Regulating Traffic Flow and Speed on Large Networks: Control and Geographical Self Organizing Map (Geo-SOM) Clustering

Maha Elouni

Dissertation submitted to the Faculty of the Virginia Polytechnic Institute and State University in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Computer Engineering

Hesham A. Rakha, Chair
Amos L. Abbott
Ioannis M. Besieris
Leanna L. House
Leyla Nazhandali
Haibo Zeng

May 10, 2021
Blacksburg, Virginia

Keywords: Traffic flow control, speed harmonization, network fundamental diagram, macroscopic fundamental diagram, GeoSOM clustering

Copyright ©2021, Maha Elouni
Regulating Traffic Flow and Speed on Large Networks: Control and Geographical Self Organizing Map (Geo-SOM) Clustering
Maha Elouni
(ABSTRACT)

Traffic growth and limited roadway capacity decrease traveler mobility and increase traffic congestion and fuel consumption. Traffic managers employ various control techniques to mitigate the aforementioned problems. One well-known network-wide control strategy is perimeter control (or gating). Perimeter control is based on the Network Fundamental Diagram (NFD). NFD-based perimeter control techniques are used to solve congestion problems in transportation networks. One well-known method used in the literature is Proportional Integral Control (PIC). PIC solves the congestion problem, but suffers from sensitivity to parameter tuning and the need for model linearization. A weather-tuned perimeter control (WTPC) and a jam density-tuned perimeter controller (JTPC) were developed to cope with parameter sensitivity for different weather conditions and jam densities, respectively. In an attempt to overcome PIC problems, a sliding mode controller (SMC) was developed. SMC does not require model linearization and parameter tuning. It is also robust to varying demand patterns. SMC computes the flow that needs to enter a protected network and converts it to corresponding traffic signal timings to achieve the desired control strategies. Another approach to implementing the sliding mode controller is to control vehicle speeds on the links entering the protected network. Coupling speed harmonization (SH) with sliding mode control (SMC), an SMC-SH was developed and implemented in the INTEGRATION microscopic traffic simulator. The mentioned controllers are all tested on a mid-size grid network replicating downtown Washington DC. SMC-SH improved different performance metrics on the whole grid network compared to the no control case. Specifically, it improved average travel time, total delay, stopped delay, fuel consumption, CO\(_2\) emissions by 17.27%, 18.18%, 12.76%, 5.91%, and 7.04%, respectively. In order to test the SMC-SH on a real large-scale network, the downtown Los Angeles (LA) network is used. The LA network is known for its congested freeways, so a development of a Freeway-SMC-SH controller is performed and tested. It shows good results in improving the performance not only of freeways, but also the overall LA network performance. Particularly, the network-wide average travel time, total delay, stopped delay, fuel consumption and CO\(_2\) emissions improved with respect to the no control case by 12.17%, 20.67%, 39.58%, 2.6%, and 3.3%, respectively. An identification of a homogeneously congested area is needed to apply SMC-SH on LA roads (not freeways). The geographical self organizing maps (GeoSOM) clustering algorithm is applied and tested on the LA network. The clustering goal is to identify a geographically connected region with small density variance. GeoSOM is able to achieve that objective with better performance than the state-of-the-art Kmeans and DBSCAN clustering algorithms. The enhancements reached up to 15.15% for quantization error, 61.05% for spatial quantization error, and 43.96% for variance. Finally, the SMC-SH is tested on the protected region of the
LA network identified by the GeoSOM algorithm. SMC-SH succeeds in improving network-wide vehicle travel time, total delay, stopped delay, fuel consumption and CO$_2$ emissions by 6.25%, 9.4%, 16.47%, 1.7%, and 2.19%, respectively.
Road congestion causes vehicular delays and increases travel time and fuel consumption. The goal of the research is to prevent or relieve traffic congestion in a network. That region that we attempt to address is termed the congested network or the protected network (PN). One way to solve the traffic jam problem is to set up gates on the PN borders so that the number of vehicles that enter the network is limited, and consequently traffic jams do not occur. However, the number of vehicles should not be limited too much to avoid overcrowding outside the PN. The developed controller calculates the right number of cars that should enter the network in order to improve the performance inside and outside the PN. The first way to apply the controller commands is to adjust traffic signal timings at the traffic signals located along the PN border. The second way (called SMC-SH) is to adjust the speed of the vehicles entering the network through these gates. In the first part of the work, all the controllers are implemented and tested in a mid-size grid network. In the second part of the work, the goal is to implement the controller on the real large-scale Los Angeles (LA) network. Since the LA network suffers from congestion on freeways, a freeway controller is developed and tested. It does not only succeed in reducing traffic jams on freeways, but also enhances the overall LA network traffic performance. In order to apply the SMC-SH controller on the LA network, we identify homogeneously congested regions. GeoSOM clustering is implemented to achieve this goal and compared to other clustering methods, and is shown to outperform them. Finally, the SMC-SH controller is tested on the congested region of LA, and succeeds in reducing travel time, total delay, and fuel consumption for the LA network.
Dedicated to

My husband Mohamed and my kids Missen and Youssef
My mother Wahida and the memory of my father Hachemi
My parents in law Nejiba and Youssef
My siblings Tahani and Abdessalem
My siblings in law Asma and Mohamed Ali
All my family and friends
My supervisor
Acknowledgments

First of all, I praise ALLAH, the most Merciful and most Gracious, for this accomplishment. I would like to express my heartfelt gratitude to my advisor Dr. Hesham Rakha for his unwavering support and guidance and invaluable encouragement. His trust in me pushed me forward to achieve greater heights. Besides that, his understanding and kindness and his human side can only make me proud and thankful that he is my advisor. I would like to thank my committee members, Dr. Lynn Abbott, Dr. Ioannis Besieris, Dr. Leanna House, Dr. Leyla Nazhandali, and Dr. Haibo Zeng for their insightful comments and valuable feedback and time.

This work was supported in part by the Department of Energy through the Office of Energy Efficiency and Renewable Energy (EERE), Vehicle Technologies Office, Energy Efficient Mobility Systems Program. I would like to thank them for funding this research.

My sincere thanks go to Dr. Hossam Abdelghaffar for his continuous help, support, and guidance. I learned a lot from him and from our fruitful discussions. I would like to thank Dr. Youssef Bichiou for his assistance and encouragement, especially during the beginning of my Ph.D. research.

I would like to acknowledge my colleagues in the Center for Sustainable Mobility (CSM) especially Dr. Ihab El-Shawarby, Dr. Ahmed Elbery, Dr. Mohammed Elhennawy, Dr. Mohammed Almanna, Dr. Mohammad Aljamal and Seifeldeen Eteifa. Special thanks to my collegues, friends and sisters in CSM Dr. Taqwa Alhadidi and Asmaa Alazmi. I will not forget the amazing moments together at CSM.

My friends in Blacksburg have contributed greatly by supporting me with the enjoyable time we spent together as a family. I would like to extend my thanks particularly to Dr. Saher Lahouar, Afek Jaballah, Dr. Bilel Aidi, Hana Dammak, Dr. Karim Fadhloun, Asma Khouildi, Dr. Yassine Mahjoubi, Dr. Inem Tamniche, and Amira Haouel.

Special thanks go to my friends for their encouragements and support even after they left Blacksburg, namely Dr. Abdessattar Abdelkefi, Dr. Samah Ben Ayed, Dr. Najib Masghouni, Marwa Assali, Said Mseddi, Dr. Bochra Bouchhima, Dr. Amara Loulizi, Sonia Loulizi, Faycal Beji and Autumn Taylor.

I would like to thank my dearest childhood friends in Tunisia Dr. Loubna Ghammam and
Ahlem ben Abdallah who believed in me and encouraged me a lot.

I dedicate this work to the memory of my father Hachemi and I wish he is proud of me as I fulfilled the promise I made to him to obtain a Ph.D. degree. My deep thanks go to my mother Wahida who has given me infinite moral support, encouragement, and prayers over the years. She is the one who always believes in me and pushes me to do better. I would like also to thank my parents in law Youssef and Nejiba for being supportive and encouraging and for accepting to spend time with us here during difficult busy moments. Special thanks go to my siblings Tahani, Abdessalem, and my siblings-in-law Asma, Mohamed Ali, Saber, Amina, and Mohamed Hedi.

My companion and my dear husband Mohamed, I will not be able to thank you enough for your love, encouragement, advice, help, and patience. You have been always by my side during my happy and challenging moments. As a Hokie, you have been always my guide since the beginning of my graduate studies.

I would like to thank my sweetheart and my lovely daughter Missen who has accompanied me on this whole Ph.D. journey, and my dear little son Youssef who joined us at the end of my Ph.D. and added joy to our lives. I love you both to the moon and back.
# Contents

List of Figures xi

List of Tables xiii

1 Introduction 1
   1.1 Network Control Challenges ........................................... 2
   1.2 Research Contributions ................................................ 3
   1.3 Document Layout ......................................................... 5

2 Literature Review 6
   2.1 Weather Responsive Controllers ........................................ 6
   2.2 Speed Harmonization ..................................................... 7
   2.3 Self Organizing Maps .................................................... 9

3 Work Environment 11
   3.1 Derivation of the NFD Equations ...................................... 11
   3.2 Grid Network ............................................................... 12
   3.3 Los Angeles (LA) network .............................................. 13
   3.4 INTEGRATION Micro-Simulator ........................................ 14
      3.4.1 Description of INTEGRATION ..................................... 14
      3.4.2 Weather Modeling in INTEGRATION .............................. 15

4 Proportional Integral Controller 17
4.1 Proportional Integral Controller (PIC) ............................................ 17
4.2 Weather Impact on the NFD .......................................................... 19
    4.2.1 Clear Weather Conditions .................................................. 19
    4.2.2 Rain impacts ................................................................. 20
    4.2.3 Snow Impacts ............................................................... 21
    4.2.4 Comparison of Clear, Rainy and Snowy Conditions ............... 23
    4.2.5 Conclusion ................................................................. 24
4.3 Weather and Jam Density Tuned PIC ........................................... 24
    4.3.1 PC for Clear Weather Conditions and Base Jam Density ........ 25
    4.3.2 Weather-Tuned Perimeter Control (WTPC) ............................. 26
    4.3.3 Jam Density-Tuned Perimeter Control (JTPC) ......................... 27
    4.3.4 Conclusion ................................................................. 30
4.4 Conclusion .................................................................................... 30

5 Sliding Mode Controller .................................................................. 31
    5.1 SMC based on Traffic Signals .................................................. 32
        5.1.1 Controller Equations ..................................................... 32
        5.1.2 Performance of the SMC and PIC for Different Parameters .... 36
        5.1.3 Response to Different Demand Profiles .............................. 39
    5.2 Sliding Mode Controller based on Speed Harmonization .......... 41
        5.2.1 Model and Controller Equations ..................................... 41
        5.2.2 Implementation and Results .......................................... 43
    5.3 Conclusion ................................................................................ 45

6 Freeway Speed Control .................................................................. 47
    6.1 Proposed Model & Controller Formulation ............................... 48
        6.1.1 Model Formulation ....................................................... 48
        6.1.2 Controller Formulation .................................................. 50
    6.2 Simulation Setup & Results ..................................................... 52
# List of Figures

1.1 Network Fundamental Diagram (NFD) ........................................... 1
1.2 Research Flow Chart ................................................................. 3

3.1 Grid network modeled in INTEGRATION ........................................ 13
3.2 NFD of the grid network ............................................................ 13
3.3 LA network .................................................................................. 14

4.1 An abstract schematic of an urban network that features a congestion in the protected subnetwork ......................................................... 17
4.2 System and feedback controller structure ........................................ 18
4.3 NFDs of the high and average demand scenarios for clear weather conditions ................................................................. 19
4.4 NFD for rain weather conditions .................................................... 20
4.5 NFD of snowy weather conditions .................................................. 22
4.6 Critical TTS and maximum TTD for snow weather conditions .......... 22
4.7 Comparison of clear, rainy and snowy NFDs .................................... 23
4.8 Comparison of the NFDs for NPC and PC during clear weather conditions ................................................................. 25
4.9 NFD for rain conditions and snow conditions ................................... 26
4.10 NFDs of PN for different jam densities ........................................... 28
4.11 Bar graphs of the performance metrics for the FN using different jam densities ................................................................. 29

5.1 Phase plane plot of sliding control system ...................................... 31
5.2 Time series of the density k from the protected area ......................... 38
5.3 NFD of the grid subnetwork without and with control ...................... 38
List of Tables

4.1 Rainy Weather Conditions Parameters. ..................................... 21
4.2 Snowy Weather Conditions Parameters. .................................... 23
4.3 Performance metrics of the PC for the FN using the average of ten different seeds. ................................................................. 25
4.4 Performance Metrics on the FN for the NPC, PC and WTPC Cases. .... 27

5.1 Sensitivity of SMC and PIC with respect to NPC (base case) for various values of controller parameters $\lambda, \eta, \mu,$ and $\zeta$. .................................................. 37
5.2 Average Simulation Results for the Demand Profiles of Figure 5.4 ........ 40
5.3 Percentage Improvements Using SMC-SH Over NPC for Protected sub-Network 45
5.4 Network wide results MOEs and (%) Improvement Using SMC-SH Over NPC 45

6.1 Network Average MOEs and (%) Improvement Using Freeway-SMC-SH Over Base Controller ......................................................... 55
6.2 Freeway Links MOEs and (%) Improvement Using Freeway-SMC-SH Over Base Controller ............................................................. 55

7.1 Mean densities for different classes of DBSCAN clustering .................. 64
7.2 Mean density for different classes .................................................. 66
7.3 Performance Metrics for different Clustering Algorithms ................... 67
7.4 Improvement Percentage of GeoSOM over DBSCAN and Kmeans .......... 67
7.5 Network wide results MOEs and (%) Improvement Using SMC-SH Over NPC in LA network ............................................................. 68
Chapter 1

Introduction

Traffic growth and limited roadway capacity increase traffic congestion, decrease traveler mobility, and increase fuel consumption. Traffic managers employ various control techniques to mitigate the aforementioned problems. One well-known network-wide control strategy is perimeter control (or gating). Perimeter control is based on the Network Fundamental Diagram (NFD). The NFD gives an aggregated view of the road network characteristics, namely: (1) the average network density ($k$): the number of vehicles per unit length of the road or lane; (2) traffic stream flow ($q$): the number of vehicles passing a fixed point per unit of time; and (3) space-mean speed ($u$): the traffic stream density weighted average speed.

![NFD of a Network](image1)

(a) NFD of a Network

![NFD scatter and interpolated NFD](image2)

(b) NFD scatter and interpolated NFD

Figure 1.1: Network Fundamental Diagram (NFD)

The NFD has three regimes, as illustrated in Figure 1.1, namely: the free-flow regime, the saturation or capacity regime, and the over-saturation or congested regime. Perimeter control attempts to maintain the accumulation around a set point (which corresponds to the...
maximum throughput) to avoid network over-saturation or congestion building up (Figure 1.1(a)).

1.1 Network Control Challenges

In general, obtaining a clear and well-behaved NFD can be difficult. The scatter of points (as presented in Figure 1.1(b)) might not generate a visible curve. However, if the distribution of congestion is uniform across the network, the existence of a well-defined NFD is guaranteed [1–3]. Godfrey et al. [4] were the first to introduce the concept of the NFD for the center of London. The authors demonstrated that the relationship between the average velocity and the vehicle-travelled distance is parabolic and that the speed is inversely proportional to the density. Geroliminis et al. [5] observed the NFD in the congested urban network of Yokohama, Japan, demonstrating that under homogeneous conditions, even with large discrepancies between the different link fundamental diagrams, the NFD of the network can be reproduced. This means that the NFD is a direct result of the infrastructure and thus is independent of the number of vehicles in circulation, the demand, and the selected routes (i.e., independent of the road taken by each individual vehicle and the origin-destination table). Leclercq et al. [6] acknowledge that the shape and scatter of the NFD is subject to local traffic heterogeneities. This is achieved through a non-homogeneous distribution of congestion due to heterogeneous local capacities and route choice decisions. They also conclude that it is difficult to link and understand the connection between local phenomena and the NFD.

Many clustering algorithms have been developed in order to determine small regions where there is small variance in density and a NFD can be generated [7–13]. With the determination of multi-region networks, researchers developed multi-region controllers [14–16].

Other studies focused on controlling only one region. Different real-time perimeter control techniques were used, such as the standard proportional-integral controller (PIC) [17–20], a robust PIC [8], a linear quadratic controller [14], and a model predictive controller. Haddad et al. [21–23] introduced various adaptive perimeter controller schemes that take into account model uncertainty and the NFD scatter. However, due to the nonlinear nature of the NFD, model linearization is essential for controller design.

