Macroscopic Modeling of the Motorway Traffic in Sweden: A Calibration Study

by

Anahita Zahertar

B.S., Civil Engineering, K.N. TOOSI University of Technology, 2016

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Civil Engineering

Boğaziçi University

2019

Macroscopic Modeling of the Motorway Traffic in Sweden: A Calibration Study

APPROVED BY:

Assoc. Prof. Dr. Ilgın GÖKAŞAR .....
(Thesis Supervisor)

Assist. Prof. Gürkan GÜNAY .....

Assist. Prof. Dr. Mehtap IŞIK .....

DATE OF APPROVAL: 03.12.2019

ACKNOWLEDGEMENTS

I would first like to thank Assoc. Prof. Dr. Ilgın GÖKAŞAR from Boğaziçi University, and Assoc. Prof. Dr. Xiaoliang MA from KTH Royal Institute of Technology whose expertise were invaluable in the formulating of the research topic and methodology in particular. In addition, my sincere thanks also goes to Dr. Jing GONG from KTH Royal Institute of Technology for his precious help and contribution throughout my thesis.

Furthermore, I would like to thank my parents and my sister for their wise counsel and sympathetic ear and support. You are always there for me. Finally, there are my friends, who were of great support in deliberating over our problems and findings, as well as providing happy distraction to rest my mind outside of my research, especially, SAMEANTA.

ABSTRACT

Macroscopic Modeling of the Motorway Traffic in Sweden: A Calibration Study

Traffic congestion is an important factor in today's urban life which has a direct influence on people's quality of life. This phenomenon usually happens on peak times when people are commuting to work. As people are stuck in the traffic congestion, not only it will take more time to reach their destination, but also this wasted time can be of their work's time meaning a loss of money. In addition to that, congestion may cause pollution. Since more vehicles are stuck in traffic and cannot advance properly, higher amount of gas is ejected to the air. Thus, tackling congestion problem is critical. There are several factors that lead to traffic congestion including the times when roads are full of vehicles, or when there are constructions, maintenance or incidents on roads. These factors cause a disruption in traffic flow. It is important to note that resolving congestion problem cannot only be done by increasing capacity and by constructing new roads and motorways; It also requires developing a transportation management technology. Although the solution of building new roads may seem interesting at first sight, it is not practical as it may cause even more problems. Therefore, utilizing the existing roads and enhancing the traffic management technology would be a better solution. Having a calibrated traffic flow model that reproduces the behavior of drivers on the real road can be of benefit. A calibrated and simulated microscopic traffic flow model is used in this thesis to calibrate a second-order macroscopic traffic flow model to represent the traffic flow on the motorway in Sweden by validation. For this purpose, different scenarios which represent free flow, intermediate flow and congestion have been studied. This model can be used to better manage the traffic and implement control strategies on motorways.

ÖZET

İsveç'teki Otoyol Trafiğinin Makroskopik Modellenmesi: Bir Kalibrasyon Çalışması

Trafik sıkışıklığı, günümüz şehir yaşamında, insanların yaşam kalitesini doğrudan etkileyen bir faktördür. Yolcular trafik sıkışıklığı dolayısıyla zaman kaybetmenin yanı sıra, ekonomik olarak da olumsuz etklineceklerdir. Ayrıca trafik sıkışıklığı kirliliğe de neden olabilmektedir. Trafikte sıkışmış ve düzenli hareket edemeyen araçlardan dolayı havaya salınan zararlı gaz miktarı da artış gösterebilir. Bu nedenlerden dolayı, trafik sıkışıklığı önemli bir problemdir. Yolda yüksek miktarda araç olması ya da yol üzerinde çalışma ya da kaza olması gibi olaylar trafik sıkışıklığına yol açan örneklerdendir. Bu faktörler trafik akışında bir bozulmaya neden olur. Trafik sıkışıklığı sorunu yalnızca kapasiteyi yeni yollar yaparak arttırmakla çözülemez. Yeni bir trafik yönetimi teknolojisi oluşturulmalıdır. Her ne kadar yeni yollar inşa etmek ilk anda daha cazip gelse de, pratik bir çözüm değildir ve daha büyük problemlere yol açabilir. Bu yüzden var olan yolları trafik yönetimi teknolojileriyle iyileştirmek daha iyi bir çözüm yoludur. Sürücü davranışlarını gerçekçi bir şekilde gösterebilen, gerçek bir yol üzerinde kalibre edilmiş bir trafik akımı modeli bu amaç için faydalı olacaktır. Bu amaçla serbest akım, orta seviye akım ve sıkışık trafik akımı senaryoları incelenmiştir. Kurulan model ile trafiği daha iyi yönetmek ve kontrol stratejilerini uygulamak mümkün olacaktır.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS . . . iii
ABSTRACT . . . iv
ÖZET . . . v
LIST OF FIGURES . . . ix
LIST OF TABLES . . . xv
LIST OF SYMBOLS . . . xvii
LIST OF ACRONYMS/ABBREVIATIONS . . . xviii
1. INTRODUCTION . . . 1
1.1. Motivation . . . 1
1.2. Aim and Objectives . . . 3
1.3. Scope . . . 4
1.4. History of Traffic Flow Theory and Models . . . 4
1.5. Background and Concepts . . . 6
1.5.1. Microscopic Models . . . 6
1.5.2. Macroscopic Models . . . 7
1.5.2.1. Flow . . . 7
1.5.2.2. Speed . . . 7
1.5.2.3. Density . . . 8
1.5.2.4. Fundamental Diagram . . . 9
1.5.2.5. First-Order Macroscopic Models . . . 10
1.5.3. Higher-Order Macroscopic Models . . . 13
1.5.3.1. The METANET Model . . . 15
1.5.3.2. METANET Model's Applications . . . 16
1.5.4. Calibration of Second-Order Traffic Flow Models . . . 17
1.5.5. Validation of Second-Order Traffic Flow Models . . . 19
1.6. Thesis Outline . . . 21
2. Methodology . . . 22
2.1. METANET . . . 22

2.2. Calibration . . . 28
2.3. Choice of methods for data collection, data processing and analysis . . 41
2.4. Validation of the model . . . 42
3. Computational Experiments and Results . . . 43
3.1. SUMO . . . 43
3.2. METANET . . . 47
3.3. Calibration and Validation Results . . . 51
3.3.1. First Scenario: One-Lane Closed with a Flow of 4500 veh/h . . 56
3.3.1.1. Density . . . 59
3.3.1.2. Velocity . . . 63
3.3.1.3. Flow . . . 67
3.3.1.4. Analysis . . . 71
3.3.2. Second Scenario: One-Lane Closed Scenario with a Flow of 3000 veh/h . . . 71
3.3.2.1. Density . . . 72
3.3.2.2. Velocity . . . 76
3.3.2.3. Flow . . . 79
3.3.2.4. Analysis . . . 82
3.3.3. Third Scenario: Free Flow Scenario . . . 82
3.3.3.1. Density . . . 83
3.3.3.2. Velocity . . . 87
3.3.3.3. Flow . . . 90
3.3.3.4. Analysis . . . 93
4. Conclusion . . . 94
4.1. Discussion . . . 94
4.2. Future Work . . . 95
REFERENCES . . . 97
APPENDIX A: The Generalized Pattern Search Method . . . 102
A.1. The Pattern . . . 102
A.2. Bound Constrained Exploratory Moves . . . 103
A.3. Generalized Pattern Search Method . . . 103

A.4. Update Δ_k 103
APPENDIX B: Syntax and Parameters of the `patternsearch` function 104
APPENDIX C: Convergence of the parameters for Segments 2 to 5 and 7 . . . 109

LIST OF FIGURES

Figure 1.1. Greenshields using his camera to measure the traffic flow. 4

Figure 1.2. Fundamental diagram of flow - density. 10

Figure 2.1. Representation of METANET's variables on a road segment. . . . 25

Figure 2.2. Comparison of the METANET and SUMO data for segments 3, 4 and 5 for the different objective functions. 32

Figure 2.3. Flowchart of the MATLAB calibration. 33

Figure 2.4. Flowchart of the MATLAB calibration. 36

Figure 2.5. Influence of the parameters (from left to right) $\rho_{cr}, \rho_{max}, \alpha, \nu, \kappa, \tau$ on the objective function. 38

Figure 2.6. Convergence of the parameters $[\rho_{cr}, \rho_{max}, \nu, \kappa]$ of Segment 1 over 400 Iterations. 40

Figure 2.7. Convergence of the parameters $[\rho_{cr}, \rho_{max}, \nu, \kappa]$ of Segment 6 over 400 Iterations. 40

Figure 3.1. The road stretch viewed from Google Maps. 43

Figure 3.2. Scheme of the network modeled in SUMO. 44

Figure 3.3. View of the studied road stretch in `sumo-gui`. 47

Figure 3.4. Flowchart of the `main.py` Python file. 48

Figure 3.5. Flowchart of the progression of the simulation in the file `main.py`. 49

Figure 3.6. Flowchart of the METANET model. 50

Figure 3.7. Convergence of the *patternsearch* function. 53

Figure 3.8. Lane-Closure on Segment 4 and 5 in SUMO using the Rerouter feature. 56

Figure 3.9. Stop-and-go waves phenomenon with the SUMO's data. 57

Figure 3.10. 3D plot of the density from METANET data 59

Figure 3.11. 3D plot of the density from METANET data (View from the top) 60

Figure 3.12. 3D plot of the density from SUMO data 60

Figure 3.13. 3D plot of the density from SUMO data (View from the top) . . . 61

Figure 3.14. 2D plots of the density for segments 1 to 7 (starting at top left position). The blue curve is METANET's data and the orange curve is SUMO's data. 62

Figure 3.15. 3D plot of the Velocity from METANET data 63

Figure 3.16. 3D plot of the Velocity from METANET data (View from the top) 64

Figure 3.17. 3D plot of the Velocity from SUMO data 64

Figure 3.18. 3D plot of the Velocity from SUMO data (View from the top) . . 65

Figure 3.19. 2D plots of the velocity for segments 1 to 7 (starting at top left position). The blue curve is METANET's data and the orange curve is SUMO's data. 66

Figure 3.20. 3D plot of the Flow from METANET data 67

Figure 3.21. 3D plot of the Flow from METANET data (View from the top) . 68

Figure 3.22. 3D plot of the Flow from SUMO data 68

Figure 3.23. 3D plot of the Flow from SUMO data (View from the top) 69

Figure 3.24. 2D plots of the flow for segments 1 to 7 (starting at top left position). The blue curve is METANET's data and the orange curve is SUMO's data. 70

Figure 3.25. 3D plot of the density from METANET data 73

Figure 3.26. 3D plot of the density from METANET data (View from the top) 73

Figure 3.27. 3D plot of the density from SUMO data 74

Figure 3.28. 3D plot of the density from SUMO data (View from the top) . . . 74

Figure 3.29. 2D plots of the density for segments 1 to 7 (starting at top left position). The blue curve is METANET's data and the orange curve is SUMO's data. 75

Figure 3.30. 3D plot of the Velocity from METANET data 76

Figure 3.31. 3D plot of the Velocity from METANET data (View from the top) 76

Figure 3.32. 3D plot of the Velocity from SUMO data 77

Figure 3.33. 3D plot of the Velocity from SUMO data (View from the top) . . 77

Figure 3.34. 2D plots of the velocity for segments 1 to 7 (starting at top left position). The blue curve is METANET's data and the orange curve is SUMO's data. 78

Figure 3.35. 3D plot of the Flow from METANET data 79

Figure 3.36. 3D plot of the Flow from METANET data (View from the top) . 79

Figure 3.37. 3D plot of the Flow from SUMO data 80

Figure 3.38. 3D plot of the Flow from SUMO data (View from the top) 80

Figure 3.39. 2D plots of the flow for segments 1 to 7 (starting at top left position). The blue curve is METANET's data and the orange curve is SUMO's data. 81

Figure 3.40. 3D plot of the density from METANET data 84

Figure 3.41. 3D plot of the density from METANET data (View from the top) 84

Figure 3.42. 3D plot of the density from SUMO data 85

Figure 3.43. 3D plot of the density from SUMO data (View from the top) . . . 85

Figure 3.44. 2D plots of the density for segments 1 to 7 (starting at top left position). The blue curve is METANET's data and the orange curve is SUMO's data. 86

Figure 3.45. 3D plot of the Velocity from METANET data 87

Figure 3.46. 3D plot of the Velocity from METANET data (View from the top) 87

Figure 3.47. 3D plot of the Velocity from SUMO data 88

Figure 3.48. 3D plot of the Velocity from SUMO data (View from the top) . . 88

Figure 3.49. 2D plots of the velocity for segments 1 to 7 (starting at top left position). The blue curve is METANET's data and the orange curve is SUMO's data. 89

Figure 3.50. 3D plot of the Flow from METANET data 90

Figure 3.51. 3D plot of the Flow from METANET data (View from the top) . 90

Figure 3.52. 3D plot of the Flow from SUMO data 91

Figure 3.53. 3D plot of the Flow from SUMO data (View from the top) 91

Figure 3.54. 2D plots of the flow for segments 1 to 7 (starting at top left position). The blue curve is METANET's data and the orange curve is SUMO's data. 92

Figure B.1. Representation of the subset $[\alpha, \rho_{cr}]$ of the parameters when calibrating from a starting point x_0 equal to the mean of the boundaries. 107

Figure C.1. Convergence of the parameters $[\rho_{cr}, \rho_{max}, \nu, \kappa]$ of Segment 2 over 400 Iterations. 109

Figure C.2. Convergence of the parameters $[\rho_{cr}, \rho_{max}, \nu, \kappa]$ of Segment 3 over 400 Iterations. 110

Figure C.3. Convergence of the parameters $[\rho_{cr}, \rho_{max}, \nu, \kappa]$ of Segment 4 over 400 iterations. 110

Figure C.4. Convergence of the parameters $[\rho_{cr}, \rho_{max}, \nu, \kappa]$ of Segment 5 over 400 Iterations. 111

Figure C.5. Convergence of the parameters $[\rho_{cr}, \rho_{max}, \nu, \kappa]$ of Segment 7 over 400 Iterations. 111

LIST OF TABLES

Table 1.1. Traffic model classifications. 5

Table 2.1. Parameters of the model and their units. 23

Table 2.2. Variables' weights combinations for the objective functions. 32

Table 2.3. Initial point and precision each processed parameter's values. . . . 37

Table 3.1. Lengths of the segments of the road stretch. 44

Table 3.2. Textual output of the *patternsearch* function. 55

Table 3.3. Best obtained parameters for the one-lane closed scenario with a flow of 4500 veh/h after the per segment calibration. 57

Table 3.4. Comparison of the objective function values obtained in the calibration and validation phases for the one-lane closed scenario with a flow of 4500 veh/h. 58

Table 3.5. Best obtained parameters for the 1-lane closed scenario after the per segment calibration. 72

Table 3.6. Comparison of the objective function values obtained in the calibration and validation phases for the 1-lane closed scenario. 72

Table 3.7. Best obtained parameters for the free flow scenario after the per segment calibration. 83

Table 3.8. Comparison of the objective function values obtained in the calibration and validation phases for the free flow scenario. 83

LIST OF SYMBOLS

k	Current Number of Iteration
K	Total Number of Iterations $(-)$
L_i	Length of the ith Segment (km)
N	Total Number of Segments $(-)$
$q_i(k)$	Flow on the ith Segment at time k (veh/h)
T	Time step $(hours)$
$v_i(k)$	Velocity on the ith Segment at time k (km/h)
$v_0(k)$	Velocity of the incoming flow of the road stretch (km/h)
v_{free}	Free flow speed (km/h)
v_{max}	Maximum Velocity (km/h)
v_{min}	Minimum Velocity (km/h)
$V_{eq}(\rho_i(k))$	Driver's desired speed (km/h)
α	Equation Parameter $(-)$
Δ_λ	Number of lanes dropped $(-)$
κ	Equation parameter $(veh/km/lane)$
λ_i	Number of lanes on the segment i $(-)$
ν	Equation Parameter (km^2/h)
$\rho_0(k)$	Density of the road stretch's incoming flow $(veh/km/lane)$
ρ_{cr}	Critical Density $(veh/km/lane)$
$\rho_i(k)$	Density on the segment i at time k $(veh/km/lane)$
$\rho_{N+1}(k)$	Density of the road stretch's outcoming flow $(veh/km/lane)$
ρ_{max}	Density of the road stretch's incoming flow$(veh/km/lane)$
τ	Equation Parameter $(hours)$
ϕ	Equation Parameter $(-)$

LIST OF ACRONYMS/ABBREVIATIONS

AUS	Akıllı Ulaşım Sistemi
CTM	Cell Transmission Model
GPS	Generalized Pattern Search
ID	IDentification
IDM	Intelligent Driver Model
ITS	Intelligent Transport System
LWR	Lighthill Whitham Richards
METANET	Modele d'Ecoulement du Traffic Autoroutier: NETwork
MIDAS	Motorway Incident Detection and Automatic Signalling
PDE	Partial Differential Equation
RMSE	Root Mean Square Error
SUMO	Simulation of Urban MObility
TraCI	Traffic Control Interface
VSL	Variable Speed Limit

1. INTRODUCTION

1.1. Motivation

Traffic congestion is an important factor in today's urban life which has a direct influence on people's quality of life. This phenomenon usually happens on peak times when people are commuting to work. As people are stuck in the traffic congestion, not only it will take more time to reach their destination, but also this wasted time can be of their work's time meaning a loss of money. In addition to that, congestion may cause pollution. Since more vehicles are stuck in traffic and cannot advance properly, higher amount of gas is ejected to the air. Thus, tackling congestion problem is critical. There are several factors that lead to traffic congestion including the times when roads are full of vehicles, or when there are constructions, maintenance or incidents on roads. These factors cause a disruption in traffic flow. It is important to note that resolving congestion problem cannot only be done by increasing capacity and by constructing new roads and motorways; It also requires developing a transportation management technology.

Although the solution of building new roads may seem interesting at first sight, it is not practical as it may cause even more problems: The congested roads can be located in areas leaving no place for new constructions. In addition, building new roads has an important cost which might refrain many cities to do so. New roads would also bring more pollution in long run, considering the building phase and the vegetation removed to let place for the new constructions. Another issue with constructing new roads is the new demand of drivers that would drive on them. By increasing the supply, the demand will also increase, leading traffic congestion on the new roads too after a short time.

Taking all of these into account, utilizing the existing roads and enhancing the traffic management technology would be a better solution. To do so, an infrastructure has to be put in place. This infrastructure can be based on sensors that retrieve the data, computers that process what actions to be taken from the sensors' output, and some equipment to apply the required action. This procedure is a sub-category of Intelligent Transport System, commonly shortened as ITS. One ITS control method currently in use in most of the roads is Variable Speed Limit (VSL). VSL would limit the speed of certain segments of the road to fluidify the congestion. At the same level as VSL, some display screens as gantries can inform the drivers that a congestion is happened further downstream of the road or inform them about the limited speed. This way, they can choose to change their route if possible or adjust their speed. Another famous ITS control system is the ramp-metering that could be in a form of a traffic light. This system regulates the number of vehicles that enter the road stretch, in order to diminish any possible accidents or congestions. Therefore, having a calibrated traffic flow model that reproduces the behavior of drivers on the real road can be of benefit.

Multiple types of traffic flow models exist, the main ones are called first-order and second-order models. First-order models are simple and easier to apply, however, their breakdowns such as not being able to capture traffic instability, capacity drop, and instantaneous speed adaptation lead researchers to propose higher order models. As the model should give a precise vision of the network's state in a short time, all of this under a limited cost. If the model is accurate, the actions taken will have a better impact on the congestion and will solve it faster or even prevent it from happening. Thus, the choice of the model is crucial. For this thesis, which is a calibration study, a second-order macroscopic traffic flow model has been chosen in order to better capture the dynamics of traffic.

1.2. Aim and Objectives

A calibrated and simulated microscopic traffic flow model is used in this thesis to calibrate a second-order macroscopic traffic flow model to represent the traffic flow on the motorway in Sweden by validation. For this purpose, different scenarios which represent free flow, intermediate flow and congestion have been studied. This model can be used to better manage the traffic and implement control strategies on motorways.

In all the previous research, the applied models were considering the network as a whole or several links, a link contains one or more segments depending on their geometric characteristics, thus, the applied parameters were the same for all segments in a link/ network. Since each segment has its own characteristics, not only due to its type and geometry, but also due to its distance from an incident, especially when congestion happens, as a contribution in this project, a model that applies the parameters and calibration per segment is tested. Therefore, instead of having a global model parameters, each segment has its own parameters. Thus, instead of calibrating 9 parameters for the 7 studied segments, 63 parameters have been calibrated. This will lead to have a better vision of the real situation in different mentioned scenarios alongside with lane drops, changes in geometry, speed limits and congestions.

Consequently, the primary objective of this thesis is to calibrate a second-order macroscopic model with microscopic model; the microscopic model used in this thesis is a modified version of a model from SUMO, which was conducted and calibrated by the real-time data in a study named: Modeling and Analysis of PID-Controlled Heavy-Duty Vehicle Platoons in Real Traffic in 2016. However, some secondary objectives are coming along with the primary one such as finding a method to obtain the best match between the two models' behaviour, and finding optimal parameter values for different scenarios of the studied second-order macroscopic traffic flow model to reflect the behaviour of the drivers and traffic dynamics on a specific motorway in Sweden.

1.3. Scope

A second-order traffic flow model is modeled and calibrated for part of the E4 highway, thus, the model is tested, calibrated and validated for the mentioned roadway with traffic regulations of Sweden.

1.4. History of Traffic Flow Theory and Models

Traffic flow theory and modeling started in the 1930s, pioneered by the American professor Bruce D. Greenshields.

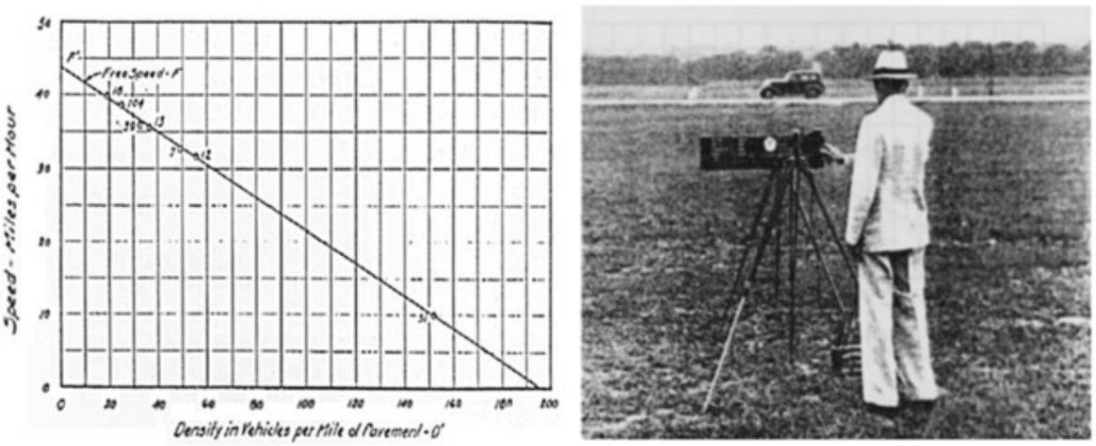

Figure 1.1. Greenshields using his camera to measure the traffic flow.

By using a 16 mm Simplex movie camera, he measured traffic flow, density and speed; he claimed that the relationship between density and speed is linear which made a parabolic relationship between traffic flow and density. (Greenshields, 1935) He also did some investigation at intersections for obtaining the traffic performance. (Greenshields 1947)

Greenshields mostly talked about the linear relationships between fundamental variables of traffic flow namely speed and density. (A Study of Traffic Capacity, 1935) However, in upcoming years, investigations had been made to put dynamics into ac-

count for traffic models. Specially in the 1950s, with the significant use of vehicles and the development of highway systems, numerous traffic flow theories were studied and developed. Some of which were based on different approaches such as car-following, queue and traffic wave theory. Seminal studies of that period include the works by Reuschel (1950a; 1950b; 1950c), Wardrop (1952), Pipes(1953), Lighthill and Whitham (1955), Newell (1955), Richards (1956), Webster (1957), Edie and Foote (1958), Chandler et al. (1958) and other papers by Herman et al. (see Herman 1992). However, only after the 1990s, the field has gained considerable attraction as overall traffic demand has increased and, easier access to the traffic data and improved computational power were available. (Traffic Flow Dynamics,2013)

Traffic flow models have been developed and used since the beginning of the twentieth century. With the use of mathematical models, researchers and engineers develop a traffic flow model whose characteristics are representative for a real-world system. (Traffic Flow Modelling, 2019)
Traffic flow models can be categorized by their attributes. As discussed by Hoogendoorn & Bovy in 2000, followings are the examples of the traffic model classifications.

Table 1.1. Traffic model classifications.

Level of Detail	Sub-Microscopic Traffic Flow Model Microscopic Traffic Flow Model Mesoscopic Traffic Flow Model Macroscopic Traffic Flow Model
Extend of Independent Variables	Continuous Semi-Discrete Discrete
Depiction of the Operations	Deterministic Stochastic
Scale of Application	Networks - Stretches Links Segments Intersections

In the following, two models of microscopic and macroscopic traffic flow will be discussed. For further information about other models please refer to (Hoogendoorn & Bovy, 2000).

1.5. Background and Concepts

1.5.1. Microscopic Models

A microscopic model of traffic flow attempts to analyze the flow of traffic by modelling driver-driver and driver-road interactions within a traffic stream which respectively analyses the interaction between a driver and another driver on road and of a single driver on the different features of a road. (Mathew 2014). Moreover, It describes both temporal and spatial performance of the stream in a very high level of detail. Car-following and lane-changing models belong to this category. The first car-following model was introduced by Pipes in 1953 (An Operational Analysis of Traffic Dynamics, 1953) which included the dynamics behaviour of the traffic and drivers. This model was based on the safe-distance between vehicles. After a while, self-distance models were improved and revised by Gipps in 1981. He introduced two mechanisms. In one mechanism, limitation of speed is due to either a speed limit implementation or speed limit imposed by the vehicle. The other mechanism shows that the speed is reduced regarding to keep the safe-distance between the leading and following vehicle. Later, in 1997, Krauss represented a car-following model based on safe-speed, it is worth to note that this model is used in this project for SUMO simulation. (Krauß 1998)

In microscopic flow characteristics, the key parameters are the speed of individual vehicles, headway, and spacing while the three principal macroscopic characteristics that illustrate a traffic stream are volume or flow rate, speed, and density. The relationship between these parameters and the bridge between microscopic and macroscopic characteristics of flow help traffic engineers to plan, design, and evaluate the effectiveness of implements that were and are going to be applied in a traffic stream.

1.5.2. Macroscopic Models

These models describe traffic at a high level of aggregation as a flow without distinguishing its constituent parts, for instance, the traffic stream is represented in an aggregate manner using characteristics as flow-rate, density, and speed.(Hoogendoorn & Bovy, 2000) The three main characteristics of macroscopic models as discussed above are flow, speed and density.

1.5.2.1. Flow. Flow represents the number of the vehicles passing a certain point in a period of time usually per hour.

Capacity is the maximum traffic flow that can be accommodated in a highway facility during a given time period under prevailing roadway, traffic and control conditions, depending on the design of the roadway, geometry, number of lanes, speed limitations and other parameters. Capacity of highways depending on their geometry and characteristics usually has values between 1800 (veh/hour/lane) to 2300 (veh/hour/lane). (See Trafikverket,2013)

1.5.2.2. Speed. There are three types of mean speeds, the time-mean speed is the arithmetic average speed of all vehicles for a specified period of time.

While the space-mean speed is the average speed of vehicles traveling a given segment of a roadway during a specified period of time and is calculated using the average travel time and length for the roadway segment.

However, Wardrop (1952) showed that in mathematical terms, the space-mean speed can be equivalent to harmonic mean speed. (See Eq 1.1)

$$V_s(x,t) = \frac{n}{\sum_{i=1}^{n} \frac{1}{v_i}} \tag{1.1}$$

Where:

v_i : Speed of the ith vehicle (kph)

n : Number of observations

(x, t) : Position and time of the vehicle for which the speed is calculated

As Van Lindt (2004) indicated, there is a relationship between time mean speed and space mean speed as follows:

$$V_t = \frac{\sigma_s^2}{V_s} + V_s \tag{1.2}$$

Where:

σ_s : Variance of space-mean speed

$$V_s = V_t \tag{1.3}$$

As it can be seen from the Equation 1.3, only when all individuals have the same speed those two speeds are equal.

The last one is average running speeds which can be calculated by using the average running time, which does not include any stopped delay time and can be considered as space mean speed if there is no stopped delay.

1.5.2.3. Density. Density is described as number of vehicles occupying a particular length of a roadway per lane.

It is difficult to measure the density since it requires observation of the complete road stretch at a certain time. That is to say that, density is often estimated using the continuity equation, discovered by Greenshields (1935). The continuity equation

indicates that by knowing the two macroscopic variables of speed, third one can be calculated. This formula is the foundation of the macroscopic traffic flow models.

$$\rho(x,t) = \frac{Q(x,t)}{V(x,t)} \tag{1.4}$$

Where:

$Q(x,t)$: Flow of segment x at time t

$V(x,t)$: Velocity of segment x at time t

1.5.2.4. Fundamental Diagram. The Fundamental Diagram shows the relationship between the three macroscopic variables. Since fundamental diagrams provide important information, one can easily understand the state of the traffic by looking at them. Critical density, capacity, capacity drop, jam density, free flow speed and other information can be derived from a fundamental diagram. Hence, it has a notable importance in traffic studies. As described before, capacity occurs when flow is at its maximum rate, the corresponding density for the maximum flow, capacity, is known as critical density. (See Figure 1.2)For densities inferior to the critical density, a free flow stream has been experienced by the road, thereby, traffic is stable, and the speed is usually equal to the free flow speed. Around the critical density, traffic is marginally stable, that is to say that, small perturbations such as a sudden deceleration can be dissolved without causing congestion or issues. Above the critical density point, congestion happens and traffic is unstable and congested: a perturbation, small or big, will always lead to congestion.

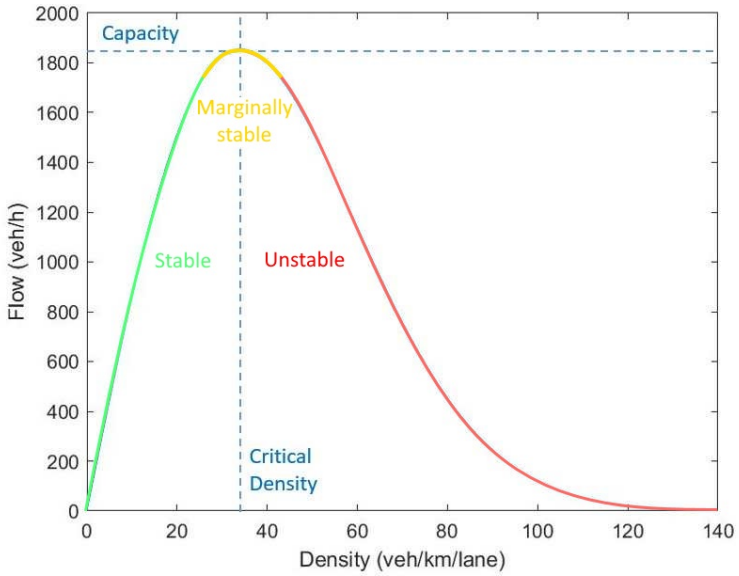

Figure 1.2. Fundamental diagram of flow - density.

Macroscopic models can be found in different number of orders, and the main difference lies in the form of modeling the speed and flow. The earliest first-order model was introduced independently by Lighthill and Whitham in 1955 and Richards in 1956, known as LWR model which is a continuous macroscopic model.

1.5.2.5. First-Order Macroscopic Models. As discussed before, the foundation of every traffic flow model is the continuity equation (See Eq. 1.4). The main difference between the models and orders lies in the form of modeling the speed and flow. In the 1950s, a first-order macroscopic flow model named as Lighthill-Whitham-Richards (LWR) was proposed. (Lighthill & Whitham 1955; Richards 1956). The base of this model was the conservation of vehicles. LWR model was introduced as a partial differential equation known as LWR PDE. It states how the density, in regard to the road conditions and initial boundaries, changes over time and space. In the following formula, the general form of the LWR model is shown.

$$\frac{\partial \rho}{\partial t} + \frac{\mathrm{d}Q_e(\rho)}{\mathrm{d}\rho} \cdot \frac{\partial \rho}{\partial x} = -\frac{Q}{M} \cdot \frac{\mathrm{d}M}{\mathrm{d}x} + q_r(x,t) \tag{1.5}$$

The first term in the right side refers to the lane-changing phenomena where M is the number of the lanes. The second term alludes to on or off-ramps where $q_r(x,t) = Q_{rmp}/M.L_r$ denotes as effective source density. Right side of the Equation 1.5 is equal to zero when the studied road stretch is homogeneous and devoid of any on or off-ramps and lane-changing. (Equation 1.6)

$$\begin{aligned} &\frac{\partial x\rho(x,t)}{\partial t} + \frac{\partial Q(\rho(x,t))}{\partial x} = 0, \qquad (x,t) \in (0,L)\text{x}(0,T), \\ &\rho(x,0) = \rho_0(x), \qquad \rho(0,t) = \rho_l(x), \qquad \rho(L,t) = \rho_r(x) \end{aligned} \tag{1.6}$$

During several years, different models based on this theory but with different fundamental diagrams and mathematical representations, were proposed. These models are also in the first-order models category since they only have one dynamic equation and known as LWR models.

The simplest first-order model applies a triangular fundamental diagram. (See Equation 1.7). One of the most famous first-order model with a triangular fundamental diagram is the Cell Transmission Model (CTM) proposed by Daganzo in 1995. CTM model can be used both as a continuous and a discrete model. The continuous version of the model (Equation 1.6)) is demanding to solve analytically, explicitly if the studied network is large and contains several segments. For larger networks and real-time applications discrete models are used, where both space and time are split. CTM is the numerical scheme to solve the discrete version of the LWR model where space is divided into cells.

$$Q_e(\rho) = \begin{cases} V_0\rho & \text{if } \rho \leq \rho_{cr} \\ \frac{1}{T}(1 - \frac{\rho}{\rho_m}) & if\, \rho_{cr} < \rho \leq \rho_m \end{cases} \tag{1.7}$$

$$\rho_{cr} = \frac{1}{V_0 \cdot T + \frac{1}{\rho_m}} \tag{1.8}$$

Where V_0 is the free flow speed, ρ is the density of the segment, ρ_{cr} is the critical density, ρ_m is the maximum density and T is the desired time headway. This formula gives a triangular fundamental diagram.

For applying the CTM model, specific steps should be followed.

