Developing an advanced collision risk model for autonomous vehicles

by

Christos Katrakazas

A doctoral thesis submitted in partial fulfilment of the requirements for the award of Doctor of Philosophy of Loughborough University

July 2017

Abstract

Aiming at improving road safety, car manufacturers and researchers are verging upon autonomous vehicles. In recent years, collision prediction methods of autonomous vehicles have begun incorporating contextual information such as information about the traffic environment and the relative motion of other traffic participants but still fail to anticipate traffic scenarios of high complexity. During the past two decades, the problem of real-time collision prediction has also been investigated by traffic engineers. In the traffic engineering approach, a collision occurrence can potentially be predicted in real-time based on available data on traffic dynamics such as the average speed and flow of vehicles on a road segment. This thesis attempts to integrate vehicle-level collision prediction approaches for autonomous vehicles with network-level collision prediction, as studied by traffic engineers.

An interaction-aware motion model (i.e. a model which describes the motion of each vehicle and the interactions between vehicles) based on Dynamic Bayesian Networks (DBNs) is extended in order to accommodate both network-level collision prediction and vehicle-level information. The corresponding datasets contain a) collision and traffic data from the M1 and M62 motorways on the Strategic Road Network of England during 2012 and 2013 and two expressways in Greece, b) highly disaggregated simulated traffic and conflict data from M62 and c) vehicle-level data acquired using the radar sensor of an instrumented vehicle.

The prevailing traffic conditions just before reported collisions as well as traffic conditions during normal operations act as inputs to the network-level classifiers in order to estimate the probability of a collision happening in real-time. Network-level collision prediction is performed by six machine learning classifiers, i.e. k-Nearest Neighbours (kNN), Support Vector Machines (SVMs), Relevance Vector Machines (RVMs), Random Forests (RFs), Gaussian Processes (GPs) and Neural Networks (NNs). Moreover, as normal traffic conditions are usually overrepresented in traditional real-time collision prediction studies all the network-level collision

prediction classifiers are treated with imbalanced learning techniques to assure proper identification of both hazardous and safe traffic.

The network-level classification results imply that imbalanced learning crucially increases the power of all network-level classifiers. Undersampling cases representing safe traffic conditions is found to work better with traffic data aggregated in 5-minute or 15-minute intervals. On the other hand, oversampling dangerous traffic conditions along with undersampling safe cases performs better in highly disaggregated data (i.e. in 30-second or 1-minute intervals).

By integrating network- and vehicle-level information in the interaction-aware DBN, it has been found that when traffic conditions are classified as hazardous, then the identification of dangerous traffic participants is notably enhanced. Even when traffic data aggregated at 30-second intervals are utilised, the identification of vehicles posing an imminent threat to the ego-vehicle is reinforced by 9-14%. However, when traffic conditions are deemed as normal, the interaction-aware model demonstrated that network-level information does not boost the detection of dangerously driving vehicles.

Acknowledgements

Foremost, I would like to express my sincere gratitude to my supervisors, Professor Mohammed Quddus and Professor Wen-Hua Chen, for their continuous support and guidance during my doctoral studies. It has been an honour to work with both of them. They advised me, motivated and encouraged me immensely, whenever I needed it.

I would also like to thank Dr Lipika Deka, who, during the first two years of my studies, dedicated time and effort to help me develop my ideas and has been always available to discuss aspects of my project.

Special acknowledgments are given to Professor George Yannis and Dr Athanasios (Akis) Theofilatos, from the National Technical University of Athens, for providing a necessary part of traffic and collision data. Furthermore, I would like to thank Professor Christian Laugier and the e-Motion Team in INRIA, Grenoble, France for hosting me as a research intern and helping me develop my methodology.

I am also thankful to the fellow researchers in the Transport Study Group and the Research Hub, who provided feedback on my work and assisted me in various times throughout my project.

I am very grateful to all of my friends for their constant support throughout the years. Special thanks go to my good friends Leonidas, Eirini, Danis, Penny, Tasos, Theofilos, Makis, Nika, Vasiliki and Stelios for making my PhD studies a much more enjoyable experience.

Last but not least, I must deeply thank my parents, Savvas and Angeliki, as well as my brother, Panagiotis for their invaluable support, patience and love. They believed in me and encouraged me from the first to the last day of this research, even when things seemed difficult. I would not have completed this PhD without them and therefore I dedicate this thesis to them.

Contents

List of Figures

List of Tables

Acronyms

ADAS: Advanced Driver Assistance Systems

ATM: Active Traffic Management

AUC: Area Under the (ROC) Curve

AV: Autonomous vehicle

AVI: Automatic Vehicle Identification

CRN: Collision risk in the network-level

CRV: Collision risk in the vehicle-level

DBN: Dynamic Bayesian Network

ENN: Edited Nearest Neighbours

GP: Gaussian Processes

ITS: Intelligent Transportation Systems

K: Vehicle kinematics

kNN: k-Nearest Neighbours

NLCP: Network-level collision prediction, i.e. real-time collision prediction as studied by traffic engineers

NN: Neural Network

PET: Post-Encroachment Time

RENN: Repeated Edited Nearest Neighbours

RF: Random Forest

ROC curve: Receiver Operating Characteristic curve

RVM: Relevance Vector Machine

SMOTE: Synthetic Minority Oversampling Technique

SMOTE-ENN: The integration of SMOTE with ENN

SRN: Strategic Road Network

SSAM: Surrogate Safety Assessment model

SVM: Support Vector Machine

TCT: Traffic Conflict Technique

TN; TP; FP; FN: True Negative; True Positive; False Positive; False Negative

TTC: Time to collision

VMS: Variable Message Signs

Z: Sensor measurements

1. Introduction

1.1.Background

Motor vehicles are an essential part of everyday life and the most popular means of transport around the world. Their invention in the late 19th century completely changed the structure of societies and transport systems worldwide. Nowadays it is estimated that there are more than 1 billion automobiles in the world (Sousanis, 2011).

Although vastly used in everyday life, vehicles are also a major reason for fatalities. According to the World Health Organisation (WHO), traffic collisions are the main non-health related cause of death - claiming 1.25 million lives in 2013 worldwide - and the main cause of death for people aged 15-29 (WHO, 2015). An additional consequence of vehicular collisions is also the cost incurred from fatalities and injuries, which burdens the states. The 2016 annual report of the International Transport Forum (ITF) showed that the cost of road collisions was £16.3 billion in 2014, accounting for a 0.9% of the GDP of the UK.

As a phenomenon, traffic collisions are characterised by complexity as well as randomness and have been heavily researched over the years. According to Treat et al. (1979), traffic collisions are primarily the outcome of road environment, vehicle and human factors. Oh et al. (2001) extended Treat et al.'s spectre of collision contributing factors by taking into account a fourth element, traffic dynamics. They argued that collisions might happen even if the environment, the vehicle and the driver point towards safe driving. Therefore, traffic engineering research focused on the identification of the traffic conditions that cause traffic collisions.

The increased availability of traffic data from loop detectors enhanced the possibility of predicting these collision-prone traffic conditions in real-time. Early studies utilised data exclusively from inductive loop detectors (e.g. Lee et al., 2003) however recent technological advances have led to the incorporation of additional data derived from video image processors (Ikeda et al., 1999, Astarita et al., 2011), microwave radars(Wang et al., 2015, Shi and Abdel-Aty, 2015), Automatic Vehicle

Identification (AVI) devices (Ahmed et al., 2012a, Yu and Abdel-Aty, 2013a), probe vehicles (Park and Haghani, 2016) and smartphones (Guido et al., 2012), all of which ensure a data-rich environment for Intelligent Transportation Systems (ITS) experts.

Real-time collision prediction is formulated on the basis that the probability of a collision occurring could be estimated for a short-time prediction horizon from traffic data retrieved online (e.g. Abdel-aty and Pande, 2005). Such models utilise traffic data either from loop detectors or from other technologies, such as AVI devices or traffic cameras. These data are usually aggregated in 5-minute intervals and are matched to the documented time of actual collision so as to represent collision-prone or "dangerous" traffic conditions. For the majority of the existing literature, traffic data collected 5-10 minutes prior to the collision event are utilised for collision-prediction.

The essence of real-time collision prediction is that if the probability of a collision is predicted for an imminent time period, then collisions can be avoided or mitigated, relieving the traffic environment from collision related congestion and delays (Quddus et al., 2010). As a result, proactive traffic management systems and Advanced Travel Information Systems (ATIS) have emerged as parts of ITS so as to estimate the probability of collisions, implement traffic calming measures at the location of a collision and inform other traffic participants about the collision event downstream of their position. Information regarding the implementation of traffic calming measures or the broadcasting of a collision event downstream is usually posted through Variable Message Signs (VMS) by a traffic management agency (Hossain, 2011). Intensive research has taken place in the past two decades to make real-time collision prediction more accurate regardless of the traffic data used for the analysis. As real-time collision prediction investigates the probability of a collision occurring at a specific link or segment of the road network, real-time collision prediction will be henceforth termed as network-level collision prediction (NLCP) in this thesis.

Nevertheless, even though proactive traffic management achieved a general decrease in collision occurrences and fatalities in the last decades (NHTSA, 2015, European

Comission, 2016) a major contributory factor to collisions is human error. In a survey by Singh (2015), up to 94% of traffic collisions were found to have been caused by driving mistakes. Similarly, Staubach (2009) stated that 90% of collisions were caused by human error and added that driving errors can be classified in three categories:

- Errors due to lack of information (e.g. obstructed vision)
- Errors due to failure to use information (e.g. inattention, omission of using turn signals or checking a blind-spot, misperception of the vehicle's speed or acceleration)
- Errors due to misuse of information (e.g. habituation of not observing hazards while driving on quiet roads, miscalculation of another vehicle's relative distance and speed)

In addition, collisions may be the result of drivers' impaired state which may be due to intoxication or fatigue. Distraction by the passengers of the car may also lead to limited attention to available information regarding the environment and the surrounding conditions on the road (Wang et al., 2013).

Aiming at improving road safety, the automotive industry and research are focussed upon creating "intelligent vehicles". Since the 1980's many research efforts have been committed to the application of technologies and systems from the fields of mobile robotics and computer science to passenger cars and road transport in general (Macek et al., 2006). To date, these technologies have been mainly used as part of Advanced Driving Assistance (ADAS), by means of on-vehicle sensors which detect the surrounding environment (e.g. lane recognition, motion prediction, emergency breaking) and post a warning to the human driver if a hazardous situation is taking place.

Autonomous vehicles (also known as robotic, driverless or self-driving) are a promising technology which is supposed to enhance road safety and simultaneously reduce congestion, fuel consumption and emissions that come with current "regular" cars, by removing the human element from the task of driving (Litman, 2014). Although human drivers can reason and perceive sufficiently the surrounding environment, AVs would be able to provide a greater perception horizon even when

illumination or visibility conditions are bad. Moreover, AVs will be able to communicate with other vehicles which can eventually lead to less space and time headways, and faster reactions than humans in cases of evasive manoeuvres. A list of possible advantages and limitations from the use of autonomous vehicles is presented in Table 1.1 below.

Table 1. 1 Benefits and problems from the use of autonomous (with information from Forrest and Konca, (2007); Thrun, (2010); Gurney, (2013); Mui, (2013); Lin, (2014); Litman, (2014) and Ross, (2014))

Benefits	Problems
•**Improved traffic safety** (Increased Reliability , Faster Reaction Time, Reduction of number and severity of crashes, confrontation of the imperfection of human driving) •**More efficient traffic flow** (Reduced traffic congestion , higher speed limits, easier merging and exiting in traffic, easier parking, fewer vehicles on roads, carpooling) •**Fuel efficiency** (Better fuel economy, no stop & go driving, fuel optimization, emmisions reduction, more efficient shipping of goods) •**Time savings** (Higher Speed Limits, Less Time for Parking and mitigation of parking scarcity- AV can leave passengers at some point and return to take them back-) •**Removal of driving constraints** (Age, Disabilities, Sleep while Driving, Intoxication or other impairments) •**Economic relief** (Reduction of accident-related costs, fuel optimization, car maintenance costs)	•**Damage/ Crash Liability** •**Increased cost** (Manufacturing and Infrastructure costs) •**Cyber Security** (Hacking, Loss of privacy) •**Loss of employment for driving related professions** (taxi drivers, public transport drivers, chauffeurs, traffic police etc) •**Inexperienced Drivers** to take control in emergency situations •**Unwise planning focus** (cost effective transport projects, no care taken for pedestrians and other road users) •**Absence of policy for autonomous vehicles**

Early approaches in the field of autonomous driving included the California PATH program, Navlab and "Hands-free across America" in the United States, as well as Dickmann's Mercedes-Benz robot van and the EUREKA Prometheus Project in Europe (Eskandarian, 2012).

The most influential projects though, the three Grand Challenges, were initiated by the Defence Research Advance Projects Agency (DARPA) in 2002. The initial two took place in an off-road course in 2004 and 2005 while the third one (in 2007) was

set in an urban environment with California traffic rules so that the interactions between agents could be exploited (Eskandarian, 2012). Despite the fact that six autonomous vehicles finished the race, fully autonomous driving capabilities were not met, as collisions between the vehicles could not be avoided (e.g. Fletcher et al., 2008, Martinez-gomez and Fraichard, 2009).

The literature regarding autonomous vehicles has been increasing at a rapid pace in the last few years, focusing on building a reliable yet efficient safety framework which could guarantee a safe navigation on a modern roadway. Autonomous cars can come up against the human error which is the predominant factor of traffic collisions, but the issue of interacting with the surroundings (e.g. human, other vehicles, environment) while safely transporting passengers is critical for the application of autonomous cars in the years to come.

1.2.Problem definition

In order to ensure the safety of its occupants and other road users, an autonomous vehicle (AV) has to perform a safe navigation when interacting with other traffic participants. This fundamental task -known as path planning within the AV literature- provides a vehicle with a safe and collision-free path towards its destination while taking into account the vehicle dynamics, its manoeuvre capabilities in the presence of obstacles, traffic rules and road boundaries. Collision prediction and situational risk assessment usually takes place in the manoeuvre or behavioural planning of current planning approaches in automated driving (Katrakazas et al., 2015, Paden et al., 2016).

Currently, a motion model (i.e. a model which describes the motion of every vehicle at a moment of time) is used to predict the intended trajectories of other vehicles and surrounding objects in a specific traffic environment and compare them with the trajectory of the interested AV in order to estimate the collision risk. Computational complexity however emerges when searching for an efficient trajectory representation in which vehicles are assumed to move independently (Agamennoni et al., 2012, Lefèvre et al., 2014). Recent approaches (e.g. Lefèvre, 2012, Agamennoni et al., 2012, Gindele et al., 2015) have emerged, trying to address the problem of risk

assessment of AVs by taking into account contextual information (i.e. information on the traffic scene and the motion of other vehicles) as well as human-like reasoning about vehicles' interaction without predicting the trajectories of all other vehicles. Nonetheless, perfect sensing or communications between vehicles are often assumed (Katrakazas et al., 2015, Paden et al., 2016).

The inherent limitations of robotics-based approaches on risk assessment in the context of organically changing dynamic and cluttered road environments indicate that alternative methods should be sought as supplements for building a robust and comprehensive risk assessment module.

NLCP, as part of proactive traffic management systems in ITS, is a potential candidate to assist AVs in their task of safe navigation. It can be understood from section 1.1 that the traffic engineering perspective of collision prediction addresses the macroscopic problem of identifying a location with high probability of a collision occurrence. This spatiotemporal risk could potentially provide a broader picture of the road network in terms of hazardous traffic conditions as an additional safety layer to AVs. It is likely that it could increase confidence that another traffic participant is dangerous if NLCP points towards collision-prone traffic and enhance safety assessment when some parts of the on-board sensor system are malfunctioning or obscured.

However, current NLCP modelling needs further enhancing in order to become available as a resource to AVs. Traditional NLCP models usually follow four steps: i) select actual traffic variables (e.g. temporal or spatial means and variance of them) as predictors, ii) collect data corresponding to historical collision cases and normal traffic conditions, iii) formulate a classification problem and utilise a collision prediction model to estimate the probability of a collision and iv) evaluate the modelling performance. Nevertheless, efficiently applying these four steps is not perfectly tractable. Traffic measurements from a particular location tend to be correlated with each other and therefore the inclusion of available but correlated traffic variables might result in misleading classification results (Hossain, 2011). Additionally, traffic data might not be available at all times and hence classifiers

need to be able to work with limited or bad quality data (Xu et al., 2015a). Machine learning classifiers have been applied to solve the problem of correlated variables and missing data, however, in most cases they act like "black-boxes"[1] which restrict the interpretability of the models. Moreover, as collisions are rare events, the data collection of collision-prone and normal traffic cases leads to an overrepresentation of cases the cases representing normal traffic which, consequently, results in biased classifiers and a large number of false alarms(Xu et al., 2016a). Oddly, there is little evidence in the literature to date to take this imbalance into account when building NLCP classifiers. Finally, the fact that AVs require good quality information to be available at high frequencies (e.g. at a second or sub-second level) from their sensors suggests that the aggregation of traffic data into 5-minute intervals is relatively high and hence more disaggregated data need to be utilised.

This thesis will attempt to develop a suitable modelling framework that can accommodate both NLCP and vehicle-level collision prediction and avoidance, as studied by traffic engineering experts and AV experts in computer science and robotics respectively. The results intend to bring the two domains (i.e. robotics and traffic engineering) together, enhance the performance of classifiers for a large spectrum of data aggregation intervals and ensure a safer navigation of AVs among other traffic participants.

1.3.Research importance

Research on autonomous vehicles is a trending topic nowadays. Automotive and technology companies such as Google (Google, 2015), BMW (Ziegler et al., 2014b), Tesla (Kessler, 2015)), as well as universities (e.g. Thorpe and Durrant-Whyte, 2009) have been in constant competition for the past years in order to make autonomous cars a reality. Nevertheless, in order for AVs to be widely accepted and replace conventional cars, the public needs to be persuaded that they are to be trusted. To date however, although people find autonomous driving a promising and fascinating technology (Kyriakidis et al., 2015), the expected level of safety is not fulfilled

[1] A "black-box" approach is a method which can be described by its input and output without any transparency on how input led to output.

(Fagnant and Kockelman, 2015), as a number of collisions has already occurred (Google, 2015, Ackerman, 2016).

On the other hand, the subject of autonomous vehicles for traffic engineers is limited to either research on traditional traffic problems such as user preference (e.g. Haboucha et al., 2017), traffic flow effects (e.g. (Le Vine et al., 2015), policy (e.g. Fagnant and Kockelman, 2015), traffic assignment (Di and Sacco, 2016) ethics (Shariff and Rahwan, 2016) or the attempt to solve control engineering problems (Ntousakis et al., 2016, Brown et al., 2017).

ITS experts have achieved milestones in the last two decades in order to be able to predict collisions in real time using aggregated traffic data. If, however NLCP information were to be utilised by AVs risk assessment modules, the prediction horizon needs to be significantly shortened and highly disaggregated data should be employed, without high false alarm rates. This PhD research attempts to enhance the performance and interpretability of NLCP in order for them to become a useful resource for AVs. By developing an integrated framework with which network-level collision prediction (NLCP) can be incorporated into AVs' safety modules, this project is intended to assist the perception and safety performance of automated driving. Consequently, AVs will come one step closer to fulfilling Asimov's zeroth law: "A robot may not harm humanity or, by inaction, allow humanity to come to harm" (Asimov, 1950).

1.4.Aim and Objectives

The aim of this PhD research is to develop an advanced collision risk model for autonomous vehicles, which integrates network-level and vehicle-level collision risk.

This aim will be fulfilled through the following objectives:

- To investigate existing motion planning and collision risk assessment algorithms for autonomous vehicles
- To explore factors and methods related to NLCP
- To refine traffic and collision data so as to enhance the quality of the analysis

- To formulate a framework for the incorporation of NLCP within the risk assessment module of an autonomous vehicle
- To enhance the performance and interpretability of current machine learning classifiers used in NLCP models
- To evaluate the framework for risk assessment of AVs

1.5.Thesis outline

This thesis is organised into eight chapters. This section provides an outline of each chapter.

Chapter 2 conducts an in-depth and critical literature review of the state-of-the-art in motion planning for autonomous vehicles so as to provide an understanding of how autonomous vehicles achieve collision-free motion and assess the risk of colliding in real time.

Chapter 3 reviews the literature on NLCP as studied by traffic engineering. The review critically compares the methodological approaches as well as the data used, reveals limitations and identifies tools and methods which can enhance the prediction of collision-prone traffic conditions.

Chapter 4 presents the methodology of this thesis. The chapter begins with the description of a probabilistic model which integrates NLCP with vehicle-level risk assessment. This is followed by the description of probabilistic machine learning algorithms and imbalanced learning techniques, which will be applied to enhance classification performance. Finally, the methods of obtaining highly disaggregated traffic and conflicting data through microsimulation are presented.

Chapter 5 illustrates the collision and traffic data which will be employed to build NLCP models, the data obtained from microsimulation and data obtained from an instrumented vehicle. All the datasets are presented along with descriptive statistics and scatterplots which demonstrate the size and complexity of the data.

Chapter 6 reveals the classification results for all the datasets and methods used. All the developed classifiers are compared and contrasted with each other, as well as with results from the literature.

Chapter 7 discusses the impact of the proposed framework on the identification of "dangerous" traffic participants by AVs.

Finally, Chapter 8 summarises the findings from this research, lays out the contribution to knowledge, as well as the drawbacks of this work. This is followed by a discussion for future research directions.

Note on the definition of risk

Risk is generally defined as the likelihood and severity of a collision that may occur for a vehicle of interest in the future (Lefèvre et al., 2014). In this thesis, however, emphasis is given only on the probability of a collision as NLCP information usually correlates traffic dynamics with the probability of a collision and not its severity. Hence, throughout the thesis the term risk indicates the probability of a collision occurring in the near future for a vehicle of interest.

2. Literature Review of Motion Planning Approaches

2.1.Introduction

AVs are a promising evolution of current vehicle technology and ADAS, and are envisaged to be the sustainable future for enhanced road safety, efficient traffic flow and decreased fuel consumption, while improving mobility and hence general well-being (e.g. Thrun, 2010, Burns, 2013, Le Vine et al., 2015). Research on autonomous vehicles has been growing rapidly in recent years encompassing different domains, including robotics, computer science, and engineering. Moreover, it should be noted that scientific advances have been made by car manufacturers who do not always publicly disclose the details on their approaches or algorithms, owing to commercial sensitivity.

Critical decision making is the key to autonomy and is realised through planning algorithms, incorporated within the middleware of an autonomous vehicle's navigation module. The main purpose of planning is to provide the vehicle with a safe and collision-free path towards its destination, while taking into account the vehicle dynamics, its manoeuvre capabilities in the presence of obstacles, along with traffic rules and road boundaries (Zhang et al., 2013). Planning is a memory consuming and computationally intensive routine, which is run in parallel with other routine operations of the vehicle (e.g. control, data fusion, obstacle tracking). The inputs and outputs of planning are in dependence with these other modules. Reliable, robust, and adaptable planning is essential, especially in an urban mixed traffic scenario. These algorithms receive inputs from the sensor framework and supplement these inputs with data from digital road maps in order to provide a full workspace in which the planning takes place.

Existing planning algorithms originate primarily from the field of mobile robotics, and have subsequently been applied to different on-road and off-road vehicles and operational environments (e.g. desert vehicles (Thrun et al., 2006), planetary rovers (Pivtoraiko and Kelly, 2009) and buses (Fernandez et al., 2013). Furthermore, a large number of algorithms have been developed for non-holonomic and car-like robots planning in abstract, simulation-based environments (e.g. Scheuer and Fraichard,

1997). In the review presented in this chapter, only approaches concerned with planning for on-road autonomous vehicles are analysed. In general, planning for autonomous or intelligent driving is divided into four hierarchical classes, as suggested by Varaiya (Varaiya, 1993): (1) route planning, (2) path planning, (3) manoeuvre choice and (4) trajectory planning (termed as control planning in the work of Varaiya). Route planning is concerned with finding the best global route from a given origin to a destination, supplemented occasionally with real-time traffic information. Route planning is not within the scope of this review and readers are referred to (Thorpe and Durrant-Whyte, 2009) for details on a route planner. Path, manoeuvre and trajectory planning components of autonomous on-road driving (often combined as one) take vehicular dynamics, obstacles, road geometry and traffic interactions into account, and are the primary focus of this review. It is important to emphasise that this review presents a state-of-the-art review of motion planning techniques, based on the works after the DARPA Urban Challenge (DUC) in 2007 (Thorpe and Durrant-Whyte, 2009) and is intended to serve as a key reference for researchers who are conducting research on the domain of autonomous vehicles. The focus on works after the DUC is given because the challenge was a milestone in autonomous driving and resembles the state-of-the-art work until 2007, thus enabling research in autonomous driving to profoundly advance.

The remainder of the review is structured as follows: foundational definitions form the body of Section 2.2; while Section 2.3 presents an extensive literature review of motion planning approaches applied to autonomous vehicles, with a focus on manoeuvre planning and risk assessment. Key limitations of the approaches are then described in Section 2.4. Finally, in Section 2.5 the review is summarised.

It should be noted here, that this review of the state-of-the-art in AV motion planning aims to identify inherent limitations in risk assessment for AVs as well as the most suitable part of motion planning routines to accommodate NLCP information. The findings of this review will form the basis for the development of the integrated methodology which integrates NLCP and AV risk assessment in section 4.3 of this thesis and its application in chapter 7.

2.2. Definitions

This section describes the key conceptual terms commonly used in the literature within the field of planning for robots and, hence, autonomous vehicles. As mentioned previously, this review focuses on planning at a local on-road level and not globally (e.g. routeing).

The set of independent attributes which uniquely define the position and orientation of the vehicle according to a fixed coordinate system is termed the *configuration* vector (Eskandarian, 2012). Consequently, the set of all the configurations of the vehicle constitute the *configuration space*.

The set of attribute values describing the condition of an autonomous vehicle at an instance in time and at a particular place during its motion is termed the '*state*' of the vehicle at that moment (Eskandarian, 2012). The most common set of attributes, defined as a vector, which are used to express the state of a vehicle are the position (x, y, z), the orientation (θ_x, θ_y, θ_z), linear velocities (v_x, v_y, v_z) and angular velocities (ω_x, ω_y, ω_z). Subsequently, *state space* represents the set of all possible states that a vehicle can be in. As will be seen in the next sections, the mathematical representation of a *state space* differs from the approach taken by vehicle planning. A trade-off between explicit representation and efficiency of the algorithms should be considered for every planning problem. Representations that can be used for constructing a configuration or a state space will be discussed in Section 2.3.

The *bicycle model* is a dynamic/kinematic model of vehicles, in which the two front and wheels along with the two rear ones, are replaced by one front and one rear wheel respectively. The vehicle moves on the plane and its coordinates are described by the vector (x, y, θ) where x, y is the position of the centre of gravity and θ is the orientation of the vehicle. Steering angle of the front wheels is denoted by φ. A basic assumption of the bicycle model is that the inner slip, outer slip and steer angles are equal.

A robot is *holonomic* if the controllable degrees of freedom are equal to the total degrees of freedom. Vehicles or car-like robots are thus *non-holonomic* because they

are described by 4 degrees of freedom (2 Cartesian coordinates, orientation and heading) but have 2 kinematic constraints: i) they can only move backwards and forwards, tangentially to the direction of their main body and ii) the steering radius is bounded. Another definition of *holonomy* is described in LaValle (2006), Siegwart et al. (2011) and Laumond (1998), where it is stated that car-like vehicles are non-holonomic because their motion is constrained by non-integrable differential constraints due to the assumption that the wheels roll without slipping.
Actions are system inputs (such as acceleration, steering angle) that result in a vehicle's state transition. Actions are defined either as a function of time or as a function of state and time. *Action space* represents the set of all possible actions that can be applied to the *state space*.

Given a *configuration space* or a *state space*, planning is a computationally intensive task, demanding high memory utilisation. Within the field of robotic motion (both in the case of on-road and off-road vehicles and objects), planning is performed at different levels. The highest level of planning is concerned with origin to destination route planning and the workspace is essentially limited to digital maps representing the underlying road network. The lowest level of planning is concerned with planning a smooth trajectory adhering to vehicular dynamics and such a plan is chalked out on a small (local) search space of high dimensional states. To facilitate the description and discussion, the following terms are defined as used in the rest of the review.

Path is expressed as a continuous sequence of configurations beginning and ending with the boundary configurations, i.e. the initial configuration and the terminating configuration respectively (Eskandarian, 2012). In other words, a path is a geometric trace that the vehicle should follow in order to reach its destination without colliding with obstacles. *Path-planning* is therefore the problem of finding a geometric path from an initial configuration to a given terminating configuration, such that each configuration and state (if time is taken into account) on the path is a feasible one. A feasible configuration/state is a configuration/state that does not result in a collision and adheres to a set of motion constraints such as road and lane boundaries, as well as traffic rules. It should be noted that, throughout the review where path planning is

discussed, importance is given in finding the best and safest geometric trace, under the constraints described above which also have a logical argument regarding the rules of traffic.

Manoeuvre is a high-level characterisation of the motion of the vehicle, regarding the position and speed of the vehicle on the road. Examples of manoeuvres include 'going straight', 'turning', 'overtaking' etc. A manoeuvre is nominal if it is performed safely according to traffic or other rules. As a result, *manoeuvre planning* addresses the problem of taking the best high-level decision for the car, while taking into account the path that is specified from path planning.

On the other hand, *trajectory* is represented as a sequence of states visited by the vehicle, parameterised by time and, possibly, velocity**.** *Trajectory planning* (also known as trajectory generation) is concerned with the real-time planning of the actual vehicle's transition from one feasible state to the next, satisfying the vehicle's kinematic limits based on vehicle dynamics and constrained by the navigation comfort[2], lane boundaries and traffic rules, while avoiding, at the same time, obstacles including other road users as well as ground roughness and ditches. Trajectory planning is parameterised by time as well as acceleration or velocity, and is frequently referred to as motion planning. During each planning cycle, the path planner module generates a number of trajectories from the vehicle's current location, with a look-ahead distance, depending on the speed and line-of-sight of the vehicle's on-board sensors, and evaluates each trajectory with respect to some cost function to determine the optimal trajectory. Trajectory planning is scheduled at regular intervals; the length of which largely depends on the frequency of receiving fresh sensor data. For example, trajectory planning was scheduled every 100 milliseconds (ms) in the controller that was tested during the VisLab Intercontinental Autonomous Challenge (Broggi et al., 2012). Error between the current vehicle location and the determined trajectory is monitored; triggering a trajectory revised plan upon detecting an error beyond a pre-defined threshold. It should be noted that there is a rich body of literature on trajectory planning of aircraft movements in the context of

[2] In terms of the acceleration (lateral & longitudinal) that the car develops, the curvature of the trajectory and other parameters which are indicated by standards such as ISO 2631-1 1997.

air traffic control. Aircraft trajectory planning is, however, quite different from trajectory planning of on-road vehicles. The operational environment and the allocated space for aircrafts to manoeuvre is different from the overpopulated, multimodal, congested road network, which is also constrained by road geometry, road lanes and the existence of a large number of obstacles which do not appear in the air. Furthermore, the degrees of freedom, the dynamics and the size of aircrafts are different from on-road vehicles in a way that trajectory planning for cars and aircraft (which must take all these parameters into account) requires different approaches. For example, as described in Schuster (2015) an aircraft's trajectory is 4D (comprising of the spatial coordinates (x, y z) and time) while a road vehicle primarily acts on a 2D space or 3D space if time is added. Furthermore, the state vector of an aircraft motion planning includes 3D position coordinates (x, y, and height), air speed and aircraft mass. However, the state vector of an on-road vehicle does not consider air speed and vehicle mass. It can therefore be understood that an aircraft trajectory is treated with 6 degrees of freedom, while the trajectory of an on-road vehicle is normally treated with only 3.

Most existing trajectory planning implementations follow two steps: (i) trajectory generated on a low resolution/lower dimensional search space in the first step and (ii) the resulting optimal trajectory smoothed out on a higher resolution/higher dimensional search space during the second step. The planning module is integral to rendering complete autonomy to the vehicle, with the outputs of the trajectory planner feeding into the low-level steering/manoeuvre control unit.

2.3. Search Space for Planning

Planning a journey for an autonomous vehicle on the road requires that the environment should be represented in a way that enables the query for a path. This means that the physical space must be transformed into a configuration or a state space. The state space, as defined in the preceding section, consists of every representation of the vehicle position, orientation, linear or angular velocities, in addition to any other measures of interest (Howard, 2009). As the vehicle travels on the road, readings from the sensors and information obtained from a digital map are used to transform the continuum of the environment into a digital representation of

the road network, which is the essential space for planning. This discretisation must efficiently be dealt with in terms of efficiency, density and expressiveness (Howard, 2009), as high density network may result in high computational costs and power. Similarly, inadequate representation, though it would improve computational speed, may introduce sub-optimality and inexpressiveness, not to mention collision risks.

Some of the existing algorithms initiate a search in continuous coordinates using only the road boundaries and positions of the obstacles, for example, driving corridors (Jeon et al., 2013, Hardy and Campbell, 2013, Wille and Form, 2008, Wille et al., 2010a). Decomposition (or tessellation) techniques analyse the space with higher resolution and include Voronoi Diagrams (Dolgov et al., 2010, Lee and Vasseur, 2014), occupancy grids (Kolski et al., 2006; Bohren et al., 2008; Hundelshausen et al., 2008; Kammel et al., 2008; Leonard et al., 2008; Zhao et al., 2011; Xu et al., 2014), cost maps (Bacha et al., 2008; Rauskolb et al., 2008; Schröder et al., 2008; Himmelsbach et al., 2009; Murphy and Newman, 2011; Broggi et al., 2012;) and lattices (Pivtoraiko and Kelly, 2005; Pivtoraiko et al., 2009; Ziegler and Stiller, 2009; McNaughton et al., 2011). Diagrammatic representations of these search spaces are depicted in Figure 2.1.

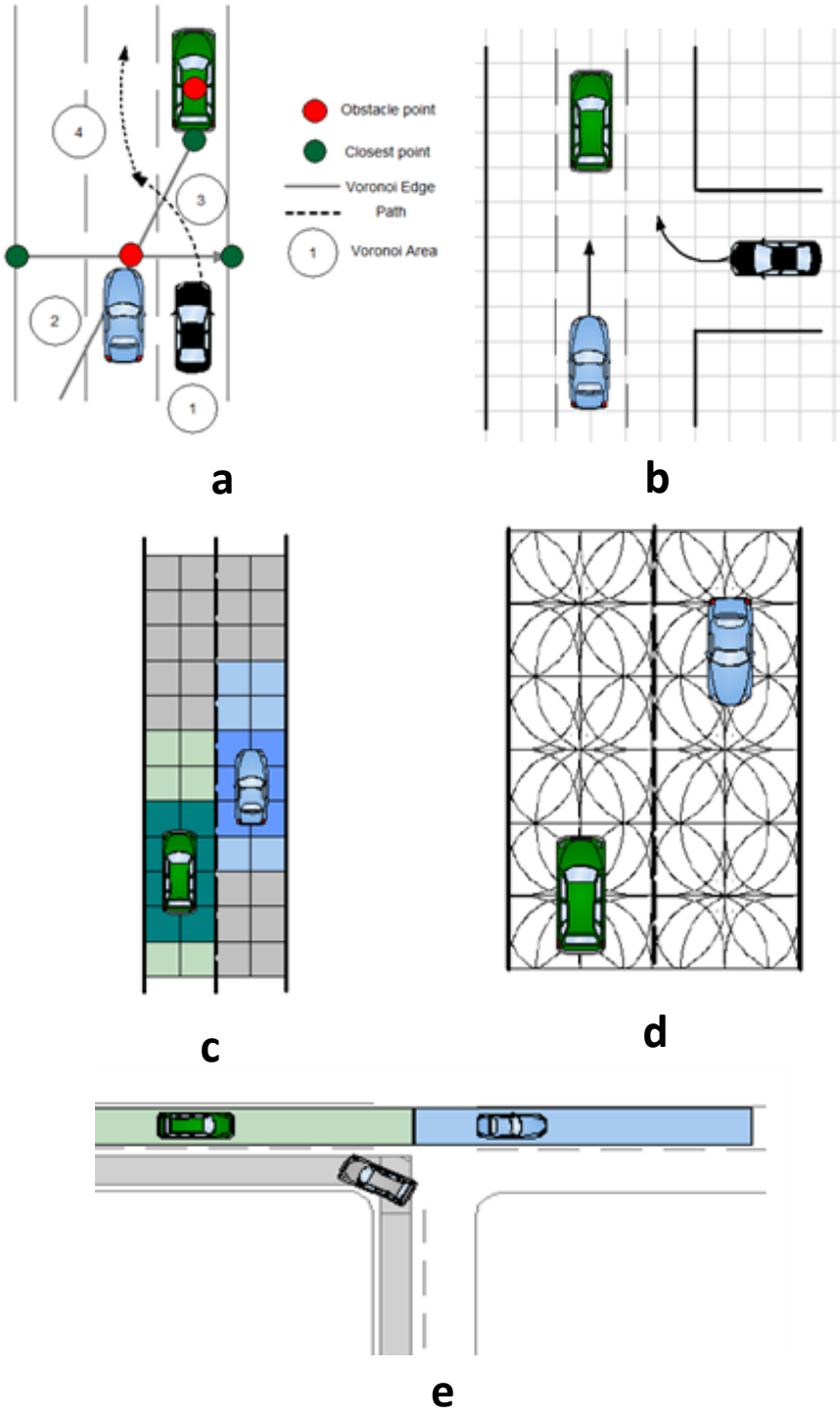


Figure 2. 1 Graphs used in planning

(a) Voronoi Diagram (Lee and Vasseur, 2014);b) Occupancy Grid (Schröder et al., 2008); c) Costmap (Ferguson and Likhachev, 2008);d) State Lattice (Ziegler and Stiller, 2009); e) Driving Corridor (Wille et al., 2010b)

Voronoi Diagrams or Dirichlet tessellation techniques, generate paths which maximise the distance between the vehicle and surrounding obstacles (Takahashi and

Schilling, 1989). Algorithms which are used for searching on Voronoi Diagrams are complete in the sense that, if a path exists in the free space, it would also appear on the Voronoi Diagram (Siegwart et al., 2011). As depicted in Figure 2.1(a), grey lines represent Voronoi edges (i.e. edges with maximum distance from detected obstacles), and produce a space where the vehicle can perform its trip. Dolgov et al. (2010) used Voronoi Diagrams for path-planning of autonomous vehicles in parking lots by combining Voronoi Diagrams with potential fields; an obstacle avoidance algorithm derived from mobile robotics. This combined approach, referred to as Voronoi fields, was developed to overcome the issue of conventional potential field approaches in narrow passages (that generate high potential), which rendered such passages virtually non-traversable. Voronoi Diagrams are typically used for planning in static environments, such as parking lots. Furthermore, Voronoi diagrams on their own are not suitable for on-road path-planning, since Voronoi edges, along which a car navigates, can potentially be discontinuous and unsuitable for non-holonomic cars.

Occupancy grids (Kolski et al., 2006; Bohren et al., 2008; Hundelshausen et al., 2008; Kammel et al., 2008; Leonard et al., 2008; Zhao et al., 2011; Li et al., 2013; Xu et al., 2014;) and *costmaps* (Bacha et al., 2008; Rauskolb et al., 2008; Schröder et al., 2008; Himmelsbach et al., 2009; Murphy and Newman, 2011; Broggi et al., 2012) work in a similar way; they both discretise the *state space* into a grid and each cell of the grid is associated with a probability of the cell being occupied by an obstacle, or a cost proportional to the feasibility or risk of traversal. Risk or feasibility is primarily calculated by considering the presence of obstacles, lane and road boundaries. Grid-based approaches are fast in finding a solution with low computational power (Pivtoraiko et al., 2009) but have difficulties in accounting for nonlinear dynamics in a robust way (Kushleyev and Likhachev, 2009), and in the presence of obstacles (Pivtoraiko et al., 2009). As seen in Figures 1b and 1c, occupancy grids consist of a grid with the position of the obstacles and (sometimes) an attached velocity showing their expected motion; while in cost maps, the higher the cost of a certain cell, the more intense its presentation is on the map.

State Lattices can be seen as a generalisation of grids (Pivtoraiko and Kelly, 2005). In the same way that grids are built by the repetition of rectangles or squares to

discretise a continuous space, lattices are constructed by regularly repeating primitive paths which connect possible states for the vehicle, in terms of position, curvature or time, as can be seen in Figure 1d. The problem of planning then reduces to a boundary value problem of connecting the original state with the required final state (McNaughton et al., 2011). State Lattices overcome the limitations of grid based techniques in efficiency without increasing computational power (Pivtoraiko et al., 2009).

Driving Corridors represent a continuous collision-free space, bounded by road and lane boundaries as well as other obstacles, where the car is expected to move. Driving corridors are based on lane boundary information given on the detailed digital maps, or a map built by using a Simultaneous Localisation and Mapping (SLAM) technique. Lane boundaries form the outer bound of the driving corridors, restricted in the presence of obstacles. In Figure 2.1e, a driving corridor is constructed for each car according to the chosen manoeuvre. The centre line of the determined corridor forms the path around which the trajectory to be followed by an autonomous vehicle is planned. The major drawback of planning in a continuous way is that, since intensive computational power[3] is needed for planning for the entire range of coordinates regarding the road network, representation of roads or lanes may constrain the motion of the vehicle (Fletcher et al., 2008).

It should be noted that the above techniques of search space representation for planning are not always employed independently. For example, Voronoi Diagrams and potential fields have been combined to produce Voronoi fields by Dolgov et al. (2010) to generate a safe trajectory. In most of the cases, they are combined in order not only to provide better results for a single planning level but also to offer planning capabilities in all three levels (i.e. path, behaviour and trajectory planning). Their advantages and disadvantages are summarised in Table 2.1.

[3]The continuous nature of driving corridors, leads to an exponential increase in the dimensions of state vector for each one of the coordinates included in the driving corridor. Thus, at each time moment a large number of attributes need to be calculated for each of the coordinates, necessitating more computational resources.

Table 2. 1 Comparison of Search Space for planning

Representation	Advantages	Disadvantages
Voronoi Diagrams	• Completeness • Maximum distance from obstacles	• Limited to static environments • Discontinuous edges
Occupancy Grids Cost Maps	• Fast discretisation • Small computational power[4]	• Problems with vehicle dynamics • Errors in the presence of obstacles
State Lattices	• Efficiency without increasing computational time[5] • Pre-computation of edges is possible	• Problems with curvature • Restrict motion • Difficulties in dealing with evasive manoeuvres
Driving Corridors	• Continuous collision free space for the car to move	• Computational cost[6] • Constraints on motion

Once a search space is constructed, then the planning algorithms are initiated in order to select the best path, behaviour and trajectory respectively.

2.4. Planning Techniques

This section presents a review of planning techniques used in existing studies in the areas of autonomous on-road driving. Given a route provided by the route planner, motion planning for on-road driving (*hereinafter* planning) concentrates on finding the best path for the vehicle to follow while taking into account the constraints of the

[4] Computational power refers to computations needed to construct the cells and estimate their costs. The space, in which the planning problem is solved, is discretised. Furthermore, the number of attributes needed to define each of the cells is small (the attributes just need to show if the cell is occupied or not, plus the cost of traversing the cell). As a result, the dimensions of the state matrix of each of the cells are manageable in real-time.

[5] Similar to (3), computational time refers to computations needed to construct the lattice: Because of the predefined shape of the curve with which the lattice is constructed and the pre-computation of edges, the space for planning is discretised and thus less time is needed to find the correct solution.

[6] Computational cost for driving corridors is analysed in footnote 2.

vehicle's motion model, waypoints that the vehicle should follow and the traffic environment, including static and dynamic obstacles. Planning can be divided into incremental approaches which try to find the best sequence of state transitions (which are not fully specified from the beginning) by re-using information from previous searches and local approaches which attempt to find the best single state transition for the vehicle to follow. A global or local path also has a strong correlation with the decisions or manoeuvres that the car performs, so manoeuvre planning will also be addressed. As shown in Figure 2.2, path search is initiated after a route has been chosen from the route planner and acts as input to the search for the best manoeuvre (i.e. the manoeuvre which places the car with the most correct and safe behaviour). The final path may however change, based on the best manoeuvre, as shown with a feedback loop between these two modules. Once the path is finalised, the final trajectory planning is generated.

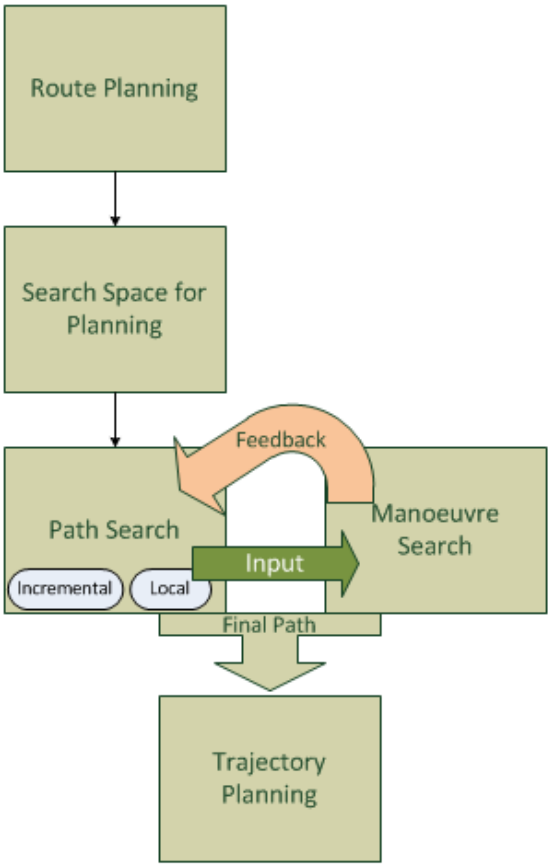


Figure 2. 2 A flow chart of planning modules

As such, planning is divided into three levels of planning, namely:

1) Finding the best geometric path for the vehicle to follow

    a. Finding the best sequence of actions through incremental sampling or discrete geometric structures (i.e. Incremental search);
    b. Finding the best action from multiple final states (i.e. Local search);

2) Finding the best manoeuvre to perform;

3) Finding the best trajectory to follow through the optimisation of a geometric curve, according to given constraints.

For example, when a vehicle is on the road it follows the sequence of waypoints taken from the route planner and then constructs the geometric path of the vehicle (Figure 2.3a). These waypoints must be obstacle-free since the car needs to interact with the other vehicles so as to cooperatively move along the road. According to the geometric path that has been derived and the interactions with other vehicles, the automated vehicle must decide its next 'high level' action (Figure 2.3b); i.e. should it overtake the leading vehicle to reach the next waypoint in time? As implied, these high-level decisions depend on the path, because the vehicle needs reference waypoints in order to decide its best action. If the waypoints and the proper manoeuvre are finalised, then trajectory planning describes the procedure of searching the best way to connect the determined waypoints (Figure 2.3c).

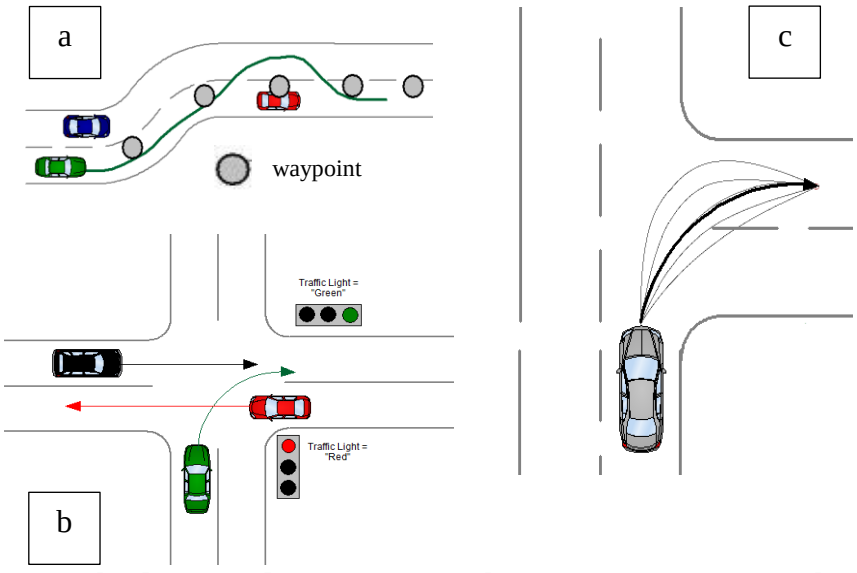


Figure 2. 3(a) Path Planning (b) Manoeuvre Planning (c) Trajectory Planning (adapted from Lee and Vasseur, 2014)

This chapter will review only approaches on manoeuvre planning and risk estimation as this part of planning has the most potential to incorporate network-level information. For more information on motion planning techniques the reader is prompted to Katrakazas et al., (2015).

2.4.1. *Manoeuvre Planning and Decision Making*

During the DARPA Urban Challenge in 2007, analysis showed that there was a lack of interactions between cars and driving in a human-like manner, with many incidents of a behavioural nature being faced during the challenge (Fletcher et al., 2008).

While driving autonomously on public roads, the car at each moment should be capable of deciding the best and safest manoeuvre to undertake after finding the best geometric sequence of waypoints to follow. This decision must be made without overlooking the ego-vehicle's [7]interactions with the surrounding traffic environment. Manoeuvre planning therefore, incorporates techniques which anticipate the behaviour of both the motorised and non-motorised traffic participants and assesses the surrounding traffic situation, thus arming the driverless car to decide on its best manoeuvre. Techniques which are described in this section work on a more high-level basis. Manoeuvre planning moves away from searching for a path or generating a trajectory; instead acting as a 'brain' which filters the results of path search, interacts with other traffic participants and gives the approval for the geometric path before it is transformed to a feasible trajectory.

Techniques for manoeuvre planning can be divided into two categories:

1) Those that emphasise motion modelling and obstacle prediction (Section 2.4.1.1); and

2) Those that are concerned with the decision-making module of autonomous vehicles, based on the modelling of the traffic environment (Section 2.4.1.2).

