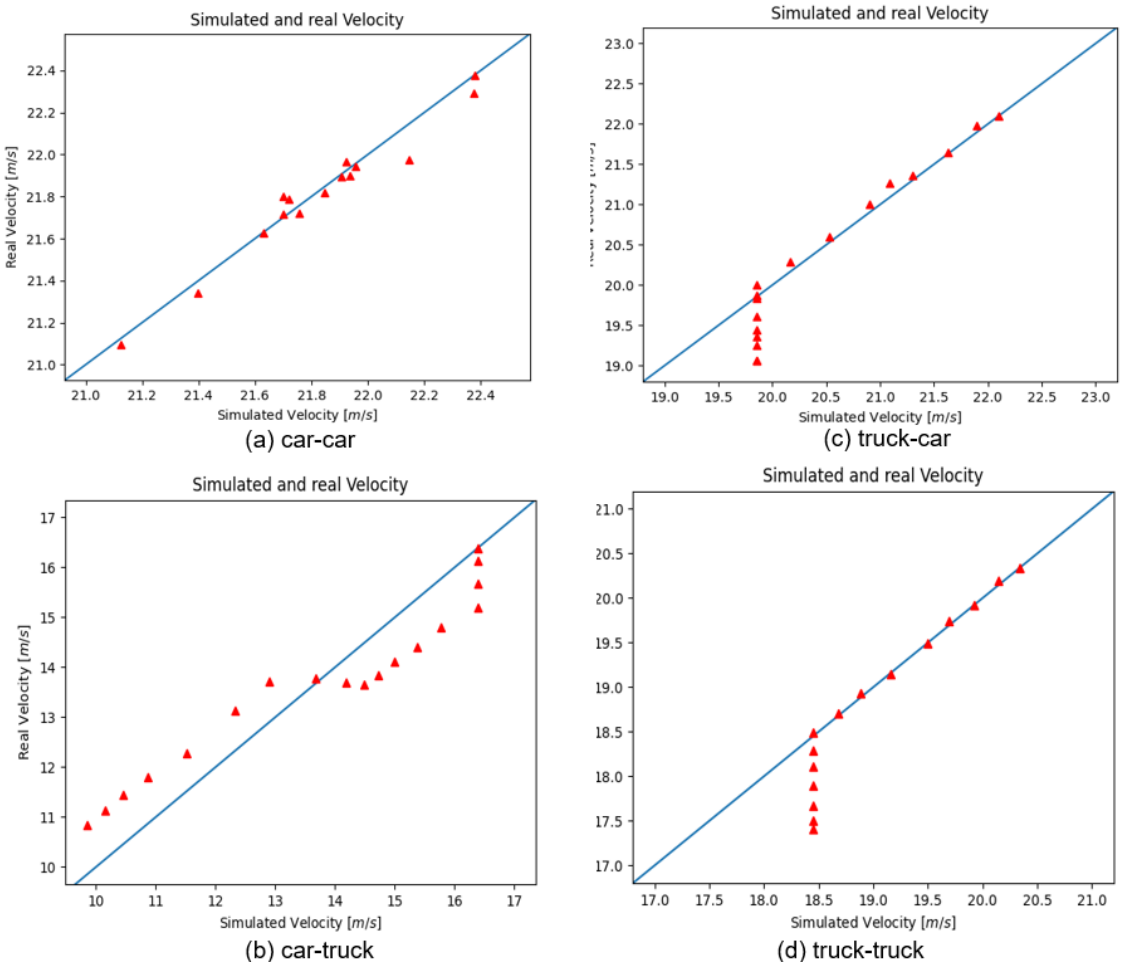


Figure 5.12 45-degree-line of simulated velocity by GA calibrated W-99 model and the observed velocity of different following-pair in accelration and velocity: (a) car-car, (b) car-truck, (c) truck-car, and (d) truck-truck

6. Conclusion and Outlook

In this chapter, a summary and outlook are given for this study. Section 6.1 reviews and summarises the methodology, experiments, and analysis of this thesis. In section 6.2, some shortcomings in this study and possible directions for improvement in subsequent work are discussed.

6.1. Conclusion

The main objective of this thesis is to calibrate the car-following model, and to obtain a model that is able to simulate the car-following behaviour and characteristics in microscopic traffic flows in the entrance/exit areas of German highways. The thesis uses the exiD dataset to obtain car-following data for this specific traffic scenario, which was collected for seven entrance/exit areas on a section of the German highway by using a drone. The filtered following pairs were divided into four datasets based on the vehicle types of the preceding and following vehicles, and in a subsequent process, these data were fed individually into a calibrator to calibrate the target car-following model and ultimately resulting in different sets of parameters.