- Determining the Supply and Demand at the Cell Boundaries:

$$S_i(k) = \begin{cases} Q_i^a(k) & \rho_i(k) > \rho_{cr} \\ C_i & \text{otherwise} \end{cases} \tag{1.9}$$

$$D_i(k) = \begin{cases} Q_i^a(k) & \rho_i(k) \leq \rho_{cr} \\ C_i & \text{otherwise} \end{cases} \tag{1.10}$$

  Where $Q_i^a(k)$ is the flow of the ith segment at time k, $\rho_i(k)$ is the density of the ith segment at time k, ρ_{cr} is the critical density and C_i is the ith segment's capacity.
- Determining the Flow through the Cell Boundaries

$$Q_i^0 = Q_{i-1}^1 = min(D_{i-1}, S_i) \tag{1.11}$$

$$Q_i^1 = Q_{i+1}^0 = min(D_i, S_{i+1}) \tag{1.12}$$

- Updating the Traffic State of the Cell

$$\rho_i^a(k+T) = \rho_i^a(k) + \frac{1}{L_i}\left(Q_i^0 - Q_i^1\right)T \tag{1.13}$$

$$Q_i^a(k+T) = \lambda_i \cdot Q_e\left(\frac{\rho_i^a(k+T)}{\lambda_i}\right) \tag{1.14}$$

Where k is the current time, T is the elapsed time, L_i is the length of the roadway, and λ_i is the number of lanes.

In first-order models, one noticeable feature is the model's consistency (under several assumptions and simplifications) with a class of car-following model.(Papageorgiou, 1998) However, due to some shortcomings and breakdowns of the first-order models, higher orders introduced shortly after. Further information about the characteristics of the higher-order models are covered in the following and methodology part. Howbeit, for additional and supplementary information please refer to (Lighthill & Whitham 1955; Richards 1956 ; Daganzo 1994 ; Daganzo 1995 ; Payne 1971 ; Papageorgiou et al. 1980).

1.5.3. Higher-Order Macroscopic Models

Although first-order models are simple and easier to apply, their breakdowns led researchers to propose higher-order models. Some of the shortcomings of the first-order models are as following:

- They cannot capture traffic instability, capacity drop or etc.
- Speed adaptation in most of the LWR models are instantaneous which leads to infinite deceleration or acceleration which is unrealistic.
- Local speed V(x,k) does not show any independent dynamics as it is coupled to

speed-density relation.

Therefore, the key idea was to add an equation to represent the velocity dynamics in order to have an accurate model to represent the traffic phenomena. (See Equation 1.15)

$$\frac{\mathrm{d}V}{\mathrm{d}k} = \left(\frac{\partial}{\partial k} + V \cdot \frac{\partial}{\partial x}\right) \cdot V = \mathrm{F}(\rho, V, \frac{\partial \rho}{\partial x}, \frac{\partial V}{\partial x}) \tag{1.15}$$

Where V is the velocity of the vehicles on the roadway and ρ is the vehicles density on the roadway.

In 1971, Payne proposed a model which consists of the dynamics for both speed and density. (Payne, 1971)

$$\frac{\partial V}{\partial k} + V \cdot \frac{\partial V}{\partial x} = \frac{V_e(\rho) - V}{\tau} + \frac{V_e'(\rho)}{2\rho\tau} \cdot \frac{\partial \rho}{\partial x} \tag{1.16}$$

Where V is the velocity of the vehicles on the roadway and ρ is the vehicles density on the roadway. A constant speed relaxation time τ is introduced for speed adaptation. Moreover, a traffic pressure term is stated to represent driver-driver interactions, which is proportional to the difference of the desired speed and the current steady state speed. (See Equation 1.17)

$$-\frac{1}{\rho} \cdot \frac{\partial P}{\partial x} = -\frac{1}{\rho} \cdot \frac{\partial}{\partial x} \cdot \left(\frac{V - V_e(\rho)}{2\tau}\right) = -\frac{V_e'(\rho)}{2\rho\tau} \cdot \frac{\partial \rho}{\partial x} \tag{1.17}$$

Payne indicated that the average speed in a segment of a roadway is affected by

three major terms (Payne 1971)

- **Relaxation Term:** Which describes the propensity of traffic flow to relax in order to obtain a desired speed or homogeneous steady state in the flow.
- **Convection Term:** Which describes that vehicles in the upstream segment will adapt their speed to the following segment's speed moderately and not instantaneously.
- **Anticipation Term:** Which describes the drivers' prediction of the traffic state in downstream and act accordingly.

Later, this model is formulated in a discrete spatial and temporal way by the METANET model developed by Papageorgiou et al. in 1989. (Papageorgiou et al. 1989 and 1990)

1.5.3.1. The METANET Model. The Modele d'Ecoulement du Trafic Autoroutier :Network, known as METANET is a second-order macroscopic traffic model developed by Messmer and Papageorgiou in 1989 (Papageorgiou et al. 1989 and 1990). This model is a modification of the Payne's model which was presented in 1971.

The modification was done by adding further extensions to the Payne's model to take the merging and weaving phenomena caused by on and off-ramps and lane drops into account. (Papageorgiou et al. 1989 and 1990)

Due to its accuracy and low computation time alongside the ability of predicting traffic jams and break-downs with high precision in size and location, METANET has been able to take the attention of the researches shortly after being introduced.

As in most of the macroscopic models, the network is divided into segments, and aggregated variables namely flow, speed and density are calculated which help a lot with the computation time since, only three values of a segment are calculated. With the help of these variables, behaviour of the traffic stream and a clear image of

the traffic state can be obtained. Kotsialos et al. in 2002 stated that macroscopic models are more suitable for designing a control system given their analytical nature. Furthermore, it is advantageous that the number of equations does not change if there is high or low density as it is the case with microscopic models. (A.D. van der Horst, 2011) More details about the model will be discussed in the following chapter.

1.5.3.2. METANET Model's Applications. Macroscopic models have been applied in several projects. Most of them are used for applying the models to do predictions about the future state of the traffic in a network.

In 1997, A. Kotsialos et al. have applied a modified version of METANET near the city of Amsterdam. The network is composed of three different roads, one orbital motorway around the city and two motorways connecting the north and south parts of the city. Another type of a link called store-and-forward was added to the METANET model. This link is used at the junctions of two or more roads. This link will keep the vehicles in a queue during a certain time before redirecting them to a downstream node. The goal of this METANET application was to model peri-urban network in METANET in a simple way, and the creation of a new link helped them to model the motorway network around Amsterdam with ease.

In 2000, W.J.J.Knibbe and the Dutch road authority has tested an on-line traffic forecasting system on a 20 kilometers long highway. The goal was to coordinate the traffic control measures and give the operators some decision support. They used METANET with real-time loop detector data to create forecast simulations up to one hour in the future. The obtained simulations corresponded to what would happen on the roads, and it gave them a powerful tool to take actions in advance. The issue they observed was that the model could not model incident that would happen outside of the studied network, for instance a bottleneck happening a few kilometers downstream of the studied segments.

In 2004, a study by M. Van Den Berg, applied the METANET model to a ramp-metering predictive traffic control framework. The goal of this study was to integrate the spontaneous re-routing actions of the drivers into a predictive model framework. The re-routing effect can lead to a consequent improvement of the network performance and is usually not present in the predictive framework. The model has been tested with simulated data, and the results showed an improvement of the results as drivers would have their total time spent reduced when taking the re-routing into account.

In 2017, a framework has been designed by A. Sayegh to predict the speed-based emissions. It uses two different models, the Cell Transmission Model which is a first-order model and METANET. Historical data from the MIDAS network have been used with some real-time data. The framework has been tested on three different highways in England that were between 3 to 5 kilometers long. The results were satisfying as the predictions were correct over 80% of the time.

1.5.4. Calibration of Second-Order Traffic Flow Models

The calibration part is an important process as it searches for the best value of the parameters of the model. This process varies in processing time, which could result in an on-line (real-time) or off-line (run before generating the model data) calibration. The calibration process also depends on the type pf the objective function: Each objective function has its own characteristics, some can be discontinuous/ discrete or continuous, some can be linear where others can have degree of freedoms and be nonlinear, therefore, these characteristics should be taken into account in a calibration study while choosing the objective function. If the calibration has not been processed by a proper objective function, the obtained results would be incorrect. Alongside with the calibration, the validation process is needed in order to verify whether the calibration is acceptable and our model can be validated.

In 2011, A.D. van der Horst applies a calibration method for the METANET and IDM models using a normalized quadratic measure. This calibration method applies a visual inspection combined with a literature comparison for the validation part; another calibration method called Theil's inequality is applied to obtain the best accuracy.

In 2008, Lamon has applied different methods for calibration using the functions `fmincon` and `patternsearch` packed in the MATLAB software. The local minimums issue was pointed out from the `fmincon` function and showed that `patternsearch` is a better candidate to obtain the global minimum.

In 2014, Spiliopoulou et al. compared different optimization methods for the calibration part. The Nelder-Mead algorithm (Nelder and Mead, 1965; Lagarias et al., 1998) is compared to a stochastic genetic algorithm (Goldberg, 1989; Holland, 1992) and a cross-entropy method (Rubinstein and Kroese, 2004; de Boer et al. 2005). All these three methods have different pros and cons which make them more appropriate to use in different situations. All of them do not need derivative information, which makes them suitable for problems with discontinuous, nonlinear or non-differentiable objective functions. All algorithms were applied to calibrate the METANET model. Every algorithm gave robust calibration of the model, but the Nelder-Mead algorithm was the one which took the shortest time to give a result, with a time of 0.5 minutes compared to the 122 and 197 minutes required by the genetic algorithm and cross-entropy method respectively.

The Nelder-Mead algorithm has been tested in this project thesis, but the obtained results were not as stated in Spiliopoulou et al.'s research. The robustness was correct, but the computation time was slightly greater than the `patternsearch` function used in MATLAB. This last function gave also robust results in a shorter time, and for this reason, it is the one that has been applied in this thesis.

In 2015, Zhong et al. proposed an automatic calibration model to calibrate a traffic model under the constraint of dysfunctional and aging sensors. The result shows an effective algorithm that can adapt to the current traffic conditions, even if they were unexpected ones.

In 2016, Konstantinos Katerinis studied the calibration of a first and second order models in real-time. This calibration was based on the software CALISTO (Spiliopoulou et al., 2014) which combines the calibration and validation processes of macroscopic traffic flow models.

READ IT**

1.5.5. Validation of Second-Order Traffic Flow Models

The model has to pass through a validation phase. This validation phase ensures that the model is giving the same accuracy for the results with different data sets. If a set of data gives excellent results, the model might be effective only for this set and might give worse results on all other data sets due to over-fitting. Once a model is validated through the validation phase, it means it can be generalized to all data sets without the fear of obtaining false results.

In 1990, Papageorgiou et al. showed a validation method for traffic flow models. The validation consists of applying the identified set of parameters to other data sets. The objective function value is evaluated for each data set with the initial parameters and compared to the objective function value calculated with the original data set. Then, a new calibration is launched for each data set to observe any possible improvement with a new parameters set. The study showed that the improvement is inferior to 20% when using a new parameters set compared to the original one. The graphical curves' shapes of the density and velocity variables were compared with each other and shown that they almost perfectly match. Considering all these facts, Papageorgiou et al. considered that the model could be validated.

In 2008, Lamon validated a METANET model calibrated with the help of the MATLAB functions `patternsearch` on one side and `fmincon` on the other one, using generated data with the Paramics software. Both models were validated using the same method: The same parameter sets were applied on another data set generated with Paramics, and the errors for density, velocity and flow were calculated. The errors were high for the density and flow, due to a segment that is too short for METANET to give good results, and this segment also influenced the next segments' error. Considering only the first segments, the errors were decreased to acceptable results. Then, a visual comparison of the results is done between the calibrated data set and the other generated Paramics' data set. This comparison showed that the model's data was close to the data from Paramics. Hence, Lamon considered the model is validated.

In 2015, Ramon validated a METANEL model using data from a freeway in Spain. The model was calibrated using collected data by detectors during 3 days, and then was validated with collected data with the same detectors during 7 days. To validate the model, a visual comparison of the mean speed and the flow is done, comparing the generated data from METANET and the ones from the detectors. This visual comparison showed that both data were almost exact during all the data gathering period. In addition to this, a measurement of the error is done, displaying an error of 13% for the mean speed and 5% for the flow. Therefore, the model was validated.

In 2015, Seybold calibrated fundamental diagram for macroscopic traffic models. In his research, Seybold aimed to validate the models using visual analysis and a comparison of the travel times with the obtained data in real life. Multiple scenarios were tested and calibrated before going through the validation process. While validating, the best error he obtained was 19%, and his accepted range for the validation was errors below 30%.

1.6. Thesis Outline

The remainder of this report is organised as follows:

- **Chapter 2** — provides a broad view of the studied traffic flow model and its formulations. Moreover, it covers calibration algorithm and its procedure. The model's characteristics, boundaries, and assumptions that are implemented in the studied project are also clarified in this section. Afterwards, validation method will be discussed.
- **Chapter 3** — presents the numerical experiments of the study and output results of the each studied scenario and their consequences and findings and analyzes them.
- **Chapter 4** — concludes the work by deducing the findings and remarks and possible future work in this field.

2. Methodology

In this chapter, a general view of the methods used in the thesis has been discussed. As mentioned before, the goal is to calibrate a second-order macroscopic traffic flow model, METANE, by a simulated and detailed microscopic traffic flow model from SUMO. First a detailed explanation of the METANET model is going to be covered. Secondly, the calibration phase will be discussed in the general term, as well as the validation phase. However, in the next chapter, all these sections will be further covered by the project's characteristics and assigned values.

2.1. METANET

As discussed in the literature review, for the second-order macroscopic traffic flow model, METANET model is used in this thesis project due to its characteristics. The model represents the dynamics of a traffic flow using the three main macroscopic variables named velocity, density and flow. To process these variables, the model divides the time and space in intervals. Time is subdivided into K intervals of T seconds, each interval being identified by the discrete time index $k = 1, 2, ..., K$. The road stretch is divided into N segments of length $L_i, i = 1, 2, ..., N$.

At each time step for each segment, the model calculates two states described as following:

$$\rho_i(k+1) = \rho_i(k) + \frac{T}{L_i} \cdot (q_{i-1}(k) - q_i(k)) \tag{2.1}$$

Where $q_i(k)$ is the flow of the ith segment at the time k.

This equation indicates that in each time step density is updated regarding the number of vehicles entering and exiting the segment. Moreover, it is multiplied by the time step and divided by both length of the segment and number of lanes in each

segment. Therefore, its unit is vehicle per kilometer per lane ($veh/km/lane$).

$$\begin{aligned} v_i(k+1) = v_i(k) &+ \frac{T}{\tau} \cdot (V_{eq}(\rho_i(k)) - v_i(k)) \\ &+ \frac{T}{L_i} \cdot v_i(k) \cdot (v_{i-1}(k) - v_i(k)) \\ &- \frac{\nu \cdot T}{\tau \cdot L_i} \cdot \frac{\rho_{i+1}(k) - \rho_i(k)}{\rho_i(k) + \kappa} \\ &- \frac{\delta T}{L_i \cdot \lambda_i} \cdot \frac{q_u(k) \cdot v_i(k)}{\rho_{i+1}(k) + \kappa} \\ &- \frac{\phi \cdot T \cdot \Delta_\lambda \cdot \rho_i(k) \cdot v_i(k)^2}{L_i \cdot \lambda_i \cdot \rho_{cr}} \end{aligned} \tag{2.2}$$

where τ ($hour$), ν ($kilometer^2/h$), κ ($vehicle/kilometer$) are parameters of the model which have to be determined during the calibration phase. V_{eq} is the equilibrium function, $q_i(k)$ is the flow of the ith segment at the time k, $v_i(k)$ is the vehicles' mean speed of the ith segment at the time k, T is the timestep duration, L_i is the length of the ith segment, λ_i is the number of lanes of the ith segment and Δ_λ is the number of dropped lanes occurring between the segment i and $i+1$.

Table 2.1. Parameters of the model and their units.

Parameter	**Unit**
ρ_{cr}	veh/km
ρ_{max}	veh/km
v_{min}	km/h
v_{max}	km/h
α	-
τ	$hours$
ν	km^2/h
κ	$veh/km/lane$
ϕ	-

As it can be seen, this equation has six terms. The first term is the speed of previous iteration which will be updated into the current speed value by adding and subtracting other terms. The second term is known as relaxation term which describes the propensity of traffic flow to relax to have a desired speed or homogeneous steady state in the flow if possible. However, the desired speed depends on the segment's density and speed limit of the segment if there is any. The third term is named as convection term. This term indicates that vehicles in the upstream segment will gradually adapt their speed to the very next segment's speed. The fourth term is named anticipation term, this term explains the adjustment of the speed due to the downstream's traffic state. If the downstream state is congested, vehicles will decelerate, else they will accelerate to adjust the speed regarding the segment's traffic state. The fifth term indicates the changes in regard to presence of any on or off-ramps which in this study this term is equal to zero since no on or off-ramps are involved.

Last term is added to the equation in order to represent the speed drop due to the reduction of lanes from one segment to another. ϕ is one of the parameters which should be defined in the calibration phase.

$$V_{eq}(\rho_i(k)) = v_{free} \cdot exp\left[\frac{-1}{\alpha} \cdot \left(\frac{\rho_i(k)}{\rho_{cr}}\right)^{\alpha}\right] \tag{2.3}$$

Where v_{free} is the free flow speed, ρ_{cr} is the critical density, $\rho_i(k)$ is the density of the ith segment at time k, α is a parameter of the model.

Equilibrium Velocity, is the speed each driver on the segment desires to have. This value depends directly on the current density of the segment, $\rho_i(k)$. If the current density is under the critical density, the drivers are willing to drive at the free flow speed as there is no congestion. If the density is above the critical density, the driver will obtain a speed that fits the current flow of vehicles on the segment. The unit is in kilometer per hour. where $v_{free}$is the free flow speed ($kilometer/hour$), α is

a parameter of the model and ρ_{cr} is the critical density $(vehicle/kilometer/lane)$. These three parameters have to be determined during the calibration phase. Since, the network used has a speed limit set to 90 (km/h), free flow speed will not be defined as a parameter to calibrate in this thesis.

By having the speed and the density of each segment, flow can be calculated.

$$q_i(k) = \rho_i(k) \cdot v_i(k) \cdot \lambda_i \tag{2.4}$$

Where λ_i is number of the lanes in the given segment.

The variables are graphically represented in Figure 2.1, showing that each segment has its own values of density, velocity and flow, and that it uses the flow of the previous segment to calculate the formulas.

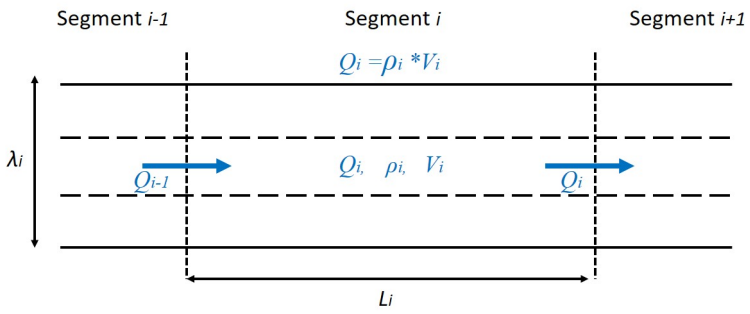

Figure 2.1. Representation of METANET's variables on a road segment.

Since METANET is a dicrete model, a time step should be set. A time step with a low value might include too much noise in the data set, and a high value may make the model less precise as information about a congestion period might be lost.

The length of the segments, L_i, and the time step, T, have to be carefully chosen in accordance of satisfying the condition in which during a time step, a vehicle driving at the free flow speed should not cross the length of the segment. (See Equation 2.5)

$$v_{free} \leq \frac{L_i}{T} \tag{2.5}$$

Each segment relies on data from its direct neighbors of the previous iteration, in order to be able to calculate the speed and density. In the density Equation 2.1 for a given segment, the flow of the previous segment has an impact on the density of this given segment. In the mean velocity equation, for a given segment, the speed of the previous segment is required for the convection term and the density of the next segment is needed to calculate the anticipation term. On the studied road stretch in this thesis, every segment can obtain these data as they were calculated for all segments at the previous time step, but the incoming flow for the first segment and the downstream of the last segment cannot be obtained by using the calculated values of the previous iteration as the needed segments are not in the road stretch. These values have to be provided externally, from real data or from simulated data. They are called the boundary conditions, which are q_0 for the flow entering the road stretch, ρ_0 and v_0 which are respectively the density and the mean speed of the vehicles entering the road stretch, and ρ_{N+1} for the density of the out-coming flow of the road stretch. The density of the out-coming density ρ_{N+1} is calculated with the density of the last segment of the road stretch ρ_N, by satisfying the following condition:

$$\rho_{N+1} = \begin{cases} \rho_N, & \text{if } \rho_N < \rho_{cr} \\ \rho_{cr}, & \text{otherwise} \end{cases} \tag{2.6}$$

This condition is applied to make sure that there is no congestion happening at the downstream of the road stretch. Other conditions are applied to the mean speed and

density for every segment: The mean speed cannot have a value under v_{min}, this value is determined during the calibration process. This condition is applied to obtain more realistic behaviour of the model when a congestion is ending (Papageorgiou et al., 1990). Density is also constrained by a maximum value ρ_{max} that limits the density to reasonable values when congestion appears. ρ_{max} is also determined during the calibration phase. The following constraints are applied for the mean speed and density on each segment :

$$\rho_i(k+1) = \begin{cases} \text{Equation 2.1}, & \text{if } \rho_i(k+1) < \rho_{max} \\ \rho_{max}, & \text{otherwise} \end{cases} \tag{2.7}$$

$$v_i(k+1) = \begin{cases} \text{Equation 2.2}, & \text{if } v_i(k+1) > v_{min} \\ v_{min}, & \text{otherwise} \end{cases} \tag{2.8}$$

In order to represent the speed drop due to the reduction of lanes from one segment to another, the following term has to be added to the Equation 2.2:

$$-\frac{\phi \cdot T \cdot \Delta_\lambda \cdot \rho_i(k) \cdot v_i(k)^2}{L \cdot \lambda_i \cdot \rho_{cr}}$$

where ϕ (phi, with no unit) is a parameter of the model and Δ_λ is the number of dropped lanes occurring between the segment i and $i+1$. Without this extra term, the speed would not correctly represent the loss of speed happening when there are lanes removed from a segment to another.

2.2. Calibration

The calibration phase is applied to obtain the values for the parameters of the model that match the sample data accurately. It is applied off-line, which means that the process is done before using the model in a real-time configuration. The off-line calibration has a long process time of several minutes which could not be fit for predicting and processing the data in real-time. The calibration process relies on an objective function, also called error function. This function represents the difference between the real-time data or a historical data from a network and the obtained data from a simulated model. The goal of the calibration phase is to apply different configuration values to the parameters of the model and to extract the one which gives the smallest value for the objective function.

$$\min_{x} f(x) \tag{2.9}$$

Different measures for the objective function exist. In this project, the Root Mean Square Error (RMSE) is applied due its robustness. This measure applies a variant of the quadratic mean. For each type of data, the square difference between the real time data and the model data is summed up, and the sums are then divided by the number of data available and its square root is calculated. Weights can be applied to each type of data or to some data in particular, and their differences will have a bigger impact on the result of the objective function.

The data used are the density and velocity coming from each segment of the network, noted as $\rho_i, v_i \quad i = 1...N$.

They are grouped in separate arrays called rho and vel as seen in Equations , each array contains all the values for all segments at each iteration. These two arrays

constitute the model's state which represents the state of the whole network. The boundary conditions are taken from the previous arrays for the density and velocity, using the values corresponding to the extra segments. Finally, all the parameters used in the model's formulas are placed in a parameter vector (Equation 2.13).

$$rho = \begin{pmatrix} \rho_0(0) & \rho_0(1) & ... & \rho_0(K-1) & \rho_0(K) \\ \rho_1(0) & \rho_1(1) & ... & \rho_1(K-1) & \rho_1(K) \\ ... & ... & ... & ... & ... \\ \rho_7(0) & \rho_7(1) & ... & \rho_7(K-1) & \rho_7(K) \\ \rho_8(0) & \rho_8(1) & ... & \rho_8(K-1) & \rho_8(K) \end{pmatrix} \tag{2.10}$$

$$vel = \begin{pmatrix} v_0(0) & v_0(1) & ... & v_0(K-1) & v_0(K) \\ v_1(0) & v_1(1) & ... & v_1(K-1) & v_1(K) \\ ... & ... & ... & ... & ... \\ v_7(0) & v_7(1) & ... & v_7(K-1) & v_7(K) \\ v_8(0) & v_8(1) & ... & v_8(K-1) & v_8(K) \end{pmatrix} \tag{2.11}$$

$$x_0 = [\rho_0(0) \quad v_0(0) \quad \rho_1(0) \quad v_1(0) \quad ... \quad \rho_7(0) \quad v_7(0) \quad \rho_8(0) \quad v_8(0)]^T \tag{2.12}$$

$$p = [\rho_{cr} \quad \rho_{max} \quad v_{min} \quad \alpha \quad \tau \quad \nu \quad \kappa \quad \phi \quad v_{max}]^T \tag{2.13}$$

To calculate the dynamic state of the model, these arrays are used. They use the values previously processed to obtain the state of the model at the next time step, using the model's formulas stated in Equations 2.1 and 2.2.

To identify the unknown parameters, the RMSE function is applied. The objective is, given an initial state of the network x_0 (see Eq. 2.12), to find the parameters p that minimizes the objective function J, shown in Equation 2.14.

$$J_{RMSE} = \sqrt{\frac{1}{K} \cdot \sum_{n=1}^{K} (d_{\text{MET}} - d_{\text{SUMO}})^2} \tag{2.14}$$

Where K is the number of iterations, d_{MET} is the METANET's data and d_{SUMO} is the SUMO's data.

For this objective function, there are two different set of data, velocity and the density. The RMSE function is applied, but each set of the data is divided by the mean of the corresponding real time data set (here simulated data from SUMO): The objective function should return the global error of the model and as the errors are added together, the velocity's error and the density's error cannot have an assigned unit. No weight has been applied to the data as velocity and density have the same importance in the model.

The RMSE always returns a positive value which can be equal to zero. The closer the result value to zero, the better the model matches the real segment data. With a result of 0, it means that the model matches with the data perfectly.

The objective function used in this thesis is shown in the Equation 2.15.

$$J(p) = \sqrt{\frac{1}{N \cdot K} \cdot \sum_{k=1}^{K} \sum_{i=1}^{N} \left(\left(\frac{v_{\mathrm{MET},i}(k) - v_{\mathrm{SUMO},i}(k)}{\bar{v}_{\mathrm{SUMO}}} \right)^2 + \left(\frac{\rho_{\mathrm{MET}}(k) - \rho_{\mathrm{SUMO},i}(k)}{\bar{\rho}_{\mathrm{SUMO}}} \right)^2 \right)} \tag{2.15}$$

where p, as mentioned before in the Equation 2.13, is the vector that contains all the model's parameters which have to be determined. N is the total number of segments in the road stretch, K is the number of iterations, $\bar{v}_{\mathrm{SUMO}}$ is the mean of the speeds from SUMO data set, $\bar{\rho}_{\mathrm{SUMO}}$ is the mean of the densities from the SUMO data set, $v_{\mathrm{MET},i}(k)$ and $\rho_{\mathrm{MET},i}(k)$ are the mean speed and density of the segment i at the iteration k of the data simulated from the METANET model, $v_{\mathrm{SUMO},i}(k)$ and $\rho_{\mathrm{SUMO},i}(k)$ are the mean speed and density of the segment i at the iteration k of the SUMO data.

Other objective functions could be applied, using different variables. Tests have been done using the velocity and the flow instead of the density. The obtained results were awry and inaccurate compare to the other objective function with velocity and density, with an objective function value three times greater than the value obtained previously, and the obtained plots showed an important gap between the SUMO and METANET data.

Therefore, the objective function with velocity and density as variables is chosen. Different weights have been applied to the variables in order to find the most precise combination. In Table 2.2, the different combinations of weights are shown. As it can be observed from Figure 2.2, all the objective functions provide a result that is close to the SUMO's data. But when looking more specifically at the 4th segment, it can be seen that there is a gap between the METANET and SUMO data when the lanes are closed for most of the combinations. The closest one to the SUMO's data, considering all segments, is the combination number 4, which is equivalently applying the weight to both variables. Therefore, this objective function has been chosen.

Table 2.2. Variables' weights combinations for the objective functions.

Configuration number	Density's weight (%)	Velocity's weight (%)	Color in the plots
1	0	100	Green
2	20	80	Turquoise
3	40	60	Blue
4	50	50	Orange
5	60	40	Brown

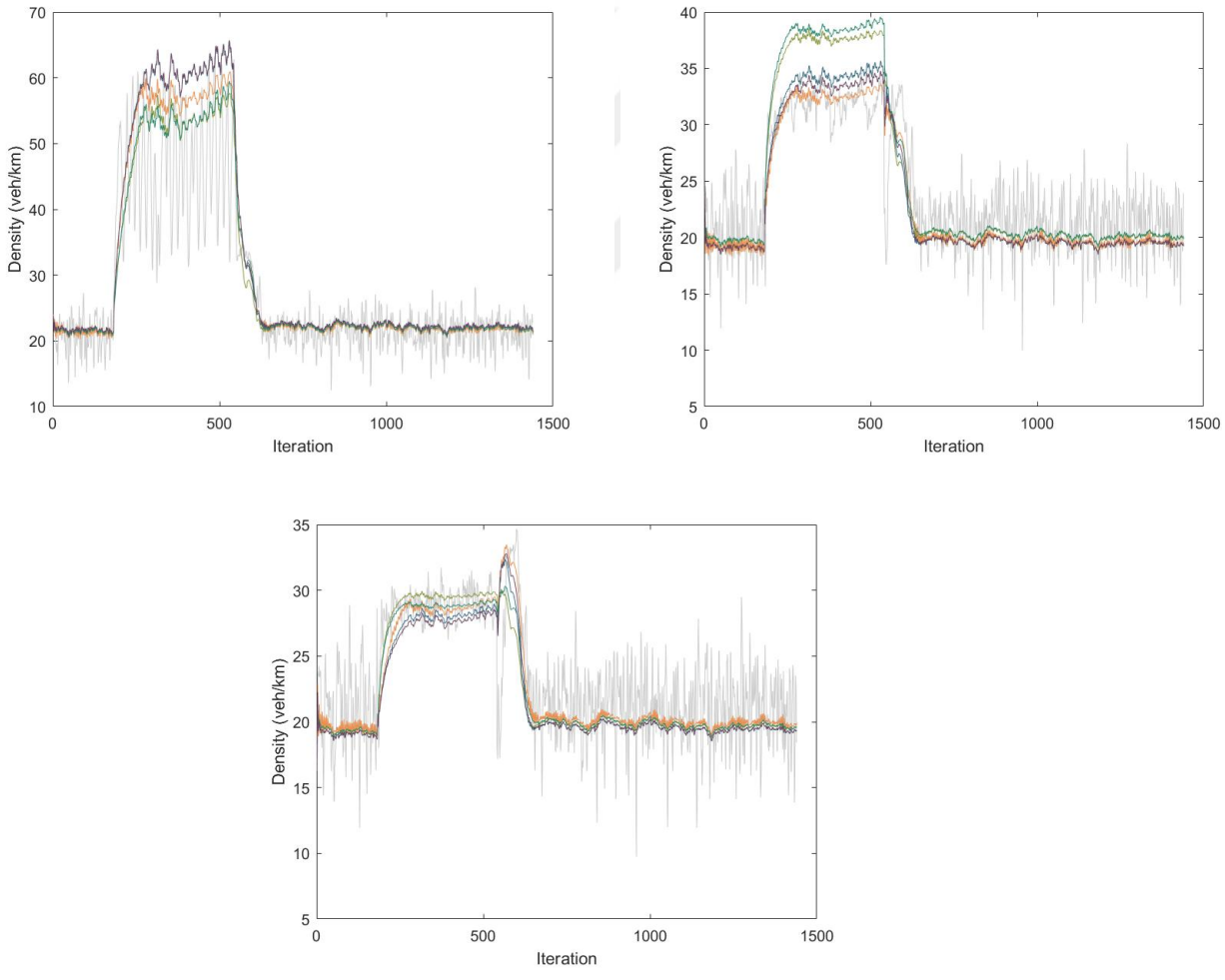

Figure 2.2. Comparison of the METANET and SUMO data for segments 3, 4 and 5 for the different objective functions.

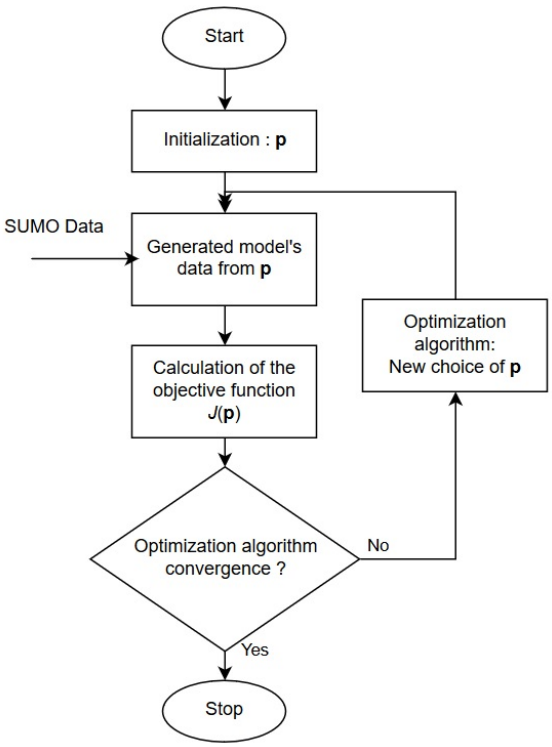

Figure 2.3. Flowchart of the MATLAB calibration.

The calibration is performed by segment: each segment has its own set of parameters that are optimized during the calibration. All segments are optimized in the same iterations of the calibration phase: since each segment's data have an influence on its neighbours' data, running calibrations by considering only one segment at a time would not produce great results. Therefore, all segments are considered during the optimization phase, and parameters are optimized depending on the other ones.

The off-line calibration has been applied in the MATLAB software. MATLAB is a mathematical programming platform used to perform complex mathematical computations. It is composed of a programming language based on the use of matrices which allows the user to manipulate them easily, and it is also composed of a robust core that contains multiple optimization algorithms that are able to solve complex problems in a short time. The interest of using this software is to obtain the results in the shortest way and to have the ones that are more accurate. A lot of available functions in MAT-

LAB are also available in other programming languages, but MATLAB comes with a database which optimizes all the calculation to shorten the process' time.