[7] ego-vehicle: the AV which is the focus of the research

2.4.1.1.Motion Modelling

Lefèvre et al. (2014) present a detailed survey to classify recent research on traffic environment modelling and prediction and introduce several risk estimators for intelligent vehicles. According to their work, motion models are classified into physics-based, manoeuvre-based and interaction-aware models. The first category describes motion models according only to the laws of physics, while the second relies on estimating the intentions of other traffic participants, based either on clustered trajectories or on manoeuvre estimation and execution. These two categories of motion models do not take into account the environment, but rather, view vehicles as independent entities. Interaction-aware models were developed in order for the inter-vehicle relationships to be exploited, so that dangerous situations can easily be modelled and identified in real-time.

As far as risk estimators are concerned, Lefèvre et al. indicate that a collision can be predicted through collision prediction (binary or probabilistic) through estimated trajectories, but also through unexpected behaviour or conflicting manoeuvres between vehicles. The readers are referred to the survey of Lefèvre et al. for further details on the description of techniques and risk indicators.

Most of the approaches for obstacle prediction (also included in the survey of Lefèvre et al.) refer to straight roads and do not apply well to the context of each manoeuvre.

A grid-based Bayesian filter is used by Alin et al. (2012) to model behaviours as spline functions to anticipate curvy roads and infer the trajectories of other vehicles. The technique shows better results than Bayesian filters that do not take into account context, but considers only cut-in and lane change manoeuvres.

A hybrid-state system using hierarchical hidden Markov models and Finite State Machines is used by Gadepally (2013) to predict future state of traffic participants at intersections. This model is motivated by the fact that vehicle behaviours (such as turning in different directions) can easily be estimated by human drivers but are not

efficiently anticipated by automated vehicles. Nevertheless, the approach needs extensive training and extensive data acquisition to train the models.

Ontology, a formal description of entities, hierarchies and interrelationships used in computer and information science, is used by Armand et al. (2014) to reason about the behaviour of traffic participants. Only a limited number of situations (going straight, following and reaching a vehicle or pedestrian) are evaluated using few rules and time efficiency issues are also noticed.

Recent approaches were formulated to better describe the traffic environment by including network-related information. Gindele et al. (2015), for instance, included information on car-following models and the interactions among the vehicle in the adjacent lanes so as to faster recognise the intention of each vehicle and assessed risk using the TTC metric. Their DBN approach requires many variables which consequently need to be trained to efficiently describe, for example, the relationship between traffic participants, the influence of traffic rules to traffic participants and the influence of the geometry of the road on the actions. In order to address some of these issues, Kuhnt et al. (2015) proposed to use a static street model in order to provide an extra hint to a motion model. Their approach, however, fails to provide an efficient description of the inter-vehicle dependencies. Recently, Bahram et al. (2016) showed that even without vehicular communications, if the knowledge of the road geometry and traffic rules is available, the prediction time for anticipating the manoeuvres of other vehicles can be significantly improved. Nevertheless, network-level knowledge was limited to train classifiers that have the capability of detecting any manoeuvre associated with the acceleration and deceleration of vehicles as well as lateral offsets in relation to the centre-line of a lane.

2.4.1.2. Decision-theoretic approaches

Furda and Vlacic (2011) use Multiple Criteria Decision Making (MCDM) and Deterministic Finite Automata (DFA) for driving manoeuvre execution. The inputs come from a priori known data, sensor measurements and vehicular communications. Traffic rules and a hierarchy of objectives during driving are considered for decision making (namely, motion within road boundaries, safety distances, collision avoidance and minimisation of waiting time). The approach needs accurate

information and manually specified weights for each objective in the decision-making routine.

In Hardy and Campbell (2013), driving corridors are constructed according to the predicted motion of dynamic obstacles and the presence of static ones. Vehicles are modelled as rectangles; their trajectories are clustered for easier identification and conflicting trajectories are used to estimate the risk at each moment. In this work, planning is seen as a non-linear constrained optimisation problem. The function which is to be optimised includes terms for static and dynamic obstacles, possible collisions and distance to goal. Single and multiple obstacles are considered but building the driving corridor increases the computational effort linearly (as described in section 2.2.2.), according to the number of obstacles and interactions between cars that are ignored.

A similar approach is used by Ziegler et al. (2014b) where hierarchical concurrent state machines are used with respect to static and dynamic obstacles, as well as yield and merge rules. Driving corridors are also indicated in order for the car to have optimal free space for each part of the journey, while avoiding collisions. The main drawback of this technique, however, is that other cars are presumed not to accelerate and to keep safe distances from the road boundaries.

Kala and Warwick (2013) consider a relatively unstructured road environment. They assume that no road lanes exist and that the majority of the traffic participants are non-autonomous and that there exists no communication between vehicles. At each moment, the vehicle is supposed to display certain behaviour according to the motion of vehicles nearby. Obstacle avoidance, centring (driving in the centre of the road/lane), lane changes, overtaking and being overtaken, slowing down, detecting conflicting behaviours and travelling straight are the pre-designed behaviours. Distance and velocity constraints are used to classify different behaviours online. This work studies only straight roads with infinite length and shows that there is a delay in the decision making of the car in cases such as centring on curvy roads or overtaking. The fast and correct identification of conflicting behaviours between road users is another drawback of this approach.

A Prediction and-Cost-function Based (PCB) approach is adopted by Wei et al. (2014). Using a reference trajectory, as well as static and dynamic obstacles as inputs, multiple candidate trajectories are generated and, after predicting the evolution of the traffic environment, the best strategy is chosen according to comfort[8], safety, fuel consumption and the progress towards the goal. The motion of the vehicle in the vicinity of other cars is considered, and controller reactions and time delays are also simulated for better performance. The approach was validated with simulation and on-road testing and leads to smoother results, as compared to the spatio-temporal lattice planner and with a reduction of 90% on computational cost. However, only single lane behaviours are considered.

White and White (1989) employ Markov Decision Processes (MDPs) in estimating the best manoeuvre for the vehicle to undertake. MDPs incorporate a presumed set of actions which are performed under uncertainty and try to maximise the total rewards or weights for every action. MDPs work on the state space to try to determine a rule which describes the decision to act from one state to another.
Unlike MDPs, which assume that the states are fully observable, partially observable Markov Decision Processes (POMDPs) assume that the state of a robot or a vehicle is not known (Ong et al., 2010). Thus, POMDPs transform the state space into a belief space, which contains all the possible probability distributions for every possible state of the system that is being modelled. If, however, some features of the state of a vehicle are known (for example, the orientation is known but the position is not), we are referring to mixed observability MDPs or MOMDPs (Ong et al., 2010).

In the work of Bandyopadhyay et al. (2012), intention prediction about human traffic participants is embedded into planning. A discrete Mixed-Observability Markov Decision Process (MOMDP) models the interaction between the autonomous vehicle and pedestrians, while making a prediction about the pedestrians' intentions. The behaviours of the ego-car towards the pedestrians that are considered include: 'Reasonable but Distracted', 'Oblivious', 'Impatient' and 'Opportunistic Driving'. Experiments are carried out with simulations and a real-world golf-cart; wherein it is

[8] In the study of Wei et al. (2014), comfort is evaluated according to the acceleration of the car.

assumed that information about pedestrians' positions and velocity is perfectly known. Furthermore, instead of simultaneously treating the set of pedestrians, a MOMDP is separately calculated for each pedestrian and it is assumed that intentions do not change over time. Lastly, experiment results are presented for only half of the behaviours considered.

In contradiction with the previous work, Brechtel et al. (2014) implement a continuous partially- observable MDP, assuming that the belief state is infinitely large because driving is a continuous-space problem. The inputs are the position and velocities of the traffic participants, which are presumed known. Merging scenarios are simulated where the ego-car has occluded vision due to a hypothetically illegally parked car. Having a continuous belief space may lead to a large number of samples needed to make the autonomous vehicle decide. This large number of samples may consequently lead to large computational effort and may also increase the number of close calls for decision making.

Game Theory has also been used by researchers to take into account the interactions between vehicles. For example, Aoude et al. (2010a) examine an intersection environment and try to formulate a perfect information game between traffic participants. Each game terminates if a collision happens, and each vehicle tries to maximise the time to collision, while all other vehicles take on the role of 'enemies' which try to minimise this time. This threat assessment model is then embedded into an RRT-like global planner which generates the path to follow. Real-time capabilities of the approach are provided by evaluation which takes place with two model cars with maximum speeds of 0.5m/sec.

The same concept of Game Theory is followed in the work of Martin (2013) where, again, a perfect information game is used to predict the motion of other vehicles for planning on highways. For the payoff function to be maximised by the ego-car, position, speed and accelerations are taken as input, producing as output the best possible manoeuvre, using a manoeuvre set which includes driving straight, as well as left or right lane changes. The road is assumed to be infinitely straight and simulations are carried out with up to 4 vehicles in the traffic scene.

Planning approaches which emphasise obstacle prediction and decision making are summarised in Table 2.2.

Table 2. 2 Planning approaches with emphasis on obstacle prediction and decision making

Study	Method	Criteria	Environment description	Drawbacks
Furda and Vlacic (2011)	Multiple Criteria Decision Making	Traffic rules Road boundaries Safety distance Collisions Waiting time	Non-intersection segments	Need precise information and manually specified weight for each criterion
Hardy and Campbell (2013)	Driving Corridors and Non-Linear Constrained Optimisation	Behaviour towards static and dynamic obstacles Vehicle dynamics Distance to obstacles Distance to goal	Intersections	Computational effort rises with number of obstacles Ignorance of social interactions between traffic participants
Ziegler et al. (2014b)	Driving Corridors and Hierarchical State Machines	Static and dynamic obstacles behaviour Yield and Merge rules	Intersections and non–intersection segments	Other cars are presumed not to accelerate and to keep safe distances from road boundaries
Kala and Warwick (2013)	Behaviour Choice according to Obstacle Motion	Distance and velocity constraints	No road lanes	Infinite straight roads Problems on curvy roads, overtaking and conflicting behaviours
Wei et al. (2014)	Prediction and Cost-function	Comfort, safety, fuel consumption, distance to goal	Straight roads	Only single-lane behaviours tested

Bandyopadhyay et al. (2012)	Mixed-Observability MDP	Pedestrian position and velocity	Pedestrian crossings	Different modelling required for each pedestrian Intentions are assumed unchangeable
Brechtel et al. (2014)	Partially Observable MDP	Vehicle position and velocity	Merging scenarios with occluded vision	Continuous belief space may lead to large number of samples and large computational effort.
Aoude et al. (2010a)	Game Theory	Time to collision	Intersections	Model-car evaluation at low speeds Perfect information assumed
Martin (2013)	Game Theory	Position, speed, acceleration and manoeuvre choice	Straight roads	Perfect information required

To summarise, manoeuvre planning relies heavily on the relative positions of other traffic participants at the moment of making a decision and estimating the risk of a certain situation. Risk estimation can be performed using risk indicators, such as the Time-to-Collision (TTC), as suggested by Ward et al. (2014; 2015), probabilistic gap acceptance models, as proposed by Lefevre (2012), or by forming situation assessment and choosing the best manoeuvre as a decision theoretic problem (using Markov decision processes or Game Theoretic principles). In the first category of planning (obstacle prediction and risk assessment), more accurate results are provided but context is often omitted from planning. Heavy computational burden may also arise while predicting the motion of the obstacles in the vicinity of the autonomous vehicle. Decision-theoretic approaches cope well with context and may provide solutions to problems like negotiating intersections (such as in urban or suburban environments) or complying with manoeuvres on a highway.

2.5.Constraints and Limitations

The approaches discussed in the previous section have potential to work well in choosing the best manoeuvre and constructing a feasible trajectory. However, limitations still exist and autonomous driving is yet to achieve the levels of human driving competence. This section describes the most significant of the identified limitations.

2.5.1. *Obstacle Handling*

In terms of handling obstacles, existing approaches primarily rely on predicting the trajectories of other traffic participants, either by taking their trajectories into account, or by making assumptions of constant velocities or constant accelerations (Kushleyev and Likhachev, 2009). This leads to a huge computational power requirement, since the obstacles' trajectories need to be calculated and checked at each moment. Such trajectory predictions are performed while disregarding the context within the traffic environment; thereby, leading to interactions between cars or other traffic participants being ignored. Some of the approaches (e.g. Aoude et al., 2010a; 2010b; Bandyopadhyay et al., 2012; Martin, 2013) also assume that there is no uncertainty in the obstacles' motions; this assumption is not valid in real-world situations, especially in a mixed traffic scenario with the presence of human drivers. The lack of understanding between autonomous vehicles and human drivers can be demonstrated by recent experiments conducted by Google (2015). In these experiments, the Google autonomous vehicle could not ascertain the intention of human drivers in its vicinity and this confusion resulted in minor collisions.

Another important limitation of the existing approaches is the simple representation of obstacles as rectangles or circles. In the latter case, the problem is that close proximity motions cannot be performed, due to lack of accuracy in the approximation (as shown in Ziegler et al., 2014a). Interaction-aware models, as presented in Lefèvre et al. (2014) can take interaction between traffic participants into account, but pre-suppose perfect knowledge or communication between the cars. Furthermore, motorcycles and non-motorised traffic participants are usually ignored in most approaches. Another major limitation in terms of obstacle handling is the inability to see around corners and detect obstacles such as pedestrians and bicycles approaching from blind corners. Such a disadvantage leads the planning algorithm to take a 'cautious' and hence inefficient approach, such as slowing down even in the absence of any obstacle.

2.5.2. *Sensing and Perception*

Sensing and perception within existing approaches treat the car as an individual and isolated entity; limiting the perception horizon of autonomous vehicles to the

perception horizon of its individual sensors. Furthermore, most approaches either assume perfect knowledge of the environment (e.g. Aoude et al., 2010b; Bandyopadhyay et al., 2012; Brechtel et al., 2014) or depend on expensive sensing (e.g. Ziegler et al., 2014b) to perceive near-perfect knowledge of the environment and the obstacles. Approaches fail to take into account the limited field of view that driverless cars have and possible blind-spots that may occur, for example, in curved road segments or blind and closed intersections (i.e. intersections with restricted views).

2.6. Summary

Planning for an autonomous vehicle can be divided into three main levels: search for the best path, search for the best manoeuvre and search for the best trajectory. Searching for the best path can be further divided into searching for the best series of paths towards the goal and searching within a limited 'local' time and space horizon. As far as manoeuvre planning is concerned, obstacle prediction and risk assessment are employed, while decision-theoretic approaches (such as Markov decision processes and game theory) have recently emerged to account for interactions within the traffic environment. Lastly, in trajectory planning, the chosen geometric path is bounded with kinematic and motion model constraints and further optimised to assure a smooth and feasible journey along it. This optimisation is based either on the choice of geometric curve to represent the path or on model predictive control. It should be noted that these approaches are rarely treated independently in current research; instead they are typically combined in order to provide a complete plan for the vehicle.

This chapter critically examined existing planning approaches applied to autonomous on-road driving after the milestone of the DARPA Urban Challenge with an emphasis on manoeuvre planning because this subsection of motion planning was deemed the most suitable to accommodate NLCP information. Through manoeuvre planning an AV needs to account for the behaviour of other traffic participants and evaluate the traffic scene, hence collision risk assessment is crucial in order for safety to be assured.

The most important limitations in collision risk assessment by AVs were found to concern the handling of obstacles and their perception capabilities. The majority of existing risk assessment approaches treat traffic participants as independent entities, predict their trajectories and then detect potential collisions. However, such an approach incurs significant computational cost generation through checking of all possible trajectories. Instead of exhaustively calculating and predicting the trajectories of other traffic participants at each epoch (i.e. sensing cycle), a useful proposition would be to perform the trajectory calculation and collision checking only if unusual or dangerous manoeuvres are detected as suggested by Lefèvre et al.,(2012). Interaction-aware motion models take context into account but in many cases perfect communications or sensing is assumed. Hence, alternative methodologies should be formulated which can realistically represent the traffic environment and enhance the perception horizon of AVs.

3. Literature review of network-level collision prediction approaches

3.1. Introduction

In order for traffic engineering principles to be incorporated in the risk assessment module of autonomous vehicles, real-time traffic safety modelling needs to be reviewed so as to comprehend the emerging gaps in the current literature. Therefore, in this chapter the literature regarding NLCP as studied in the area of ITS is thoroughly reviewed and synthesised. This chapter will also review approaches on the use of traffic microsimulation as a possible solution to some of the drawbacks of existing methods on real-time NLCP.

3.2. Typical collision prediction and real-time collision prediction

Conventionally, approaches regarding network-level safety aimed at predicting collision rates by road segment, based on the traffic characteristics, its geometrical characteristics (e.g. inclination, curvature) and environmental conditions (e.g. weather or visibility conditions). To accomplish this, researchers initially employed the Average Annual Daily Traffic (AADT) as it was widely recorded and publicly available. Other traffic related variables contained in safety analyses included vehicle composition, speed limits and congestion indices. As Abdel-Aty and Pande (2007) state, these approaches addressed the issue of identifying locations where most collisions are likely to occur. Such approaches are termed as "collective" or "macroscopic" (Abdel-Aty and Pande, 2007). However, approaches that utilise AADT along with historical collision databases induce a highly aggregated manner of investigating collision occurrence, thus deeming themselves inappropriate for real-time risk assessment and traffic management.

The advances in data collection and management technologies (e.g. installation of loop detectors, automatic vehicle identification (AVI) devices, probe vehicles and traffic cameras) led researchers to the exploration of more microscopic traffic data for traffic collision prediction. Simultaneously, the purpose of safety studies moved from detecting "collision-hotspots" (i.e. locations which tend to have more collisions)

to detecting “collision-prone” traffic conditions (i.e. traffic conditions which may lead to a collision). In more detail, research was not exclusively focused on predicting how many collisions were going to happen during a particular period of time based on aggregated traffic characteristics like AADT or the ratio of Heavy Goods Vehicles (HGVs), but was also into discovering if the traffic conditions at a specific time moment resembled the traffic conditions before a collision or not(Abdel-Aty et al., 2010). The latter type of studies is widely termed as real-time collision prediction or proactive safety studies in the ITS community. In this thesis NLCP is used to denote real-time collision prediction models, as mentioned in the previous chapters.

NLCP models are usually part of road safety systems i.e. systems that monitor the risk of collision using traffic data in real-time and apply all the necessary interventions to smooth traffic after a collision occurrence (Hossain, 2011). Hossain (2011) defines NLCP as follows: “A NLCP model predicts the chance of a crash occurrence within a short time window in the near future for a specific road section mainly using instantaneous traffic flow data (e.g. speed, flow, occupancy) and their descriptive statistics”. This definition of a NLCP model will also be used in this PhD thesis.

This literature review focuses on studies aiming at NLCP as part of traffic management systems. These studies form the state-of-the-art in network-level collision detection and will therefore be reviewed in order to fully understand the underpinning methodology, as well as identify potential knowledge gaps for further research and their incorporation into the respective modules of an autonomous vehicle. More specifically, the studies included in the review are examined with regards to their methodological approaches, the data that they used and their applications.

3.3. Review of real-time collision prediction studies

NLCP models statistically connect real-time traffic measurements with the probability of a traffic collision. Figure 3.1 depicts a typical topology of traffic data collection devices used to build NLCP models. As Figure 3.1 shows, after the location of the collision has been determined, the closest loop detectors (either

upstream only or on both flow directions) are identified and are marked as collision detectors. The data from the conflict detectors during a certain time period before a collision are used as an example for collision-prone conditions, while data from the same detectors at other time intervals illustrate normal driving conditions.

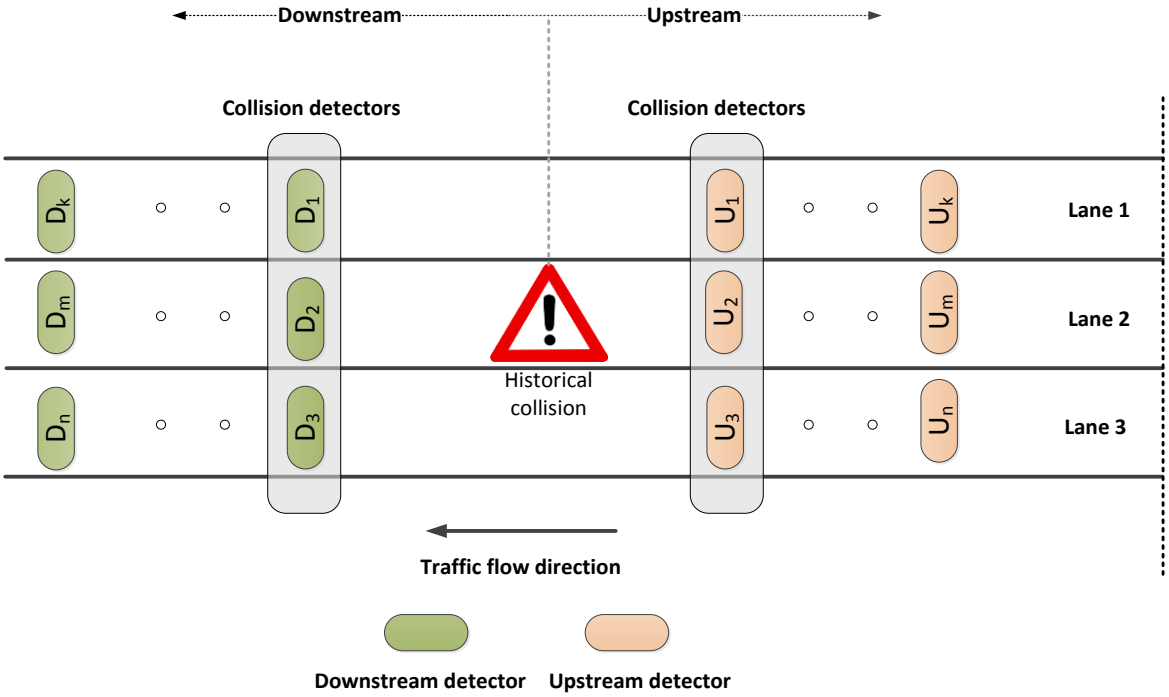


Figure 3. 1 Topology of loop detectors used in NLCP studies

3.3.1. *Early approaches*

Early studies of NLCP models concentrate on analysing traffic data from the upstream segment of a collision location only (e.g. Oh *et al.*, 2001; Lee et al 2002; Golob & Recker, 2004)). Using relatively simple statistical techniques such as non-parametric Bayesian filters (Oh *et al.*, 2001), log-linear modelling (Lee et al., 2002) and non-linear canonical correlation analysis(Golob and Recker, 2004) those studies succeeded in linking the probability of a collision with real-time traffic data obtained from loop detectors.

The study by Oh et al. (2001) is considered to be one of the primary studies on NLCP models. Using traffic data aggregated at 5-minute intervals they proved that the most important predictor of disruptive traffic conditions on motorways was the standard deviation of speed. They also demonstrated that a system which combines real-time traffic data and historical collision data could potentially reduce the

likelihood of a collision occurrence. On the same principle, Lee et al. (2002) and Golob and Recker (2004) demonstrated that within-lane and between-lane variations of speed, volume and traffic density prove to be important variables in real-time crash prediction models. The highlight of the research by Lee et al. (2002), however, was their insistence that studies which aim at detecting collision potential in real-time should be of a proactive nature and looking at identifying disruptive traffic conditions before a collision rather than being reactive, focusing on investigating traffic oscillations after a collision event. The difference between proactive and reactive traffic safety and management systems is depicted in Figure 3.2. On the other hand, Golob and Recker only used collision-related traffic data to develop their model, failing to represent "normal" traffic operations.

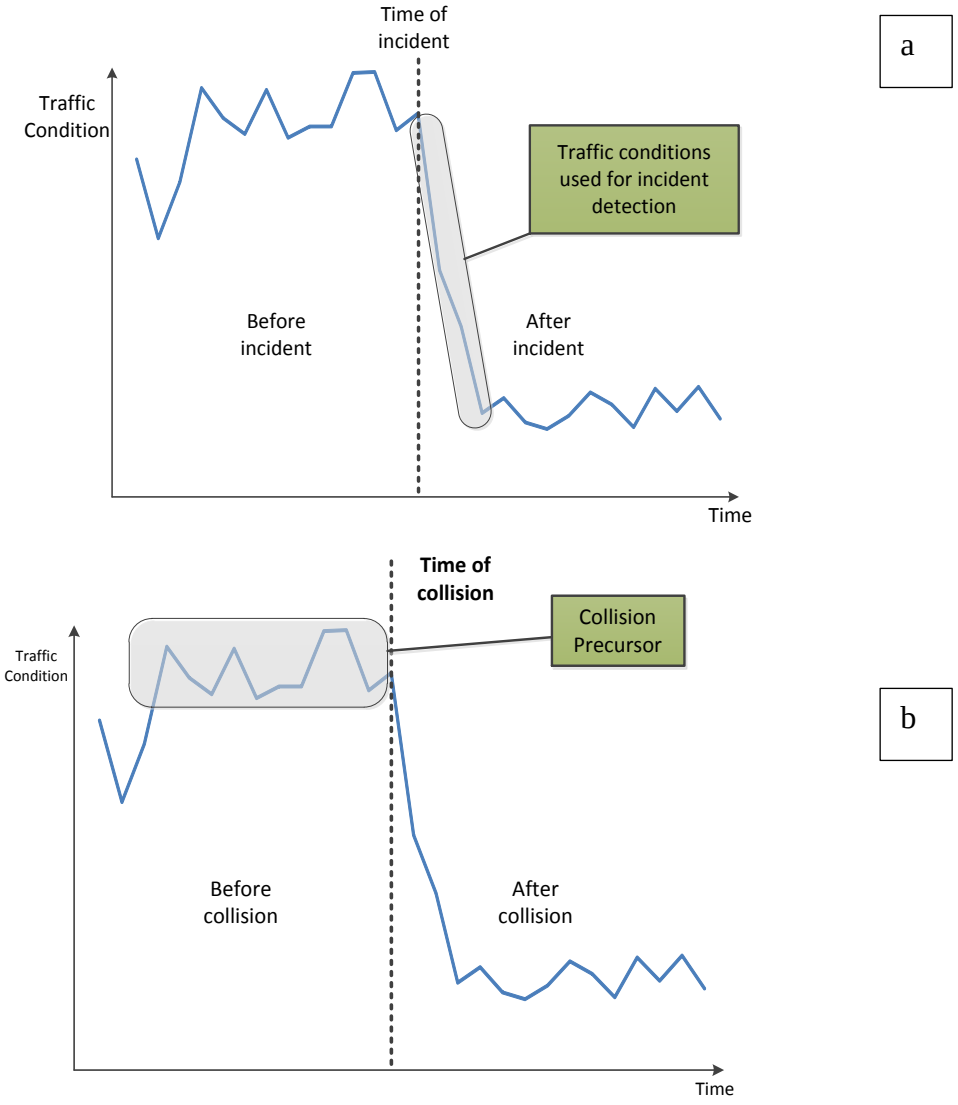


Figure 3. 2 Reactive (a) vs Proactive (b) safety approaches (adjusted from Abdel-Aty and Pande, 2007)

Although the above mentioned studies accomplished a statistical relationship between real-time traffic and collision occurrences, they lacked in terms of sample size, classification accuracy and transferability issues and the implementation of their results was is not suggested by other researchers(Abdel-Aty and Pande, 2005, Xu et al., 2016a).

3.3.2. *State-of-the-art approaches*

Most recent approaches in NLCP modelling require the utilization of data just before a collision occurrence (termed as collision-prone) as well as data of collision-free (also termed as normal) traffic conditions. The two types of data (collision-free and collision-prone), need to be acquired for both upstream and downstream of the collision's location so as to capture the effect of traffic oscillations in both flow directions. These data categories describe traffic conditions on a road segment where a collision took place. Traffic data resembling collision-prone and normal traffic are usually employed as a matched-case control methodology, in which every collision-prone traffic condition is matched with a number of normal traffic cases. This is so as to single out collision precursors (i.e. traffic indications of an imminent collision). The technique of matched-case control for NLCP studies was initially introduced by Abdel-Aty et al. (2004) and has thereafter been used massively because it eliminates the effects of location, time and weather conditions on the probability of a collision occurrence. In studies employing matched-case control research design, the ratio of collision-prone to safe traffic conditions varies from 1:4 (e.g. Ahmed and Abdel-Aty, 2013) and 1:5 (e.g. Abdel-Aty et al., 2008; Ahmed and Abdel-Aty, 2012) to 1:34 (e.g. Hossain and Muromachi, 2012). In the literature, there is no set rule for choosing a ratio between cases and controls as normally the number depends on the available data. However, according to Roshandel et al., (2015) ratios greater than 1:5 do not result in a statistically significant difference in predicting performance.

3.3.2.1. Traffic data considerations

Regardless of the methodology chosen for real-time NLCP analyses, the traffic data and their quality form the most important factor for the predictive performance of the model. As mentioned before, NLCP models are part of traffic safety management systems, specifically proactive traffic safety management systems. As a result, the aim of such models is to predict a collision occurrence on a road segment, so that

drivers on that segment are informed (through Video Message Signs) of the hazardous traffic conditions and therefore will become more cautious, adjust their speeds and adopt a safer driving behaviour.

3.3.2.2. Temporal Aggregation of traffic data

As the application of NLCP models is the proactive identification of collision-prone traffic conditions, researchers aggregate the raw traffic data coming from various traffic sensors into different intervals of temporal aggregations. Oh et al. (2001), for instance, aggregated traffic data into 5-minute intervals and suggested that 5 minutes just before the collision occurrence should represent *hazardous* traffic conditions while 30 minutes of aggregated traffic data before the crash should imply *safe* traffic. Golob and Recker (2004) discarded 2.5 minutes of data just before the collision event and utilised 30 minutes of aggregated traffic data for modelling real-time collision risk. Abdel-aty and Pande, (2005) stated that raw data (e.g. 20-second, 30-second or 1-minute data) from loop detectors or other traffic measuring devices include random noise and therefore their utilization in collision prediction modelling is burdensome. They divided the 30-minute interval just before a collision into six 5-minute time intervals and concluded that the best results for collision prediction are obtained using traffic data 5-10 minutes before a collision. The same authors (Pande and Abdel-Aty, 2005) utilised 3-minute traffic data aggregation and concluded that it performed worse than 5-minute aggregation. In the study of Ahmed and Abdel-Aty(2012) 1-minute speed data were aggregated to four different levels (2,3,5 and 10 minutes) to estimate the best accuracy for the model and again 5-minute aggregation resulted in the best results.

In the following years, the vast majority of the literature on real-time NLCP (Abdel-Aty and Pande, 2005, Pande and Abdel-Aty, 2006, Abdel-Aty and Pemmanaboina, 2006, Ahmed et al., 2012b, 2012a, Hossain and Muromachi, 2012, Shew et al., 2013, Yu et al., 2013, Hassan and Abdel-Aty, 2013, Wu et al., 2013, Xu et al., 2015b, Wang et al., 2015, Fang et al., 2016, Xu et al., 2016b) followed similar methodologies; traffic data are aggregated in 5-minute intervals and the five-minute interval 5-10 minutes before the crash is used for predicting if a collision is imminent or not. The only differentiations from the majority of studies were found in Xu et al.

(2012) who utilised traffic data from the interval 0-5 minutes before the collision and Xu et al., (2013b), Lin et al., (2015), where traffic data from the interval 10-15 minutes before each collision were used for modelling. Figure 3.3. summarises the temporal variance of the data used to predict collisions in real-time from the reviewed literature. As it can be seen from Figure 3.3., the prediction of each approach is relative to the traffic data used to calibrate the model. For example, if the model is calibrated using data 5-10 minutes before the collision, the model would be able to identify whether the traffic conditions at a specific time moment are hazardous enough to cause a collision in the next 10 minutes.

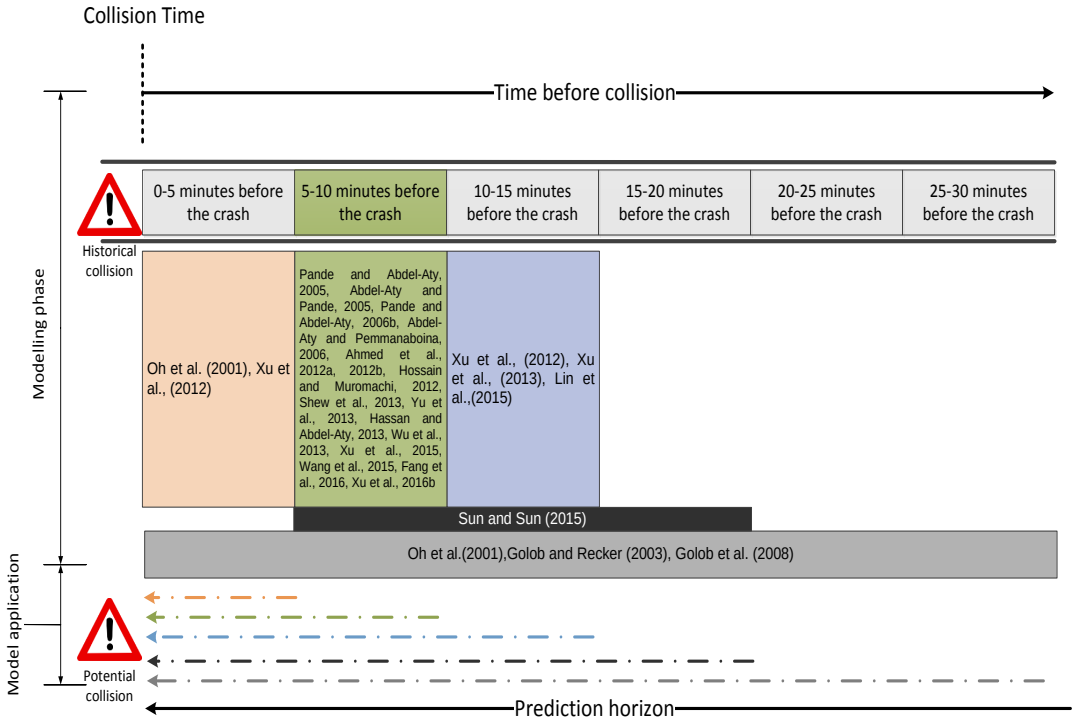


Figure 3. 3 Summary of temporal aggregation used for predicting collisions in real-time and the corresponding predicting horizon

More recently, Peng et al. (2017) attempted to correlate collision risk with microscopic traffic data (raw loop detector data) along with surrogate safety measurements (e.g. Time-to-Collision or TTC). However, their focus was the identification of weather and kinematic characteristics leading to fog-related collisions only and not the identification of collision-prone traffic conditions.

3.3.2.3. Predictors included in real-time collision prediction models

NLCP models utilise traffic surveillance devices which measure speed and count of overpassing vehicles over time. In the case of loop detectors occupancy is also measured. As a result, the traffic variables included in models are temporal aggregations or transformations (linear or logarithmic) of the above-mentioned measurements. When raw traffic data are aggregated, the average and standard deviations are calculated according to the temporal aggregation interval. The relative difference of the traffic measurements between the closest upstream and downstream detectors to a collision location (e.g. detectors U1 and D1 in Figure 3.1) is also usually employed to identify the traffic oscillations prior to a collision occurrence and has been found to increase the probability of a collision occurrence (Roshandel et al., 2015). The coefficient of variation (i.e. the standard deviation of a variable divided by the average of the same variable) is a typical transformation used in the NLCP analyses (Abdel-Aty and Pande, 2005, Pande and Abdel-Aty, 2006, Yu et al., 2013). Within the literature, the coefficient of variation of speed has been found to be associated with an increase in collision probability (Lee et al., 2002, 2003, Abdel-Aty et al., 2004, Ahmed and Abdel-Aty, 2012, Xu et al., 2015b). In order to account for the effect of lane changes, some studies (e.g. Pande and Abdel-Aty, 2006, Xu et al., 2013b, 2014) consider the difference in traffic variables between adjacent lanes as a predictor. The difference in volume between adjacent lanes was associated with secondary collision probability as demonstrated by Xu et al. (2016b) and was explained by the tendency of drivers to change lanes if the number of cars is imbalanced across lanes. In those studies, it was found that the difference in occupancy between adjacent lanes led to increased collision probability. In the study by Hossain and Muromachi (2012) congestion index[9] was also utilised but was not found to be significant in predicting collisions.

It should be noted here that although the average and standard deviation of a traffic variable as well as its transformation are heavily used, there are differences in the spatial aggregation of the measurement. Some of the studies aggregate the measurement of a traffic variable over all the lanes of the motorway (e.g. Pande and Abdel-Aty, 2005, Ahmed and Abdel-Aty, 2012) to avoid having to deal with missing

[9] $Congestion\ Index = \frac{free\ flow\ speed - speed}{free\ flow\ speed}$ (Hossain, 2011)

values or an erroneous detector at a specific moment of time. Other studies only employ the measurements of the lane where the collision happened and thus traffic data are aggregated temporally by lane. The advantage of using lane-based data is that the level of detail in the analyses increases and as a result, specific spatial differences can be further explored more without problems such as the shockwave effect (Abdel-Aty et al., 2005).

3.3.2.4. Variable space reduction

In order to improve the predictability and interpretability of the proposed NLCP models, researchers often perform a technique for the variable selection (Guyon and Elisseeff, 2003). This is a procedure for selecting a subset of predictors from a dataset in order to construct a simpler model which consequently needs less training time and can be generalised more easily (Guyon and Elisseeff, 2003). Variable selection is a more robust technique to select the predictors included in the model compared to engineering judgement and statistical tests such as the t-test (Hossain and Muromachi, 2012).

Regarding the techniques employed in the variable selection procedure, the trend in the literature is to use classification tree algorithms to rank the importance of variables so as to include them in the model. Examples of such algorithms are Classification and Regression Trees (CART) (e.g. Pande and Abdel-Aty, 2005, 2006a, Yu and Abdel-Aty, 2013a), Random Forests (RF) (e.g. Ahmed and Abdel-Aty, 2012, Hassan and Abdel-Aty, 2013, Xu et al., 2013b), Random Multinomial Logit (RMNL) (e.g. Hossain and Muromachi, 2012) and Frequent Pattern Trees (FPT) (e.g. Lin et al., 2015). CART and RFs are usually biased towards variables that have the largest presence in the dataset or have many categories (Strobl et al., 2007) and thus the latter two approaches provide better results than the former. However, it is considered questionable by some researchers (Saeys et al., 2007,Lin et al., 2014)whether the RMNL and FPT can overcome a known drawback of trees, namely the classifier-dependent selection and if their performance is hindered in computationally constrained applications .

Another approach to reduce the variable space in NLCP models is to use clustering

methods like k-means (e.g. Golob and Recker, 2004, Xu et al., 2012, Lin et al., 2015) or Principal Component Analysis (PCA) (e.g. Golob and Recker, 2004, Golob et al., 2008, Wu et al., 2013). The underlying scope is to replace individual traffic measurements into groups of observations (i.e. traffic states or traffic congestion levels) and correlate these groups with the probability of a collision. Nevertheless, microscopic traffic characteristics might be neglected if collision probability is linked with groupings of traffic variables. Furthermore, if traffic variables are to be grouped before their utilization, valuable computational time might be lost in real-time.

3.3.2.5. Methods utilised for analysis

Methodologically, recent real-time NLCP approaches are divided into two broad categories: (1) statistical (e.g. Abdel-Aty et al. 2004, Xu et al., 2014)) and (2) artificial intelligence (AI) or machine learning (e.g. Abdel-Aty and Pande, 2005, Pande and Abdel-Aty, 2006, Hossain and Muromachi, 2012, Yu and Abdel-Aty, 2013a, Xu et al., 2013b, Sun and Sun, 2015).

With regards to statistical approaches, traditional binary logit (Abdel-aty and Pande, 2005) and Bayesian logit ; (Yu et al., 2013) as well as and random parameters logit models (Yu and Abdel-Aty, 2013c, Xu et al., 2015b)) have been applied. In a traditional logit model (i.e. with fixed effects) the estimated coefficients correspond to averaged effects without considering individual diversity. Random parameter models can account for the heterogeneity of road geometry, weather conditions or driving behaviour and have superior performance when compared to traditional logit (Yu and Abdel-Aty, 2014).However, regression models require the determination of a critical odds ratio as a threshold for the identification of collision-prone traffic conditions (Xu et al., 2013a) and also rely heavily on distribution assumptions for both the collision frequency and the traffic parameters.

The first approaches within the machine learning domain for NLCP were concerned with Neural Network (NN) applications. For example, a number of studies (Pande and Abdel-Aty, 2006, Abdel-Aty and Pande, 2005, Pande, 2005) utilised three types of NNs: (i) Probabilistic (Abdel-Aty and Pande, 2005), (ii) Radial Basis Function

(Pande and Abdel-Aty, 2006, Pande, 2005) and (iii) Multilayer Perceptron (Pande and Abdel-Aty, 2006, Pande, 2005)) for real-time collision estimation on American freeways, demonstrating that NNs which do not require any distributional assumptions outperform statistical approaches. NNs usually require a large dataset for training (Vogt and Bared, 2008). However, their major drawback is related to the incorporation of the "black-box" effect, which prevents clear understanding of the model's underpinning properties, interpretation of the model's results and model transferability (Sargent, 2001). Furthermore, NN models often suffer from over-fitting (Yu and Abdel-Aty, 2013b) and require extra computational resources to overcome(Vogt and Bared, 2008). The same "black-box" effect was also documented as a problem for other machine-learning approaches such as Support Vector Machines (SVMs) (Yu and Abdel-Aty, 2013b), although SVMs usually do not result in over-fitting and are flexible with the incorporation of predictors (Dreiseitl and Ohno-Machado, 2002).

Genetic Programming, an extension of Genetic Algorithms (Holland, 1992), was proposed by (Xu et al., 2013b) to remove the "black-box" effect of machine learning approaches, but their model faced difficulties with regards to transferability and practical implementation. In another attempt to tackle the effect of "black-box" Lv et al. (2009) and Lin et al. (2015) utilised the non-parametric algorithm of k-Nearest Neighbours (k-NN). *k*NN is a simple "data-driven" classifier which provides explanation on classification results, addresses the black-box effect and is easily transferrable because it does not require prior knowledge of any datasets.

In order to deal with the drawbacks of previous approaches (both logistic regression and machine learning ones), Hossain and Muromachi proposed Bayesian Networks (Hossain and Muromachi, 2012). They investigated collision prediction on main motorway segments and ramp vicinities by using traffic flow variables and finding an ideal arrangement of detectors for data collection, after hypothesizing that the collision mechanism is different on main segments and ramps. Their study, however, had limited transferability. Sun and Sun (2015) implemented Dynamic Bayesian Networks, an extension of Bayesian Network able to model temporally sequential data. The focal point of their approach is that they treated collisions as an event

triggered by dynamically changing precursors, which is a more realistic view of investigating collision probability over focusing on making point predictions based on aggregated traffic data. Bayesian Networks combine the probability and the graph theory to represent dependencies between predictors and the dependent variable. In order to be able to represent the probabilities of each of the included variables, Bayesian Networks require a sufficiently large dataset which makes them difficult to be implemented with small and unbalanced datasets.

3.4. Drawbacks of existing approaches

NLCP approaches have been constantly improving over time. They have been tested on a variety of motorways (mostly in the U.S.A and China) and have incorporated new types of data due to the improvements in data collection technologies. However, there are some issues that prevent these models from being widely utilised by traffic management agencies. The following subsections aim to identify the particular drawbacks based on the findings from the literature review.

3.4.1. *Modelling methods*

As more data become available in traffic management, machine learning and data mining techniques are becoming more and more of an option in handling large datasets with highly correlated variables in comparison with statistical methods (i.e. logistic regression). On the other hand, the main drawback of machine learning is that their results incorporate the "black-box" effect. NNs and SVMs have provided results of sufficient accuracy (i.e. above 75% according to Abdel-Aty et al., 2005) and low false-alarm rates. However, the interpretation of their classification results is a challenging task. Bayesian Networks may seem a more transparent technique. Yet they require high representations of both collision and safe traffic conditions. In Hossain and Muromachi (2012), for example, 722 collisions and 26,899 normal traffic conditions were used to build the model which shows that requirements for an effective Bayesian Network model might not work in segments with a low number of collisions. Consequently, alternative classifiers within the machine learning domain should be researched to tackle the black-box effect and provide interpretable results.

Furthermore, as the matched-case control study design is the most prominent in NLCP studies, the ratio between control and cases differs significantly. As discussed in the introduction of section 3.3.2, in some cases the ratio was 1:4 (dangerous: normal conditions) while in others it was 1:34. Nonetheless, the ratio between cases and controls can prove essential for the classification results (He and Garcia, 2009, Xu et al., 2016a). The importance of the ratio for the classification results is derived from the fact that if one of the classes (i.e. collision-prone or safe traffic conditions) is overrepresented, the classifier would easily recognise cases of that class and will fail to recognise the class which is underrepresented. It goes without saying that in NLCP studies the underrepresented class is the one associated with collision-prone traffic conditions because of the rarity of collision events. Thus, a potential real-time NLCP classifier needs to be performed well without over-representing safe traffic conditions. This could be done with special imbalanced data classification techniques (He and Garcia, 2009) but has not yet been researched in NLCP models as also suggested by a meta-analysis from Roshandel et al. (2015).

3.4.2. *Temporal aggregation of traffic data*

As discussed in section 3.3.2.1, the majority of the studies aggregate the raw data coming from loop detectors every 20 or 30 seconds into 5-minute intervals. After the aggregation, the time interval 5-10 minutes before a collision is utilised for building the model. The justification behind the 5-minute aggregation is that raw data include random noise and are difficult to implement in a modelling framework. The aggregation, on the other hand, does not reflect the vehicles' trajectories efficiently and also highlights the absence of an underpinning theory of selecting temporal aggregation intervals for NLCP models (Roshandel et al., 2015). Moreover, it is argued that the interval 0-5 minutes before a collision is not a sufficient one for effective traffic management interventions to prevent a collision and smooth traffic after an occurrence. As depicted in Figure 3.3., the prediction horizon is relative to the upper limit of the time interval of which traffic data are utilised. Thus, the majority of approaches are looking into predicting a collision occurring in the next 10 minutes.

In the current era, where autonomous vehicles are closer to reality, it becomes essential that collision prediction should utilise more disaggregated data and should aim at a reduction of the prediction horizon. If transport engineering methods are to be incorporated into autonomous vehicles' planning, then prediction should be performed in a second-base rather than a 5-minute base. If, for example, NLCP classifiers were calibrated using the raw detector data (e.g. with 20-second or 30-second data) or even traffic data aggregated every minute, then the prediction horizon would decrease significantly. A potential collision would be predicted not for the next 5 or 10 minutes, but for a time interval of a few seconds, which is much more essential for applications in autonomous vehicles, vehicular communication and modern ITS. The advances in machine learning research should also become easier in the tasks of reducing noise in raw traffic data and utilizing them in collision prediction. As a result, researching the use of highly disaggregated traffic data should be further explored.

3.4.3. *Temporal precision and underreporting*

Traffic data used in NLCP models are in accordance with historical collision data that exist in national databases. These databases include the reported collision time and this time is used in the modelling process as the starting point for traffic data aggregation. However, a known problem regarding the reported time of collision is that its correct reporting is in the volition of the officer who is first on the spot of the collision. The reported time is usually rounded up to the nearest 5-minute time period (Kockelman and Ma, 2007, Imprialou, 2015, Roshandel et al., 2015). Consequently, this leads to traffic data which are misrepresenting the traffic conditions just before a collision. It also induces ambiguity in the comparison between traffic conditions that lead to a collision versus the traffic conditions under normal operations. Moreover, another limitation of existing collision databases is that only serious or fatal collisions are usually reported. Slight collisions or near misses are not easily documented(Yamamoto et al., 2008, Tsui et al., 2009, Department for Transport, 2016) and thus they are not included in safety databases. Thus, documentation of these cases and their utilization in safety analyses should enhance proactive real-time collision modelling.

3.5. Safety analyses using traffic microsimulation

Traffic simulation models provide a mathematical or logical representation of a traffic network in order to quantitatively describe the performance of a network and that of a user. In theory, they could provide a potential solution to the problems of collision underreporting and erroneously reported collision time. Hence in this section safety analyses based on traffic microsimulation will be critically reviewed.

In general, there are two types or abstraction levels for modelling traffic; macroscopic and microscopic. The former, is usually termed as traffic macrosimulation and the latter is known as traffic microsimulation. Macrosimulation follows a top-to-bottom approach focusing on modelling traffic as combined clusters of vehicles. More specifically, traffic dynamics are characterised by the spatial vehicle density and the average vehicle speed as a function of a motorway location and time (Helbing et al., 2002). In other words, traffic microsimulation aims at describing the time-space evolution of the fundamental traffic variables (i.e. speed, volume and density) and analysing traffic flow in a way analogous to that in which fluids are studied in hydrodynamics (Barcelo, 2011). However, this high-level description does not allow looking at the independent movement of vehicles and the acquisition of highly disaggregated data.

On the other hand, traffic microsimulation analyses the independent vehicle motion within a specific traffic course. Accelerations, decelerations, lane changing behaviour, gap acceptance and car-following models are conceived as integral parts of the modelling procedure. Microsimulation models have primarily been developed in order to evaluate alternative treatments at sites and as an essential part of designing and visualizing transport designs so as to optimise traffic operations. However, transport research has recently begun to utilise traffic microsimulation for safety assessment.

3.5.1. *Conflicts as a surrogate for collisions*

Recent research on traffic microsimulation and road safety (e.g. El-Basyouny and Sayed, 2013, Shahdah et al., 2015) showed that it is possible to estimate surrogate measures of safety performance based on dangerous vehicle interactions. If these

risky vehicle interactions are filtered with established risk indicating thresholds, they are termed as "traffic conflicts". The traffic conflict technique (TCT) was a procedure firstly described by Perkins and Harris (1967) which quantified evasive manoeuvres as a surrogate to reduce safety-critical situations. Amundsen and Hyden (1977) provided the definition of traffic conflicts. According to that definition, traffic conflicts occur when two or more road vehicles are in such a collision course that a high probability of a collision exists if their motion remains uninterrupted.

TCT has been criticised because traffic conflicts need to be validated from on-spot observers and usually the quantity of conflicts is subjective to the observer's judgement and hence it is difficult to link conflicts with observed crash data (Cunto, 2008).