In this research a nonlinear controller that does not require model linearization is developed based on the sliding mode theory. The Sliding Mode Controller (SMC) alters the dynamics of the system by applying strong control actions when the system deviates from the desired behavior [24, 25]. It does not require accurate modeling, and is robust to system disturbances [26, 27]. Sliding mode control has been applied in many fields, such as mechanical, robotics, electrical, control systems, chaos theory, network control, etc. Sliding mode has also been applied in traffic systems to control vehicle platooning [28], ramp metering control [29], and traffic routing [30]. Mirkin et al. [31] developed an adaptive sliding mode controller that
1.2. Research Contributions

takes into account uncertainties and unknowns in the NFD, delayed input, and adjustable gains. However, it requires numerous design parameters and needs model linearization. Details about the SMC can be found in [25].

In order to assess the performance of the developed Sliding Mode Controller (SMC), a well-known simple and effective Proportional Integral Controller (PIC) used in the literature was implemented [17]. In general, the PIC requires a number of input parameters, namely: two gains that have to be tuned a priori. The optimum gain values are typically difficult to compute using the various methods to estimate the best gains [32–34]. Anandanatarajan et al. [35] and Sung et al. [36] showed further limitations of proportional-integral (PI) and proportional-integral-derivative (PID) controllers. Kouvelas et al. developed robust adaptive tuning techniques that alleviated the tuning problem [37]. However, the implementation of PIC in [17] was followed in this work since researchers proved its effectiveness in solving perimeter control problems.

1.2 Research Contributions

Figure 1.2: Research Flow Chart
Figure 1.2 presents the research flow chart. In the first part of the work, the developed methods were applied to a mid-size grid network having a well-defined homogeneously congested region requiring gating to relieve the congestion. Since the formulation of the PIC is based on the NFD, a study of the weather impacts on the NFD was performed. Subsequently, we developed a Weather-and-Jam-Density-Tuned-PIC and tested it on the grid network. Results showed that Weather-and-Jam-Density-Tuned-PIC was superior to PIC, which proves that PIC is sensitive to the tuned parameters. Besides that, PIC requires a well-defined linearized NFD in order to work properly. Those are two main drawbacks of the PIC that the developed SMC solved. SMC was developed and tested on the same grid network, and was demonstrated to have many advantages over PIC.

The developed controllers (PIC and SMC) compute the flow that should enter the protected network and convert it into corresponding traffic signal timings that enforce the desired flow. In order to be independent from traffic signals, the widespread technology of connected automated vehicles (CAVs) is used in this work to develop a vehicle-centric SMC controller based on speed harmonization (SMC-SH). The implementation entails controlling CAV speeds on the links entering the protected network instead of controlling the traffic signal timings. This also allows the system to be implemented on freeways and reduces the system cost.

The second part of the work consists in applying the SMC-SH to a large-scale network (Downtown Los Angeles (LA)). Since, the LA network is also known for its congested freeways, a dynamic freeway controller based on Sliding mode and Speed Harmonization was developed to mitigate that problem. The proposed Freeway-SMC-SH controller is unique in three ways: first, the system identifies bottlenecks dynamically without having to do so a priori; second, the problem formulation and the solution are based on the space-mean speed, which may be obtained from probe vehicles (CVs)—no fixed road sensors are needed; and third, the impacts are studied on a large real world network composed of multiple connected freeways and signalized urban roads.

Identifying a homogeneously congested and spatially connected region in LA network was needed before applying the SMC-SH controller. In order to do that, an adaptation of the Geographical Self Organizing Map (GeoSOM) clustering algorithm was performed. GeoSOM is special because it takes into consideration geographical locations in its architecture and does not treat them as part of the attributes. To the best of our knowledge, GeoSOM was not applied before in the clustering of a transportation network. GeoSOM was compared to state-of-the-art DBSCAN and Kmeans clustering methods and was shown to outperform them.

Finally, SMC-SH was applied to the congested region that was identified using the GeoSOM algorithm on the LA network, and achieved success in reducing network-wide travel time, total delay, and fuel consumption.
1.3 Document Layout

The layout of the work is as follows. We start with a literature review chapter that provides an extensive review of the literature related to this research. Then, a chapter of 'work environment' will be presented. It covers a description of the road networks used in this work and a description of the INTEGRATION traffic micro-simulator used. Its purpose is to present the tools used in this work one time, instead of repeating them in several chapters. Chapter 4 presents the PIC with its different version of Weather and Jam Density Tuned PIC. The SMC is presented in Chapter 5 along with SMC based on Speed Harmonization (SMC-SH). Chapter 6 presents the Freeway SMC-SH controller. GeoSOM clustering is presented in Chapter 7 along with the application of the SMC-SH controller to the LA network. Chapter 8 presents the conclusions of the study and presents recommendations for future research.
Chapter 2

Literature Review

In this chapter, the most relevant literature review of the main areas of our research is presented.

2.1 Weather Responsive Controllers

The NFD is exploited in many studies to control the flow entering the network borders. This strategy has proved highly effective in decreasing network congestion and delays. However, these methods always assume clear weather conditions. Pisano et al. introduced the idea of weather-responsive traffic management, enumerating research needs to advance the state of the practice in weather-responsive traffic control [38]. In order to develop an NFD-based weather responsive perimeter control strategy, a knowledge of the weather’s impact on the NFD is necessary. However, no comprehensive studies in the existing literature have investigated this. Existing studies concerning weather impact on traffic have some shortcomings. First, some studies focus on rural expressways and freeways rather than urban roads [39–41]. Second, the impact of inclement weather has been studied on a limited number of road segments rather than whole networks, largely due to the lack of data [42]. Sabir et al. did investigate the effect of adverse weather at the trip level for all of the Netherlands [43]. However, only a few studies have been performed at a network level, and this still warrants more investigation. Xu et al. studied the impact of rainfall on an urban road network in China. They used the NFD for the macroscopic analysis of the network and reported average reductions in production, accumulation, and weighted speed for rainy days of 0.2%, 7.8% and 5.4%, respectively [44]. In another study, they investigated the impact of different rainfall intensities on the network, finding that heavy rain and rainstorms reduced critical accumulation and maximum production by 10.5%, 16.7% and 21%, and 18.7% respectively. However, both light and moderate rains had relatively little impact on the network’s operation [45]. Furthermore, the impact of snow was not studied in their
2.2. Speed Harmonization

Congestion on highways is a severe problem resulting in delays, vehicle emissions, fuel consumption increase due to frequent accelerations/decelerations [50], and vehicle crashes as there are more sudden braking maneuvers when vehicles traveling at high speeds approach a queue. Various efforts and studies have tried to address this issue through the use of speed harmonization (SH) techniques, an intelligent transportation system application. SH regulates the velocity of vehicles in order to improve traffic conditions, effectively reducing congestion, enhancing mobility, safety, and therefore reducing environmental impact. Variable speed limit (VSL) is one SH technique that is widely used in the literature. VSL consists of changing the speed limit (i.e., generally reducing it) on a particular upstream of a bottleneck link. This is achieved via dynamic messages displayed on roadway signs for traditional vehicles, a suggested speed limit sent to connected vehicles (CVs), or an imposed speed on connected and automated vehicles (CAVs). Further details about SH strategies can be found in Ma et al.’s review [51], where research efforts in traditional SH and in SH with emerging technologies (CAVs) are listed.

SH techniques can be categorized into two main categories: reactive and proactive (i.e., RSH and PSH). The RSH approach is activated after a given threshold (i.e., when congestion at
a bottleneck starts to build up), and has been proven to be successful. Different control approaches have been used to solve the VSL problem. Jin et al. used a proportional-integral to prove the effectiveness of the VSL in increasing the bottleneck discharge rate [52]. However, they used a kinematic wave model, which does not account for the stochastic nature of the traffic. Yang et al. studied the impact of a dynamic SH on a lane drop bottleneck using a bang-bang feedback control [53]. However, they tested their method on a simple freeway section, and more testing on larger real networks with multiple SH zones is needed. Malikopoulos et al. developed an analytic solution for the optimal control problem applicable in real time [54, 55]. This method was also tested on a simple freeway section. In order to implement this method in real networks, some of the authors’ assumptions need to be relaxed; more precisely, lane changing and mixed traffic (human vehicles and CAVs) should be considered. In optimal control problems, different optimization objectives are used. Yang et al. implemented a tri-objective bi-level programming model aimed at improving safety and reducing congestion and emissions [56]. Their model is effective but not suitable for dynamic conditions, as it was solved statically using a genetic algorithm. Moreover, the authors assume advanced knowledge of the bottleneck location, around which a set of traffic sensors are deployed; this is not always possible in real world applications.

Unlike RSH, PSH performs an estimation of the traffic flow before congestion occurs, then, based on this estimation, the SH algorithm is activated [54]. Ghiasi et al. used mixed traffic in their optimization approach, i.e. human driven vehicles, CVs, and CAVs [57]. The methodology showed improvements in throughput, speed variations, fuel consumption, and surrogate safety measures. However, it is also computationally expensive, as it has many steps, each of which requires a great deal of calculation. The authors’ model considers a one-lane roadway, but they show that it could be extended to a multi-lane scenario. Yang et al. used a Kalman filter to estimate traffic states, then used optimal control to implement VSL [58]. They obtained good results; however, the simulation was conducted for a simple freeway segment with two on-ramps and one off-ramp. Two VSL signs and seven detectors were installed along the roadway segment. A larger test network is needed in the future studies with a determination of the optimal detector locations. Mittal et al. used a traffic estimation and prediction system relying exclusively on a mesoscopic simulator to estimate traffic conditions [59]. However, mesoscopic models are limited in providing detailed traffic operations [60]. Numerous research efforts combined estimation and control using Model Predictive Control (MPC) to predict and adjust the speeds of vehicles moving on the bottleneck’s upstream link [61–63]. Each of the cited references had a different optimization objective, and some had multiple optimization objectives. For instance, Khondaker et al. [62] tried to enhance mobility and safety while at the same time promoting sustainability. Although MPC is popular, it has been proven to be either expensive to implement or sub-optimal. Frejo et al. [64], for example, showed that local MPC provides sub-optimal solutions and that global MPC is complicated to implement in real time. In addition, using MPC methods requires sufficient accuracy of the traffic state prediction, which is a challenging problem when the nonlinearity induced by capacity drop is encountered. It is also important to note here that most of the research efforts with regards to VSL focus on small
2.3 Self Organizing Maps

The original Self-Organizing Maps (SOM), also called self-organizing feature maps (SOFM), was proposed by Kohonen in 1982 [67]. The idea behind SOM is mapping an input space into a low-dimensional output space (one or two dimensions) composed of units called neurons. This mapping preserves the topology of the input space. This means that inputs close to each other are mapped to neurons close to each other too, and vice versa. It is a very useful tool that helps understand the data structure, especially in high dimensional spaces [68].

SOM is primarily a data-driven dimensionality reduction and data compression method. In data mining and machine learning, it is widely used in data clustering, data classification and graph mining. SOM is used in many applications like image segmentation, vector quantization and image compression, density modeling, gene expression analysis, text mining and
information management, data visualization, object classification, skin detection, learning robot behaviors, learning a motion map, object recognition, etc. [69].

SOM is an artificial neural network based on competitive learning. Every competitive neuron is represented by a weight vector and calculates a similarity measure between the weights and an input vector. The winning neuron (also called the Best Matching Unit (BMU)) is the neuron that has weights closest to the input vector. The weights of the neurons are then updated such that the neighborhood of the winning neuron approaches the input vector.

There are two types of SOM, a traditional sequential SOM, and a batch SOM [70]. The sequential algorithm takes the inputs one by one recursively, whereas in the batch algorithm the inputs are provided to the neurons as a whole.

There are two types for clustering with SOM: k-means SOM and emergent SOM. In k-means SOM, the number of neurons is chosen to be the number of clusters, requiring an estimated number of clusters beforehand. The number of neurons in Emergent SOM is much higher than the number of clusters. The number of clusters can be deduced by visualizing the unified distance matrix (called U-matrix [71]) after the end of the simulation [72]. The U-matrix calculates the distances between each neuron and its neighbouring neurons. A projection of the U-matrix to the input space helps to identify the real clusters. When distances between neurons are small, the neurons are included in the same clusters. Large distances between neurons represent cluster borders. Instead of visual clustering on top of the SOM, Vesanto et al. [73] show the performance of two different clustering methods (hierarchical agglomerative clustering and partitive clustering using k-means) to cluster SOM units, and compared their performance to the performance of using them to cluster the input vectors directly. Results show that clustering on top of the SOM gives similar results to direct clustering, but is computationally more efficient. In [74], the authors developed an automated clustering method based on the U-Matrix. This method is based on Vellido’s algorithm [75] with a local density based pruning procedure. The method produced better results than k-means clustering and Vellido’s algorithm.
Chapter 3

Work Environment

The purpose of this chapter is to present the tools used in this work one time, instead of repeating them in several chapters.

3.1 Derivation of the NFD Equations

NFD equations are used to plot the NFD curve in order to determine the set point for the perimeter controllers used in this work. It has to be noted that NFD equations are used in the PIC modelling but not in the SMC. The NFD is computed based on the average link density \((k)\) in vehicles per unit length and the average vehicle flow \((q)\) inside the network in vehicles per unit time. These quantities can be computed from loop detectors placed throughout the network, where \(k\) is computed using equation (3.1)

\[
k[n] = \frac{1}{L} \sum_{z \in Z} k_z[n].l_z
\]

where \(z\) is the index for the link; \(Z\) is the set of all links belonging to the protected area where measurements are taking place; \(n\) is the time index and reflects the cycle number; \(L\) is the total length of the PN (i.e., the sum of the length of all links \(L = \sum_{z \in Z} l_z\); these links also feature loop detectors.); \(l_z\) is the length of link \(z\); and \(k_z[n]\) is the traffic stream density on link \(z\) during cycle \(n\) and is calculated using equation (3.2)

\[
k_z = n l_z k_{jz} \frac{o_z[n - 1]}{100}
\]

where \(n l_z\) is the number of lanes of link \(z\); \(k_{jz}\) is the jam density of link \(z\); \(o_z\) is the measured time-occupancy (in percent) on link \(z\).
The flow inside the network \((q)\) is calculated using equation (3.3)

\[
q[n] = \frac{1}{L} \sum_{z \in Z} q_z[n].l_z
\]  

(3.3)

where \(q_z[n]\) is the measured flow on link \(z\).

The NFD is the plot relating the flow \(q[n]\) to the network density \(k[n]\)

\[
q[n] = \phi(k[n])
\]  

(3.4)

where \(\phi\) is an unknown NFD function.

It is important to mention here that knowledge of the exact \(\phi\) function is not necessary. The plot of the NFD scatter (measurements of \(q\) and \(k\) during the simulation) is used to determine the set point that corresponds to the density having the highest flow.

### 3.2 Grid Network

The introduced grid network (Figure 3.1) is used to conduct the controllers testing. The network was created to replicate downtown Washington, DC, both in terms of its one-way streets and block lengths. This network includes a protected region, also referred to as the PN. The PN corresponds to the region surrounded by the green rectangle (Figure 3.1). All the access points to this subnetwork are identified with the yellow arrows. In total, there are eight links that feed directly into the protected area. The minimum and maximum allowable vehicle flow into the protected area via those eight links (Figure 3.1) are \(q_{\text{min}} = 480\text{veh/h}\) and \(q_{\text{max}} = 12,960\text{veh/h}\).

The PN contains 48 links, all of which are one-way roadways and each of which has only one lane of the same length of 150 m. The full network contains 36 signalized intersections. Origin and destination zones for trips are represented by blue circles. Loop detectors are placed on each link of the network and collect measurements every cycle. The cycle length is taken to be 60 s (i.e., \(\Delta t = 60s\)).

The base demand for this network is generated during the first 75 minutes of the simulation, increasing during the first 37.5 minutes and decreasing after that to model the buildup and decay of traffic congestion. The simulation time is taken to be 176 minutes in order to provide sufficient time for all vehicles to clear the network. Dynamic traffic assignment is also activated during the simulation to reflect realistic driver behavior during congested conditions (i.e., rerouting of vehicles is activated).
3.3 Los Angeles (LA) network

The NFD associated with the network presented in Figure 3.1 is shown in Figure 3.2. The point cloud, shown in Figure 3.2, presents snapshots of the simulated protected sub-network state. Note that congestion is observed beyond an average network density of $\bar{k} = 48.76\text{veh/km}$ (NFD curve starts to decrease after $\bar{k}$). Beyond this point, vehicles experience significant delays and consume additional fuel.

The real downtown LA network presented in Figure 3.3 has 457 signalized intersections, 285 stop signs, 23 yield signs, and a total of 3,556 links, with 331 freeway links [76]. The origin-destination demand was calibrated based on vehicle count data from loop detectors.
using QueensOD software [78] with 143,957 total trips. The phasing scheme for the intersections varied from 2 to 6 phases, which is consistent with the phases implemented in downtown LA. The minimum free-flow speed on the network was 15 (km/h), and the maximum free-flow speed on the network was 120 (km/h). The minimum link length on the network was 50 m, and the maximum link length was 4,400 m. The jam density on all links of the network was equal to 180 (veh/km/lane).