The function used in MATLAB to process the calibration is `patternsearch`. This function is a numerical optimization function that can be used to find a minimum function's value of a stated problem. The syntax of the function and its parameters are described in Appendice B.

The `patternsearch` function in *MATLAB* applies the *Generalized Pattern Search (GPS)* algorithm to optimize the objective function. The algorithm's goal is to find the minimum values of the parameters that minimize the function, as shown in Equation 2.16.

$$\min_{x} f(x) \quad \text{under} \quad lb \leq x \leq ub \tag{2.16}$$

where $x \in \mathbb{R}^n$ and $f : \mathbb{R}^n \longrightarrow \mathbb{R}$
and ub, lb are the upper and lower boundaries as explained in section 3.3.

The whole instruction to apply the *Generalized Pattern Search* algorithm is described in Appendix A.

When calculating the flow dynamic to generate data, some conditions are applied. The drop of lanes is calculated with the formula shown in Equation 2.17. This formula assures that when a lane-drop is occurring, the speed-drop is applied on the upstream segment of where the closing lanes happened, at segment 3 for the studied network.

$$\texttt{lanedrop} = \lambda_k - \lambda_{k+1} \tag{2.17}$$

$$\text{if} \quad \texttt{lanedrop} < 0, \text{ then } \texttt{lanedrop} = 0 \tag{2.18}$$

Where λ_k and λ_{k+1} are respectively the number of lanes of the segment k and $k+1$.

The density also needs conditions about the possible values it can have. When calculating the density with the formulas, it is possible to obtain a negative value in certain cases when there is a strong difference with the previous segment's flow and the current segment's flow. At the opposite, the density should not be over a maximum value. On the segments where the flow is null and the incoming flow is important, the density could rise up to colossal values. A specific condition is applied to the last segment of the road stretch. The last extra segment, corresponding to the 9th segment in this thesis' network, has to certify that it is not congested to assure that the road stretch unclogs when congestion happens. Else, once congestion happens, the density would stay at its maximum until there is a drop in the incoming flow. The condition shown in Equation 2.20 ensures that the last segment will never be congested.

$$0 \leq \rho_i(k) \leq \rho_{max} \cdot L_k \tag{2.19}$$

$$\text{if } \rho_N(k) > \rho_{cr}, \text{ then } \rho_N(k) = \rho_{cr} \tag{2.20}$$

Comparatively to the density, the mean speed is limited to its range of possible values. A minimum value has to be set to avoid negative values. However, in contrast of density, the condition for the velocity should not be set to zero. The value should be close to zero but not null in order to have a more realistic behaviour of the cars when the congestion is ending. The velocity should not obtain values over a certain limit to guarantee reasonable values. Therefore, the mean speed is constrained to the Equation 2.21.

$$v_{min} \leq v_i(k) \leq v_{max} \tag{2.21}$$

The calibration part process applies the flowchart visible in Figure 2.4.

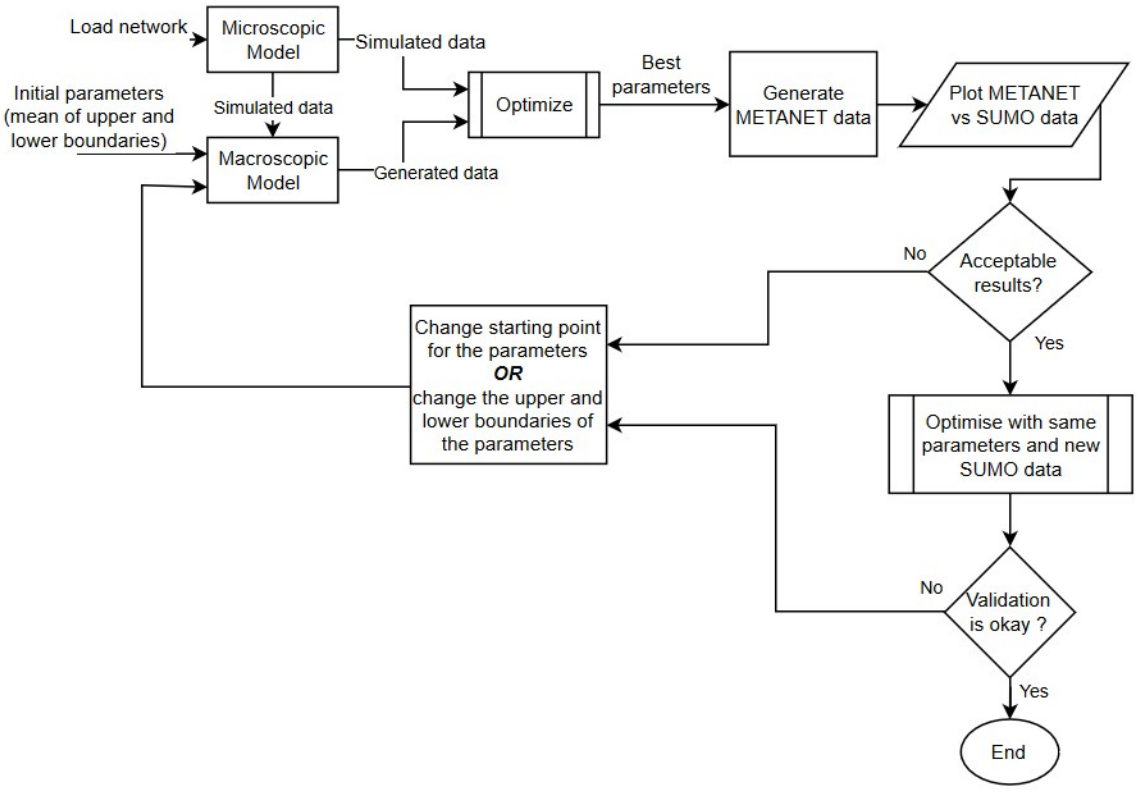

Figure 2.4. Flowchart of the MATLAB calibration.

Each parameter has its influence on the objective function's value (see Equation 2.15). Analysing the cost function's behaviour is interesting to see if there are any possible limits due to an overly narrow boundary. Indeed, if the objective function has its minimum value possible on one of the parameter's boundaries, it means that a lower value can be found if the boundary is expanded. However, a boundary should not be enlarged automatically due to this phenomenon: the boundaries should respect values that represent the real world. The boundaries of the parameters are depicted in the Table 2.3.

Table 2.3. Initial point and precision each processed parameter's values.

Parameter	ρ_{cr}	ρ_{max}	v_{min}	α	τ	ν	κ	ϕ	v_{max}
Unit	$veh/km/lane$	$veh/km/lane$	km/h	$-$	$hours$	km^2/h	$veh/km/lane$	$-$	km/h
Initial values	37.5	111	4	2.15	0.0084	60	100	2.515	95
Precision	0.1	0.1	0.01	0.01	0.1	0.1	0.1	0.001	0.1

In Figure 2.5, the parameters which showed a variation on the objective function value. Most of the parameters have the minimum value of the objective function at one of their boundaries. That is the case of ρ_{cr}, ρ_{max}, τ, ν and κ. For κ, expanding the boundary would set a negative value to the parameter, which should not happen since it might give false result due to the negativity. For the other parameters, increasing the upper boundary would not give realistic values. α is the only parameter which has not its minimum at the boundaries. The parameters v_{min}, v_{max}, ϕ stay constant on all their possible values. This does not prove that they have no impact on all the objective function: When varying one parameter at a time, the other parameters have their values fixed. If the other parameters' values change, then parameters as v_{min}, v_{max} and ϕ might not stay constant anymore.

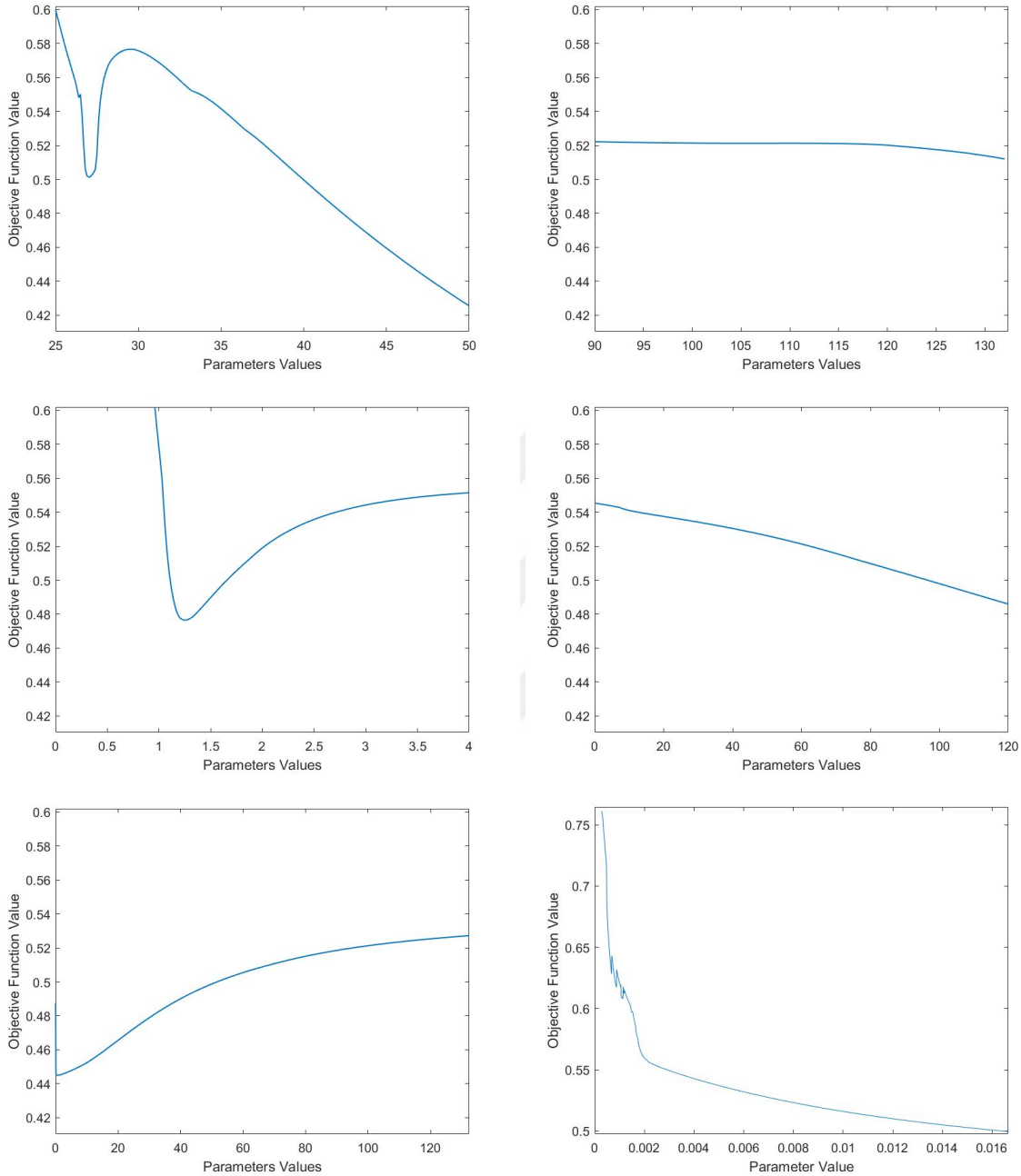

Figure 2.5. Influence of the parameters (from left to right) $\rho_{cr}, \rho_{max}, \alpha, \nu, \kappa, \tau$ on the objective function.

To observe the parameters' evolution during the calibration, the calibration is launched using the mean of the lower and upper boundaries as the initial point (see Table 2.3).

The `patternsearch` function is run over 400 iterations and the most varying parameters are plotted for the segments 1 and 6 in Figure 2.6 and Figure 2.7 respectively. Some parameters as ϕ do not vary for most of the segments and stay constant within the whole calibration, thus, only the parameters that showed variations on every segment are plotted.

The critical density ρ_{cr} rises up to the upper limit of the boundaries on most of the segments except segment 1, 3 and 5 where it rises up to 31, 37 and 45 respectively.

The maximum density ρ_{max} changes only for the segments 1 which is concerned by the congestion due to the lane closure.

For κ parameter, the variations are important for the segments where the lane is closed and segments placed in the downstream of the network.

The same phenomenon happens with ν. This parameter is linked to the value of κ, as it can be seen, when the value of κ decreases, the value of ν increases for the last two segments.

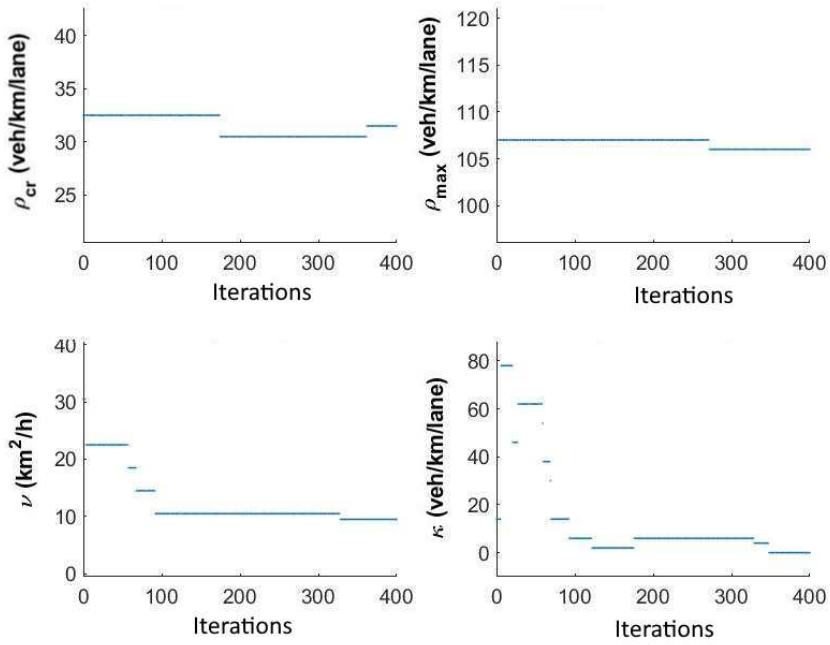

Figure 2.6. Convergence of the parameters $[\rho_{cr}, \rho_{max}, \nu, \kappa]$ of Segment 1 over 400 Iterations.

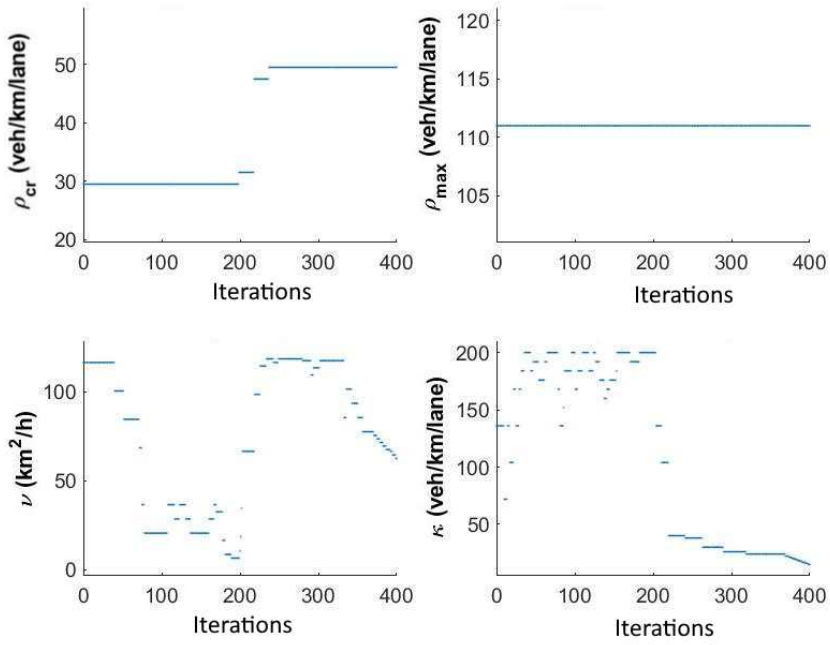

Figure 2.7. Convergence of the parameters $[\rho_{cr}, \rho_{max}, \nu, \kappa]$ of Segment 6 over 400 Iterations.

2.3. Choice of methods for data collection, data processing and analysis

Multiple type of data can be used to study and to test the different traffic flow models. Each type has its advantages and inconvenient, which can give differences in the results of the models. The following types of data exist:

- Real-time data: These data are collected using detectors placed on the road. The common type of detectors used is the induction loop detectors, which gives information about the speed and flow of vehicles driving on it. The data are transmitted in real time to a server and can be used directly for the model. They can be aggregated during a certain period of time, from some seconds to some minutes, depending on the required precision. The interest of this type is to simulate the traffic state in order to predict possible future congestion.
- Collected data: The collected data are data coming from detectors, which can be induction loop detectors or even cameras. As for real-time data, the data can be aggregated over a period of time. This type of data is useful when calibrating and testing models as they allow the user to have a referential that will not change from a simulation to another. The data can also be processed before using a model if there is a need.
- Simulated data: The simulated data are data obtained using simulation software. Using simulations, it is possible to simulate existing roads in different cases, creating an important traffic flow during a period of time or closing lanes during some time. These situations can be rare to observe in real life and the ability to simulate them is a real advantage. The down side of this solution is to obtain a realistic behaviour of the drivers and of the events happening on the road: even if there are a lot of settings and actions that can be applied in the simulation, it is not possible to perfectly mimic what would happen in real life, however by calibrating the simulation with real data, acceptable results can be obtained. The simulated data can be used in real-time or as collected data.

2.4. Validation of the model

The model has to pass through a validation phase. This validation phase ensures that the model is giving the same accuracy for the results with different data sets. If a set of data gives excellent results, the model might be effective only for this set and might give worse results on all other data sets due to over-fitting. Once a model is validated through the validation phase, it means it can be generalized to all data sets without the fear of obtaining false results.

In this thesis, once the model is calibrated, a validation process will be applied to ensure that it can be generalized to other dataset. After the calibration, the parameter set is saved. A brand new set of data from SUMO is generated, using the same network characteristics as before, while applying a different seed to add randomness in the traffic. This new data set is then used as an input for the METANET's model with the saved parameters set. The error is measured for the objective function and the result is compared to the error of the calibrated dataset. The maximum accepted error value in this thesis is 20 %, this value has been chosen according to the literature. Then, a visual comparison of the velocity, density and flow variables are done. If the METANET's generated data with the new dataset matches the calibrated dataset almost perfectly, then the model is considered as validated.

3. Computational Experiments and Results

3.1. SUMO

The studied road for collecting the data is the part of the E4 road located in the Salem area of Stockholm, Sweden, which an aerial view is shown in Figure 3.1. The studied road is a 3-lanes highway that is approximately five kilometers long. E4 is one of the Sweden's most important highways since this road is passing through or by fifteen largest cities of Sweden.



Figure 3.1. The road stretch viewed from Google Maps.

A part of the E4 road has been modeled in SUMO. The network file is taken from the Modeling and Analysis of PID-Controlled Heavy-Duty Vehicle Platoons in Real Traffic paper (Jin, Ma and Johansson 2016). The road stretch is 4.53 kilometers long and is divided in nine segments of different lengths, although the lengths of the model are modified for the purposes of this study to have the same or nearly the same value. A scheme of the road is shown in Figure 3.2, and the graphical view of this road is displayed in Figure 3.3. Previously, in this paper (Jin, Ma and Johansson 2016), the

lengths varied between 259.57 to 675.73m. New values for each segment can be found in the Table 3.1. All the segments are composed of three lanes and have a speed limit of 90 km/h. There is neither a lane reduction nor on-ramp/ off-ramp. The vehicles are generated from the first segment which corresponds to the further south-west point of the road. A continuous flow of 1000 vehicles per hour per lane is generated on the segment's departure lanes, resulting in a total flow of 3000 vehicles per hour on the road stretch. The first and last segments are considered as extra segments, called Extra segment 0 and Extra segment 8 respectively. Their values are retrieved and used as boundary conditions in the second order model. The Extra segment 0 cannot be used as an internal segment as the vehicles are generated on it, and some time is needed before the vehicles obtain their desired speed.

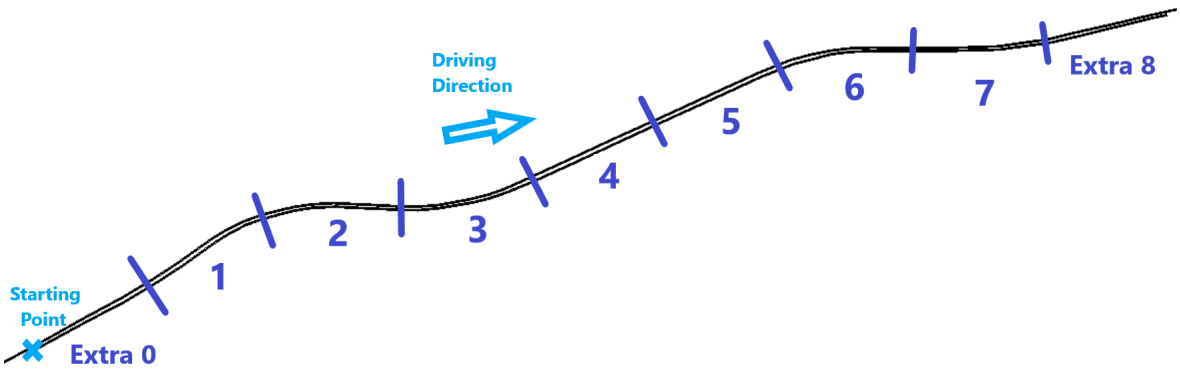

Figure 3.2. Scheme of the network modeled in SUMO.

Table 3.1. Lengths of the segments of the road stretch.

Segment	Length (meters)
Extra segment 0	517.23
1	499.51
2	512.43
3	507.88
4	508.03
5	502.88
6	501.89
7	495.90
Extra segment 8	469.97

The generated vehicles are passenger vehicles. The car-following model used for this model is Krauss, which is the model applied by default in SUMO. The lane changing model applied in this SUMO simulation is as well the default one named *LC2013*. Each car has a length of five meters, a maximum speed of 92.52 km/h and an acceleration and deceleration of 4.8 m/s^2 and 4.0 m/s^2 respectively. Each driver has a driving imperfection σ set to 0.7. Concerning the car following model parameters, the cars have a value for τ, the drivers desired (minimum) time headway, of 0.9 and a deviation of the speed factor equal to 0.1.

Different scenarios are simulated in SUMO. All of them run with the same following simulation time which is set to 4 hours, with 15 minutes of warm up before. The warm up is mandatory to obtain convenient data when the data collection starts: When the SUMO software is launched, the road stretch is empty. The vehicles will be spawned by the indicated flow, however, it requires some time to have generated vehicles through our whole road stretch. A period of 15 minutes is sufficient to populate the road stretch and obtain a realistic situation.

The road network, the vehicles' type and flow files are already created before launching the simulation. The simulation is launched through Python, which sets the seed for the randomness of the drivers behaviour and then launch the simulation in SUMO. Python controls the progression of the SUMO simulation. The applied mehod to retrieve the values in SUMO is to use the functions to retrieve the values of a specific lane or a segment. TraCI provides functions to retrieve some important data about a lane or a segment, such as the mean speed or the number of vehicles driving on it. When a function is selected and called in Python, TraCI returns the requested values that gathered during the last second in the SUMO simulation. Unfortunately, as it is not possible to set the time step for SUMO, which is the time during which SUMO collects data and gives the mean of them at the end, to a value bigger than 1 second, the Python code retrieves the values of every lane at each second of the simulation. At each 10 seconds, both SUMO and METANET models' time step, the collected values are processed to obtain an average of the mean speed and the number of vehicles that

drove on the lane over the last ten seconds. TraCI does not have functions to directly retrieve the flow and density of a lane or segment, therefore, the mean speed and number of vehicles are the only retrievable values from TraCI that can be used in the METANET model's formulas. The values per lane are stored in arrays, and the values per segment are computed as follow :

$$v_i(k) = \frac{1}{T} \cdot \sum_{t=1}^{T} \frac{\sum_{l=1}^{\lambda_i} v_{i,l}(t)}{\lambda_i} \tag{3.1}$$

$$\rho_i(k) = \frac{1}{T} \cdot \sum_{t=1}^{T} \frac{\sum_{l=1}^{\lambda_i} vehicles_{i,l}(t)}{\lambda_i \cdot L_i} \tag{3.2}$$

Where $v_i(k)$ is the mean speed of the vehicles driving on the segment i at the iteration k, T is the global timestep, t is the seconds passed in the simulation, λ_i is the number of lanes of the segment i, $v_{i,l}(t)$ is the mean speed of the vehicles driving on the lane l of segment i at the second t, L_i is the length of the segment i, $vehicles_{i,l}(t)$ is the number of vehicles driving on the lane l of segment i at the second t. The density that is calculated for the segment has the unit of $vehicle/kilometer/lane$ which is the unit that has to be used in the model's formulas. Once the simulation is finished, the arrays are saved to text data files to be used later in the calibration part.

This method is convenient to obtain exact data, as the retrieved values give the mean speed and the number of vehicles at a time t with high precision. However, it is hardly possible to obtain such data in real-life, as it would require a lot of sensors and cameras to be able to obtain the exact values of the requested parameters.

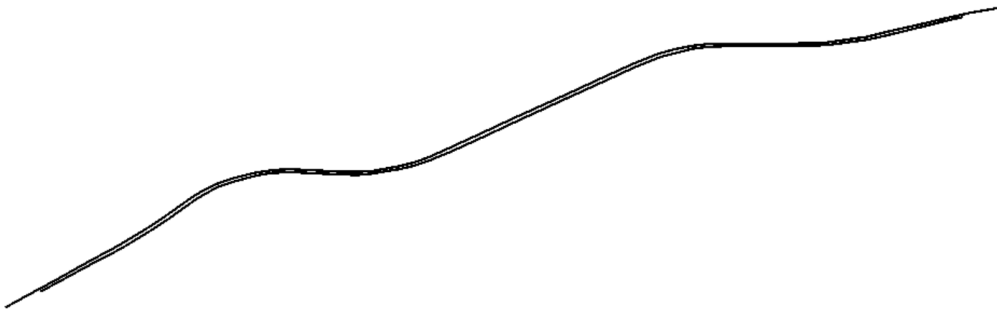

Figure 3.3. View of the studied road stretch in `sumo-gui`.

3.2. METANET

The network modeled in METANET and SUMO is the same. The E4's road stretch is divided into 9 segments, thus $N = 9$. The segments' lengths L_i are detailed in the Table 3.1. The lengths have been set to values around 500 meters in order to use the same time step of 10 seconds for all segments while respecting the conditions shown in Equation 2.5. The time step of 10 seconds sets the value of the parameter to $T = \frac{10}{3600} = 0.0027$. .The first and last ones located at the ends of the road stretch are considered as extra segments. In this thesis, the free flow speed is chosen as constant with $v_{free} = 90\ kilometer/hour$. The free flow speed value corresponds to the speed limit applied on the studied highway in Sweden.

A variable ρ_{max} has been set to the value of 132 veh/km/lane. This value has been calculated using SUMO's simulation and counting the number of vehicles on one 0.5 km long lane that is congested at its maximum. This number depends on the segment's length, since no segment are 1 kilometer long. The density $\rho_i(k)$ has to respect the conditions stated in Equation 2.19. In this thesis, the values for the density ρ_0 and mean speed v_0 of the incoming flow are retrieved from simulation data from SUMO.

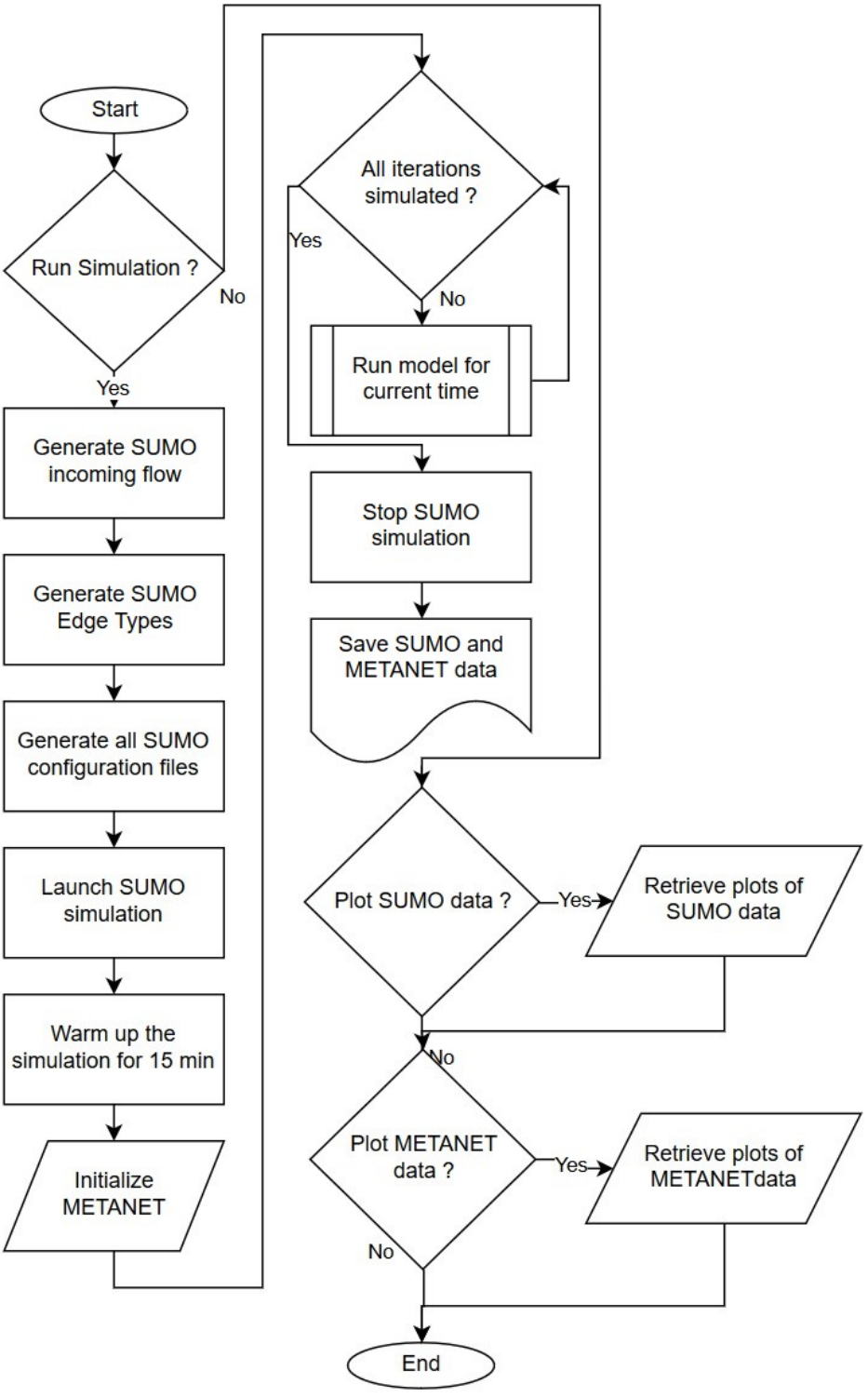

Figure 3.4. Flowchart of the `main.py` Python file.

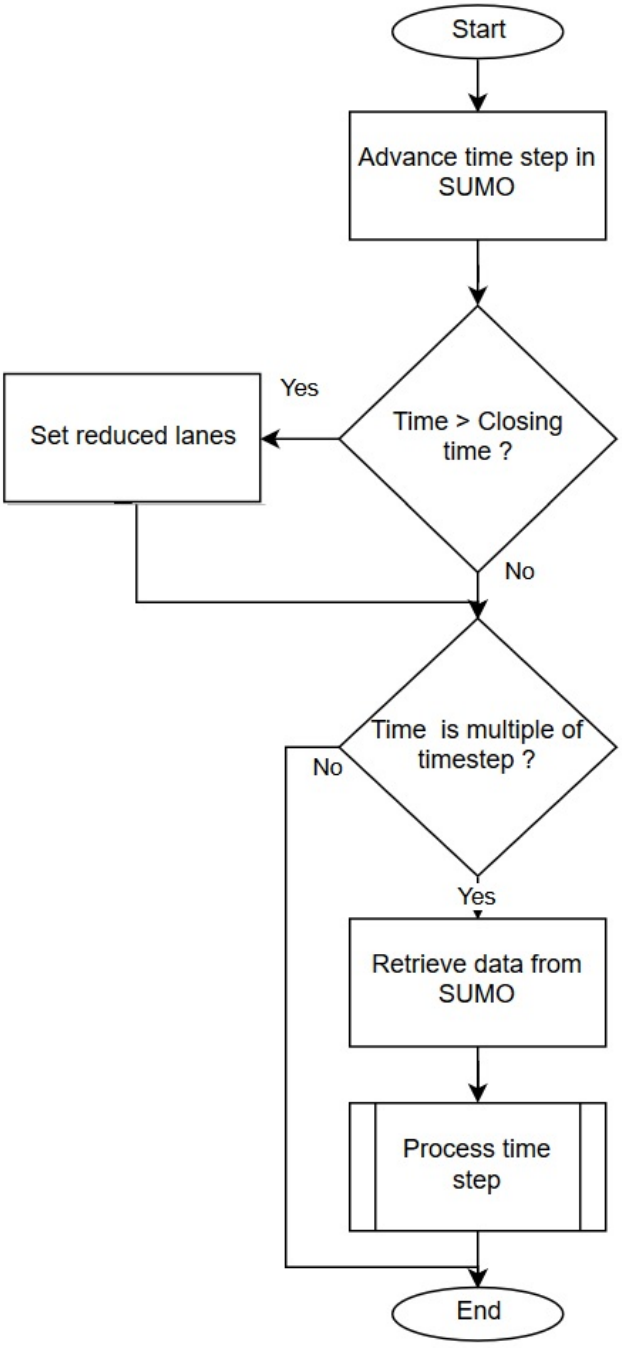

Figure 3.5. Flowchart of the progression of the simulation in the file `main.py`.