Hyden (1987) introduced the renowned pyramid which depicts the transition from normal vehicle interactions to collisions as seen in Figure 3.4 indicating that the largest proportion of vehicle interactions are safe and that collisions are only a small fraction of serious conflicts. The representation of traffic interactions as a continuum leads to a conclusion that there exists a relationship between the number of serious conflicts and collisions (Yang, 2012). In the same work of Hyden (1987), it is stated that the collision severity distribution is similar between traffic conflicts and traffic collisions. This is also supported by other studies (Archer and Kosonen, 2000, Shahdah, 2014) which showed that a higher rate of traffic conflicts at a specific location indicates a lower level of safety.

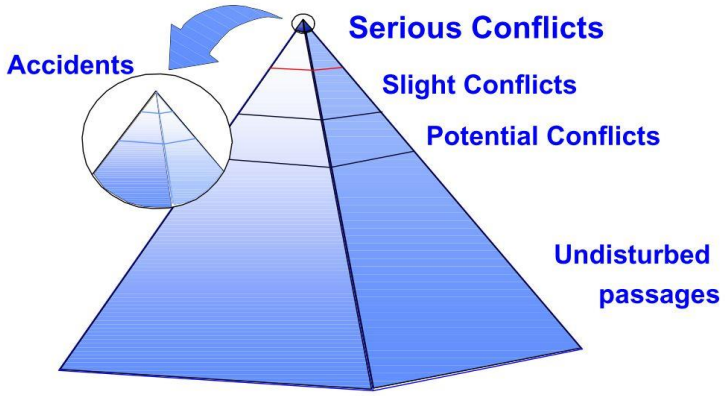


Figure 3. 4 Hyden's Pyramid depicting the proportion of safe and dangerous traffic incidents (Hyden, 1987)

Using traffic conflicts can, therefore, address issues related to traffic collisions such as the incorrectly reported collision time as discussed above. Furthermore, studying conflicts can enhance the understanding of the specific characteristics that lead road users to drive unsafely and cause collisions (Yang, 2012, El-Basyouny and Sayed, 2013). Approaches that use traffic conflicts are, however, criticised in the literature, because the correlation between traffic conflicts and traffic collisions on a segment may be low (Shahdah et al., 2015). Nevertheless, it is also admitted that the mechanism that triggers collisions and conflicts is analogous (Shahdah et al., 2015, El-Basyouny and Sayed, 2013).

In all traffic microsimulation platforms, simulating traffic collisions is not possible because such software is programmed according to a number of safety-related parameters. These parameters include the free-flow speed of cars, inter-vehicle headways, acceleration or deceleration profiles, the interaction between priority and non-priority vehicles, appropriate overtaking and lane-changing gaps as well as the obedience of traffic regulations (Bonsall et al., 2005). Despite these safety related constraints, the fact that vehicles can come very close to each other and the information on vehicles' exact positions, speeds, headings and accelerations can provide a relevant safety index for vehicle interactions (Huguenin et al., 2005).

Minderhoud and Bovy (2001) suggested that traffic micro-simulation can overcome the need to collect collision data and also provide alternatives to the safety evaluation of ITS technologies. They indicated that safety indexes, such as TTC and the vehicles' headway distribution as provided by traffic microsimulation software, can reveal safe and unsafe driving patterns. Likewise, Archer (2005) stated that the traffic conflict technique based on the results from micro-simulation could have a practical impact and provide an insight into the identification of safety problems in real-world traffic environments. Archer indicated that a simulation model represents in great detail the geometric, traffic control and traffic flow characteristics of a location, parameters which directly influence traffic safety. In order to assess safety within traffic microsimulation environments, Gettman and Head (2003) investigated the potential of detecting traffic conflicts from surrogate safety indicators such as TTC, PET, the maximum speed of the vehicles, the deceleration rate and the speed

differential between the vehicles. Their work was reflected in the development of SSAM, a post-processing software which investigates simulated vehicle trajectories and detects the number and severity of traffic conflicts accompanied by surrogate safety measures for each conflict. Currently, SSAM is probably the only exceptional tool for exploiting traffic conflicts from microsimulation (Huang et al., 2013).

The convenience in terms of the reduced need for on-field data collection and the relatively easy identification of hazardous vehicle encounters through safety indices led to several safety-related microsimulation studies. A detailed overview of approaches concerning safety-related traffic simulation was published by Young et al. (2014). In their review, it is revealed that researchers are looking to establish a correlation between the numbers of simulated conflicts with the number of expected real-world collisions. El-Basyouny and Sayed (2013) justified the attempt to link conflicts with collisions by indicating that conflicts are based on vehicle interactions compared to typical collision predictors such as exposure. Essa and Sayed (2015a) and Huang et al (2013) however emphasised that the link between conflicts and collisions depends heavily on the calibration of the simulation model. On the same principle, Fan et al. (2013), who investigated the safety of motorway merging areas, suggested that SSAM should be used with caution because of the purely stochastic nature of real-world collisions.

3.5.2. *Surrogate safety measures*

The use of traffic conflicts in road safety assessment using traffic microsimulation has gained popularity within the ITS research community over the recent years due to the development of SSAM as a post-processing tool. The underpinning nature of safety studies using microsimulation is based on using the trajectories of the simulated vehicles and filtering them with safety indicators, so as to extract conflicts. TTC and Post-Encroachment Time (PET) are the two mostly utilised safety indicators and the two measures that are utilised from SSAM as thresholds to filter trajectories. Other surrogate safety measures across the literature include the Deceleration Rate to Avoid a Crash (DRAC) (Cunto and Saccomanno, 2008), Time-to-Accident (TTA) (Archer and Young, 2010) Time Exposed Time-to-Collision (TET) (Minderhoud and Bovy, 2001) and Time Integrated Time-to-Collision (TIT)

(Minderhoud and Bovy, 2001). However, DRAC is usually used to account for the severity of the conflict while TTA requires estimation of trained observers on the studied site. Furthermore, TET and TIT require the initial identification of TTC for all the vehicles on the study site along with further computation, something which makes them useful mostly for post-processing rather than utilization in real-time studies.

The most frequently used metric acting as a surrogate measure of conflict is TTC. Hayward (1972) and Hyden (1987) initially utilised TTC as a surrogate measure and defined it as the time required by two vehicles to collide if they continue having the same speed on the same path. Due to the necessity that speed is kept constant TTC is meaningful if a positive difference between vehicle speeds exists (Yang, 2012). Moreover, as pointed out by Ward et al. (2015), TTC is used only for car-following scenarios. Regarding thresholds that indicate conflicts Sayed and Zein(1999) suggested that TTC values between 1.6 and 2.0 indicate low collision risk, TTC between 1 and 1.5 seconds are associated with moderate risk, while values below 1 second suggest high risk of collision. On the same principle, Archer (2005) suggested an upper threshold of 1.5 seconds should be indicative of dangerous vehicle encounters. Most recently, Dijkstra (2013) defined a threshold of 2.5 seconds to indicate conflicts which is indicative of commercially available forward collision warning systems (Scanlon et al., 2016).

PET is defined as the temporal difference between the moment an "aggressive" vehicle departs from a potential collision area and the moment another vehicle arrives at the same spot (Cunto, 2008). PET is more easily extracted than TTC because it does not require the indication of a collision course between vehicles, nor any relative speed or distance data (Archer, 2005).To elaborate more on the definition of PET, at a specific intersection or road segment, a stationary conflict area is defined and the time difference between two vehicles passing over this conflict area is used to extract PET. However, PET utility is limited to vehicle trajectories that interfere with each other and this originates from the fact that the collision area should be stationary and not dynamically changing per the vehicles' kinematics.

3.5.3. *Microsimulation in safety analyses*

The underpinning nature of using traffic microsimulation for safety analyses is the use of TCT for identifying traffic conflicts on a network that has been coded and simulated.

As it can be interpreted from the previous sections, traffic microsimulation overcomes drawbacks of safety analyses related to underreporting and erroneous collision times. However, one of the main concerns in the use of microsimulation is the connection that surrogate safety measures have with actual crashes. It has been argued among researchers (e.g. Archer, 2005, Cunto and Saccomanno, 2008, Dijkstra et al., 2010) that there is an actual relationship but at the same time a part of the research (e.g. Sharma and Collins, 2014) suggests that the randomness of conflicts and collisions affects a correct TCT-based collision prediction. This ambiguity is also based on the fact that microsimulation models are built on car-following, gap acceptance or lane changing models, which do not allow actual collisions. The underpinning models allow for a pseudo-realistic motion of the vehicles inside the simulation but the reliability of those models in user-defined parameters might result in debatable results which do not resemble human driving behaviour (Huang, F. et al., 2013, Yang, 2012).

Furthermore, although SSAM utilises only TTC and PET, a tide of research aims at developing new surrogate safety measures. This leads to the fact that there is still not a specific surrogate measure validly correlated with collision occurrence. On the other hand, SSAM's acceptability as a post-processing tool has undoubtedly enhanced safety analyses of microsimulation data.

Perhaps the greatest distress of using microsimulation is the calibration and validation of the traffic data used. In order to extract conflicts, the simulation model needs to be well validated. According to FHWA guidelines (Dowling et al., 2004) the validation should be done mainly in terms of volume and travel times. However, according to the same guidelines, the performance of the vehicles in the simulation is up to the analyst's opinion, which can be differentially interpreted. Nevertheless, if

the simulation model is well calibrated, then its use in safety analyses is generally accepted (Young et al., 2014, Shahdah et al., 2015).

Finally, the number of simulation runs needs to be such so as to ensure stochasticity in the model. Multiple runs are crucial so as to ensure that the behaviour of simulated vehicles for each model is different, thus resembling the randomness in human driving behaviour(Dowling et al., 2004). The number of runs, however, differs significantly among studies. For example, Sobhani et al. (2013) used three runs for their research, Dijkstra et al., (2010) used 36 runs while Shahdah et al. (2014) and Habtemichael and De Picado Santos, (2014) ran their model 50 and 60 times respectively. Consequently, it is understood that the higher the number of runs, the more random will the resulting model be and thus more representative of real-world traffic environments.

A thorough examination of papers attempting to link conflicts with collisions (e.g. (El-Basyouny and Sayed, 2013, Essa and Sayed, 2015b, Shahdah et al., 2015) reveals that the primary aim of these papers is the before-and-after evaluation of new technologies or infrastructure modifications with regards to safety. More specifically, these approaches seek to estimate if alterations to the current state of (a part of) the traffic environment will increase or reduce the number of collisions on specific spots.

As a consequence, an emerging research gap is that of using the simulated conflicts for the identification of real-time conflict-prone traffic conditions. Although vehicles in microsimulation do not collide, they have abundant interactions with each other and their motions are realistic because of the built-in car-following and lane-changing models. Hence, if proper attention to the correct calibration of the microsimulation model is given, traffic conditions before a traffic conflict can be used as a surrogate measurement to identify traffic collisions.

3.6. Summary

This chapter presented a literature review of studies researching network-level collision prediction models. The models were reviewed with regard to the methods utilised for the study, the traffic data and their prediction horizon, as well as the variables included in the models.

As far as the applied methods are concerned, machine learning approaches have recently become more popular due to their effectiveness in handling correlated predictors and erroneous or missing data. However, their weakness lies in the fact that most of the methods act as “black-boxes” and do not allow an easy interpretation of their results.

The traffic data utilised in the studies are obtained from data collection devices every 20 or 30 seconds. These traffic data are later aggregated in 5-minute intervals and the time interval 5-10 minutes before a collision is used to build the models and predict future collisions. As a result, the prediction horizon of current studies aims to predict collisions in the future 5 or 10 minutes and warn drivers to adjust their driving attitude.

Regarding the variables utilised in the models, they usually include the 5-minute average and standard deviation of speed and volume. More importantly, standard deviation of speed and the coefficient of variation of speed are mostly correlated with high collision probability. Additionally, recent studies utilise variable selection methods to reduce the variable space with Random Forests being the most utilised method for that task.

Current approaches concerned with real-time NLCP are mainly limited due to three problems:

- The imbalance of the traffic datasets which over-represent safe traffic conditions and underrepresent collision-prone traffic conditions
- The failure to utilise raw traffic data or traffic data aggregated at time intervals smaller than 5-minutes
- The problem of erroneous reporting of the time of collisions by attending police officers and the underreporting of slight collisions and near-misses.

Traffic microsimulation can provide highly disaggregated details on the vehicle motions within a coded network. The vehicles inside a simulation environment are bounded by car-following, lane-changing and gap-acceptance models and therefore cannot collide with each other. The development of SSAM, however, brought about

a boost in safety-related studies along with microsimulation. These studies use the traffic conflict technique to obtain "dangerous" vehicle encounters but to date are limited to before-after studies which attempt to correlate the total number of crashes at a site based on simulated conflicts. Nonetheless, simulated vehicles' motion is realistic and the exact time of a conflict could be obtained from SSAM. Hence the study of traffic conditions just before a conflict and its use as a surrogate for representing collision-prone traffic conditions should be investigated.

3.7.Identification of research gap

As reviewed in Chapter 2, the planning module of an autonomous or self-driving car should ensure safety and comfort for the passengers. It should also put the car in the right behaviour with respect to the kinematic and motion model constraints surrounding the car. Many of the collisions taking place today are as a result of imprecise perception and decision making on the part of the human driver. Autonomous driving is envisaged to drastically reduce such mistakes since accurate risk assessment is vital for preventing collisions. Although current AV systems have been successfully applied to finding paths and detecting obstacles in real environments, collisions still occur. Hence, greater emphasis must be given to accurate risk assessment in real-time.

Existing approaches to the problem of planning originate from earlier developments within robotics which treat the car as an individual isolated entity. An autonomous vehicle will be a participant of a wider (mixed) traffic system. Complex traffic scenarios are difficult to tackle and learning specific manoeuvres of the drivers and classifying them as *safe* or *dangerous* are time-consuming due to the massive datasets needed. In order to address these challenges traffic–related information is starting to become part of risk assessment models. Nevertheless, the complexity of the proposed models is high and assumptions regarding the communications between vehicles may hinder a comprehensive but simple representation of the traffic environment. Last but not the least, existing planning approaches incorporating traffic-related information is limited to road geometry and the obedience to traffic rules which do not provide a wider picture regarding the safety level of traffic conditions.

The literature review in Chapter 3 revealed that NLCP is a problem that has been researched by ITS experts for many years; resulting in tried and tested methods which indicate network-level collision risks (Figure 3.5). The extensive research on NLCP has provided techniques which have been tested on ATM systems and can predict and prevent actual collisions in real-time (e.g. Xu et al., 2016a, Hossain, 2011 Abdel-aty and Pande, 2005). As real-time highly disaggregated traffic flow data and historical collision data are publicly available, the indication of hazardous situations can be implemented relatively easy and support the planning module of autonomous vehicles. If the NLCP problems of data imbalance, traffic data aggregation and misreported collision time, are addressed, it could lead to a potential improvement in AV decision making as an early indication of dangerous road segments would be provided. Moreover, it could ease the computation and evaluation of hazardous situations for AVs in real-time, and, at the same time, increase their perception horizon. This can be achieved if NLCP models were to be incorporated within autonomous vehicle planning modules and more specifically manoeuvre planning modules. However, an integrated approach to bridge vehicle-level and network-level risk assessment is yet to be fully understood and utilised.

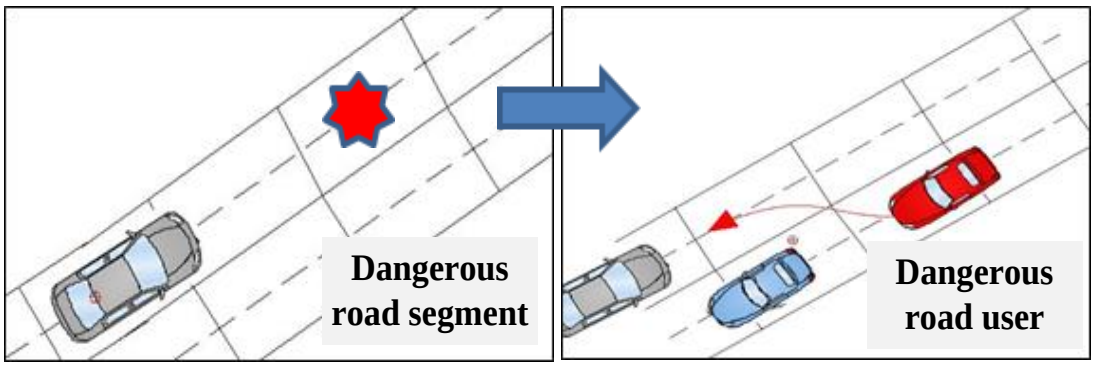


Figure 3.5 From network level risk to vehicle level risk

4. Research Methodology

4.1.Introduction

The body of research reviewed in the previous chapters indicated that there are two main approaches in predicting a collision in real-time, namely network-level (i.e. NLCP) and vehicle-level (i.e. collision risk assessment within AV manoeuvre planning).

NLCP is the outcome of traffic engineering research attempts to identify patterns of traffic dynamics which may lead to traffic collisions. More specifically, the temporal horizon of collision prediction in NLCP models has been found to correspond to the aggregation interval of traffic data that has been collected and collated so as to denote collision-prone or safe traffic.

However, as yet, this approach has not been integrated within the risk assessment module of an autonomous vehicle. Vehicle - level collision prediction is mainly based on the classical *trajectory prediction and collision detection* process which may introduce complexity to the computation in estimating the probability of a collision. In the majority of the techniques every vehicle is assumed to be an independent entity. Only a limited number of the most recent approaches (e.g. Lefèvre, 2012, Ward et al., 2014), indicate the importance of taking the context into account while assessing risk. The term "risk" henceforth will represent the probability of a collision happening.

One of the primary ambitions of this work is to formulate a framework where network-level risk and vehicle-level risk are simultaneously taken into account, so as to examine the influence, for instance the improvement/decline of correct predictions, that network-level risk assessment can have in predicting or mitigating a collision in autonomous driving applications. This framework is based on the interaction-aware motion models that were found in Chapter 2 to be the state-of-the-art in risk assessment for autonomous vehicles.

The most important problems of current NLCP models relate to the erroneous reporting of collisions, the temporal aggregation of traffic data and the imbalance of datasets. To improve the interpretability of NLCP models a machine learning classifier which combines the prediction of a collision with an associated probability is developed. In order to overcome the problem of the erroneous collision reporting, simulated traffic data in conjunction with available real-world data are utilised. Moreover, in order to enhance NLCP, so as to tackle the data imbalance problems, imbalanced learning is applied.

The developed models will assist the incorporation of NLCP models in autonomous vehicles applications considering their more robust and interpretable predictions. The development of a framework in which NLCP models can be included in the routines of autonomous vehicles will also be described, in an attempt to bridge the gap between the vehicle-level and network-level perception of collision prediction.

4.2.Research Design

The aim of this PhD study, as described in Chapter 1 has been divided into 6 objectives. Table 4.1 illustrates the objectives and methods utilised to accomplish the aim of this PhD study.

Table 4. 1 Research objectives, methods and corresponding chapters

Objective ID	Objectives	Methods	Chapter
1	To investigate existing motion planning and collision risk assessment for autonomous vehicles	Literature review	Chapter 2
2	To explore factors and methods related to NLCP	Literature review	Chapter 3
3	To refine traffic and collision data so as to enhance the quality of the analysis	Utilization of highly aggregated and disaggregated data from real-world databases and traffic microsimulation	Chapter 5
4	To formulate a framework for the incorporation of NLCP within the risk assessment module of an AV	Development of a collision risk assessment framework based on an interaction-aware model	Chapters 4
5	To enhance the performance and interpretability of current machine learning classifiers used for NLCP	Utilization of probabilistic machine learning algorithms and imbalanced learning	Chapters 6
6	To evaluate the framework for risk assessment of AVs	Estimation of the expected changes in collision prediction using the integrated collision risk assessment model	Chapter 7

Objectives 1 and 2 have been discussed earlier in Chapters 2 and 3 which reviewed the relevant literature. The following sections will be discussing the methods used to approach the remaining objectives.

4.3. Integration of network-level and vehicle-level collision prediction (Objective 4)

4.3.1. *Introduction*

As mentioned before, this work aims to enhance the estimation of network-level collision risk so that it can be incorporated into vehicle-level collision risk assessment modules. Interaction-aware motion models which perform under the Bayesian principle and take the traffic into account are a potential candidate for the formulation of an integrated risk assessment framework, as found in the literature review of Chapter 2.

Time-varying traffic scenes have to be modelled appropriately so that an ego-AV is able to reliably estimate the risk of a collision according to the surrounding vehicles, as well as the interactions between them. Therefore, an appropriate framework for modelling dynamic systems (i.e. systems with characteristics that change through time such as a continuously changing traffic scene) must be applied.

Data acquisition for AVs is dependent on the temporal frequency of their built-in sensor unit. As a result, input data to the risk assessment algorithm are inherently sequential. Approaches for handling such data can be divided into two parts according to Murphy (2002):

- Classical approaches such as ARIMA, ARMAX, Neural Networks (NNs) and Decision Trees;
- State-space models such as Hidden Markov Models (HMMs) and Kalman Filter Models (KFMs)

State-space models outperform classical approaches in problems associated with finite-time windows, discrete and multivariate inputs or outputs and they can be easily extended (Murphy, 2002). A known drawback of HMMs is that they suffer from high sample and high computational complexity. More specifically, consider the problem of modelling the motion of objects through a camera images sequence. If there are M objects, each of which has k positions and orientations, there are k^M

possible states of the underlying system. An HMM would require k^M distinct states to model the system and the necessary parameters would need to be fully described by the available data or severe over-fitting might occur (Ghahramani, 2001).This means that learning the structure of the model and inferring the required probability necessitates more time to accomplish. Furthermore, simple HMMs require a single discrete random variable, which cannot cope with the description of a constantly changing environment, such as a traffic scene. Although, factorial HMMs and coupled HMMs enable the use of multiple data streams, the former has problems related to the correlation between the hidden variables, while the latter needs the specification of many parameters in order to perform an inference (Murphy, 2012). Finally, KFMs rely on the assumption that the system is jointly Gaussian which makes it inappropriate to jointly accommodate both discrete and continuous variables (Murphy, 2002).

In order to overcome the above limitations in handling sequential data, Murphy (2002) proposed the use of DBNs. DBNs are an extension of Bayesian Networks ,a graphical representation of a joint probability distribution of random variables, to handle temporal sequential data (Koller and Friedman, 2009). DBN representation of the probabilistic state-space is straightforward and requires the specification of the first time slice, the structure between two time slices and the form of the Conditional Probability Distribution (CPDs). A crucial part in defining a DBN is the declaration of hidden (i.e. latent) and observed variables.

When applied for the anticipation of the motion of the vehicles and risk assessment for automated driving, a typical DBN layout that takes the inter-vehicle dependencies into account is shown in Figure 1a (Lefèvre, 2012). The DBN requires the definition of three layers as seen in Figure 1:

Layer 1: the top level corresponds to the context of the vehicle's motion. It can be seen as a symbolic representation of the state of the vehicle (Agamennoni et al., 2012). It can contain information about the manoeuvre that the vehicle performs as seen in Lefèvre (2012 or the geometric and dynamic relationships between vehicles as seen in Agamennoni et al. (2012). The variables contained in this level are usually

discrete and hidden. The variables of the layer are discrete, since they usually denote categorical variables such as manoeuvre undertaken, (e.g. “going straight”, “overtaking”, “lane change”) or the compliance to traffic rules (e.g. “safe lane change” or “illegal left turn”). The hidden nature of the variables corresponds to the fact that they describe quantities not directly observable by the vehicle’s sensor systems, and hence need to be inferred by other measurements. For example, the category of the manoeuvre can be inferred, based on the kinematic characteristics (i.e. vehicle position, speed and acceleration).

Layer 2: this level corresponds to the physical state of the vehicle, more particularly, the kinematics and dynamics of the vehicle). It usually includes information about the position, the speed and the heading of the vehicle, but can accommodate information coming from a dynamic model for the motion of the vehicle such as the bicycle model. The variables contained in this level are usually continuous because they refer to physical quantitative measurements (e.g. speed, position, acceleration) and hidden because they describe filtered or edited measurements which are inferred from the corresponding raw sensor measurements.

Level 3: the lowest level corresponds to the raw sensor measurements that are accessible (e.g. measured speed). In turn, these measurements are processed in order to remove noise and create the physical state subset in Layer 2. The variables at this level are observable.

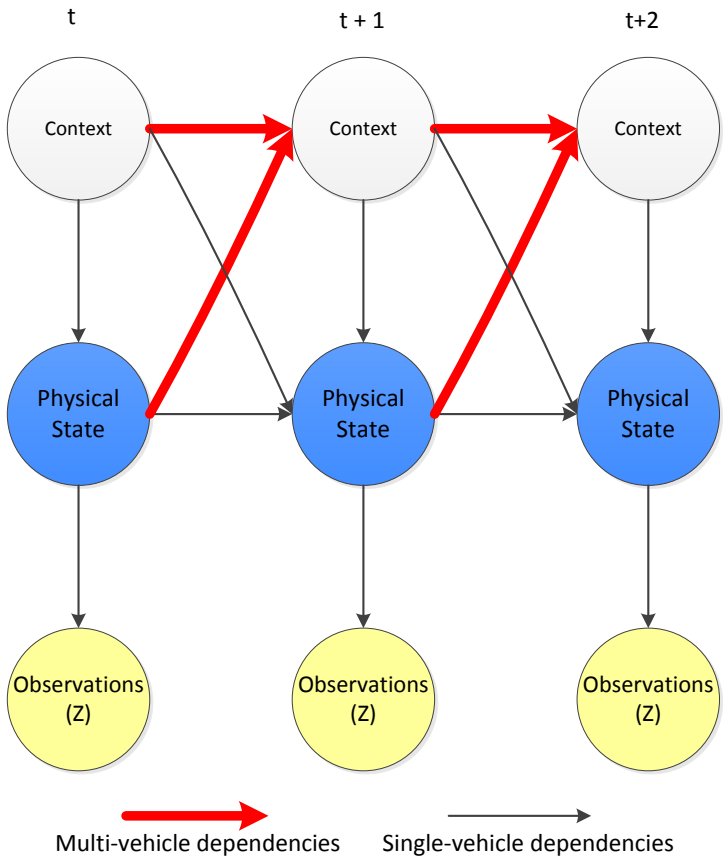


Figure 4. 1 Graphical representation of a typical DBN-based interaction aware model

In Figure 4.1, it is noticeable that for every time moment, the specific context of each vehicle influences the physical state of the vehicle and consequently the physical state is depicted on the observations from the sensors. Accordingly, it is noticeable when focussing on the thick solid arrows that the context of each vehicle at a specific time slice is dependent on the context and the physical state of every vehicle in the traffic scene at the previous time slice. This means that the probability of a vehicle belonging to a specific context in the next time slice requires the estimation of the union of probabilities which describe the context for each of the vehicles in the scene along with the probability distributions of variables related to their physical states. To clarify, assume that an ego-vehicle is travelling in the middle lane of a motorway and senses that a lead vehicle on the left lane intends to change its lane. Based on the traffic rules, it is logical to assume that the ego-vehicle would slow down or change its lane to the right. If a vehicle already occupies the right lane, then the context of

"slowing-down" would have a higher probability than the context of "change its lane to the right" or "change its lane to the left" and the differences in the context would depend on the physical measurements of all vehicles in the scene (i.e. the position and speed of the ego-vehicle and the other two vehicles).

As discussed in Chapter 2, however, this form of DBN is either based on the assumption that vehicular communications are enabled (Lefèvre, 2012) or presents problems in complex traffic scenarios (Agamennoni et al., 2012). To enhance risk assessment for automated driving without increasing the complexity of such DBN-based interaction-aware motion models, a new DBN structure is proposed in this PhD project. This is briefly discussed below.

4.3.2. *Proposed DBN model for motion prediction and risk assessment*

In order to include the NLCP in the motion prediction and risk assessment routine, a new layer and the corresponding dependencies of this specific layer need to be added to the model as depicted in Figure 4.2.

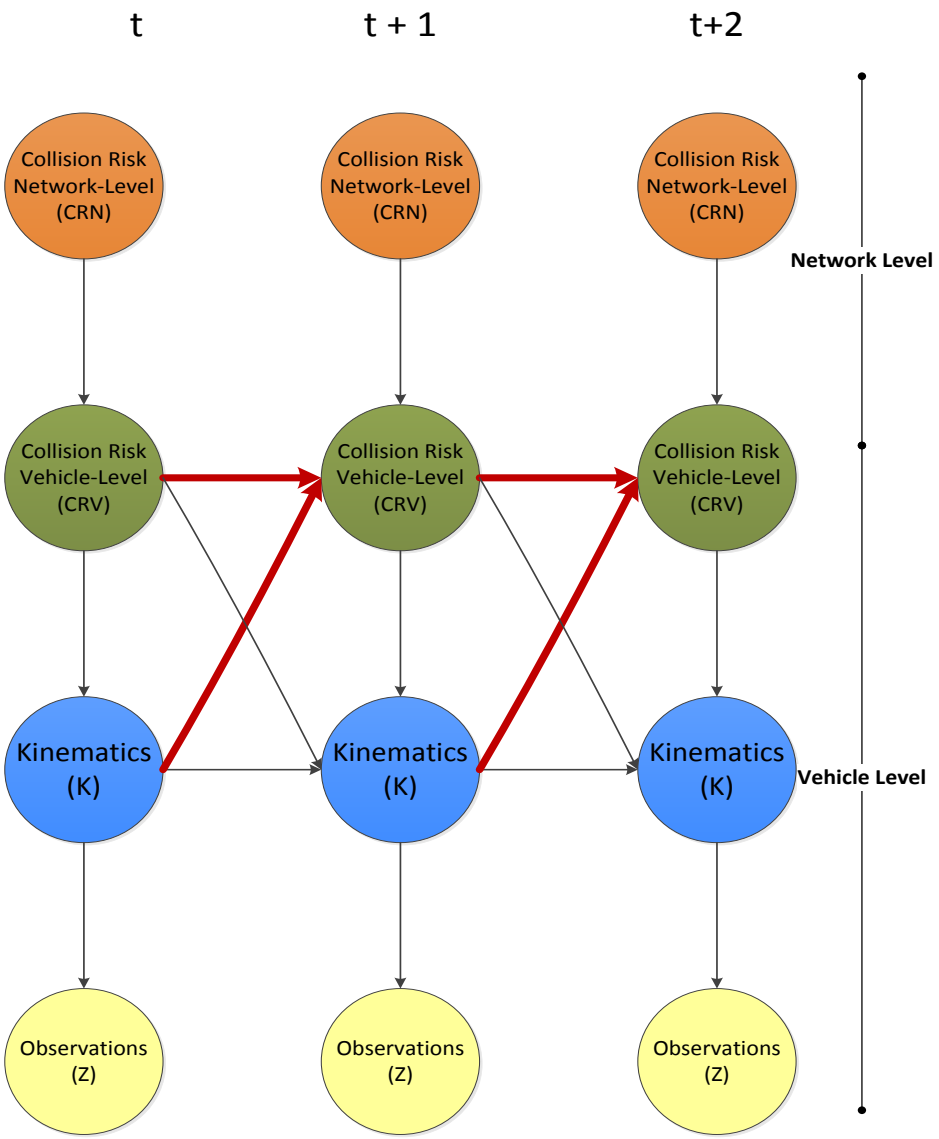


Figure 4. 2 Proposed DBN Network

When comparing Figures 4.1 and 4.2, it can be observed that the context layer is divided into two distinct safety-related contexts: (i) network-level collision risk and (ii) vehicle-level collision risk. The DBN is designed in such a way to represent the dependencies between the layers: i) If there is a safety risk at the network-level, it should be depicted at the vehicle-level, ii) the vehicle-level safety risk is depicted on the motion of the vehicles and iii) the motion of the vehicles is depicted on the observations from the sensors. To elaborate more on the structure of the network, in a potential situation, where traffic dynamics are deemed as collision-prone, the

vehicle-level safety will also be compromised because one or more vehicles will have such a behaviour that may cause a collision. As a result, CRN affects the CRV level. Due to the fact that CRN heavily depends on loop detector or other traffic measurement devices which are sparsely located and because the event of a collision at a time moment is independent of a collision occurring at another time moment, the CRN context is independent at each time moment. Moreover, the safety level of each individual vehicle depends on the motion and the safe or dangerous driving behaviour of other vehicles in its vicinity and therefore CRV depends on the CRV and K probabilities of all the vehicles in the traffic scene as shown with the bold red arrows in Figure 4.1. Furthermore, if the driving behaviour of a vehicle is "dangerous" then the vehicle's motion would also have "dangerous" characteristics such as high speeds or small time and space headways. Hence, the kinematics level of a specific vehicle depends on its current and previous vehicle-level safety context, as well as its previous motion characteristics. Finally, all of the characteristics needed to define the motion of vehicles and their safety context would be depicted on the measurements that the ego-vehicle is receiving through its sensors (i.e. Z depends on K).

The model presented above could, in theory, be applied to any traffic situation by defining the variables CRN, CRV, K, and Z accordingly. However, it is common knowledge that traffic data are mostly available for motorways where magnetic loop detectors and automatic vehicle identification devices exist. These data are utilised to develop NLCP models as described in chapter 2 and hence are necessary to define CRN. Therefore, the developed method is demonstrated for the case of motorway driving, while risk assessment of AVs at junctions is not considered as an example having been the focus of previous research (Lefèvre, 2012, Ward et al., 2014a).

4.3.2.1.Variable definitions

Network-level real-time collision risk (CRN): Represents the safety context of the road segment on which the ego-vehicle is currently travelling on (i.e. whether the traffic conditions on the road segment are *collision-prone* or *safe*). For real-time Bayesian modelling the use of discrete variables is preferable compared to the use of continuous probability distributions as it reduces complexity and enhances computational speed (Bessiere et al., 2013). Hence, the variable in this layer is discrete assuming two values:

1. *Safe traffic conditions*
2. *Collision-prone traffic conditions*

As a result, (CRN_n^t) indicates the probability that the traffic conditions of a road segment on which a vehicle n travels at time t are "*collision-prone*" or "*safe*" based on traffic dynamics. The input variables for estimating network-level collision risk consist of the aggregated traffic conditions data of the road segment (e.g. the mean speed of the vehicles, the mean number of the vehicles and the mean occupancy). Since many vehicles are travelling on the road segment, it is implicit that once the network-level collision risk is estimated for the segment, then its value is the same for all the vehicles travelling on the same segment during the temporal aggregation interval of the network-level prediction.

Vehicle-level risk (CRV): Represents the safety context of a vehicle in a traffic scene, i.e. whether a vehicle can potentially cause a collision with the ego-vehicle. The variable in this layer is also discrete for ease of computations and better real-time results, however it requires four values describing the safety context of each vehicle depending on the network-level safety context:

1. *Safe driving in a road segment having safe traffic conditions*
2. *Safe driving in a road segment having collision-prone traffic conditions*
3. *Dangerous driving in a road segment having safe traffic conditions*
4. *Dangerous driving in a road segment having collision-prone conditions*

As stated, this work is focused on the applications of autonomous driving on motorways. The terms "*Safe*" and "*Dangerous*" driving characterise the manoeuvres undertaken by the vehicle in the traffic scene. More specifically, safe driving does not pose a threat to the ego-vehicle, while dangerous driving indicates that the motion of the vehicle could be considered *unsafe* for another vehicle in the traffic.

In Figure 4.2 it can also be observed that the estimation of the vehicle-level safety context depends on the network-level safety context as well as the union of safety contexts and kinematics of all the vehicles in the vicinity of the ego-vehicle. Hence,

NLCP provides a hint to the estimation of vehicle-level collision probabilities in which the multi-vehicle dependencies are taken into consideration.

Sensor measurements (Z): Represents the available observations derived from the sensors of the ego-vehicle. Z_n^t denotes the available measurements that describe the state of the vehicle n at time t. The variables in this layer are continuous.

The measurements for each vehicle are assumed to include:

$P_{m_t}^{\ n} = (X_n^t, Y_n^t, \theta_n^t) \in \mathbb{R}^3$: the measured lateral and longitudinal position (X_n^t, Y_n^t) and heading of the vehicle (θ_n^t).

$V_{m_t}^{\ n} \in \mathbb{R}$: the measured speed of the vehicles

To distinguish between the raw measurements and the filtered ones, the subscript *m* denotes the measured physical quantities.

Kinematics of the vehicles (K): Represents the physical state of a vehicle, i.e. all the variables that need to be specified in order to localise traffic participants in the vicinity of the ego-vehicle. Specifically, $\boldsymbol{K_n^t}$ denotes the conjunction of all the variables that describe the physical state of the vehicle *n* at time *t*. The variables in this layer are continuous as they are referring to continuously measured quantities such as position and speed.

Based on the available measurements described previously, the following variables are selected to represent the physical state of a vehicle:

$P_n^t = (X_n^t, Y_n^t, \theta_n^t) \in \mathbb{R}^3$: the real values of the position and heading of the vehicle

$V_n^t \in \mathbb{R}$: the real value of the speed of the vehicle

4.3.2.2.Joint Distribution

For the proposed DBN depicted in Figure 4.2 the joint distribution of all the vehicles is estimated as :

$$\boldsymbol{P}(\boldsymbol{CRN}^{\boldsymbol{0:T}}, \boldsymbol{CRV}^{\boldsymbol{0:T}}, \boldsymbol{K}^{\boldsymbol{0:T}}, \boldsymbol{Z}^{\boldsymbol{0:T}})$$

$$= \boldsymbol{P}(\boldsymbol{CRN}^{\boldsymbol{0}}, \boldsymbol{CRV}^{\boldsymbol{0}}, \boldsymbol{K}^{\boldsymbol{0}}, \boldsymbol{Z}^{\boldsymbol{0}}) \prod_{\boldsymbol{t=1}}^{\boldsymbol{T}} \prod_{\boldsymbol{n}}^{\boldsymbol{N}} P(\mathbf{CRN}_{\boldsymbol{n}}^{\boldsymbol{t}})$$

$$\times P(\mathrm{CRV}_{\boldsymbol{n}}^{\boldsymbol{t}} | \mathbf{CRV}_{\boldsymbol{N}}^{\boldsymbol{t-1}} \mathbf{K}_{\boldsymbol{N}}^{\boldsymbol{t-1}} \mathrm{CRN}_n^t) \times P(\mathrm{K}_{\boldsymbol{n}}^{\boldsymbol{t}} | \mathrm{CRV}_n^{t-1}\, \mathrm{K}_n^{t-1} \mathrm{CRV}_n^t)$$

$$\times P(\mathrm{Z}_n^t | \mathrm{K}_n^t) \qquad (4.1)$$

where *n* is the vehicle ID in the vicinity of the ego-vehicle, *t* is the time moment, *T* is the total time duration of the measurements, i.e. the length of the time buffer or the sensor measurements and *N* is the total number of vehicles that are observed in the traffic scene. Equation 4.1 is derived from the DBN in Figure 4.1 by defining the initial state of the network (i.e. $\boldsymbol{P}(\boldsymbol{CRN}^{\boldsymbol{0}}, \boldsymbol{CRV}^{\boldsymbol{0}}, \boldsymbol{K}^{\boldsymbol{0}}, \boldsymbol{Z}^{\boldsymbol{0}})$) and multiplying the probabilities of every node given its parents (i.e. the nodes which affect each probability level) for every vehicle n and for every time moment t. As an example, the parents of the CRV node of vehicle n at time t, is the network-level risk context (i.e. CRN_n^t), and the vehicle-level risk context (i.e. $\mathbf{CRV}_{\boldsymbol{N}}^{\boldsymbol{t-1}}$) and kinematics (i.e. $\mathbf{K}_{\boldsymbol{N}}^{\boldsymbol{t-1}}$) of all the vehicles in time t-1. Bold letters denote that the indicated layers are calculated for all the vehicles. For example, $\mathbf{CRV}_{\boldsymbol{N}}^{\boldsymbol{t-1}}$ indicates the vehicle-level risk context for time *t*-1 for all the vehicles in the traffic scene.

4.3.2.3.Estimating the risk of collision using network-level collision prediction information

Modelling the motion of the vehicles with regards to network- and vehicle-level risks requires that a new estimation framework should be developed. In order to quantify the influence that the network-level risk estimation has on assessing the vehicle-level collision risk, it is essential to infer the probability that there is a vehicle-level "*unsafe*" situation, given the hint from the network and the measurements from the sensors.

In the majority of recent studies on NLCP (e.g. Sun and Sun, 2015), traffic conditions at 5-10 minutes prior to a collision are deemed to be the most suitable for the identification of collision events in time and the initiation of an intervention by the responsible traffic agencies. However, a 5 to 10-minute aggregation may not be adequate for the real-time safety assessment of AVs where sensor information is available at a higher sampling frequency (e.g. 1 Hz, 0.1 Hz). It is, however, a reality

that traffic agencies aggregate traffic data at pre-defined time intervals (e.g. 30-second or 1-minute, 5-minute and 15-minute). Because of the difference at the temporal horizon between NLCP and vehicle-level measurements, it is assumed that the CRN layer is an *observable* layer while CRV and K are *hidden* layers since the variables in these layers are inferred through the sensor measurements. The sensor measurements layer (Z) is obviously an *observable* layer.

Exact inference in such non-linear and non-Gaussian models is not tractable. Therefore, in order to estimate the probability of a "dangerous" vehicle-level context given the traffic situation and the sensor measurements the use of particle filters is proposed (Merwe et al., 2000) as they have been proven to perform well in similar situations (Lefèvre, 2012, Murphy, 2002).

If an inference algorithm is chosen, then the probability to be inferred is as follows:

$$P\left([CRV_n{}^t \in \{dCP, dSA\}] | CRN_t, Z_{0:t}\right) > \lambda \quad \textbf{(}4.2\textbf{)}$$

where:

- $\boldsymbol{CRV_n{}^t}$ denotes the vehicle-level safety context of vehicle *n* at time *t*;
- $\boldsymbol{dCP}, \boldsymbol{dSA}$ denote a "**d**angerous" vehicle travelling on a road segment with **C**ollision-**P**rone traffic conditions and a "**d**angerous" vehicle travelling on a road segment with **SA**fe traffic conditions respectively;
- $\boldsymbol{CRN_t}$ denotes the network-level collision risk for all the vehicles on a specific road segment;
- $\boldsymbol{Z_{0:t}}$ denote the sensor measurements until time moment t;
- $\boldsymbol{\lambda}$ is a threshold to identify "dangerous" encounters between the surrounding traffic participants and the ego-vehicle.

In order to infer the probability in Equation 4.2, the DBN in Figure 4.1 needs to be defined through the joint distribution in equation 4.1 by providing information on the network-level and vehicle-level nodes. Information on NLCP would be provided by traffic management agencies through a communication channel or VMS messages and information on the vehicle-level would be inferred through the sensor

measurements of the ego-vehicle, or would be transmitted by other vehicles through vehicular communications. Equation 4.2 indicates that given a hint for the safety assessment of a road segment, the motion of the vehicles in that specific segment is affected. This resembles how human drivers are also affected when the information of traffic incidents such as a broken vehicle on the roadway or a queue formation in the downstream is displayed via Variable Message Signs.

The following section (i.e. 4.3.2.4) describes how each probability in equation 4.1. is estimated.

4.3.2.4.Parametric forms

In order to estimate the joint distribution of the DBN network for inference, the functions for the probabilistic distributions of each layer need to be defined. Since a large number of variables exist in the problem and the focus of the approach is the incorporation and enhancement of NLCP into existing motion models for automated driving, a brief description of the parametric forms for vehicle-level risk, kinematics and sensor measurements is presented. A more analytic description of the parametric form for network-level collision risk estimation is presented in section 4.3.2.4.4.

4.3.2.4.1. Vehicle-level risk $P(CRV_n{}^t)$

The context of vehicle-level risk is derived from the previous vehicle-level risk context and kinematics of all the vehicles on the scene and is influenced by the current NLCP. For the initiating step, it is assumed that the vehicle-level risk for all the vehicles is "safe driving on a road segment having safe traffic conditions". The estimation of the probability that the motion of a vehicle is considered "dangerous" or "safe" is derived through a feature function that receives as input to the current network-level risk, the previous vehicle-level risk context of the vehicle and the previous vehicle kinematics of all the vehicles in the scene:

$$P\left(\mathrm{CRV}_{\boldsymbol{n}}^{\boldsymbol{t}} \middle| \mathbf{CRV}_{\boldsymbol{N}}^{\boldsymbol{t-1}} \mathbf{K}_{\boldsymbol{N}}^{\boldsymbol{t-1}} \mathrm{CRN}_n^t\right) = \boldsymbol{f}(\mathbf{CRV}_{\boldsymbol{N}}^{\boldsymbol{t-1}}, \mathbf{K}_{\boldsymbol{N}}^{\boldsymbol{t-1}}, \mathrm{CRN}_n^t) \quad (4.3)$$

In order for this feature function to be defined, three steps need to be considered:

a) Using a Kalman Filter (Murphy, 2012), the physical state of the vehicles in a traffic scene can be estimated. For example, after applying a Kalman Filter

algorithm the elements $\{X_{ego}^t, Y_{ego}^t, \theta_{ego}^t, v_{ego}^t\}$ and $\{X_n^t, Y_n^t, \theta_n^t, v_n^t\}$ will be known. $v_{ego}^t \; and \; v_n^t$ denote the speeds of ego-vehicle and vehicle-n respectively.

If p_{ego}^t indicates the position of the ego-vehicle and p_n^t indicates the position of vehicle n, whereas Δv_t denotes the speed difference between the ego-vehicle and vehicle *n,* then the distance-to-collision (δ) and the time-to-collision (TTC) between the ego-vehicle and vehicle *n* are expressed as follows (Agamennoni et al., 2012):

Distance to collision: $\delta_n^t = p_n^t - p_{ego}^t$ (4.4)

Time to collision*:* $TTC_n^t = \frac{\delta_n^t}{\Delta v_t^T}$ (4.5)

If $P_n^t = (X_n^t, Y_n^t, \theta_n^t)$ denote the position and heading of vehicle *n* at time moment *t* and v_n^t denotes the speed of the vehicle, an indicator function (f_K) can display if vehicle *n* brakes dangerously, changes lane dangerously or drives safely with regard to the ego-vehicle. TTC-based thresholds such as the one in (Toledo et al., 2003) could be of use to detect dangerous driving behaviours:

$$f_K = f(\mathrm{TTC}_n^{t-1}) = \begin{cases} 1: \text{dangerous } if \ \mathrm{TTC}_n^t < Critical\ TTC \\ 0: safe; otherwise \end{cases} \quad (4.6)$$

b) If a vehicle in the previous time epoch was identified as “dangerous” in the road segment that the ego-vehicle is driving on, then it is assumed that the CRV context was “dangerous”. Otherwise, it is assumed that the motion of all the vehicles was “safe”. Thus, another indicator function that takes the previous vehicle-level risk of all vehicles into account can be defined as:

$$f_{CRV_N} = \begin{cases} 1 \; if \; \sum_{n=1}^{N} CRV_n^{t-1} > 0 \\ 0, otherwise \end{cases} \quad (4.7)$$

where N is the total number of vehicles that the ego-vehicle can sense.

c) In order to take network-level collision risk into consideration and easily identify dangerous traffic participants, the network-level classification metrics are considered as a coefficient:

d) $$f_{CRN_n} = \begin{cases} \frac{Accuracy+Recall}{2} \; if \; \mathrm{CRN}_N^t = dangerous \; and \; f_{CRV_N}^{t-1} = 1 \\ 1 - \frac{Accuracy+Specificity}{2} \; if \; \mathrm{CRN}_N^t = safe \; and \; f_{CRV_N}^{t-1} = 0 \\ 1 - recall \; if \; \mathrm{CRN}_N^t = safe \; and \; f_{CRV_N}^{t-1} = 1 \\ 1 - specificity \; if \; \mathrm{CRN}_N^t = dangerous \; and \; f_{CRV_N}^{t-1} = 0 \end{cases}$$

(4.8)

By that definition if a vehicle is detected as *dangerous* and the traffic conditions are *collision-prone*, a compromise between the accuracy of the classifier and its recall is enhancing the identification of hazardous road users. If traffic conditions are indicated as safe, then the compromise is made between the accuracy and the specificity of the classifier which exhibits its ability to correctly classify safe traffic conditions. Afterwards, this compromise is subtracted from 1 to indicate the probability of a vehicle being dangerous. When the network-level classifier indicates safe traffic but a vehicle is sensed to be posing a "threat" to the ego-vehicle, then the prediction is boosted by the false negative rate given by the formula: $1 - recall$. Lastly, when traffic conditions are indicated as dangerous but no vehicle posing a threat exists, then the vehicle-level risk is boosted by the false alarm rate (i.e. $1 - specificity$).

Having all three indicative functions, the probability of the current vehicle-level collision risk context could be calculated as in the following example:

$$P(\mathrm{CRV}_{\boldsymbol{n}}^{\boldsymbol{t}} = "dCP \; or \; dSA" | \mathrm{CRV}_N^{t-1} \mathrm{K}_N^{t-1} \mathrm{CRN}_n^t) = \frac{\sum_{n=1}^{N}(f_{K_n}=1) + \sum_{n=1}^{N}(f_{CRV_n}=1) + f_{CRN_N}}{3N}$$

(4.9)

where N is the total number of vehicles that the ego-vehicle can sense. 3N is chosen as a normalising factor, in order for the probability to be within [0,1], even when one vehicle is posing a threat (i.e. $\sum_{n=1}^{N}(f_{K_n}) = 1, \sum_{n=1}^{N}(f_{CRV_n}) = 1 \; and \; f_{CRN_N} = 1$). Equation 4.9 was derived after trials with the three indicative functions (i.e. f_K, f_{CRV}, f_{CRN}) and its purpose is to resemble the dependence between the current

vehicle-level risk context with the kinematics and vehicle-level context of the all the vehicles in the previous time slice, as well as the NLCP predictions at the current time slice. It is assumed that the sampling and risk estimation frequencies will be adjusted as soon as a risk is estimated in a timestep.