3.4 INTEGRATION Micro-Simulator

3.4.1 Description of INTEGRATION

The INTEGRATION microscopic software, which traces individual vehicle movements every deci-second, was used to evaluate the proposed controller’s performance [79, 80]. INTEGRATION is a microscopic model that replicates vehicle longitudinal motion using the Rakha–Pasumarthy–Adjerid collision-free car-following model, also known as the RPA model [81]. Vehicle movements are constrained by a vehicle dynamics model described in [82]. Vehicle lateral motion is modeled using lane-changing models described in [83]. The model estimates of vehicle delay were validated in [84], while vehicle stop estimation procedures are described and validated in [85]. Vehicle fuel consumption and emissions are modeled using the VT-Micro model [86]. It captures the capacity drop associated with the onset of congestion. This phenomenon has been validated against empirical data [87]. Examples of stochastic variables integrated in the software include driver specifications such as acceleration/deceleration rates, reaction times, desired speeds, and lane-changing behavior.

The following measures of effectiveness (MOEs) were measured to assess the system’s operation:
3.4. INTEGRATION Micro-Simulator

- Average Travel Time (seconds/vehicle): aggregated trip times divided by the total number of vehicles.
- Average Total Delay (seconds/vehicle): aggregated vehicle delay (i.e., the sum of the difference in travel time between travelling at instantaneous vehicle speed and travelling at the free-flow speed) divided by the total number of vehicles.
- Average Stopped Delay (seconds/vehicle): the sum of instances where vehicle speed is less than or equal to 1(kilometer/hour) divided by the total number of vehicles.
- Average Queue Length (vehicles): aggregated vehicles in queue each second divided by the simulation time.
- Average Fuel (liters/vehicle): aggregated fuel consumed by vehicles divided by the total number of vehicles.
- Average CO2 (grams/vehicle): aggregated CO2 produced divided by the total number of vehicles.

INTEGRATION is written in Fortran and has around 40,000 lines of code. All developed controllers were implemented within that code. Knowing the code logic and variables was necessary for a successful incorporation of the new controllers.

3.4.2 Weather Modeling in INTEGRATION

“Clear” is the default weather in the INTEGRATION micro-simulator. In order to model different weather conditions, the following set of parameters were calculated and inputed to the software: free-flow speed, speed-at-capacity, capacity, rolling coefficient, and coefficient of friction. These were calculated based on the work of Rakha et al., who studied the impact of weather on different traffic stream parameters [49, 88]. They developed weather adjustment factors (Equation (3.5)) to compute the free-flow speed, speed-at-capacity, and capacity, based on precipitation intensity \( i(cm/h) \) and visibility level \( v(km) \) for each of the rain and snow cases.

\[
WAF = a_1 + a_2i + a_3i^2 + a_4v + a_5v^2 + a_6 \quad (3.5)
\]

The calibrated model parameters \( a_1 \) through \( a_6 \) were chosen to be the Twin Cities parameters because it has the highest WAF (Table 1 in [49]). These WAF were multiplied by the clear conditions link-specific fundamental diagram parameters to reflect the change in the link fundamental diagram as a result of inclement weather. In addition, the rolling coefficients and coefficient of friction were modified to reflect the modified inclement weather road surface conditions. These factors affect vehicle deceleration and acceleration behavior within the
car-following model. In order to model different weather conditions in INTEGRATION, the set of inputs containing the free-flow speed, speed-at-capacity, capacity, rolling coefficient, and coefficient of friction were calculated for each precipitation type (rain and snow) and precipitation intensity, and given to the software.
Chapter 4

Proportional Integral Controller

4.1 Proportional Integral Controller (PIC)

Any given network can be represented by an abstract schematic (Figure 4.1). This network experiences moderate to heavy congestion during the course of the day, leading to delays, excessive fuel consumption, and pollution. To alleviate these effects, the identified area of congestion (PN) is protected. That is, all the inflows (i.e., $q_{in}^1, q_{in}^2$, etc.) are monitored and in some cases restricted to the extent that the protected area operates at capacity.

\[ \frac{dNV(t)}{dt} = q_{in}(t) - q_{out}(t) + q_d(t) \] (4.1)

Figure 4.1: An abstract schematic of an urban network that features a congestion in the protected subnetwork

The time rate of change of the number of vehicles inside the PN is computed using equation (4.1).
where \( q_{\text{in}}(t) = \sum_{i=1}^{m_i} q_{\text{in}i}(t) \) is the sum of all inflows at the instant \( t \); \( q_{\text{out}}(t) = \sum_{i=1}^{m_o} q_{\text{out}i}(t) \) is the sum of all vehicle outflows from the subnetwork; \( q_d(t) \) is the disturbance flow that might occur inside the protected area; and \( NV = k \times L \) is the number of vehicles inside the PN. NV is the product of the vehicle density \( k \) and the total length of the links \( L \) inside the protected region.

In order to avoid congestion inside a PN, a desired density \( \bar{k} \) is sought. This density is usually chosen to be the density at capacity (Figure 1.1(b) in the Introduction). Therefore, the error defined in equation (4.2) is driven to zero when the control is activated.

\[
e(t) = k(t) - \bar{k} \tag{4.2}
\]

where \( k \) is the current density of vehicles in the protected area, and \( \bar{k} \) is the desired density (density at capacity).

Figure 4.2 represents the system and feedback controller structure, where the controller tries to drive the error \( e \) to zero and commands an input flow \( q_{\text{in}} \) that needs to enter the PN ("system" in Figure 4.2).

The PIC equation for discrete time steps is presented in equation (4.3) as in [17]

\[
q_{\text{in}}[n] = q_{\text{in}}[n-1] - K_p(k[n] - k[n-1]) + K_I(\bar{k} - k[n]) \tag{4.3}
\]

where \( n \) is the time index and \( K_p = \frac{\mu}{\zeta} \) and \( K_I = \frac{1-\mu}{\zeta} \). Based on measured data from the NFD in the capacity regime and the least squares method, the values of \( \mu \) and \( \zeta \) are estimated using equation (4.4).

\[
k[n+1] - \bar{k} = \mu.(k[n] - \bar{k}) + \zeta.(q_{\text{in}}[n] - \bar{q}_{\text{in}}) + \varepsilon[n] \tag{4.4}
\]

where \( \bar{q}_{\text{in}} \) is the average measured inflow of vehicles at the capacity regime and \( \varepsilon[n] \) is an error term.

Tuning, in practice, could deliver negative and/or zero values. Particular care must be taken with respect to the calibration data so that appropriate \( \mu \) and \( \zeta \) values are obtained. During simulation, the values of density \( k \) at the previous time step, as calculated using loop detectors, and at the current time step are needed. These are used to compute the controlled flow that should enter the PN while avoiding congestion (equation (4.3)).
4.2 Weather Impact on the NFD

In this work, different rain and snow intensities were modeled in INTEGRATION based on the work of Rakha et al. [49]. A study of their impact on the critical accumulation and maximum production of the network’s NFD was conducted, and the impacts of inclement weather on average speed and travel times were reported. It has to be noted here that accumulation is a scaled density $k$, and it is referred in this subsection as Total Time Spent ($TTS$). Production is a scaled flow $q$, and it is referred in this subsection by Total Traveled Distance ($TTD$). Also, for this study, the whole grid network (Figure 3.1) was used, not only the Protected Network.

4.2.1 Clear Weather Conditions

The parameters used for the clear weather conditions were as follows: free flow speed was 45 km/h, speed at capacity was 35 km/h, and saturation flow rate was 1800 veh/h. This was applied for all links in the network.

Two scenarios were run for the clear weather conditions: a high demand scenario, causing a gridlock at the end of the simulation (Figure 4.3(a)), and an average demand scenario (Figure 4.3(b)). In each scenario, 10 simulations with 10 different random seeds were performed in INTEGRATION.

Figure 4.3: NFDs of the high and average demand scenarios for clear weather conditions.

Figure 4.3(a) shows the three parts of the NFD: the free-flow regime corresponding to the increasing part, the saturation regime where throughput is maximal, and the congested regime corresponding to the decreasing part of the NFD.

Two key parameters of the NFD are the maximum throughput (or maximum TTD) and the
critical accumulation (TTS corresponding to the maximum TTD). In this case, the maximum throughput occurred around a TTS range of [8500, 11000] veh.h/h.

### 4.2.2 Rain impacts

In the literature, light rain ranges from 0 to 0.2 mm/h, moderate rain ranges from 0.2 to 6 mm/h and heavy rain intensity is above 6 mm/h [46]. Different rain intensities from each range were considered in this work. The coefficient of friction was set to 0.9 and the rolling coefficient was set to 2.5. Based on the calculated WAF (equation 3.5), each rain intensity gave a set of free-flow speeds, speed-at-capacity, and capacity, which were provided as inputs to INTEGRATION.

The NFD variation for each rain category is represented in Figure 4.4. The blue rectangle in Figure 4.4(a) is zoomed in 4.4(b) where variation of the NFD’s parameters for light, moderate and heavy rain is more noticeable.

Moving from light to heavy rain, the maximum TTD decreases and the critical TTS is shifted to the right. Note that the TTS values for all rain intensities at the saturation regime vary between 8,230 and 8,355 vehicles, thus varying in a small range.

![Figure 4.4: NFD for rain weather conditions.](image)
4.2. Weather Impact on the NFD

Table 4.1 shows that as rain intensity increases, reduction in average speed and increase in total travel time become more pronounced.

<table>
<thead>
<tr>
<th></th>
<th>Rainy condition</th>
<th>Average difference (%)</th>
<th>Min and max difference (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max TTD</td>
<td>Light rain</td>
<td>-2.15</td>
<td>[-5.41, 0.56]</td>
</tr>
<tr>
<td></td>
<td>Moderate rain</td>
<td>-2.51</td>
<td>[-5.95, 2.44]</td>
</tr>
<tr>
<td></td>
<td>Heavy rain</td>
<td>-2.50</td>
<td>[-6.37, 2.13]</td>
</tr>
<tr>
<td>Critical TTS</td>
<td>Light rain</td>
<td>-0.30</td>
<td>[-3.61, 5.14]</td>
</tr>
<tr>
<td></td>
<td>Moderate rain</td>
<td>-1.48</td>
<td>[-9.46, 4.74]</td>
</tr>
<tr>
<td></td>
<td>Heavy rain</td>
<td>1.15</td>
<td>[-3.03, 6.28]</td>
</tr>
<tr>
<td>Average speed</td>
<td>Light rain</td>
<td>-3.38</td>
<td>[-18.67, -0.01]</td>
</tr>
<tr>
<td></td>
<td>Moderate rain</td>
<td>-3.59</td>
<td>[-8.10, 0.43]</td>
</tr>
<tr>
<td></td>
<td>Heavy rain</td>
<td>-4.93</td>
<td>[-9.03, -0.79]</td>
</tr>
<tr>
<td>Total Travel time</td>
<td>Light rain</td>
<td>3.49</td>
<td>[-0.20, 25.28]</td>
</tr>
<tr>
<td></td>
<td>Moderate rain</td>
<td>3.57</td>
<td>[-0.65, 9.15]</td>
</tr>
<tr>
<td></td>
<td>Heavy rain</td>
<td>5.03</td>
<td>[0.52, 10.18]</td>
</tr>
</tbody>
</table>

4.2.3 Snow Impacts

In the literature, light snow ranges from 0.0 to 1.3 mm/h, moderate snow ranges from 1.3 to 2.5 mm/h, and heavy snow intensity is above 2.5 mm/h [49]. Different snow intensities from each range are considered in this work. The coefficient of friction was set at 0.2 and the rolling coefficient was set at 2.5. NFD for different snow intensities is presented in Figure 4.5.
The general trend of variation in the maximum TTD and critical TTS for each snow intensity is represented by boxplots in Figure 4.6. As snow intensity increases, maximum TTD decreases and critical TTS increases.

The comparison of the different snowy conditions (light, moderate, heavy) with respect to the clear weather condition is presented in Table 4.2. The reported parameters correspond to the relative differences of the maximum TTD, critical TTS, average speed, and total travel time with respect to clear weather parameters.

Table 4.2 shows that the reductions in average speed and increase in total travel time are more pronounced for moderate and heavy than for light snow.
Table 4.2: Snowy Weather Conditions Parameters.

<table>
<thead>
<tr>
<th></th>
<th>Snowy condition</th>
<th>Average difference (%)</th>
<th>Min and max difference (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Max TTD</strong></td>
<td>Light snow</td>
<td>-47.49</td>
<td>[-50.14, -44.32]</td>
</tr>
<tr>
<td></td>
<td>Moderate snow</td>
<td>-49.26</td>
<td>[-51.23, -48.15]</td>
</tr>
<tr>
<td></td>
<td>Heavy snow</td>
<td>-49.36</td>
<td>[-51.17, -47.95]</td>
</tr>
<tr>
<td><strong>Critical TTS</strong></td>
<td>Light snow</td>
<td>-20.29</td>
<td>[-25.51, -16.79]</td>
</tr>
<tr>
<td></td>
<td>Moderate snow</td>
<td>-19.88</td>
<td>[-22.42, -16.96]</td>
</tr>
<tr>
<td></td>
<td>Heavy snow</td>
<td>-19.56</td>
<td>[-21.92, -17.34]</td>
</tr>
<tr>
<td><strong>Average speed</strong></td>
<td>Light snow</td>
<td>-59.41</td>
<td>[-68.85, -53.76]</td>
</tr>
<tr>
<td></td>
<td>Moderate snow</td>
<td>-62.72</td>
<td>[-64.28, -61.05]</td>
</tr>
<tr>
<td></td>
<td>Heavy snow</td>
<td>-62.66</td>
<td>[-63.74, -61.71]</td>
</tr>
<tr>
<td><strong>Total Travel time</strong></td>
<td>Light snow</td>
<td>61.19</td>
<td>[40.06, 112.93]</td>
</tr>
<tr>
<td></td>
<td>Moderate snow</td>
<td>74.27</td>
<td>[66.77, 82.31]</td>
</tr>
<tr>
<td></td>
<td>Heavy snow</td>
<td>73.97</td>
<td>[69.55, 79.21]</td>
</tr>
</tbody>
</table>

### 4.2.4 Comparison of Clear, Rainy and Snowy Conditions

The parameter averages of all intensities were calculated for each weather condition. The network NFDs are represented in Figure 4.7. On average, there is a very slight shift to the bottom and left for the rainy condition compared to the clear weather condition. However, the shift is very evident in the snowy condition. Even the shape of the NFD changes slightly. The decreasing part of the NFD in snowy conditions starts to appear, which means that congestion occurred.

![Figure 4.7: Comparison of clear, rainy and snowy NFDs.](image)
On average, the reductions in the maximum TTD, critical TTS, and average speed were 48.7%, 19.9%, 61.59%, respectively, for the snow case and 2.38%, 0.21%, 3.96%, respectively for the rain case. The total travel time increased by 69.8% for the snow case and by 4.03% for the rain case. Consequently, snowy conditions have a significant impact on the network’s performance, while only a small impact was observed for the rainy conditions.

4.2.5 Conclusion

Different weather conditions were modeled in the INTEGRATION micro-simulator. The impact of different rain and snow intensities on the NFD was investigated. Moving from light to heavy rain, the maximum TTD decreases and the critical TTS increases. This same trend is observed for snowy conditions. The reduction in average speed and the increase in total travel time become more evident as the rain intensity increases. For snow, the reduction in average speed and the increase in total travel time are more evident for moderate and heavy compared to light snow. Comparing the three different weather conditions, the reductions in the NFD parameters and average speed are much higher for snowy than for rainy conditions. These results can be used by traffic managers to develop weather-responsive traffic controllers.

4.3 Weather and Jam Density Tuned PIC


In the previous section, it was shown that inclement weather affects the critical Total Time Spent (TTS) and maximum Total Traveled Distance (TTD), which are the x and y axes of the NFD, respectively. Also, the impact of weather on critical TTS is minimal compared to its impact on maximal TTD. In other words, the weather affects the y-axis of the NFD (or the flow) much more than it affects the x-axis of the NFD (or density). This was confirmed by Rakha et al., who demonstrated that traffic stream jam density is not affected by weather conditions [88]. The weather-tuned perimeter controller (WTPC) represents a proportional integral perimeter controller (PC) tuned based on weather (which affects the y-axis of the NFD). In this work, we are also interested in investigating a PC tuned based on jam density (which affects the x-axis of the NFD). Jam density tuned PC is referred as (JTPC). The PIC (referred in this section as PC) is the proportional integral perimeter control with clear weather conditions and base jam density ($k_j = 160$veh/km). PC, WTPC, and JTPC
controllers are applied to the protected region of the grid network. The No Perimeter Control Case is called (NPC).