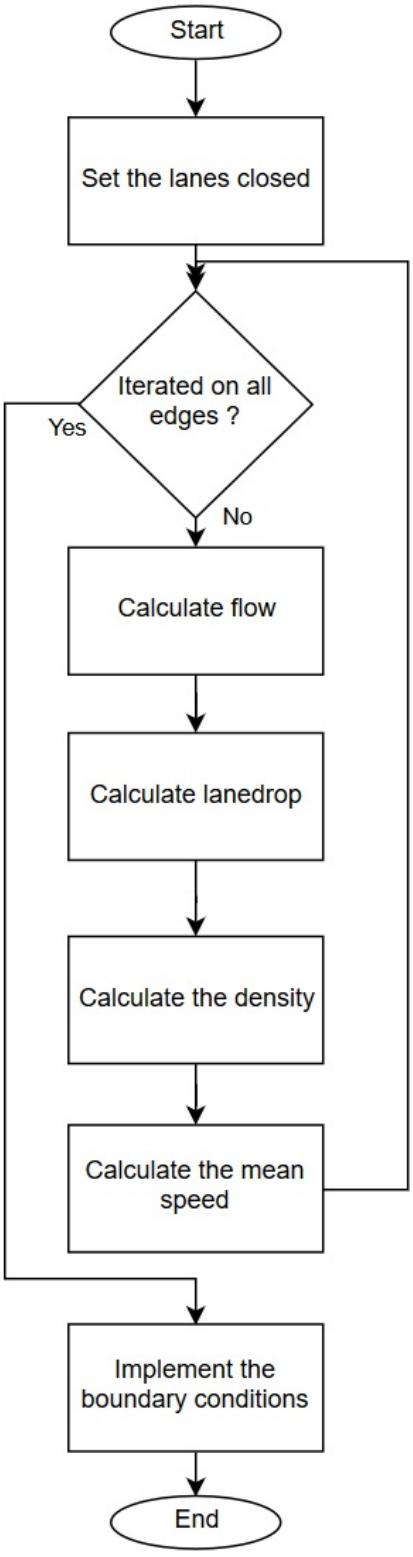

Figure 3.6. Flowchart of the METANET model.

3.3. Calibration and Validation Results

In the objective function formula, as described in Equation 2.15, N is set to 7, corresponding to the 7 studied segments, and K is set to 1440, corresponding to the number of iterations obtained after 4 hours of simulation with a time step of 10 seconds. The parameters given in p require to have lower and upper boundaries in order to obtain realistic results when running the calibration. If these boundaries are not set, the parameters that will be returned by the calibration might have unrealistic values, for instance, a negative number for the critical density. The following values have been chosen as lower and upper boundaries for the parameters:

$$
\begin{aligned}
25 &\leq \rho_{cr} \leq 50 \\
0.1 &\leq \alpha \leq 7 \\
\frac{1}{3600} &\leq \tau \leq \frac{30}{3600} \\
5 &\leq \nu \leq 120 \\
0 &\leq \kappa \leq 200 \\
0.0001 &\leq \phi \leq 5 \\
90 &\leq \rho_{max} \leq 135 \\
1 &\leq v_{min} \leq 7 \\
90 &\leq v_{max} \leq 100
\end{aligned}
$$

α parameter has been tested with multiple values, and values between 0.1 and 7 were the ones giving the best results to match the data.

τ is one of the parameters of the anticipation term. The values of 1 second to 30 seconds give a range up to the extreme values a driver could have when driving, since a driver would not take more than 30 seconds to adapt its driving speed in the real world.

ν and κ can be found in the anticipation term. Both have been tested with values in a wide range in order to observe their influence on the results, and the values between 0 to 200 for ν and 5 to 120 for κ were the closest to what would happen in real life.

ϕ is used to represent the influence of the lanes that might be dropped due to any incident that may happen, as any accident or roadworks. When testing the influence of ϕ on the velocity, the best values were obtained within a range of 0.0001 to 7.

The calibrations are launched in parallel in MATLAB using the calibration tool `parpool`. This tool allows to run specific code in parallel on the available CPU cores of the computer, instead of running it sequentially. When running MATLAB code that contains a parallel function, like `parfor` which executes the `for` loop in parallel, the `parpool` function is called and launches the parallel mode in the software. For each processor's cores available on the computer, a worker is created. When created, a worker is assigned an ID: for `parfor`, a randomly taken value from the range of the `for` loop and that has not already been taken by another worker is used as ID. Then, each worker executes the code written in the loop. It is possible to use external variables from the loop, but they cannot be modified by any worker, and when a worker creates a variable, it is only visible and accessible to the worker. The code has to respect some requirements to be able to run in parallel: No global variables can be used and if there are data that should be saved before exiting the function, they have to be saved in an external array at the position equal to the ID of the worker. The time saved by running the code in parallel is important, as a single calibration can take up to 5 minutes to run, running it on multiple cores divides the total time by 4 to 6, depending on the number of cores available.

When running the calibration using the options :

$$optimoptions('Display','iter','PlotFcn',@psplotbestf)$$

The convergence of the objective function is shown. Starting from the given starting point, the calibration will run for the set amount of iterations, or until no improvement can be made. The graphical plot of this convergence can be seen in the Figure 3.7. As it can be observed, the function quickly optimizes the objective function during the first iterations. After around 70 iterations, the calibration functions takes more time to optimize and the gap in the objective function value between each iteration is smaller.

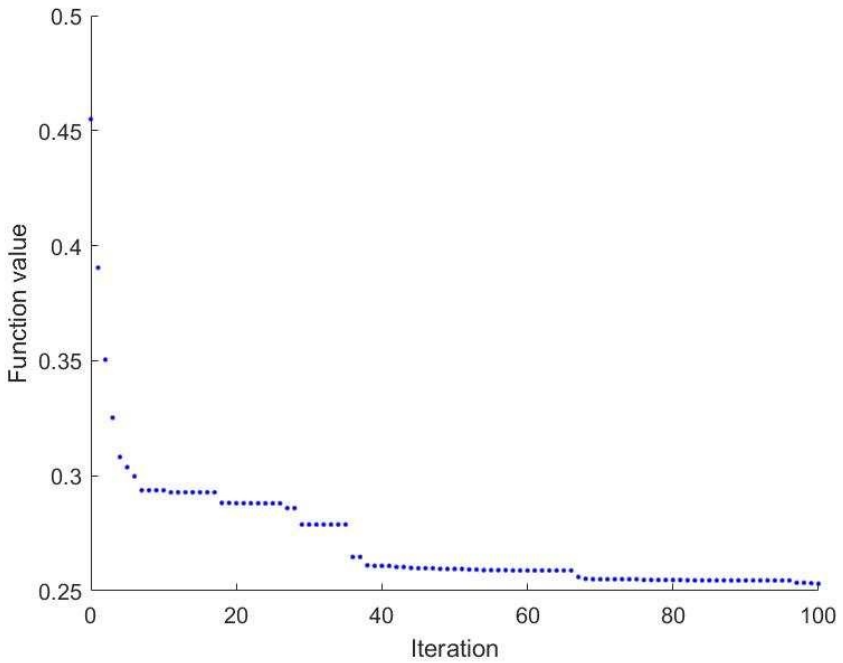

Figure 3.7. Convergence of the *patternsearch* function.

In the Table 3.2, the textual advancement of the `patternsearch` function is shown. In this advancement, it is possible to observe how the `patternsearch` algorithm works. At the beginning, the algorithm can reduce the objective function over multiple iterations, that is why it increases its mesh size to browse the solutions faster. These iterations are called Successful Poll, and the size of the mesh is multiplied by 2. At the iteration 9, it is not possible to find a better solution. Thus, the point and value of the objective function stay the same, the iteration is called Refine Mesh and the size of the mesh is divided by 2. At the iteration 22, it becomes more complicated to obtain a successful poll where a value is found, and the mesh is refined over multiple iterations to find a new solution. The functions count, that is the number of time the

objective function is evaluated, is also shown for each iteration. Once the calibration phase is finished, the results from the 200 iterations are analyzed. The parameters combination that minimizes the most the objective function are kept as the result of the whole calibration process. At the end of the iterations, all the obtained parameters should converge toward the same values.

Table 3.2. Textual output of the *patternsearch* function.

Iter	Func-count	f(x)	MeshSize	Method
0	1	0.454957	1	
1	134	0.390429	2	Successful Poll
2	256	0.350459	4	Successful Poll
3	342	0.325241	8	Successful Poll
4	413	0.30815	16	Successful Poll
5	466	0.303684	32	Successful Poll
6	495	0.299732	64	Successful Poll
7	514	0.293673	128	Successful Poll
8	522	0.293673	256	Successful Poll
9	522	0.293673	128	Refine Mesh
10	530	0.293673	64	Refine Mesh
11	550	0.292791	128	Successful Poll
12	559	0.292791	64	Refine Mesh
13	578	0.292791	128	Successful Poll
14	587	0.292791	64	Refine Mesh
15	606	0.292791	128	Successful Poll
16	615	0.292791	64	Refine Mesh
17	633	0.292791	32	Refine Mesh
18	662	0.288174	64	Successful Poll
19	680	0.288174	32	Refine Mesh
20	708	0.288034	64	Successful Poll
21	726	0.288031	128	Successful Poll
22	734	0.288031	64	Refine Mesh
23	753	0.288031	32	Refine Mesh
24	780	0.288003	64	Successful Poll
25	799	0.288003	32	Refine Mesh
26	826	0.288003	16	Refine Mesh
27	878	0.28592	32	Successful Poll
28	905	0.28592	16	Refine Mesh
29	957	0.278789	32	Successful Poll
30	984	0.278789	16	Refine Mesh

3.3.1. First Scenario: One-Lane Closed with a Flow of 4500 veh/h

The scenario is introduced to have a congested traffic flow condition. For this manner, one lane is closed for one hour during the whole simulation and will be re-opened after 90 minutes of simulation, with a flow of 4500 vehicles per hour. One example of this scenario can be a harsh accident happening on the road which will occupy around 1 km on one lane during the peak hours. Like the second scenario, segments 4 and 5 have their left-most lanes closed.



Figure 3.8. Lane-Closure on Segment 4 and 5 in SUMO using the Rerouter feature.

With this flow, stop-and-go waves can be observed in the SUMO simulation. This waves are composed of a Generation phase, where the congestion starts to happen. Then, the congestion grows and the vehicles have to reduce their speed, which can go down to zero for some waves. Then, after some time, the congestion gets dissolved, and another waves appear again. The phenomenon happening in SUMO can be seen in Figure 3.9.

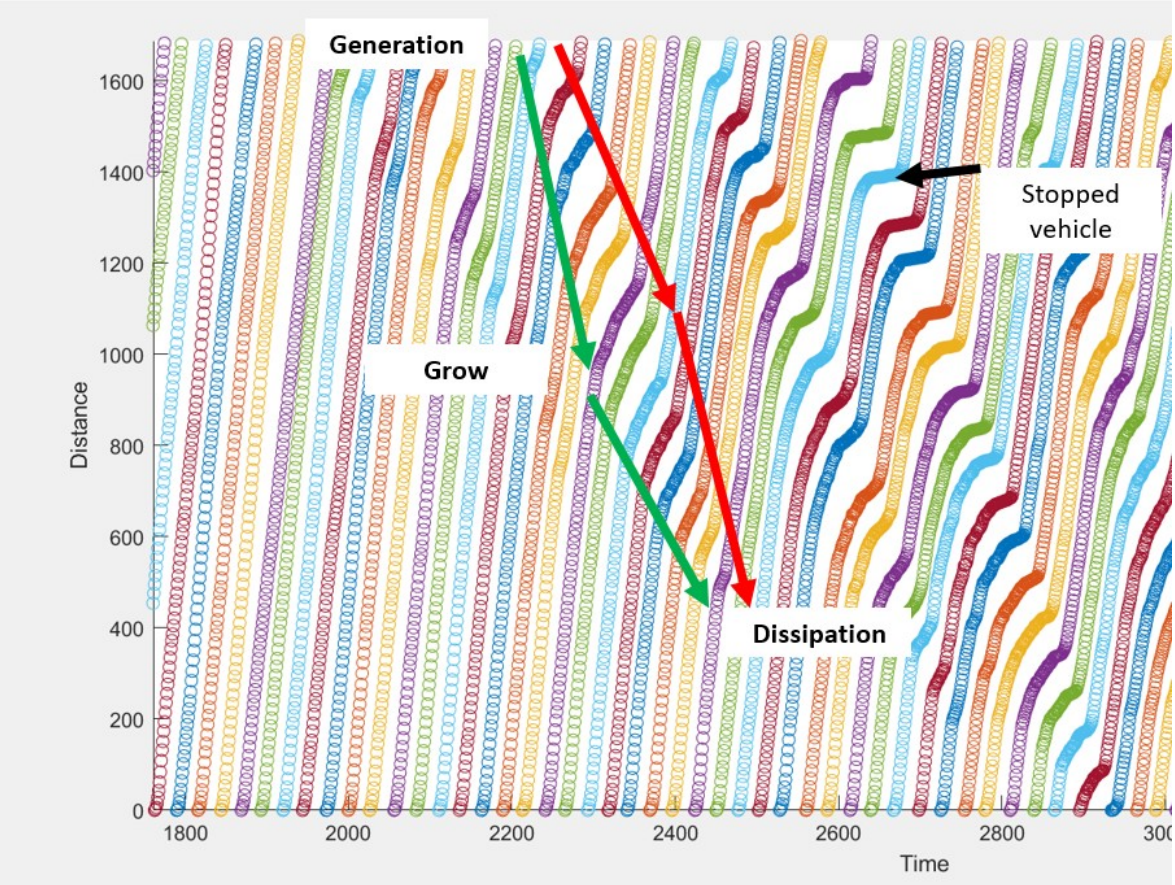

Figure 3.9. Stop-and-go waves phenomenon with the SUMO's data.

Table below, shows the results of the model's parameter values after the off-line calibration phase in MATLAB.

Table 3.3. Best obtained parameters for the one-lane closed scenario with a flow of 4500 veh/h after the per segment calibration.

Segment	ρ_{cr}	ρ_{max}	v_{min}	α	τ	ν	κ	ϕ	v_{max}
	$veh/km/lane$	$veh/km/lane$	km/h	–	$hours$	km^2/h	$veh/km/lane$	–	km/h
Before calibration	37.5	111	4	2.15	0.0084	60	100	2.515	95
Segment 1	32.50	111	4	2.050	0.0043	30.50	14	2.500	95
Segment 2	49.50	111	4	1.050	0.0043	66.50	0	2.500	95
Segment 3	37.50	111	4	1.550	0.0082	64.50	115	4.875	95
Segment 4	42.50	111	4	2.050	0.0043	110.5	34	2.500	90
Segment 5	30.50	111	4	2.050	0.0043	70.50	68	2.500	95
Segment 6	29.50	111	4	2.050	0.0043	116.5	136	2.500	95
Segment 7	33.50	111	4	2.050	0.0043	6.500	200	2.500	95

The calibration phase was run using a SUMO dataset that was formed from running five simulations with a different seed each time, in order to have smoother data and reduce the noise present in the data. The obtained result for the objective function value was 0.1581577. When using the RMSE function, the value should be close to 0 in order to have a good fit, which already indicates that the model and parameters give good results. The objective function is calculated again with the obtained parameters during the calibration, but using another SUMO dataset. The objective function value calculated along with the obtained plots in the validation phase are compared to the previous obtained ones in order to see whether the model can be considered as validated. After running the objective function with a new dataset, the retrieved value was 0.1508746, which represents a difference of -4.6049%. This improvement is a good indicator of the model's accuracy, and during the plots comparison, no big differences have been captured. Thus, the model is validated. It should be mentioned that the model is not over fit since over-fitting occurs when there is a huge difference between calibration error value with validation error value. If the model has a very low error value in calibration phase but a large one in the validation phase, it means the model mimicked not only the signal patterns but also all the noises. If the model was calibrated regarding noises, the objective function value for the validation phase should have been way larger than 0.1581577.

Table 3.4. Comparison of the objective function values obtained in the calibration and validation phases for the one-lane closed scenario with a flow of 4500 veh/h.

DataSet	Objective Value	Difference (%)
Calibration	0.1581577	-
Validation	0.1508746	-4.6049

3.3.1.1. Density. Outside of the closed lane part, the density is exactly the same between SUMO and METANET. The small variations happening in SUMO cannot be reproduced in METANET as it is a Macroscopic model, whereas SUMO is microscopic. When the lane is closed, the density shows the same ascent over the 50 first iterations, but then, the oscillations are less than SUMO.

On segment 4 and 5, METANET generates data that match almost perfectly with the data from SUMO on all the simulation's duration. Around the 540th iteration, where all the lanes are opened, a difference appears between METANET and SUMO : This is due to the fact that SUMO opens and closes the lane instantly. When the lane is opened, over one iteration, segments 4 and 5 will be considered as having 3 lanes, but they will have 1 empty lane, which creates these down peak at the 540th iteration. Instead, a descending (going down) curve should happen when all the lanes are opened again. For the last two segments, the density generated with METANET matches perfectly the SUMO data.

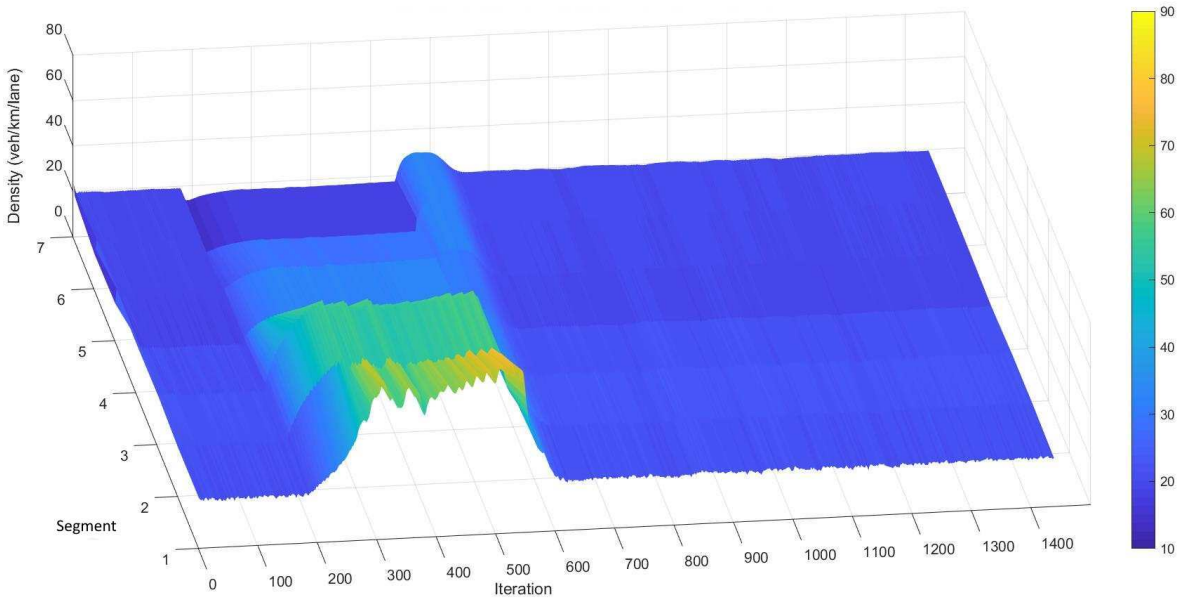

Figure 3.10. 3D plot of the density from METANET data

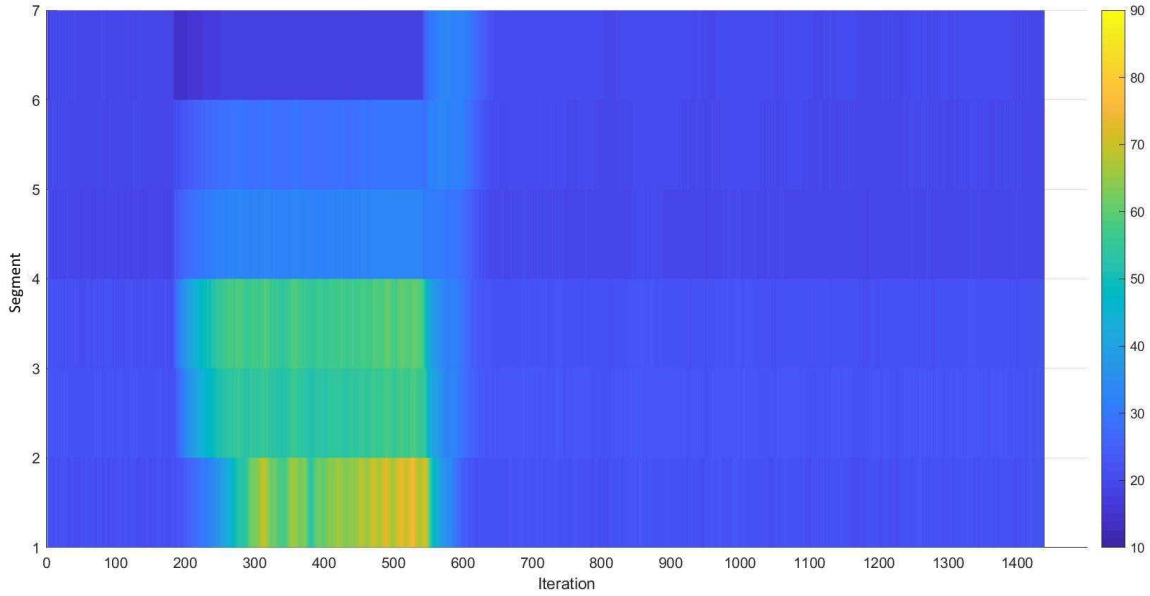

Figure 3.11. 3D plot of the density from METANET data (View from the top)

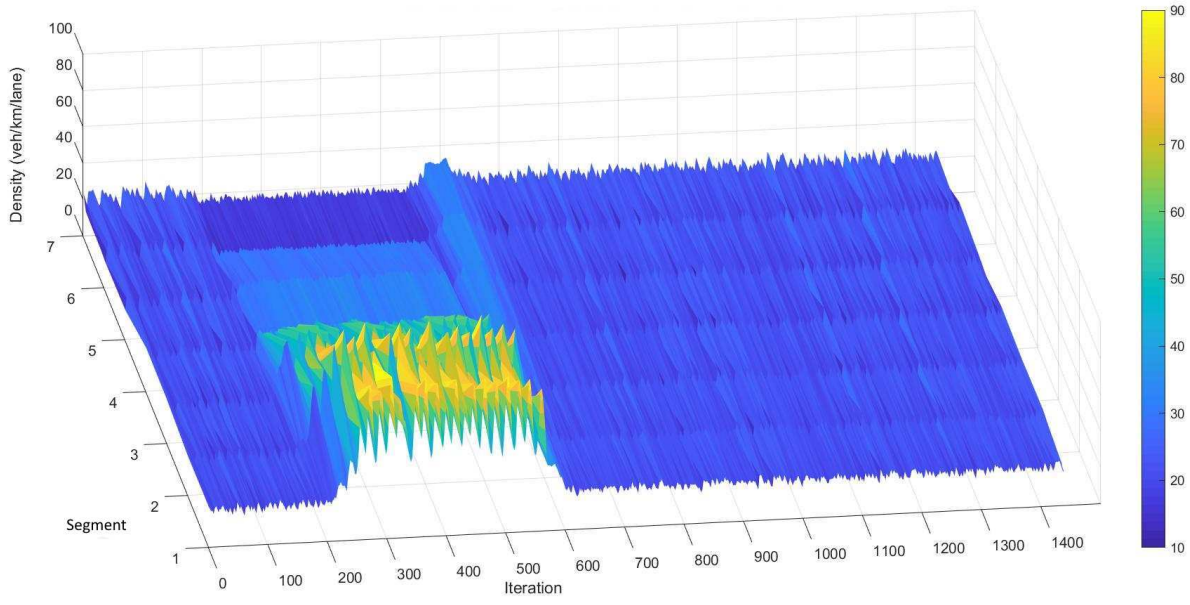

Figure 3.12. 3D plot of the density from SUMO data

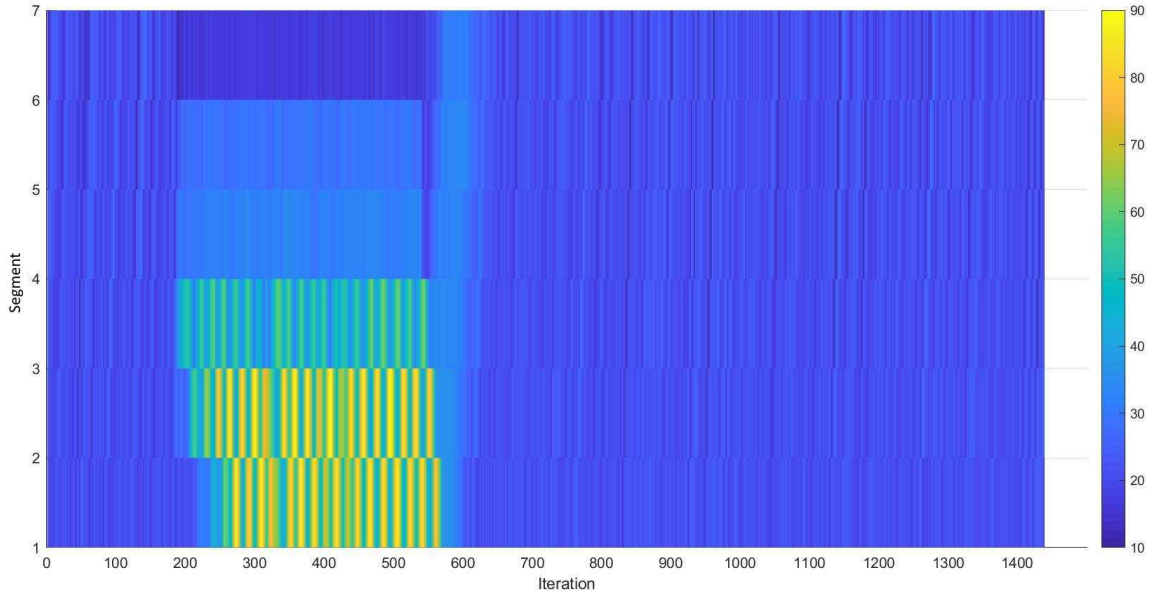

Figure 3.13. 3D plot of the density from SUMO data (View from the top)

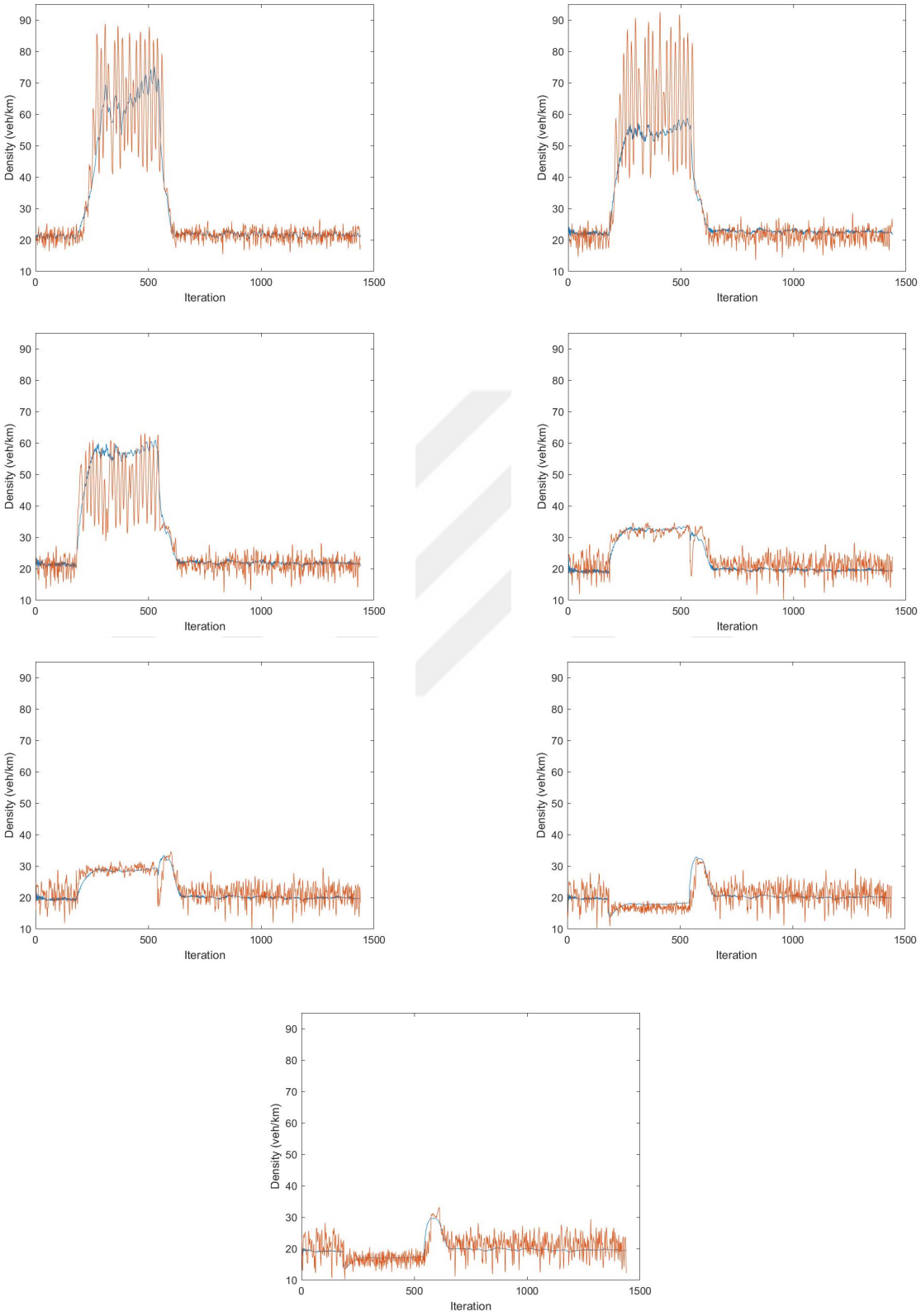

Figure 3.14. 2D plots of the density for segments 1 to 7 (starting at top left position). The blue curve is METANET's data and the orange curve is SUMO's data.

3.3.1.2. Velocity. On the first three segments, the velocity approximately matches the input data from SUMO. Oscillations are happening during the closed lane period, which is induced by the limits imposed to the density and velocity with v_{min}, v_{max} and ρ_{max}.

On the segments where the lane is closed, some differences occur. For segment 4, the shape from METANET is similar to SUMO, with less variations during the closed lane period. For segment 5, a rise in the velocity happens, but its variations have less fluctuations than in SUMO which is the case also in following segments. To explain it, it is worth to mention that, since SUMO is a microscopic and stochastic model variations can happen intensively, however, to a deterministic and macroscopic model, forcing the model to have those kind of oscillations can lead to over-fitting, considering that the model will mimic the noises along with signals or the global trend. This explanation is also valid for segment 6 and 7.