4.3.2.4.2. *Kinematics* $P(K_n^t | CRV_n^{t-1}\ K_n^{t-1} CRV_n^t)$

The variables describing the kinematics layer must contain all the information needed in order to characterise the contexts. In this work, it was explained that the physical state vector will contain information on the position of a vehicle in an absolute reference system, its heading and its speed. It is assumed that vehicles move according to the bicycle model as shown in Figure 4.3 (Snider, 2009). The kinematic bicycle model merges the left and right wheels of the car into a pair of single wheels at the centre of the front and rear axles as seen in Figure 4.3. Finally, it is assumed that wheels have no lateral slip and only the front wheel is steerable.

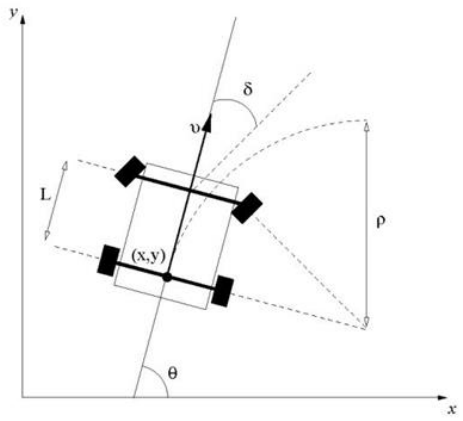


Figure 4. 3 Bicycle model kinematics

The equations of motion for all vehicles in the traffic scene can be integrated over a time interval Δt using a simple forward Euler integration method (Press et al., 1993) in order to acquire the evolution of kinematics over time.

In the proposed model as shown in Figure 4.2 and in its joint distribution as shown in equation (4.1) it is observed that the current kinematics depend on the previous and current vehicle-level risk context as well as on the current kinematics of the vehicle. It is assumed that vehicles moving in a specific context will follow kinematics

according to that context. As a result, the parametric forms of the position, heading, and speed of each of the vehicles should be defined according to the current vehicle context and the previous kinematics only. For example:

$$\mathrm{P}(\mathrm{P}_n^t|\mathrm{CRV}_n^{t-1}\,\mathrm{K}_n^{t-1}\mathrm{CRV}_n^t) = P(\mathrm{P}_n^t|\mathrm{CRV}_n^t\mathrm{K}_n^{t-1}) \tag{4.10}$$

In order to expose the dependency of current kinematic measurements on the previous vehicle-level safety context, context-specific constraints (e.g. constraints on the TTC between ego-vehicle and another vehicle) should be defined so as to distinguish between contexts. For example, if the derived TTC is below 1 second, this could indicate a "dangerous driving" in a road segment with safe or collision-prone traffic conditions. The parametric forms of the probability distribution of position and speed of the vehicles can be assumed to follow normal distributions (Lefèvre, 2012).

For example, the likelihood of the position and heading of a vehicle is defined as a tri-variate normal distribution with no correlation between x, y, and θ

$$P(\mathrm{P}_n^t|[\mathrm{CRV}_n^{t-1} = C_i][\mathrm{P}_n^{t-1} = X_n^{t-1}Y_n^{t-1}, \theta_n^{t-1}][\mathrm{V}_n^{t-1} = v_n^{t-1}]) =$$
$$N\left(\boldsymbol{\mu_{xy\theta}}(X_n^{t-1}Y_n^{t-1}, \theta_n^{t-1}, C_n), \boldsymbol{\sigma_{xy\theta}}\right) \tag{4.11}$$

where $\boldsymbol{\mu_{xy\theta}}(X_n^{t-1}Y_n^{t-1}, \theta_n^{t-1}, C_n)$ is a function which computes the mean position and heading of the vehicle $(\mu_x, \mu_y, \mu_\theta)$ according to the bicycle model and the context-specific constraints, $C_n$denotes the context of vehicle-n and $\boldsymbol{\sigma_{xy\theta}} = (\sigma_x, \sigma_y, \sigma_\theta)$ is the standard deviation which can be acquired from the covariance matrix of the Kalman Filter algorithm.

4.3.2.4.3. *Sensor measurements* $(Z_n^t|K_n^t)$

The sensor model used is adopted from Agamennoni et al.(2012) due to the use of the Student t- distribution, which performs better with outlier data. The sensor model can be defined as:

$$P(Z_n^t/K_n^t) \sim Student(C^T K_n^t, \sigma^2 I, \nu) \quad (4.12)$$

where C is a rectangular matrix that selects entries from the kinematic (physical state), ν are the degrees of freedom, I is the identity matrix and σ is related to the accuracy of the sensor system.

4.3.2.4.4. Network-level collision risk $P({CRN_n}^t)$

In theory, every technique which can be utilised for NLCP can be applied to estimate the probability of a road segment having collision-prone traffic conditions in the proposed DBN. As the problem of identifying whether the traffic conditions at a specific road segment are collision-prone or safe is a binary classification problem, the outcome of every technique would be a binary indication (e.g. 1 for collision-prone conditions and 0 for safe traffic).

In order to transform the classification result, a probability of a road segment having collision-prone traffic conditions can be estimated as:

$$P({CRN_n}^t = "dangerous") = (\frac{Acc+Rec}{2}), \text{if CR} = 1 \quad (4.13)$$

where *CR* is the classification result for the aggregated traffic conditions in real-time (i.e. 0 or 1), and *Acc* and *Rec* are accuracy and recall of the calibrated classifier respectively. The accuracy metric shows the general classification performance of the classifiers and the recall metric shows the ability of the classifier to detect collision-prone conditions. It can be observed from equation 4.13 that if the classifier indicates a collision-prone situation then the probability of the road segment being "dangerous" is estimated by taking into account the overall accuracy of the classifier and its performance in identifying conflict-prone conditions (i.e. recall). It is worth mentioning that when CR=1 the probability of the road segment being safe is:

$$P({CRN_n}^t = "safe") = 1 - P({CRN_n}^t = "dangerous") \quad (4.14)$$

Accordingly, for CR=0: $P({CRN_n}^t = "safe") = (\frac{Acc+Spec}{2})$ (4.15)

$$P({CRN_n}^t = "dangerous") = 1 - P({CRN_n}^t = "safe") \quad (4.16)$$

where *Spec* is the specificity of the classifier (i.e. the classifier's performance in identifying safe traffic conditions). The metrics of accuracy, specificity and recall will be explicitly described in section 4.5.

Based on equations (4.13) - (4.16), the importance of building robust classifiers with fewer false alarms and solid identification of both normal and collision-prone traffic is observable.

Figure 4.4(left) gives a simple flowchart of the procedures and data needed to infer the probability of a "dangerous" road user and Figure 4.4(right) depicts the necessary online steps analytically for updating the joint distribution in Equation 4.1. identifying hazardous vehicles.

The next sections will describe the procedures undertaken in this work to overcome the limitations of existing NLCP classifiers and consequently fulfil objective 5 of this thesis.

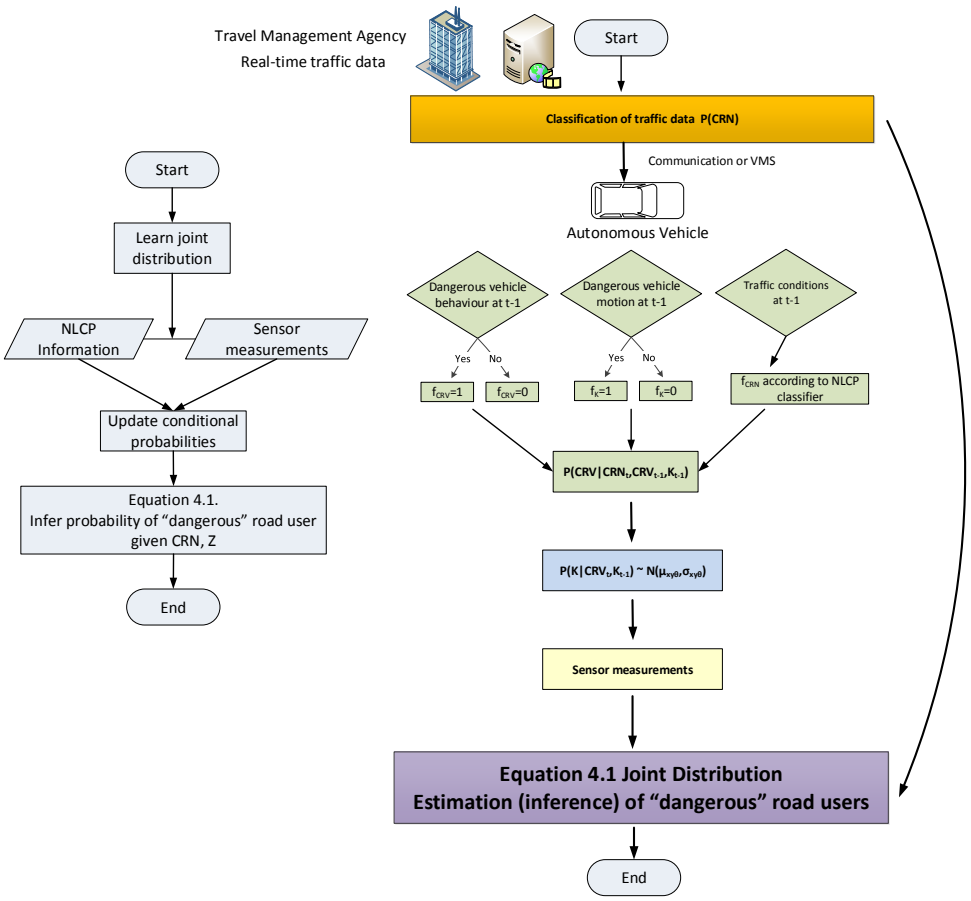


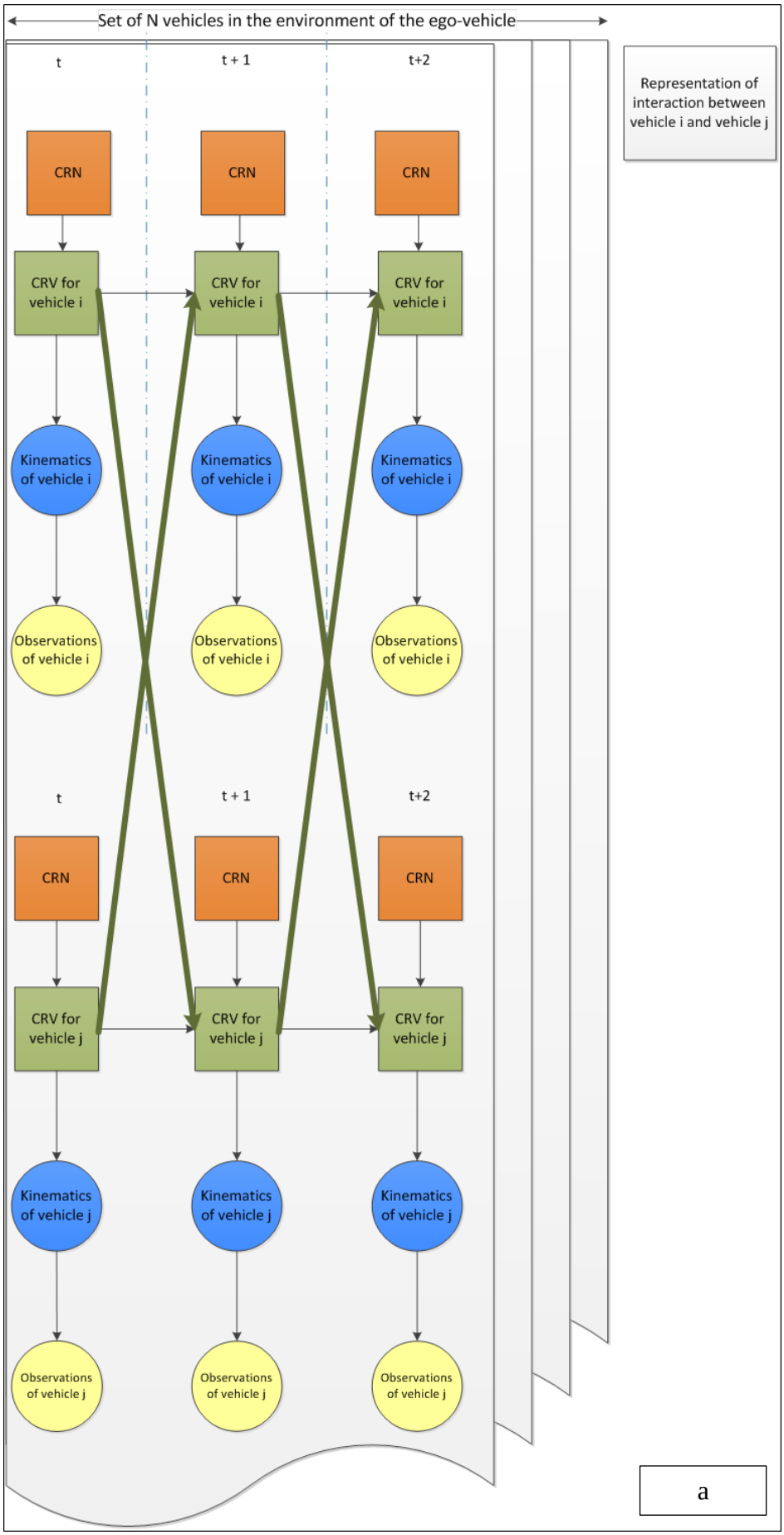
Figure 4.4 Flowcharts of DBN joint distribution estimation

4.3.2.5.Note on the similarities and differences with other probabilistic models

The model depicted in Figure 4.2 bears a resemblance to a Switching State Space Model (SSSM) with regard to the explanation of the dynamics of the traffic scene by switching between a discrete number of contexts. In SSSMs the switching process would be regulated by a discrete Markov process indicating which context is active at every time step. However, in the proposed model, this switching process is conditionally Markov, because the context variable in the vehicle level (CRV) depends not only on the discrete variable of the previous time step but also on the continuous kinematics of the vehicles at the previous time step.

The structure of the proposed model also resembles a Coupled Hidden Markov Model (CHMM) (Brand et al., 1997) given the way the different time slices connect. In CHMMs the current hidden layer depends on the hidden layer in the previous time step as well as the hidden layer of a neighbouring Markov Chain. However, CHMMs are usually intended for the maximum posterior inference, while this work places emphasis on prediction. The obvious difference is that the proposed model accommodates continuous nodes, whereas CHMMs only function with discrete-valued variables. Furthermore, the use of CHMMs so as to solve the problem this PhD work addresses, introduces computational complexity, since a different CHMM should be constructed for each interaction between two vehicles.

For comparison reasons, the CHMMs and the SSSMs adjusted to the problem of this work are presented in Figure 4.5.


Set of N vehicles in the environment of the ego-vehicle
t
t + 1
t+2
Representation of interaction between vehicle i and vehicle j
CRN
CRN
CRN
CRV for vehicle i
CRV for vehicle i
CRV for vehicle i
Kinematics of vehicle i
Kinematics of vehicle i
Kinematics of vehicle i
Observations of vehicle i
Observations of vehicle i
Observations of vehicle i
t
t + 1
t+2
CRN
CRN
CRN
CRV for vehicle j
CRV for vehicle j
CRV for vehicle j
Kinematics of vehicle j
Kinematics of vehicle j
Kinematics of vehicle j
Observations of vehicle j
Observations of vehicle j
Observations of vehicle j
a

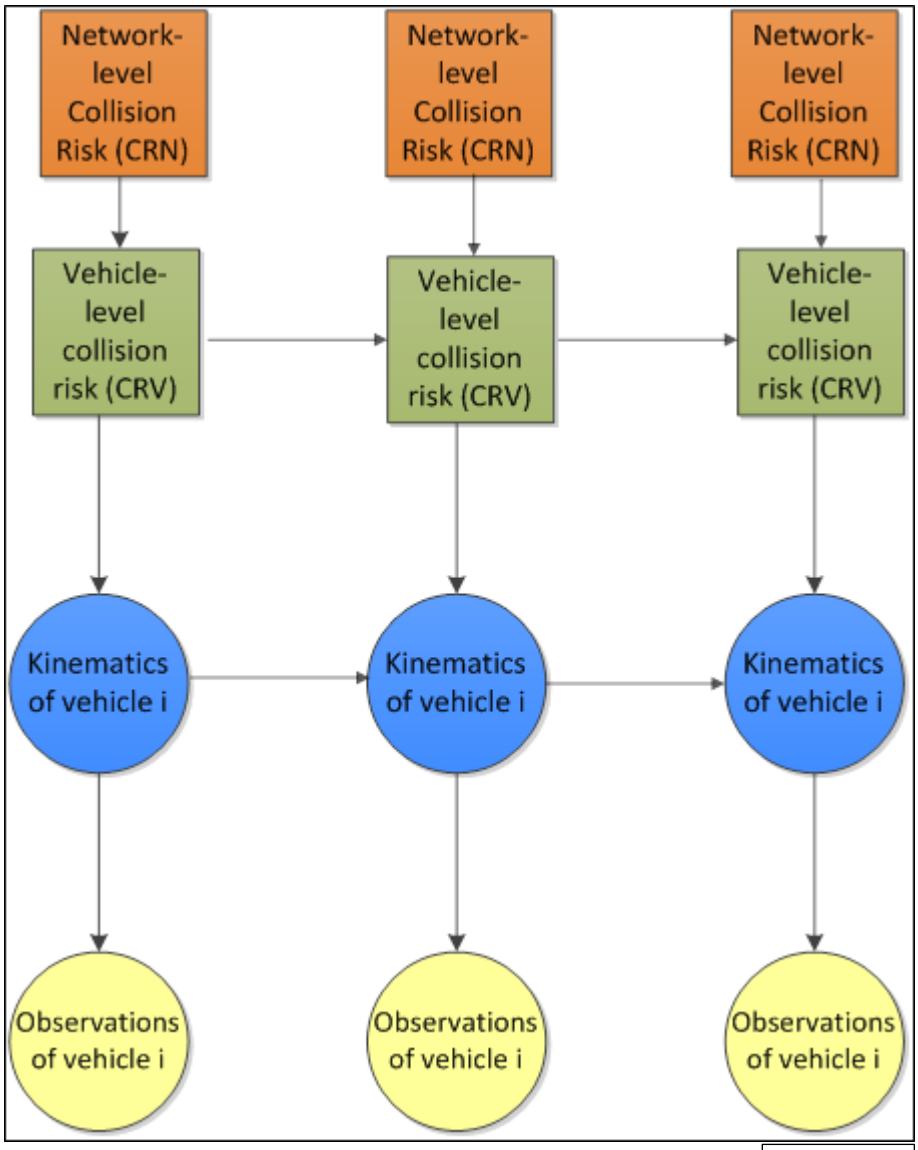


Figure 4. 5 Graphical representation of similar models applied to the problem tackled in this work: a) CHMMs b) SSSMs

4.4. Improving interpretability of machine-learning classifiers for real-time NLCP (objective 5)

As concluded in Chapter 3, which reviewed approaches concerned with NLCP, machine learning approaches are deemed better than classical regression methods to model and analyse highly disaggregated traffic data with respect to the estimation of network-level collision risk. However, the lack of interpretability is a well-

documented disadvantage of machine-learning classifiers. Therefore, the literature was explored to identify methods which overcome this deficiency.

Machine learning approaches aim at identifying a vigorous description of a dataset given a limited sample (Herbrish, 2002). Machine learning is usually divided into two clusters, namely supervised and unsupervised learning.

In supervised learning, a given dataset is labelled regarding the interested response variable. For example, in a collision prediction dataset, traffic variables at a specific time moment are labelled according to their existence in collision-prone or safe time periods. If the response variable in supervised learning is categorical, the problem is known as classification or pattern recognition (Murphy, 2012). More specifically, in supervised classification for every data point, the output is known a-priori and learning aims at discovering an underpinning function so that if new data become available, they can be correctly labelled. After learning, the predicted response of a data point is contrasted with the initial one in order to assess the classification performance.
If the aim is to discover underpinning patterns in the dataset without labels or any a-priori known information, then learning is termed as "unsupervised". Unsupervised learning, otherwise known as knowledge discovery, usually includes grouping data according to similar characteristics (i.e. clustering) or indicating a specific data distribution (i.e. density estimation) (Bishop, 2006). As to the nature of collision prediction is to state if the traffic conditions at a specific time moment could trigger a collision, the collision prediction problem is a classification one with two outputs, namely collision-prone and safe traffic as mentioned earlier. Hence, supervised learning classifiers were reviewed.

One of the most appraised classifiers in machine-learning literature (Dreiseitl and Ohno-Machado, 2002, Ben-Hur and Weston, 2010) are Support Vector Machines (SVMs). SVMs have proven to perform efficiently in collision prediction tasks (e.g. Xu et al., 2012, Yu and Abdel-Aty, 2013, Wang et al., 2013). Even though SVMs result in less over-fitting according to Dreisetl and Ohno-Machado (2002), the incorporation of the "black-box" effect is prominent. That is due to the fact that

SVMs aim at aiding decision-making thus providing an indication of which option is correct or not. For example, in the case of collision-prediction, the decision to be made is if the current traffic conditions reflect collision-prone conditions or not. However, a number of significant and practical disadvantages are also identified in the literature regarding SVMs (Tipping, 2001, Bishop, 2006):

1) The number of Support Vectors (SVs) usually grows linearly with the size of the training set, and the use of basis functions[10] is considered rather liberal;
2) Predictions are not probabilistic, and classification problems which require the estimation of posterior probabilities for each class membership are sometimes intractable;
3) SVMs require a cross-validation procedure which can lead to misuse of data and computational time;
4) The kernel function must satisfy the Mercer's condition (i.e. it must be a continuous symmetric kernel of a positive integral operator).

Relevance Vector Machines (RVMs) are a sparse Bayesian supervised machine learning algorithm which resembles SVM characteristics (Bishop, 2006). The output of RVMs is a posterior probability and not only a suggestion of the preferred class of a specific data point. The need for probabilistic machine learning prediction has been declared by authors such as Murphy, (2012), however, it has not been widely implemented for NLCP. RVMs have been applied in many different areas of pattern recognition and classification including channel equalisation (Chen et al., 2001), feature selection (Carin and Dobeck, 2003), hyperspectral image classification (Demir and Ertürk, 2007), as well as biomedical applications (Wei et al., 2005, Phillips et al., 2011).

Gaussian Processes (GPs) belong to the same group of kernel methods which can provide probabilistic predictions According to Rasmussen (2006), GPs form a Bayesian framework for regression and classification and perform similarly to SVMs. Moreover, the power and efficiency of GPs in binary classification is further justified in Williams and Barber, (1998) where it is concluded that SVMs and GPs predict in

[10] In SVMs and RVMs, a basis function is defined for each of the data points using a kernel. More explanation is given in the following section, which describes the RVM algorithm.

the same fashion although GPs usually require more computational power. Hence GPs will also be tested for comparison reasons with RVMs and SVMs.

4.4.1. *RVMs description*

In this study, RVMs are going to be utilised in comparison with SVMs in order to improve interpretability of NLCP models. Both techniques belong to the greater group of supervised learning algorithms as well as kernel methods.

In supervised learning, there exists a set of example input vectors $\{\boldsymbol{x}_n\}_{n=1}^{N}$ along with corresponding targets $\{\boldsymbol{t}_n\}_{n=1}^{N}$, the latter of which corresponds to class labels. In this study, the two classes are defined as dangerous when t=1 and safe when t=0. The purpose of learning is to acquire a model of how the targets rely on the inputs and use this model to classify or accurately predict future and previously unseen values of $\boldsymbol{x}$.

For SVMs and RVMs these classifications or predictions are based on functions of the form:

$$\mathrm{y} = f(x; w) = \sum_{i=1}^{N} w_i K(x, x_i) + w_0 = w^T \varphi(\mathrm{x}) \qquad (4.17)$$

where $K(x, x_i)$ is a kernel function, which defines a basis function for each data point in the training set, w_i are the weights (or adjustable parameters) for each point, and w_0 is the constant parameter. The output of the function is a sum of M basis functions ($\varphi(\mathrm{x}) = [\varphi_1(\mathrm{x}), \varphi_2(\mathrm{x}), \dots, \varphi_M(\mathrm{x})]$) which is linearly weighted by the parameters w.

SVM, through its target function, tries to find a separating hyperplane to minimise the error of misclassification while simultaneously maximising the distance between the two classes (Yu and Abdel-Aty, 2013b). The produced model is sparse and relies only on the kernel functions associated with the training data points which lie either on the margin or on the wrong side. These data points are referred to as "Support Vectors" (SVs).

RVMs use a similar target function as in Equation 4.17, but introduce a prior distribution over the model weights, which are governed by a set of hyperparameters. Every weight has a corresponding hyperparameter and the most probable values of

those are estimated from the training data during each iteration. Finally, RVMs, in most cases, use fewer kernel functions compared to SVMs, without compromising the performance.

In the binary classification problem ($\{\boldsymbol{t}_n\}_{n=1}^{N} = \{0,1\}$), a Bernoulli distribution is adopted for the prior distribution $\mathrm{p(t|\mathbf{x})}$. The logistic sigmoid function $\sigma(y) = \frac{1}{1+e^{-y}}$ is applied to y(x) so as to combine random and systematic components. This leads to a generalised linear model such that:

$$f(x; w) = \sigma\big(w^T \varphi(\mathrm{x})\big) = \frac{1}{1+e^{-w^T \varphi(\mathrm{x})}} \tag{4.18}$$

It should be noted here that there is no constant weight (e.g. noise variance). By making use of the Bernoulli distribution, the likelihood of the training data set is defined as:

$$\mathrm{p(t|x, w)} = \prod_{n=1}^{N} \sigma\big(w^T \varphi(\mathrm{x}_n)\big)^{t_n} (1 - \sigma(w^T \varphi(\mathrm{x}_n)))^{1-t_n} \tag{4.19}$$

Using a Laplace approximation to calculate the weight parameters and for a fixed value of hyperparameters (α), the mode of the posterior distribution over w is obtained by maximizing:

$$\log\big(\mathrm{p(w|x, t, \alpha)} = \log\big(\mathrm{p(t|x, w)\ p(w|\alpha)}\big) - \log\big(\mathrm{p(t|x, \alpha)}\big) =$$

$$\sum_{n=1}^{N} (\,\mathrm{t}_n \log f(x_n; w) + (1 - \mathrm{t}_n) \log(1 - f(x_n; w))\big) - \frac{1}{2} w^T A w + c \tag{4.20}$$

where A=diag($\alpha_o\ \alpha_1\ ,\ldots,\ \alpha_N$) and c is a constant.

The mode and variance of the Laplace approximation for w are:

$$\mathrm{w}_{MP} = \Sigma_{MP} \Phi^{\mathrm{t}} \mathrm{Bt} \text{ and } \Sigma_{MP} = (\Phi^{\mathrm{t}} \mathrm{B} \Phi + A\,)^{-1} \tag{4.21}$$

where B is an NxN diagonal matrix with:

$$\beta_{nn} = f(x_n; w)(1 - f(x_n; w) \tag{4.22}$$

$$\mathrm{p(t|x, \alpha)} = \int \mathrm{p(t|x, w)\ p(w|\alpha) dw} = \mathrm{p(t|x, w}_{MP}\mathrm{)p(w}_{MP}\mathrm{|\alpha)}(2\pi)^{M/2} |\Sigma_{MP}|^{1/2} \tag{4.23}$$

By maximising the previous equation, with respect to each α_{i}, the update rules are obtained as shown below:

$$\alpha_i^{new} = \frac{1-\alpha_i \Sigma_{ii}}{m_i^2} \quad (4.24)$$

$$(\beta^{\text{new}})^{-1} = \frac{\|t-\Phi m\|^2}{N-\sum_{i=1}^{N}(1-\alpha_i \Sigma_{ii})} \quad (4.25)$$

where m_i is the i-th element of the estimated posterior weight w and Σ_{ii} is the i-th diagonal element of the estimated posterior covariance matrix Σ_{MP}.

4.4.2. *GPs brief description*

Similar to RVMs, GPs model the posterior probability of the target variable (which in this thesis are collisions) for every new input vector given a set of training data.

Firstly, consider a target variable $t \in \{0,1\}$. A GP aims to estimate the probability $\pi(\text{x}) = \sigma\big(\text{y}(\text{x})\big)$, where $\sigma(\text{y}) = \frac{1}{1+e^{-t}}$, x are the predictors and y is the response in the classification task. The probability $\pi(\text{x})$ denotes the probability that an input x belongs to class 1. If we consider that $\text{y}(\text{x}) = w^T x + b$, where b denotes a bias, using a GP allows y(x) to be non-linear. The required predictive distribution for new data is given by:

$$p(t_{n+1}|t_n) = \int p(t_{n+1} = 1|a_{n+1})\, p(a_{n+1}|t_n)\; da_{n+1} \quad (4.26)$$

where $p(t_{n+1} = 1|a_{n+1}) = \sigma(a_{n+1})$.

As the integral is analytically intractable usually a Laplace approximation is commonly used to estimate the posterior distribution(Bishop, 2006, Rasmussen, 2006).

4.5. Improving performance of machine-learning classifiers for real-time NLCP (objective 5)

One of the primary limitations of NLCP models as indicated in Chapter 3 is the imbalance of the datasets used in NLCP modelling where safe traffic condition cases are over-illustrated against collision-prone conditions due to the rarity of collision events. This subsection will discuss the methods used to improve the performance of real-time NLCP classifiers.

To begin with, consider a training dataset $X_{training} = \{(x_n, y_n), n = 1, \dots N$ being available where x_n is a predictor variable and $y_n = \{0,1\}$ is a response. A binary classification problem is the one attempting to build a function f which, given new

data instances will assign them to the correct class. Moreover, the classification performance of every classifier is initially assessed through the confusion matrix as seen in Table 4.2. In a confusion matrix, the predictions of each data instance are contrasted with the original class to which they belonged, so as to ascertain whether they are correctly classified. In the NLCP task, the binary classification problem is concerned with the identification of collision-prone traffic, hence Class 1 in Table 1 represents "*collision-prone*" traffic and Class 0 represents "*safe"* traffic.

Table 4. 2: A confusion matrix example

	Predicted Class	
Actual (True) Class	0	1
0	True Negative (TN)	False Positive (FP)
1	False Negative (FN)	True Positive (TP)

A similar performance metric is the area under the Receiver Operating Characteristics (ROC) curve, which plots the true positive rate against the true negative rate.

Based on the confusion matrix, other widely used metrics include:

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \tag{4.27}$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN+FP}} \tag{4.28}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \tag{4.29}$$

$$\text{G-means} = \sqrt{Recall * Specificity} \tag{4.30}$$

$$\text{F-measure} = \frac{2*precision*sensitivity}{precision+sensitivity} \tag{4.31}$$

The recall statistic shows the correct classification accuracy with respect to collision-prone traffic conditions, while the specificity statistic shows the classification accuracy in terms of safe conditions. Precision is used for identifying the classification accuracy among the classes. G-means is used to ensure whether the use of an imbalance dataset (1:3; conflicts vs safe) has any negative impact on the balanced qualification accuracy. Lastly, the *F*-measure is a metric which resembles the conflict-prone classification ability of the classifier models(Sun and Sun, 2015).

Classification of imbalanced datasets is a documented problem in data mining (He and Garcia, 2009, Sun et al., 2009, López et al., 2013). The most important problem with imbalanced data is the high misclassification rate for the under-represented class, because the classifier favours the majority class. To overcome this problem proposed solutions from the literature can be grouped into three groups:

1) Data sampling
2) Algorithm alteration
3) Cost-sensitive learning

The first solution requires that the sampling of training cases should be modified to a certain extent, in order for a more balanced dataset to be produced. Next, the algorithm alterations solution relates to modifications made in learning algorithms e.g. in the kernels for kernel-based approaches such as SVMs or RVMs or in the construction of trees for tree-based approaches such as Random Trees or Random Forests (RFs). The third solution applies higher misclassification costs for instances of the minority class (i.e. for false positives) and lower misclassification costs for the majority class (i.e. for false negatives).

4.5.1. *Data Sampling*

In order to obtain a less imbalanced dataset, a low cases to controls ratio (e.g. 1:3 or 1:4) between hazardous and safe traffic is going to be investigated in this thesis. This will result in a more balanced dataset than the ones used in recent literature and will potentially help in identifying collision-prone conditions more reliably. In order to achieve this objective, He and Garcia (2009) propose random oversampling or undersampling. Random oversampling is a technique which artificially appends data in the original dataset while random undersampling is a technique that randomly selects cases from the majority class so that a more balanced dataset is acquired. However, it is suggested in He and Garcia (2009) that oversampling might lead to over-fitting. Thus, undersampling would be preferable for the purposes of this thesis. However, data cleansing in conjunction with oversampling is also suggested as a solution to address over-fitting and hence it will also be utilised in this thesis.

Reviewing the literature in undersampling and oversampling with data cleansing, it was found that Repeated Edited Nearest Neighbours (RENN) (Tomek, 1976), its integration with Synthetic Minority Oversampling TEchnique (SMOTE) (Chawla et al., 2002) and neighbourhood cleaning (Laurikkala, 2001) performed well for classes that are difficult to recognise (Batista et al., 2004).

RENN utilises the Edited Nearest Neighbour (ENN) algorithm (Wilson, 1972) repeatedly until all the instances in the dataset have a majority of their neighbours within the same class. ENN applies the kNN algorithm and removes all misclassified instances from the training dataset. In this way, the difference between classes is more obvious and a smooth decision threshold is obtained. The RENN algorithm developed by Guan et al., (2009)is briefly discussed below:

- If D_e is the dataset acquired from the ENN algorithm and D_o is the original dataset repeat:
    - At every iteration i for each instance x_i in D_e discard x_i if it is misclassified using kNN
- Until ${D_e}^i = {D_e}^{i-1}$ where ${D_e}^i$ is the edited dataset in iteration i and ${D_e}^{i-1}$ is the edited dataset in Iteration i-1.

SMOTE integrated with ENN aims at producing well-defined class clusters which can potentially improve classification results. After artificially generating instances of the minority class through SMOTE, ENN is implemented to conduct the data cleaning in depth and removes data instances from both classes when the three nearest neighbours of a data instance are misclassified (Batista et al., 2004). This is beneficial, especially for datasets with a small number of instances in the positive class, for instance collision-prone traffic, in datasets containing collision data which are rare events.

Neighbourhood cleaning rule splits the dataset according to the class of interest. If there are noisy data exist within the dataset, they are identified using ENN. After the identification of noisy data for every data instance belonging to the minority class (e.g. x_{min}) its 3-nearest neighbours are tested. If these three neighbours misclassify x_{min} , then any neighbours of x_{min} belonging to the majority class are removed.

4.5.2. *Algorithmic treatment*

In addition to data sampling approaches, ensemble learning has been argued to work perform well with imbalanced datasets (Galar et al., 2012). This is further justified in Sun et al. (2009) and López et al. (2013), where it is stated that ensemble classifiers form a potential solution to the class imbalance problem. Therefore, an ensemble-based classifier such as RFs is going to be tested for the first time in collision prediction studies at the network-level.

RF has mainly been applied in the area of NLCP for variable selection purposes. Its purpose within NLCP was to select the most important variables to be used in the subsequent modelling. Abdel-Aty et al. (2008) initially combined RF for variable selection with NNs and suggested that the resulting classifiers can efficiently differentiate collision-prone traffic conditions. Moreover, improved classification results were also demonstrated when RF was combined with logistic regression (Hassan and Abdel-Aty, 2013) and genetic programming (Xu et al., 2013b), in order to identify important variables to be used in real-time collision models. To the author's knowledge, however, there is no study employing RF for distinguishing between collision-prone and safe traffic conditions.

4.5.2.1. RFs Description,

In order to understand how RFs function, a brief description of the algorithm is given in this section.

A base classifier or weak learner is a predicting function $f(x, X_{training})$ which performs slightly better than random chance. Boosting is the combination of base classifiers in order to acquire a committee of classifiers which outperforms any of the base classifiers (Rokach, 2010). In addition, Bagging is the technique of using bootstrap datasets to assess the performance of a classifier. A bootstrap dataset X_B is a dataset created by randomly choosing n points from $X_{training}$ such that points in X_B may or may not co-exist in $X_{training}$. In bagging separate training takes place for every bootstrap dataset and a "committee" gathers the training results into a unanimous prediction (Breiman, 1996). Both techniques are used in ensemble learning which, as stated before, describes the procedure of constructing a predictive

model by consolidating multiple individual ones so as to improve predictability (Rokach, 2010).

RFs are an ensemble classifier using Classification and Regression Trees (CART) as a base classifier. In particular, CART is a nonparametric method, used for classification and regression purposes which divides the data space into smaller subspaces in order to obtain concise blocks of limited size, which are descriptive of one dominant class (Hossain, 2011). CART operates based on recursive partitioning and is described by the following sub-tasks which are usual in constructing tree-based algorithms:

1. The best split for each predictor is found. The best split indicates the value of the predictor that leads to the biggest separation on the response variable
2. Start with one predictor and divide into two "sub-groups" according to the splitting threshold. Divide each of the sub-groups into two subgroups according to another predictor and its splitting threshold
3. Repeat for all the predictors until a finishing threshold is reached

If the size of the resulting tree is larger than the requested ones, the tree is pruned and the best sub-tree is chosen so that it can act as a classifier for new data.

RFs combine CARTs in such a way that each tree grows dependent on values from an independently sampled random vector. Its performance improves on CART regarding stability and incorporation of correlated predictors.

RF use the bagging algorithm in conjunction with the random subspace method proposed by Ho (Ho, 1998). Each tree is built using the impurity Gini index (Breiman, 2001). Nevertheless, only a random subset of the input features is used for the construction of the tree and no pruning occurs. For each new training dataset, one-third of the samples is randomly neglected and forms the out-of-bag (OOB) samples. Next, the samples that are not neglected are used for building the tree. For every constructed tree, the OOB samples are used as a validation dataset and the misclassification OOB error is estimated. When a new data record needs to be classified, it is run through all the constructed trees and a classification result for every tree is obtained. Following this, the majority vote over all the classification

results from all the constructed trees is chosen as the classified label for that specific data record (Verikas et al. 2011). However, an appropriate value for the number of features used for splitting a node of a tree needs to be tuned by the user in order for the OOB misclassification error to be as low as possible (Verikas et al., 2011).

To elaborate more on the steps followed for the construction of an RF:

1. For the training dataset $X_{training} = \{(x_n, y_n), n = 1, \dots N$ as described before, let X_b be the b-th bootstrap sample which is constructed through random selection from n samples out of $X_{training}$. The rest of the data (i.e. $X_{training} - X_b$) are considered the OOB sample
2. For each tree T_b, m number of predictors are randomly selected at every node and the one that results in maximum two pure nodes is used for splitting. A pure node is one that contains data belonging to the same class. Thus, a pure node does not require further splitting.
3. In order to make predictions new data are ran down through every grown tree. The predicted class is the class of the leaf where the data instance ended up.
4. During every iteration, the OOB sample is ran down T_b and the class of every data instance is obtained through majority voting and the error rate is calculated for every tree. The aggregation of the misclassification errors defines the OOB error rate.

4.5.3. *Cost-effective classification*

In order to take into consideration all the options available for disaggregated traffic data classification, the assignment of weights to misclassifications was also preliminary tested but did not yield sufficient result and therefore was discarded. However, for completion, cost-effective classification will be described in this section.

An integral part of cost-sensitive learning is the cost matrix which is actually a numerical representation of the penalties given if a data instance belonging to one class is classified to the other (i.e. in the binary classification problem). To elaborate more, let $C(min, maj)$ denote the cost of misclassifying a majority class data instance as one of the minority class and $C(maj, min)$ indicate the cost of misclassifying a minority class data instance as of one belonging to the majority class.

A typical example of a cost matrix is given in Table 4.3. Usually in cost-sensitive learning, the cost of the misclassifying minority class data instances is significantly higher than its majority class counterpart (He and Garcia, 2009). According to He and Garcia, (2009) in order to find the optimal ratio for misclassifications (i.e. the relationship between $C(min, maj)$ and $C(maj, min)$), an initial approach is to apply the costs on the data and select the best training distribution for the classifier. Other solutions include the combination of meta-techniques[11] and ensemble classifiers as well as the incorporation of cost-sensitive functions into the classification example so as to "adapt" cost-sensitive principles to the classifiers.

Table 4. 3 A cost matrix example

	Predicted Class	
True Class	**Majority class**	**Minority class**
Majority class	0	$C(min, maj)$
Minority class	$C(maj, min)$	0

4.6.Addressing misreported collision time and traffic data aggregation (Objective 3)

The problem associated with erroneous collision time reporting as well as the underreporting issue of less serious collisions are attempted to be solved through the use of traffic microsimulation. More specifically, in microsimulation, traffic characteristics and vehicle kinematics are explicitly documented for every time moment, and can be linked to the time of traffic conflicts which is also accurately recorded as described in Chapter 3 (Section 3.5.3). Therefore, it is assumed that the problem of erroneous collision time is addressed by the temporal precision of the recorded conflicts while underreporting is not a matter of concern because the extraction of conflicts is the outcome of a computer software and does not relate to the severity of the conflict.

Archer (2005) reviewed the microsimulation platforms of VISSIM (PTV Planug Trasport Verker AG, 2013), HUTSIM (Kosonen, 1996) and PARAMICS (SYSTRA Limited, 2009) and concluded that VISSIM is the most appropriate option for

[11] Metatechniques: Metalearning techniques, i.e. techniques which study and learn the effect of classification procedure and the estimated results, rather than aiming at learning the underpinning patterns from data.

modelling traffic with regards to safety. Archer, also, stated that the main advantages of VISSIM were a) the high-level of detail in defining road-user behaviour parameters and the underpinning models (i.e. car-following, gap-acceptance, lane-changing), b) the feasibility of defining time-specific traffic inputs and detailed rules for the interaction of different traffic participants (e.g. lane changing or car-following rules), c) the high-level of detail regarding the simulation output and the simulation resolution which is $1Hz$ and d) the capability of using an Application Programmer Interface (API) for enhanced functionality. In the report of Gettman and Head (2003), VISSIM is also praised for the high-level representation of vehicles' motion state and vehicles' interaction rules. In the same report, in the summary of the comparison between several traffic microsimulation software, it is also affirmed that VISSIM supports most of the necessary features to obtain surrogate safety measures. This is further supported by Shahdah (2014) who employed VISSIM for traffic safety analysis and selected it due to its flexibility and easy manipulation of built-in features for representing driving behaviour. A recent comparison of traffic simulators by Saidallah et al. (2016) demonstrated that VISSIM exceeds the capabilities of other simulators because it can continuously simulate traffic, is flexible, allows easy coding of the traffic network and can accommodate Geographic Information System (GIS) maps.

The outlined advantages of VISSIM in comparison with other microsimulation software as recommended by the literature, led to its choice as the platform to be utilised for real-time NLCP.

4.6.1. *Description of VISSIM micro-simulation software*

VISSIM is a time-based microscopic traffic simulator which utilises several driver behaviour and vehicle performance sub-models to efficiently model traffic (Archer, 2005). The user needs to construct the network based on a series of aerial photographs or maps and add the necessary objects (e.g. data collection points or traffic signals) to the necessary points. Following the construction of the network, traffic flow and speed distributions of the vehicles need to be inputted. In order to input traffic flow, the user initially must define the vehicle composition (e.g. the percentage of cars, heavy goods vehicles and other traffic). With vehicle composition

known, then a desired speed distribution needs to be defined so as to imitate the speed of the vehicles in the real-world. This speed distribution is the cumulative distribution of all the vehicles on the road segment that is studied (Yu, 2013). Moreover, the behaviour of vehicles is configured according to car-following and lane changing models. A brief description of the car-following and lane-changing models employed for motorway environments is given below.

4.6.1.1.Car-following in VISSIM

For motorway environments, the Wiedemann 99 car-following model is indicated as the appropriate one by the VISSIM manual. In the Wiedemann psycho-physical car-following model, four driving states or regimes are considered: a) un-influenced driving, b) closing process, c) following process and d) emergency braking. In un-influenced driving, a vehicle is attempting to reach its desired speed if there is no lead vehicle within 150m. If the longitudinal distance between a leading and a following vehicle is less than 150m and the longitudinal speed difference is greater than a "safe" threshold, the following vehicle enters the "closing" phase. During the closing phase, the driver of the following vehicle realises an approach towards a slower vehicle and begins to decelerate in order to reach a desired following distance safely separating the two vehicles. In the following scenario, the following vehicle attempts to retain the same speed of the lead vehicle without reacting to the motion of the leading vehicle while in the emergency braking regime the drivers react in order to avoid an imminent collision.

In VISSIM, three parameters need to be specified correctly to calibrate the car-following model (PTV Planug Trasport Verker AG, 2013). These parameters include:

- The standstill distance (in ft or m): the average desired distance between two vehicles (i.e. the distance between the front bumper of the rear vehicle to the rear bumper of the leading vehicle), as seen in Figure 4.6

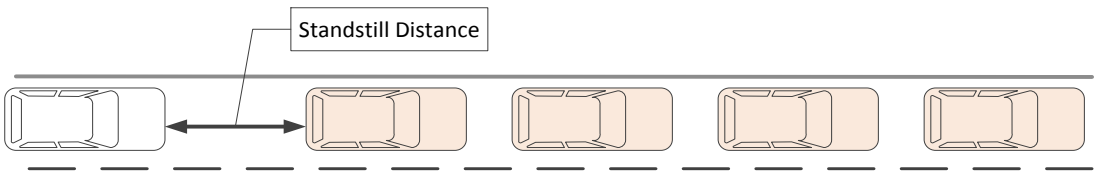


Figure 4. 6 Standstill Distance illustration

- The following distance or time headway (in seconds): the distance in seconds which a vehicle needs to maintain when having a certain speed (see Figure 4.7). A higher headway is an indicator of a more cautious driver.

  If the following and the standstill distance are given, then the safety distance can be calculated as:

  $$d_{safe} = d_{standstill} + t_{headway} * v \quad (4.32)$$

  where d_{safe} denotes the safety distance, $d_{standstill}$ is the standstill distance, $t_{headway}$is the following distance and v is the speed of the vehicle

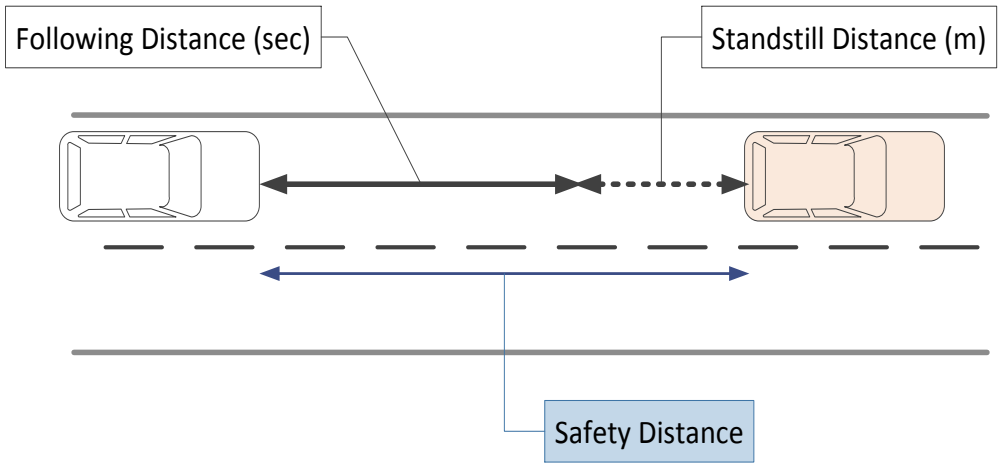


Figure 4. 7 Following Distance and Safety Distance illustration

- The following variation: the longitudinal fluctuation during a car-following scenario (e.g. a scenario where a vehicle follows the leading one and attempts to maintain a similar speed as well as a safe distance). More specifically, the following variation indicates the extra distance that a driver is willing to provide before moving within the safety distance area. The concept is illustrated in Figure 4.8.

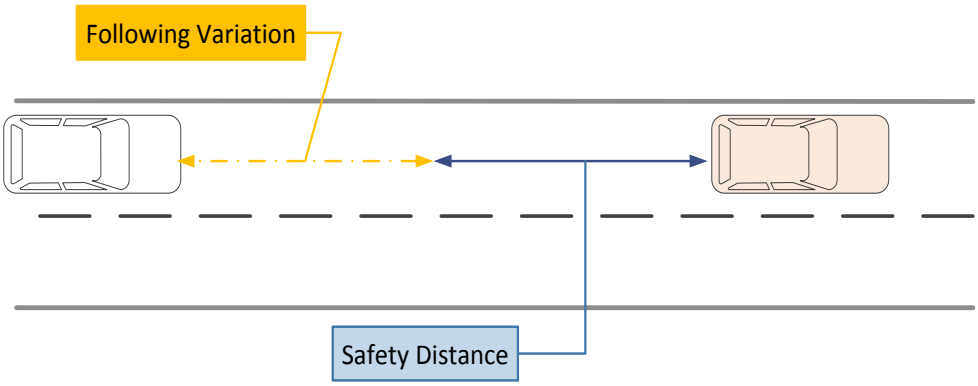


Figure 4. 8 Following Variation illustration

4.6.1.2.Lane changing in VISSIM

Regarding lane changing, VISSIM differentiates between two scenarios, namely necessary lane changing and free lane changing. Necessary lane changing refers to scenarios where one vehicle needs to reach a connecting road to fulfil its route, while free lane changing occurs when more space and higher speeds are at present. For free lane changing VISSIM investigates if the distance between the lane-changing vehicle and a vehicle in the destination lane is sufficient or not.

To model lane changing more realistically, VISSIM offers the cooperative lane changing option as depicted in Figure 4.9. Let a vehicle A driving on lane l_A understand that a preceding vehicle B driving on lane l_B wants to change lanes to get to l_A. If vehicle A changes lanes to l_B in order to stimulate the initial lane changing of vehicle B, then the scenario is termed as cooperative lane changing. In VISSIM the options are free lane selection (i.e. vehicles can overtake on any lane) or right/left-side rule lane changing. For this thesis, the right-side rule was chosen as this is the practice in UK motorways.