4.3.1 PC for Clear Weather Conditions and Base Jam Density

Since INTEGRATION is a stochastic micro-simulator, simulations were run for the PC case and the NPC case using ten different random seeds. The parameters used in the simulations of the PC cases were as follows: $\mu = 0.782$, $\zeta = 0.00124$, $K_p = 631$, $K_I = 176$, $\bar{q}_{in} = 4340veh/h$, and $\bar{k} = 48.76veh/km$.

- **Performance within the PN**

  Figure 4.8 shows that congestion is reduced in the Protected Network (PN).

  ![Figure 4.8: Comparison of the NFDs for NPC and PC during clear weather conditions.](image)

- **Performance of the FN**

  Table 4.3 shows that the PC improves the full network (FN) travel time, total delay, fuel consumption and average speed compared to NPC.

Table 4.3: Performance metrics of the PC for the FN using the average of ten different seeds.

<table>
<thead>
<tr>
<th>MOE</th>
<th>System</th>
<th>NPC</th>
<th>PC</th>
<th>% Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Travel Time (s)</td>
<td></td>
<td>575.25</td>
<td>529.25</td>
<td>8.0</td>
</tr>
<tr>
<td>Total Delay (s)</td>
<td></td>
<td>213.22</td>
<td>188.84</td>
<td>11.43</td>
</tr>
<tr>
<td>Total Fuel Consumption (L)</td>
<td></td>
<td>0.40</td>
<td>0.38</td>
<td>3.5</td>
</tr>
<tr>
<td>Speed (km/h)</td>
<td></td>
<td>16.90</td>
<td>18.18</td>
<td>7.55</td>
</tr>
</tbody>
</table>
4.3.2 Weather-Tuned Perimeter Control (WTPC)

The PC uses the same set of parameters for inclement weather as for clear weather conditions: 
\[ \mu = 0.782, \ \zeta = 0.00124, \ K_p = 631, \ K_I = 176, \ q_{in} = 4340 \text{veh/h}, \text{ and } \bar{k} = 48.76 \text{veh/km}. \]
However, the WTPC uses a specific set of parameters for each weather condition. For clear weather, it uses the parameters defined above: 
\[ \mu = 0.782, \ \zeta = 0.00124, \ K_p = 631, \ K_I = 176, \ q_{in} = 4340 \text{veh/h}, \text{ and } \bar{k} = 48.76 \text{veh/km}. \] The re-tuned parameters obtained for rain conditions are 
\[ \mu = 0.588, \ \zeta = 0.00138, \ K_p = 425, \ K_I = 298, \ q_{in} = 4300 \text{veh/h}, \text{ and } \bar{k} = 48.76 \text{veh/km}. \] For snow conditions, the parameters are 
\[ \mu = 0.775, \ \zeta = 0.000139, \ K_p = 5590, \ K_I = 1620, \ q_{in} = 3900 \text{veh/h}, \text{ and } \bar{k} = 49 \text{veh/km}. \] These values demonstrate, as would be expected, that the desired controlled flow decreases for rain (4300 veh/h) and further decreases for snow conditions (3900 veh/h).

- Performance within the PN

The NFD plots of the PN for both rain and snow conditions are presented in Figure 4.9. The red, blue, and green curves correspond to the NFDs for the NPC, the PC, and the WTPC, respectively. As the curves show, the values of density \( k \) for the WTPC case were lower than those for the PC and NPC cases, especially under snow conditions. Thus, the WTPC performed better than the PC in decreasing congestion inside the PN.

![Figure 4.9: NFD for rain conditions and snow conditions.](image)
• **Performance of the FN**

The average speed, total delay, travel time and fuel consumption were calculated for the FN for the rain and snow cases (Table 4.4). We notice that using the WTPC, the improvements are better than when using the PC, especially for the snow case.

Table 4.4: Performance Metrics on the FN for the NPC, PC and WTPC Cases.

<table>
<thead>
<tr>
<th>Performance Metric</th>
<th>Strategy</th>
<th>Rain</th>
<th>Snow</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>value</td>
<td>%improv w.r.t NPC</td>
</tr>
<tr>
<td>Speed (km/h)</td>
<td>NPC</td>
<td>12.42</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>PC</td>
<td>14.12</td>
<td>13.70</td>
</tr>
<tr>
<td></td>
<td>WTPC</td>
<td>14.19</td>
<td>14.22</td>
</tr>
<tr>
<td>Total Delay (s)</td>
<td>NPC</td>
<td>315.20</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>PC</td>
<td>257.22</td>
<td>18.39</td>
</tr>
<tr>
<td></td>
<td>WTPC</td>
<td>255.73</td>
<td>18.86</td>
</tr>
<tr>
<td>Travel Time (s)</td>
<td>NPC</td>
<td>802.66</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>PC</td>
<td>690.36</td>
<td>13.99</td>
</tr>
<tr>
<td></td>
<td>WTPC</td>
<td>686.68</td>
<td>14.50</td>
</tr>
<tr>
<td>Fuel Consumption (L)</td>
<td>NPC</td>
<td>0.450</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>PC</td>
<td>0.417</td>
<td>7.05</td>
</tr>
<tr>
<td></td>
<td>WTPC</td>
<td>0.416</td>
<td>7.20</td>
</tr>
</tbody>
</table>

**4.3.3 Jam Density-Tuned Perimeter Control (JTPC)**

In this section, we applied the same principle as above but with re-tuning based on jam density rather than weather. We used clear weather with jam densities of 100, 120, 140, 160, 180, and 200 veh/km. The base case is the jam density used in the previous section (i.e., 160 veh/km/lane). These jam densities reflect a truck percentage of 29%, 20%, 13%, 7%, 3% and 0%, respectively considering that a truck is equivalent in length to 4.4 passenger car vehicles (the length of a trailer-truck is 21.3m and the length of a typical car is 4.9m). The PC case uses the set of parameters for clear weather and a jam density of 160 veh/km, as above: $\mu = 0.782$, $\zeta = 0.00124$, $K_p = 631$, $K_i = 176$, $q_{in} = 4340veh/h$, and $k = 48.76veh/km$. For the re-tuning, the set point $\bar{k}$ was changed to 12.76, 35, 25, 48.76, 50 and 51 veh/km for jam densities of 100, 120, 140, 160, 180, and 200 veh/km, respectively.
• Performance inside the PN

Figure 4.10 presents the NFDs of the PN for each jam density. The jam density of 160 veh/km is not shown in the figure because it is the base case used in the PC (clear weather plotted in Figure 4.8), so there is no re-tuning. We then have two groups of jam densities: less than 160 veh/km and higher than 160 veh/km.

Starting with the first group of jam densities, we notice that for the smallest one used (100 veh/km), the PC could not alleviate the congestion inside the PN, and grid-lock occurred. However, the JTPC did a very good job at totally avoiding congestion. For the jam densities of 120 and 140 veh/km, it is clear from Figure 4.10 that the JTPC decreased the congestion better than the PC. Concerning the second group
(jam densities of 180 and 200 veh/km), we notice that the NFDs obtained using the PC and JTPC are very similar, and both decreased the congestion compared to the NPC case. We can conclude that re-tuning using a smaller jam density was more beneficial for decreasing congestion inside the PN compared to re-tuning using a larger jam density.

- **Performance for the FN**

![Figure 4.11: Bar graphs of the performance metrics for the FN using different jam densities.](a)(b)(c)(d)

We calculated the average speed, total delay, travel time and fuel consumption for the FN using different jam densities with the NPC, PC and JTPC. Since gridlock occurred for the jam density of 100 veh/km, it was not possible to compare different performance metrics. Therefore, we did not include this jam density in the bar plots in Figure 4.11. For the jam density of 120 veh/km, gridlock was observed only for NPC, so we compared PC to JTPC. The JTPC result is not included for the jam density of 160 km/h since it is the base case. For the first group of jam densities (less than 160 veh/km), the results of the PC and NPC were similar for all performance metrics.
However, the JTPC improved all of the performance metrics. The PC outperformed the NPC for the jam density of 160 veh/km. Concerning the second group of jam densities (higher than 160 veh/km), the JTPC and PC performed similarly, and both were better than the NPC.

4.3.4 Conclusion

A proportional-Integral perimeter control strategy based on the NFD was implemented in the INTEGRATION micro-simulator. It was originally tuned based on data from clear weather conditions and a jam density of 160 veh/km/lane. The control strategy was shown to be efficient for different weather conditions. A WTPC was implemented and shown to outperform the original PC in decreasing congestion inside the PN and improving the network-wide performance of the FN. A JTPC was also implemented and tested. The JTPC performed well for jam densities less than the one used in the original PC tuning (more trucks in the network). However, its performance was similar to that of the PC for higher jam densities (less trucks in the network). Both WTPC and JTPC performed better than the NPC in decreasing congestion inside the PN and improving the overall performance of the FN. These results demonstrate the need to tune the PC to the actual conditions of the network by accounting for weather and truck effects.

4.4 Conclusion

In general, the tuning process and the estimation of the density (k) add a layer of complexity to the controller. Consequently, developing an approach that does not require tuning, does not require linearization of the control equation, and provides robustness is of interest.
Chapter 5

Sliding Mode Controller

In this work, we developed a Sliding Mode Controller SMC [25] that has comparable performance to the Proportional Integral Controller PIC, avoids the need for tuning, and makes no assumptions about the governing model (i.e., no linearization is needed for the NFD). The present effort shows that only a set point (i.e., a target network vehicle density) is needed. This value is obtained only once from the NFD. The other parameters for this new controller can be evaluated in a systematic manner.

The design of the sliding mode controller basically involves two steps: i) the design of the sliding plane to ensure stability of motion towards the origin of coordinates, referred to as the existence problem (Sliding phase in Figure 5.1), and ii) the selection of discontinuous control functions together with appropriate switching logic so that the system state is driven from an arbitrary initial condition and hastened to the sliding plane, where it continues towards the origin, referred to as the reachability problem (Reaching phase in Figure 5.1).

![Figure 5.1: Phase plane plot of sliding control system](image)

The first design step is to choose a sliding (switching) surface \( S \), to ensure stability of motion, where the system is expected to converge to and remain on.
This sliding motion occurs when the state reaches the sliding surface defined by \( S(t) = 0 \). The control that moves the state along the sliding surface is called the equivalent control \( u_{eq} \), and the dynamics of sliding motion are governed by \( \dot{S}(t) = 0 \) (i.e., necessary condition for existence of a sliding mode).

Once the sliding surface with an appropriate control signal has been selected, the second design step (reachability problem) involves the selection of a state feedback control function, called the hitting control, which can drive the state towards the surface and thereafter maintain it on the sliding surface, if the initial conditions of the system are not on \( S \).

One popular design method is to augment the equivalent control with a discontinuous or switched control, as shown in Equation 5.1, where \( u_{eq}(t) \) is a continuous control, and \( u_h(t) \) \( \text{sign}(S(t)) \) is added to satisfy the reaching condition, where \( \text{sign}(.) \) is the sign function.

\[
    u_{in}(t) = u_{eq}(t) - u_h(t) \text{sign} (S(t)) \tag{5.1}
\]

### 5.1 SMC based on Traffic Signals

#### 5.1.1 Controller Equations

This section introduces the developed SMC. Equation 4.1 from the PIC section can be rewritten as

\[
    \frac{dk(t)}{dt} = \frac{q_{in}(t) - q_{out}(t) + q_{d}(t)}{L} = u(t) - V_{out}(t) + V_{d}(t) \tag{5.2}
\]

where we assume that \( u = \frac{q_{in}}{L} \) is the system’s input. This input governs the maximum number of vehicles allowed to enter the protected area at instant \( t \) (i.e., \( q_{in} \)). For discrete time steps, \( u[n] \) (i.e., \( n \) is the corresponding time step) would determine the maximum number of vehicles allowed over a given time horizon \( \Delta t \). Equation 5.2 governs the time rate of change of the density \( k \) in the protected area. Here \( k \) is the state variable used in this work and represents the vehicle density inside the protected area. Specifically, it is the length weighted average network density computed as the sum of the product of the link densities and lengths divided by the total length of the links in the protected area.

It is important to note here that \( u(t) \) is proportional to the sum of all inflows entering the protected area through the \( m_i \) access points (i.e. access links).

\[
    u(t) = \frac{q_{in}(t)}{L} = \frac{1}{L} \sum_{i=1}^{m_i} q_{in}^i(t) \tag{5.3}
\]

To derive an expression for this input, we use sliding mode control theory [51] and introduce
the error $e$ as $e = k - \bar{k}$, where $\bar{k}$ is a set point around which the PN is desired to operate. In this case, this point corresponds to the density at capacity. We introduce the variable $x$ defined as

$$x(t) = \int_0^t e(\tau)d\tau = \int_0^t (k(\tau) - \bar{k})d\tau$$

and $S$ as

$$S = \frac{dx(t)}{dt} + \lambda x(t) = \dot{x}(t) + \lambda x(t)$$

where $\lambda$ is strictly positive real number. The sliding surface $S$ is defined in Equation (5.6). This leads to the relationship $\dot{x}(t) + \lambda x(t) = 0$. In other words, $x$ decays exponentially to zero given that $\lambda$ is strictly positive.

$$S(x(t)) = 0$$

Using (5.2), (5.4), and (5.5), we obtain

$$\frac{dS(t)}{dt} = u(t) - V_{out}(t) + V_d(t) + \lambda(k(t) - \bar{k})$$

Since the trajectories are expected to remain on the surface (i.e., Equation (5.6)) for all time, $(\frac{dS(t)}{dt})$ has to remain at zero (i.e., $\frac{dS(t)}{dt} = 0$). This in turn leads to Equation (5.8)

$$u^*(t) = V_{out}(t) - V_d(t) - \lambda(k(t) - \bar{k})$$

The values of $V_{out}(t)$ and $V_d(t)$ are not known since they represent the current outflow and disturbance flow in the network. Consequently, we use bounded estimates $\hat{V}_{out}$ and $\hat{V}_d$ with

$$\left|\hat{V}_{out}(t) - V_{out}\right| \leq \alpha$$

$$\left|\hat{V}_d(t) - V_d\right| \leq \beta$$

where $\alpha$ and $\beta$ are positive real numbers. They quantify how close the estimated outflow and disturbance flows are to the actual values at a particular instant in time. It is important to note here that the bounds might not be always available or known. In that case, adaptive SMC can be used, where the control gains are adapted dynamically to counteract the uncertainties. For further details, the reader is referred to the work of Utkin et al. [89]. In this effort, we assume that $\alpha$ and $\beta$ can be determined.

Using the previous estimates, the new estimated controller $\hat{u}$ becomes
\[ \dot{u}(t) = \hat{V}_{out}(t) - \hat{V}_d(t) - \lambda(k(t) - \bar{k}) \]  

(5.10)

It is important to note here that we chose our estimates to be in the following form: \( \hat{V}_{out}(t) = V_{out}(t - \Delta t) \) and \( \hat{V}_d(t) = V_d(t - \Delta t) \). That is, our estimates are the observed measurements of the mentioned quantities at the previous time step. This assumption requires loop detectors placed at the exits and on links in the PN.