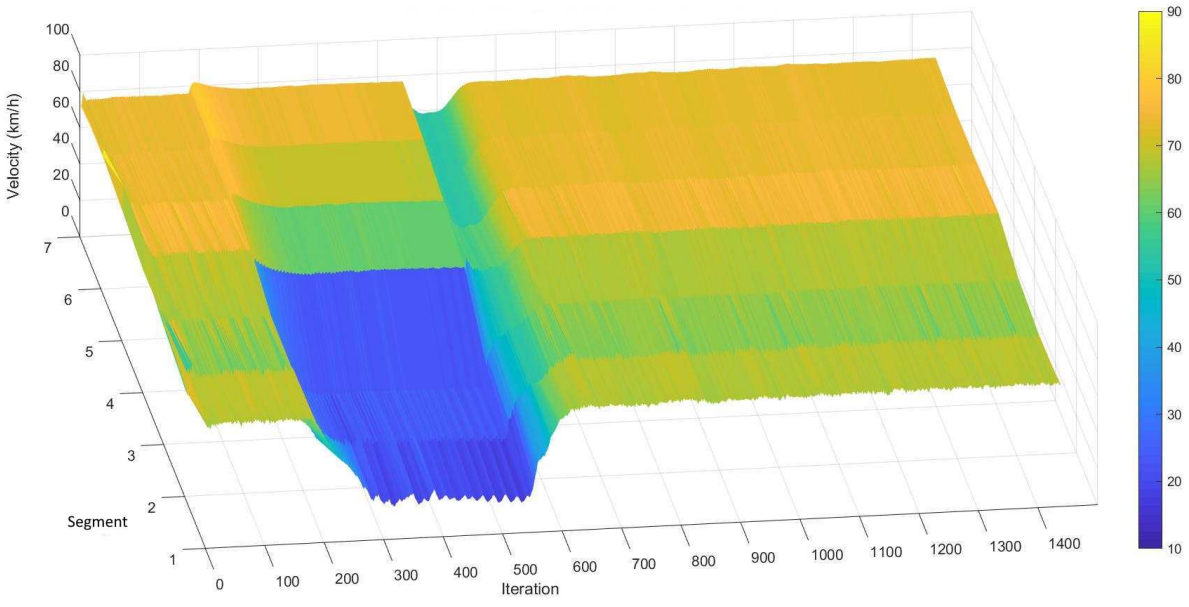

Figure 3.15. 3D plot of the Velocity from METANET data

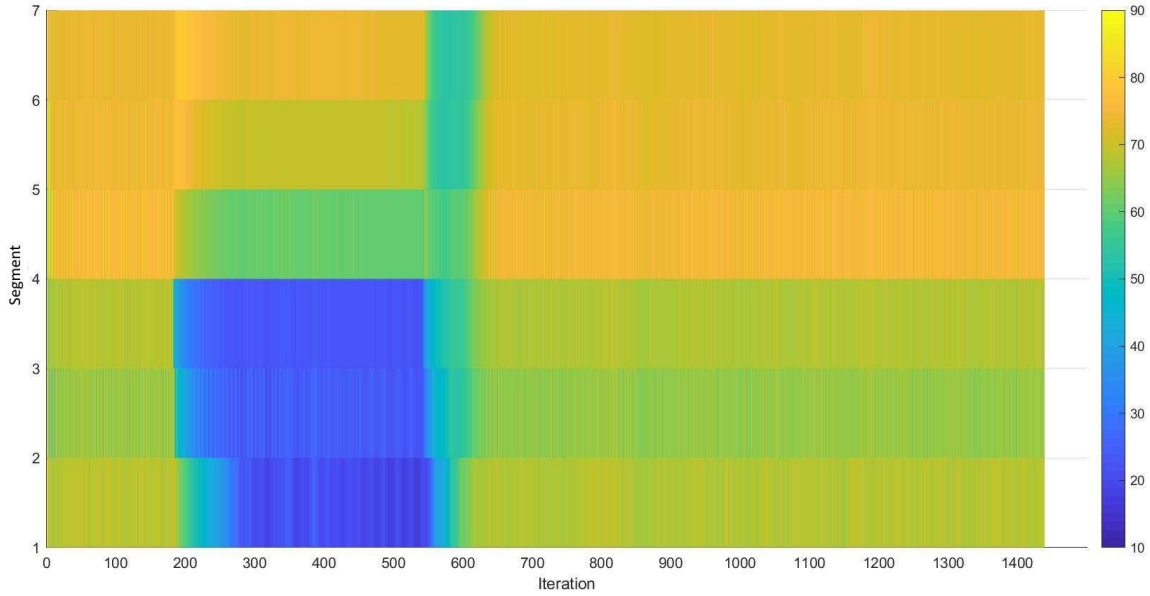

Figure 3.16. 3D plot of the Velocity from METANET data (View from the top)

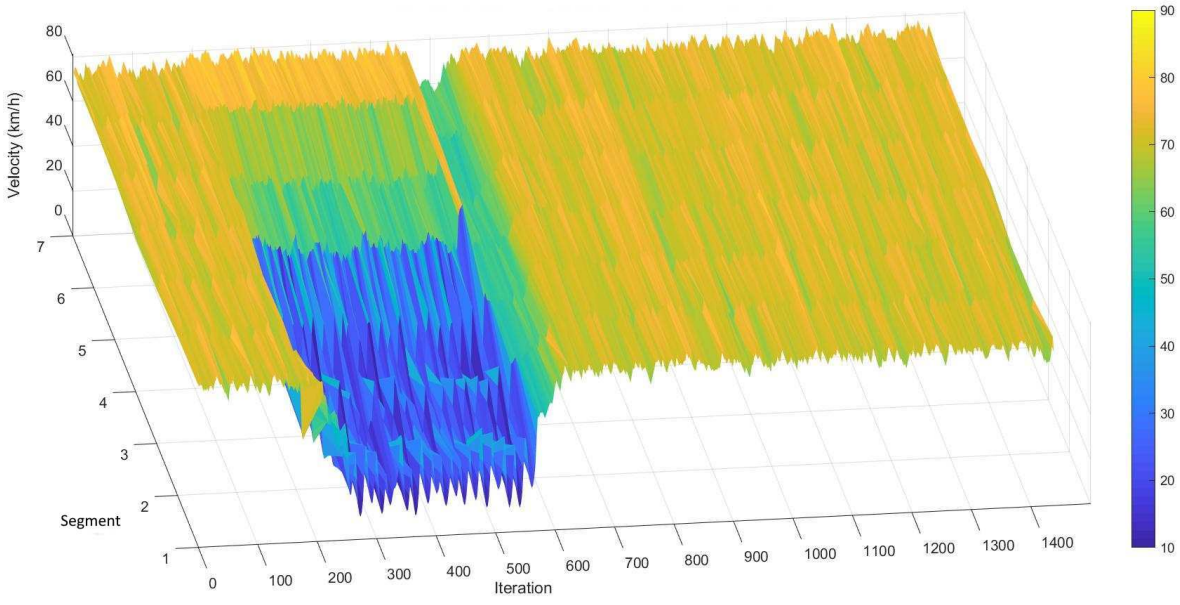

Figure 3.17. 3D plot of the Velocity from SUMO data

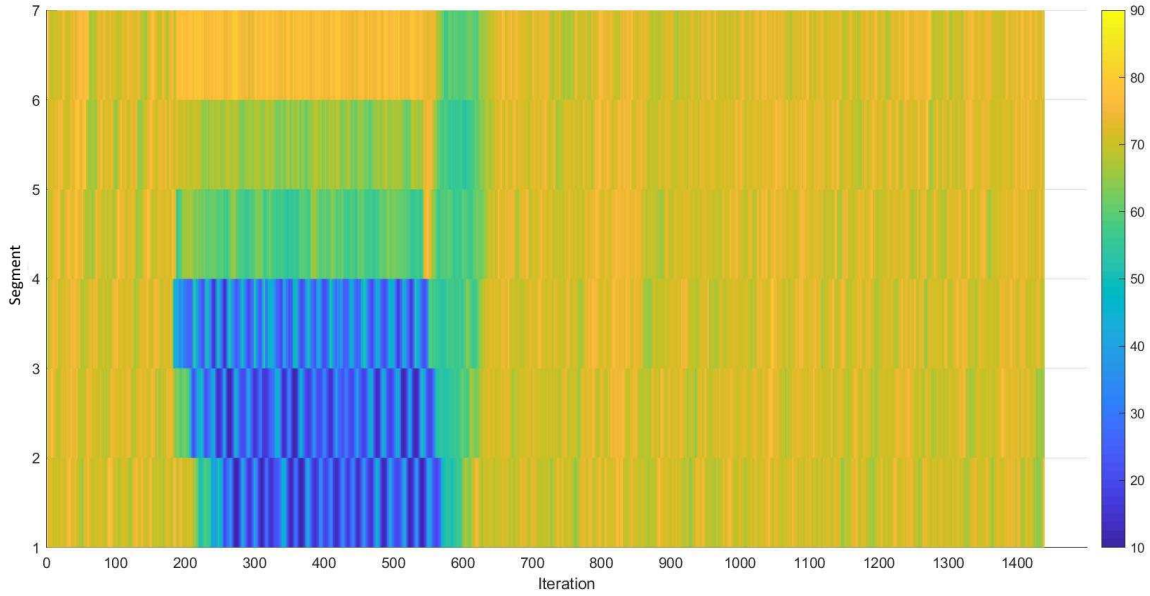

Figure 3.18. 3D plot of the Velocity from SUMO data (View from the top)

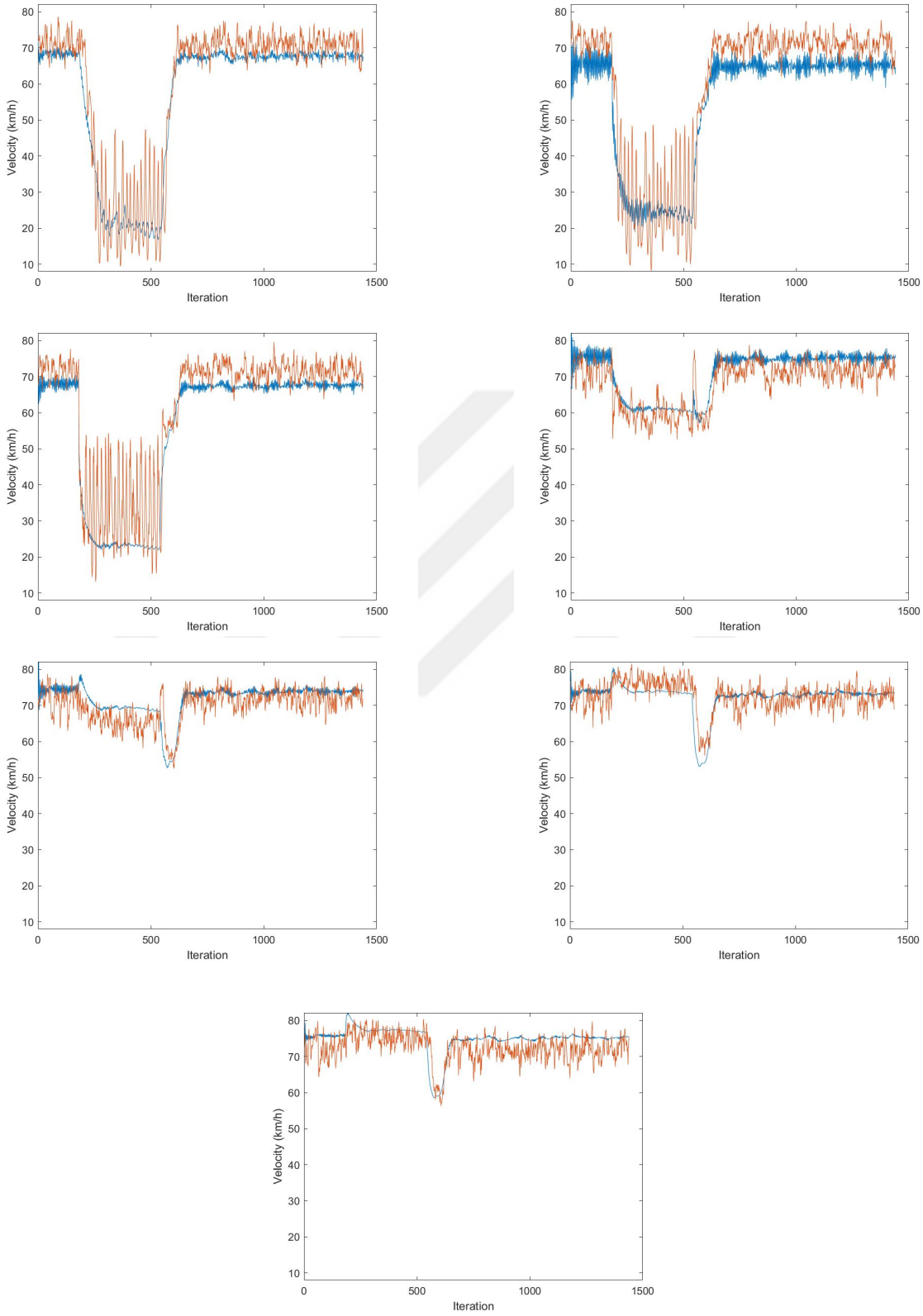

Figure 3.19. 2D plots of the velocity for segments 1 to 7 (starting at top left position). The blue curve is METANET's data and the orange curve is SUMO's data.

3.3.1.3. Flow. On all segments, the obtained flow from METANET matches almost exactly the flow from SUMO's data. Although in the first three segments, when the lane is closed, fluctuations are less in METANET model, the METANET model was able to mimic the pattern and the values that correspond to that period is in the mean range of the oscillations.

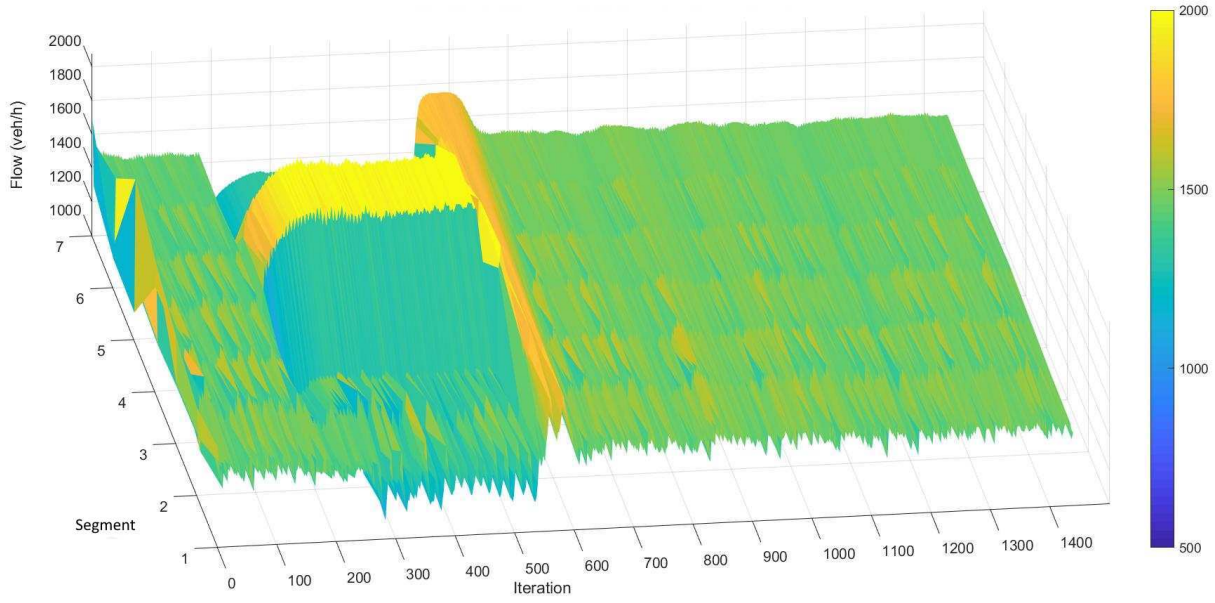

Figure 3.20. 3D plot of the Flow from METANET data

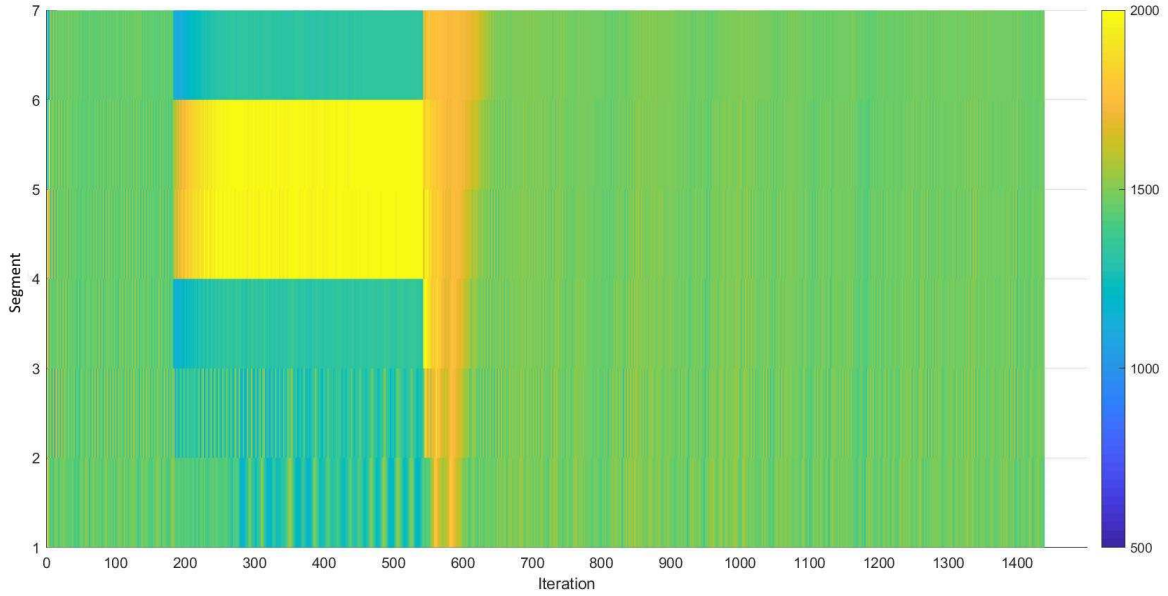

Figure 3.21. 3D plot of the Flow from METANET data (View from the top)

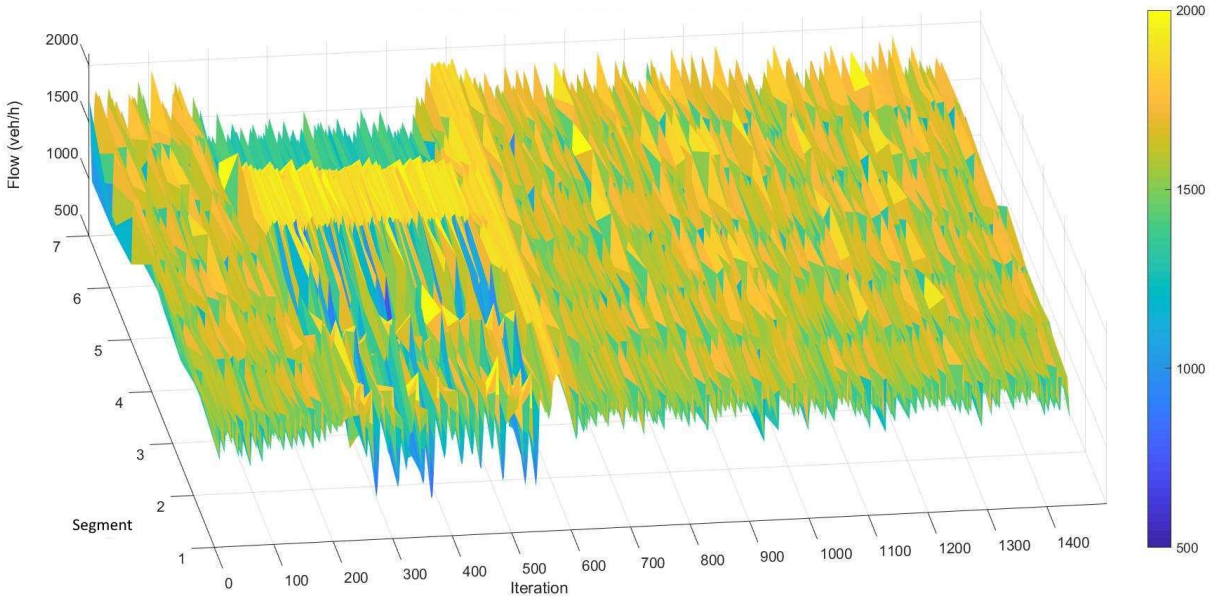

Figure 3.22. 3D plot of the Flow from SUMO data

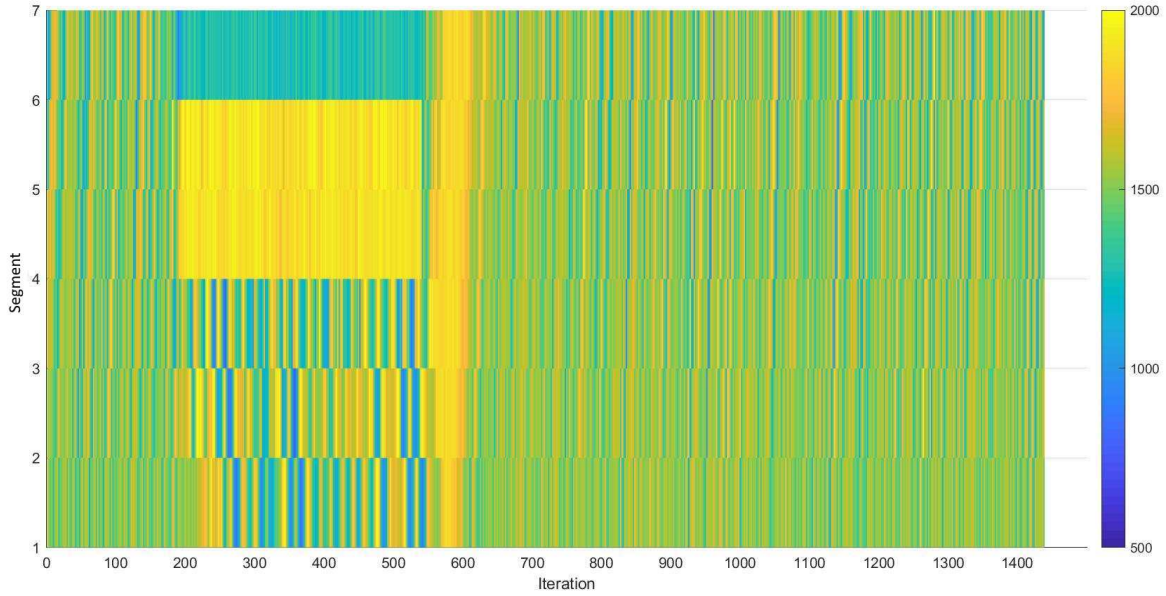

Figure 3.23. 3D plot of the Flow from SUMO data (View from the top)

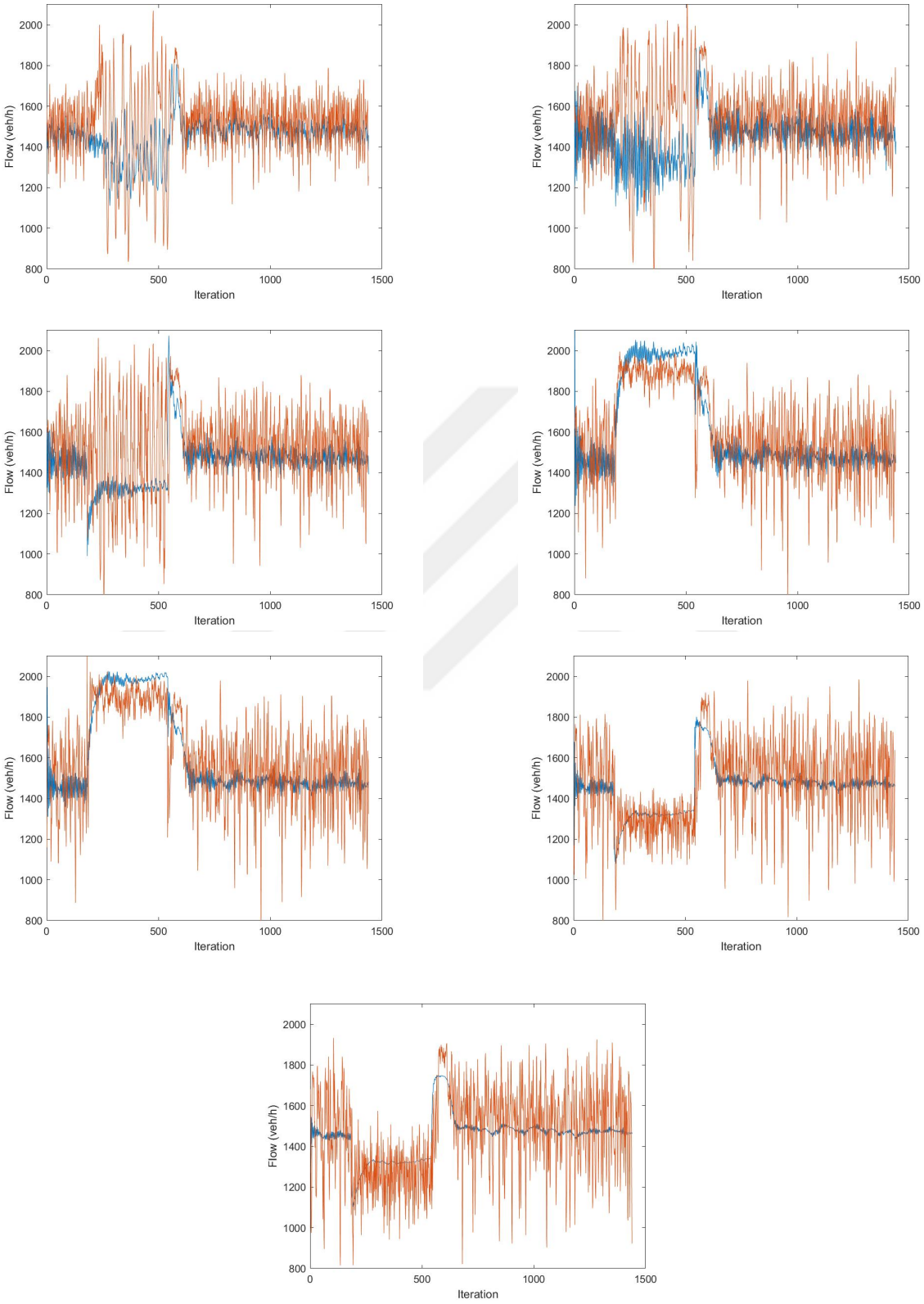

Figure 3.24. 2D plots of the flow for segments 1 to 7 (starting at top left position). The blue curve is METANET's data and the orange curve is SUMO's data.

3.3.1.4. Analysis. As it can be seen from the plots, a stop-and-go phenomenon has been occurred due to closing one lane. A backward shockwave can be seen. Segment one, two and three are facing congestion and they are in an unstable traffic situation since the density is greater than their critical density. Therefore, any perturbation regardless of its size will lead to a congestion. The parameter τ in segment three is almost double of the other segments, which means vehicles in segment three need more time to adjust their speed.

Speeds in METANET are restricted by upper and lower boundaries. In segments one to three, due to a backward shockwave, a drastic speed reduction is seen. The lowest values can be seen in segment number three which is the closest segment to the incident.

Segments one to three have reached and passed their capacity, however, remaining segments are in a stable situation and have not reach their capacity. The calibrated model fits perfectly with the SUMO model and it is validated by the different set of data from SUMO.

3.3.2. Second Scenario: One-Lane Closed Scenario with a Flow of 3000 veh/h

The second scenario is an intermediate traffic follow situation with a one-lane closing incident for one hour during the whole 4 hours simulation. Segments 4 and 5 have their left-most lane closed. The lane is closed after 30 minutes of simulation (without counting the warm up time), and are re-opened after 90 minutes of simulation. This lane closing is achieved using the rerouter feature of SUMO. A rerouter will redirect the vehicles on the best possible path, thus, it is possible to close a lane for a certain period of time during which all the vehicles will be redirected to the opened lane. An example for this scenario in real life can be a road work or a situation which one car can stop and pull over.

Table 3.5. Best obtained parameters for the 1-lane closed scenario after the per segment calibration.

Segment	ρ_{cr}	ρ_{max}	v_{min}	α	τ	ν	κ	ϕ	v_{max}
	$veh/km/lane$	$veh/km/lane$	km/h	$-$	$hours$	km^2/h	$veh/km/lane$	$-$	km/h
Before calibration	37.5	111	4	2.15	0.0084	60	100	2.515	95
Segment 1	50	110.75	4.6956	4	0.0083	5	200	3	90
Segment 2	50	118.75	6.6956	1.125	0.0083	93.5	156	3	90
Segment 3	50	132	4.6956	1.5	0.0083	5	14.250	1	90
Segment 4	50	90.750	4.6956	1	0.0083	120	8.50	3	90
Segment 5	50	120.75	4.6956	1.375	0.0083	43.5	125	3	90
Segment 6	50	120.75	4.6956	1.25	0.0083	86.5	4	3	90
Segment 7	50	120.75	4.6956	1.25	0.0083	5.25	200	3	90

After the calibration, the obtained objective function value was 0.0596763. After running the objective function again with a new dataset from SUMO and with the parameters listed above, the objective function value was 0.06163621. This value has a difference of +3.284% compared to the calibration phase. The plots obtained are similar to the ones from the calibration phase. Therefore, the model is validated.

Table 3.6. Comparison of the objective function values obtained in the calibration and validation phases for the 1-lane closed scenario.

DataSet	Objective Value	Difference (%)
Calibration	0.0596763	-
Validation	0.06163621	+3.284

3.3.2.1. Density. The files from SUMO for this scenario have a lot of noise in it. Running simulations more than 5 times does not give any improvement to the noise. Therefore, METANET cannot reproduce all the small variations happening in SUMO exactly. For segment 3, SUMO data show a bump just after the closure of a lane in a segment that is not reproduced in METANET. This bump is an issue from SUMO when closing a lane in both segments: some vehicles are totally stopped just in front of the closed lane. As they are present on the segment and have a speed of 0, they create an impact on the velocity of the segment, therefore, the density is also affected by the mentioned issue using the velocity in its formula.

Beside the issue on segment 3 and the noise from the simulated data, METANET's data matches perfectly with the SUMO data.

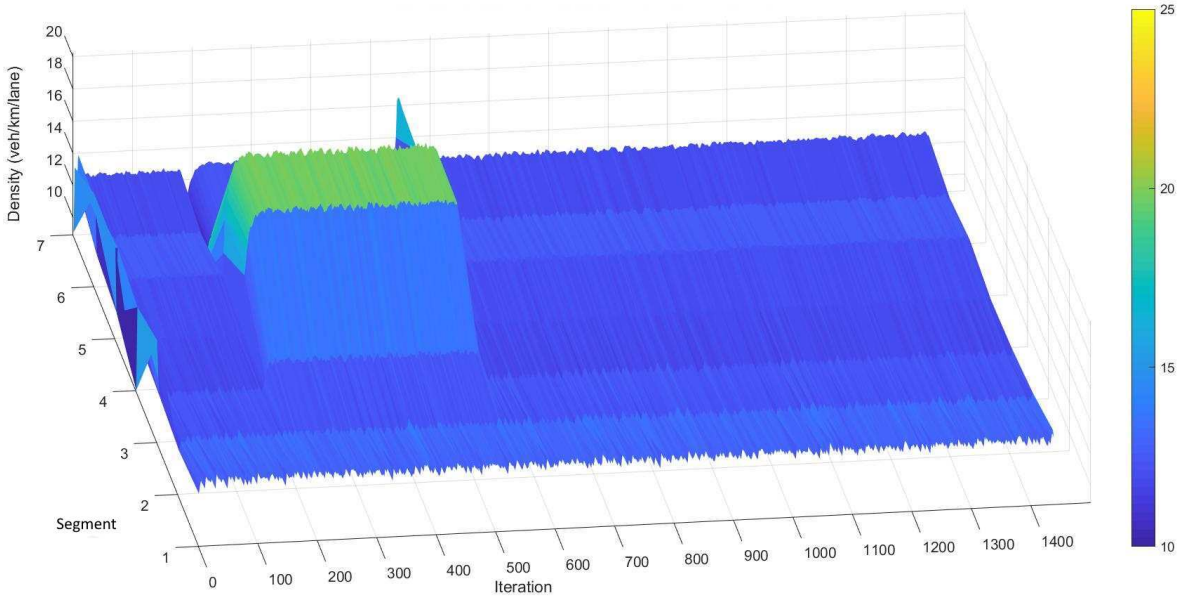

Figure 3.25. 3D plot of the density from METANET data

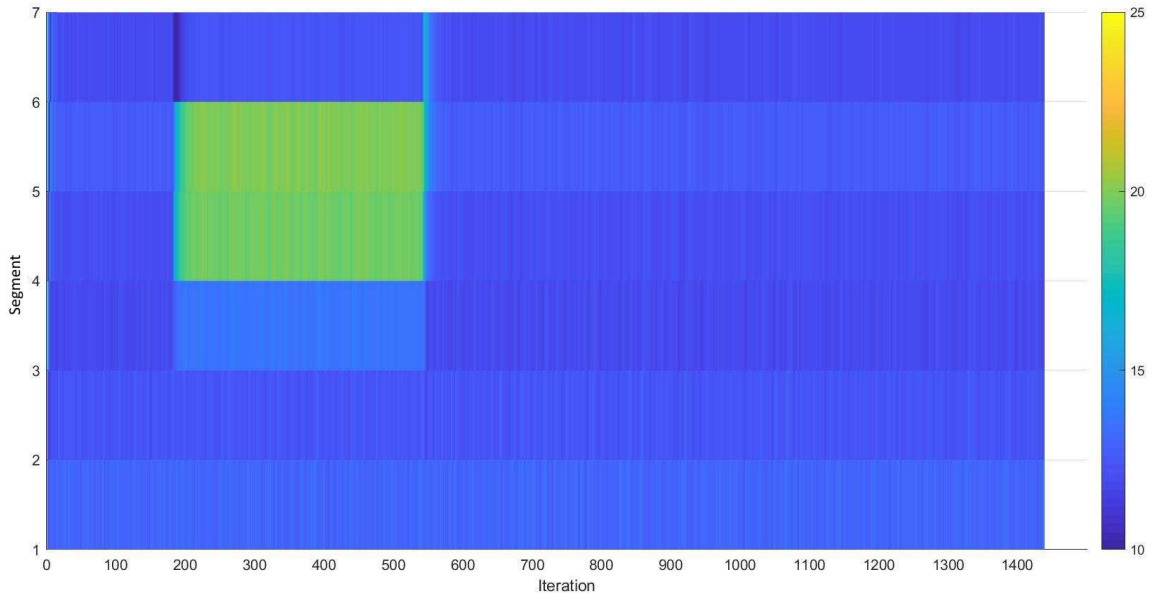

Figure 3.26. 3D plot of the density from METANET data (View from the top)

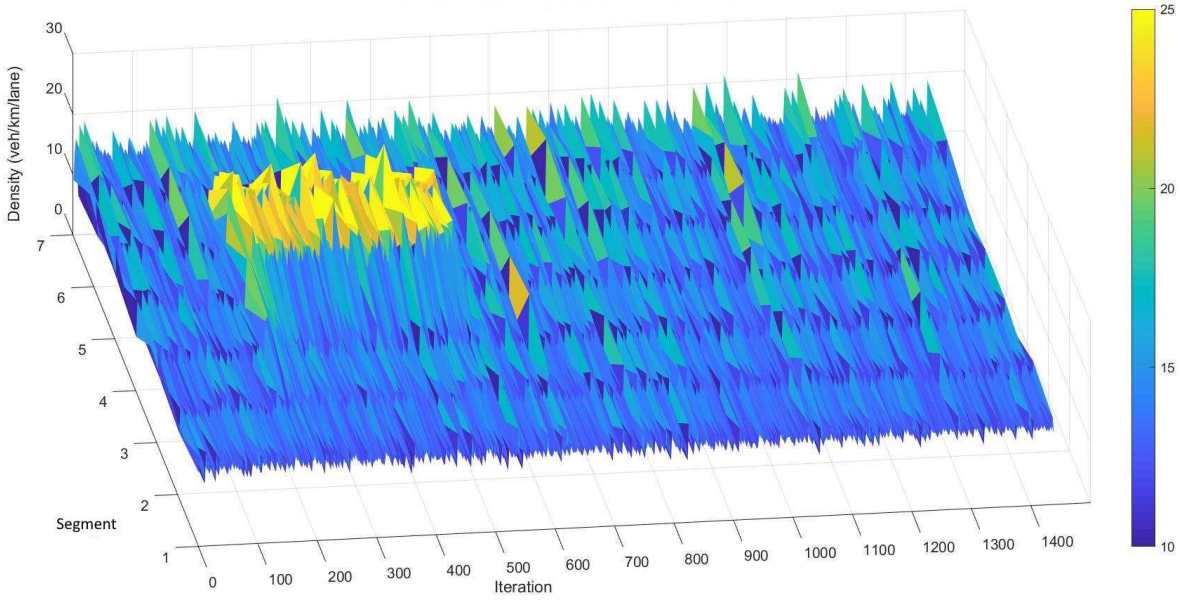

Figure 3.27. 3D plot of the density from SUMO data

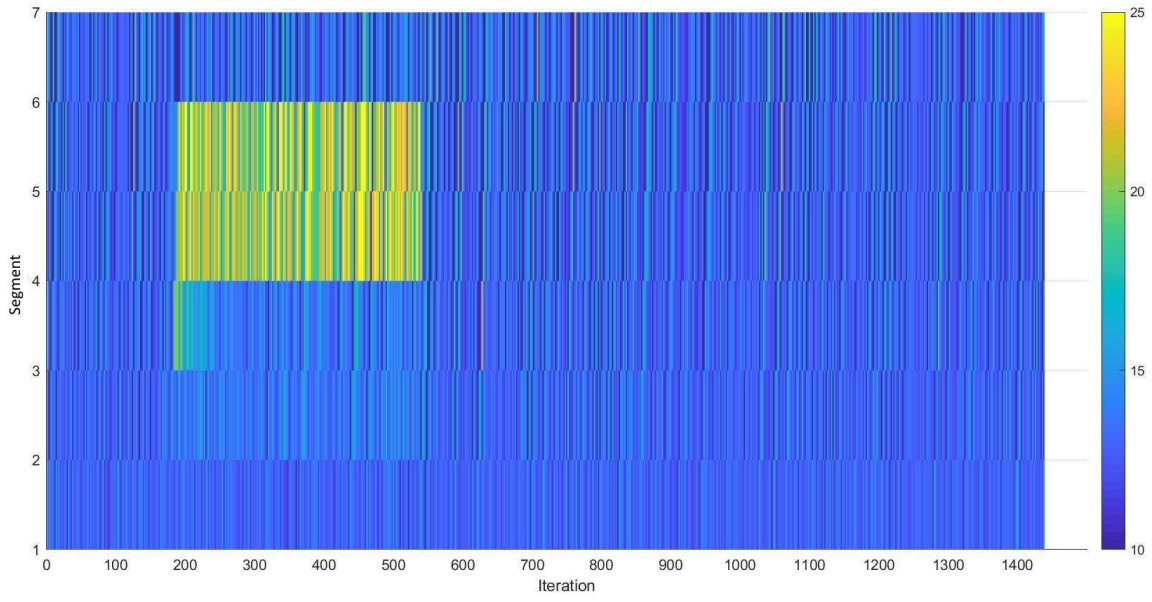

Figure 3.28. 3D plot of the density from SUMO data (View from the top)

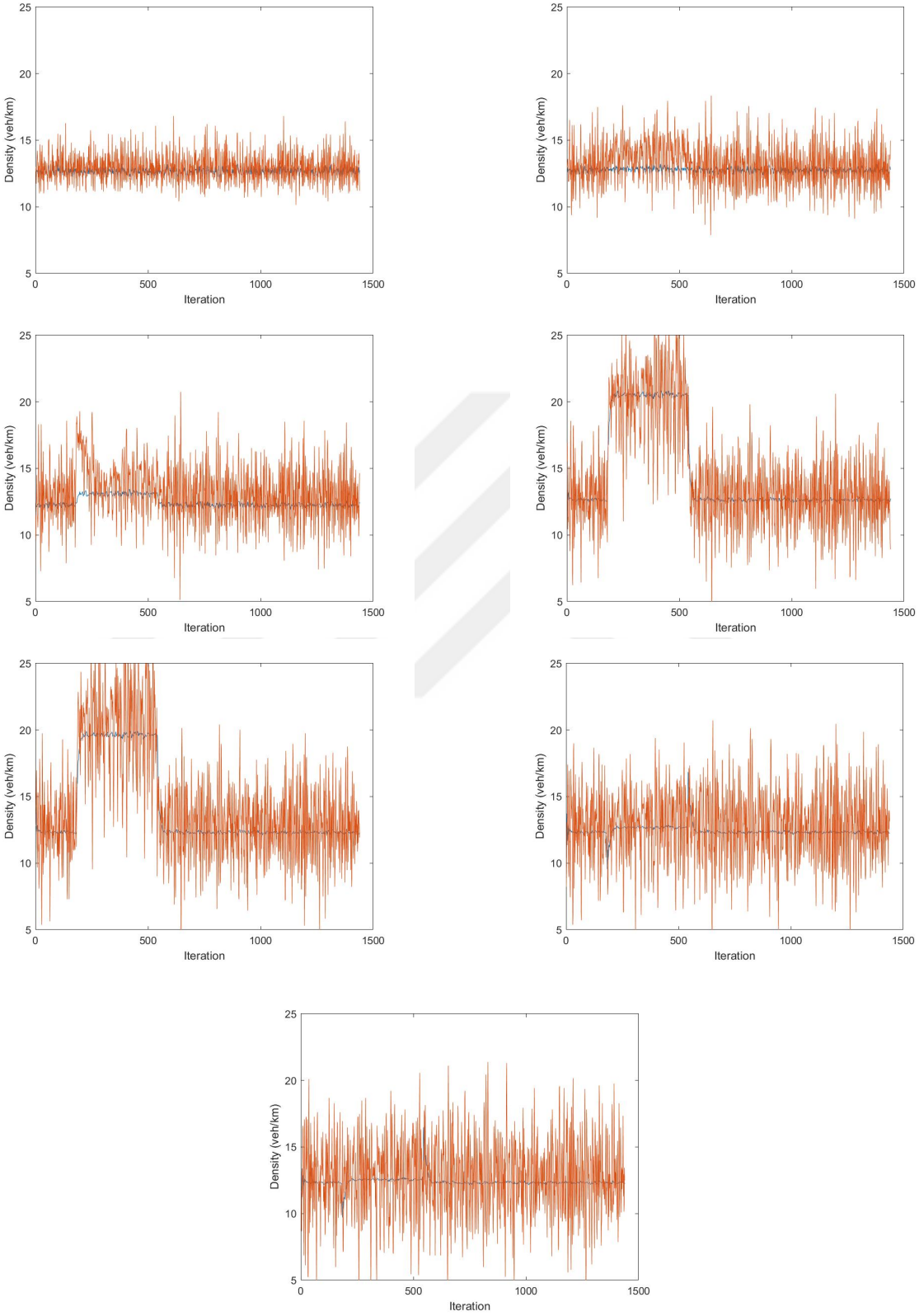

Figure 3.29. 2D plots of the density for segments 1 to 7 (starting at top left position). The blue curve is METANET's data and the orange curve is SUMO's data.