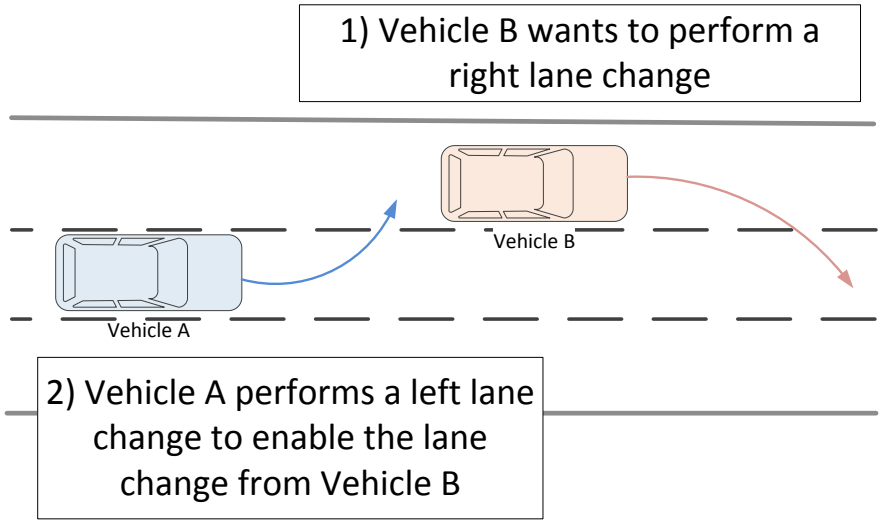


Figure 4. 9 Cooperative lane changing in VISSIM

After defining the driving behaviour, the vehicle composition and the vehicles' speed distribution the simulation needs to be run and validated. The following section will describe how the simulation is validated.

4.6.2. *Validation of the simulation*

According to Barcelo (2011), a methodological flow chart to validate a traffic simulation can be depicted in Figure 4.10.

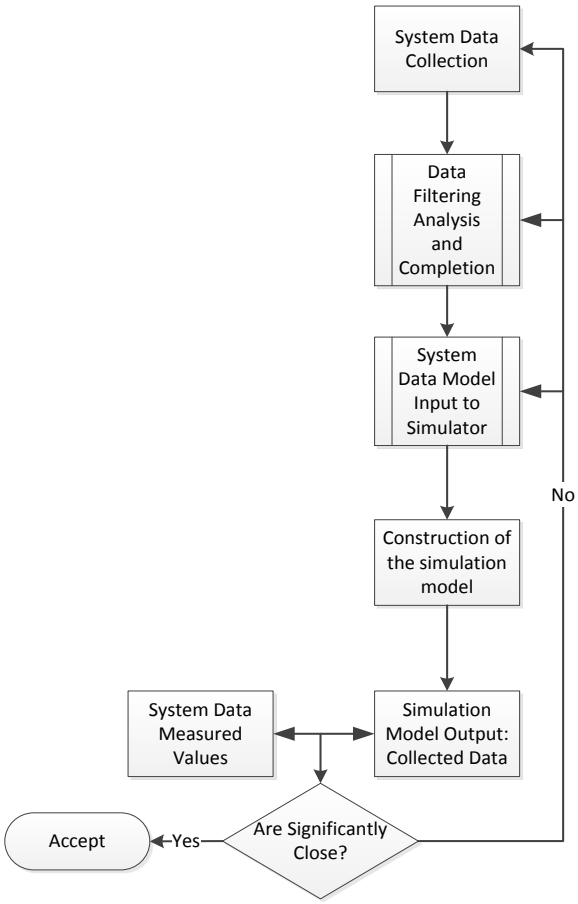


Figure 4. 10 Methodological approach to validating traffic microsimulation

The crucial question of the flowchart presented in Figure 4.10 is the one concerned with the proximity of the measured "real-world" data and the simulated ones. To quantify this proximity, traffic agencies (e.g. FHWA - Dowling et al., 2004, Transport For London, 2010) have proposed several metrics regarding the simulated traffic volumes, travel times or speeds of the vehicles. More specifically, the Wisconsin Department of Transportation (WDOT) (Dowling et al., 2004) has proposed using the GEH Statistic(Transport For London, 2010) and the travel times to account for the validity of the model.

The GEH-statistic is given by the formula (Dowling et al., 2004):

$$GEH = \sqrt{\frac{(SV-TV)^2}{\frac{(SV+TV)}{2}}} \tag{4.33}$$

where *SV* is the estimated volume from the simulated model and *TV* is the real-world traffic volume. According to the Wisconsin Department of Transportation (WDOT, 2014) if the value of the GEH statistic is less than 5 for at least 85% of the time for

all individual link flows, then it is assumed that the simulated model is well calibrated. If the GEH-statistic is between 5 and 10 it is assumed that there is a possible model error or bad data, whereas if GEH is greater than 10 there is a high probability of modelling error or bad data.

Regarding link travel times, if the difference between the simulated link travel time and the real-world travel time is within 15% then the model is considered calibrated satisfactorily.

As a result, the two criteria used for validating the simulation results in this thesis are going to be the GEH-statistic for the traffic volume and the difference between the simulated and observed link travel times. According to WDOT, (2014), during the simulation, the queuing patterns should be realistic, there should be no bottleneck in free flow conditions, and the freeway lane choices should be consistent with real-world observations.

After the simulation and validation has been completed, the next step should be the extraction of conflicts from the simulated traffic data. The literature review revealed that simulated traffic data can be filtered through SSAM. As mentioned in Chapter 3 (section 3.5.2.) SSAM is the only post-processing software which investigates simulated vehicle trajectories and detects the number and severity of traffic conflicts, accompanied by surrogate safety measures for each conflict.

4.6.3. *SSAM description*

After the completion of a simulation session, VISSIM outputs a number of files regarding the simulation results. These include files describing the individual (raw) and aggregated traffic data collection measurements, the travel time measurements and the vehicle trajectories file among others. Simultaneously, SSAM utilises the vehicle trajectories file to output conflicts.

Using thresholds on TTC[12] and PET[13], SSAM filters the trajectories to detect conflict events. Furthermore, analysing the angle between the vehicles, SSAM can classify if the conflict is a rear end, crossing or lane-changing one. As this thesis studies motorway environments, only rear-end and lane-changing conflicts were analysed. For each conflict event the exact time of the event as well as the speeds, and the accelerations/decelerations of the vehicles are documented. Figure 4.11 illustrates the timeline of a conflict between two vehicles (A and B) as well as the major variables used and outputted.

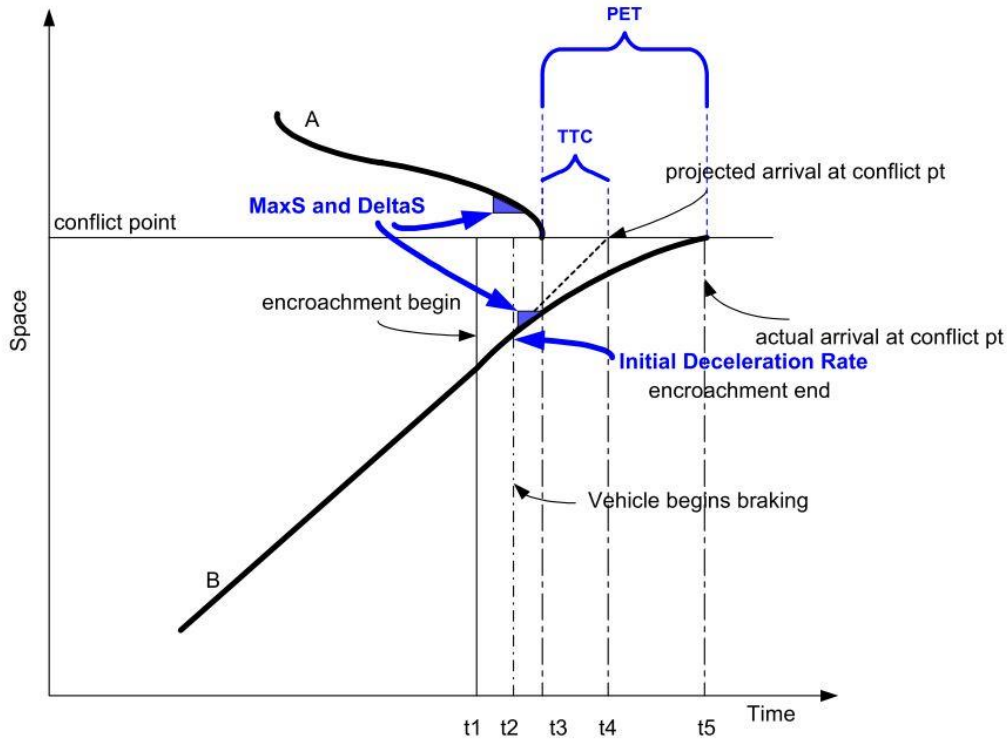


Figure 4. 11 Timeline of a conflict event in SSAM

(from Gettman and Head, 2003)

[12] TTC is a proximal safety indicator showing the remaining time until a potential collision between two vehicles if the collision course and speed difference remain unchanged(Hayward, 1972)

[13] PET is used to measure the temporal difference between two road users over a common point or area. It does not require that the vehicles are on a collision course but does require transversal trajectories (Archer, 2005)

In Figure 4.11, t1 is the time when vehicle A enters the encroachment area (i.e. starts its turn to the left), while t2 is the time vehicle B realises a potential collision and begins braking. Next, t3 is the time the corner of the rear bumper of the crossing vehicle leaves the encroachment point and t4 is the time the vehicle B was projected to arrive at the conflict point if it kept its speed and trajectory constant while t5 is the time vehicle B arrives at the conflict point. Additionally, maxS is the maximum speed of either vehicle during the conflict event and DeltaS is the difference in vehicle speeds at the time of the minimum TTC (Gettman and Head, 2003, Pu and Joshi, 2008).

After the identification of conflict events from SSAM, knowing the exact time of the conflict as well as the vehicles that were involved, the traffic conditions before the incident can be obtained using the raw traffic data measurements from VISSIM. Figure 4.12 provides a flow chart of the procedure which can lead to the identification of pre-conflict conditions. If several simulations runs are performed, then one run in conjunction with its conflict data could set the example from where pre-conflict traffic can be exported, while the other runs would be the normal traffic paradigms. Figure 4.13 provides a visual insight into the procedure.

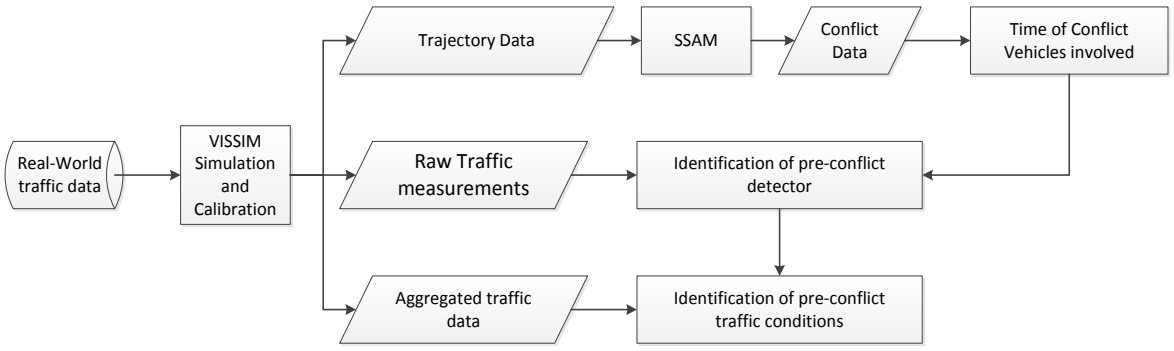


Figure 4. 12 Flowchart for identifying pre-conflict traffic conditions from VISSIM and SSAM

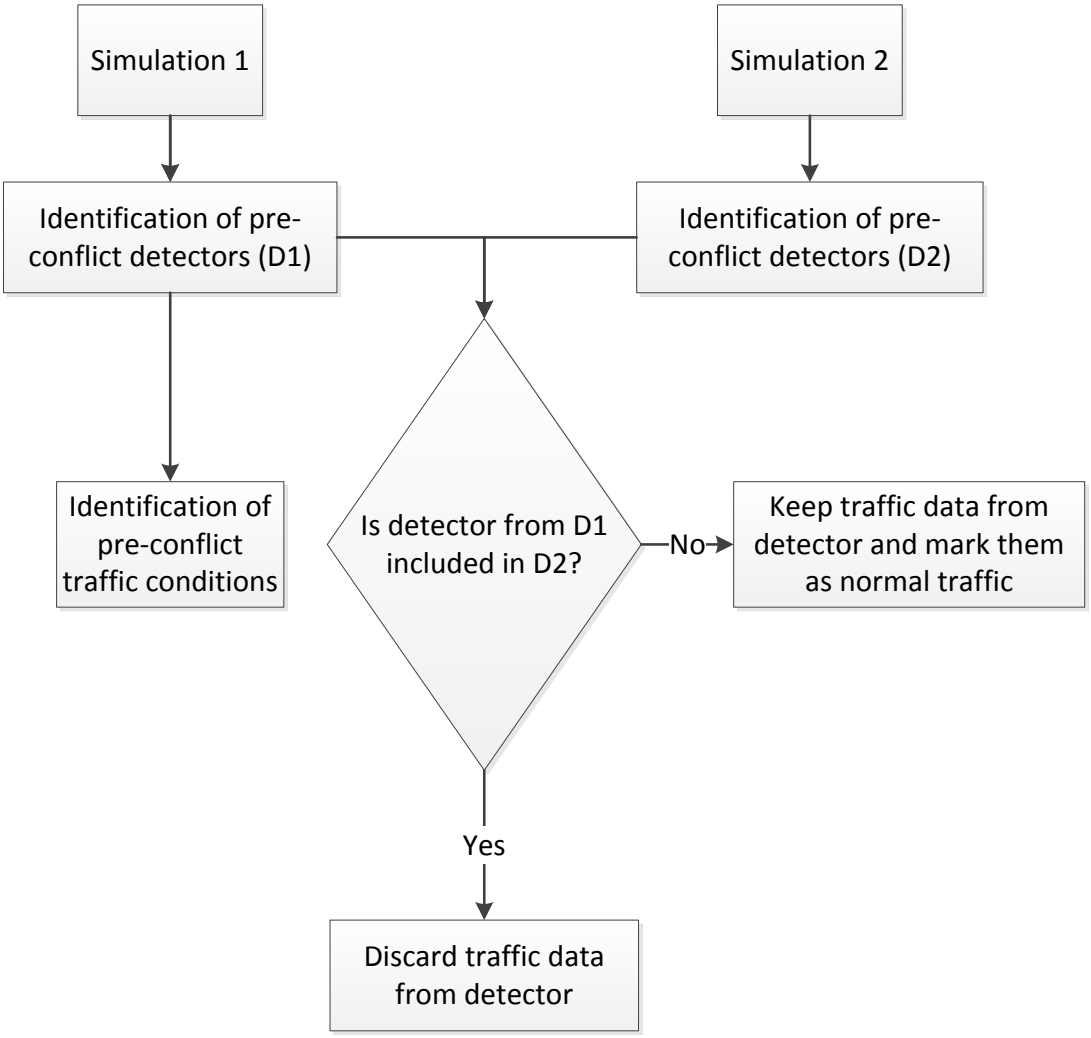


Figure 4. 13 Example flowchart for creating pre-conflict and normal traffic datasets

After the identification of conflict events and their matching with representative traffic conditions, machine learning classifiers can be trained to detect conflict-prone conditions in real-time.

4.7.Summary

This chapter provided a discussion of the methodology to be followed in this work. Following the research design, a method to incorporate NLCP in current autonomous vehicles' risk assessment modules was introduced. In order for NLCP models to be incorporated in such modules, however, problems with the performance and interpretability of machine learning classifiers as well as the erroneous collision time reporting should be addressed.

The theoretical model is based on interaction-aware models which were found to be the state-of-the-art in risk assessment for autonomous vehicles. The model is based on a Dynamic Bayesian Network, a graphical probabilistic model, and incorporated layers describing the network-level and vehicle-level risk as well as the vehicles' motion characteristics and sensor-measurements. The description of each variable as well as the estimation of each variable's probability was explained in depth in section 4.3.

To improve the interpretability of current machine-learning classifiers used for NLCP, RVMs a Bayesian counterpart of the popular SVM algorithm, were tested for their performance in NLCP. Furthermore, to address the problem of imbalanced collision datasets, where safe traffic conditions form the vast majority of the dataset and collision-prone traffic is under-presented, this thesis explored two solutions inspired by imbalanced learning literature. These solutions include a) the construction of a more balanced dataset using RENN, NC and SMOTE-ENN and b) the utilization of RFs which are an ensemble classifier and potentially perform better when classifying imbalanced datasets.

Finally, to overcome the problem of existing collision databases, which include erroneous collision time and underreporting of less serious collisions, simulation data obtained from the traffic microsimulation software VISSIM are going to be utilised. The traffic data from VISSIM are going to be used along with SSAM, a post-processing software, which outputs traffic conflicts. The methods to obtain pre-conflict and normal traffic conditions through VISSIM and SSAM were also described in section 4.6.

In conclusion, Figure 4.13 presents a flowchart of the overall methodology followed in this thesis.

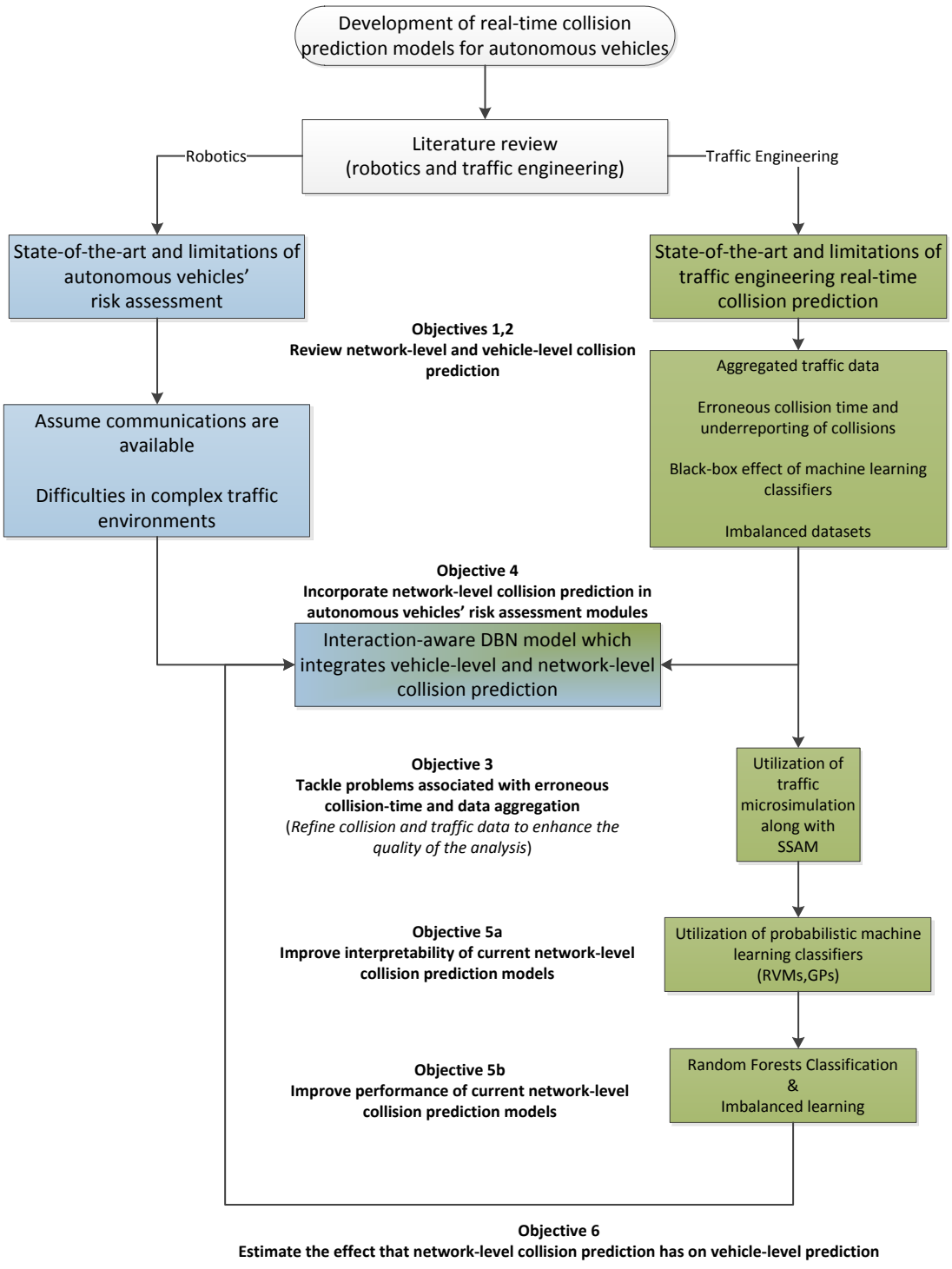


Figure 4. 14 Flowchart of the methodology followed in the present thesis

5. Data description and Pre-Processing

5.1.Introduction

In safety-critical applications such as NLCPs and collision risk assessment for autonomous vehicles, the quality and availability of data are crucial. In order to implement all the classifiers and assess the impact on AV risk estimation, traffic data and the corresponding collision data as well as vehicle-level data need to be collected, processed and analysed.

This chapter describes the features and limitations of the datasets which were employed in the analysis. Due to the fact that simulated traffic data will also be utilised in this thesis, a part of this chapter will be dedicated to these simulated data and their validation.

5.2.Network-level data description

This section will describe the network-level data utilised in this PhD thesis. The data include:

- 15-minute UK traffic and the corresponding collision data (Dataset 1)
- 5-minute traffic and corresponding collision data from Athens, Greece (Dataset 2)
- 30-second, 1-minute,3-minute,5-minute simulated traffic and the corresponding conflict data based on a section of the M62 UK motorway (Dataset 3)

5.2.1. *UK traffic and collision data*

The first dataset which was utilised in this work derives from the Strategic Road Network of England (SRN). The SRN consists of all the motorways and major A-roads as depicted in Figure 5.1. Since 2006, the governing unit of SRN (i.e. Highways England formerly known as Highways Agency) has been operating the Smart motorways programme (Highways Agency , 2014). The scheme of smart motorways is the state-of-the-art in traffic management and aims at increasing capacity, addressing the issue of congestion and increasing safety. These aims resemble the aspirations of autonomous vehicles which could potentially enhance traffic flow and road safety. Hence, traffic data from smart motorways along with the

corresponding collision data would provide detailed insight into the pre-collision conditions on a continuously monitored environment. To keep up with the state-of-the-art on network-level data and since the aim of the study is the incorporation of network-level information in AVs it was decided to choose segments from smart motorways for the analysis.

5.2.1.1.UK traffic data

In this thesis, SRN traffic data are obtained through the HATRIS (Highways Agency Traffic Information System, which is the base network of the TRAffic flow Data System (TRADS) and the Journey Time Database (JTDB). Traffic data for this study were obtained through the HATRIS JTDB query tool. Traffic data were extracted for 2012 and 2013 (01/01/2012 – 31/12/2013) for junctions J10-J13 of the M1 motorway as well as segments AL634 and AL2291 of the A3 and A12 roads respectively. This dataset was later enhanced with traffic data from junctions J25-J30 from M62 during the time period from 01/01/2012 to 31/12/2014. JTDB provides traffic data for 15-minute intervals aggregated for all lanes of the road segment. More specifically, the average travel time, travel speed and total flow are provided for every requested link, date and time period. Moreover, information is given on the day type, such as weekday, weekend day, school or bank holiday.

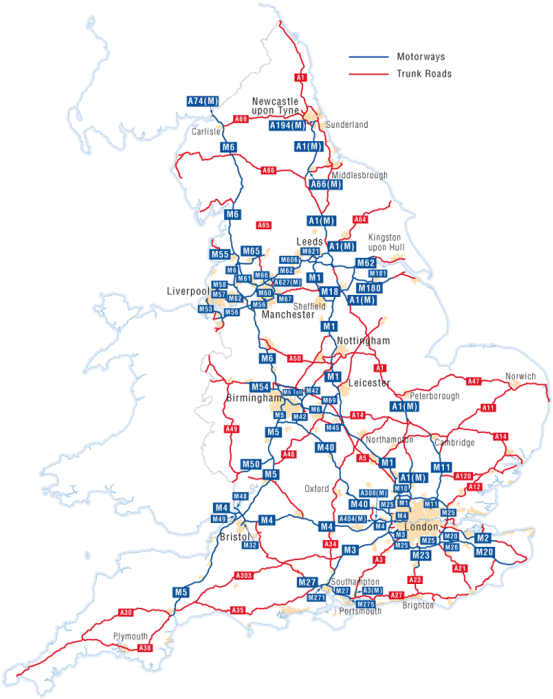


Figure 5. 1 Map of the Strategic Road Network (SRN) of England (Highways England, 2017)

5.2.1.2.UK collision data

Data from collisions occurring on the links described in the previous sections were obtained from the national collision database STATS 19. STATS 19 include all road collisions on public highways that are reported to the police and involve human injury or death. For every collision dataset included in STATS19 the most crucial variables consist of:

- The collision reference number, which is a unique seven-digit string utilised to differentiate road collisions
- The date of the collision
- The time of the collision
- The location of the collision in terms of easting and northing
- The class of the road where the collision occurred (e.g. M for motorway or A for main single carriageway)
- The road number which corresponds to the road segment that the collision took place
- The speed limit, which corresponds to the posted speed limit on the road where the collision took place

5.2.1.3.Combining traffic and collision data

For the development and testing of machine-learning algorithms discussed in Chapter 4, traffic conditions related to non-collision cases (i.e. normal driving) and collision-prone cases need to be extracted. The number of collision and non-collision cases was derived using the following process:

15-minute aggregated traffic data (i.e. 96 unique observations per day) from 2012 – 2014 were available for the entire SRN. In order to obtain traffic conditions for each of the collision cases, traffic data associated with the two unique observations were extracted: (i) the observation that coincides with the time of the collision and (ii) the observation of the 15-minute time period before (i). These traffic conditions would represent 'collision-prone' situations. Similarly, in order to represent 'safe' traffic conditions, the JTDB measurements for the same two 15-minute intervals representing traffic conditions at one week before and after the collision, as well as two weeks before and after the collision, were extracted. For example, if a collision happened at 14:08 on the 25th of June, the traffic conditions from the 15-minute interval beginning at 14:00 and 13:45 were matched to the collision case, while traffic conditions on the 11th of June and the 18th of June (i.e. before the collision

date) and the 2nd of July and the 9th of July (following the collision date) at 14:00 and 13:45 were matched to the non-collision case if no collisions happened on these dates and times. As a result, each collision case was matched with two 15-minute intervals which indicate the traffic conditions immediately before the collision, and eight 15-minute intervals which exhibit 'safe' traffic conditions at the same time on a similar day.

In order to have one value for each of the traffic variables, (e.g. flow), a weighted average for the two 15-minute intervals was calculated using the same aggregation technique as in Imprialou et al. (2015):

$$Flow_w = \beta_1 \cdot Flow_t + \beta_2 \cdot Flow_{t-1} \tag{5.1}$$

where β_1 and β_2 are the weighting parameters that satisfy the following conditions:

$$\beta_1 = \frac{t}{T}\ ;\quad \beta_1 + \beta_2 = 1;\ \ T = 15$$

where *t* is the time difference between the reported collision time and the beginning of the corresponding 15-minute interval; $Flow_w$ is the weighted 15-minute flow, $Flow_t$ is the 15-minute flow at the interval when the collision has occurred, $Flow_{t-1}$ is the 15-minute flow at the preceding interval.

By using the matched-case control structure indicated, the influence of road geometry on the collisions is assumed to be eradicated, because each collision case is matched with variables related to the entire length of the link and not to a limited area of it (e.g. neighbouring loop detectors).

The collision data corresponding to J10-J13 of the M1 Motorway, the AL634 link of the A3 road and the AL2291 link on the A12 road are shown in Table 5.1. The explanatory variables of average speed (km/h) and total traffic flow (vehicles) were chosen.

The total collision and non-collision cases which were taken into account for the development of the models are shown in Table 5.1.

Table 5. 1 Collision and non-collision cases for each of the studied links

Road	Non-collision Cases	Collision Cases	Total
M1 (Junctions 10-13)	344	86	430
A3 (Link AL634)	96	24	120
A12 (Link AL2291)	88	22	110
M62 (Junctions 25-30)	620	155	775
Total	1148	287	1435

The scatterplots (Figures 5.2.-5.5) as well as the average and standard deviation (Tables 5.2.-5.5) for the 15-minute average traffic speed and flow among collision and non-collision cases for every road are presented next.

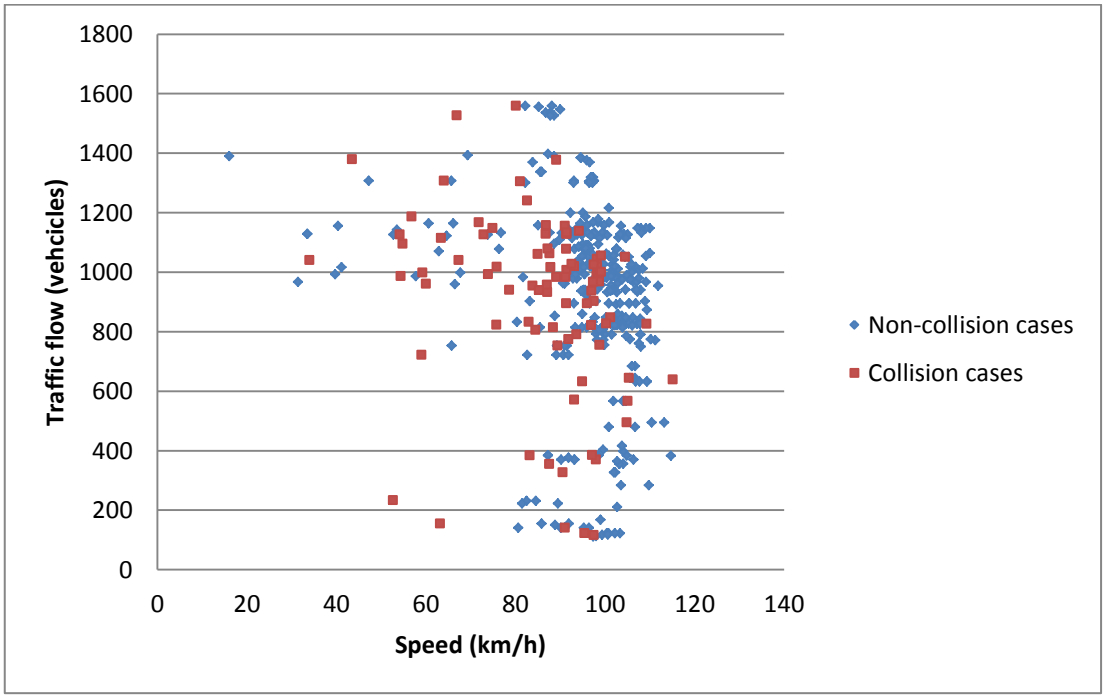


Figure 5. 2 Scatterplot of speed and flow of collision and non-collision cases for M1 (j10-j13)

Table 5. 2 Average and standard deviation of speed and flow for collision and non-collision cases for M1 (j10-j13)

Cases	Average of Speed	Average of Flow	StdDev of Speed	StdDev of Flow
Non-collision cases	82.1175	922.9011	21.7252	332.1784
Collision cases	85.2015	934.4552	35.7459	331.0074
Grand Total	82.7343	925.2119	25.1639	331.7631

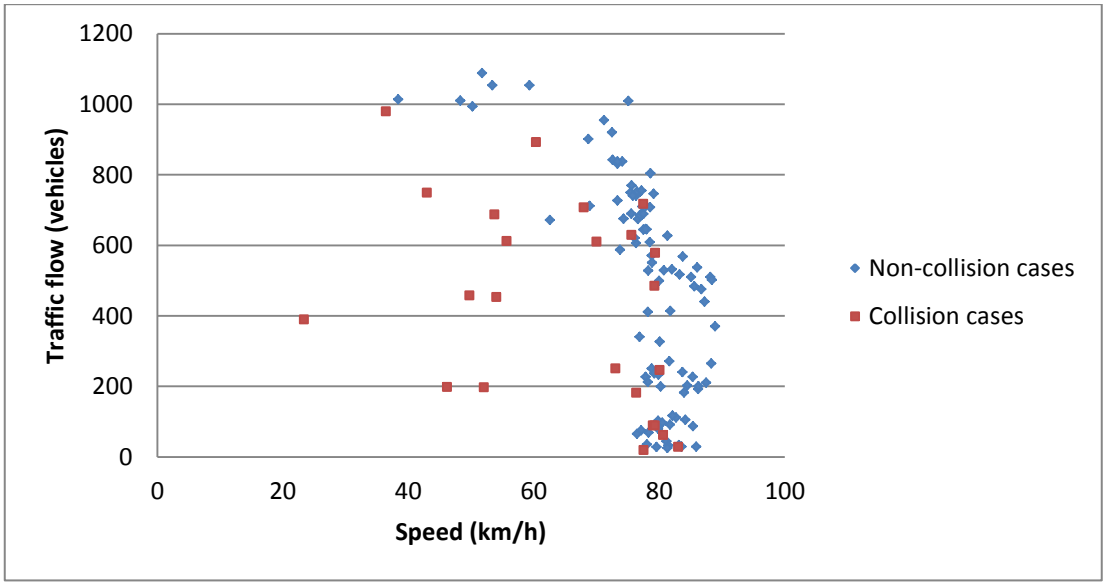


Figure 5. 3 Scatterplot of speed and flow of collision and non-collision cases for A3 (Link AL634)

Table 5. 3 Average and standard deviation of speed and flow for collision and non-collision cases for A3 (Link AL634)

Cases	Average of Speed	Average of Flow	StdDev of Speed	StdDev of Flow
Non-Collision cases	77.6636	487.1825	8.7199	307.9262
Collision Cases	64.7316	429.4403	16.6802	288.0735
Grand Total	75.0555	475.5370	11.9146	303.7270

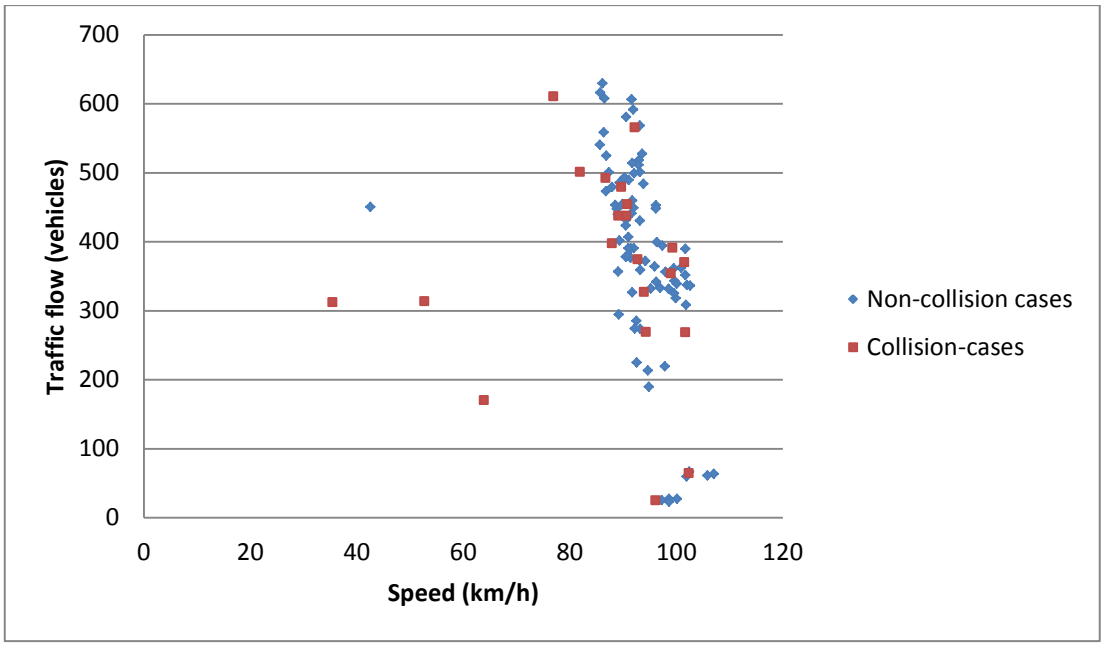


Figure 5. 4 Scatterplot of speed and flow of collision and non-collision cases for A12 (Link AL2291)

Table 5. 4 Average and standard deviation of speed and flow for collision and non-collision cases for A12 (Link AL2291)

Cases	Average of Speed	Average of Flow	StdDev of Speed	StdDev of Flow
0	93.3453	380.1909	7.4704	145.3434
1	86.6384	362.5479	16.9364	147.9691
Grand Total	92.0039	376.6623	10.3424	145.3271

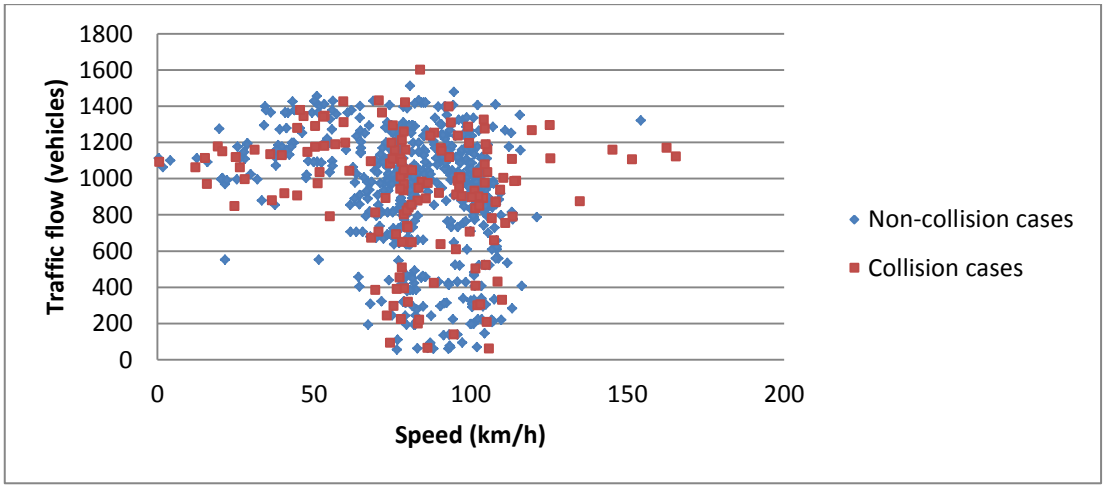


Figure 5. 5 Scatterplot of speed and flow of collision and non-collision cases for M62 (J25-30)

Table 5. 5 Average and standard deviation of speed and flow for collision and non-collision cases for M62(J25-30)

Cases	Average of Speed	Average of Flow	StdDev Speed	StdDev Flow
Non-Collision cases	82.1175	922.9011	21.7252	332.1784
Collision-cases	85.2015	934.4552	35.7459	331.0074
Grand Total	82.7343	925.2119	25.1639	331.7631

5.2.2. *Traffic and Collision data from Athens, Greece*

Due to the fact that traffic data from the UK were highly aggregated, alternatives were sought to locate and utilise disaggregated traffic data. As a result, traffic and collision data were provided by the Department of Transportation Planning and Engineering of the National Technical University of Athens. The data contain traffic and collision information during a 6-year period (2006-2011). Collision and traffic data concerned two major roads of the metropolitan area of Athens (i.e. Mesogeion and Kifisias avenues).

The collision database that was provided included the following variables:

- Collision : 0 for non-collision cases and 1 for collision cases
- Average of speed, occupancy and volume upstream and downstream of the collision location (3 * 2 locations= 6 traffic variables) in 5-minute intervals for 1-hour before the collision time

It should be noted that the 5-minute average correspond to the closest upstream detection from the location of the collision. For more information on the dataset the reader is referred to Theofilatos (2015).

As the focus of this thesis is the analysis of disaggregated traffic data, only the 5-minutes prior to the collision were extracted and used for the development of the models. The distribution of the crucial variables for the analysis is given in Figure 5.6.

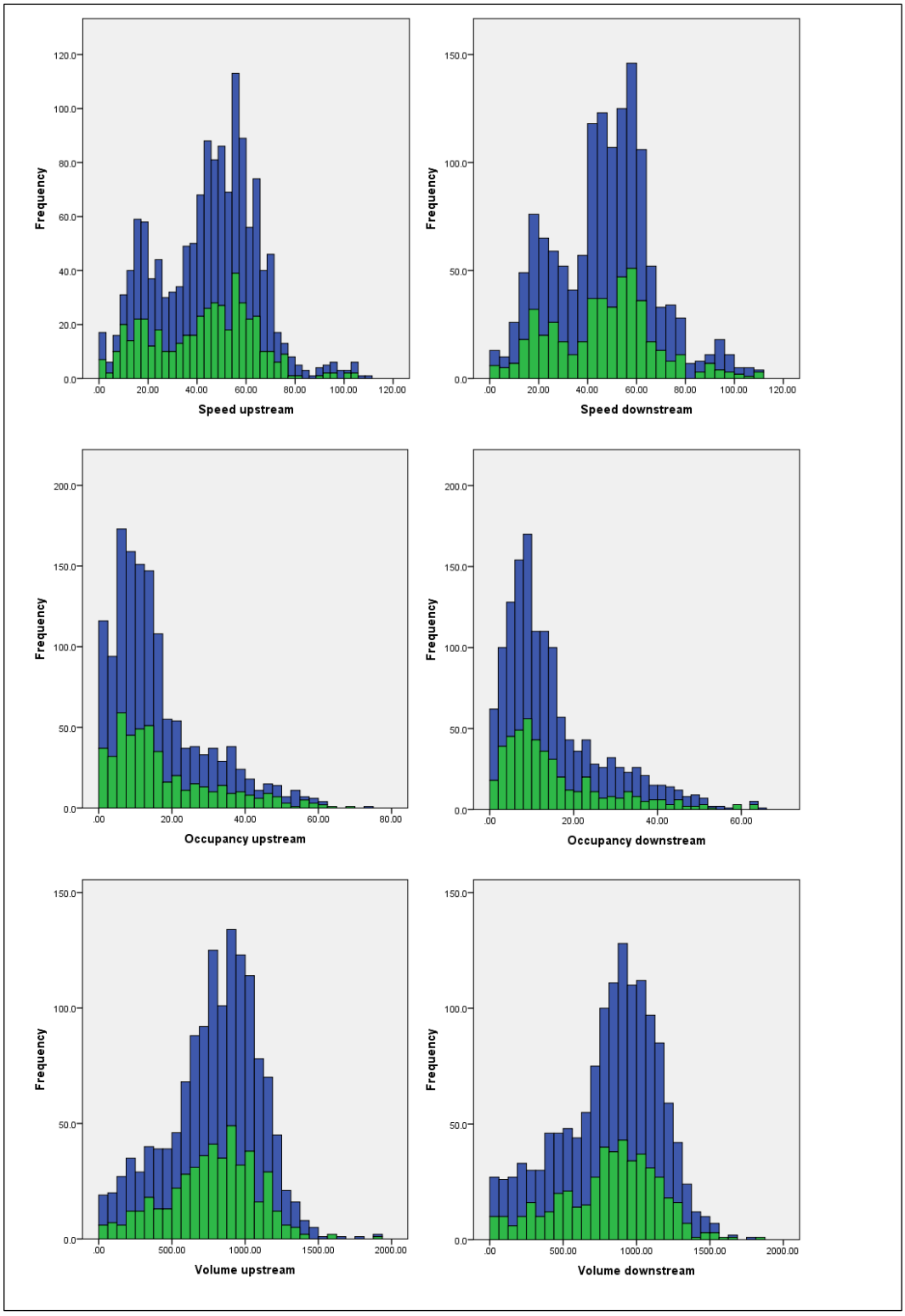


Figure 5. 6 Distribution of the interested variables in the Athens dataset (blue: collision-free cases, green: collision cases)

In order to obtain a clean dataset, rows with blank cells were deleted. The collision and non-collision cases from the obtained dataset are presented in Table 5.6 while the descriptive statistics of the included variables are presented in Table 5.7.

Table 5. 6 Total number of collision and non-collision cases in the Athens dataset

Cases	Total
Non-collision	917
Collision	472
Grand Total	1389

Table 5. 7 Descriptive statistics of the included variables for collision and non-collision cases of the Athens dataset

Statistic	Non-collision cases	Collision cases	Grand Total
Average of Speed upstream	45.80529	42.36091	44.63485
Average of Speed downstream	47.01363	46.22566	46.74587
Average of Occupancy upstream	15.50844	17.43996	16.16479
Average of Occupancy downstream	14.97205	15.10947	15.01875
Average of Volume upstream	801.8975	771.6759	791.6278
Average of Volume downstream	824.8129	806.8646	818.7138
StdDev of Speed upstream	19.6904	20.82605	20.14223
StdDev of Speed downstream	20.22874	21.16537	20.54758
StdDev of Occupancy upstream	12.47717	14.22642	13.12463
StdDev of Occupancy downstream	12.00751	12.5671	12.19619
StdDev of Volume upstream	313.406	305.7688	311.0514
StdDev of Volume downstream	337.4028	339.6497	338.1526
Min of Speed upstream	0	0	0
Min of Speed downstream	0	0	0
Min of Occupancy upstream	0	0	0
Min of Occupancy downstream	0	0	0
Min of Volume upstream	0	0	0
Min of Volume downstream	0	0	0
Max of Speed upstream	109.5	105	109.5
Max of Speed downstream	110	110	110
Max of Occupancy upstream	73.33	68.67	73.33
Max of Occupancy downstream	64	63.33	64
Max of Volume upstream	1886	1897.34	1897.34
Max of Volume downstream	1773.84	1849.34	1849.34

Figures 5.7. and 5.8. illustrate the relationship between speed and volume upstream and downstream of a collision location respectively. These figures depict the difficulty of the classification task between collision and safe cases as the data points overlap in the majority of the scatter plot.

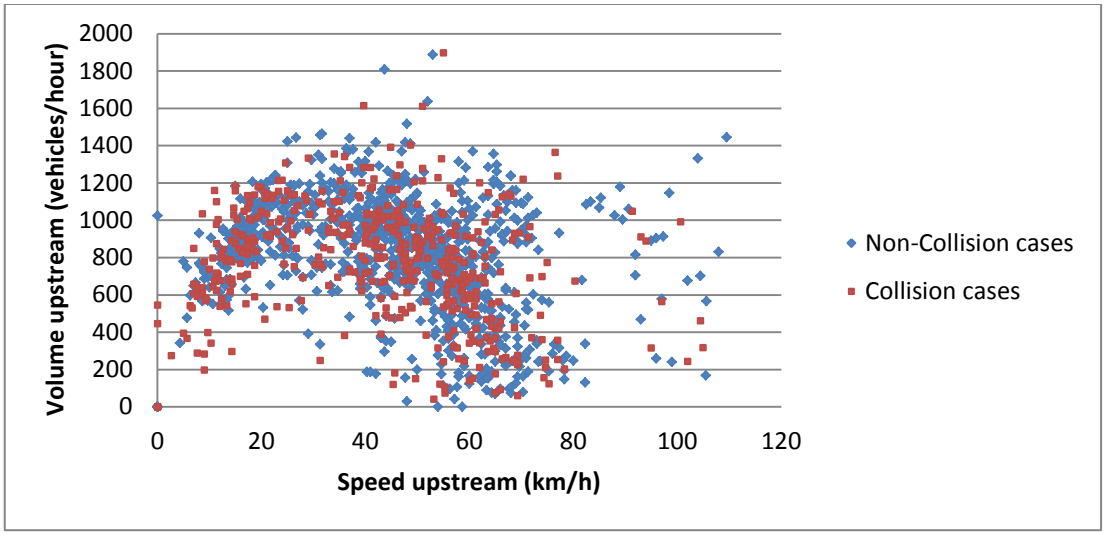


Figure 5. 7 Scatterplot of speed and volume upstream of the collision location for collision and non-collision cases in the Athens dataset

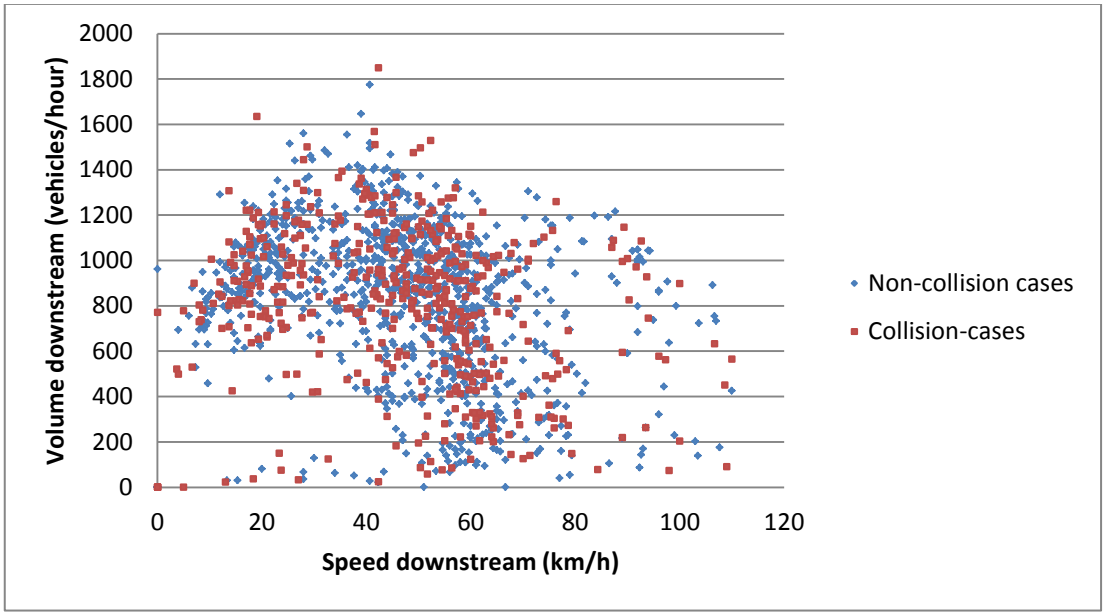


Figure 5. 8 Scatterplot of speed and volume downstream of the collision location for collision and non-collision cases in the Athens dataset

5.2.3. *Simulated traffic and conflicts data*

As the UK dataset contains highly aggregated data, it was decided to utilise these data to obtain highly disaggregated data for real-time safety evaluation.