The Wiedemann 99 model was chosen as the model to be calibrated in this thesis, as it is the most comprehensive model at present for the logical description of driver behaviour and can describe most of the observed features of everyday driving behaviour well and is used frequently in the simulation of highway microscopic traffic flows. Because of the large number of parameters (11 parameters) that need to be calibrated in the Wiedemann model, calibration by using traditional numerical methods is difficult and therefore this thesis has chosen to use meta-heuristic algorithms, namely GA and PSO, for calibration.

After initialising the parameters of the Wiedemann model and the calibrators of GA and PSO, the following-pair data is fed into the calibrators respectively and the calibrated results are obtained after optimization of the algorithm. The performance of both GA and PSO algorithms in this optimisation problem is discussed by analysing the convergence of the algorithms. In order to evaluate the effectiveness of the calibration, the acceleration and velocity of the following vehicle in all following pairs were simulated using the calibrated car-following model, and the resulting velocity time series were taken as relative errors to the actual observations in the dataset. By looking at the distribution of the relative errors, it was observed that after calibration, the relative errors in the model predictions were less than 10%.

To observe the simulation of the calibrated model for an individual vehicle, a single vehicle was selected, and the velocity and acceleration of that vehicle were predicted by using the calibrated W99 model and the time series of predicted values were plotted on the same line graph as the real values. We can observe that for acceleration the prediction error is greater than for velocity. In some of the line plots, the velocities simulated by the model essentially overlap with the observed values. At some points in time, the model appears to fail in its predictions, with the predicted acceleration ceasing to change after a certain point is reached. Presumably this occurs because a certain threshold of the model has been reached. This may mean that the model is still not a good enough description for the particular traffic scenario it is calibrated for.

6.2. Outlook

The experiment results show that GA and PSO are applicable to calibrate Wiedemann models for highway on- and off-ramps in general. However, there are still some issues that can be improved in future work.

1) The lack of convergence in the GA algorithm. Although we have observed convergence in the GA, the minimum of fitness value obtained is significantly higher than the PSO. By increasing the population size and the number of iterations, it is possible to increase the possibility of good mutant genes appearing and thus GA could potentially gain further convergence.
2) The particles of PSO always converge at the bound for some parameters. For this problem, the bounds can be set manually in subsequent operations to avoid particles converging at the boundary values. Another approach that could be attempted is referencing Wang's operation when using PSO for multi-parameter optimization (Wang et al., 2023). Before the optimization starts, a correlation analysis is performed on the parameters to be calibrated and those with high correlation are selected for calibration. The distribution of the error values in space is obtained by Monte Carlo methods and Latin hypercube sampling, allowing visual observation of whether the lowest value of the error is indeed located at the boundary.
3) The calibrator operates inefficiently. This is one of the reasons for the convergence of the algorithm. To solve this problem, multiprocessing can be added to the GA and PSO calibrators to speed up the objective function in subsequent operations, which have been shown to significantly reduce the operation time.
4) The simulated situation for a single vehicle is not representative of the global simulation of traffic flow. When performing single-vehicle simulations, we have also observed that the model gives better results for some vehicles and worse results

for others. This is a bias due to the individual driving style and behaviour of the vehicle driver. Therefore, in order to evaluate the simulation of the model for the traffic flow overall, it is necessary to use traffic simulation software with the Wiedemann model, such as SUMO, to simulate each particle of the traffic flow separately using the calibrated parameters. The simulation data is then analysed. An assessment of the consistency of the traffic flow can be achieved by comparing the fundamental diagram of traffic flow between simulated and observed values.

References

Aghabayk, K., Sarvi, M., & Young, W. (2016). Including heavy vehicles in a car‐following model: modelling, calibrating and validating. *Journal of Advanced Transportation*, *50*(7), 1432-1446.

Ahmed, H. U., Huang, Y., & Lu, P. (2021). A review of car-following models and modeling tools for human and autonomous-ready driving behaviors in micro-simulation. *Smart Cities*, *4*(1), 314-335.

Bando, M., Hasebe, K., Nakayama, A., Shibata, A., & Sugiyama, Y. (1995). Dynamical model of traffic congestion and numerical simulation. *Physical Review E*, *51*(2), 1035.