We add to the estimated controller the term \( \gamma \text{sign}(S) \) as

\[ u_s = \hat{u} - \gamma \text{sign}(S) \]  

(5.11)

where \( \gamma \) is a positive real number, to force the controller to always move toward the sliding surface. That leads to

\[ u_s = \dot{V}_{out} - \dot{V}_d - \lambda(k(t) - \bar{k}) - \gamma \text{sign}\left(k(t) - \bar{k} - \frac{1}{2} \int_0^t (k(\tau) - \bar{k}) d\tau\right) \]  

(5.12)

for a discrete time step \( n \)

\[ u_s[n] = \dot{V}_{out} - \dot{V}_d - \lambda(k[n] - \bar{k}) - \gamma \text{sign}\left(k(n) - \bar{k} - \lambda \sum_{i=0}^{n} (k[i] - \bar{k}) \Delta t\right) \]  

(5.13)

To ensure the surface defined by Equation (5.6) is a stable surface for the chosen controller Equation (5.12), we introduce the Lyapunov function, defined by \( LF(S) \).

\[ LF(S(x)) = \frac{1}{2} S(x)^T S(x) = \frac{1}{2} \|S(x)\|^2 \]  

(5.14)

The equilibrium (5.6) is stable if

\[ \frac{d}{dt}(LF(S)) \leq 0 \]  

(5.15)

Using Equation (5.7) we derive

\[ \frac{d}{dt} \left( LF(S) \right) = S(u(t) - u_s(t)) + S u_s(t) - S V_{out}(t) + S V_d(t) + \lambda S \left(k(t) - \bar{k}\right) \]  

(5.16)

Using Equations (5.10) and (5.11) we compute

\[ \frac{d}{dt} \left( LF(S) \right) = S \left( \dot{V}_{out}(t) - V_{out}(t) \right) + S \left( V_d(t) - \dot{V}_d(t) \right) - \gamma |S| \]  

(5.17)
Using Equation (5.9) we obtain
\[
\frac{d}{dt} \left( LF(S) \right) \leq \alpha \ |S| + \beta \ |S| - \gamma \ |S| \tag{5.18}
\]

Choosing \( \gamma = \alpha + \beta + \eta \) (with \( \eta \) strictly positive) leads to
\[
\frac{dLF(S)}{dt} \leq -\eta \ |S| \tag{5.19}
\]

In other words, the distance to the sliding surface decreases with time. This sliding surface \( S = \dot{x} + \lambda \ x = 0 \) will be reached in a finite time that is bounded by \( |S(t = 0) / \eta| \). Consequently, the choice of \( \eta \) will impact the time it takes the system to reach the sliding surface. Once on the surface, the system will remain there and will slide to the desired state, which in this case is \( x = 0 \), exponentially with a time constant \( \lambda^{-1} \). It is important to note here that the value of \( k[n] \) is unknown at the current time step \( n \). Therefore, quantities measured from the previous time steps were used as shown in equation (5.20).

\[
\hat{V}_{out} [n] = q_{out} [n - 1] / L \\
\hat{V}_{d} [n] = q_{d} [n - 1] / L 
\]

Also, the value of \( k[n] \) was replaced with \( k[n - 1] \). Hence, the controller becomes

\[
u[n] = \frac{q_{out} [n - 1]}{L} - \frac{q_{d} [n - 1]}{L} - \lambda \left( k [n - 1] - \bar{k} \right) \\
- (\alpha + \beta + \eta) \ \text{sign} \left( k [n - 1] - \bar{k} - \lambda \sum_{i=0}^{n-1} \left( k [i] - \bar{k} \right) \Delta t \right) \tag{5.21}
\]

It is important to note here that the SMC suffers from a known problem—control/input chattering. This can be solved by changing the “\( \text{sign()} \)” function in equation (5.11) into a saturation “\( \text{sat()} \)” function or a “\( \text{tanh()} \)” function. Chattering usually results in the failure of components in mechanical systems. However, in this research, the values of the inflow allowed in the protected area are transformed into green times (equation (5.22)). For each new cycle \( \Delta t \) (assumed to be 60 s in this work) a new value of the green time is obtained. Chattering in this case will result in the change in the green time by a large or small value from one cycle to the next. This change is expected not to cause a failure of any component and thus is less of an issue in this application. In contrast to mechanical systems, chattering (i.e., sudden change of the control input) might cause the failure of components, for instance a sudden variation of the velocity input signal to a motor might damage it.

For the controller to function in an ideal manner, detectors are needed on the majority of the links of the protected area as well as at the entrance points. This is also true for the PIC to
monitor the state of the area in terms of associated vehicle density, inflow, and outflow. This information can then be used to determine the estimated values needed by the controller (i.e., \( \tilde{V}_{\text{out}}(t) \) and \( \tilde{V}_{\text{d}}(t) \)) (equation 5.20).

It is important to mention here that inflow computed by the controller \( q_{\text{in}} \) is transformed into green times \( G \) that are allocated to the different approaches feeding into the protected area. This is performed using equation (5.22).

\[
G = C \frac{q_{\text{in}}}{q_s}
\]  

(5.22)

where \( q_s \) is the saturation flow rate and \( C \) is the traffic signal cycle length.

### 5.1.2 Performance of the SMC and PIC for Different Parameters

Tuning is required for the PIC. This requirement is not needed for the SMC. Only two parameters, \( \lambda \) and \( \eta \), are considered. \( \eta \) governs how fast the system converges to the sliding surface. \(|S(x(t=0))/\eta|\) is the maximum time it takes for convergence to occur. Since we choose the control to be activated when \(|(k - \bar{k})/k| \leq 0.15 \) (i.e., control is activated when \( k \) reaches \( 0.85\bar{k} \)), the value of \(|S(x(t=0))/\eta|\) is small for a wide range of \( \eta \).

\[
\left| \frac{S(x(t=0))}{\eta} \right| \leq 0.15 \frac{\bar{k}}{\eta}
\]

(5.23)

Once on the surface (i.e. \( \dot{x} + \lambda x = 0 \)), the system converges exponentially to zero with a time constant of \( \lambda^{-1} \) with the requirement that \( \lambda \) must be less than the minimum frequency of the un-modeled dynamics of the system [90]. This means that we can choose \( \lambda \) as small as adequately possible.

Table 5.1 presents various responses of the system using different control parameters for the PIC and SMC for the traffic demand presented in Figure 5.4(a). A negative change means a decrease in value with respect to the base case (no control case), while a positive change indicates an increase in value. For the SMC, we tested various combinations for the control parameters. In all cases we get an improvement with respect to the base case. This holds true for the five carefully tuned cases of the PIC.
Table 5.1: Sensitivity of SMC and PIC with respect to NPC (base case) for various values of controller parameters $\lambda, \eta, \mu$, and $\zeta$.

<table>
<thead>
<tr>
<th>SMC</th>
<th>$\lambda$</th>
<th>$\eta$</th>
<th>TT Change (%)</th>
<th>Delay Change (%)</th>
<th>Fuel Change (%)</th>
<th>Speed Change (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.5</td>
<td>2.0</td>
<td>-12.31</td>
<td>-12.88</td>
<td>-4.63</td>
<td>11.86</td>
</tr>
<tr>
<td>2</td>
<td>1.5</td>
<td>20.0</td>
<td>-12.92</td>
<td>-15.57</td>
<td>-5.75</td>
<td>12.06</td>
</tr>
<tr>
<td>3</td>
<td>1.5</td>
<td>200.0</td>
<td>-8.33</td>
<td>-7.90</td>
<td>-2.65</td>
<td>7.93</td>
</tr>
<tr>
<td>4</td>
<td>15.0</td>
<td>2.0</td>
<td>-13.19</td>
<td>-14.56</td>
<td>-4.77</td>
<td>13.46</td>
</tr>
<tr>
<td>5</td>
<td>15.0</td>
<td>20.0</td>
<td>-12.61</td>
<td>-15.82</td>
<td>-5.41</td>
<td>12.23</td>
</tr>
<tr>
<td>6</td>
<td>15.0</td>
<td>200.0</td>
<td>-12.56</td>
<td>-19.51</td>
<td>-7.11</td>
<td>11.30</td>
</tr>
<tr>
<td>7</td>
<td>150.0</td>
<td>2.0</td>
<td>-8.99</td>
<td>-12.84</td>
<td>-4.58</td>
<td>7.90</td>
</tr>
<tr>
<td>8</td>
<td>150.0</td>
<td>20.0</td>
<td>-7.90</td>
<td>-13.24</td>
<td>-4.48</td>
<td>7.00</td>
</tr>
<tr>
<td>9</td>
<td>150.0</td>
<td>200.0</td>
<td>-4.62</td>
<td>-6.08</td>
<td>-2.07</td>
<td>4.02</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PIC</th>
<th>$\mu$</th>
<th>$\zeta$</th>
<th>TT Change (%)</th>
<th>Delay Change (%)</th>
<th>Fuel Change (%)</th>
<th>Speed Change (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.801</td>
<td>0.002</td>
<td>-10.63</td>
<td>-14.75</td>
<td>-5.294</td>
<td>9.63</td>
</tr>
<tr>
<td>2</td>
<td>0.547</td>
<td>0.003</td>
<td>-8.451</td>
<td>-10.54</td>
<td>-3.506</td>
<td>8.08</td>
</tr>
<tr>
<td>3</td>
<td>0.885</td>
<td>0.003</td>
<td>-12.5</td>
<td>-16.1</td>
<td>-5.7</td>
<td>11.76</td>
</tr>
<tr>
<td>4</td>
<td>0.747</td>
<td>0.003</td>
<td>-11.53</td>
<td>-15.36</td>
<td>-5.29</td>
<td>11.03</td>
</tr>
<tr>
<td>5</td>
<td>0.847</td>
<td>0.002</td>
<td>-12.95</td>
<td>-17.52</td>
<td>-6.25</td>
<td>12.12</td>
</tr>
</tbody>
</table>

Figure 5.2(a) and Figure 5.2(b) present a comparison between the evolution of the density $k$ inside the protected area in the presence and absence of control. In Figure 5.2(a), the five tuned cases of the PIC were picked, and in Figure 5.2(b) the best performing SMC cases in terms of travel time reduction were picked. It is important to mention here that for the SMC the density in the protected area is generally below the target density $\bar{k}$ with the exception of SMC 6, which is slightly above $\bar{k}$ for the first few time steps when the control is activated. PIC 1 and 2 are clearly above the target value by a significant margin and over the entire duration of the control interval (i.e., the time steps where the density is above 0.85 $\bar{k}$). This is clear for PIC 1, which exceeds the NPC case in some time steps (i.e., congestion forms in the protected area). This in turn underscores the criticality of the tuning process and its importance in the PIC performance. In the rest of the work, and since the performance of most of the various controllers with different parameters is almost the same with respect to the reduction in travel time (TT in Table 5.1), the parameters of SMC 6 and PIC 5, which deliver the highest reduction in delay, will be used.
Chapter 5. Sliding Mode Controller

(a) PIC using the tuned parameters.  (b) SMC control for the five best cases are shown.

Figure 5.2: Time series of the density $k$ from the protected area.

Fig. 5.3 shows a comparison between the NFDs of two controllers: the proposed SMC and the PIC for the demand profile D1. The plot shows similar performance between the two controllers. The tuned parameters for the PIC are taken to be $\mu = 0.847$ and $\zeta = 0.002$ (i.e., $K_I = 73.7$, $K_P = 408$), and $\bar{k} = 48.76\text{veh/km})$. For the SMC, the parameters are $\lambda = 15$ and $\eta = 200$.

Figure 5.3: NFD of the grid subnetwork without and with control.

In the absence of control, we notice that the average network density of the protected area exceeds the optimum value $\bar{k}$. Once the control is activated, congestion is consistently eliminated from the protected area.
5.1.3 Response to Different Demand Profiles

To further demonstrate the effectiveness of the SMC, we considered other demand profiles for this study. These are shown in Figure 5.4. Each demand period spans 300 s. Figure 5.4(b) shows a dome-shaped demand (i.e., demand D2), and Figure 5.4(c) shows a sinusoidal demand profile (i.e., demand D3).

![Figure 5.4: Network demand profiles.](image)

The evolution of the vehicle density for the different demands D1, D2, and D3 in the protected area is presented in Figures 5.5(a), 5.5(b), and 5.5(c), respectively. It is important to note here that both controllers start from the $0.85\bar{k}$ threshold to regulate the density inside the protected area to values around $\bar{k}$.

![Figure 5.5: Evolution of the density inside the protected area for different demand profiles without and with control.](image)

Table 5.2 presents the change in travel time, delay, fuel consumption, and speed with respect to a base case for the PIC and SMC for the demand profiles D1, D2, and D3. The results show the improvement percentages of the mean values for each of the performance metrics. The different control methods could be tested with more seeds and a variance calculation could be added to confirm the superiority of one control method over the other.

For the demand profile D1, we notice that both controllers have quite similar performance, with one or the other exceeding slightly in one or two measures (i.e., for instance, the PIC is slightly better in travel time with respect to the SMC). For demand profile D2, we notice a slight advantage of the SMC with respect to the PIC. This advantage is clear in demand D3. The SMC outperforms the PIC by a relatively significant margin in the case of D3. This
shows that the SMC adapts to a changing demand pattern whereas the PIC is less adaptable. It should be noted here that dynamic re-routing is activated. Therefore, the demands at the various traffic signals might vary.

Table 5.2: Average Simulation Results for the Demand Profiles of Figure 5.4

<table>
<thead>
<tr>
<th>Demand Profile</th>
<th>TT(s)</th>
<th>Delay (s)</th>
<th>Fuel (L)</th>
<th>Speed (veh/km)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Demand Profile D1</td>
<td>NPC</td>
<td>659.26</td>
<td>247.28</td>
<td>0.42</td>
</tr>
<tr>
<td></td>
<td>SMC Change (%)</td>
<td>-12.56</td>
<td>-19.51</td>
<td>-7.11</td>
</tr>
<tr>
<td></td>
<td>PIC Change (%)</td>
<td>-12.95</td>
<td>-17.52</td>
<td>-6.25</td>
</tr>
<tr>
<td>Demand Profile D2</td>
<td>NPC</td>
<td>956.11</td>
<td>401.66</td>
<td>0.51</td>
</tr>
<tr>
<td></td>
<td>SMC Change (%)</td>
<td>-20.34</td>
<td>-26.11</td>
<td>-11.36</td>
</tr>
<tr>
<td></td>
<td>PIC Change (%)</td>
<td>-18.76</td>
<td>-27.04</td>
<td>-11.12</td>
</tr>
<tr>
<td>Demand Profile D3</td>
<td>NPC</td>
<td>950.76</td>
<td>415.29</td>
<td>0.52</td>
</tr>
<tr>
<td></td>
<td>SMC Change (%)</td>
<td>-14.23</td>
<td>-25.16</td>
<td>-11.43</td>
</tr>
<tr>
<td></td>
<td>PIC Change (%)</td>
<td>-8.28</td>
<td>-17.78</td>
<td>-7.37</td>
</tr>
</tbody>
</table>
5.2 Sliding Mode Controller based on Speed Harmonization

The sliding mode control (SMC) theory was coupled with speed harmonization (SH) in order to apply a perimeter control strategy on a network level. The approach uses connected automated vehicles (CAVs) technology to gather data and as a control mechanism. Instead of controlling traffic signal lights on the gated links, the speed of the CAVs entering the network through the gated links were controlled, making the controller vehicle-centric. This makes the system independent from infrastructure and consequently reduces the system cost.

5.2.1 Model and Controller Equations

This subsection derives the relationship between the PN density and the entering vehicles’ speed.

The model equation is the same equation used in the previous section (equation 5.2). Rewriting it in the form of time state space, we obtain:

\[ \dot{k}(t) = g(k, t)q_{in} + f(k, t) \] (5.24)

where \( g(k, t) = \frac{1}{L} \) and \( f(k, t) = -\frac{q_{out}}{L} + \frac{q_d}{L} \)

In the previous section, SMC used two different parameters \( \lambda \) and \( \eta \). In order to simplify more the equations, the sliding surface is chosen differently (equation 5.25), so that at the end, only one parameter is needed for this controller.

\[ S(k(t)) = k(t) - \bar{k} = 0 \] (5.25)

Using equations (5.24) and (5.25), we obtain

\[ S(t) = \dot{k}(t) = g(k, t)q_{in} + f(k, t) \] (5.26)

The dynamics of sliding motion are governed by \( \dot{S} = 0 \) (i.e., necessary condition for existence of a sliding mode).

\[ \dot{S} = 0 \Rightarrow f + gq_{in} = 0 \] (5.27)

The control that moves the state along the sliding surface is called the equivalent control \( (q_{in}|S=0) \), and is obtained by equation (5.28)

\[ q_{in}|S=0 = -g^{-1}f = -L\frac{q_{out}(t) + q_d(t)}{L} = q_{out}(t) - q_d(t) \] (5.28)
Once the sliding surface with an appropriate control signal has been selected, the second stage (reachability problem) of the design procedure involves the selection of a state feedback control function, called the hitting control, which can drive the state towards the surface and thereafter maintain it on the sliding surface. One popular design method is to augment the equivalent control with a discontinuous or switched part as

\[ q_{in} = q_{in}|S=0 - \gamma \text{sign}(S) \] (5.29)

where \( \gamma \) is a positive real number. That leads to

\[ q_{in} = q_{out}(t) - q_d(t) - \gamma \text{sign}(k(t) - \bar{k}) \] (5.30)

To ensure that the surface defined by equation (5.25) is a stable surface for the chosen controller (5.30), we introduce the Lyapunov function defined by \( LF(S) \)

\[ LF(S(x)) = \frac{1}{2} S(x)^T S(x) = \frac{1}{2} \|S(x)\|^2 \] (5.31)

The equilibrium (5.25) is stable if

\[ \frac{d}{dt}(LF(S)) \leq 0 \] (5.32)

Using equation (5.26)

\[ \frac{d}{dt}(LF(S)) = S.\dot{S} = S.(g.q_{in} + f) \] (5.33)

Using equation (5.30)

\[ \frac{d}{dt}(LF(S)) = S.(g.(q_{out}(t) - q_d(t) - \gamma \text{sign}(k(t) - \bar{k}))) + f) \] (5.34)

Substituting \( g \) and \( f \), we obtain

\[ \frac{d}{dt}(LF(S)) = S.\left(\frac{1}{L}.(q_{out}(t) - q_d(t) - \gamma \text{sign}(k(t) - \bar{k})) + \frac{-q_{out}(t) + q_d(t)}{L}\right) \]

\[ = S.(-\frac{\gamma}{L} \text{sign}(k(t) - \bar{k})) \]

\[ = -\frac{\gamma}{L} S.\text{sign}(S) \]

\[ = -\frac{\gamma}{L} |S| \] (5.35)

Let \( \eta = \frac{\gamma}{L} \) be a positive number. Equation (5.35) becomes

\[ \frac{d}{dt}(LF(S)) \leq -\eta |S| \] (5.36)
Consequently, the stability of the equilibrium was proven, and the total controlled flow entering the protected sub-network is calculated using equation (5.30). Substituting $\gamma$, the controller input flow could be calculated by

$$q_{in}(t) = q_{out}(t) + q_d(t) - L.\eta . \text{sign}(S)$$  \hspace{1cm} (5.37)

For a discrete time step $n$, we obtain

$$q_{in}(n) = q_{out}(n) + q_d(n) - L.\eta . \text{sign}(k(n) - \bar{k})$$  \hspace{1cm} (5.38)

The calculated flow is then divided by $nb_{\text{gatedLinks}}$, the number of gated links (controlled links entering the protected sub-network) to obtain the flow that should enter each gated link, as shown in the following equation:

$$q_{in,\text{link}} = \frac{q_{in}}{nb_{\text{gatedLinks}}}$$  \hspace{1cm} (5.39)

In order to account for the traffic signal effect, a multiplication of the flow by the cycle length of the signal $C$ over the green time $G$ is necessary. We obtain the following equation:

$$q_{\text{link}} = q_{in,\text{link}} * C/G$$  \hspace{1cm} (5.40)

Recall that the objective is to control the vehicles’ speed, and therefore, the speed on the controlled links was calculated using the hydrodynamic equation:

$$\text{speed}_{\text{link}} = \frac{q_{\text{link}}}{k_{\text{link}}}$$  \hspace{1cm} (5.41)

where $k_{\text{link}}$ is the density of the controlled link.