3.3.2.2. Velocity.

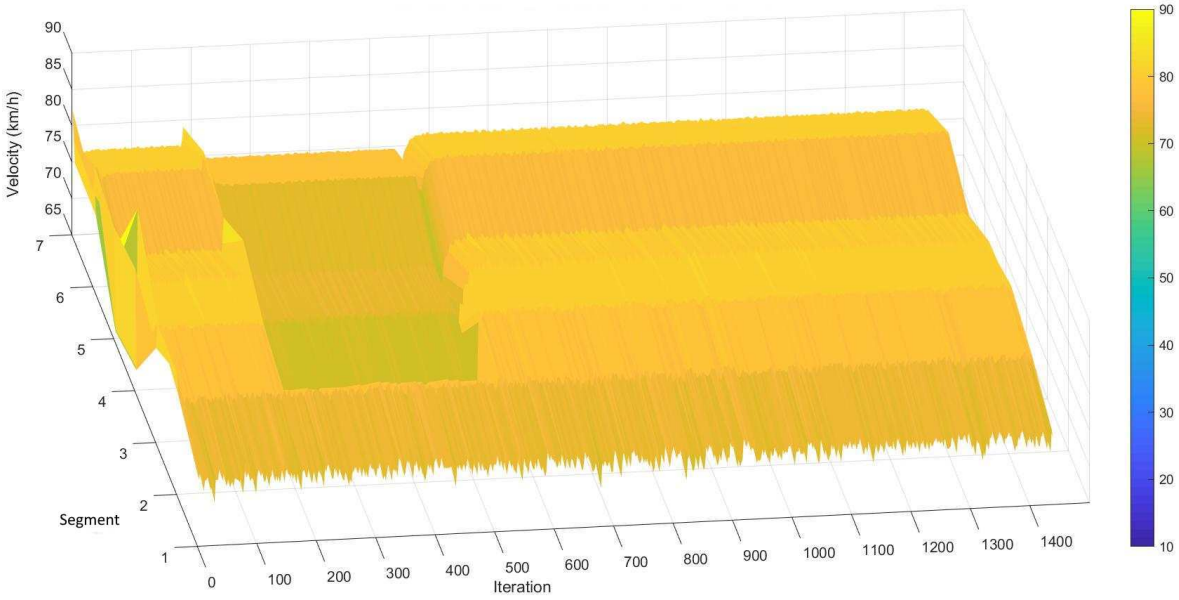

Figure 3.30. 3D plot of the Velocity from METANET data

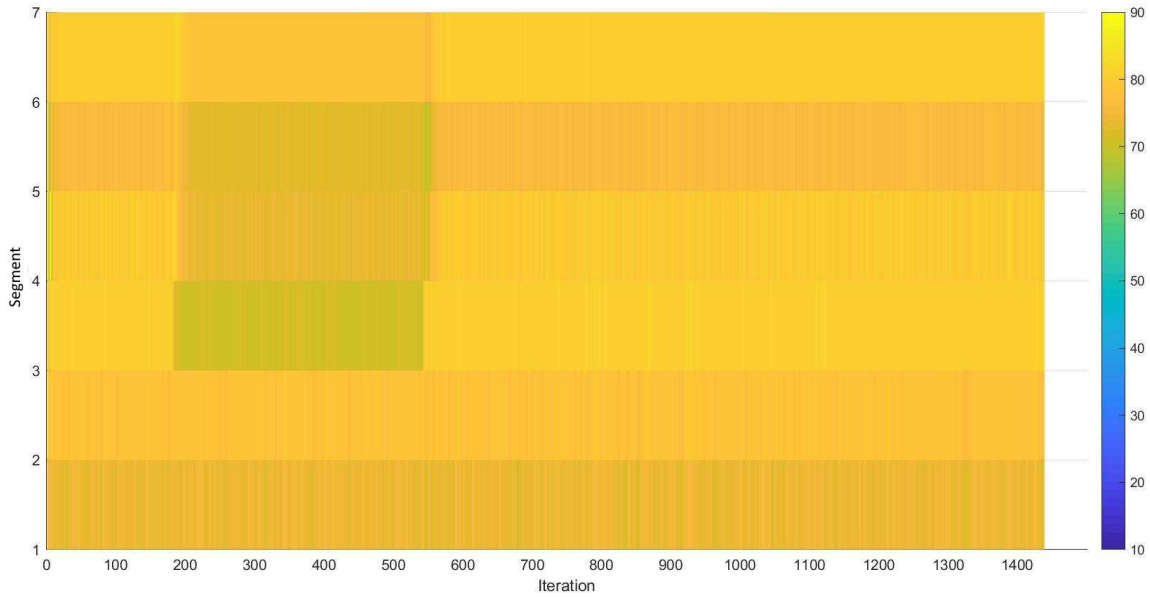

Figure 3.31. 3D plot of the Velocity from METANET data (View from the top)

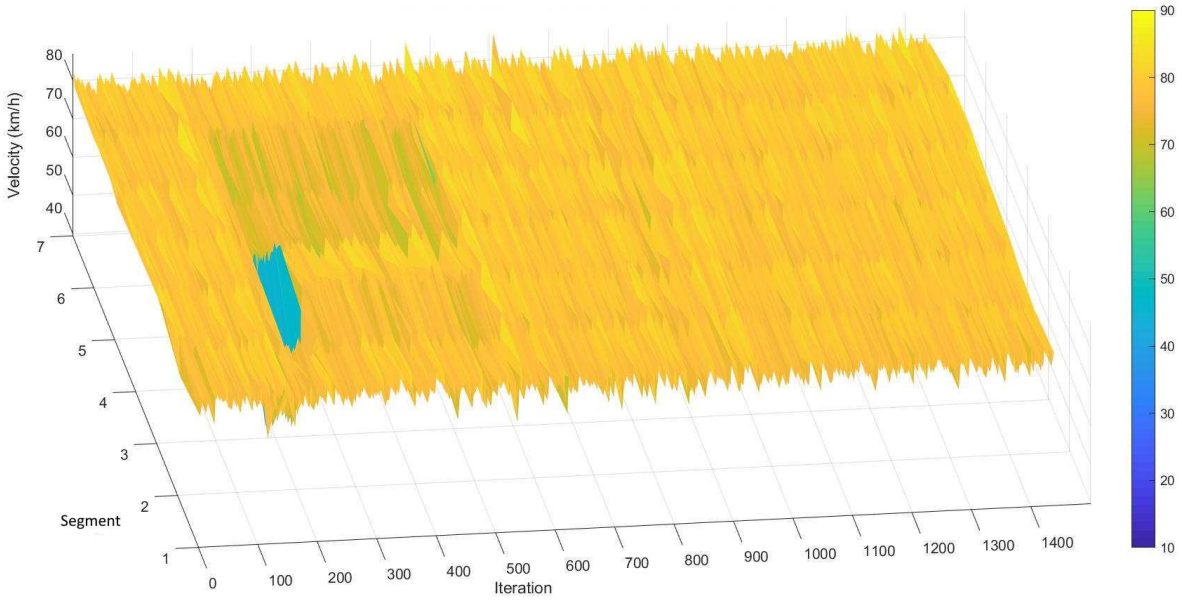

Figure 3.32. 3D plot of the Velocity from SUMO data

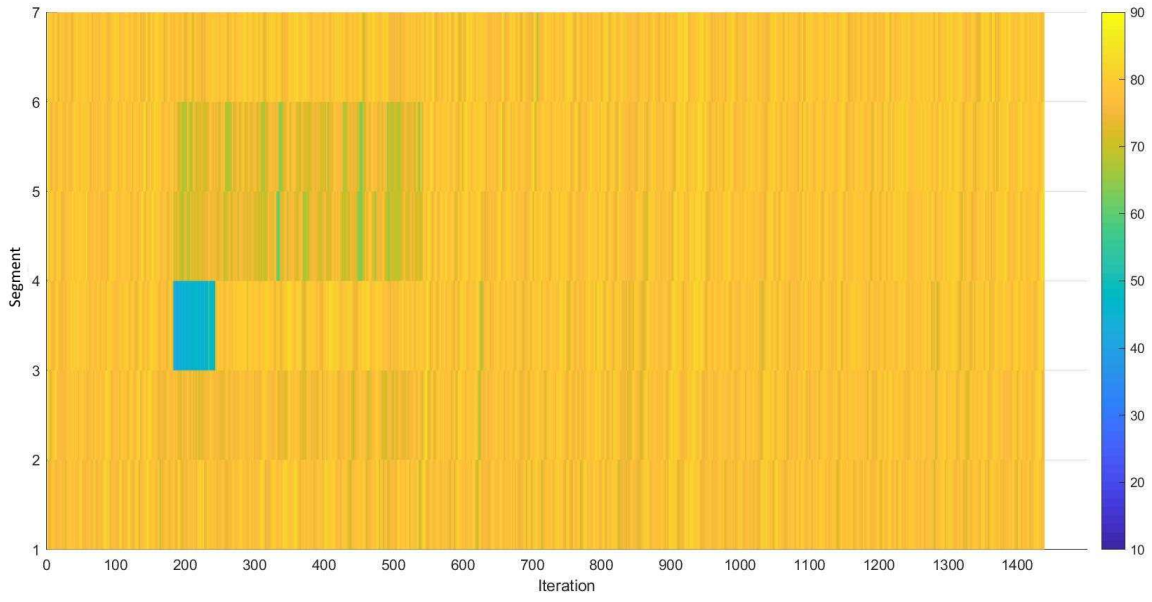

Figure 3.33. 3D plot of the Velocity from SUMO data (View from the top)

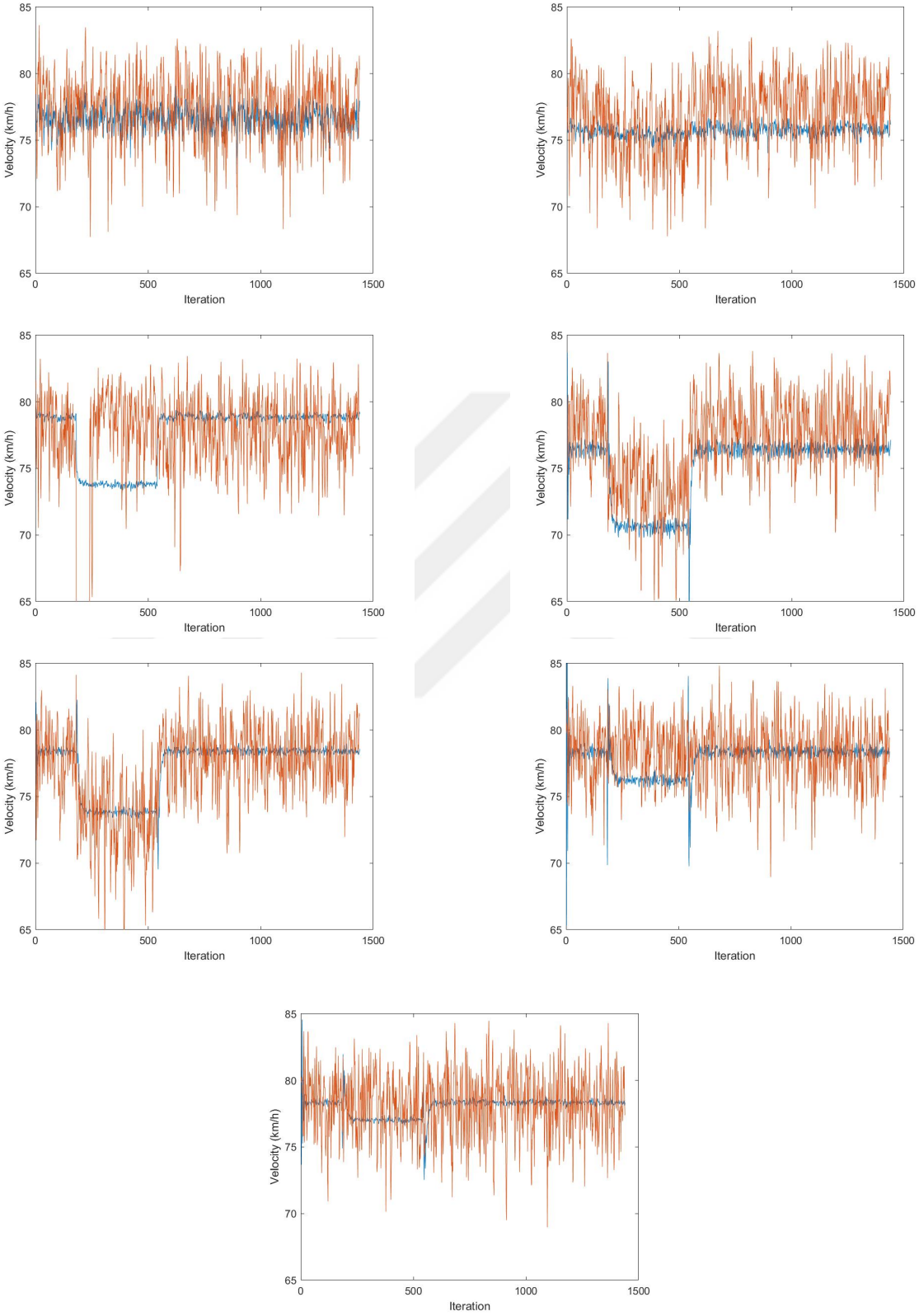

Figure 3.34. 2D plots of the velocity for segments 1 to 7 (starting at top left position). The blue curve is METANET's data and the orange curve is SUMO's data.

3.3.2.3. Flow.

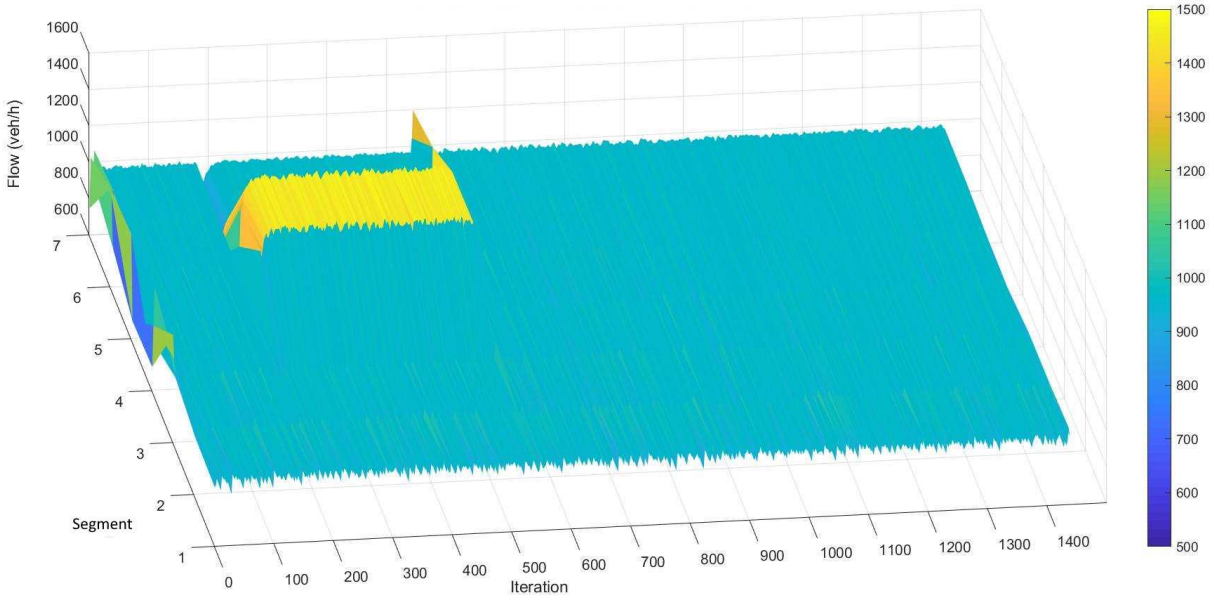

Figure 3.35. 3D plot of the Flow from METANET data

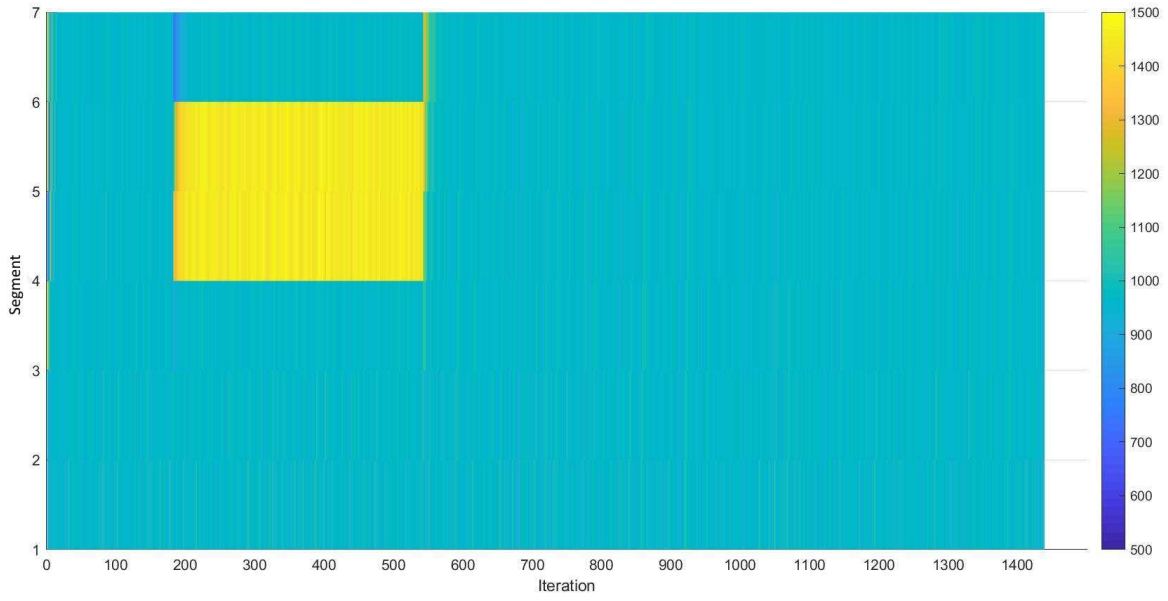

Figure 3.36. 3D plot of the Flow from METANET data (View from the top)

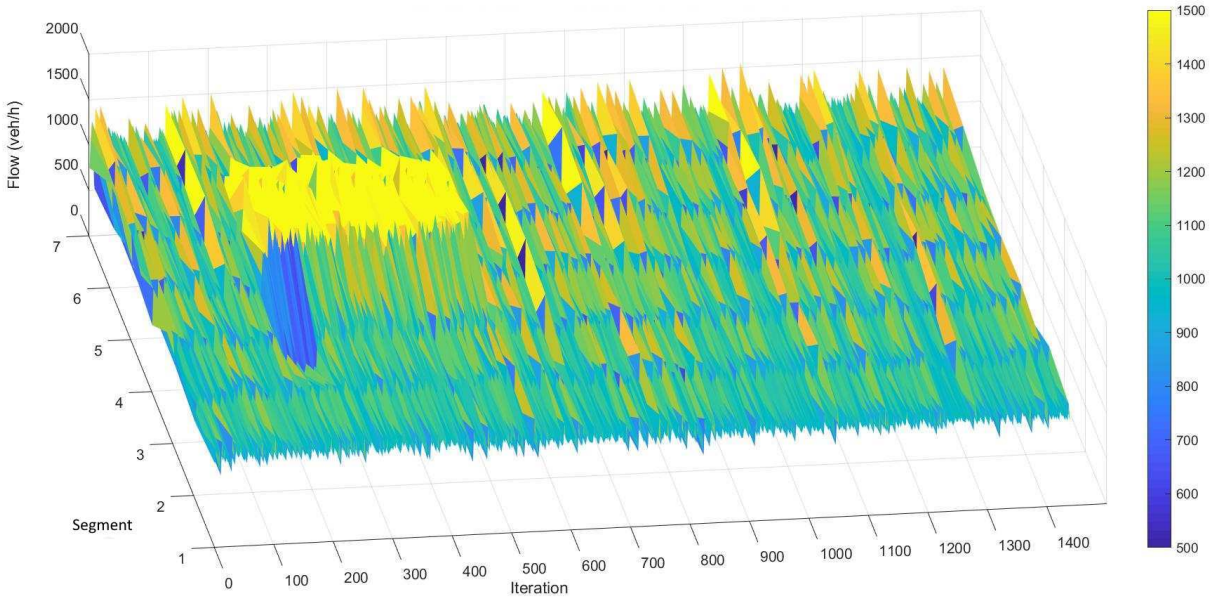

Figure 3.37. 3D plot of the Flow from SUMO data

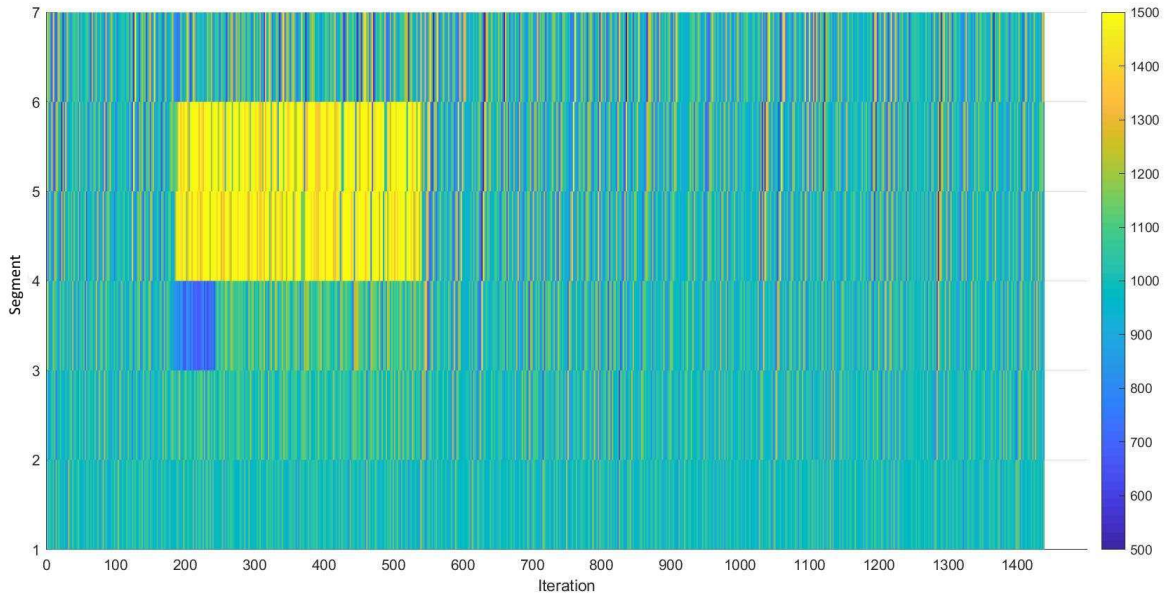

Figure 3.38. 3D plot of the Flow from SUMO data (View from the top)

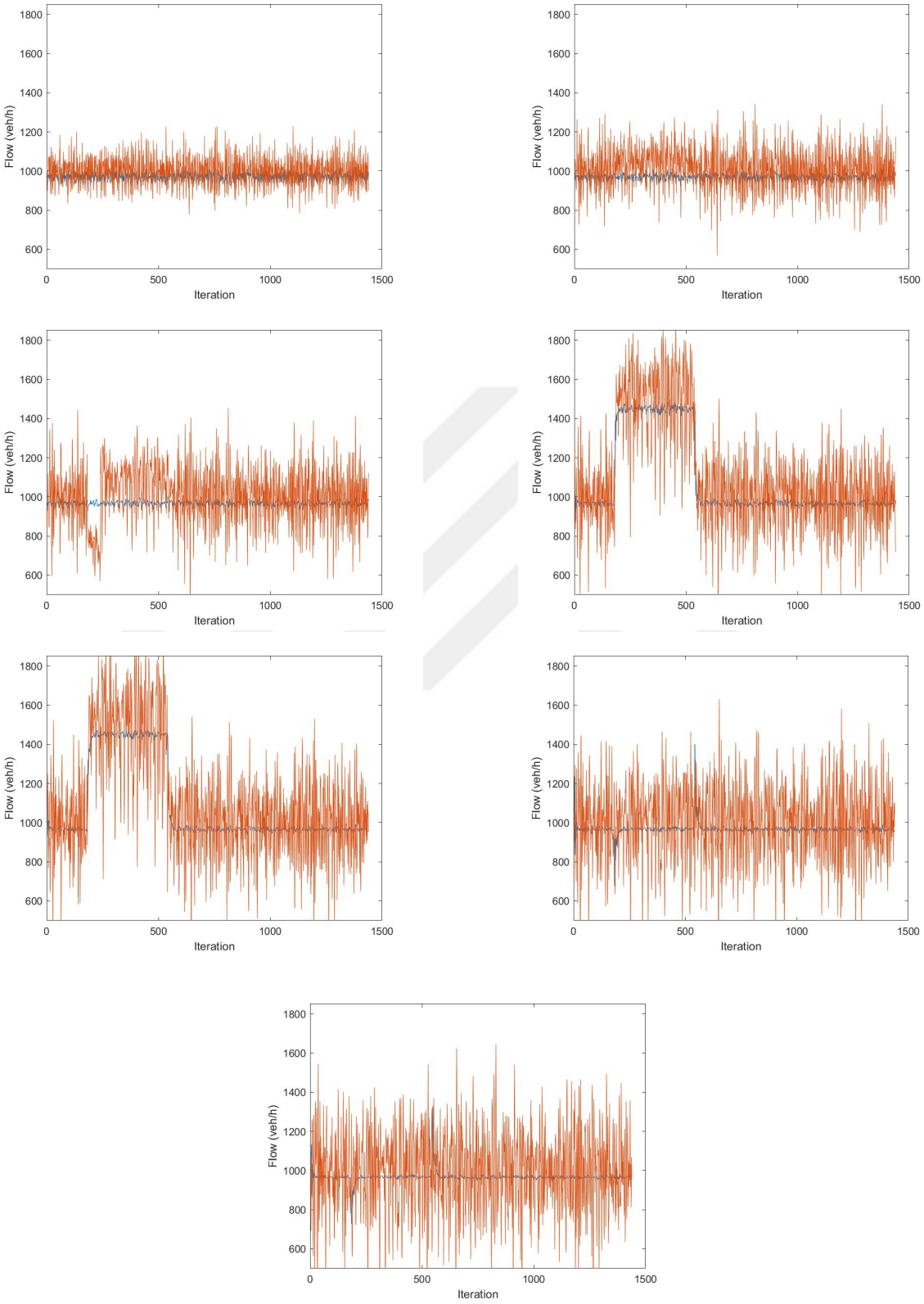

Figure 3.39. 2D plots of the flow for segments 1 to 7 (starting at top left position). The blue curve is METANET's data and the orange curve is SUMO's data.

3.3.2.4. Analysis. In this scenario, as it can be also seen from Figures 3.25 to 3.39 an intermediate flow stream is occurred, therefore, a very light congestion has found in the results. In segments 4 and 5, density is greater than the others, which means due to the low inflow, other segments have not been affected by the incident of lane closure. Sudden speed reduction in segment three in SUMO data is due to some stopped vehicles when the lane started to be closed. A rise in the density of segments 4 and 5 is happening when the lane is closed. The incoming vehicles to segments 4 reduce their speed when approaching the segment which corresponds to a normal behaviour from the drivers. In SUMO data, a sudden speed reduction occurs on the third segment when the lane is closed. This phenomenon is due to some vehicles that are stopped instantly when the lane is closed, and this value would not happen in real life. Besides segments 4 and 5, the other segments have not been affected by the incident. The calibrated model fits perfectly with the SUMO model and it is validated by the different set of data from SUMO.

3.3.3. Third Scenario: Free Flow Scenario

The third scenario is a non-congested traffic situation. A continuous flow of 3000 vehicles per hour is generated at the entrance of the road stretch, and no incident is happening during the whole simulation. The flow, as the density and mean speed, should stay approximately the same through all segments during all the time since the road is experiencing the free flow stream without any bottlenecks or incidents.

Table 3.7. Best obtained parameters for the free flow scenario after the per segment calibration.

Segment	ρ_{cr}	ρ_{max}	v_{min}	α	τ	ν	κ	ϕ	v_{max}
	$veh/km/lane$	$veh/km/lane$	km/h	$-$	$hours$	km^2/h	$veh/km/lane$	$-$	km/h
Before calibration	37.5	111	4	2.15	0.0084	60	100	2.515	95
Segment 1	37.5	111	4	2.05	0.0083	60	100	2.5	95
Segment 2	49.5	111	4	3.05	0.0083	5	200	2.5	95
Segment 2	49.5	111	4	1.55	0.0083	70	21	2.5	95
Segment 3	49.5	111	4	1.05	0.0083	54	14	2.5	95
Segment 4	49.5	111	4	2.55	0.0083	119	6	2.5	95
Segment 5	49.5	111	4	1.05	0.0083	107	56	2.5	95
Segment 6	49.5	111	4	2.05	0.0083	57	4	2.5	90
Segment 7	49.5	111	4	1.05	0.0083	5	200	2.5	95

The first dataset used for the calibration that lead to the best set of parameters stated above has an objective function value of 0.052901277. When using another set of data, also composed of 5 simulation runs, the obtained result is 0.054554922, which represents a difference of 3.125%. The values are in the same range and the obtained plots do not present differences. Therefore, the validation is approved and our model is validated.

Table 3.8. Comparison of the objective function values obtained in the calibration and validation phases for the free flow scenario.

DataSet	Objective Value	Difference (%)
Calibration	0.052901277	-
Validation	0.054554922	+3.125

3.3.3.1. Density. For all segments, the density and flow correspond to the ones in SUMO excluding the noise. However, for the velocity, gaps can be observed on segments 1, 3, 5 and 7: The shapes are the same as SUMO, but the mean values are shifted about some kilometers per hour.