A 4.52-km section of the M62 motorway between junctions 25 and 26 in England was selected as the study area. In order to build a robust micro-simulation model, the JTDB traffic data were split into four scenarios for the years 2012 and 2013:

- Morning peak hours (06:00 – 09:30)
- Morning off-peak hours (09:30-13:00)
- Afternoon off-peak hours (13:00-15:45)
- Afternoon peak hours (15:45-19:15)

For each of these scenarios the 15-minute traffic volumes and the cumulative speed distribution of the roadway segment were extracted and employed as input to VISSIM. An example of how the cumulative speed distribution was entered into VISSIM is shown in Figure 5.9.

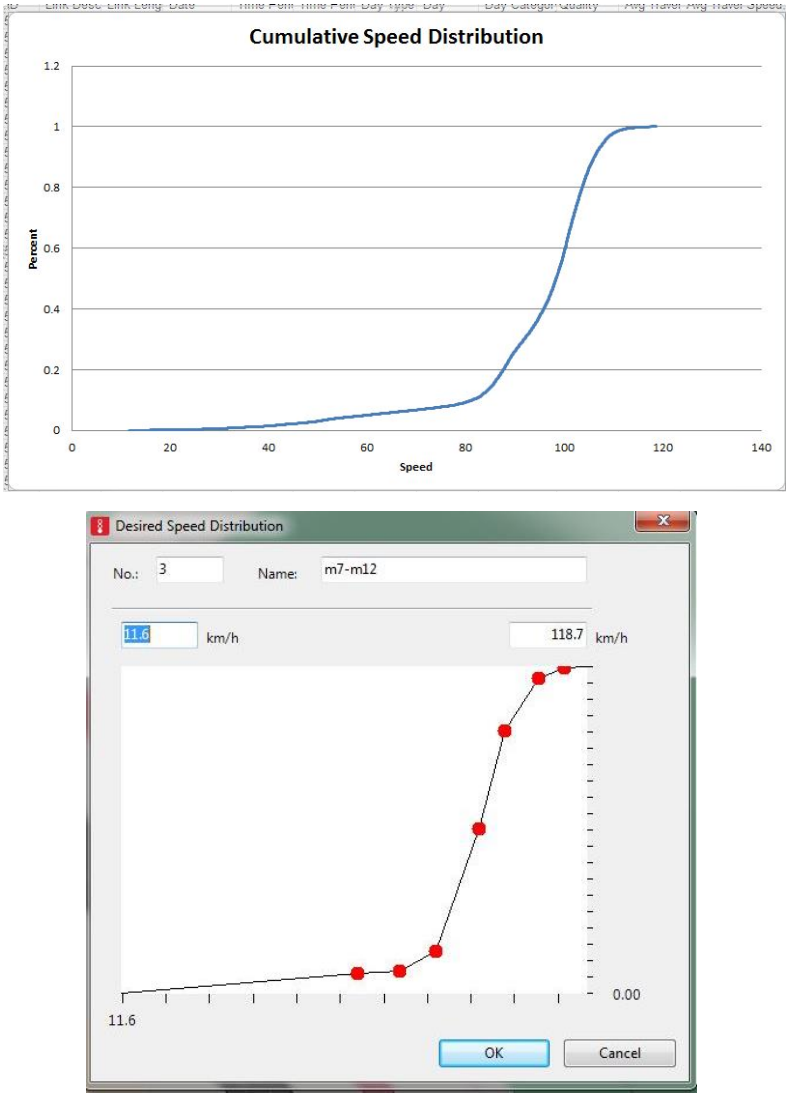


Figure 5. 9 Definition of cumulative speed distribution in VISSIM

Furthermore, the vehicle composition for 2012 and 2013 was also obtained from the UK Department of Transport (Department of Transport, 2012) and was used to build a micro-simulation model. The vehicle composition for the studied road segment is shown in Table 5.8.

Table 5. 8 Vehicle composition for the studied link segment (M62 motorway, junctions 25-26)

Year	**2012**		**2013**	
Vehicle category	**Number of vehicles**	**Ratio**	**Number of vehicles**	**Ratio**
Cars and LGV	57136	0.84100209	62591	0.85727
HGV	10643	0.156657541	10238	0.140224
Buses	159	0.002340369	183	0.002506
Total	67938	1	73012	1

The road segment was manually coded in VISSIM using a background image from OpenStreetMap (OpenStreetMap®, 2016) as seen in Figure 5.10. It was decided to allocate data collection detectors every 300m in order to acquire detailed traffic data. The spacing of the detectors was inspired by previous studies on NLCP on motorways (e.g. Hossain and Muromachi, 2012, Yu and Abdel-Aty, 2013).

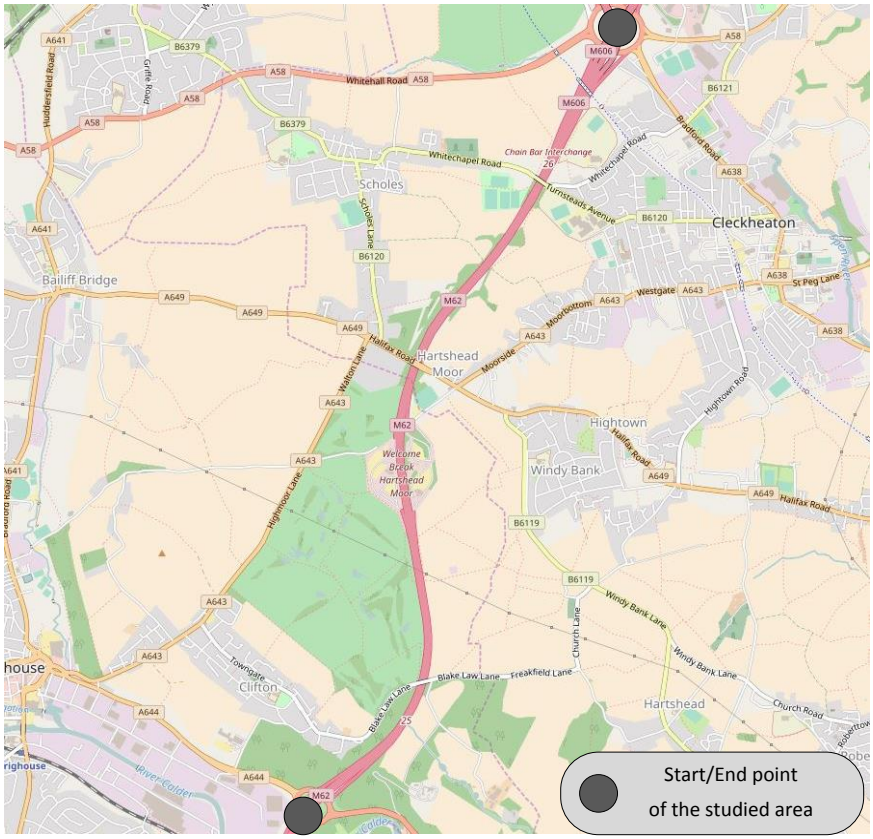


Figure 5. 10 The studied area viewed in OpenStreetMaps

In order for the micro-simulation to be initiated, the car-following model needed to be defined in VISSIM. The Wiedemann 99 model was selected because it applies to motorway scenarios (PTV Planug Trasport Verker AG, 2013). The Wiedemann model is characterised mainly by three parameters in VISSIM; the standstill distance, the headway time and the following variation (PTV Planug Trasport Verker AG, 2013). The standstill distance describes the average standstill distance between two vehicles. The headway time is the time gap (in seconds) which a driver wants to maintain at a certain speed. On the other hand, the following variation defines the desired safety distance a driver allows before moving closer to a car in front.

According to the guidelines from the Federal Highway Administration (FHWA) (Dowling et al., 2004) in order to validate the simulation results the GEH-statistic (Transport For London, 2010) and the link travel time were used. The GEH statistic correlates the observed traffic volumes with the simulated volumes, as shown below:

$$GEH = \sqrt{\frac{(V_{sim} - V_{obs})^2}{\frac{V_{sim} + V_{obs}}{2}}} \quad (5.2)$$

where V_{sim} is the simulated traffic volume and V_{obs} is the observed traffic volume.

After a number of trial simulations, the best GEH values were obtained by using the following parameters for the Wiedemann 99 car following model:

- Standstill distance: 1.5 m
- Headway time: 0.9 sec
- Following variation: 4 m

For the simulation to efficiently resemble real-world traffic it is essential that (Dowling et al., 2004):

1. GEH statistic < 5 for more than the 85% of the cases
2. The difference between observed and simulated travel times is equal or below 15% for more than 85% of the simulated cases.

The validation results are summarised in Fig. 5.11 and 5.12, and the comparison between traffic flow and travel time in simulation and reality are depicted in Figures 5.13 and 5.14.

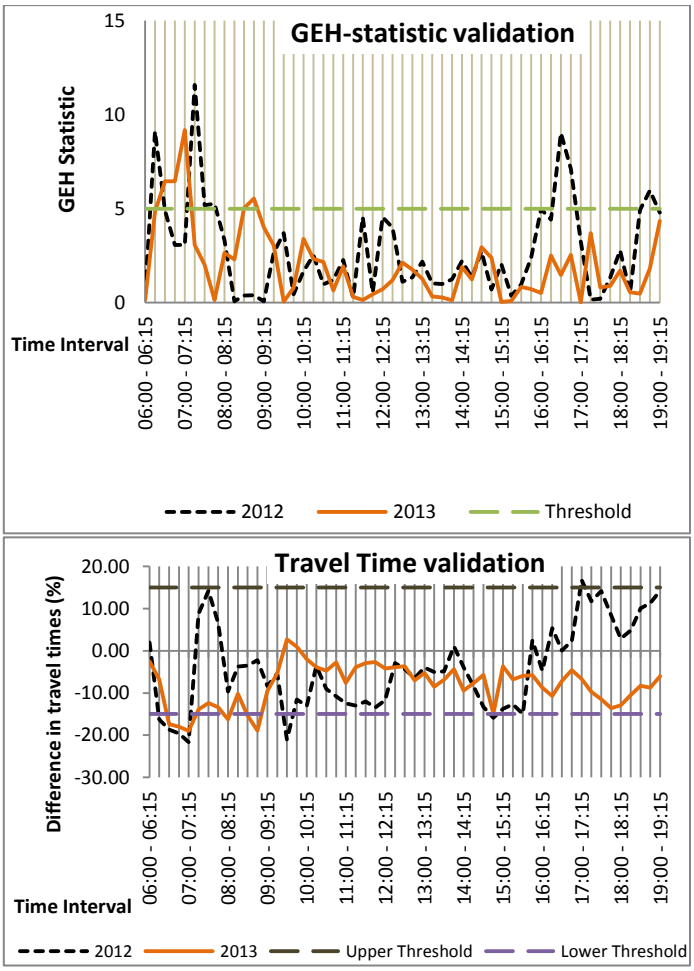


Figure 5. 11 GEH statistic and Travel time validation for each time interval and year.

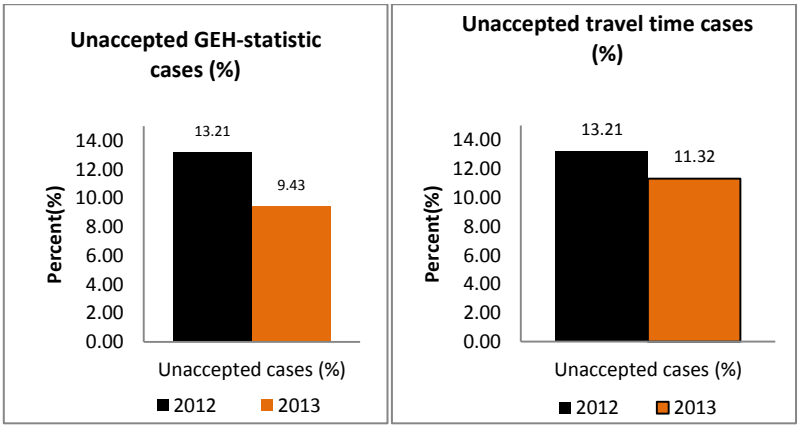


Figure 5. 12 Percentage of unaccepted cases for each year regarding the GEH statistic and travel time.

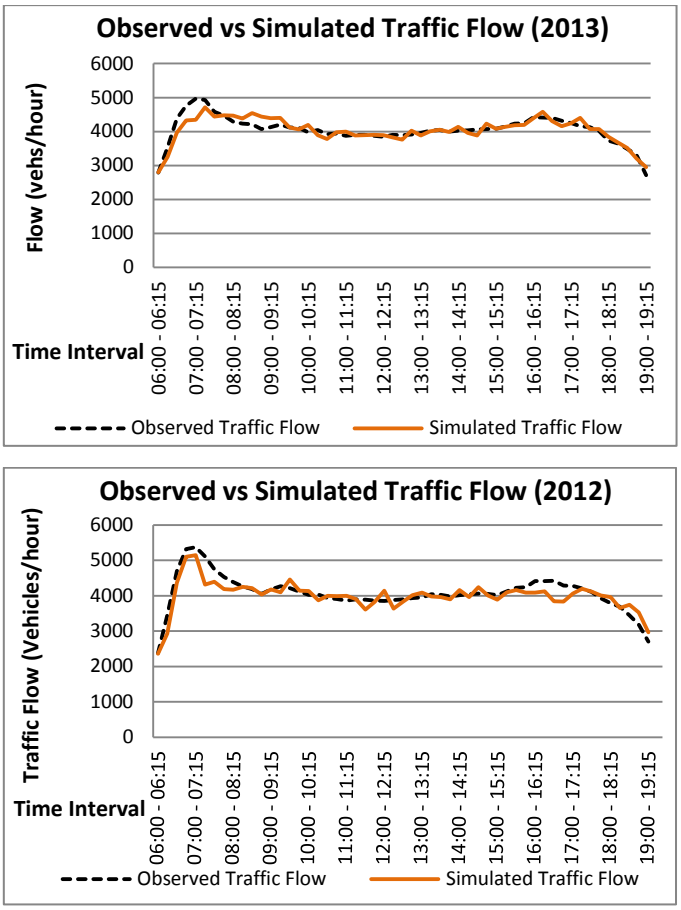


Figure 5. 13 Observed vs Simulated Traffic flow for each year

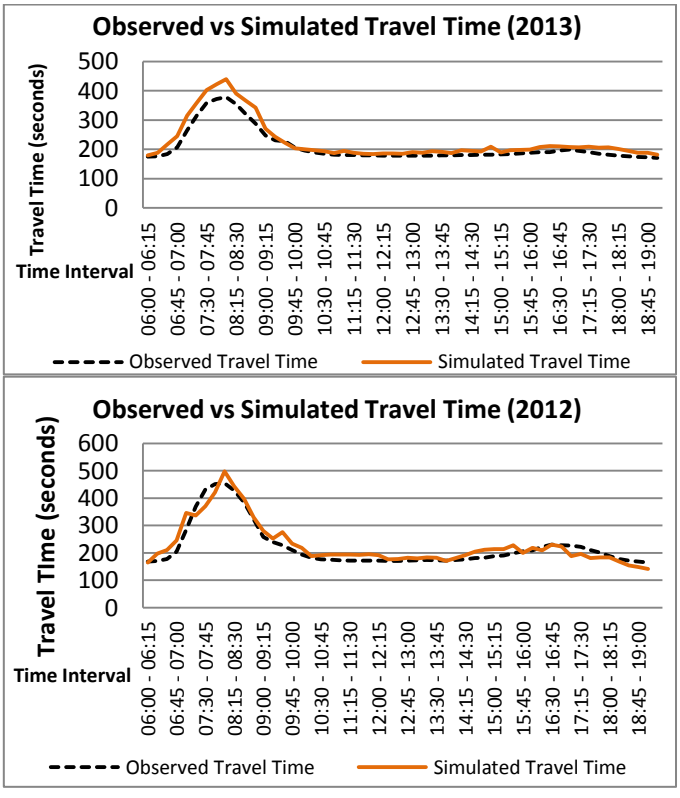


Figure 5. 14 Observed vs Simulated travel time for each year

In the simulations that were undertaken, the GEH values for most of the time intervals were found to be less than five. However, there were intervals where GEH values were found to be between 5 and 10. According to the Wisconsin Department of Transportation (WDOT, 2014) these values indicated either a calibration problem or a data issue. Because of the large number of simulations undertaken (~1000 for every scenario) it was assumed that the bad GEH values related to the bad quality of the available data (i.e. 15-minutes aggregated road-level traffic data). Therefore, it was decided to keep the simulation results for the corresponding intervals where GEH was slightly higher than the required value.

After calibrating the simulations, three additional simulations with different random seeds were run, resulting in a total of four different simulation results for each of the scenarios. The number of additional runs was chosen in order to address the imbalance between conflict and safe conditions which can prove essential for classification purposes (He and Garcia, 2009). The four different simulations were used for the matched-case control structure, where the first simulation was used to acquire the traffic conflicts and the other three were used to resemble the normal traffic conditions.

For the extraction of traffic conflicts, the vehicle trajectory files exported from VISSIM were inputted into the SSAM. Conflicts were detected if the TTC value between two vehicles was below 1.5 seconds and the PET value was below 4 seconds, which are the default values used in SSAM (Pu and Joshi, 2008). In the last step of the data processing, a MATLAB (Mathworks, 2016) code was developed in order to match the conflicts, exported from the SSAM, with the traffic conditions, acquired from VISSIM. The estimated conflicts were filtered again to identify conflicts with TTC below 1.3 seconds and PET below 1 second in order to obtain conflicts which are difficult to avoid. That is because TTC below 1.3 seconds is lower than the average human reaction time (Triggs and Harris, 1982) and PET values close to zero show imminent collisions (Pu and Joshi, 2008). Conflicts with TTC=0 and PET=0 are software errors according to the SSAM manual, as they resemble virtual crashes which are erroneously detected by SSAM and the

simulation model (Gettman et al., 2008). Therefore, such cases were also eliminated from the final dataset.

In order for the conflicts to be validated, the Crash Potential Index (CPI) was used as suggested by Cunto, (2008). CPI is calculated through the equation:

$$CPI_i = \frac{\sum_{t=t_{i_i}}^{tf_i} (P(MADR^{(a_1,a_2,\dots,a_n)} \leq DRAC_{i,t}) \cdot \Delta t \cdot b}{T_i} \quad (5.3)$$

where CPI_i is the CPI for vehicle i, while $DRAC_{i,t}$ is the deceleration rate to avoid the crash (m/s^2). Also, $MADR^{(a_1,a_2,\dots,a_n)}$ is a random variable following normal distribution for a given set of environmental attributes, t_{i_i} and $t_{f_i}$are the initial and final simulated time intervals for vehicle i. In addition, Δt is the simulation time interval (sec) and T_i is the total travel time for vehicle i while b is a binary state variable denoting a vehicle interaction. For MADR according to Cunto, (2008), a normal distribution with average of 8.45 for cars and 5.01 for HGVs with a standard deviation of 1.4 was assumed for daylight and dry pavements. The results for the calibration of the conflicts are shown in Figure 5.15

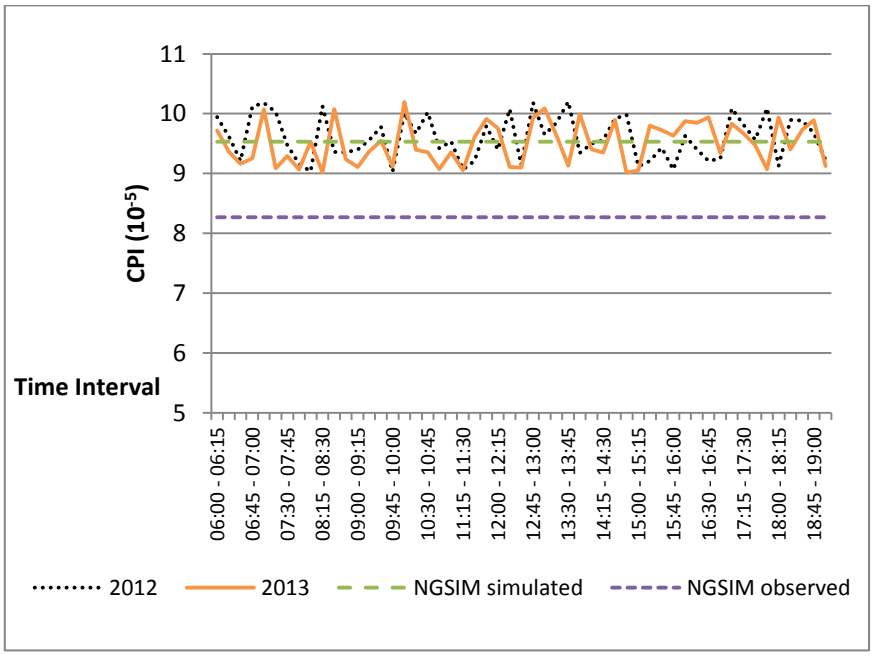


Figure 5. 15 Conflicts validation

Figure 5.15 shows that for the majority of the time intervals, CPI is similar to the simulated CPI of the NGSIM dataset and close to the values of the observed NGSIM

CPI. Therefore, it was assumed that the simulated conflicts resembled realistic hazardous scenarios.

As conflicts extracted by SSAM and traffic conditions acquired by VISSIM were time stamped, it was concluded that the issue of incorrectly reported collision times has been resolved. The overall methodology of capturing the required data for classification purposes is shown in Figure 5.16.

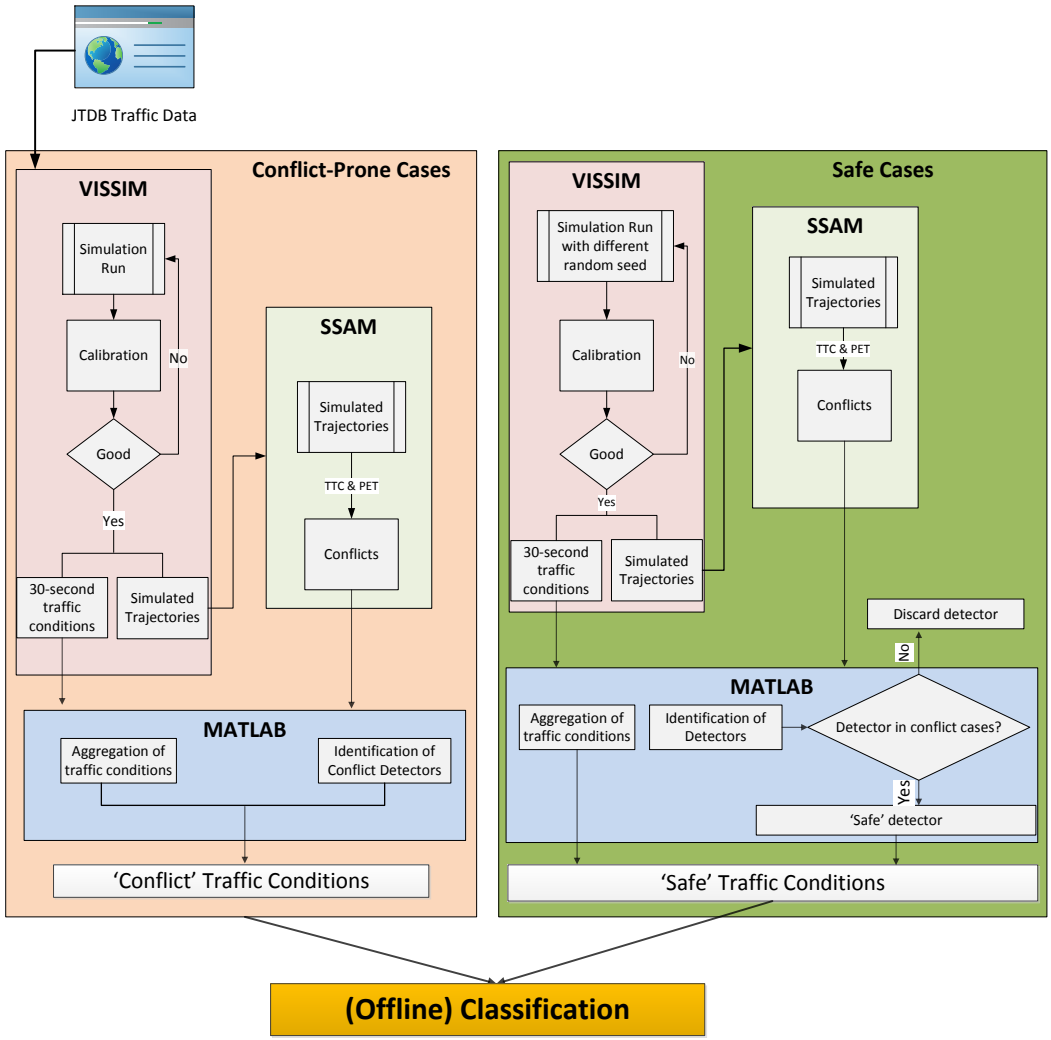


Figure 5. 16 Flow chart of the procedure followed to classify traffic conditions from simulated data

To elaborate more on the procedure followed to obtain collision and non-collision cases, for every conflict, the nearest upstream detector on the road segment was identified by comparing the time of the conflict with the time the vehicles passed from every detector. This specific detector was marked as "conflict detector". Traffic

data were extracted for every conflict detector, the corresponding upstream and downstream detectors on the same lane and the detector in the adjacent lane for every time interval. The traffic measurements for these detectors were marked as “conflicts” because they represent the traffic conditions near the time when the conflict occurred.

In order to obtain the non-collision cases for every conflict detector the conflicts for the other three simulation runs were assessed to see if any conflicts occurred in their vicinity in these runs. If there was no conflict, the traffic measurements from that detector were obtained to represent safe conditions. Otherwise the detector was discarded.

For each of the detectors and for every time interval the number of vehicles, the vehicle speeds and the vehicle accelerations were extracted. The traffic data exported from VISSIM were then aggregated in 30-second, 1-minute, 3-minute and 5-minute intervals prior to the conflict occurrence. The 30-second measurements were considered the “raw” traffic measurements and hence for the 30-second data only the average 30-second measurements from the detectors mentioned above were used. For the 1-minute, 3-minute and 5-minute aggregation intervals, the average and standard deviation of the 30-second raw measurement was estimated for every detector. As four simulations were run, having used one simulation for the extraction of conflict-prone conditions and the three other simulations for the extraction of collision-free conditions, the procedure was repeated an additional three times so that every simulation run was used for the extraction of both conflict-prone and safe conditions. After extracting the safe and conflict-prone conditions the final dataset contained 7800 conflicts and the corresponding 23400 non-conflict cases. The descriptive statistics of the raw 30-second measurements are given in Table 5.9. The predictors used for the 30-second dataset, as well as the 1-minute, 3-minute and 5-minute datasets, is given in Table 5.10. Moreover, the distribution of the raw 30-second measurements for conflict and non-conflict cases is shown in Figure 5.17. Finally, to illustrate the massive scale and complexity of the 30-second raw measurements a scatterplot of speed and flow at the conflict detector is presented in Figures 5.18.

Table 5. 9 Descriptive statistics for the simulation dataset

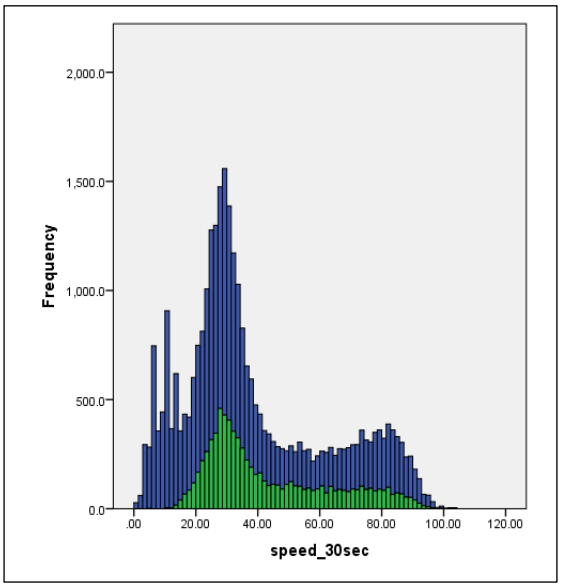
Collision		speed_30sec	vehicles_30sec	acceleration_30sec
Non-Conflict cases	Mean	37.398	9	0.011
	Std. Dev	24.281	5	0.472
	Min	1.000	0	-3.440
	Maxi	103.290	25	2.430
Conflict cases	Mean	43.581	11	-0.011
	Std. Dev	20.201	4	0.516
	Min	5.630	0	-3.070
	Max	102.130	25	2.110
Total	Mean	38.944	10	0.005
	Std. Dev	23.481	5	0.483
	Min	1.000	0	-3.440
	Max	103.290	25	2.430
Collision		**speed_30sec_downstream**	**vehicles_30sec_downstream**	**acceleration_30sec_downstream**
Non-Conflict cases	Mean	37.467	9	0.015
	Std. Dev	24.329	6	0.479
	Min	1.000	0	-4.740
	Max	105.280	24	1.990
Conflict cases	Mean	44.098	10	0.019
	Std. Dev	21.742	4	0.489
	Min	5.310	0	-2.970
	Max	103.880	26	1.820
Total	Mean	39.124	10	0.016
	Std. Dev	23.882	5	0.481
	Min	1.000	0	-4.740
	Max	105.280	26	1.990
Collision		**speed_30sec_upstream**	**vehicles_30sec_upstream**	**acceleration_30sec_upstream**
Non-Conflict cases	Mean	37.816	9	-0.001
	Std. Dev	24.346	5	0.473
	Min	1.000	0	-6.870
	Max	103.140	25	2.110
Conflict cases	Mean	45.958	10	-0.017
	Std. Dev	22.255	4	0.482
	Min	7.810	0	-2.760
	Max	99.080	23	2.420
Total	Mean	39.852	9	-0.005
	Std. Dev	24.100	5	0.476
	Min	1.000	0	-6.870
	Max	103.140	25	2.420

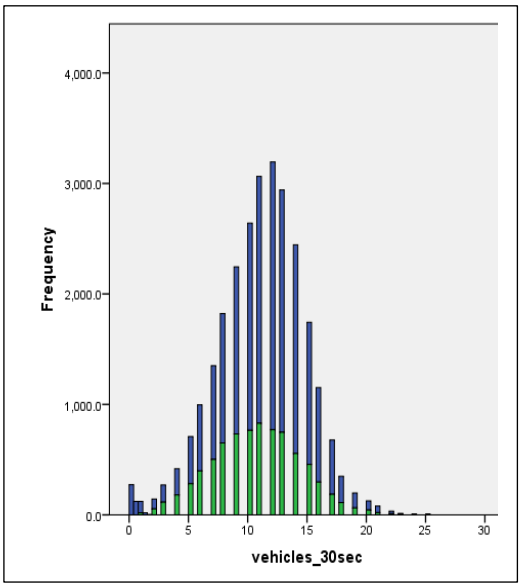
Collision		speed_30sec_adjacentlane1	vehicles_30sec_adjacentlane1	acceleration_30sec_adjacentlane1
Non-Conflict cases	Mean	38.550	9	0.026
	Std. Dev	24.833	6	0.497
	Min	1.000	0	-6.180
	Max	104.570	25	2.480
Conflict cases	Mean	44.838	11	0.018
	Std. Dev	20.848	4	0.532
	Min	4.820	0	-2.970
	Max	104.440	25	2.310
Total	Mean	40.122	10	0.024
	Std. Dev	24.054	5	0.506
	Min	1.000	0	-6.180
	Max	104.570	25	2.480

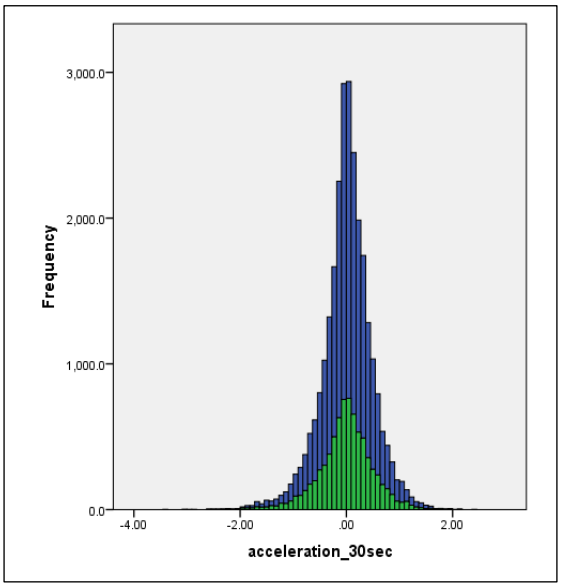
Table 5. 10 Description of the variables included in the simulation dataset

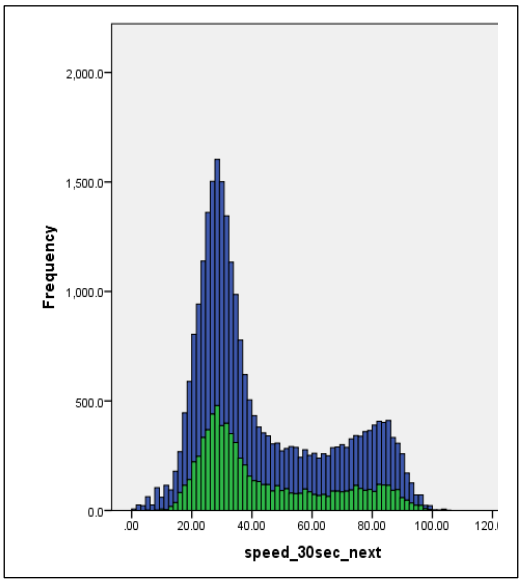
Dataset	Variable	Description
30-second	Collision	Conflict case (1) or non-conflict case (0)
	speed_30sec	Average 30-second speed at the conflict detector
	vehicles_30sec	Average 30-second flow at the conflict detector
	acceleration_30sec	Average 30-second acceleration at the conflict detector
	speed_30sec_next	Average 30-second speed at the downstream detector
	vehicles_30sec_next	Average 30-second flow at the downstream detector
	acceleration_30sec_next	Average 30-second acceleration at the downstream detector
	speed_30sec_previous	Average 30-second speed at the upstream detector
	vehicles_30sec_previous	Average 30-second flow at the upstream detector
	acceleration_30sec_previous	Average 30-second acceleration at the upstream detector
	speed_30sec_nextlane1	Average 30-second speed at the detector in the adjacent lane
	vehicles_30sec_nextlane1	Average 30-second flow at the detector in the adjacent lane
	acceleration_30sec_nextlane1	Average 30-second acceleration at the detector in the adjacent lane
1-minute 3-	Collision	Conflict case (1) or non-conflict case (0)
	speed_1min	Average 1-minute speed at the

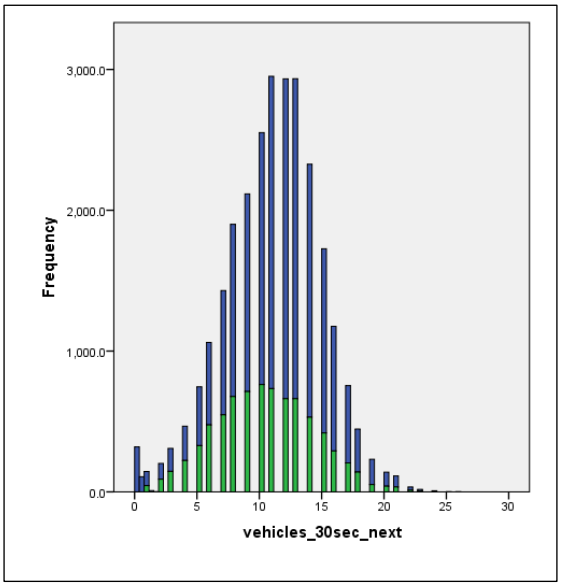
minute 5-minute		conflict detector
	vehicles_1min	Average 1-minute flow at the conflict detector
	acceleration_1min	Average 1-minute acceleration at the conflict detector
	speed_1min_next	Average 1-minute speed at the downstream detector
	vehicles_1min_next	Average 1-minute flow at the downstream detector
	acceleration_1min_next	Average 1-minute acceleration at the downstream detector
	speed_1min_previous	Average 1-minute speed at the upstream detector
	vehicles_1min_previous	Average 1-minute flow at the upstream detector
	acceleration_1min_previous	Average 1-minute acceleration at the upstream detector
	speed_1min_nextlane1	Average 1-minute speed at the detector in the adjacent lane
	vehicles_1min_nextlane1	Average 1-minute flow at the detector in the adjacent lane
	acceleration_1min_nextlane1	Average 1-minute acceleration at the detector in the adjacent lane
	speed_1min_stddev	1-minute standard deviation of speed at the conflict detector
	vehicles_1min_stddev	1-minute standard deviation of flow at the conflict detector
	acceleration_1min_stddev	1-minute standard deviation of acceleration at the conflict detector
	speed_1min_stddev_next	1-minute standard deviation of speed at the downstream detector
	vehicles_1min_stddev_next	1-minute standard deviation of flow at the downstream detector
	acceleration_1min_stddev_next	1-minute standard deviation of acceleration at the downstream detector
	speed_1min_stddev_previous	1-minute standard deviation of speed at the upstream detector
	vehicles_1min_stddev_previous	1-minute standard deviation of flow at the upstream detector
	acceleration_1min_stddev_previous	1-minute standard deviation of acceleration at the upstream detector
	speed_1min_stddev_nextlane1	1-minute standard deviation of speed at the detector in the adjacent lane
	vehicles_1min_stddev_nextlane1	1-minute standard deviation of flow at the detector in the adjacent lane
	acceleration_1min_stddev_nextlane1	1-minute standard deviation of acceleration at the detector in the adjacent lane

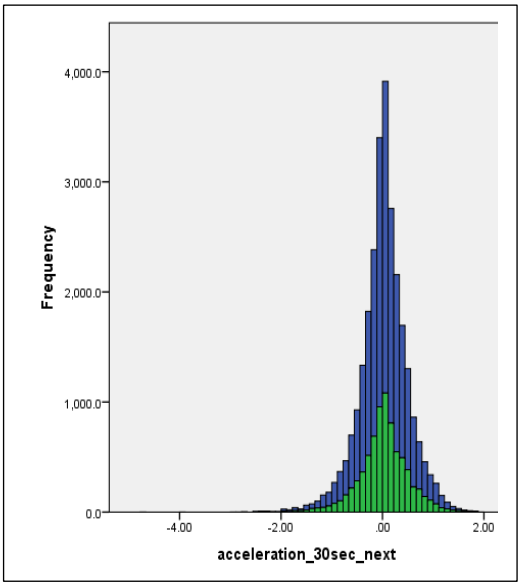

Frequency
2,000.0
1,500.0
1,000.0
500.0
0.0
.00
20.00
40.00
60.00
80.00
100.00
120.00
speed_30sec


Frequency
4,000.0
3,000.0
2,000.0
1,000.0
0.0
0
5
10
15
20
25
30
vehicles_30sec


Frequency
3,000.0
2,000.0
1,000.0
0.0
-4.00
-2.00
.00
2.00
acceleration_30sec


Frequency
2,000.0
1,500.0
1,000.0
500.0
0.0
.00
20.00
40.00
60.00
80.00
100.00
speed_30sec_next


Frequency
3,000.0
2,000.0
1,000.0
0.0
0
5
10
15
20
25
30
vehicles_30sec_next


Frequency
4,000.0
3,000.0
2,000.0
1,000.0
0.0
-4.00
-2.00
.00
2.00
acceleration_30sec_next

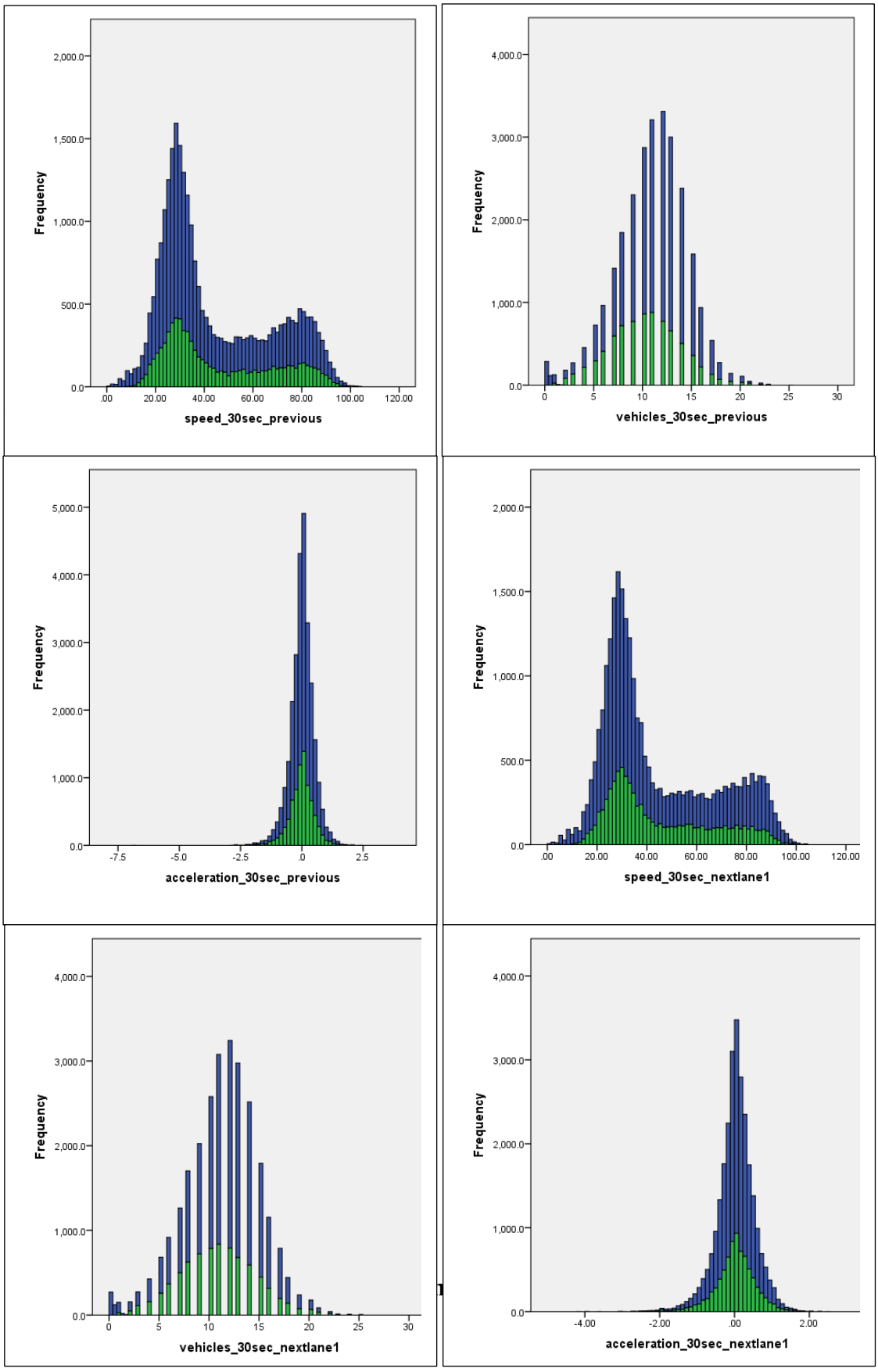


Figure 5. 17 Distribution of the variables included in the 30-second simulation dataset (blue: non-conflict cases, green: conflict cases)

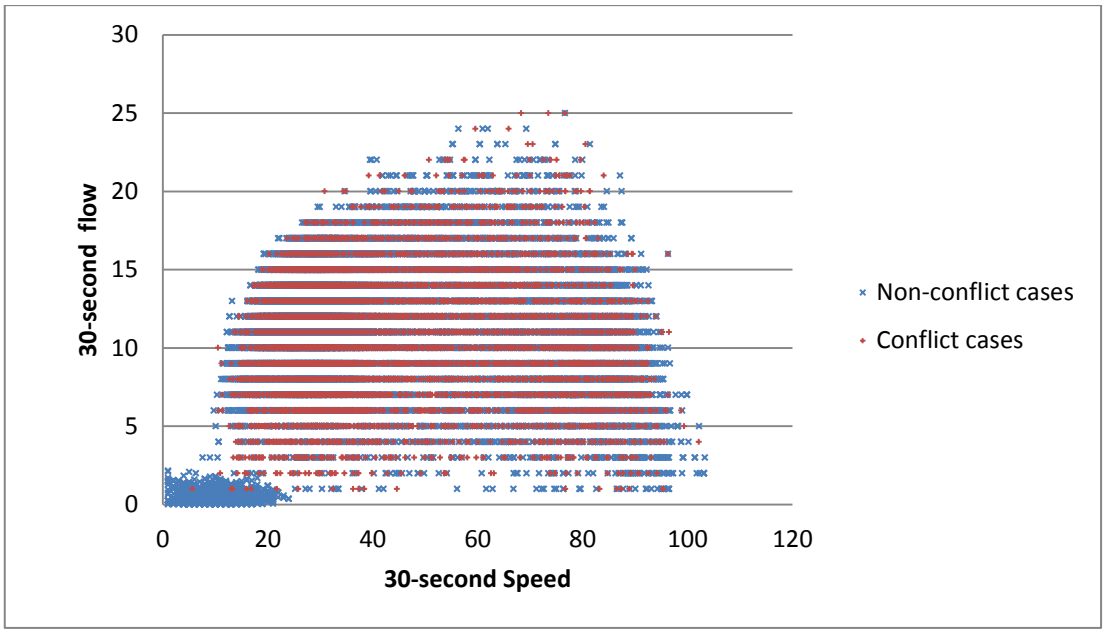


Figure 5. 18 Scatterplot of 30-second speed and flow at the conflict detector for conflict and non-conflict cases

5.3.Limitations of the network-level datasets

5.3.1. *Limitations of the UK dataset*

The UK dataset comprises of highly aggregated traffic data both temporally and spatially. Although this type of aggregation enables the creation of a flexible and small dataset, it inevitably leads to loss of data comprehensiveness. 15 minutes is not representative of the traffic conditions just before a collision because traffic conditions can vary significantly in between. Furthermore, one observation per an entire motorway segment cannot explicitly describe traffic conditions. Therefore, such temporal and spatial resolution cannot precisely define pre-collision conditions and may potentially lead to a lot of misclassifications.

Regarding the collision data, the largest problem is the effect of the reported collision time. As also seen in the literature erroneous reporting time is prominent in collision databases (Kockelman and Ma, 2007, Imprialou, 2015). STATS 19 data include that error as well and nothing can be done to correct such a mistake. However it is argued by the literature (i.e. Imprialou, 2015) that using 15-minute traffic data counterbalances the existing error in the reported collision time.

5.3.2. *Limitations of the Athens dataset*

The Athens dataset was obtained in its entirety after pre-processing as it has been used in previous research from the National Technical University of Athens. Hence erroneous traffic data had already been dismissed from the dataset. However, access to the raw traffic data was not possible, thus limiting the predictors which could be used for the analysis to only the 5-minute averages of speed, volume and occupancy. Regarding the collision data, it is assumed that the issue of erroneous collision time reporting has not been resolved in the Athens dataset either.

5.3.3. *Limitations of the simulated dataset*

The dataset provided by the simulations from VISSIM has been calibrated and validated. It contains highly disaggregated traffic data as realistic as possible, however it is the outcome of computer software and as much realistic as it can be, it could never replace real-world traffic data. Furthermore, the fact that the simulation was based on highly aggregated traffic data does not allow the simulated environment to be described in much detail and more thorough calibrated.

Furthermore, the fact that microsimulation cannot result in the extraction of collisions but only conflicts limits the classifiers to predict only conflict events. In order for collisions to be predicted, the conflict events extracted from SSAM need to be validated with real-world observations from the same site, however this was not inside the scope of this study. Hence, the simulation dataset was utilised only for predicting the traffic conditions which lead to conflicts within the simulation software.

5.4.Vehicle –level data

In order to integrate NLCP and vehicle-level risk assessment as needed by AVs, vehicle-level data need to be collected. This section will describe the data-collection platform and the available data.

5.4.1. *Data collection platform*

All the vehicle-level data was collected using the instrumented vehicle of the School of Civil and Building Engineering of Loughborough University. The vehicle is equipped with the following sensors:

- a PointGrey© Grasshopper3 4.1 MP Near InfraRed (NIR) Camera
- an ARS 308-21 short and long-range Continental© automotive radar
- a u-blox© NEO M8-L GNSS and 3D Dead Reckoning system
- a Mobileye© 560 lane-departure and forward collision warning camera system

All the sensors are aligned along the centre of the longitudinal axis of the car. The position of the sensors and the experimental vehicle are depicted in Figure 5.19.

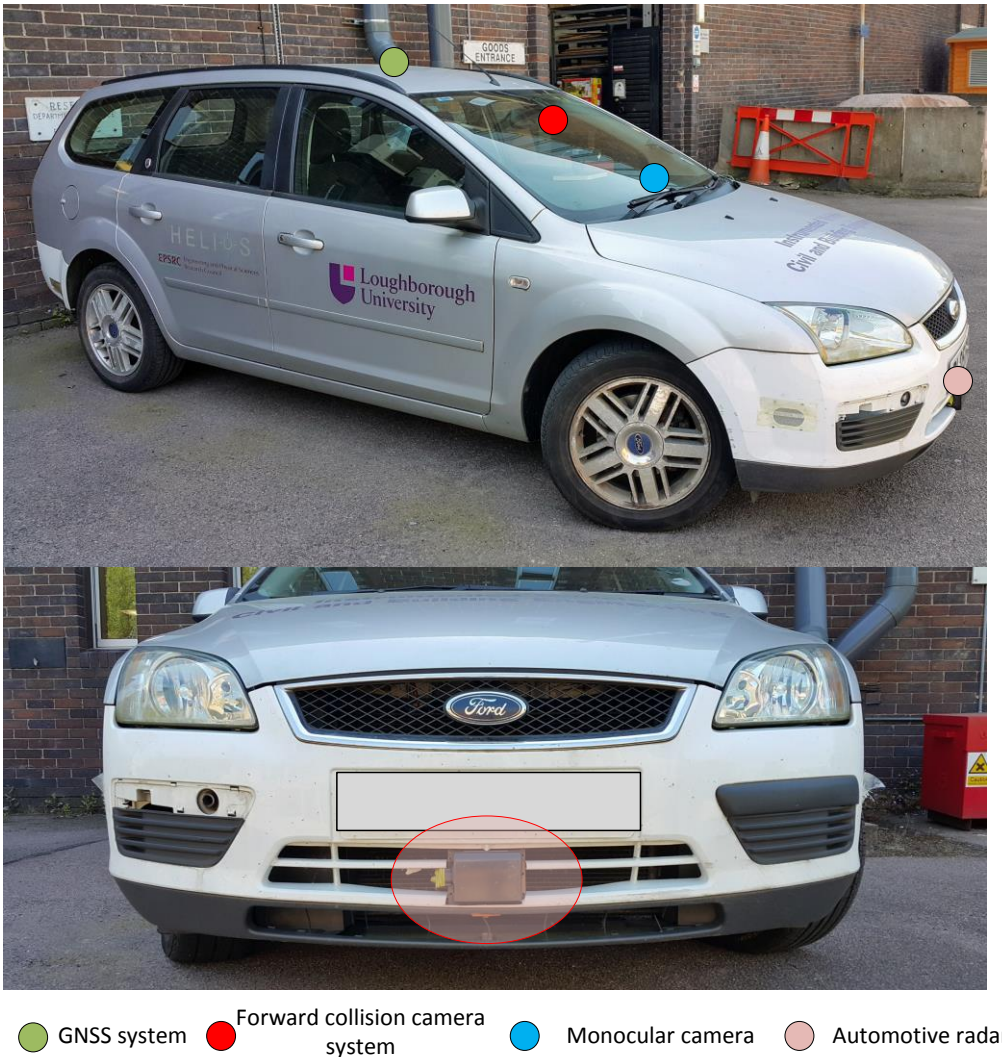


Figure 5. 19 The experimental vehicle along with its sensors

For the purposes of this thesis, only data from the GNSS system and the automotive radar have been used.