### 5.2.2 Implementation and Results

The SMC-SH is implemented on the grid network. The network has eight access points called gated links, each of which is 900 m in length. On the right of Figure 5.6, a zoomed controlled gated link is presented.
Chapter 5. Sliding Mode Controller

The controlled Distance $CD$ is where the speeds of the vehicles are controlled. It is calculated as twice the minimum of two different distances calculated from the middle of the link as

$$CD = 2 \times \min(Ratio_{LinkDistance} \times length_{link}, Range_{LinkDistance})$$

(5.42)

where

- $Ratio_{LinkDistance}$: controlled distance as a ratio of the link length from the middle of the link.
- $Range_{LinkDistance}$: maximum controlled distance in kilometers from the middle of the link.

It is already known from previous sections that the set point $\bar{k}$ of the NFD; which corresponds to the density where the throughput is maximum; is 48.76 veh/km. The controller is activated before reaching the set point, specifically when the protected network density is greater than or equal to $Density_{percentage} \times \bar{k}$. The calculation of vehicles'speed is done at every update interval $T$.

After performing sensitivity analysis using Particle Swarm Optimization [91], the following parameters were used: $T = 60s$, $Ratio_{LinkDistance} = 0.39$, $Range_{LinkDistance} = 0.436km$, $Density_{percentage} = 0.5$, $\eta = 0.277$

Using the Integration micro-simulator, different measures of effectiveness (MOE) are calculated. Table 5.3 shows the improvement percentage of the sliding mode speed harmonization (SMC-SH) over the no perimeter control case (NPC) inside the protected area. Results of the whole network (Protected and non-protected areas) are shown in Table 5.4.
5.3 Conclusion

In this work, a SMC that attempts to alleviate traffic congestion in regions within large urban networks was developed. This controller was applied to the traffic stream flow continuity equation that governs the density of vehicles on any given link and thus a congested region. The controller was tested on a grid network and the results suggest that it has similar performance to the PIC and in some cases outperforms it. The SMC, however, does offer additional benefits, namely it makes no assumptions with regards to 1) the form or simplifications of the governing equations, 2) the existence or nonexistence of the NFD, or 3) the shape of the NFD (i.e., linearization of the NFD around the set point, then calibrating the PIC gains using the linearized data). This controller also has consistent performance with varying demand patterns. Unlike the PIC, it requires no tuning, but its parameters need to be within a specified range. The user decides how fast the system should converge to the sliding surface, then once on the sliding surface how fast it converges to the desired control (error of zero). The controller consists of a single equation with various inputs that

Table 5.3: Percentage Improvements Using SMC-SH Over NPC for Protected sub-Network

<table>
<thead>
<tr>
<th>System</th>
<th>Improvement SMC-SH/NPC (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average link Travel Time</td>
<td>15.17</td>
</tr>
<tr>
<td>Average Queued Vehicles</td>
<td>18.22</td>
</tr>
<tr>
<td>Total Fuel Consumption</td>
<td>6.68</td>
</tr>
<tr>
<td>Total CO₂ Emissions</td>
<td>6.71</td>
</tr>
</tbody>
</table>

Table 5.4: Network wide results MOEs and (%) Improvement Using SMC-SH Over NPC

<table>
<thead>
<tr>
<th>System</th>
<th>NPC</th>
<th>SMC-SH</th>
<th>Improvement (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average Travel Time</td>
<td>757.44</td>
<td>626.65</td>
<td>17.27</td>
</tr>
<tr>
<td>Total Delay</td>
<td>299.42</td>
<td>244.97</td>
<td>18.18</td>
</tr>
<tr>
<td>Stopped Delay</td>
<td>144.85</td>
<td>126.38</td>
<td>12.76</td>
</tr>
<tr>
<td>Acceleration/Deceleration</td>
<td>154.57</td>
<td>118.6</td>
<td>23.27</td>
</tr>
<tr>
<td>Fuel</td>
<td>0.45</td>
<td>0.42</td>
<td>5.91</td>
</tr>
<tr>
<td>CO₂(g)</td>
<td>1029.38</td>
<td>956.89</td>
<td>7.04</td>
</tr>
</tbody>
</table>
are collected from the detectors assumed to be available on all links within and feeding to the congested region.

SMC has two controller parameters to be determined. Also, it calculates traffic flow that needs to enter the protected network and converts it to green times in order to apply it on gated links traffic signals. In order to improve on that, SMC based on speed harmonization (SMC-SH) was developed. The controller has only one variable, and instead of adjusting traffic lights, a vehicle-centric approach is used. Using the connected and automated vehicles (CAVs) technology, vehicles could adjust their speeds according to the SMC-SH controller calculations. This reduces the system cost since it is independent from the infrastructure. Results of simulations of SMC-SH on the grid network show great improvements over the NPC case. Improvements reached 17.27% for average travel time and 18.18% for total delay.
Chapter 6

Freeway Speed Control

In order to solve the freeway congestion problem on LA network, a variable speed limit (VSL) or speed harmonization (SH) controller is developed using sliding mode control (SMC).

The developed controller was implemented and evaluated on the downtown Los Angeles road network, which is composed of multiple connected freeways and signalized arterials. The testing provides a large-scale dynamic evaluation of the system. The proposed controller was implemented in the INTEGRATION microscopic traffic assignment and simulation software [80].

The Freeway-SMC-SH controller was allowed to operate on all freeway links (a total of 331 links), and its operation was compared to the no SH control base case. The proposed controller is a dynamic and adaptable solution to morphing and location-changing bottlenecks. The performance of the proposed controller was evaluated on a network level (including all freeways and signalized arterial roadways) and along the freeway. The contributions of this work are as follows:

- The dynamic identification of bottlenecks along freeways;
- A simple sliding mode control is used to apply the SH control algorithm;
- The problem formulation and the solution are based on the space mean speed, which may be obtained from probe vehicles (CVs)—no road sensors are needed;
- The developed methodology is dynamic and adapts to the conditions of the network (i.e., the locations of the bottlenecks are not needed to be known a priori);
- The impacts are studied on a large real world network composed of multiple connected freeways and signalized urban roads.
6.1 Proposed Model & Controller Formulation

6.1.1 Model Formulation

In this section, the developed governing equations employed in the proposed Freeway-SMC-SH controller are presented. The Freeway-SMC-SH controller intends to maintain a desired average space mean speed on the given network links. For a given link $l$ (Figure 6.1), the time rate of change of the density, assuming there is no disturbance flow, is given by Equation 6.1

$$\frac{dk_l(t)}{dt} = \frac{q_{in,l}(t)}{L_l} - \frac{q_{out,l}(t)}{L_l}$$

(6.1)

where, $k_l$ is the density (i.e., number of vehicles per unit length on the link $l$), $L_l$ is the length of link $l$, $q_{in,l}$ is the inflow of vehicles into the link $l$, and $q_{out,l}$ is the outflow of vehicles from the link.

Using the Van Aerde fundamental diagram [92], the density $k_l$ can be expressed as a function of the link parameters and the current space mean speed on the link (Equation 6.2)

$$k_l(t) = \frac{1}{c_1 + \frac{c_2}{u_{f,l} - u_l(t)} + c_3 u_l(t)}$$

(6.2)

where $u_{f,l}$ is the free flow velocity on link $l$, $u_l(t)$ is the current space mean speed of vehicles on link $l$ at time $t$, and the constants $c_1, c_2, c_3$ [92] are expressed below:

$$c_1 = \frac{u_{f,l}}{k_{j,l} u_{c,l}^2} \left(2 u_{c,l}^2 - u_{f,l}\right), \quad c_2 = \frac{u_{f,l}}{k_{j,l} u_{c,l}^2} \left(u_{c,l}^2 - u_{f,l}\right)^2$$

$$c_3 = \frac{1}{q_c} - \frac{u_{f,l}}{k_{j,l} u_{c,l}^2}$$
Using Equation 6.2, the time rate of change of the density $k_l$ is given by Equation 6.3

$$\frac{dk_l(t)}{dt} = \alpha(u_l(t)) \frac{du_l(t)}{dt}$$

where

$$\alpha(u_l(t)) = \frac{-c_2 - c_3(u_{f,l} - u_l(t))^2}{(u_{f,l} - u_l(t))^2 \left(c_1 + \frac{c_2}{u_{f,l} - u_l(t)} + c_3 u_l(t)\right)^2}$$

(6.3)

To find the system state equation ($\frac{du_l(t)}{dt}$), two assumptions were made [17, 93]: (1) the outflow of link $l$ is proportional to the average flow on link $l$, and (2) the inflow into link $l$ is equal to the outflow of link $l-1$. These assumptions are expressed in Equations 6.4 and 6.5.

$$q_{out,l}(t) = A.q_l(t) = A.k_l(t).u_l(t), \text{ where } 0 \leq A \leq 1$$

(6.4)

$$q_{in,l}(t) = q_{out,l-1}(t) = A.q_{l-1}(t) = A.k_{l-1}(t).u_{l-1}(t) = A.k_{l-1}(t).u_{in}(t)$$

(6.5)

Noting that $u_{l-1}(t)$ was labeled $u_{in}(t)$ (i.e., $u_{l-1}(t) = u_{in}(t)$), which will be considered as an input to the system at a later stage.

Using Equations 6.1, 6.3, 6.4 and 6.5, Equation 6.1 can be re-written as shown in Equations 6.6 and 6.7.

$$L_l.\alpha(u_l(t)) \frac{du_l(t)}{dt} = q_{in,l}(t) - q_{out,l}(t) = A.k_{l-1}(t).u_{in}(t) - A.k_l(t).u_l(t)$$

(6.6)

$$\frac{du_l(t)}{dt} = -\frac{A.k_l(t)}{\alpha(u_l(t)) L_l}.u_l(t) + \frac{A.k_{l-1}(t)}{\alpha(u_l(t)) L_l}.u_{in}(t)$$

(6.7)

Assuming that

$$f(u_l(t)) = -\frac{A.k_l(t)}{\alpha(u_l(t)) L_l}.u_l(t) \quad \text{and} \quad g(u_l(t)) = \frac{A.k_{l-1}(t)}{\alpha(u_l(t)) L_l}$$

An ordinary differential equation governing the time rate of change of the space mean speed ($u_l$) on the link $l$ with respect to an input $u_{in}$ (i.e., the space mean speed of vehicles in the link $l-1$) is shown in Equation 6.8.

$$\frac{du_l(t)}{dt} = f(u_l(t)) + g(u_l(t)) \ u_{in}(t)$$

(6.8)
The nonlinear state equation (Equation 6.8) is employed in the controller (as described in following subsection) to determine the appropriate vehicle speeds \( u_{in} \) in the speed harmonization zone (Figure 6.1), so that the speed of the vehicles on the downstream link \( u_l \) are regulated at the speed of capacity.

6.1.2 Controller Formulation

The system block diagram is shown in Figure 6.2 with an objective to control vehicle speeds \( u_{in} \) in the speed harmonization zone on link \( l-1 \) (Figure 6.1) so that the vehicles’ space mean speed \( u_l \) on the downstream link \( l \) are regulated at a specific set-point; i.e., the speed at capacity \( \bar{u}_l \). The feedback speed \( u_l \) can be measured using information provided by the CVs, and estimated if not all vehicles are connected. The advisory speed \( u_{in} \) can be displayed as a dynamic message on roadway signs, and/or sent to CVs. This study assumes that all the vehicles are connected.

The central idea of sliding mode control is to apply the control action when the system deviates from the desired behavior. The controller is allowed to change its structure; it can switch at any instant from one state to another, and doesn’t require accurate mathematical dynamic system modeling. Sliding mode control provides excellent performance in the presence of modeling uncertainties and disturbances [94].

The sliding surface is described below:

\[
S(t) = u_l(t) - \bar{u}_l = 0
\]  
(6.9)

Solving \( \dot{S}(t) = \dot{u}_l(t) = 0 \) for the control input \( u_{in}(t) = u_{eq}(t) \) yields:

\[
u_{eq}(t) = -g \left( u_l(t) \right)^{-1} f \left( u_l(t) \right)
= -\frac{\alpha \left( u_l(t) \right)}{A.k_l(t)} L_l \frac{u_l(t)}{\alpha(u_l(t))} L_l
= \frac{k_l(t)}{k_{l-1}(t)} u_l(t)
\]

(6.10)
Now, we augment the equivalent control with a discontinuous or switched control, as shown in Equation 6.11
\[ u_{in}(t) = u_{eq}(t) - u_h(t) \text{sign} (S(t)) \] (6.11)

The hitting control will be responsible for pointing the system toward the sliding surface and depleting its energy (Lyapunov’s second method [95]); this is translated through Equation 6.12
\[ S(t) \cdot \dot{S}(t) \leq -\eta |S(t)| \] (6.12)

where \( \eta \) is a positive real number.

\[ S(t) \cdot [f(u_l(t)) + g(u_l(t)) u_{in}(t)] \leq -\eta |S(t)| \] (6.13)

namely,
\[ S(t) \cdot [f(u_l(t)) + g(u_l(t)) (u_{eq}(t) - u_h(t) \text{sign} (S(t)))] \leq -\eta |S(t)| \] (6.14)

which results in
\[ u_h(t) \geq \eta g(u_l(t))^{-1} = \eta \frac{\alpha(u_l(t)) L_l}{A. k_{l-1}(t)} \] (6.15)

A convenient choice for \( u_h(t) \) is given by Equation 6.16
\[ u_h(t) = \eta \frac{\alpha(u_l(t)) L_l}{A. k_{l-1}(t)} \] (6.16)

Substituting \( u_{eq}(t) \) (Equation 6.10) and \( u_h(t) \) (Equation 6.16) in Equation 6.11 yields to Equation 6.17. \( u_{in}(t) \) is expressed as a function of the density of the link in control \( (k_{l-1}) \), the density of the downstream link \( (k_l) \), and the velocity of the downstream link \( (u_l) \).
\[ u_{in}(t) = \frac{k_l(t)}{k_{l-1}(t)} u_l(t) - \eta \frac{\alpha(u_l(t)) L_l}{A. k_{l-1}(t)} \text{sign} (u_l(t) - \bar{u}_l) \] (6.17)

where \( \alpha(u_l(t)) \) is given by Equation 6.3.
For a discrete time step \( n \), \( u_{in}(n, \Delta t) \) is written as \( u_{in}[n] \) in Equation 6.18, where \( \Delta t \) is the time step duration and is fixed a priori.

\[
    u_{in}[n] = \frac{k_l[n]}{k_{l-1}[n]} u_l[n] - \eta \left( \frac{\alpha(u_l[n]) L_l}{A k_{l-1}[n]} \right) \text{sign}(u_l[n] - \bar{u}_l)
\]  

(6.18)

To maintain vehicles’ space mean speed on the link \( l \) at the speed at capacity, the vehicle speeds in the speed harmonization zone on the upstream link \((l-1)\) have to follow Equation 6.18. To prevent the creation of artificial congestion on the upstream link \((l-1)\), we opted for a localized application. The computed speed \( u_{in}[n] \) will be enforced on a small specific region on link \((l-1)\). This region is referred to as the “speed harmonization zone,” as shown in Figure 6.1, and has a length \( D \) given by Equation 6.19.

\[
    D = \min(d, \; R_d \times \text{length(link } l-1))
\]  

(6.19)

where \( d \) is the maximum speed harmonization zone length, and \( R_d \) is a ratio of the link length. Note that the speed harmonization zone is always centered at the middle of the link.