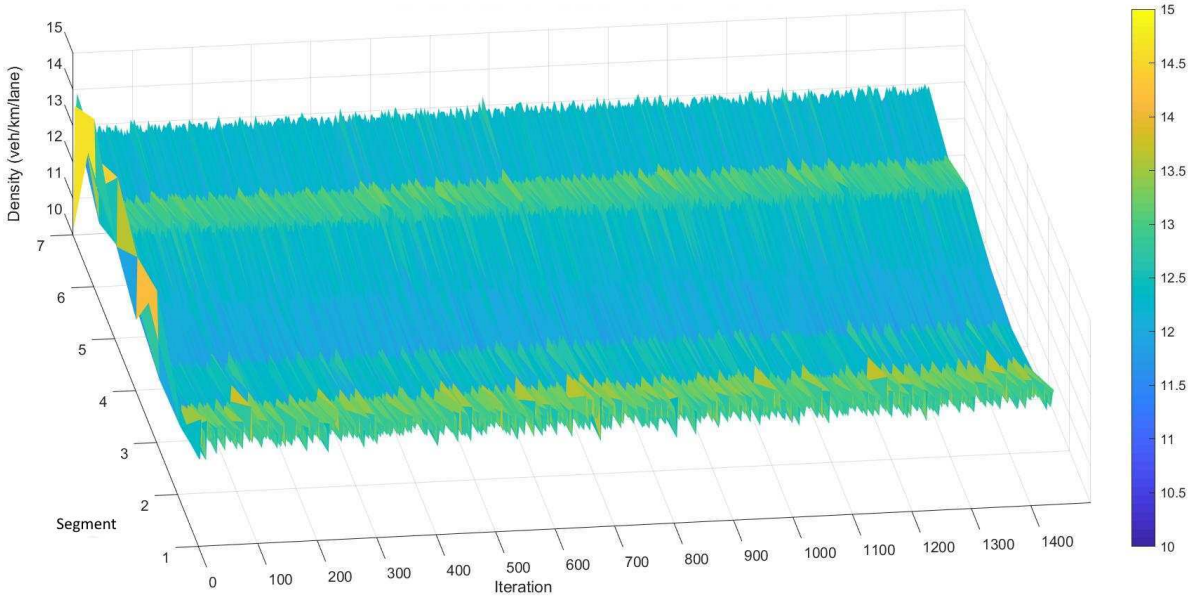

Figure 3.40. 3D plot of the density from METANET data

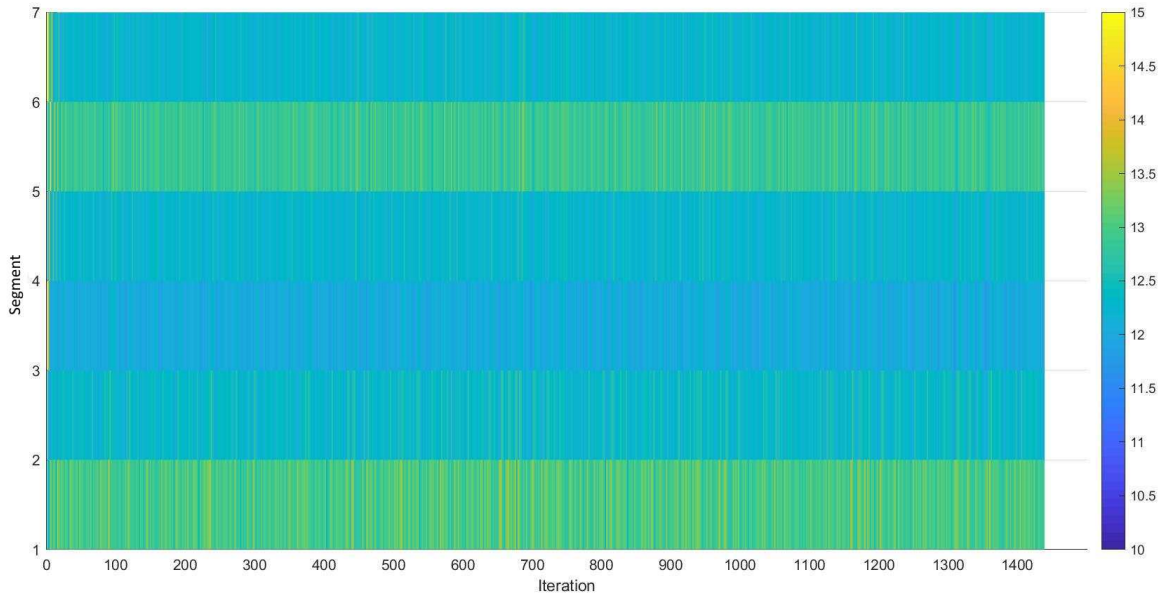

Figure 3.41. 3D plot of the density from METANET data (View from the top)

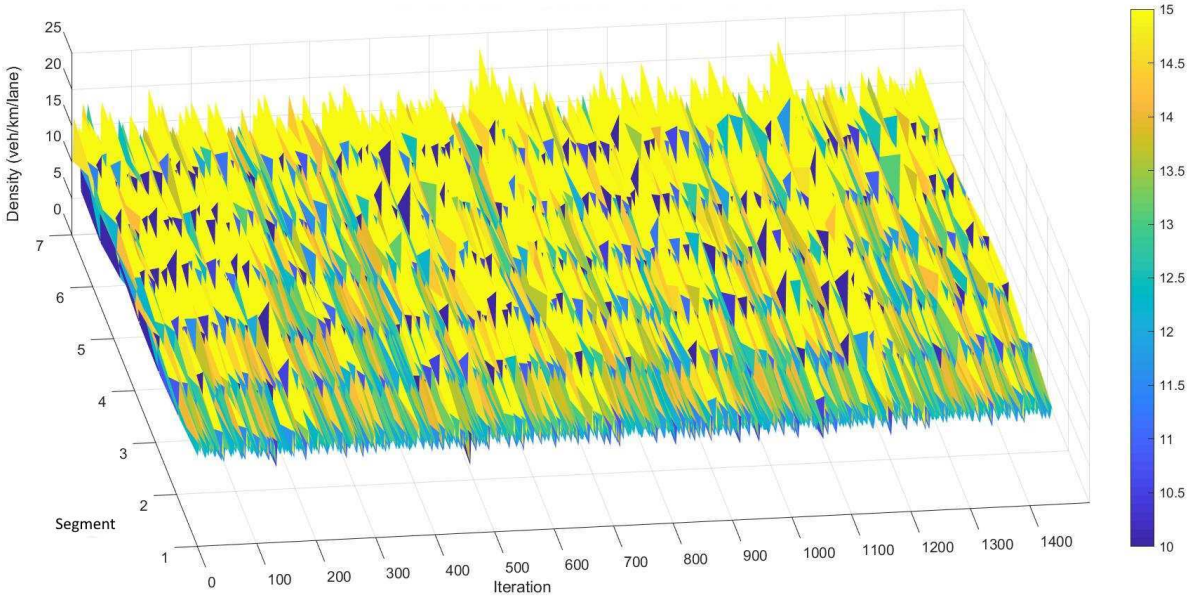

Figure 3.42. 3D plot of the density from SUMO data

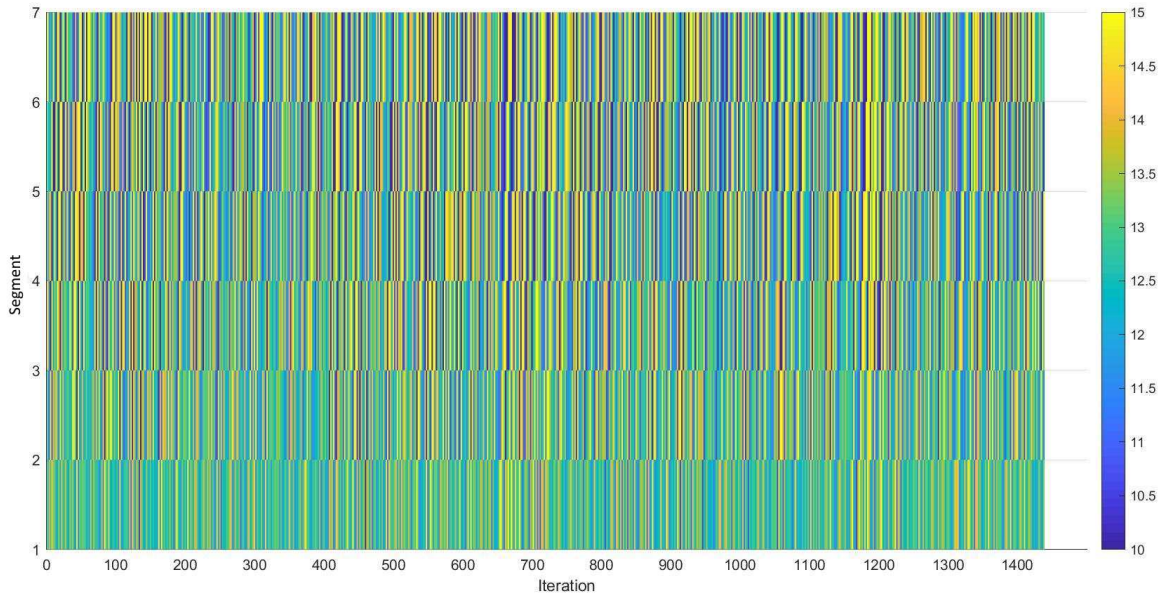

Figure 3.43. 3D plot of the density from SUMO data (View from the top)

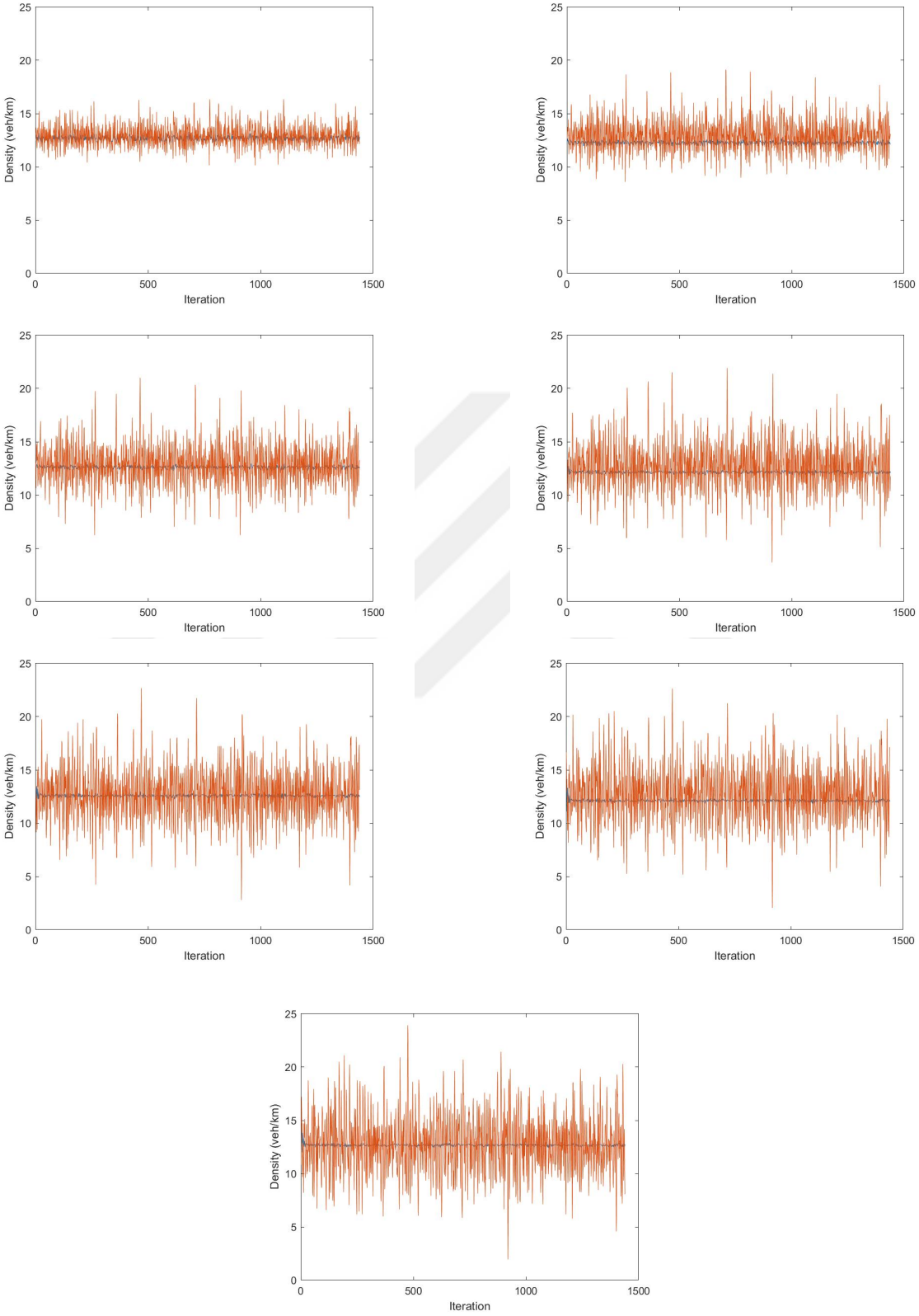

Figure 3.44. 2D plots of the density for segments 1 to 7 (starting at top left position). The blue curve is METANET's data and the orange curve is SUMO's data.

3.3.3.2. Velocity.

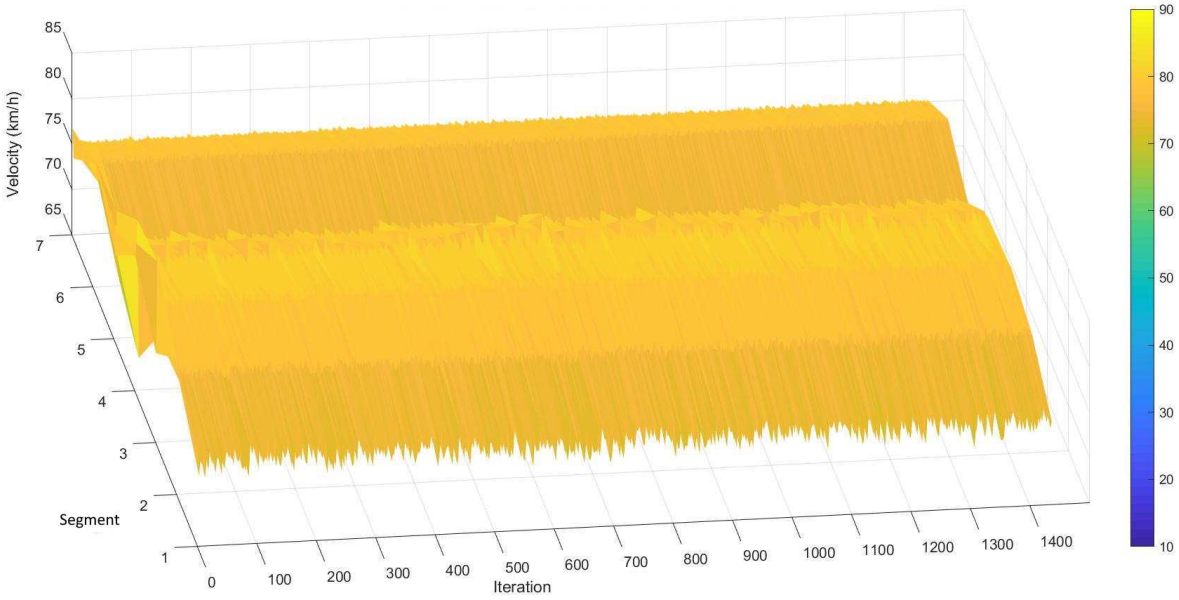

Figure 3.45. 3D plot of the Velocity from METANET data

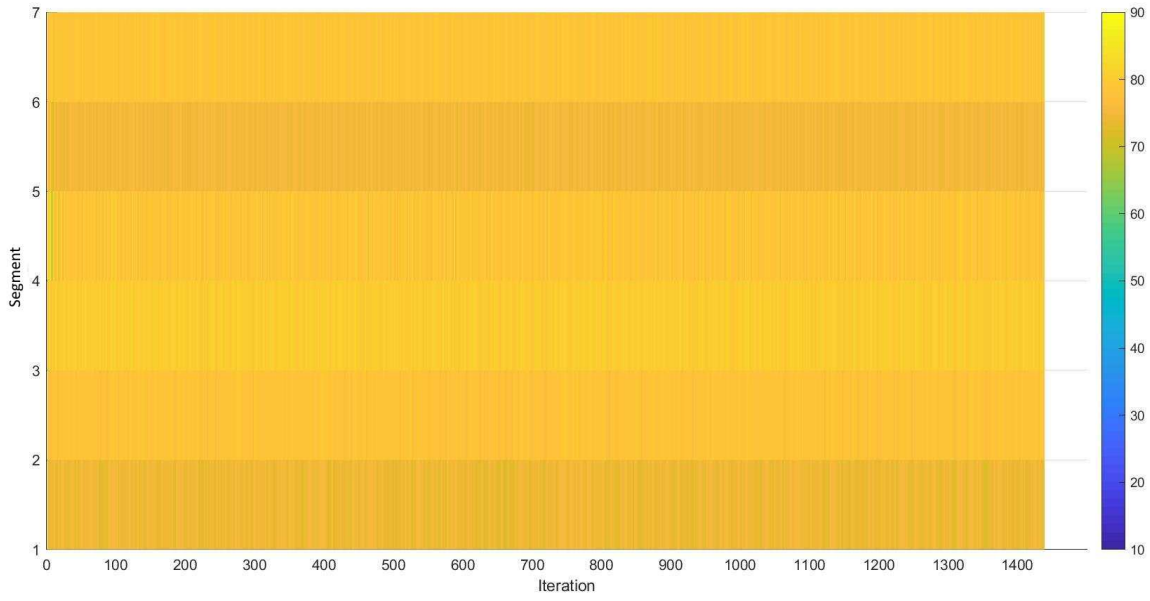

Figure 3.46. 3D plot of the Velocity from METANET data (View from the top)

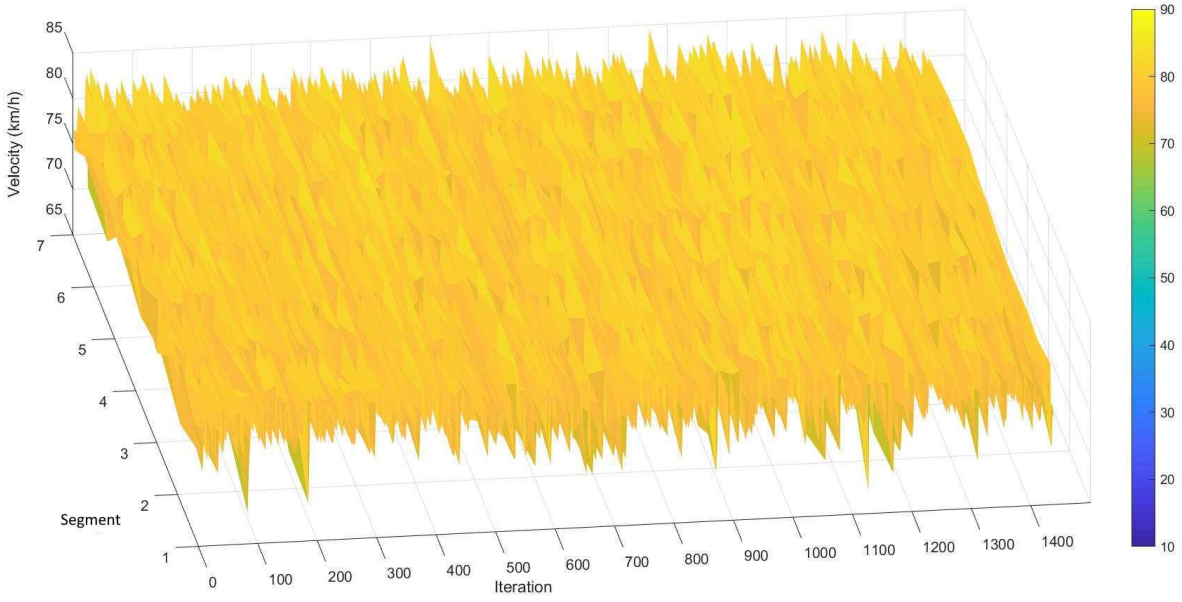

Figure 3.47. 3D plot of the Velocity from SUMO data

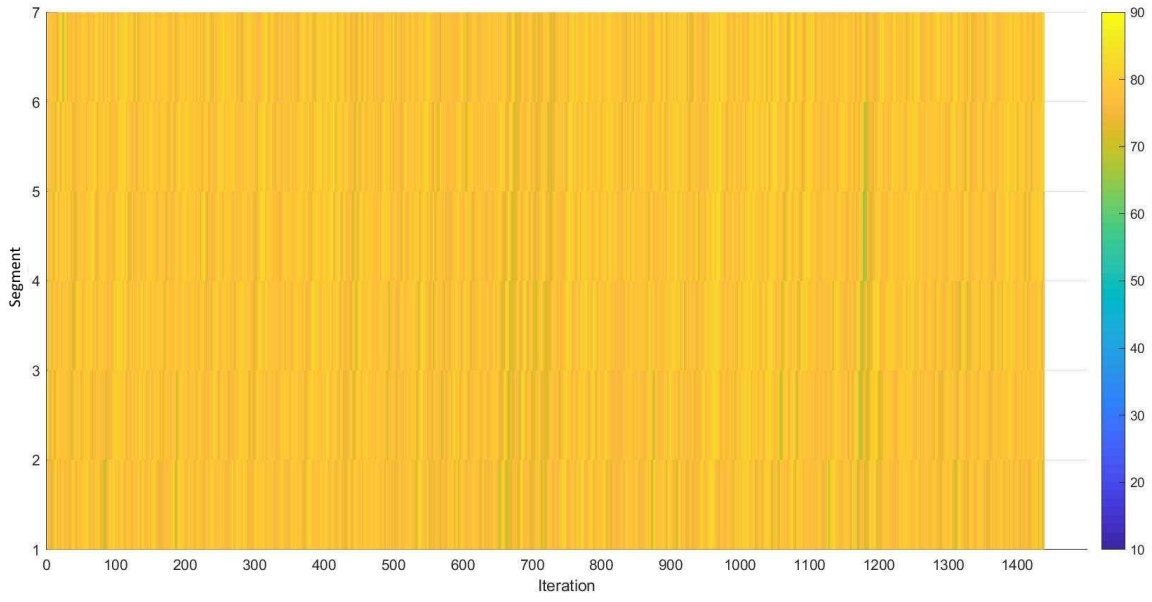

Figure 3.48. 3D plot of the Velocity from SUMO data (View from the top)

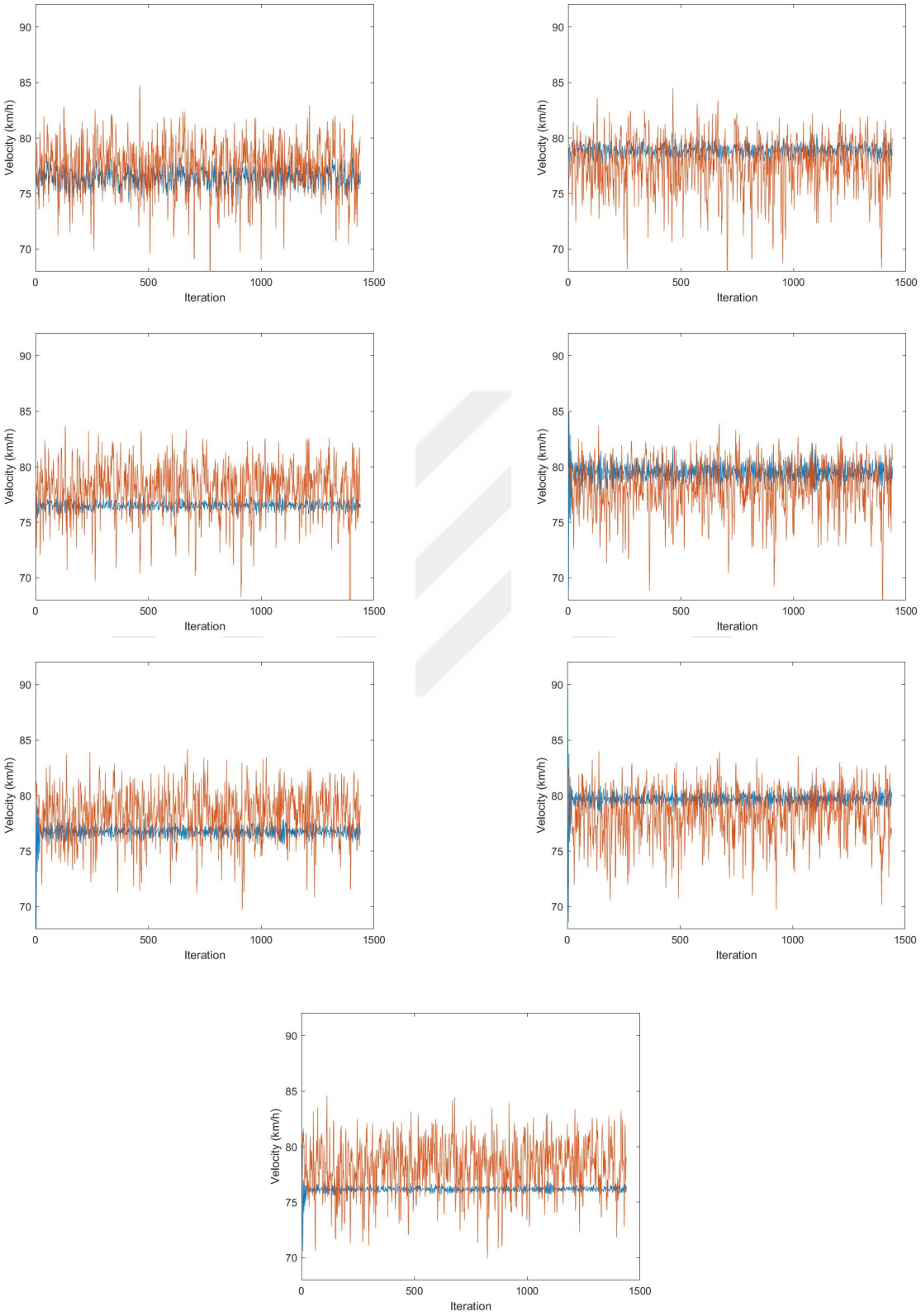

Figure 3.49. 2D plots of the velocity for segments 1 to 7 (starting at top left position). The blue curve is METANET's data and the orange curve is SUMO's data.

3.3.3.3. Flow.

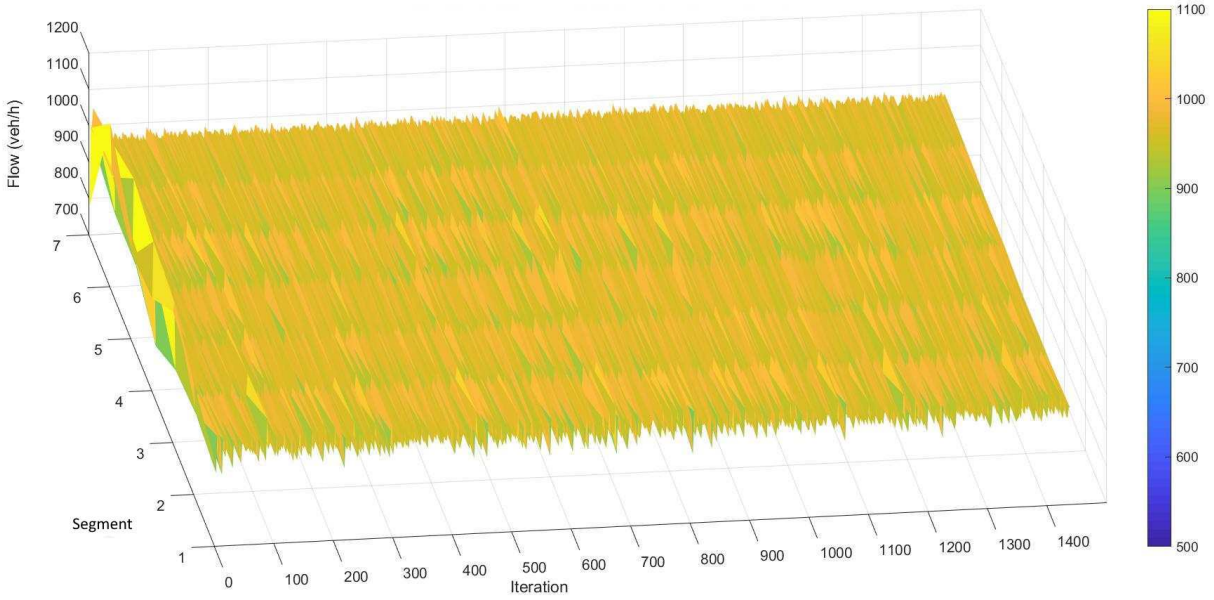

Figure 3.50. 3D plot of the Flow from METANET data

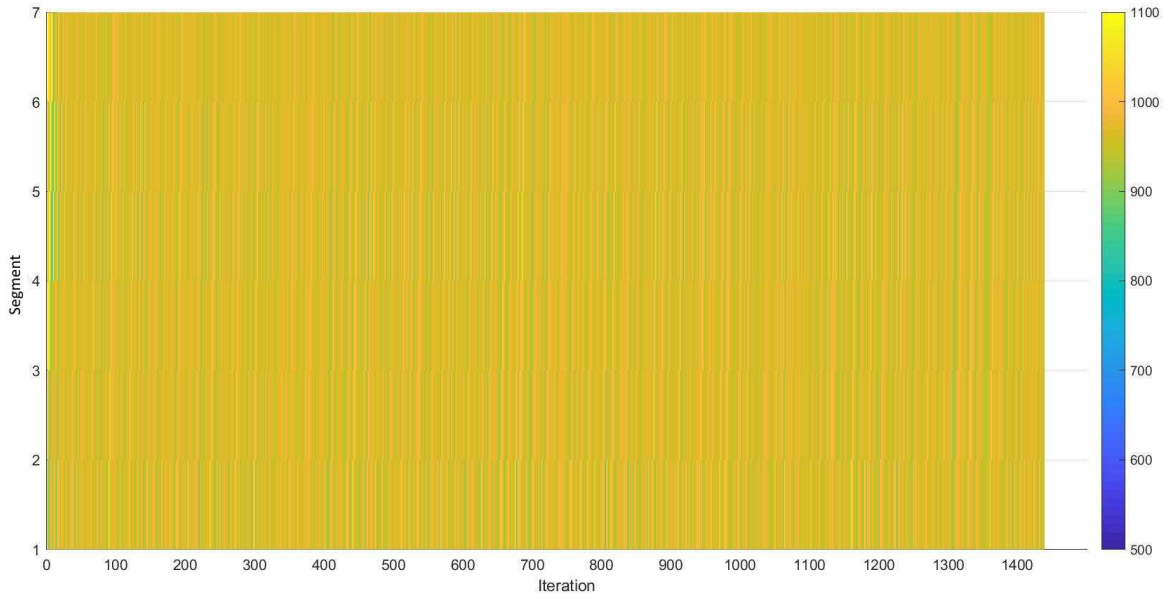

Figure 3.51. 3D plot of the Flow from METANET data (View from the top)

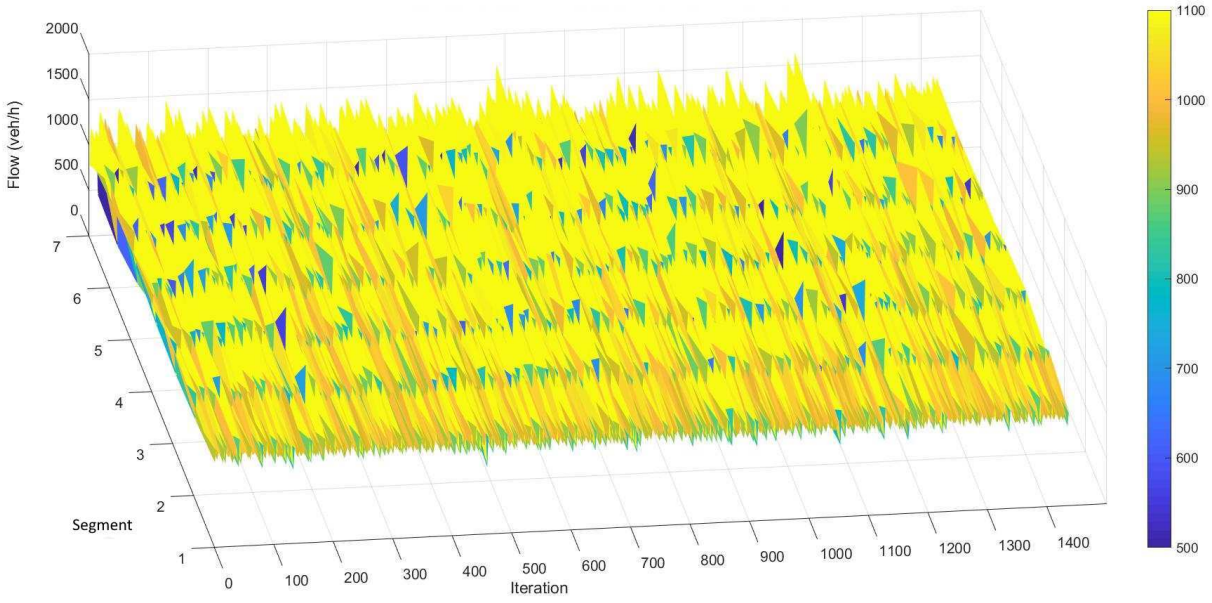

Figure 3.52. 3D plot of the Flow from SUMO data

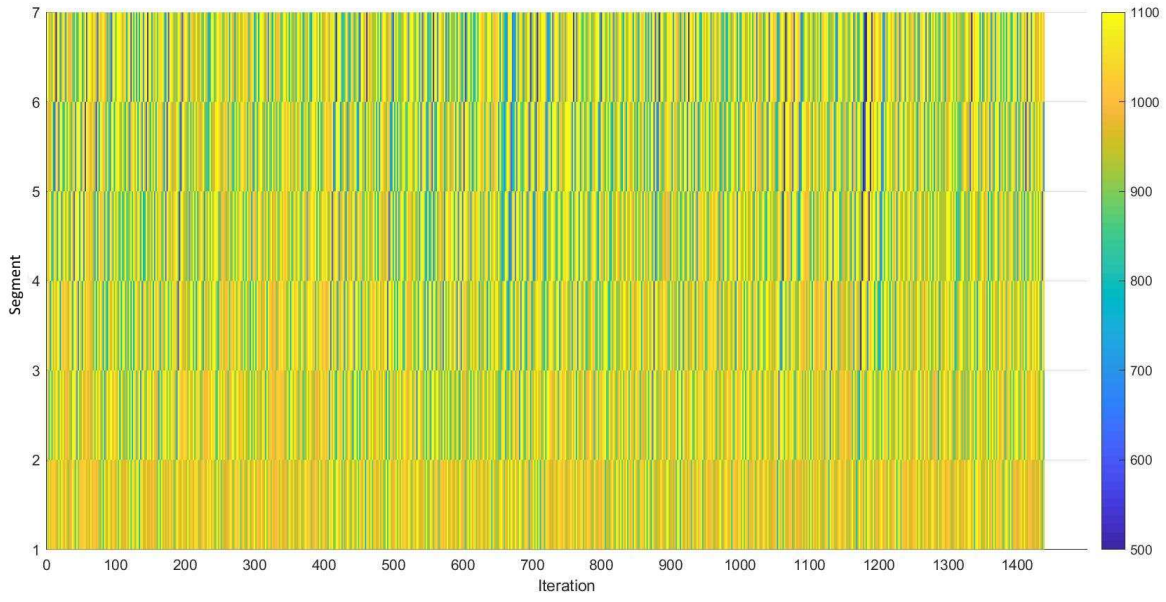

Figure 3.53. 3D plot of the Flow from SUMO data (View from the top)

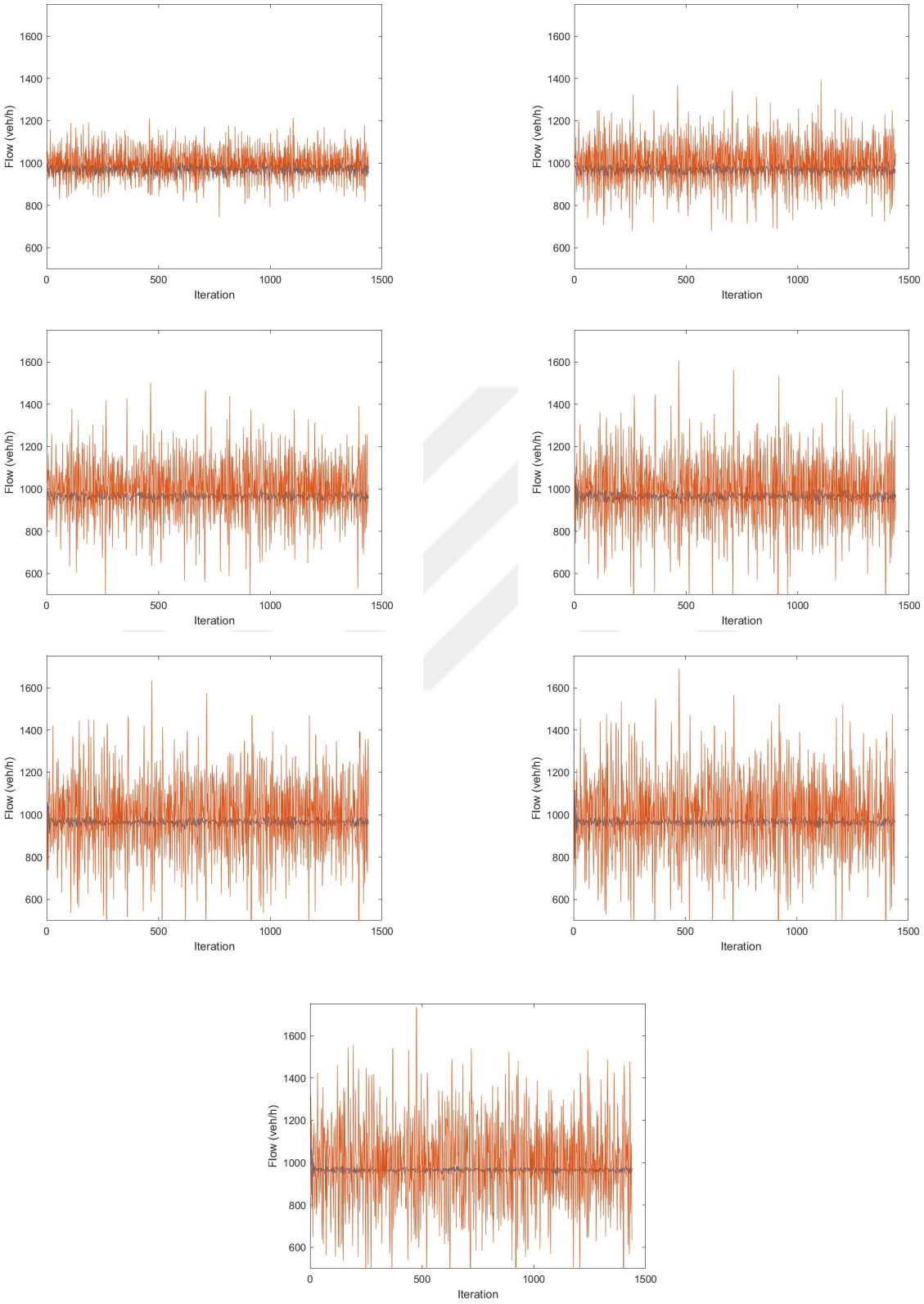

Figure 3.54. 2D plots of the flow for segments 1 to 7 (starting at top left position). The blue curve is METANET's data and the orange curve is SUMO's data.