Barceló, J., & Casas, J. (2005). Dynamic network simulation with AIMSUN. In *Simulation approaches in transportation analysis* (pp. 57-98). Springer.

Brackstone, M., & McDonald, M. (1999). Car-following: a historical review. *Transportation Research Part F: Traffic Psychology and Behaviour*, *2*(4), 181-196.

Carlisle, A., & Dozier, G. (2001). Tracking changing extrema with particle swarm optimizer. *Auburn Univ., Auburn, AL, Tech. Rep. CSSE01-08*.

Chaudhari, A. A., Karthik, K., Chilukuri, B. R., Treiber, M., & Okhrin, O. (2022). Calibrating Wiedemann-99 model parameters to trajectory data of mixed vehicular traffic. *Transport. Res. Rec*.

Chong, L., Abbas, M. M., Flintsch, A. M., & Higgs, B. (2013). A rule-based neural network approach to model driver naturalistic behavior in traffic. *Transportation Research Part C: Emerging Technologies*, *32*, 207-223.

Dadashzadeh, N., Ergun, M., Kesten, S., & Žura, M. (2019). An automatic calibration procedure of driving behaviour parameters in the presence of high bus volume. *Promet-Traffic&Transportation*, *31*(5), 491-502.

Durrani, U., Lee, C., & Maoh, H. (2016). Calibrating the Wiedemann's vehicle-following model using mixed vehicle-pair interactions. *Transportation Research Part C: Emerging Technologies*, *67*, 227-242.

Eberhart, R., & Kennedy, J. (1995). A new optimizer using particle swarm theory. MHS'95. Proceedings of the sixth international symposium on micro machine and human science,

Fellendorf, M., & Vortisch, P. (2010). Microscopic traffic flow simulator VISSIM. In *Fundamentals of traffic simulation* (pp. 63-93). Springer.

Fernandez, A. (2011). *Secondary tasks in steady state car following situations*

Gazis, D. C., Herman, R., & Rothery, R. W. (1961). Nonlinear follow-the-leader models of traffic flow. *Operations research*, *9*(4), 545-567.

Gipps, P. G. (1981). A behavioural car-following model for computer simulation. *Transportation Research Part B: Methodological*, *15*(2), 105-111.

Guo, F. (2020). *scikit-opt* https://github.com/guofei9987/scikit-opt/

Hamdar, S. H., Mahmassani, H. S., & Treiber, M. (2015). From behavioral psychology to acceleration modeling: Calibration, validation, and exploration of drivers' cognitive and safety parameters in a risk-taking environment. *Transportation Research Part B: Methodological*, *78*, 32-53.

He, Z., Zheng, L., & Guan, W. (2015). A simple nonparametric car-following model driven by field data. *Transportation Research Part B: Methodological*, *80*, 185-201.

Holland, J. H. (1992). *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press.

Hu, X.-B., & Di Paolo, E. (2009). An efficient genetic algorithm with uniform crossover for air traffic control. *Computers & Operations Research*, *36*(1), 245-259.

Koutsopoulos, H. N., & Farah, H. (2012). Latent class model for car following behavior. *Transportation Research Part B: Methodological*, *46*(5), 563-578.

Lazar, H., Rhoulami, K., & Rahmani, D. (2016). A review analysis of optimal velocity models. *Periodica Polytechnica Transportation Engineering*, *44*(2), 123-131.

LeBlanc, D. J., Bao, S., Sayer, J. R., & Bogard, S. (2013). Longitudinal driving behavior with integrated crash-warning system: Evaluation from naturalistic driving data. *Transportation research record*, *2365*(1), 17-21.

Leyn, U., & Vortisch, P. (2015). Calibrating VISSIM for the German highway capacity manual. *Transportation research record*, *2483*(1), 74-79.

Li, L., Chen, X. M., & Zhang, L. (2016). A global optimization algorithm for trajectory data based car-following model calibration. *Transportation Research Part C: Emerging Technologies*, *68*, 311-332.

Liu, P., & Fan, W. (2020). Exploring the impact of connected and autonomous vehicles on freeway capacity using a revised Intelligent Driver Model. *Transportation planning and technology*, *43*(3), 279-292.

Liu, Y., Zou, B., Ni, A., Gao, L., & Zhang, C. (2021). Calibrating microscopic traffic simulators using machine learning and particle swarm optimization. *Transportation letters*, *13*(4), 295-307.