The Freeway-SMC-SH controller is activated on the speed harmonization zone on the upstream link once the density on the downstream link is within a specific range, as outlined in the following equation:

\[
    R_{k_{\text{min}}} \times k_{j,l} \leq k_l \leq R_{k_{\text{max}}} \times k_{j,l}
\]  

(6.20)

where \( k_{j,l} \) is the jam density on the link \( l \) and \( R_{k_{\text{min}}} \) and \( R_{k_{\text{max}}} \) are given parameters. The density \( (k_l) \) can be measured using information provided by the CVs or estimated if not all vehicles are connected [96]. This activation condition ensures that the Freeway-SMC-SH controller is only activated when the downstream link starts to become congested in order to avoid artificial congestion (needless reduction in vehicles speeds) on the upstream link.

## 6.2 Simulation Setup & Results

This section describes the simulation setup and results of a large scale study on a real network in downtown Los Angeles, California, to control vehicle speed on freeway links using the SH controller.

### 6.2.1 Simulation Setup

The microscopic simulations were conducted on a large network—the downtown LA area—as shown in Figure 6.3(a). The controller is applied on the 331 freeway links, colored in red in
6.2. Simulation Setup & Results

Figure 6.3(b).

![Downtown Los Angeles network](image)

All traffic signals in the network are optimized using a decentralized phase split and cycle length controller [97]. The Freeway-SMC-SH controller was applied on freeways only and its operation was compared to the base no-control case.

Simulations were conducted using the calibrated morning peak hour traffic (7:00 – 8:00 a.m.). Vehicles were loaded for one hour and were given extra time at the end of the simulation to guarantee that all vehicles departed the network. This was to ensure an equal number of vehicles were present when comparing the performance of the proposed controller to the benchmark case.

Driver characteristics such as reaction times, acceleration and deceleration rates, desired speeds, and lane-changing behavior are examples of stochastic variables that are incorporated in the INTEGRATION software.

6.2.2 Simulation Scenarios

Several simulations were conducted using different control parameter values to study their effect on the performance of the proposed Freeway-SMC-SH controller; however, only three scenarios (SH-1, SH-2, SH-3) are described below, with $\Delta t = 3 \text{ s}$, $\eta = 1$, $R_{kmin} = 0.5$, $R_{kmax} = 0.9$:

- SH-1 $\Rightarrow R_d = 0.1, d = 10 \text{ m}, A = 1$
- SH-2 $\Rightarrow R_d = 0.06, d = 6 \text{ m}, A = 0.8$
- SH-3 $\Rightarrow R_d = 0.04, d = 4 \text{ m}, A = 1$
Chapter 6. Freeway Speed Control

\begin{itemize}
  \item \textbf{Impact of the Time Step} $\Delta t$
  Extensive sensitivity analysis showed that as the time step ($\Delta t$) increases, the recommended vehicle velocities ($u_{in}$) in the control zone will not be updated frequently, and might not be in phase with the dynamic changes on the downstream link. On the other hand, decreasing ($\Delta t$) will result in the vehicles having a short time interval to receive and apply the recommended speed. Also, vehicles may update their speeds multiple times in the control zone, which could create turbulence and raise safety issues.

  \item \textbf{Effects of the Control Zone Length} $D$
  The sensitivity analysis showed that as the length of the control zone ($D$, Figure 6.1) increases, more vehicles will be in that zone and consequently will adopt the recommended speed, which, in turn, might cause artificial congestion (needless reduction in the vehicle speeds) on the upstream link. In contrast, decreasing the length of the control zone showed that not enough vehicles will get the chance to apply the controller recommendation to improve the system’s performance.

  \item \textbf{Activation Condition Impact} ($R_{kmin}$ & $R_{kmax}$)
  The controller activation condition (Equation 6.20) ensures that the SH controller is only activated when the downstream link starts to become congested in order to avoid artificial congestion on the upstream link. The analysis showed that early activation of the controller will result in artificial congestion, and late activation will not have an impact on the system’s performance.
\end{itemize}

6.2.3 Simulation Results and Discussion

The performance of the proposed controller was evaluated at a network level and at the freeway links level.

\begin{itemize}
  \item \textbf{Overall Network Performance}
  The average measure of effectiveness (MOE) values over the entire network (Figure 6.3(a)) for the base case and for different scenarios of the Freeway-SMC-SH controller are shown in Table 6.1, in addition to the improvement percentage relative to the base case.

  The simulation results (SH-3 scenario) demonstrate a significant reduction in average travel time (12.17%), average total delay (20.67%), average stopped delay (39.58%), average fuel consumption (2.6%), and average CO$_2$ emissions (3.3%).
\end{itemize}
6.2. Simulation Setup & Results

Table 6.1: Network Average MOEs and (%) Improvement Using Freeway-SMC-SH Over Base Controller

<table>
<thead>
<tr>
<th>MOE</th>
<th>Base</th>
<th>SH-1</th>
<th>SH-2</th>
<th>SH-3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average Travel Time (s/veh)</td>
<td>1034.27</td>
<td>944.36</td>
<td>932.03</td>
<td>908.37</td>
</tr>
<tr>
<td>Improvement %</td>
<td>8.69</td>
<td>9.88</td>
<td>12.17</td>
<td></td>
</tr>
<tr>
<td>Average Total Delay (s/veh)</td>
<td>557.46</td>
<td>476.22</td>
<td>466.46</td>
<td>442.25</td>
</tr>
<tr>
<td>Improvement %</td>
<td>14.57</td>
<td>16.32</td>
<td>20.67</td>
<td></td>
</tr>
<tr>
<td>Average Stopped Delay (s/veh)</td>
<td>256.77</td>
<td>170.41</td>
<td>162.47</td>
<td>155.13</td>
</tr>
<tr>
<td>Improvement %</td>
<td>33.63</td>
<td>36.72</td>
<td>39.58</td>
<td></td>
</tr>
<tr>
<td>Average Fuel (L/veh)</td>
<td>1.16</td>
<td>1.14</td>
<td>1.13</td>
<td>1.12</td>
</tr>
<tr>
<td>Improvement %</td>
<td>1.51</td>
<td>1.84</td>
<td>2.6</td>
<td></td>
</tr>
<tr>
<td>Average CO2 (grams/veh)</td>
<td>2482.13</td>
<td>2430.88</td>
<td>2421.67</td>
<td>2400.15</td>
</tr>
<tr>
<td>Improvement %</td>
<td>2.06</td>
<td>2.44</td>
<td>3.3</td>
<td></td>
</tr>
</tbody>
</table>

• Freeways’ Performance
The LA network has 457 signalized intersections, 459 stop signs and 30 yield signs, where the SMC-SH controller is not activated. This might conceal the full degree of improvement achieved using the proposed controller. To have a clear picture of the benefits using the Freeway-SMC-SH controller, we evaluated the impact of the introduced logic on the freeways only. In this case, the benchmark scenario will involve freeways only, as shown in Figure 6.3(b).

Table 6.2: Freeway Links MOEs and (%) Improvement Using Freeway-SMC-SH Over Base Controller

<table>
<thead>
<tr>
<th>MOE</th>
<th>Base</th>
<th>SH-3</th>
<th>Imp. (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average Travel Time (s/veh)</td>
<td>41.99</td>
<td>33.39</td>
<td>20.48</td>
</tr>
<tr>
<td>Average Queued Vehicles (veh/link)</td>
<td>17.09</td>
<td>13.39</td>
<td>21.63</td>
</tr>
<tr>
<td>Total Fuel Consumption (L/link)</td>
<td>297.29</td>
<td>289.67</td>
<td>2.56</td>
</tr>
<tr>
<td>Total CO2 Emissions (grams/link)</td>
<td>615800</td>
<td>592710</td>
<td>3.75</td>
</tr>
</tbody>
</table>

Table 6.2 presents the improvement percentage in MOEs using the Freeway-SMC-SH controller (SH-3 scenario) over the base case. A reduction in the average travel time of
20.48% was achieved. Similarly, there was a reduction in the number of average queued vehicles per link, total fuel consumption per link, and total CO$_2$ emissions per link of 21.63%, 2.6%, and 3.75% respectively. These results reveal that the Freeway-SMC-SH controller is so efficient on the freeway links that it weighs on the total averages of the whole network.

6.3 Conclusion

In this work, we developed a new solution exploiting the connectivity of CVs, inspired by the theory of sliding mode control (SMC) coupled with speed harmonization (SH), to effectively alleviate traffic congestion on highways. The developed control logic was implemented and tested in downtown Los Angeles, California, on a roadway network that includes urban roads and congested highways, using the INTEGRATION microscopic traffic assignment and simulation software. The performance of the developed Freeway-SMC-SH controller (only employed on freeway links) was compared to a base case control scenario.

The results show significant improvements in network-wide (freeway and non-freeway links) measures of performance. Specifically, there was a reduction in the average travel time of 12.17%, in the average total delay of 20.67%, in the average stopped delay of 39.58%, and in the average CO$_2$ emissions of 3.3% over the base case scenario where there is no freeway control. Moreover, the results show significant improvements in the freeway operations, with a reduction in the average travel time of 20.48%, in the average queue length of 21.63%, in the fuel consumption of 2.56%, and in CO$_2$ emissions of 3.75%.

The results demonstrate significant potential benefits of using the proposed Freeway-SMC-SH controller activated on highways on large scale networks. This controller makes efficient use of the current network infrastructure, increases the traffic handling capacity of highway roads, and reduces congestion.
Chapter 7

Identifying & Controlling a Congested Area in LA Network

In this chapter, identifying a homogeneously congested and compact area of the large scale LA urban roads is performed using Geographical Self Organizing Maps (GeoSOM) clustering. Then controlling the identified area using SMC-SH controller was conducted.

7.1 GeoSOM Clustering

In this section, GeoSOM is presented and compared to Kmeans and DBSCAN clustering methods.

7.1.1 GeoSOM

7.1.1.1 Overview

Self Organizing Maps (SOM) described in Section 2.3 are widely used in clustering. However, they do not give a special consideration to geographical location. Baccao et al. [98] stressed the importance of considering geographical locations when trying to cluster geo-referenced data. They cited the first Law of Geography saying “everything is related to everything else, but near things are more related than distant things” [99]. This means that distant elements (having large geographical distances) should not be clustered together even if they match in all other attributes. Baccao et al. tested two methods for incorporating geographical features in clustering. First, they included geographical coordinates in the input vectors and they highly weighted them to give them big importance. Another way of filling the gap was by developing a new architecture called GeoSOM (Geographical SOM) [100]. The
GeoSOM is based on the SOM (described in Section 2.3) with the exception that the BMU (best matching unit) search procedure is divided into two phases. During the first phase, a geographical BMU is found. The search is only based on the geographical coordinates of the neurons. The geographically closest neuron to the input vector is the geoBMU. The second phase consists of choosing the final BMU within a radius $R$ of the vicinity of the geoBMU, based on non-geographical components. The radius $R$ is termed the geographical tolerance. GeoSOM restricts the candidates for being a BMU only to the units geographically close to the input vector.

GeoSOM is used for clustering in different applications. However, to the best of our knowledge it has not been used before in clustering transportation networks, where spatially connected and homogeneously congested clusters are needed in order to apply traffic controllers.

### 7.1.1.2 Methodology

In this section, the GeoSOM algorithm is presented. Let $X$ be the set of the $N$ input data vectors. For each input vector $x \in X$, $x = [x_{\text{geo}}, x_{\text{att}}]$, where $x_{\text{geo}}$ are the geographical coordinates of the input and $x_{\text{att}}$ represent the non-geographical attributes of the input. Let $G$ be a grid of $N_{\text{neu}}$ neurons having weights $W$, where each neuron has a weight $w = [w_{\text{geo}}, w_{\text{att}}]$, where $w_{\text{geo}}$ are the geographical weights and $w_{\text{att}}$ are the non-geographical. Initially $w$ are randomly selected.

The algorithm is considered to have three stages:

- Competition between neurons to find the best matching unit BMU (the closest to the input).
- Collaboration: the BMU shares its winning with its neighbouring neurons by exciting them through the neighborhood function $h_{BMU_{\text{final}},j}(t)$ (generally gaussian). The collaboration is high at the beginning and then it decreases over time by making $\sigma(t)$ decrease over time $h_{BMU_{\text{final}},j}(t) = \exp\left(-\frac{||r_{BMU_{\text{final}}}-r_{j}||^2}{2\sigma^2(t)}\right)$, where $r_{BMU_{\text{final}}}$ and $r_{j}$ are the positions of the $BMU_{\text{final}}$ and the neuron $j$, respectively, on the neurons’ grid.
- Weight update: the neurons move toward the input vector based on a learning rate $\alpha(t)$ which decreases with time.
Algorithm 1: GeoSOM Algorithm

initialize $\alpha(1), \sigma(1), w(1), \text{TrainingSteps}, R$ ;

for $t = 1 : \text{TrainingSteps}$ do
    for $i = 1 : N$ do
        $BMU_{geo} = \arg \min_j (\|x_{i,geo} - w_{j,geo}\|)$ ;
        $S_R := \{l : \|w_{l,geo} - w_{BMU_{geo}}\| < R \}$ ;
        $BMU_{final} = \arg \min_{l \in S_R} (\|x_{i,att} - w_{l,att}\|)$ ;
        for $j = 1 : N_{neu}$ do
            $h_{BMU_{final},j}(t) = \exp(-\|r_{BMU_{final}} - r_j\|^2) / 2\sigma(t)^2) ;$
            $w_j(t + 1) = w_j(t) + \alpha(t)h_{BMU_{final},j}(t)(x_i - w_j(t)) ;$
        end
    end
    update $\alpha(t)$ ;
    update $\sigma(t)$ ;
end

7.1.1.3 Case Study

Los Angeles (LA) network (Figure 7.1) is a congested network that suffers from high travel times and delays. In order to apply the network sliding mode controller based on speed harmonization (SMC-SH) to reduce congestion, we need to identify a geographically connected region with homogeneous high densities that we want to protect from congestion. In order to do so, we will apply the GeoSOM algorithm on network links considering their geographical locations and their link densities.

Every link is represented by its midpoint location (x and y coordinates). Every link then had three attributes: x coordinate of the midpoint, y coordinate of the midpoint, and density k. x, y and k are normalized between 0 and 1 to avoid having one attribute weighting more

Figure 7.1: LA links
than the others.

At the end of the simulation, the Unified-distance matrix (U-matrix described in Section 2.3) is calculated. It is based on the difference between neurons’ densities. Neurons having densities far from each others belong to different clusters. Each neuron has a color code representing how far it is (density-wise) from its neighbouring neurons. The results of the U-matrix are interpolated and projected into the input space to define a color code for each link. Then a visual identification of the clusters is performed based on these colors.

### 7.1.1.4 Sensitivity Analysis

The goal of this work is to identify a highly congested homogeneous region in order to be able to control it from congestion. For this reason, in order to test the performance of GeoSOM, we only considered its capability in identifying one cluster; the one having high densities (the yellow zone on the left of LA network in Figure 7.2). A sensitivity analysis of the GeoSOM algorithm is performed for three different parameters: number of neurons, learning rate, and geographical tolerance \( R \).

![Figure 7.2: Real Link Densities in LA Network](image)

**Number of Neurons**

Different number of neurons are used for the GeoSOM clustering. Three cases are presented in Figure 7.3. Figures 7.3(a), 7.3(b), and 7.3(c) represent the neurons at the end of the simulation. Figures 7.3(e), 7.3(f), and 7.3(g) show an interpolation of the color code for network links. On top of the interpolation are the real network links colored in red. Blue color in the interpolation plots means that the links have similar densities and could be clustered together. Yellow color means that the difference in densities between the links is high and then they should not be clustered together.
Figure 7.3(a) shows that the neurons didn’t cover all the map, and consequently 5×5 neurons are not enough. 25×25 neurons are too many (Figure 7.3(c)). The clustering result for 25×25 neurons shows that all the links are clustered in one big cluster (Figure 7.3(g)). 15×15 neurons is the best number of neurons, it covers all the network (Figure 7.3(b)), and it succeeds in obtaining homogeneous clusters (dark blue) with clear boundaries (light color) (Figure 7.3(f)).