3.3.3.4. Analysis. In this scenario, the model mimics the SUMO model precisely. Although some parameters such as κ, ν , and α are different for different segments, they do not make significant changes on the results. The calibrated model for this scenario also fits perfectly with the SUMO model and it is validated by the different set of data from SUMO. The obtained errors during the calibration and validation phases show if the model is precise. If the error in the validation phase is much greater than the calibration phase, it means the model is over-fit and not precise, since it mimics the noises. However, in this scenario, the validation error value is within the 20% range and it is acceptable. Therefore, the model is validated.

4. Conclusion

4.1. Discussion

A calibrated and simulated microscopic traffic flow model is used in this thesis to calibrate a second-order macroscopic traffic flow model to represent the traffic flow on the motorway in Sweden by validation. For this purpose, different scenarios which represent free flow, intermediate flow and congestion have been studied. This model can be used to better manage the traffic and implement control strategies on motorways.

In all the previous research, the applied models were considering the network as a whole or several links, a link contains one or more segments depending on their geometric characteristics, thus, the applied parameters were the same for all segments in a link/ network. Since each segment has its own characteristics, not only due to its type and geometry, but also due to its distance from an incident, especially when congestion happens, as a contribution in this project, a model that applies the parameters and calibration per segment is tested. Therefore, instead of having a global model parameters, each segment has its own parameters. Thus, instead of calibrating 9 parameters for the 7 studied segments, 63 parameters have been calibrated by RMSE objective function.

MATLAB is used to process the calibration, applying the General Pattern Search algorithm through its `patternsearch` function. Sensitivity tests were produced to find the best values for the parameters' lower and upper boundaries. For every segments, a convergence of the parameters are plotted after 200 iterations. The model is validated through a visual inspection and an error comparison between the calibrated dataset and another generated dataset from SUMO.

The configuration of METANET and SUMO are explained, and the results of the different studied scenarios are presented. The parameters which have a significant impact on the results while facing congestion are ρ_{cr},τ, α. In this project, as mentioned before, 63 parameters are calibrated for each scenario. With a per segment calibration, the time to calibrate is 683.998597 seconds, and the objective function value for the first scenario is 0.15815, however, while calibrating the model as a whole network, the time is 277.911846 seconds and the objective function value is 0.16826. Although the calibration time for 63 parameters is approximately three times more than the 9 parameters for each scenario, the results are evidently more precise and accurate when evaluating the plots. By knowing the ρ_{cr} and α for each segment, fundamental diagram for each segment can be obtained. As discussed before, by having the fundamental diagram, useful information such as traffic state estimation can be retrieved, which will help to take ITS control actions, or to be able to predict the traffic flow and state by implementing filters such as Extended Kalman Filter.

To conclude the thesis, the METANET model is calibrated by segment where each segment has 9 parameters, afterwards, the model is validated with a different set of data from SUMO, therefore, it has been shown that the obtained results are in agreement with previously-calibrated SUMO data.

4.2. Future Work

Having a calibrated traffic flow model has its very own benefits. However, to have a better control on the traffic and its state, ITS applications and Traffic Management should be implemented on the roads. Nowadays, many control systems such as VSL, Ramp-Metering, dynamic traffic light sequence and collision avoidance systems are applied on the highways, nevertheless, implementing these applications with the help of a precise second-order model with micro-detailed segment parameters can be exceptionally beneficial since second-order models can capture the dynamics of the traffic. In consideration of previous works done in this field, new control strategies can be applied on the road by taking the local parameters into the account rather than having a global

value for the parameters for the whole network, in this regard, the model and control strategies would be more specific and error-free.

Different boundary conditions and upper-bound and lower-bound intervals can be tested and studied. Further scenarios and different incidents can put in the place in order to see the changes in the traffic flow on the studied motorway. Changes in the geometry, adding on or off-ramps, permanent lane reductions, weaving lane or bus lanes can be added to the network.

Furthermore, by adding filters to the model such as Kalman filter, extended Kalman filter and etc. the model can be calibrated on-line to estimate and predict the traffic state.

REFERENCES

A. Kotsialos, M. Papageorgiou, C. Diakaki, Y. Pavlis, and F. Middelham. Traffic flow modeling of large-scale motorway networks using the macroscopic modeling tool METANET. IEEE Transactions on Intelligent Transportation Systems, 3(4): 282-292, dec 2002. [23]

A. Messner and M. Papageorgiou. Metanet: A macroscopic simulation program for motorway networks. Traffic Engineering & Control, 31(8-9):466-470, 1990.

A.D. Van der Horst, Calibration of the IDM and METANET Traffic Flow Models , Delft University 2011

Chandler, R. E., R. Herman, and E. W. Montroll, (1958). Traffic Dynamics: Studies in Car Following, Opns. Res. 6, pp. 165-183.

Daganzo,C.F.:The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. Transportation Research Part B: Methodological 28 (1994) 269-287

Daganzo,C.F.:The cell transmission model part II: Network traffic.Transportation Research Part B: Methodological 29 (1995) 79-93

Edie, L. C. and R. S. Foote, (1958). Traffic Flow in Tunnels, Proc. Highway Research Board, 37, pp. 334-344.

Greenshields, B.D.: A study of traffic capacity. In: Proceedings of the Highway Research Board, Vol. 14. Highway Research Board, Washington, D.C. (1935) 448-477

Herman, R., Editor (1961). Theory of Traffic Flow. Elsevier Science Publishers.

Herman, R., (1992). Technology, Human Interaction, and Complexity: Reflections on Vehicular Traffic Science. Operations Research, 40(2), pp. 199-212.

Hoogendoorn S.P. and Bovy P.H.L. (2000). "State-of-the-art of vehicular traffic flow modelling." Tech. rep., Transportation and Traffic Engineering Section, Faculty of Civil Engineering and Geosciences, Delft University of Technology, Delft, The Netherlands. Accepted for publication in the Special Issue on Road Traffic Modelling and Control of the Journal of Systems and Control Engineering - Proceedings of the Institution of Mechanical Engineers, Part I.

J. Erdmann, German Aerospace Center (2015). SUMO's Lane-changing model. LECTURE NOTES IN CONTROL AND INFORMATION SCIENCES, 13, pp. 105-123. [Online] Available at:
https://elib.dlr.de/102254/ [Accessed on 4 April 2019].

Junchen Jin, Xiaoliang Ma, Ingrid Johansson (2016). Modeling And Analysis Of PID-Controlled Heavy-Duty Vehicle Platoons In Real Traffic

Lamon, F. 2008. Freeway Traffic modeling and calibration for the Eindhoven network. MSc. thesis, Delft University of Technology, the Netherlands.

Lighthill, M. J., and G. B. Whitham. On Kinematic Waves II: A Theory of Traffic Flow on Long Crowded Roads. Proc., Royal Society of London, Series A, Mathematical and Physical Sciences, Vol. 229, No. 1178, 1955, pp. 317-345.

M. Cremer and M. Papageorgiou. Parameter identification for a traffic flow model. Automatica, 17(6):837-843, 1981.

M. Papageorgiou, Blosseville J. M., Hadj-Salem H. (1989) Macroscopic modelling of traffic flow on the Boulevard Peripherique in Paris. Tbsp. Res., 23B, 29-47.

M. Papageorgiou , Blosseville, J. M., & Hadj-Salem, H. (1989). Macroscopic modelling of traffic flow on the Boulevard Peripherique in Paris.Transportation Research Part B: Methodological, 23(1), 29-47.

M. Papageorgiou and A. Kotsialos. Freeway ramp metering: an overview. IEEE Transactions On Intelligent Transportation Systems, 3(4):228-239, December 2002.

M. Papageorgiou (1998), Some Remarks On Macroscopic Traffic Flow Modelling. Dynamic Systems and Simulation Laboratory, Technical University of Crete, GR-73100 Chania, Greece

M. Treiber, A. Kesting (2013). Traffic Flow Dynamics. Springer

Newell, G. F. (1955). Mathematical Models for Freely Flowing Highway Traffic. Operations Research 3, pp. 176-186.

Payne, H. J. Models of Freeway Traffic and Control. Simulation Council Proceedings, 1971.

Pipes, L. A. An Operational Analysis of Traffic Dynamics. Journal of Applied Physics, Vol. 24, No. 3, 1953, pp. 274-281.

Ramon, 2015. Model Predictive Control for Freeway Traffic Networks. PhD. thesis, University of Seville.

Reuschel, A. (1950a). Fahrzeugbewegungen in der Kolonne. Oesterreichisches Ingenieur-Aarchiv 4, No. 3/4,pp. 193-215.

Reuschel, A. (1950b and 1950c). Fahrzeugbewegungen in derKolonne bei gleichformig beschleunigtem oder verzogertem Leitfahrzeug. Zeitschrift des Oesterreichischen Inge-

nieurund Architekten- Vereines 95, No. 7/8 59-62, No. 9/10 pp.73-77.

Richards, P. I. (1956). Shock waves on the highway. Operations research 4(1), 42-51.

Seybold, 2015. Calibration of fundamental diagrams for travel time predictions based on the cell transmission model. MSc. thesis, Linköping University.

Stefan Krauß, Universitat zu Koln (1998). Microscopic Modeling of Traffic Flow: Investigation of Collision Free Vehicle Dynamics [Online] Available at: https://sumo.dlr.de/pdf/KraussDiss.pdf [Access on 4 April 2019].

Stefan Krauß, Wagner P and Gawron C, "Metastable States in a Microscopic Model of Traffic Flow," Physical Review E, vol. 55, no.5, pp.55-97, 1997.

T. Bellemans, B. De Schutter, and B. De Moor, "Models for traffic control," Journal A, vol. 43, no. 3-4, pp. 13-22, 2002.

Cantarella, G., Luca, S., Gangi, M., Pace, R. and Memoli, S. (2014). Macroscopic vs. Mesoscopic traffic flow models in Signal Setting Design. IEEE.

Tom V. Mathew, IIT Bombay (2014). Transportation Systems Engineering [Online] Available at: https://nptel.ac.in/courses/105101008/downloads/cete 1 6.pdf [Accessed on 2 December 2018].

Virginia Torczon (1997). On The Convergence Of Pattern Search Algorithms, SIAM J. O PTIM. Vol. 7, No. 1, pp. 1-25.

Wardrop, J. G. (1952). Some Theoretical Aspects of Road Traffic Research. Proceedings of the Institution of Civil Engineers, Part II, 1(2), pp. 325-362, U.K.

Webster, F. V. (1958). Traffic Signal Settings. Road Research Technical Paper No. 39. Road Research Laboratory, London, U.K.

APPENDIX A: The Generalized Pattern Search Method

A.1. The Pattern

To solve this problem, pattern search uses a pattern to explore the possible solutions. The pattern is composed of two components, the basis matrix B and the generating matrix C. The basis matrix is a real non-singular matrix $B \in \mathbb{R}^{nxn}$ and the generating matrix is an integer matrix $C = \Gamma^{nxp}$ where $p > 2n$. The basis matrix is usually equal to the identity matrix to provide coordinate search (Torczon 1997). The generating matrix is divided into three components :

$$C_k = [M_k \quad -M_k \quad L_k] = [\Gamma_k \quad L_k] \tag{A.1}$$

The matrix M_k is called the core pattern, and it applies the condition $M_k \in M \subset Z^{nxn}$, where M is a finite set of non-singular matrices, in order to ensure that it has full row rank. The second condition is $L_K \in Z^{nx(p-2n)}$ and L_k must have at least one column containing only zeroes.

The pattern is formed by the columns of the matrix P_k, which is defined in the Equation A.2.

$$P_k = BC_k = [BM_k \quad -BM_k \quad BL_k] = [B\Gamma_k \quad BL_k] \tag{A.2}$$

A constraint applied to the pattern is that the matrix BM_k has to be diagonal. This constraint is mandatory when applying boundaries to the algorithm to be able to find points in the pattern that decrease the objective function. To find the feasible points, a trial step is defined. For a given $\Delta_k \in \mathbb{R}$ with $\Delta_k > 0$, a trial step is any vector of the form $s_k = \Delta_k B c_k$, where c_k is a column of the vector C_k. To be called feasible, a trial step has to respect the condition $x_k + s_k \in \{x \in \mathbb{R}^n \quad | \quad lb \leq x \leq ub\}$, where x_k is the current iteration point.

A.2. Bound Constrained Exploratory Moves

When running the pattern search algorithm, explanatory moves are executed from the current iteration point x_k to find the next iteration point $x_{k+1} = x_k + s_k$, with s_k being a feasible step. The following hypotheses are made to ensure the convergence of the method.

(i) $s_k \in \Delta_k P_k \equiv \Delta_K B C_k$
(ii) $(x_k + s_k) \in \Omega$
(iii) If $\min\{f(x_k + y) | y \in \Delta_k B \Gamma_k,\ x_k + y \in \Omega\} < f(x_k)$, then $f(x_k + s_k) < f(x_k)$

A.3. Generalized Pattern Search Method

The algorithm is applied until a certain number of iterations has been accomplished. For a given $x_0 \in \mathbb{R}^n$ and $\Delta_0 > 0$:

(i) Compute $f(x_k)$
(ii) Determine a step s_k using a bound constrained exploratory moves algorithm
(iii) If $f(x_k + s_k) < f(x_k)$ then $x_{k+1} = x_k + s_k$. Otherwise, $x_{k+1} = x_k$
(iv) Update C_k and Δ_k

A.4. Update Δ_k

The step size has finally to be updated accordingly to the obtain minimization.

$$\Delta_{k+1} = \begin{cases} \lambda_k \cdot \Delta_k, & \text{if } f(x_k + s_k) < f(x_k) \\ p \cdot \Delta_k, & \text{otherwise} \end{cases} \tag{A.3}$$

APPENDIX B: Syntax and Parameters of the `patternsearch` function

$$patternsearch(f, x0, A, b, Aeq, beq, lb, ub, nonlcon, options) \tag{B.1}$$

With the following constraints applied:

$$\min f(x) \text{ such that } \begin{cases} A \cdot x \leq b \\ Aeq \cdot x = beq \\ lb \leq x \leq ub \end{cases} \tag{B.2}$$

- f : The function that `patternsearch` has to minimize. In this thesis, the given function is the objective function as described in the Equation 2.15
- $x0$: The starting point, which acts as a reference value to calculate the other solutions. This starting point can be a real value, a vector or a matrix. For this project, the parameters vector p as shown in the Equation 2.13 is used as the starting point, since it contains all the parameters , whose their values have to be determined.
- A and b : Parameters that are respectively a matrix and a vector, used to apply an inequality constraint to the possible values of x. In the thesis, no inequality constraints are applied to the values of the model's parameters as they are limited by the boundaries, hence, A and b are both set to an empty matrix and vector.
- Aeq and beq : Parameters that are respectively a matrix and a vector, used to apply an equality constraint to the possible values of x. In this project, no equality constraints are applied to the values of the model's parameters as they are limited by the boundaries, so Aeq and beq are both set to an empty matrix and vector as well.

- *lb* and *ub* : These variables are the lower-bounds and upper-bounds of the values of which the solution parameters can have. The values used in the thesis can be seen in Section 3.3
- *nonlcon* : Defines a non linear inequalities $c(x)$ or equalities $c_{eq}(x)$ that `patternsearch` tries to optimize as $c(x) \leq 0$ and $c_{eq} = 0$. In this thesis, no non linear inequalities or equalities are applied, thus, this parameter *nonlcon* is set to an empty matrix.
- *options* : The options given to the function. Such as an algorithm to use, number of iterations and the tolerances can be considered as options that can be set. The options are explained below.

In this thesis, the function to optimize (see Equation 2.15) receives 63 parameters, corresponding to the 9 parameters to optimize (see Equation B) of the 7 segments. Therefore, when applying the *GPS* algorithm, the initial point and the mesh points are as described in Equation B.3.

$$x_0, x_k \in \mathbb{R}^{63} \tag{B.3}$$

$$\overbrace{\begin{matrix} [1 & 0 & 0 & ... & 0 & 0] \\ [0 & 1 & 0 & ... & 0 & 0] \\ & & & ... & & \\ [0 & 0 & 0 & ... & -1 & 0] \\ [0 & 0 & 0 & ... & 0 & -1] \end{matrix}}^{63} \tag{B.4}$$

The pattern vectors formed by P_k have a length of 63 elements. The vectors are multiplied by the mesh size, which has a value of 1 at the beginning, and then are added separately to x_k in order to obtain the mesh points, as it can be observed

in Figure B.1. The value of the objective function is calculated for each points in the order the vectors are listed above. If the value is inferior than the previously calculated minimization value, it is called a successful poll and the mesh point is selected as the new referential. The size of the mesh is multiplied by 2, which is the default value in *MATLAB*. On the contrary, if there is no point minimizing the objective function, the poll is called unsuccessful, the current point stays the same and the mesh is refined. When refining the mesh, its size is divided by two.

This whole process is repeated until one of the following conditions is reached:

- The maximum number of iterations is reached. This number is by default set to $100 * \text{number of variables}$.
- The mesh size has become too small and is under the default value of 1.10^{-6}. This value is set to avoid infinite loop when no more improvements can be made.
- The running time of the algorithm has reached the maximum value, which by default is set to inf.
- The number of objective function evaluations has reached the maximum value.
- The distance between the last two obtained points and the mesh size are both under the minimum value, set by default to 1.10^{-6}.
- The difference value between the last two objective function values are less than the tolerance value, set by default to 1.10^{-6}.

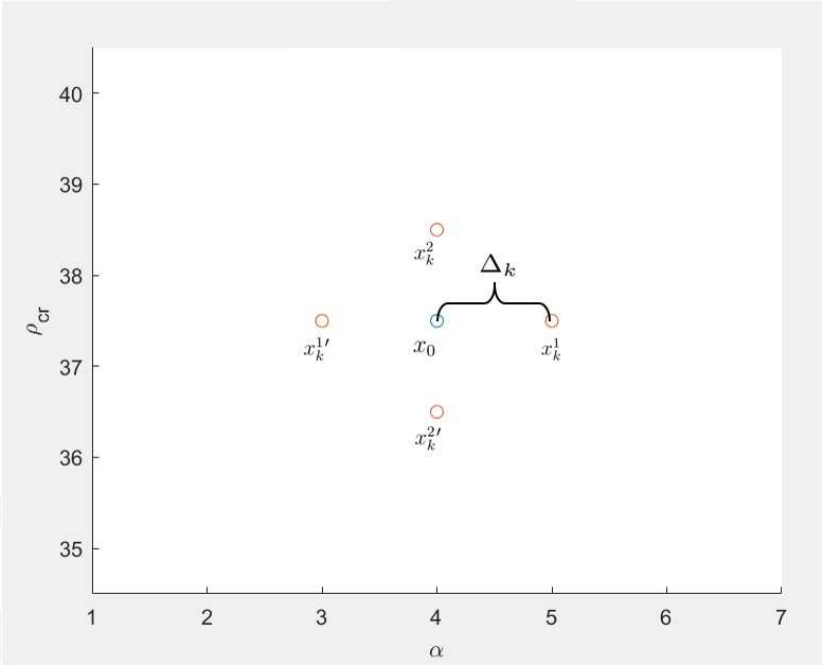

Figure B.1. Representation of the subset $[\alpha, \rho_{cr}]$ of the parameters when calibrating from a starting point x_0 equal to the mean of the boundaries.

Once the algorithm has stopped, the best minimizing point is returned. This point can be used again as a starting point to improve the minimization even more, especially if the algorithm stopped before due to the maximum number of iterations that was reached.

Options have to be set before calling the `patternsearch` function. By default, MATLAB runs the `patternsearch` function over $100 * \texttt{number of variables}$ iterations. Running 100 to 400 iterations is most of the time sufficient as the function does not improve the objective function a lot afterwards, thus, from what have been observed during the tests, 200 iterations has been chosen for the calibration. The `UseCompletePoll` option should be set to true. This option verifies all the points of the mesh instead of stopping the iteration after one minimizing point has been found. By doing so, the algorithm explores more points that could give a better minimization than the first obtained minimizing point and that could lead to the global minimum. The last two options are `UseVectorized` and `UseParallel`. The first one is to not consider the function to optimize as a vectorized objective function, which should re-

turn a vector of function values. The second one forces MATLAB to run the function with `parpool` in order to shorten the algorithm's processing time.

$$\begin{aligned} options = optimoptions('patternsearch','MaxIterations', 200, \\ 'UseParallel', true, 'UseCompletePoll', true, \\ 'UseVectorized', false); \end{aligned} \tag{B.5}$$

Finally, the function is called as below, using only the lower and upper boundaries as optional parameters, both explained in Section 3.3.

$$res = patternsearch(@J, x0, [\,], [\,], [\,], [\,], lowerbounds, upperbounds, [\,], options)$$

APPENDIX C: Convergence of the parameters for Segments 2 to 5 and 7

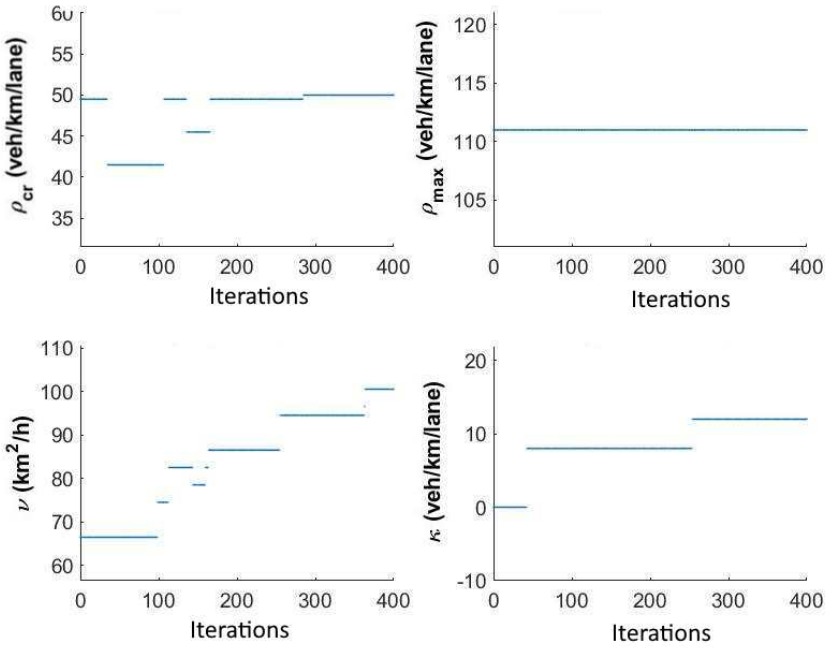

Figure C.1. Convergence of the parameters $[\rho_{cr}, \rho_{max}, \nu, \kappa]$ of Segment 2 over 400 Iterations.

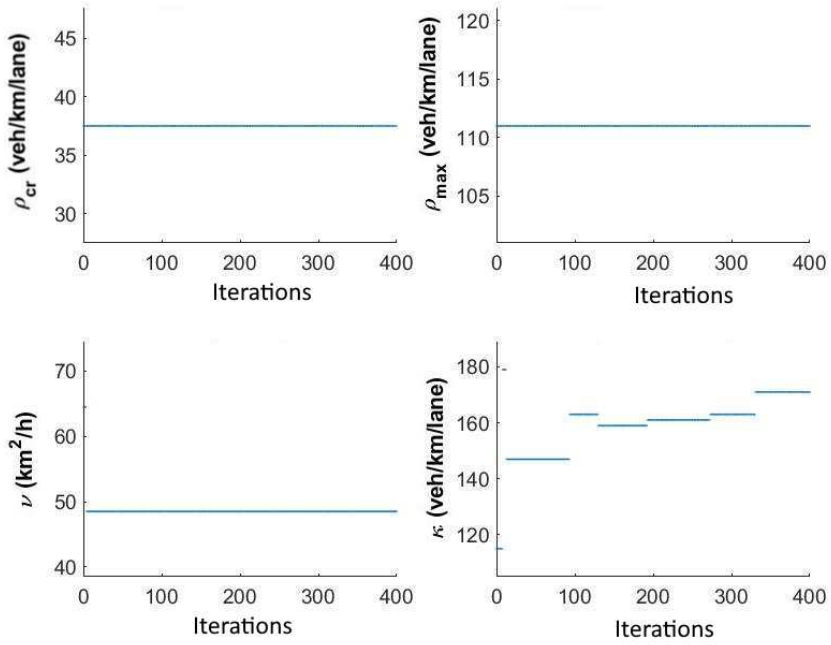

Figure C.2. Convergence of the parameters $[\rho_{cr}, \rho_{max}, \nu, \kappa]$ of Segment 3 over 400 Iterations.

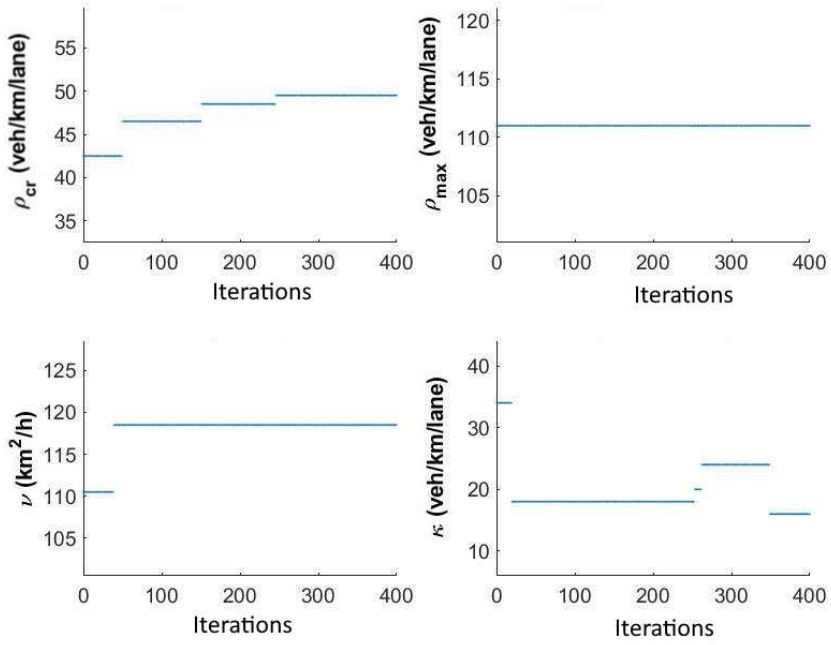

Figure C.3. Convergence of the parameters $[\rho_{cr}, \rho_{max}, \nu, \kappa]$ of Segment 4 over 400 iterations.

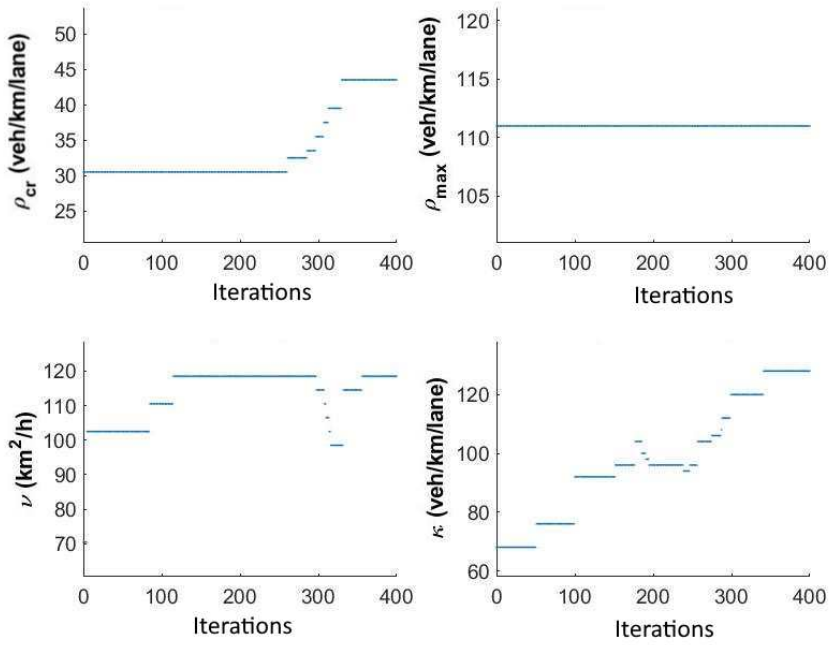

Figure C.4. Convergence of the parameters $[\rho_{cr}, \rho_{max}, \nu, \kappa]$ of Segment 5 over 400 Iterations.

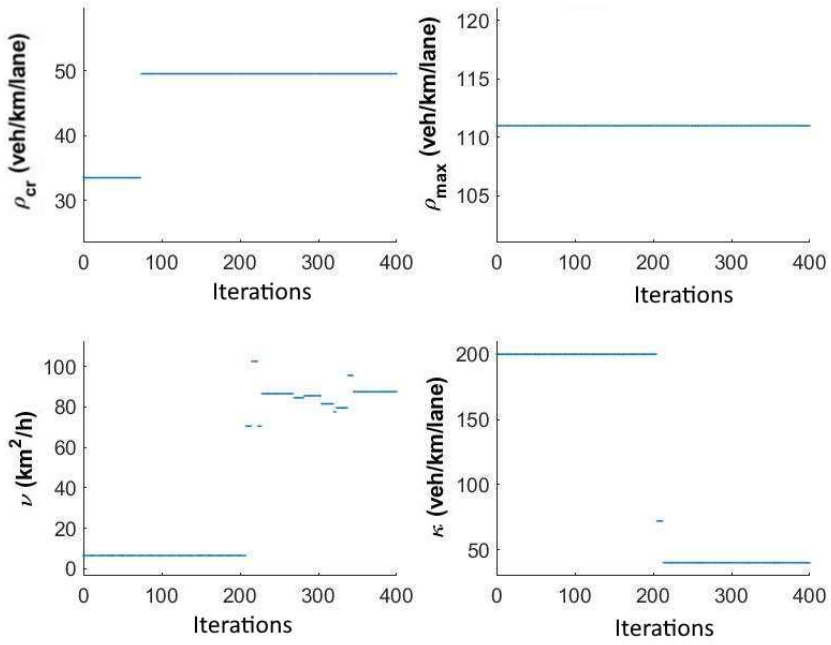

Figure C.5. Convergence of the parameters $[\rho_{cr}, \rho_{max}, \nu, \kappa]$ of Segment 7 over 400 Iterations.