Liu, Z., & Yang, G. (2022). Large-scale traffic flow simulation based on intelligent PSO. MATEC Web of Conferences,

Manjunatha, P., Vortisch, P., & Mathew, T. V. (2013). Methodology for the Calibration of VISSIM in Mixed Traffic. Transportation research board 92nd annual meeting,

Marini, F., & Walczak, B. (2015). Particle swarm optimization (PSO). A tutorial. *Chemometrics and Intelligent Laboratory Systems*, *149*, 153-165.

Moers, T., Vater, L., Krajewski, R., Bock, J., Zlocki, A., & Eckstein, L. (2022). The exiD dataset: a real-world trajectory dataset of highly interactive highway scenarios in germany. 2022 IEEE Intelligent Vehicles Symposium (IV),

Olstam, J. J., & Tapani, A. (2004). *Comparison of Car-following models* (Vol. 960). Swedish National Road and Transport Research Institute Linköping, Sweden.

Park, B., & Qi, H. (2005). Development and Evaluation of a Procedure for the Calibration of Simulation Models. *Transportation research record*, *1934*(1), 208-217.

Pipes, L. A. (1953). An operational analysis of traffic dynamics. *Journal of applied physics*, *24*(3), 274-281.

Pourabdollah, M., Bjärkvik, E., Fürer, F., Lindenberg, B., & Burgdorf, K. (2017). Calibration and evaluation of car following models using real-world driving data. 2017 IEEE 20th International conference on intelligent transportation systems (ITSC),

Punzo, V., & Simonelli, F. (2005). Analysis and comparison of microscopic traffic flow models with real traffic microscopic data. *Transportation research record*, *1934*(1), 53-63.

Rockwell, T. H., Ernst, R. L., & Hanken, A. (1968). A sensitivity analysis of empirically derived car-following models. *Transportation Research/UK/*.

Rrecaj, A. A., & MBombol, K. (2015). Calibration and Validation of the VISSIM Parameters-State of the Art. *TEM journal*, *4*(3), 255.

Saifuzzaman, M., & Zheng, Z. (2014). Incorporating human-factors in car-following models: a review of recent developments and research needs. *Transportation Research Part C: Emerging Technologies*, *48*, 379-403.

Sheppard, J., & Haberman, B. (2021). Using Particle Swarm Optimization to Learn a Lane Change Model for Autonomous Vehicle Merging. 2021 IEEE Symposium Series on Computational Intelligence (SSCI),

Sun, D., Benekohal, R. F., & Waller, S. T. (2006). Bi‐level programming formulation and heuristic solution approach for dynamic traffic signal optimization. *Computer‐Aided Civil and Infrastructure Engineering*, *21*(5), 321-333.

Treiber, M., & Helbing, D. (2003). Memory effects in microscopic traffic models and wide scattering in flow-density data. *Physical Review E*, *68*(4), 046119.

Treiber, M., Hennecke, A., & Helbing, D. (2000). Congested traffic states in empirical observations and microscopic simulations. *Physical Review E*, *62*(2), 1805.

Treiber, M., & Kesting, A. (2013). Traffic flow dynamics. *Traffic Flow Dynamics: Data, Models and Simulation, Springer-Verlag Berlin Heidelberg*, 983-1000.

Wang, H., Sun, C., Haidn, O., Aliya, A., Manfletti, C., & Slavinskaya, N. (2023). A joint hydrogen and syngas chemical kinetic model optimized by particle swarm optimization. *Fuel*, *332*, 125945.

Wiedemann, R. (1974). Simulation des Strassenverkehrsflusses.

Wiedemann, R., & Reiter, U. (1992). *Microscopic Traffic Simulation: The Simulation System MISSION, Background and Actual State* (Project ICARUS (V1052) Final Report, Issue.

Yu, M., & Fan, W. D. (2017). Calibration of microscopic traffic simulation models using metaheuristic algorithms. *International Journal of Transportation Science and Technology*, *6*(1), 63-77.

Zhu, M., Wang, X., & Tarko, A. (2018). Modeling car-following behavior on urban expressways in Shanghai: A naturalistic driving study. *Transportation Research Part C: Emerging Technologies*, *93*, 425-445.

Declaration

I hereby confirm that the presented thesis work has been done independently and using only the sources and resources as are listed. This thesis has not previously been submitted elsewhere for purposes of assessment.

Place, Date, Signature