5.4.2. *Available vehicle data*

The vehicle data were collected on April 23rd, 2017, between 10:53 am and 11:51 am on the M1 motorway (J23-J18) from Loughborough to the Watford Gap service station. The route that was followed is depicted in Figure 5.20

Figure 5. 20 The driving route for the vehicle-level data collection

The speed of the ego-vehicle as measured by the GNSS module during the driving trip is depicted in Figure 5.21

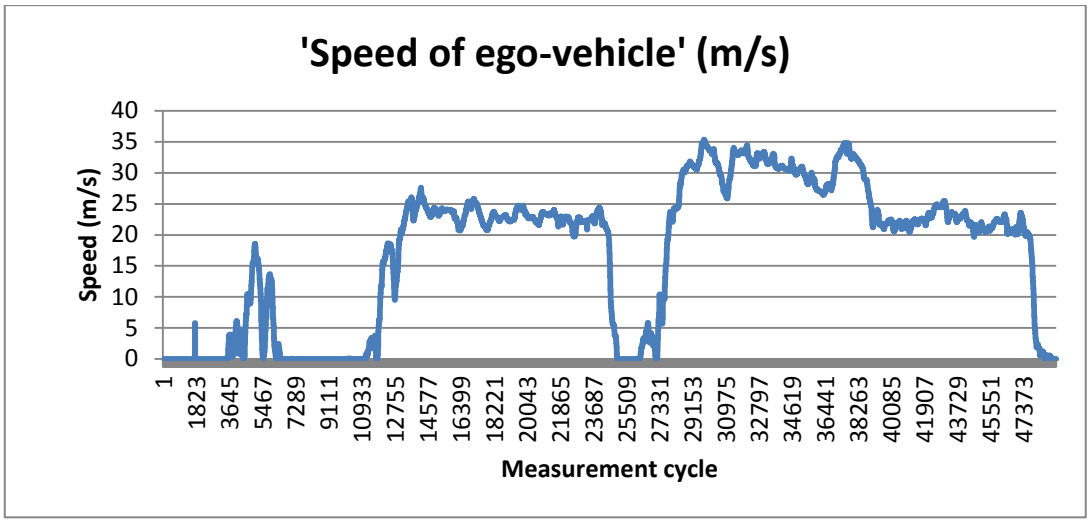


Figure 5. 21 Ego-vehicle speed during the driving trip

5.4.2.1.Radar data

For the purposes of this PhD research project, the data from the radar sensor have been primarily used. The long range sensor of the radar can detect objects with a field of view of 17° up to 200 m away, and its short-range sensor can detect with a field of view 54° up to 60 m away. The radar identifies targets and objects. A target can be anything which reflects radar waves. An object is a target which has been traced by the software used by the radar sensor over a few measurements. Only the object measurements have been used, as they are more representative of the vehicles and obstacles surrounding the ego-vehicle. The radar sensor cycle is 15.15 Hz and the variables of interest are depicted in Figure 5.22 and are:

- *NoOfObjectsTime and NoOfObjectsY*: Number of objects which have been traced in this measurement cycle; the value indicates how many rows of the output dataset include recognised objects, and which rows have been filled with random numbers
- *Obj_AccelLongTime and Obj_AccelLong[m/s2]:* Relative longitudinal acceleration of an object in m/s^2
- *Obj_DynPropTime and Obj_DynPropY*:Movement of the object; 0: unclassified, 1: standing, 2: stopped (never moved before), 3: moving, 4 oncoming
- *Obj_IDTime and Obj_IDY*:Identification number of the object; all objects, which have an identification number, that is higher than the value of NoOfObjectsY, are not a traced object and the row is filled with random numbers

- *Obj_LatDisplTime and Obj_LatDispl[m]:* Lateral displacement in m
- *Obj_LatSpeedTime and Obj_LatSpeed[m/s]:* Object lateral velocity; negative value means that the object moves to the right; positive value means that the object moves to the left in m/s
- *Obj_LengthTime and Obj_LengthY*:Length of the Object; 0: unknown;1: < 0.5 m; 2: < 2 m; 3: < 4 m; 4: < 6 m; 5: < 10 m; 6: < 20 m;7: exceeds 6m
- *Obj_LongDisplTime and Obj_LongDispl[m]:* Longitudinal displacement of the object in m
- *Obj_MeasStatTime and Obj_MeasStatY:*Object measurement status; 0: no object, 1: new object, 2: object not measured, 3: object measured
- *Obj_ObstacleProbabilityTime and Obj_ObstacleProbability:* Probability that the object is an obstacle
- *Obj_ProbOfExistTime and Obj_ProbOfExistY:* Probability of the existence of an object
- *Obj_VrelLongTime and Obj_VrelLong[m/s]:* Object relative longitudinal velocity in m/s
- *Obj_WidthTime and Obj_WidthY:*Width of the object; 0: unknown; 1: < 0.5 m (pedestrian); 2: < 1 m (bike); 3: < 2 m (car); 4: < 3m (truck); 5: < 4 m; 6: < 6 m; 7: exceeds 6m

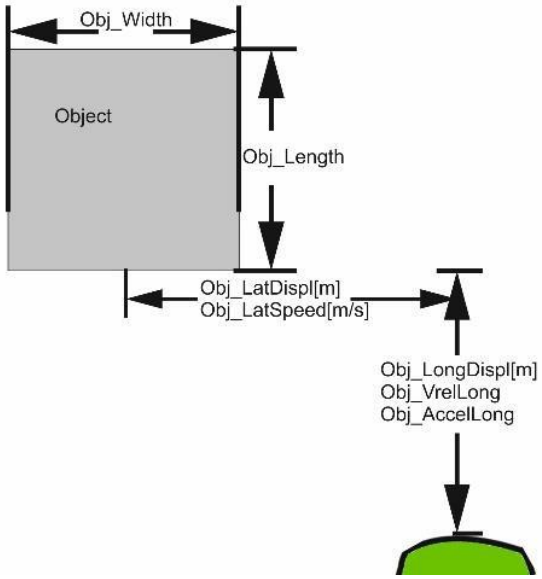


Figure 5. 22 Illustration of the variables measured by the sensor(Schnieder, 2017)

The total number of vehicles sensed by the ego-one during the driving trip is depicted in Figure 5.23

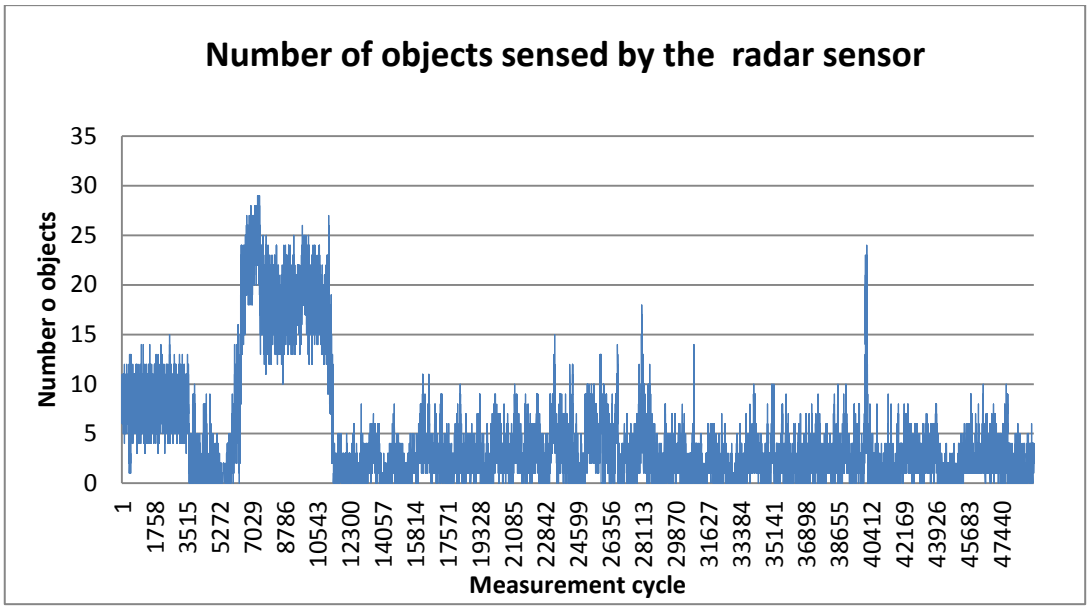


Figure 5. 23 Number of objects detected by the radar sensor per measurement cycle

5.4.3. *Estimation of Time-To-Collision(TTC)*

The next step of the vehicle data-processing was to estimate the TTC of the ego-vehicle regarding the vehicles in its vicinity. The approach used to estimate TTC was obtained from a European short project report within Loughborough University. (Schnieder, 2017). Similar to Ward et al., (2014, 2015) critical encounters between vehicles based on the bearing angle, the loom angle and the yaw rate were detected. The bearing angle γ is defined as the angle between the velocity vector of the ego-vehicle's loom point and the vector at the closest point of the target vehicle. The loom angle θ is the angle between the furthest left and furthest right point from the loom point on the ego-vehicle. The yaw rate ω is the change of the heading angle of the ego-vehicle. The bearing angle, loom angle and yaw rate are depicted in Figure 5.24

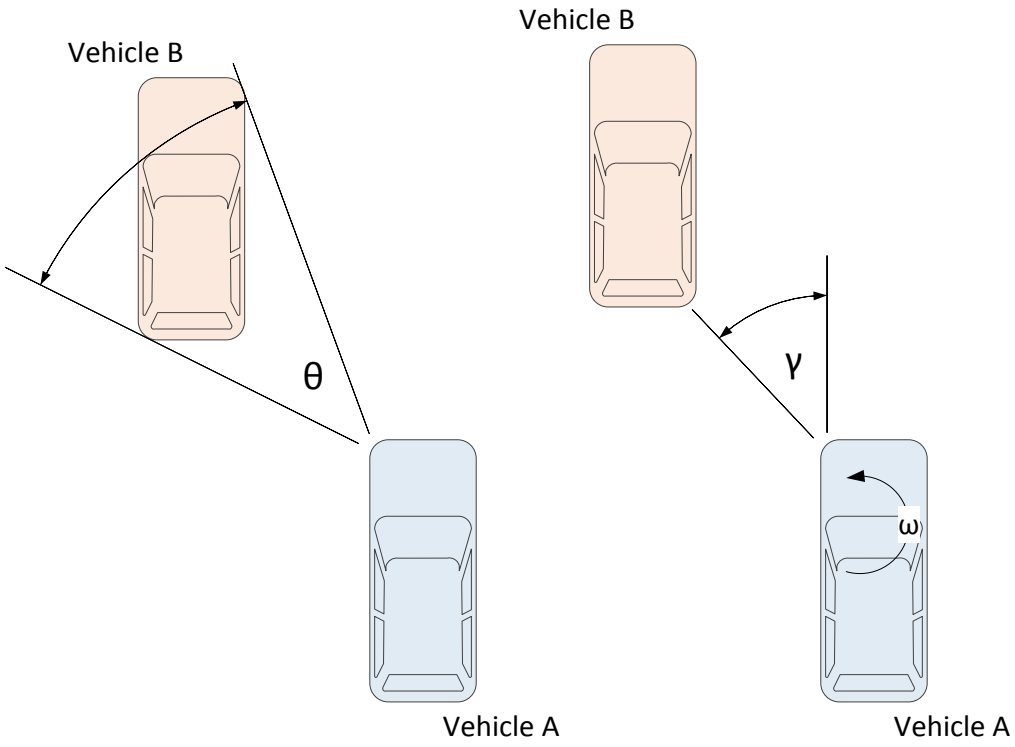


Figure 5. 24 Illustration of bearing angle (γ), loom angle (θ) and yaw rate (ω)

For the first measurement, every relevant object is identified and the angles α, β, the bearing angle γ as well as the distance d is calculated for each object; α and β are the angles between the velocity vector of the loom point on the green ego-vehicle and the furthest left and furthest right point of the red target-vehicle, respectively, as illustrated in Figure 5.25. In Figure 5.25a the angles α, β, and γ as well as the distance d are illustrated if the target object is on the left-hand side. The corresponding angles if the target object is on the right-hand side are depicted in Figure 5.25b. In order to estimate these angles, the required measurements from the radar sensor are:

o_w: Width of the object

$o_{la,}$: Lateral displacement

v_w: Width of the ego-vehicle including buffer area (here 2.1 m)

$o_{lo,}$: Longitudinal displacement of the object

o_l: Object length

b_w: Object buffer area: if the target object is a bicycle, the side buffer area is 3

feet; otherwise the buffer area depends on the ego-vehicle's velocity

b_f: Object front buffer area according to the velocity of the ego-vehicle

The above required variables are depicted in Figure 5.26.

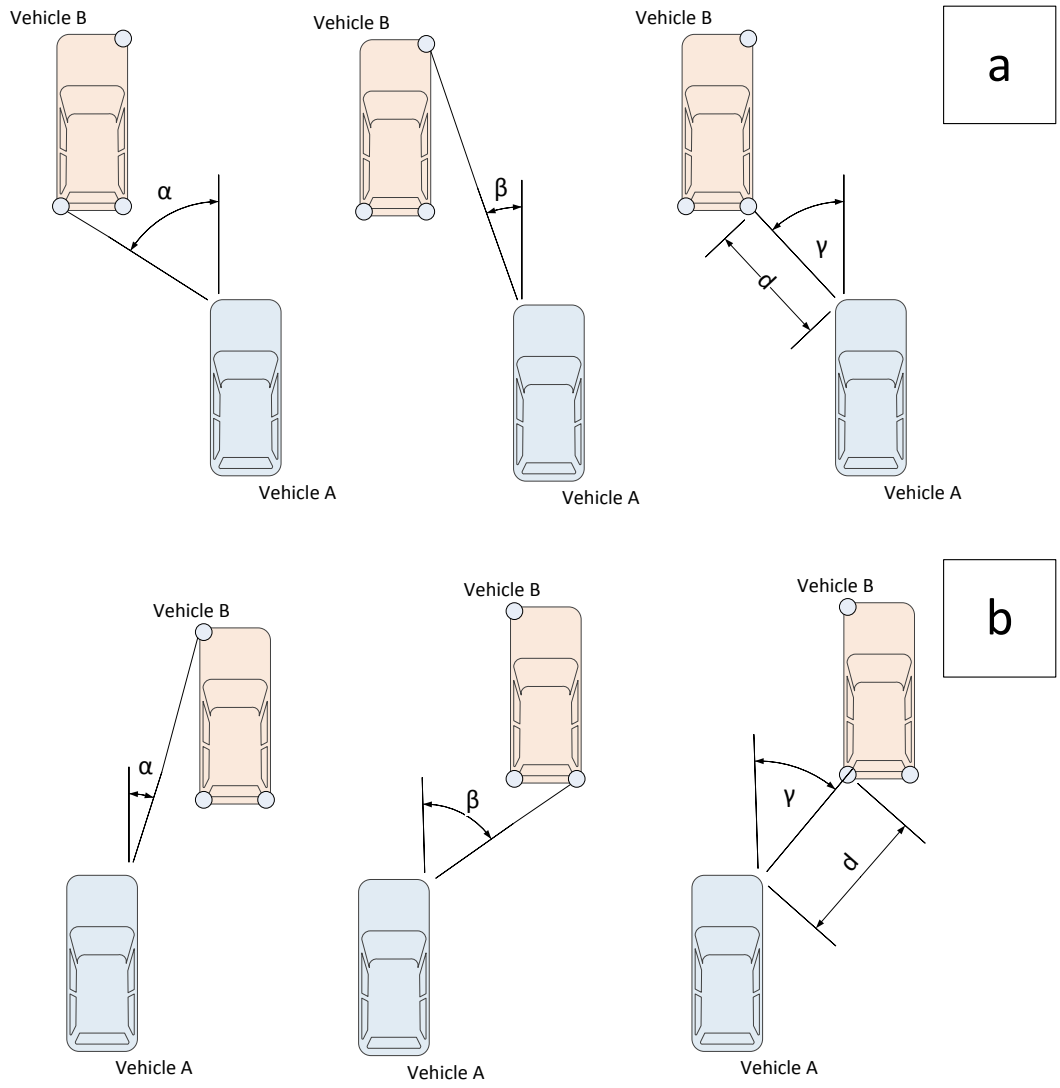


Figure 5. 25 Illustration of angles *α,β,γ* for the estimation of TTC

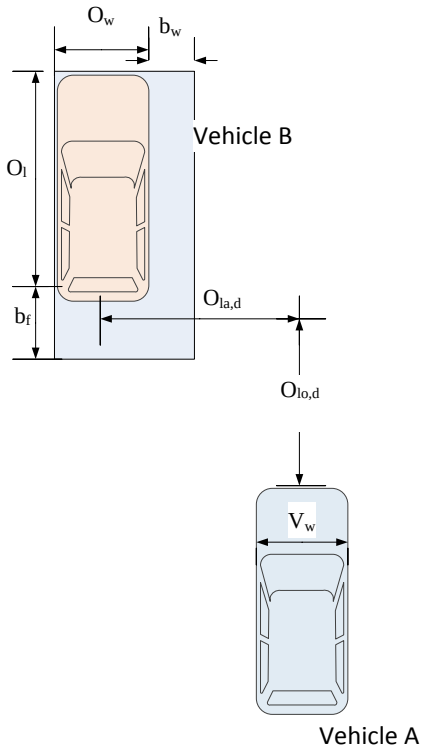


Figure 5. 26 Illustration of the required measurements to estimate the looming angles

If a vehicle were on the left-hand side of the ego-vehicle then the following formulas were used to estimate the angles α,β and γ as well as the distance d according to simple trigonometric and geometrical rules:

$$\alpha_{\mathrm{n,t}} = \tan^{-1}\left(\frac{\left(\frac{1}{2}o_{w_{n,t}}\right)+\left|o_{la,d_{n,t}}\right|-\frac{1}{2}V_w}{o_{lo,d_{n,t}}-b_f}\right) \tag{5.4}$$

$$\beta_{\mathrm{n,t}} = \tan^{-1}\left(\frac{\left(-\frac{1}{2}o_{w_{n,t}}-b_{\mathrm{w}}\right)+\left|o_{la,d_{n,t}}\right|-\frac{1}{2}V_w}{o_{lo,d_{n,t}}+o_{l_{n,t}}}\right) \tag{5.5}$$

$$\gamma_{\mathrm{n,t}} = \tan^{-1}\left(\frac{\left(-\frac{1}{2}o_{w_{n,t}}-b_{\mathrm{w}}\right)+\left|o_{la,d_{n,t}}\right|-\frac{1}{2}V_w}{o_{lo,d_{n,t}}-b_f}\right) \tag{5.6}$$

$$\mathrm{d}_{\mathrm{n,t}} = \sqrt{(o_{lo,d_{n,t}}-b_f)^2+(\left|o_{la,d_{n,t}}\right|-\frac{1}{2}V_w-\frac{1}{2}o_{w_{n,t}}-b_{\mathrm{w}})^2} \tag{5.7}$$

where n denotes the vehicle ID and t is a specific time moment.

Accordingly, if a vehicle is on the right-hand side of the ego-vehicle the following formulas are utilised:

$$\alpha_{\mathrm{n,t}} = \tan^{-1}\left(\frac{\left(-\frac{1}{2}o_{w_{n,t}}-b_{\mathrm{w}}\right)+\left|o_{la,d_{n,t}}\right|-\frac{1}{2}V_w}{o_{lo,d_{n,t}}+o_{l_{n,t}}}\right) \tag{5.8}$$

$$\beta_{\mathrm{n,t}} = \tan^{-1}\left(\frac{\left(\frac{1}{2}o_{w_{n,t}}\right)+\left|o_{la,d_{n,t}}\right|-\frac{1}{2}V_w}{o_{lo,d_{n,t}}-b_f}\right) \tag{5.9}$$

$$\gamma_{\mathrm{n,t}} = \tan^{-1}\left(\frac{\left(-\frac{1}{2}o_{w_{n,t}} - b_{\mathrm{w}}\right) + \left|o_{la,d_{n,t}}\right| - \frac{1}{2}V_w}{o_{lo,d_{n,t}} - b_f}\right) \quad (5.10)$$

$$\mathrm{d}_{\mathrm{n,t}} = \sqrt{(o_{lo,d_{n,t}} - b_f)^2 + (\left|o_{la,d_{n,t}}\right| - \frac{1}{2}V_w - \frac{1}{2}o_{w_{n,t}} - b_{\mathrm{w}})^2} \quad (5.11)$$

The loom angle θ was then calculated by subtracting α from β:

$$\theta_{\mathrm{n,t}} = \left|\alpha_{\mathrm{n,t}} - \beta_{\mathrm{n,t}}\right| \quad (5.12)$$

TTC for every vehicle was estimated if the loom angle θ is increasing over time and one of the following conditions were true:

1. The bearing angle is decreasing while the ego-vehicle is driving on a curve
2. The bearing angle is constant while the ego-vehicle is driving in a straight line
3. The bearing angle is decreasing while the yaw rate is decreasing to 0
4. The bearing angle is first increasing and then decreasing or is staying constant while the yaw rate is increasing from 0 to a higher value.

The estimation of TTC was performed by taking into account the relative acceleration between the ego-vehicle and a vehicle *n*. The equations are similar to the ones used by Brown, (2005), Ozbay et al., (2008), Saffarzadeh et al., (2013) and Ward et al., (2015), however the definition of relative velocity and acceleration is different than the definition used in this PhD thesis.

In this research, relative velocity is simply defined as $\Delta v = |v_l - v_f|$ where v_l is the speed of the leading vehicle and v_f is the speed of the following vehicle. Accordingly, relative acceleration is simply defined as $\Delta a = |\alpha_l - \alpha_f|$

If the relative acceleration between vehicle *n* and the ego-vehicle is non-zero then TTC is estimated as:

$$TTC_{n-ego} = \min\left(\frac{\Delta \mathrm{v} + \sqrt{\Delta \mathrm{v}^2 + \Delta\alpha \mathrm{d}}}{\Delta\alpha}, \frac{\Delta \mathrm{v} - \sqrt{\Delta \mathrm{v}^2 + \Delta\alpha \mathrm{d}}}{\Delta\alpha}\right) \quad (5.13)$$

Otherwise if $\Delta\alpha = 0$ and $\Delta v < 0$: $TTC_{n-ego} = -\frac{d}{\Delta \mathrm{v}}$ (5.14)

Where, Δa is the relative acceleration between the ego-vehicle and vehicle *n*, Δv is the relative velocity between the ego-vehicle and the target-object *n*, d is the distance between the loom-point (i.e. the furthest right or left point as depicted in Figure 5.25) on the ego-vehicle and the closest point of the vehicle *n*.

To summarise, the above described algorithm estimates TTC in car-following situations, taking into account the acceleration of vehicles which is advantageous for safety applications (Ward et al., 2015). It also utilises a safety buffer area around every obstacle which considers the size of the obstacle/vehicle, accounts for measurement inaccuracies and allows for safe vehicle interactions with the ego-AV(Hou et al., 2014).

Figure 5.27 gives the distribution of TTC values for the motorway driving data collection trip used for the purposes of this PhD research project.

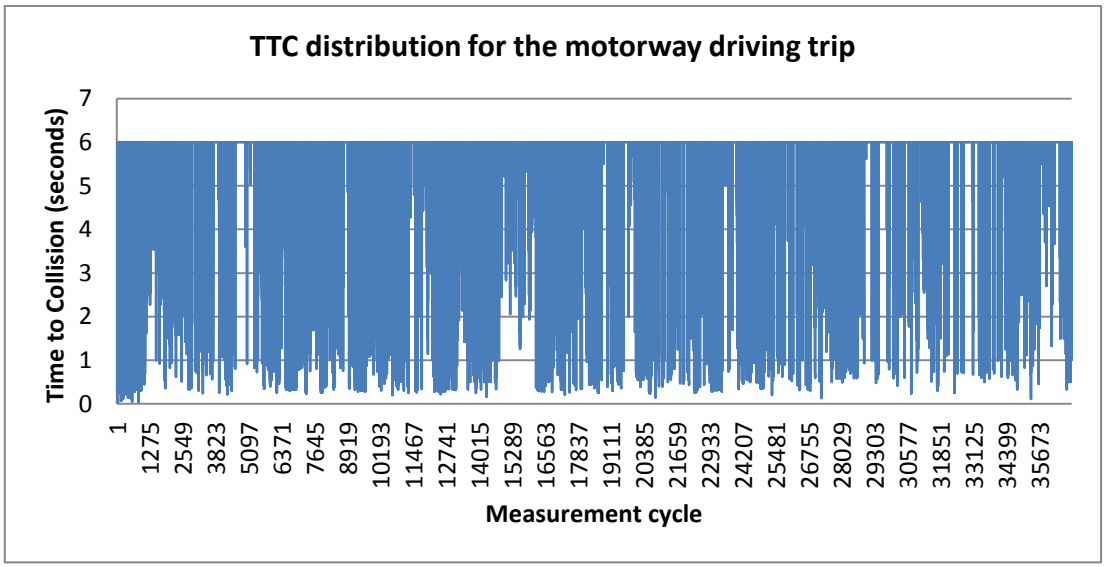


Figure 5. 27 Illustration of the TTC distribution for the motorway driving data collection trip

5.4.4. *Limitations of the vehicle-level dataset*

The dataset provided by the radar sensor of the Loughborough University instrumented vehicle provides frequent and robust measurements. These measurements in conjunction with the measurements from the camera system and the

lane departure and collision warning system could result in a rich dataset for vehicle motion prediction and risk assessment. However, for the purpose of this PhD only the radar sensor measurements were utilised. Moreover, only TTC was estimated from the radar data in order to distinguish between safe and "hazardous" road users. Ideally, the vehicle-level data could be obtained by several driving trips, and the characteristics of safe and road users could be classified according to trajectory features and more sophisticated metrics than the simplistic TTC used in this thesis.

5.5.Summary

This chapter presented the datasets that will be utilised to develop the classification models for NLCP, as well as vehicle-level data coming from the sensors of an instrumented vehicle.

Three different datasets were deployed for the estimation of the network-level risk. The first one comprises of highly aggregated traffic data in 15-minute intervals and the corresponding collision data from two motorways and two A-roads included in the Strategic Road Network (SRN) of England. The second dataset contains disaggregated traffic data just 5 minutes before collision events having occurred at two major roads within the metropolitan area of Athens, Greece. The use of such disaggregated data will enhance the development of NLCP models with lower prediction horizon in order for them to be utilised in AV risk assessment modules. The Athens dataset contains 472 collision-cases and 917 non-collision cases. The third dataset was obtained by using the PTV VISSIM microsimulation software and the Surrogate Safety Assessment Model (SSAM). Highly aggregated traffic data from the first dataset (i.e. from M62) were used to calibrate and validate the simulation and conflict data were retrieved from SSAM. After the calibration and validation of the microsimulation the traffic data were aggregated in 30-second, 1-minute, 3-minute and 5-minute intervals so as to investigate the effect of temporal aggregation on the classification results. The traffic conditions were then matched with the conflict events. For every simulation run, three additional runs were conducted to obtain non-conflict traffic data. This procedure was repeated a further three times to provide a larger dataset which is essential for classification purposes.

The final dataset contains 7800 conflicts and the corresponding 23400 non-conflict cases.

Finally, vehicle-level data were obtained from the Loughborough University instrumented vehicle. The dataset was collected during a one-hour driving trip on the M1 motorway (J23-J18). TTC values were estimated by only using the radar data from the trip. These values will be used in order to identify potential safe and dangerous road users. This will enhance the analysis in order to estimate the impact that network-level collision information will have on the identification of hazardous traffic participants.

6. Network-level collision prediction results

6.1.Introduction

This chapter presents the classification results for the NLCP models developed in the methodology chapter using both real-world (from UK and Greece) and simulated datasets. As different models are going to be investigated, a comparison between methods and results is carried out. Furthermore, some insights for identifying the optimal temporal resolution of traffic data will be offered so as to be employed in a NLCP model.

Initially, the potential of RVMs for classification of 15-minute traffic data from the UK Strategic Road Network (SRN) is going to be tested. This is followed by an analytic comparison of machine learning classifiers (i.e. RVMs, SVMs, kNN, GPs and RFs), utilised for classifying motorway traffic data into *collision-prone* and *safe*. After comparing these classifiers, imbalanced learning techniques are implemented to quantify the difference in the classification results.

Following the classification results for the real-world data, the same classification algorithms are tested on a highly disaggregated dataset, obtained from simulating traffic on the M62 motorway in the UK. Consequently, traffic conditions which potentially cause conflicts in traffic microsimulation models could be identified through these models.

6.2. RVMs in NLCP

This section investigates if RVMs could be utilised in predicting collision-prone traffic in real-time. For that purpose, the traffic and collision data from J10-J13 of the M1 Motorway (430 collision and non-collision cases) and the AL634 link of the A3 road (119 collision and non-collision cases) were collected and utilised as the training datasets, while the validation dataset was obtained from part of A12 road (105 collision and non-collision cases). Two key explanatory variables - average speed (km/h) and traffic flow were included in the classifiers whereas average travel time was omitted because it was not considered as an important indicator for predicting collision-prone conditions.

6.2.1. *Results for the preliminary dataset*

RVMs and SVMs classification methods have been applied to the datasets in order to solve the binary classification problem in distinguishing between *safe* and *collision*-prone conditions.

As mentioned before, both SVMs and RVMs rely on kernel functions to perform regression or classification. The most popular kernels used are the linear, polynomial and Gaussian or radial basis function (RBF). In this study the Gaussian kernels have been used, as they provide more powerful results (e.g. Yu and Abdel-Aty, 2013).

The Gaussian kernel is calculated through the equation $K(x_i, x_j) = \exp\left(-\gamma\|x_i - x_j\|^2\right)$, where γ determines the width of the basis function. The coefficient γ was set to 0.5 as obtained by an optimization technique in MATLAB (Statistics and Machine Learning Toolbox™) (Mathworks, 2016) .

In order to test the performance of RVMs for classifying traffic conditions, two MATLAB implementation algorithms were employed, namely SparseBayes v1 and v2 (Tipping, 2009, Michael E. Tipping, 2009). Although both algorithms perform the same task, the difference lies in the fact that v1 has a built-in function to develop RVMs, while the second version is more 'general-purpose' and requires that the user defines the basis functions to be used. Furthermore, the hyperparameters α_i are updated in v1 at each iteration using the formula $\alpha_i = \frac{\gamma_i}{\mu_i{}^2}$, where μ_i is the i-th posterior mean weight and $\gamma_i \equiv 1 - \alpha_i \Sigma_{ii}$ with Σ_{ii}being the i-th diagonal element of the posterior weight covariance used. This update technique, although simplistic, is not the most optimal (Tipping, 2009). On the contrary, the marginal likelihood function with regards to the hyperparameters is efficiently optimised continuously in v2 and individual basis functions can be discretely added or deleted as described in Tipping and Faul (2003). In that way, algorithm v2 converges faster but can prove greedy with respect to the classification results.

For the RVM models the maximum iterations were set to 100,000 with monitoring at every 10 iterations, the Gaussian kernel width was set to 0.5 and the initial β value

was set to zero. The first version of the RVM algorithm was initialised with $a = \frac{1}{N^2}$, where N is the size of the dataset. The algorithm terminates if the largest change in the logarithm of any hyperparameter α is less than 10^{-3}. On the other hand, the second version of RVM initialises with a a value which is automatically calibrated according to the size of the dataset used. The v2 algorithm terminates if the change in the logarithm of any hyperparameter α is less than 10^{-3} and the change in the logarithm of β parameter is less than 10^{-6}. SVMs were developed using the Statistics and Machine Learning Toolbox™ of MATLAB, with the Gaussian kernel width of 0.5 and the Box constraint level set of 1. The linear kernel for SVMs ($K(x_i, x_j) = x_i \cdot x_j$), is also tested for comparison reasons.

In order to test the performance of the three different algorithms (i.e. RVM_v1, RVM_v2 and SVMs), two criteria were used:

i) the classification error
ii) the decision vectors used during the training of the model, as well as when tested using the validation dataset

The training datasets consist of the 430 traffic conditions of M1 (J10-J13) and the 119 traffic conditions from link AL634 of A3 road. These training and validation datasets are relatively small. However, collision occurrence is a rare event and it is not unusual for traffic safety experts to deal with small samples (Yu and Abdel-Aty, 2013b). Furthermore, other studies on RVM classification such as Demir and Ertürk (2007) have tested training datasets of the same sample size.

Results for the training datasets are summarised in Table 6.1, while results for the validation dataset are summarised in Table 6.2.

Table 6. 1 Classification Accuracy during Training and Number of Decision Vectors for RVMs and SVMs

Method	Kernel	Training Sample Size	Training Error	Decision Vectors
RVM_v1	Gaussian	430	4.88%	358
RVM_v2	Gaussian		19.53%	93
SVM	Gaussian		15.80%	204
	Linear		20.00%	203
RVM_v1	Gaussian	120	0.84%	116
RVM_v2	Gaussian		18.49%	6
SVM	Gaussian		11.80%	6
	Linear		15.1%	42

As can be seen from Table 6.1., training for RVMs is slower than that of SVMs and this is in-line with the results of other RVM classification algorithms (Tipping, 2001, Demir and Ertürk, 2007). The delay in training for RVMs is triggered by the iterated need for calculating and inverting the Hessian matrix and which leads to more computational time as sample size increases. The best classification is performed by the RVM_v1 algorithm, with a large margin (of about 10%) to the next more successful algorithm which is the SVM with a Gaussian kernel. However, it is noticeable that this successful rate of classification by the RVM_v1 algorithm is due to the large number of decision vectors, which is about 1.5 to 2 times higher than the decision vectors used by RVM_v2 and SVM. The efficient RVM_v2 is about 4% less accurate than SVMs, however the interesting fact is that it uses less than a half of the decision vectors utilised by SVMs to perform the training classification. This classification is also performed in a non-critical time interval which can be utilised in real-time (i.e. 8 seconds). In the smaller sample size, it can also be seen that RVM_v2 uses only 6 vectors to perform the classification, while the other two approaches require a much larger number. Comparing training classification results between the small and the bigger sample size, it can be seen that all three algorithms perform better on the small sample, which also agrees with the literature (Phillips et al., 2011, Demir and Ertürk, 2007). Training time for SVMs is notably faster because of the fact that the RVMs learning algorithm is more computationally complex (Tipping, 2001).

Table 6. 2 Validation results of the algorithms using an independent sample

Datasets with sample size	Method	Kernel	Classification error (%)
Training dataset: 430 cases from a motorway; Validation dataset: 110 observations from A-class roads	RVM_v1	Gaussian	25.71
	RVM_v2	Gaussian	20
	SVM	Gaussian	18.09
		Linear	20
Training dataset: 120 cases from A-class road; Validation dataset: 110 observations from A-class roads	RVM_v1	Gaussian	21.9
	RVM_v2	Gaussian	20
	SVM	Gaussian	10.47
		Linear	20

Looking at the validation results of the classification algorithm that was trained with the larger sample (Table 6.2), it is shown that RVM_v1 is no longer the most accurate classifier. SVMs with Gaussian kernel produce the most successful result, followed by RVM_v2 and SVM with a linear kernel. RVM_v1 probably leads to worse results due to the fact that it requires a lot of decision vectors and these classifier vectors cannot perform well when applied to an unknown independent dataset. When the classifier was trained with a small sample and the algorithm was applied to a relatively small sample, the results show that SVMs with Gaussian kernel outperform RVMs with a classification error which is a half of the classification error found in each of the RVMs algorithms.

To further investigate the classification performance of the three algorithms[14], the measures of sensitivity and specificity were employed. For that purpose, four commonly employed terms as defined below are employed.

- True Positive (TP): Dangerous (collision-prone) conditions (t_{real}=1) correctly identified as dangerous ($t_{classified}$=1)
- False Positive (FP): Dangerous (collision-prone) conditions (t_{real}=1) incorrectly identified as safe ($t_{classified}$=0)

[14] SVM with linear kernel was excluded in Table 6.3 because it did not provide better results than the other algorithms

- True Negative (TN): Safe traffic conditions (t_{real}=0) correctly identified as safe ($t_{classified}$=0)
- False Negative (FN): Safe traffic conditions (t_{real}=0) incorrectly identified as dangerous ($t_{classified}$=1)

By making use of the known formula for sensitivity and specificity (Powers, 2011), the performance of these algorithms for the larger training dataset and the validation dataset are presented in Table 6.3:

$$Sensitivity = \frac{TP}{TP+FN}, Specificity = \frac{TN}{TN+FP} \quad (6.1)$$

Table 6. 3 Sensitivity and Specificity of RVMs and SVMs

Method	Kernel	Sensitivity (%)	Specificity (%)
RVM_v1	Gaussian	88.4	91.5
RVM_v2	Gaussian	66.67	80.26
SVM	Gaussian	73.8	84.6

It is noticeable from Table 6.3 that the RVM_v1 algorithm performs well in identifying the traffic conditions that lead to a collision. RVM_v1 is the best classifier with a 88.4% sensitivity and a 91.5% specificity implying minimum Type I and Type II errors. The RVM_v2 algorithm underperforms in terms of sensitivity and specificity between the two datasets. This is probably a result of the greediness of the algorithm which converges fast but at the expense of a large number of false positives.

The reason for these misclassification rates associated with all of the algorithms, especially with RVM may relate to the use of highly aggregated (i.e. 15-minute) traffic data, relatively small sample size and the use of only two variables (i.e. average speed and traffic flow) for representing traffic. In addition, the algorithm was primarily trained with traffic data from a motorway (M1 J10 – J13) but validated with traffic data from A-class roads. Traffic dynamics between these two classes of roads are quite different from each other.

6.3.Classification results for M1 and M62

As RVMs proved to perform well compared to SVMs, it was chosen to re-check its performance in the dataset containing only motorway traffic from M1 and M62 along with other classifiers. The dataset contains average 15-minute traffic speed and flow as described in section 6.2.

The algorithms tested were SVMs and NNs which have been previously applied to NLCP, RVMs and GPs (in order to obtain probabilistic predictions), RFs (an ensemble powerful classifier) and kNN (a simple data-driven classifier).

Before the initiation of each algorithm, an optimization routine was run along with 10-fold cross-validation in order to find the optimal parameters for each algorithm. In order to avoid over-fitting and assure optimal results, 2/3 of the dataset were used for training the classifiers and 1/3 of the dataset was used for testing the classification results. The models were developed in Python 2.7 using the scikit-learn (Pedregosa et al., 2012) and the sklearn-bayes (Shaumyan, 2016) packages.

6.3.1. *Results for the joined dataset*

Tables 6.4 and 6.5 present the confusion matrix and the classification metrics for the dataset containing traffic from M1 and M62. This is followed by Figure 6.1 which depicts the ROC curve for all the estimated classifiers.

Table 6. 4 Confusion matrix for the dataset utilizing traffic from M1 and M62

Classifier	TN	FP	FN	TP	Sum
kNN	*279*	*4*	*77*	*1*	*361*
RVM	*283*	*0*	*78*	*0*	*361*
SVM	*283*	*0*	*78*	*0*	*361*
GP	*283*	*0*	*78*	*0*	*361*
RF	*283*	*0*	*74*	*4*	*361*
NN	*283*	*0*	*78*	*0*	*361*

Table 6. 5 Classification metrics for the dataset utilizing traffic from M1 and M62

Algorithm	Accuracy	Precision	Recall	Specificity	f1-score	G-Means
kNN	0.775623	0.2	0.012821	0.985866	0.024096	0.050637
RVM	0.783934	NA	0	1	NA	NA
SVM	0.783934	NA	0	1	NA	NA
GP	0.783934	NA	0	1	NA	NA
RF	0.795014	1	0.051282	1	0.097561	0.226455
NN	0.783934	NA	0	1	NA	NA

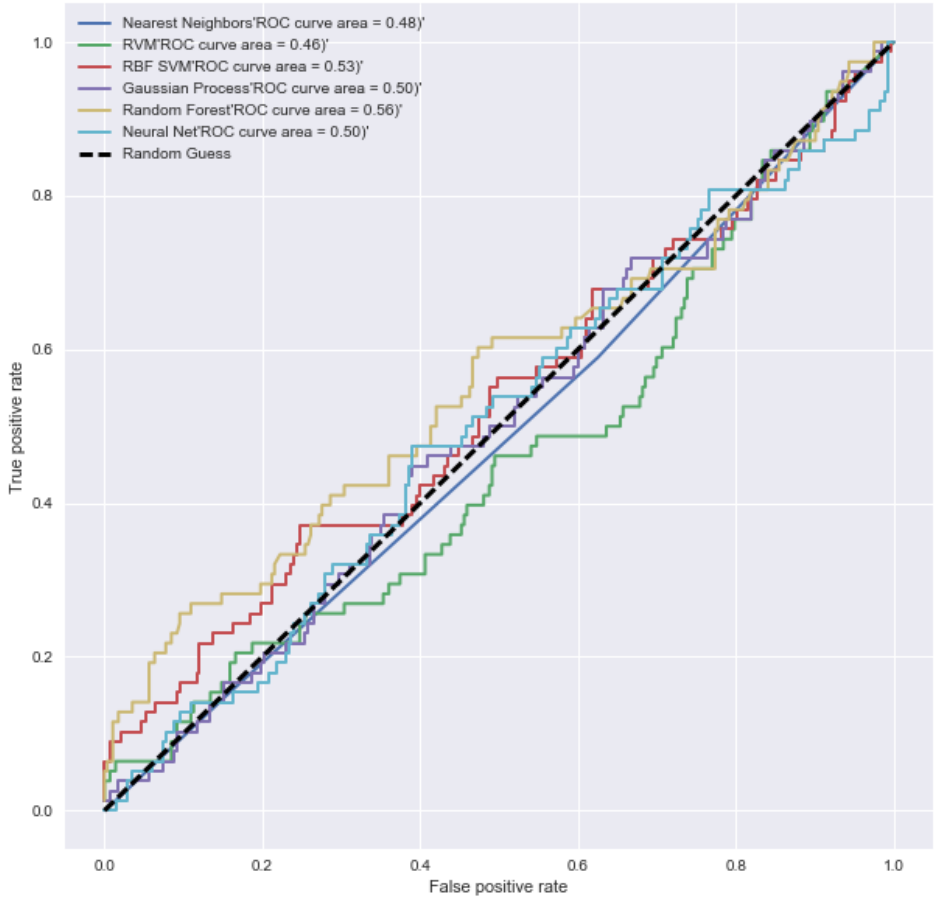


Figure 6. 1 ROC curve of classifiers using 15-minute traffic data from M1 and M62

From Table 6.5 and Figure 6.1 it can be observed that all of the classifiers are close to random guessing, which consequently indicates that their performance is not good. This is probably due to the use of only two predictors (i.e. average speed and flow) that are highly aggregated. The 15-minute traffic data cannot efficiently capture the

traffic dynamics leading to a collision and this is depicted in the classifiers performance. What can also be observed is the superiority of Random Forests in classifying for the case of aggregated data. More specifically, only RFs and kNN could predict at least one collision case, while RVMs, SVMs, GPs and NNs failed. kNN is a data-driven algorithm and, hence, they can easily adapt to every dataset. That is probably the underlying reason for kNN's recognition of some collision-prone instances. However, as the classification results are not promising, the dataset was treated with undersampling and data-cleaning techniques in order to enhance classification performance.

6.3.2. *Classification results with imbalanced learning*

As the prediction presented in section 6.3.1 did not match the expectations (i.e. a high recall with a low false alarm rate) for a successful real-time classifier, imbalanced learning approaches such as undersampling and the integration of oversampling and undersampling were tested for the same dataset. The *imbalanced-learn* package in python offers a variety of undersampling as well as combined (oversampling along with undersampling) techniques. After testing all of them, the best results were given for the Repeated Edited Nearest-Neighbours (RENN) regarding undersampling and the combination of SMOTE and Edited Nearest Neighbours (ENN). The algorithm will be henceforth termed as SMOTE-ENN. The results for these two techniques are presented in the subsequent sections. Each algorithm was trained with the balanced dataset and its performance was tested on the original (imbalanced) dataset. By testing the performance on the original dataset, it is ensured that the validation of the classification results is not based on artificially created instances from SMOTE-ENN or a smaller sample acquired through RENN, but is directly acquired from the original dataset.

6.3.2.1. *Classification results for M1-M62 after undersampling*

Tables 6.6 and 6.7 present the confusion matrix and the classification metrics of the aforementioned six classifiers (i.e. *k*NN, RVMs, SVMs, RFs and NNs), while Figure 6.2 presents the ROC curve of the classifiers.

Table 6. 6 Confusion Matrix of the classifiers using undersampling (RENN)

Classifier	TN	FP	FN	TP	Testing sample size
kNN	468	493	92	149	1202 (961 safe + 241 collision-prone)
RVM	294	667	14	227	
SVM	282	679	6	235	
GP	346	615	19	222	
RF	540	421	80	161	
NN	952	9	239	2	

Table 6. 7 Classification metrics of the classifiers using undersampling (RENN)

Classifier	Accuracy	Precision	Recall	Specificity	f1-score	G-means
kNN	0.5133	0.2321	0.6183	0.4870	0.3375	0.3788
RVM	0.4334	0.2539	0.9419	0.3059	0.4000	0.4890
SVM	0.4301	0.2571	0.9751	0.2934	0.4069	0.5007
GP	0.4725	0.2652	0.9212	0.3600	0.4119	0.4943
RF	0.5832	0.2766	0.6680	0.5619	0.3913	0.4299
NN	0.7937	0.1818	0.0083	0.9906	0.0159	0.0388

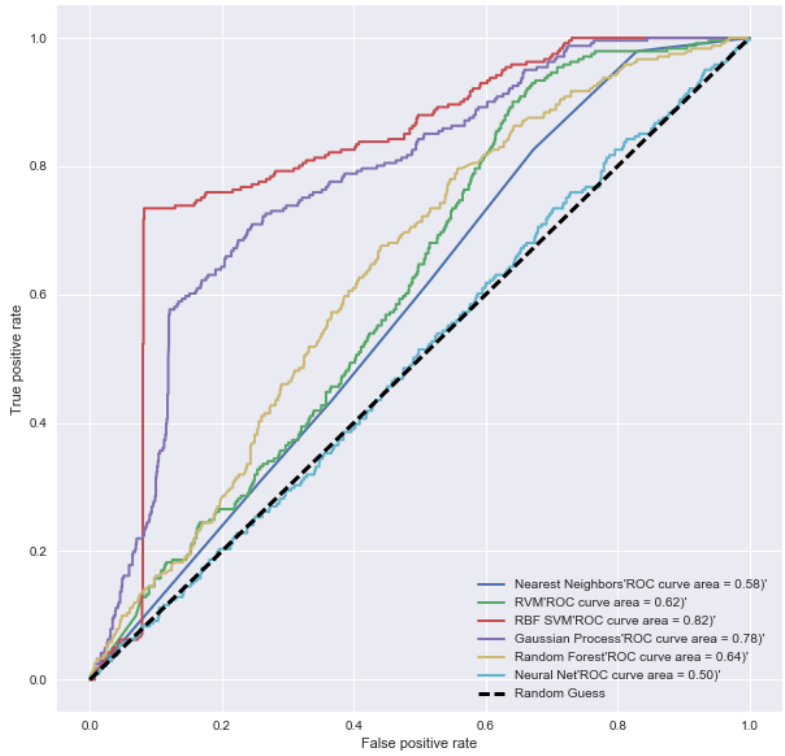


Figure 6. 2 ROC curve of classifiers for the M1-M62 dataset with RENN

From Table 6.6 the improvement of classification results can be easily identified. In the imbalanced dataset, the majority of the classifiers can predict most of the collision-prone conditions successfully. This is further resembled in the classification metrics in Table 6.7. All the classifiers, except NNs, can now identify both *collision-prone* and *safe* traffic much more efficiently. The failure of NNs could be an effect

of the dataset sample size as NNs are usually more powerful when bigger datasets are at hand (Karlaftis and Vlahogianni, 2011). Furthermore, as the G-means metric suggests, this prediction does not favour safe traffic conditions but is weighted sufficiently for both safe and dangerous traffic. Regarding the ROC curve, it is depicted that SVMs, GPs and RFs are the top classifiers while RVMs follow. The superiority of SVMs is once again assured while GPs and RVMs which can provide probabilistic predictions along with RF as an ensemble method show a good performance.

6.3.2.2. Classification results for M1 and M62 after oversampling integrated with undersampling.

Tables 6.8 and 6.9 present the confusion matrix and the classification metrics for the studied classifiers while Figure 6.3 presents the ROC curve of the classifiers after the use of oversampling along with undersampling. As mentioned previously, the algorithm which produced the best results were obtained using SMOTE-ENN.