**Learning Rate**

Different values of initial learning rates $\alpha$ were tested for GeoSOM clustering. Since $0 < \alpha < 1$, the tested values ranged from 0.05 to 0.8. Results of GeoSOM for different $\alpha$ are presented in Figure 7.4. It can be noticed that for small values of $\alpha$ ($\alpha = 0.05$ and $\alpha = 0.2$), the homogeneous dark blue regions are very small which will not be very suitable for our control objectives. When $\alpha$ is high ($\alpha = 0.8$), all the clusters are combined into one cluster which is not desired for our clustering purposes. The best values of $\alpha$ are 0.4 and 0.6 where we can find a nice homogeneous dark blue area that we can control. $\alpha = 0.4$ is chosen for the rest of the work because on top of the nice homogeneous area that it gives, it has clear distinct borders from all the sides (clear color in Figure 7.4(c)) which makes it more suitable for control.
62 Chapter 7. Identifying & Controlling a Congested Area in LA Network

Geographical Tolerance

Different values of R are tested for GeoSOM. The values range from \( R = 0 \) to \( R = 8 \). Low values of R mean giving more importance to geographical locations. High values of R mean ignoring geographical locations and giving more importance to non-geographical attributes.

Figure 7.5: GeoSOM maps for different Geographical Tolerance R.

The results are shown in Figure 7.5. It can be noticed that for \( R = 0 \), the clustering results didn’t give a nice dark blue area on the left side of the network (the congested area having high densities). Each time R increases, the dark blue area gets bigger. For \( R = 2 \), the area is still too small and could not be used for control. A better cluster appears when \( R = 5 \)
7.1. GeoSOM Clustering

in Figure 7.5(c). However, when the geographical tolerance increases to $R = 8$, the cluster borders start to disappear and all the links tend to be clustered in the same cluster which is not our objective (Figure 7.5(d)).

**Best Parameters**

In conclusion, the best set of parameters for GeoSOM clustering are $15 \times 15$ neurons, $\alpha = 0.4$, and $R = 5$. The homogeneous region with high densities is presented by the black contour in Figure 7.6.

![Figure 7.6: Cluster Identification for Best Parameters of GeoSOM.](image)

#### 7.1.2 DBSCAN

Density-based spatial clustering of applications with noise (DBSCAN) is a density-based clustering algorithm [101]. It brings together points with a large number of adjacent neighbors. It has two parameters: $\epsilon$, radius of a neighborhood with respect to some point and minPts, the minimum number of points required to form a dense region (minimum cluster size). DBSCAN is used for transportation network clustering in [11].

The input to DBSCAN is $(s.x, s.y, k)$, where $x$ and $y$ are the geographical location and $k$ is the link density. $x$, $y$, and $k$ are normalized between 0 and 1. $s$ is a scale factor $> 1$ to put in value the geographical location so that we obtain clusters with geographically connected links.

A sensitivity analysis is performed for DBSCAN. For each minPts chosen, $\epsilon$ can then be selected by using a $k$ – distance graph. Good values of $\epsilon$ are where this plot shows an “elbow” [101].

The results of the sensitivity analysis for different minPts are presented in Figure 7.7. A lot of noise points are observed on the left side of the network for minPts=17. Decreasing minPts to 12 reduces the noise level and some clusters are formed instead of the noise. Decreasing minPts further continues to reduce the noise. For minPts=9, bigger clusters are formed on the left side of the network (having the highest link densities). However, decreasing minPts...
Chapter 7. Identifying & Controlling a Congested Area in LA Network

(a) minPts=6

(b) minPts=9

(c) minPts=12

(d) minPts=17

Figure 7.7: DBSCAN Clustering

to 6 leads to partition the clusters into smaller ones which is not convenient for our control application. Overall, the best minPts is 9 and $\epsilon$ is chosen to be 0.8 by using a $k - distance$ graph (Figure 7.8).

![k-distance graph](image)

Figure 7.8: K-distance graph for DBSCAN.

The mean densities of the clusters result of $minPts = 9$ are represented in Table 7.1. The clusters with high densities (congested) are 1, 4, and 5 with mean densities 0.43, 0.5563, and 0.6732, respectively. In order to apply the traffic controller, we grouped those three congested and geographically connected clusters into one cluster (Figure 7.9).

Table 7.1: Mean densities for different classes of DBSCAN clustering

<table>
<thead>
<tr>
<th>Class ID</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean density</td>
<td>0.43</td>
<td>0.1235</td>
<td>0.0607</td>
<td>0.5563</td>
<td>0.6732</td>
<td>0.2497</td>
<td>0.12</td>
</tr>
</tbody>
</table>
The dark blue cluster in Figure 7.9(b) will be used later on in Section 7.1.4 to compute performance metrics.

![DBSCAN Clustering](image)

(a) Original DBSCAN clustering  
(b) DBSCAN cluster

Figure 7.9: DBSCAN Clustering

### 7.1.3 Kmeans

Kmeans is a clustering method that divides data points into K clusters, with each data point belonging to the cluster with the closest mean. It is widely used as a testbed to compare the results of other clustering methods [10, 11]. In this work, the best number of clusters $K$ is determined using Calinski Harabasz Method [102].

The input to Kmeans is $(s.x, s.y, k)$, where $x$ and $y$ are the geographical location, $k$ is the link density. $x$, $y$, and $k$ are normalized between 0 and 1. $s$ is a scale factor $> 1$ to put in value the geographical location so that we obtain clusters with geographically connected links.

The result of Calinski Harabasz for $K = 2$ to $K = 30$ is shown in Figure 7.10. The maximum value corresponds to the best $K$ which is $K = 12$ in this case.

![Calinski Harabasz for Kmeans](image)

Figure 7.10: Calinski Harabasz for Kmeans.
Chapter 7. Identifying & Controlling a Congested Area in LA Network

The clustering result for $K = 12$ is presented in Figure 7.11.

![Figure 7.11: Kmeans Clustering](image)

(a) Original Kmeans Clustering    (b) Kmeans Cluster

The mean density is calculated for each cluster and presented in Table 7.2. The cluster with the highest mean density is cluster number 12, which corresponds to the blue cluster in Figure 7.11. It will be used in Section 7.1.4 to compute performance metrics.

<table>
<thead>
<tr>
<th>Class ID</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean Density</td>
<td>0.1121</td>
<td>0.0787</td>
<td>0.2521</td>
<td>0.0964</td>
<td>0.1843</td>
<td>0.0915</td>
<td>0.1202</td>
<td>0.1753</td>
<td>0.1202</td>
<td>0.1403</td>
<td>0.1753</td>
<td>0.5598</td>
</tr>
</tbody>
</table>

7.1.4 Comparison of Performance of Clustering Methods

In order to compare the performance of the GeoSOM and other clustering algorithms we define the following performance metrics:

- Quantization Error: $QE = \frac{\|k(class) - \bar{k}(class)\|}{\text{length}(class)}$; where $\bar{k}(class)$ is the mean density of the class, and $\text{length}(class)$ is the number of elements in the class.

- Spacial Quantization Error: $SQE = \frac{\|xy(class) - \bar{xy}(class)\|}{\text{length}(class)}$; where $\bar{xy}(class)$ is the x and y coordinates of the center of the class.

- Variance: density variance in the class.
7.1. GeoSOM Clustering

The three performance metrics are calculated and presented in Table 7.3. Better performance is achieved when QE, SQE, and the variance are low. This holds true for GeoSOM which has the lowest values.

Table 7.3: Performance Metrics for different Clustering Algorithms

<table>
<thead>
<tr>
<th></th>
<th>QE</th>
<th>SQE</th>
<th>Variance</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBSCAN</td>
<td>0.0396</td>
<td>0.0095</td>
<td>0.1241</td>
</tr>
<tr>
<td>Kmeans</td>
<td>0.0333</td>
<td>0.0055</td>
<td>0.1494</td>
</tr>
<tr>
<td>GeoSOM</td>
<td>0.0336</td>
<td>0.0037</td>
<td>0.0837</td>
</tr>
</tbody>
</table>

In order to better see the benefits of GeoSOM over DBSCAN and Kmeans, percentage improvements of GeoSOM with respect to DBSCAN and Kmeans are calculated for each of QE, SQE, and variance. The results are presented in Table 7.4. The most important improvements are for SQE and variance, which is exactly what GeoSOM is meant for: providing spatially connected links within the cluster and having clusters with low variance in density.

Table 7.4: Improvement Percentage of GeoSOM over DBSCAN and Kmeans

<table>
<thead>
<tr>
<th></th>
<th>% Improv. GeoSOM w.r.t DBSCAN</th>
<th>% Improv. GeoSOM w.r.t Kmeans</th>
</tr>
</thead>
<tbody>
<tr>
<td>QE</td>
<td>15.15</td>
<td>-0.9</td>
</tr>
<tr>
<td>SQE</td>
<td>61.05</td>
<td>32.72</td>
</tr>
<tr>
<td>Variance</td>
<td>32.55</td>
<td>43.98</td>
</tr>
</tbody>
</table>

7.1.5 Conclusion

Our research goal in clustering is to find highly congested and geographically compact clusters with low density variance. GeoSOM succeeded in achieving our goal and outperformed DBSCAN and Kmeans with improvement percentages up to 43% in variance, up to 60% in SQE and up to 15% in QE. The result of the GeoSOM clustering could be used by traffic controllers to reduce congestion in the network.
Chapter 7. Identifying & Controlling a Congested Area in LA Network

7.2 Sliding Mode Controller based on Speed Harmonization (SMC-SH) in LA

The sliding mode control based on speed harmonization (SMC-SH) described in Section 5.2 is applied to the large scale LA network. The protected area (PN) is the result of the GeoSOM clustering presented with black borders in Figure 7.6 in Section 7.1.1.

The set point $\bar{k}$ of the NFD; which corresponds to the density where the throughput is maximum; is 40.95 $veh/km$. The controller is activated before reaching the set point, specifically when the protected network density is greater than or equal to $Density_{percentage} \times \bar{k}$. The calculation of vehicles’ speed is done every update interval $T$.

After performing sensitivity analysis using Particle Swarm Optimization [91], the following parameters are obtained: $T = 40s$, $Ratio_{LinkDistance} = 0.34$, $Range_{LinkDistance} = 98.7m$, $Density_{percentage} = 0.4$, $\eta = 0.0013$

Using the Integration micro-simulator, different measures of effectiveness (MOE) are calculated. Table 7.5 shows the improvement percentages of the SMC-SH over the no control case (NPC) for the whole network (Protected and non-protected areas).

Table 7.5: Network wide results MOEs and (%) Improvement Using SMC-SH Over NPC in LA network

<table>
<thead>
<tr>
<th>MOE</th>
<th>System</th>
<th>NPC</th>
<th>SMC-SH</th>
<th>Improvement. (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average Travel Time (s/veh)</td>
<td>NPC</td>
<td>1198.43</td>
<td>1123.53</td>
<td>6.25</td>
</tr>
<tr>
<td>Total Delay (s)</td>
<td>NPC</td>
<td>625.70</td>
<td>566.88</td>
<td>9.4</td>
</tr>
<tr>
<td>Stopped Delay</td>
<td>NPC</td>
<td>282.82</td>
<td>236.23</td>
<td>16.47</td>
</tr>
<tr>
<td>Acceleration/deceleration delay</td>
<td>NPC</td>
<td>342.87</td>
<td>330.65</td>
<td>3.56</td>
</tr>
<tr>
<td>Fuel (L)</td>
<td>NPC</td>
<td>1.13</td>
<td>1.11</td>
<td>1.7</td>
</tr>
<tr>
<td>CO2(g)</td>
<td>NPC</td>
<td>2448.75</td>
<td>2395.07</td>
<td>2.19</td>
</tr>
</tbody>
</table>

Applying SMC-SH on the protected area of LA network improved the travel time, total delay, and stopped delay by 6.25%, 9.4%, and 16.47%, respectively.
7.3 Conclusion

In this chapter, a GeoSOM algorithm for clustering transportation network is presented and implemented. Its performance was compared to state of the art clustering methods DBSCAN and Kmeans. Results show that GeoSOM outperformed both DBSCAN and Kmeans in forming spatially compact clusters with homogeneous densities. The result of the GeoSOM clustering was used by the developed SMC-SH controller presented in Section 5.2 in order to reduce congestion. Results prove that SMC-SH succeeded in improving the performance of the large scale LA network. SMC-SH achieved 6.25% improvement in travel time, 9.4% improvement in total delay, and 16.47% improvement in stopped delay. Acceleration/deceleration delay, fuel consumption and CO2 emissions also got improved.
Chapter 8

Summary, Conclusions, and Future Research

This chapter summarizes the major findings of the dissertation’s study and addresses proposals for future research.

8.1 Summary and Conclusions

Traffic growth and limited roadway capacity decrease traveler mobility and increase travel times, delays and fuel consumption. These problems can be improved by reducing traffic congestion. The goal of this research was to reduce congestion in a large scale network. In order to accomplish that, perimeter control strategy based on the network fundamental diagram (NFD) was considered. An efficient and simple perimeter control used in the literature called Proportional Integral Controller (PIC) was implemented and evaluated on a grid network and it showed good performance. However, PIC has different parameters that need to be tuned in order to give good performance. Since PIC was based on the NFD, we studied the impacts of the weather on the NFD which could impact the performance of PIC [103]. Then, by re-tuning PIC parameters for different weather conditions, a development of a weather tuned perimeter control (WTPC) was performed and compared to the original PIC [93]. Since the truck percentage in the network (jam density) impacts the NFD, re-tuning PIC parameters with respect to jam densities was performed and jam density perimeter control (JTPC) was developed [18]. Both WTPC and JTPC showed better performance than PIC for different weather conditions and different truck percentages, respectively.

This shows that PIC is sensitive to its parameters that need to be tuned effectively. Also, PIC requires having a well-defined NFD that will be linearized in order for the controller to perform well.

Trying to overcome the aforementioned problems in the PIC, a Sliding Mode Controller (SMC)
8.1. Summary and Conclusions

was developed \[104\]. It has many advantages over PIC, namely it makes no assumptions with regards to the shape of the NFD (i.e., linearization of the NFD around the set point, then calibrating the PIC gains using the linearized data). This controller also has consistent performance with varying demand patterns. Unlike the PIC, it requires no tuning, but its parameters need to be within a specified range. The user decides how fast the system should converge to the sliding surface, then once on the sliding surface how fast it converges to the desired control. SMC computes the flow that needs to enter a protected network and converts it to traffic light times in order to be implemented in road networks. Another way of implementing the sliding mode controller is to control vehicles’ speed on the links entering the protected network. Coupling Speed harmonization (SH) with sliding mode control (SMC), an SMC-SH was developed and implemented in INTEGRATION. Connected automated vehicles (CAVs) technology was used to gather data and as a control mechanism. This approach makes the control independent from infrastructure (traffic signals), and consequently reduces the system cost. SMC-SH showed good performance inside and outside the protected area. All this work was applied to the mid-size grid network replicating downtown Washington DC.

The next step was to apply the SMC-SH on the real Los Angeles (LA) large scale network. Noticing that LA has really congested freeways, a development of a freeway sliding mode-speed harmonization (Freeway-SMC-SH) controller was performed and tested in the INTEGRATION micro-simulator \[105\]. The proposed Freeway-SMC-SH controller is unique in two aspects: (1) the system identifies bottlenecks dynamically without having to do so a priori; and (2) regulates connected vehicles (CV) speeds upstream of these identified bottlenecks to disperse traffic congestion. Simulation results showed improvements in performance metrics not only on the freeway links, but also for the overall network (freeway and non-freeway links).

In order to apply SMC-SH on the road networks of LA, an identification of a homogeneously congested and spatially compact area was necessary. A Geographical Self Organizing Map (GeoSOM) clustering algorithm was implemented and tested in Matlab. Its performance was compared to the state-of-the-art DBSCAN and Kmeans clustering methods. GeoSOM outperformed them in identifying a spatially compact cluster with up to 60% improvement, and succeeded in having cluster with low variance up to 43% improvement.

Finally, SMC-SH was tested on the congested area identified by GeoSOM. SMC-SH improved the performance of the large scale LA network by 6.25% in travel time, 9.4% in total delay, and 16.47% stopped delay. It also improved fuel consumption and CO2 emissions compared to the no control case.
8.2 Recommendations For Future Research

In this work, the developed Freeway Sliding Mode Control is coupled with speed harmonization connected vehicles (CV) data. It is assumed in this work that we have 100% of market penetration rate of CV. Future work will entail testing the controller for different lower levels of market penetration rates.

All the vehicles used in this work are running with fuel. A study of the performance of the control methods for electric vehicles is of interest.

The GeoSOM clustering implemented in this work is static (for one time-period). Future work in this research axis entails developing a dynamic GeoSOM clustering that changes over time, in order to accommodate the traffic controller as congestion changes over time too.
Bibliography