Table 6. 8 Confusion Matrix of the classifiers using SMOTE-ENN

Classifier	TN	FP	FN	TP	Testing sample size
kNN	877	84	158	83	1202 (961 safe + 241 collision-prone)
RVM	942	19	162	79	
SVM	935	26	155	86	
GP	961	0	241	0	
RF	960	1	229	12	
NN	961	0	241	0	

Table 6. 9 Classification metrics of the classifiers using SMOTE-ENN

Classifier	Accuracy	Precision	Recall	Specificity	f1-score	G-Means
kNN	0.7987	0.4970	0.3444	0.9126	0.4069	0.4137
RVM	0.8494	0.8061	0.3278	0.9802	0.4661	0.5141
SVM	0.8494	0.7679	0.3568	0.9729	0.4873	0.5235
GP	0.7995	NA	0.0000	1.0000	NA	NA
RF	0.8087	0.9231	0.0498	0.9990	0.0945	0.2144
NN	0.7995	NA	0.0000	1.0000	NA	NA

Tables 6.8, 6.9 also demonstrate that using oversampling integrated with undersampling improves classification results. Four of the classifiers (i.e. kNN, RVMs, SVMs and RFs) identified safe and collision-prone traffic while the rest failed to identify collision-prone traffic. This is probably due to the failure of GPs to

estimate the prior probability from the SMOTE instances (Elrahman and Abraham, 2013) and the inability of NNs to work with relatively small datasets (Zhu et al., 2006, Karlaftis and Vlahogianni, 2011).

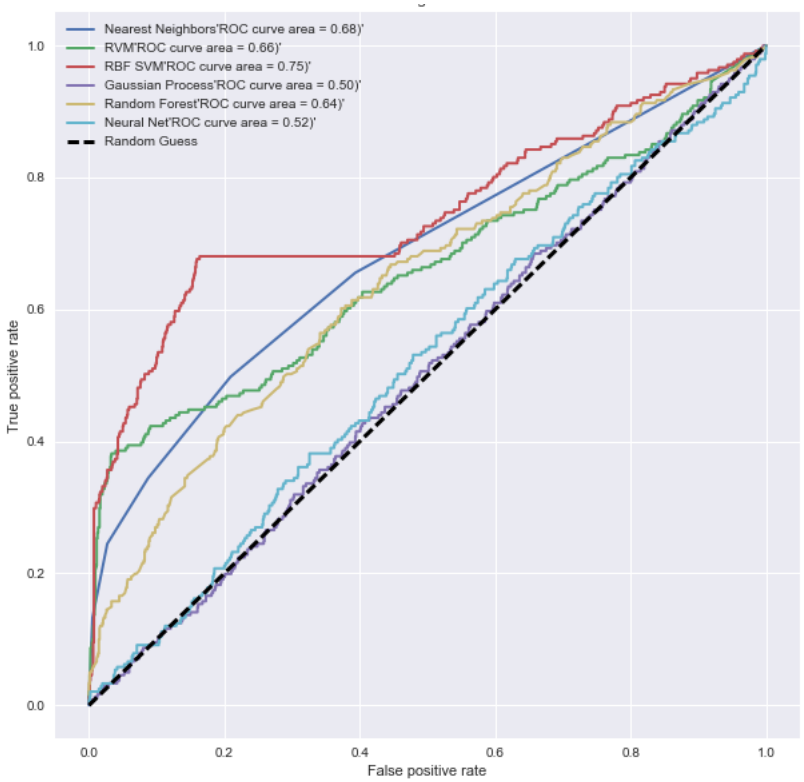


Figure 6. 3 ROC curve of classifiers for the M1-M62 dataset with SMOTE-ENN

From Tables 6.6. – 6.9 it can be observed that most of the classifiers (specifically SVMs, RVMs and RFs) under RENN perform well with respect to the *recall* statistic. Observing the f1-score in the aforementioned tables it is evident that the performance of all the classifiers is relatively poor in identifying relevant instances which is probably due to the poor data quality. On the other hand, it is shown in Tables 6.7 and 6.9 that after the treatment with imbalanced learning the classifiers are able to detect both safe and collision-prone conditions more efficiently.

6.4.Classification results for the Athens dataset

To further explore the capabilities of the classifiers as well as the power of imbalanced learning in the classification performance, the previously tested classifier algorithms and the techniques for under-sampling along with its integration with oversampling were tested for the dataset containing more disaggregated (i.e. 5-minute) traffic data from Athens, Greece.

6.4.1. *Results for the original Athens dataset*

Similarly, with the 15-minute dataset from the UK, the performance of the classifiers is evaluated through the confusion matrix (Table 6.10), the classification metrics (Table 6.11) and the ROC curve (Figure 6.5).

Table 6. 10 Confusion Matrix of the classifiers using 5-minute traffic data from Athens, Greece

Classifier	TN	FP	FN	TP	Test Sample size
kNN	249	53	129	28	459 (302 safe and 157 collision-prone)
RVM	302	0	157	0	
SVM	302	0	157	0	
GP	290	12	142	15	
RF	290	12	148	9	
NN	273	29	145	12	

From Table 6.10, it is observed that the distinction between safe and collision-prone traffic is troublesome for traffic data aggregated at 5-minute intervals. Safe traffic can easily be identified as the numbers for TN and FP suggest, but hazardous conditions are usually incorrectly classified. The simple data-driven approach of kNN detects the largest number of collision cases while the probabilistic classification of GPs and NNs also performed well compared to the other classifiers.

Table 6. 11 Classification metrics of the classifiers using 5-minute traffic data from Athens, Greece

Classifier	Accuracy	Precision	Recall	Specificity	f1-score	G-Means
kNN	0.6035	0.3457	0.1783	0.8245	0.2353	0.2483
RVM	0.6580	NA	0.0000	1.0000	NA	NA
SVM	0.6580	NA	0.0000	1.0000	NA	NA
GP	0.6645	0.5556	0.0955	0.9603	0.1630	0.2304
RF	0.6492	0.3750	0.0382	0.9669	0.0694	0.1197
NN	0.6558	0.4545	0.0318	0.9801	0.0595	0.1203

Table 6.11 reflects the limited identification of collision-prone conditions, already identified by the confusion matrix (i.e. Table 6.10). Accuracy is generally low in comparison with some existing studies (e.g. Abdel-Aty et al., 2004, Hossain and Muromachi, 2013). Furthermore, the high specificity rates show that safe traffic is conveniently identified, which is additionally observable from the very low rates of recall and f1-score. Precision rates are slightly higher compared to recall and f1-score which shows that whenever an actual collision is detected, it more likely to be a

collision than a false alarm. Nevertheless, the precision rates are still low. RVMs and SVMs could not predict a single collision as shown by Tables 6.10 and 6.11 probably due to their sparsity in decision making and therefore their performance cannot be evaluated properly. Finally, the low G-means metric shows that even though the dataset is quite balanced (i.e. the ratio of safe to collision prone traffic is approximately 2:1) the classification is not.

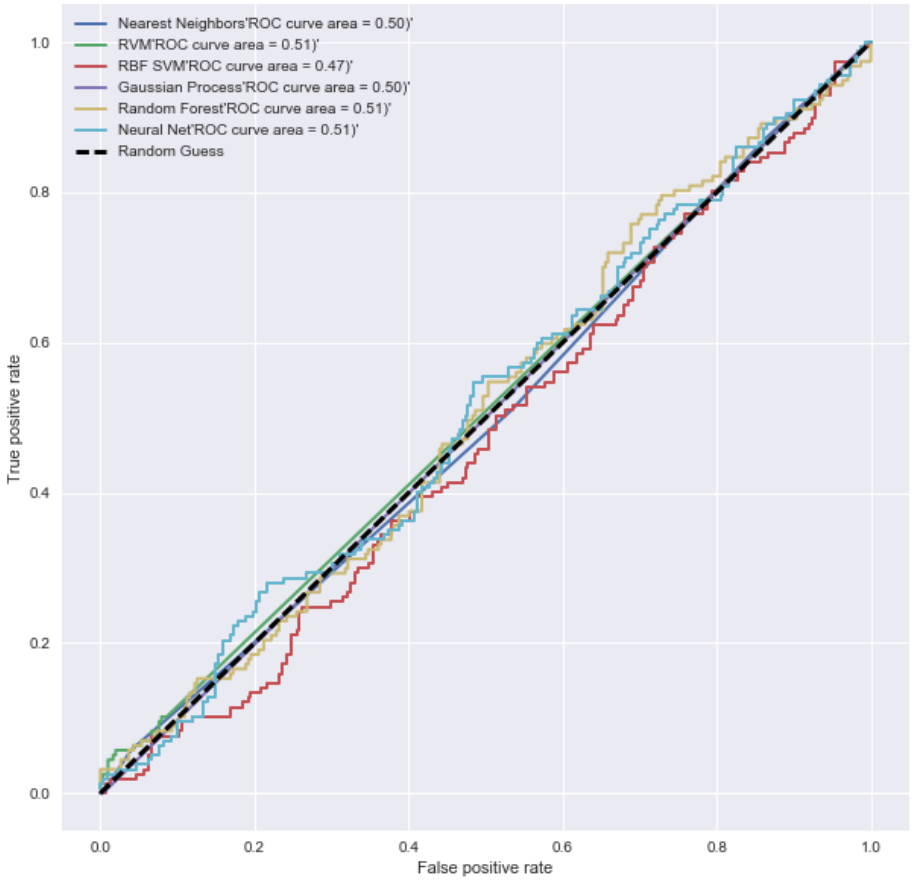


Figure 6. 4 ROC curve of classifiers for the Athens dataset

To illustrate the classification performance of the classifiers, the ROC curve was utilised as seen in Figure 6.4. From the ROC curve, the ill-defined performance of the classifiers is visible. If the area under the ROC curve is utilised as a performance metric, then NNs and RFs perform the best with 51% which is due to their (relatively) high specificity along with their recall. Overall, however, the classifiers perform similarly or just above the random guess curve which indicates that they cannot be used in safety critical applications.

6.4.2. *Results for the Athens dataset after undersampling*

To improve the classification performance, undersampling is utilised in a similar manner as in the case of the 15-minute traffic dataset from the UK. After comparing different outcomes from the undersampling techniques provided in the *imbalanced learn* package, the best results were acquired for undersampling using the neighbourhood cleaning rule (NC) (Laurikkala, 2001).

Table 6.12 presents the confusion matrix for all the classifiers. This is followed by the classification metrics (Table 6.13) and the ROC curve (Figure 6.5).

Table 6. 12 Confusion Matrix of the classifiers using 5-minute traffic data from Athens, Greece after the treatment with NC

Classifier	*TN*	*FP*	*FN*	*TP*	*Test Sample size*
kNN	444	473	140	332	1389 (917 safe and 472 collision-prone)
RVM	300	617	89	383	
SVM	487	430	75	397	
GP	841	76	147	325	
RF	171	746	38	434	
NN	395	522	170	302	

Comparing Table 6.12 with Table 6.10, the increase in the identification of collision-prone traffic is obvious. On the other hand, this increase is accompanied by the disadvantage of having a large number of false alarms (i.e. FP).

Table 6. 13 Classification metrics for the Athens dataset after the treatment with NC

Classifier	*Accuracy*	*Precision*	*Recall*	*Specificity*	*f1-score*	*G-Means*
kNN	0.5587	0.4124	0.7034	0.4842	0.5200	0.5386
RVM	0.4917	0.3830	0.8114	0.3272	0.5204	0.5575
SVM	0.6364	0.4800	0.8411	0.5311	0.6112	0.6354
GP	0.8395	0.8105	0.6886	0.9171	0.7446	0.7470
RF	0.4356	0.3678	0.9195	0.1865	0.5254	0.5815
NN	0.5018	0.3665	0.6398	0.4308	0.4660	0.4843

The number of false alarms affected the classifiers accuracy which was generally reduced as seen in Table 6.13. Moreover, by observing precision it can be detected that although collision-prone cases are correctly identified the rates are relatively low due to the false alarms. The balanced dataset also induced an increase in false negatives which is resembled in the recall metric. Recall however has significantly

increased compared to the original dataset. Specificity results validate the above statement, as it is demonstrated that identifying safe traffic is no longer an easy task for the classifiers which is probably due to the loss of classification accuracy after the balanced learning. Regarding the f1-score it can be observed that SVMs and GPs perform well in distinguishing between collision and safe-traffic cases while NNs encounter difficulties due to their complexity. The increased G-means shows that all the classifiers perform a much more balanced prediction however NNs and kNN perform the worst due to the large number of false classifications.

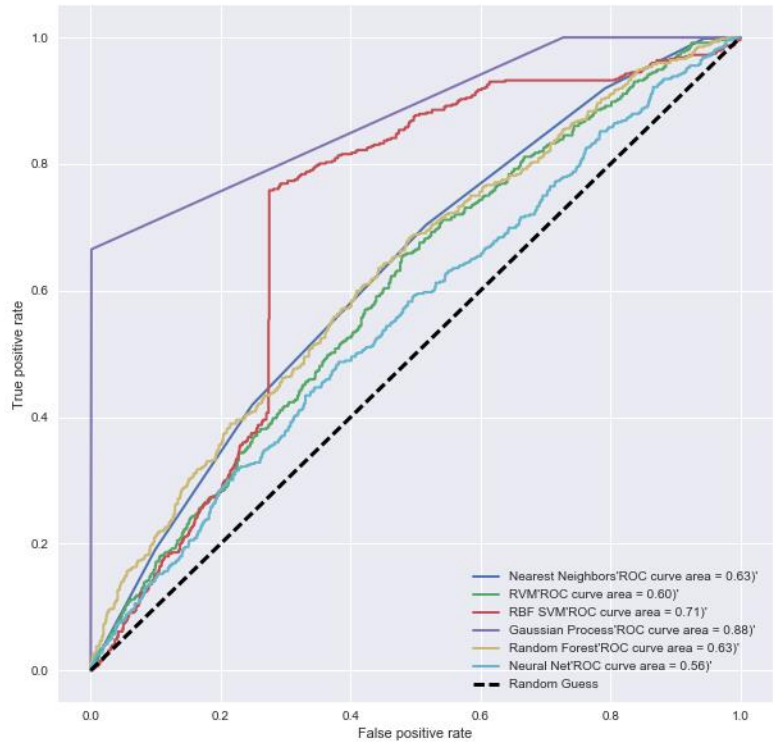


Figure 6. 5 ROC curve of classifiers for the Athens dataset after undersampling

When plotting the false positive rate against the false negative rate in Figure 6.5, the improved performance of all the classifiers is observed. Comparing the classifiers, it is evident that GPs have the better AUC, therefore, assuring best classification results without false alarms. SVMs also adapt well to the dataset while RFs and RVMs perform relatively well to the rest of the classifiers.

6.4.3. *Results for the dataset after oversampling integrated with undersampling*

The treatment of the 5-minute traffic data with undersampling effectively increased the number of correctly identified collision-prone traffic, however the number of

misclassification necessitates the utilization of other solutions. Hence, it was decided that undersampling along with oversampling should be applied. The SMOTE-ENN algorithm produced the best results from the *imbalanced-learn* package and is presented in the following subsections. Tables 6.14, 6.15 and Figure 6.6 summarise the results.

Table 6. 14 Confusion Matrix of the classifiers using 5-minute traffic data from Athens, Greece after undersampling integrated with oversampling

Classifier	TN	FP	FN	TP	Testing sample size
kNN	813	104	345	127	1389 (917 safe + 472 collision-prone)
RVM	917	0	400	72	
SVM	917	0	369	103	
GP	910	7	333	139	
RF	915	2	448	24	
NN	817	100	399	73	

After applying SMOTE along with ENN, the classification results are not significantly enhanced compared to the original dataset. GPs, SVMs and kNN perform better than the rest of the classifiers in identifying hazardous traffic, however a large number of false negatives is obvious for all algorithms.

Table 6. 15 Classification metrics for the Athens dataset after the treatment with SMOTE-ENN

Classifier	Accuracy	Precision	Recall	Specificity	f1-score	G-Means
kNN	0.6767	0.5498	0.2691	0.8866	0.3613	0.3846
RVM	0.7120	1.0000	0.1525	1.0000	0.2647	0.3906
SVM	0.7343	1.0000	0.2182	1.0000	0.3583	0.4671
GP	0.7552	0.9521	0.2945	0.9924	0.4498	0.5295
RF	0.6760	0.9231	0.0508	0.9978	0.0964	0.2166
NN	0.6407	0.4220	0.1547	0.8909	0.2264	0.2555

By comparing Table 6.15 with Tables 6.11 and 6.13, it is clear that the overall accuracy for all the classifiers is improved compared to the original dataset but does not achieve such positive results as the dataset treated with NC. The low metrics of recall and f1-score along with the high numbers of specificity demonstrate that the classifiers and especially GPs and SVMs, can distinguish safe traffic cases easily but fail to identify the majority of conflict-prone traffic. Observing the G-means metric further justifies the enhanced balanced classification of SVMs and GPs. The figures

on precision for most classifiers, additionally demonstrate that when a collision-prone case is detected, it is most likely for it to be a real collision case rather than a false alarm. On the other hand, the classifiers show high precision and low recall and hence the results are not credible enough.

From Tables 6.11, 6.13 and 6.15 it can be concluded that the best results for all the classifiers are achieved using the NC technique. As the balanced classification of the original dataset is small, undersampling the majority class along with oversampling the minority class is not able to solve the classification problems. However, when only undersampling the minority class is applied then the learned classifiers perform better. Through looking at the overall classification performance of the classifiers as given by the f1-score it is shown that GPs and SVMs perform better when used with undersampling. The same algorithms are the ones that provide the most balanced classification as seen from the G-means metric. GPs along with SVMs also achieve the best classification scores in the original datasets however as explained in the previous subsections these results are not reliable because of the generally poor performance of all the classifiers for the original dataset.

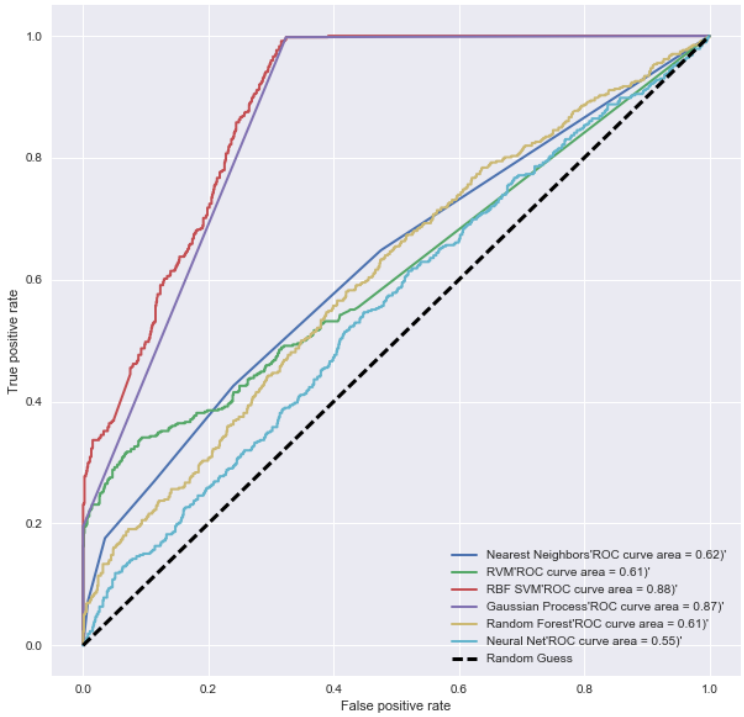


Figure 6. 6 ROC curve of classifiers for the Athens dataset after SMOTE-ENN

The prevalence of GPs and SVMs for the classification of collision cases is obvious when observing the ROC curves and the area under them for the dataset obtained from SMOTE-ENN. kNN, although simple as an algorithm, is shown to perform well achieving a high AUC percent while RVMs and RFs fall behind due to the large number of false negatives as seen in Tables 6.14 and 6.15. Finally, the poor performance of NNs is further validated as it does not achieve a high AUC rate compared to the other classifiers.

6.5.Comparison of classifiers using real-world data with literature

To further validate the performance of the classifiers and techniques presented in this chapter, a comparison is provided with results from the literature. A brief review of performance comparison between classification approaches in recent literature demonstrated that the most important parameters in a NLCP model are recall and false alarm rate. Table 6.16 summarises the prediction performance of previous literature along with the best classifiers developed in this chapter.

Table 6. 16 Recall and false-alarm rate of classifiers in the literature and the best of the developed classifiers

Previous literature	Classification method	Recall	False Alarm Rate
Abdel-Aty (2004)	Logistic Regression	0.69	N/A
Pande and Abdel-Aty (2006)	NN	0.57	0.29
Abdel-Aty (2008)	NN	0.61	0.21
Hossain and Muromachi (2012)	Bayesian Network	0.66	0.20
Ahmed and Abdel-Aty (2012)	Matched case-control logistic regression	0.68	0.46
Lin et al (2015)	Bayesian Network and kNN	0.61	0.38
Sun and Sun (2015)	Dynamic Bayesian Network	0.76	0.24
Dataset utilised in this thesis	**Classification method**	**Recall**	**False Alarm Rate**
15-minute UK traffic data	RF	0.05	0
15-minute UK traffic data with undersampling	RF	0.67	0.43
5-minute Athens traffic data	GP	0.1	0.23
5-minute Athens traffic data with undersampling	SVM	0.84	0.46
5-minute Athens traffic data with undersampling	GP	0.69	0.08

Table 6.16 demonstrates that the classifiers developed in this Chapter perform equally well or better than the classifiers in the literature. The best recall in

conjunction with low false alarm rate was achieved by Sun and Sun (2015) with 76% and 24% respectively. Although the original datasets of 15-minute traffic data from the UK and 5-minute traffic data from Athens, Greece performed poorly the performance dramatically increased when these data were treated with undersampling of the majority class. For example, GPs could predict 69% of the collision-prone cases in the undersampled Athens dataset with a very small false alarm rate (~8%) while RFs identified correctly most of the hazardous traffic in the UK dataset after the treatment with RENN and performed similarly to the majority of findings in the existing literature. However further research is required to reduce the high rates of false alarms in the undersampled Athens dataset.

6.6.Utilizing microsimulation for real-time conflict prediction

This section discusses the classification results from the models discussed in the methodology chapter using data from a highly disaggregated dataset obtained from simulating traffic on the M62 motorway in the UK. Traffic conditions which potentially cause conflicts in traffic microsimulation models could be identified through these models. Similarly, to the previous chapter, different classifiers are going to be utilised, so that a comparison between methods and results is possible.

The machine learning classifiers (i.e. RVMs, SVMs, kNN, NN and RF), which were utilised for classifying motorway traffic data into collision-prone and safe are going to be employed to identify conflict-prone conditions. After the comparison of classifiers, imbalanced learning techniques will again be implemented in order to critically compare and contrast the results before and after the treatment with imbalanced techniques.

Similar to the previous section, all the classifiers were developed in Python 2.7 using the scikit-learn (Pedregosa et al., 2012), the sklearn-bayes (Shaumyan, 2016) and the imbalanced-learn (Lemaitre et al., 2016) packages. Next, the classifiers were optimised using 10-fold cross validation and 1/3 of the dataset was used for testing to avoid overfitting. For the imbalanced learned classifiers, the testing dataset was the original simulated dataset, so as to quantify the effect of imbalanced learning on the classification performance.

6.6.1. *Results for the original simulation dataset*

Tables 6.17 and 6.18 present the confusion matrix and the classification metrics for the simulation dataset. Figure 6.7 illustrates the ROC curve of the classifiers for every temporal aggregation.

Table 6. 17 Confusion matrix of all the classifiers for the full simulation dataset

Classifier	TN	FP	FN	TP	Testing sample size
30-second data					10296 (2607 conflict cases and 7689 non-conflict cases)
kNN	6981	708	1992	615	
RVM	7689	0	2604	3	
SVM	7607	82	2331	276	
RF	7689	0	2607	0	
NN	7519	170	2105	502	
1-minute data					
kNN	7071	588	2031	606	
RVM	7659	0	2632	5	
SVM	7656	3	2505	132	
RF	7659	0	2637	0	
NN	7263	396	1670	967	
3-minute data					
kNN	7230	492	1670	904	
RVM	7722	0	2567	7	
SVM	7722	0	2431	143	
RF	7501	221	2119	455	
NN	7374	348	1158	1416	
5-minute data					
kNN	7349	348	1463	1136	
RVM	7697	0	2592	7	
SVM	7697	0	2456	143	
RF	7427	270	1726	873	
NN	7621	76	1394	1205	

Table 6. 18 Classification metrics for the simulation dataset

Classifier	Accuracy	Precision	Recall	Specificity	f1-score	G-Means
30-second data						
kNN	0.7378	0.4649	0.2359	0.9079	0.3130	0.3312
RVM	0.7471	1.0000	0.0012	1.0000	0.0023	0.0339
SVM	0.7656	0.7710	0.1059	0.9893	0.1862	0.2857
RF	0.7468	NA	0.0000	1.0000	NA	NA
NN	0.7790	0.7470	0.1926	0.9779	0.3062	0.3793
1-minute data						
kNN	0.7456	0.5075	0.2298	0.9232	0.3164	0.3415
RVM	0.7444	1.0000	0.0019	1.0000	0.0038	0.0435
SVM	0.7564	0.9778	0.0501	0.9996	0.0952	0.2212
RF	0.7439	NA	0.0000	1.0000	NA	NA

NN	0.7993	0.7095	0.3667	0.9483	0.4835	0.5101
3-minute data						
kNN	0.7900	0.6476	0.3512	0.9363	0.4554	0.4769
RVM	0.7507	1.0000	0.0027	1.0000	0.0054	0.0521
SVM	0.7639	1.0000	0.0556	1.0000	0.1053	0.2357
RF	0.7727	0.6731	0.1768	0.9714	0.2800	0.3449
NN	0.8537	0.8027	0.5501	0.9549	0.6528	0.6645
5-minute data						
kNN	0.8241	0.7655	0.4371	0.9548	0.5565	0.5784
RVM	0.7483	1.0000	0.0027	1.0000	0.0054	0.0519
SVM	0.7615	1.0000	0.0550	1.0000	0.1043	0.2346
RF	0.8061	0.7638	0.3359	0.9649	0.4666	0.5065
NN	0.8572	0.9407	0.4636	0.9901	0.6211	0.6604

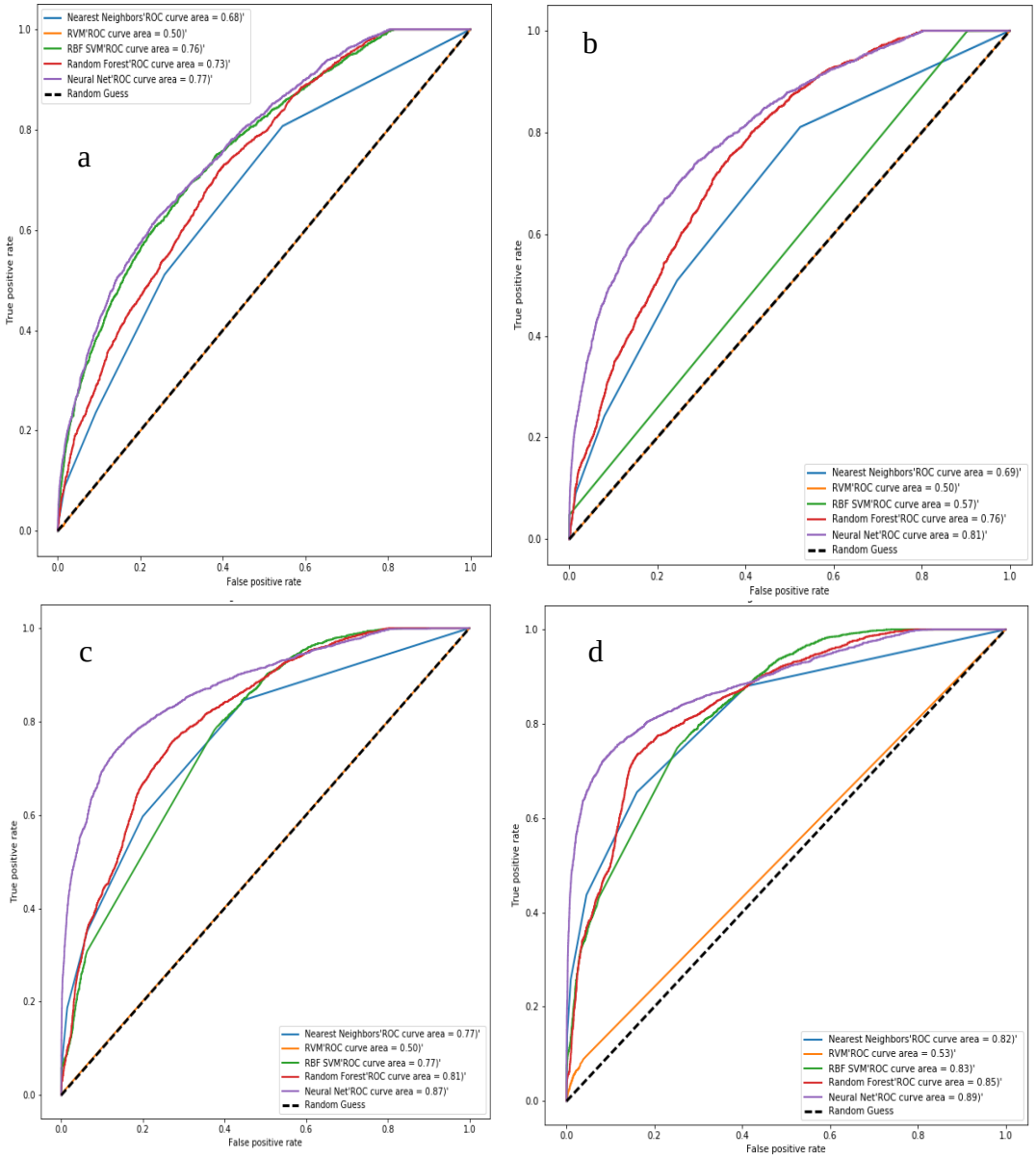


Figure 6. 7 ROC curve of the classifiers for the original simulation dataset (a: 30-second, b: 1-minute, c: 3-minute, d: 5-minute data)

From the confusion matrix (i.e. Table 6.17) it can be observed that the classification results improve with higher temporal aggregation. This is more straightforward for the conflict-prone cases as the number of TP increases when moving from the raw data to 1-minute, 3-minute and 5-minute aggregated traffic. For safe traffic cases, the classifiers in general perform well, as the total number of TN is high, while the number of false alarms (i.e. FP) drops throughout the temporal aggregation intervals.

The performance of the classifiers is further reflected in the classification metrics. From Table 6.18 it can be observed that when using 30-second data, mostly safe traffic conditions are being recognised from the classifiers which results in high specificity. When traffic data aggregation increases, both the conflicts classification accuracy (i.e. recall) and the effective detection of conflict cases (i.e. precision) increase for most of the classifiers. This is further justified by the increase in the f1-score figures, which is the harmonic mean of precision and recall. Regarding balanced classification results, the G-means for most of the classifiers decreases when 1-minute data aggregation is utilised, but increases for higher temporal aggregation promising balanced classification results for both conflict-prone and safe traffic conditions.

Looking at Table 6.18 in more detail it is also evident that the best accuracy is achieved by NNs and kNN when using 5-minute traffic data. The highest precision scores are derived from the RVM models, however when observing recall and specificity, it is shown that this is due to the correct identification of safe traffic only as the recall statistic is zero for RVMs using every temporal aggregation intervals. Moreover, the most effective classification using the f1-score as a criterion is given by NNs using 3-minute or 5-minute aggregation as well as kNN using 5-minute traffic data. The same two algorithms result in the most balanced classification result given by the G-mean metric

By observing Figure 6.7 which illustrates the ROC curves for the classifiers it is noticeable that all the curves move towards the left corner of the diagram which indicated that the overall performance of the classifiers is improved for highly aggregated traffic. Comparing the classifiers, it is shown that NNs have the best

performance throughout the aggregation intervals and justifies its use in several NLCP works (e.g. Pande and Abdel-Aty, 2006). Additionally, RFs perform slightly worse than NNs but indicate that the technique should be utilised in further studies as it performs well when 1-minute, 3-minute and 5-minute aggregated traffic data. Regarding 30-second data, the second-best classifier is SVMs occupying 76% under the ROC curve which indicates its classification power even with highly disaggregated data.

6.6.2. *Results from the simulation dataset with imbalanced learning*

This section presents the classification results for all the temporal aggregation intervals after the datasets have been treated with imbalanced learning techniques to acquire more accurate and balanced results. Similar to the corresponding real-world dataset from M62, the Repeated Edited Nearest Neighbours (RENN) and the SMOTE-ENN techniques yielded the best results. Every temporal aggregation interval was included in a different dataset; hence the classifiers and imbalanced learning techniques were applied to four different datasets. Imbalanced learning aims at producing a more balanced dataset, therefore, the cases used for the analysis were different between datasets.

6.6.2.1.Classification results for the simulated datasets after undersampling

Tables 6.19 and 6.20 present the confusion matrix and the classification metrics for the simulation dataset using RENN. Figure 6.9 illustrates the ROC curve of the classifiers for every temporal aggregation interval.

Table 6. 19 Confusion matrix of all the classifiers for the full simulation dataset under RENN

Classifier	TN	FP	FN	TP	Testing sample size
30-second data					**31200 (23400 non-conflict cases and 7800 conflict cases)**
kNN	13825	9575	1443	6357	
RVM	6725	16675	281	7519	
SVM	12928	10472	1552	6248	
RF	8594	14806	612	7188	
NN	11290	12110	1116	6684	
1-minute data					
kNN	14896	8504	1758	6042	
RVM	23400	0	7589	211	
SVM	23391	9	2429	5371	
RF	10839	12561	923	6877	
NN	14592	8808	2017	5783	
3-minute data					
kNN	16524	6876	1644	6156	
RVM	23400	0	7592	208	
SVM	23398	2	2425	5375	
RF	15657	7743	2218	5582	
NN	15029	8371	1144	6656	
5-minute data					
kNN	17989	5411	1536	6264	
RVM	23400	0	7586	214	
SVM	23400	0	2346	5454	
RF	19204	4196	2671	5129	
NN	18323	5077	1587	6213	

Table 6. 20 Classification metrics for the simulation dataset under RENN

Classifier	Accuracy	Precision	Recall	Specificity	f1-score	G-Means
30-second data						
kNN	0.6469	0.3990	0.8150	0.5908	0.5357	0.5703
RVM	0.4565	0.3108	0.9640	0.2874	0.4700	0.5473
SVM	0.6146	0.3737	0.8010	0.5525	0.5096	0.5471
RF	0.5058	0.3268	0.9215	0.3673	0.4825	0.5488
NN	0.5761	0.3556	0.8569	0.4825	0.5027	0.5521
1-minute data						
kNN	0.6711	0.4154	0.7746	0.6366	0.5408	0.5672
RVM	0.7568	1.0000	0.0271	1.0000	0.0527	0.1645
SVM	0.9219	0.9983	0.6886	0.9996	0.8150	0.8291
RF	0.5678	0.3538	0.8817	0.4632	0.5050	0.5585
NN	0.6530	0.3963	0.7414	0.6236	0.5165	0.5421
3-minute data						
kNN	0.7269	0.4724	0.7892	0.7062	0.5910	0.6106
RVM	0.7567	1.0000	0.0267	1.0000	0.0519	0.1633
SVM	0.9222	0.9996	0.6891	0.9999	0.8158	0.8300
RF	0.6807	0.4189	0.7156	0.6691	0.5285	0.5475
NN	0.6950	0.4429	0.8533	0.6423	0.5832	0.6148
5-minute data						
kNN	0.7773	0.5365	0.8031	0.7688	0.6433	0.6564
RVM	0.7569	1.0000	0.0274	1.0000	0.0534	0.1656
SVM	0.9248	1.0000	0.6992	1.0000	0.8230	0.8362
RF	0.7799	0.5500	0.6576	0.8207	0.5990	0.6014
NN	0.7864	0.5503	0.7965	0.7830	0.6509	0.6621

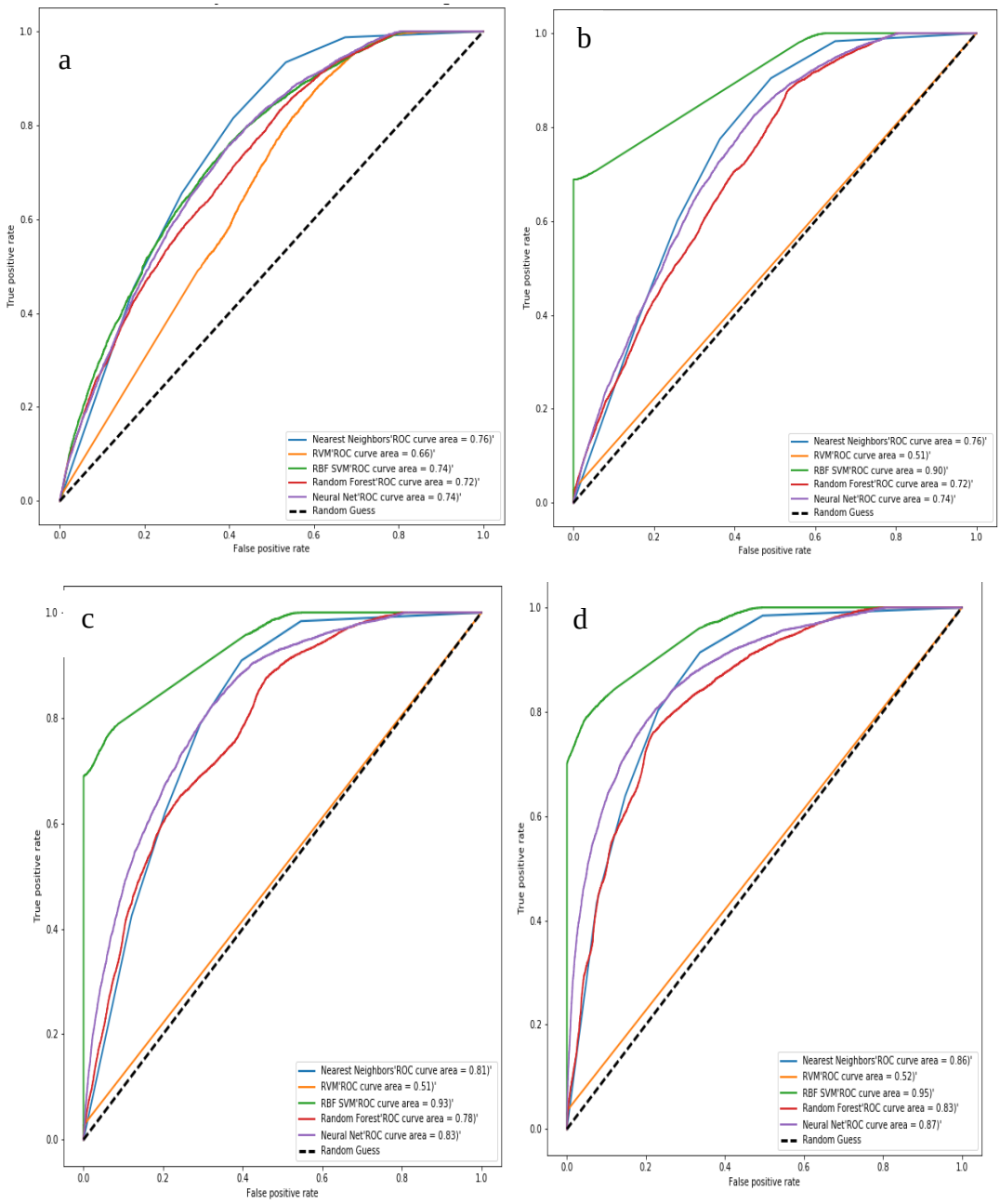


Figure 6. 8 ROC curve of the classifiers simulated traffic data under RENN (a: 30-second, b: 1-minute, c: 3-minute, d: 5-minute data)

Initially, a comparison between Table 6.19 and Table 6.17 shows that the undersampled dataset resulted in better classification results regarding the identification of conflict-prone traffic conditions, however there is an increase in false alarms. This is normal since the datasets obtained after undersampling are balanced and hence, it is not that obvious for classifiers to distinguish between the two categories. Nevertheless, in general the correct classifications (i.e. TP and TN)

are generally more than the corresponding misclassifications (i.e. FN and FP). It is also noticeable that even when using highly disaggregated data (e.g. 30-second or 1-minute) the rate of correct classification is high as well as comparable to the results from the 5-minute aggregated traffic.

The overall accuracy of the classifiers is generally lower than the accuracy achieved in the original dataset, reaching its highest rate when 5-minute aggregated data are used. The best accuracy scores are obtained using *k*NN, RFs and NN when utilizing 5-minute data. For the majority of the classifiers, except RVMs, relatively high recall along with low precision is observed. This indicated that conflict-prone conditions are correctly identified but whenever a case is classified as conflict-prone it most probably is a false alarm. Only SVMs in all temporal aggregation intervals result in high precision and recall assuring robust classification results. kNN, RFs and NNs in 1-minute, 3-minute and 5-minute data also achieve credible classification outcomes Regarding f1-score and G-means, the scores increase in general when temporal aggregation increases but without significant differences. Moreover, after overviewing Tables 6.19 and 6.20 it can be observed that when RENN is utilised with 1-minute data, the classification algorithms yield similar results to higher temporal aggregation intervals, a fact crucial for real-time safety assessment.

The ROC curves under RENN provide further insight regarding the comparison of classification results before and after undersampling. When highly disaggregated data are employed, the AUC is similar to the one occupied by classifiers trained on the original dataset. However, as data aggregation increases, the AUC increases for all classifiers in the undersampled dataset. The simple classification rules of kNN, resulted in the largest AUC score of 0.76 for 30-second data aggregation, while SVMs and NNs follow. In the ROC curves of the 1-minute, 3-minute data and 5-minute data, SVMs capture the largest area under the ROC curve. NNs, kNN and RFs also demonstrate good results while the poor performance of RVMs is further validated.

6.6.2.2. *Classification results for the simulated datasets after oversampling integrated with undersampling*

Tables 6.21 and 6.22 present the confusion matrix and the classification metrics for the simulation dataset under SMOTE-ENN. Figure 6.10 illustrates the ROC curve of the classifiers for every temporal aggregation interval.

Table 6. 21 Confusion matrix of all the classifiers for the full simulation dataset under SMOTE-ENN

Classifier	TN	FP	FN	TP	Testing sample size
30-second data					31200 (23400 non-conflict cases and 7800 conflict cases)
kNN	18183	5217	1783	6017	
RVM	16283	7117	3782	4018	
SVM	23399	1	3273	4527	
RF	17760	5640	3669	4131	
NN	18351	5049	3575	4225	
1-minute data					
kNN	17468	5932	1482	6318	
RVM	23399	1	7617	183	
SVM	23399	1	3321	4479	
RF	17020	6380	3241	4559	
NN	18825	4575	2684	5116	
3-minute data					
kNN	18681	4719	1214	6586	
RVM	23400	0	7627	173	
SVM	23400	0	2800	5000	
RF	18500	4900	2885	4915	
NN	20141	3259	2113	5687	
5-minute data					
kNN	19568	3832	1088	6712	
RVM	23400	0	7621	179	
SVM	23400	0	2455	5345	
RF	20055	3345	2369	5431	
NN	18516	4884	1338	6462	

Table 6. 22 Classification metrics of all the classifiers for the full simulation dataset under SMOTE-ENN

Classifier	Accuracy	Precision	Recall	Specificity	f1-score	G-Means
30-second data						
kNN	0.7756	0.5356	0.7714	0.7771	0.6322	0.6428
RVM	0.6507	0.3608	0.5151	0.6959	0.4244	0.4311
SVM	0.8951	0.9998	0.5804	1.0000	0.7344	0.7617
RF	0.7016	0.4228	0.5296	0.7590	0.4702	0.4732
NN	0.7236	0.4556	0.5417	0.7842	0.4949	0.4968
1-minute data						
kNN	0.7624	0.5158	0.8100	0.7465	0.6302	0.6463
RVM	0.7558	0.9946	0.0235	1.0000	0.0458	0.1528
SVM	0.8935	0.9998	0.5742	1.0000	0.7295	0.7577
RF	0.6916	0.4168	0.5845	0.7274	0.4866	0.4936
NN	0.7673	0.5279	0.6559	0.8045	0.5850	0.5884
3-minute data						
kNN	0.8098	0.5826	0.8444	0.7983	0.6895	0.7014
RVM	0.7555	1.0000	0.0222	1.0000	0.0434	0.1489
SVM	0.9103	1.0000	0.6410	1.0000	0.7813	0.8006
RF	0.7505	0.5008	0.6301	0.7906	0.5580	0.5617
NN	0.8278	0.6357	0.7291	0.8607	0.6792	0.6808
5-minute data						
kNN	0.8423	0.6366	0.8605	0.8362	0.7318	0.7401
RVM	0.7557	1.0000	0.0229	1.0000	0.0449	0.1515
SVM	0.9213	1.0000	0.6853	1.0000	0.8132	0.8278
RF	0.8169	0.6188	0.6963	0.8571	0.6553	0.6564
NN	0.8006	0.5695	0.8285	0.7913	0.6750	0.6869

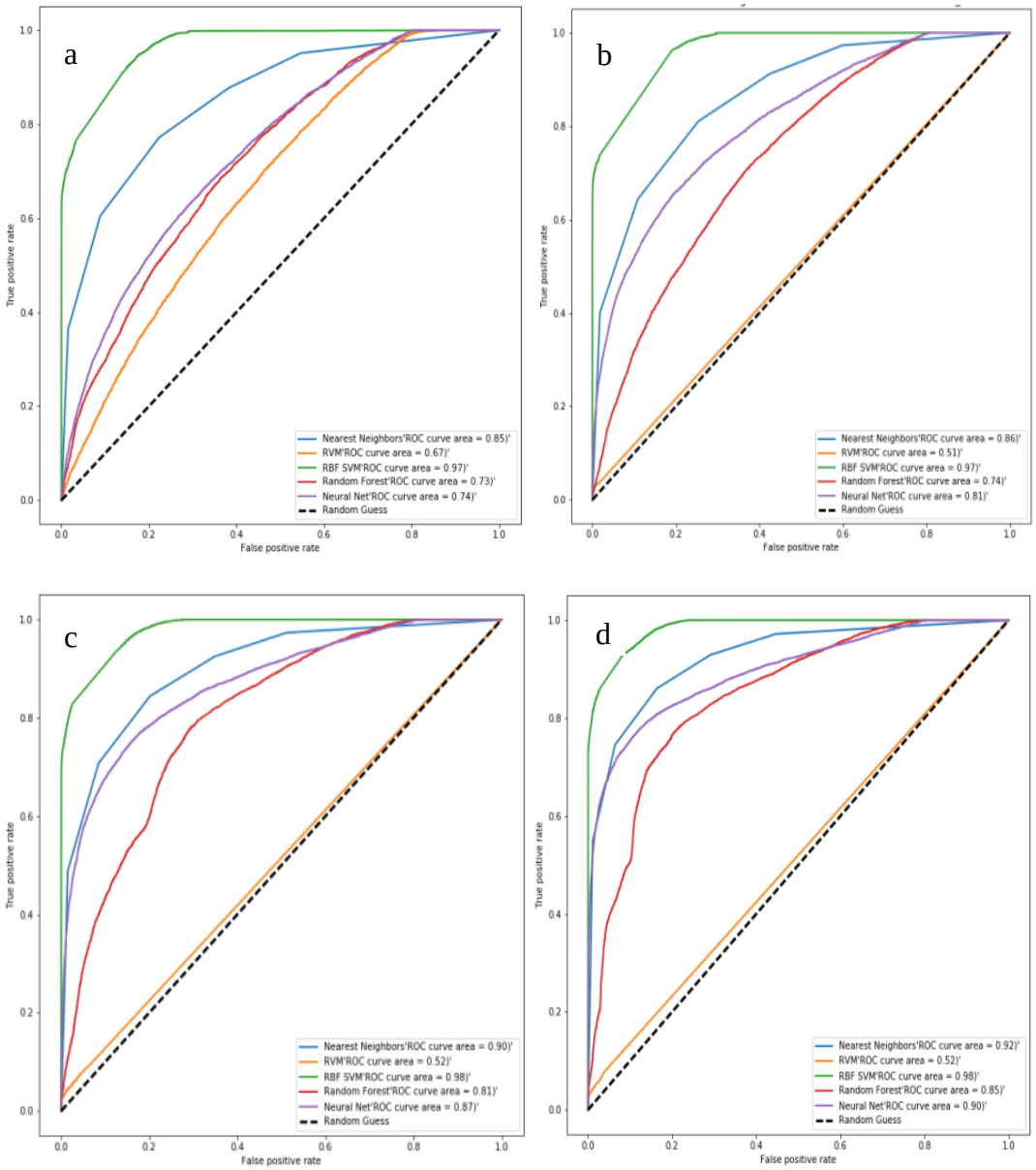


Figure 6. 9 ROC curves of the classifiers using simulated traffic data under SMOTE-ENN (a: 30seconds, b: 1-minute, c: 3-minute, d: 5-minute data)

By observing Table 6.21 it is obvious that the results of the classifiers are enhanced compared to Tables 6.17 and 6.19 for both conflict-prone and safe conditions. The number of false alarms generally decreases with higher temporal aggregation. An initial impression is than kNN and SVMs perform well regardless of the temporal aggregation while NNs and RFs improve significantly when temporal aggregation increases.

Table 6.22 indicates an enhanced performance of classifiers regarding most of the classification metrics. More specifically, recall along with precision, is generally increased which leads to the conclusion that when a conflict-prone case is identified then it is not a false alarm. Upon comparing recall and specificity it is demonstrated that for kNNs conflict-prone conditions are favoured more, while the opposite occurs for SVMs, RFs and NN regardless of the temporal aggregation. This is probably due to the fact that safe traffic conditions are the majority in the tested dataset and the data-driven approach of kNN can easily adjust the learned classifier to the testing dataset. Furthermore, the f1-score and the G-means score are enhanced as temporal aggregation increases. The reason behind that is that traffic data of higher temporal aggregation may capture traffic fluctuations that are not visible in less aggregated data. By comparing G-means it is demonstrated that the obtained classifiers lead to balanced classification results meaning both conflict-prone and safe traffic conditions could be correctly classified.

The comparison of the ROC curves (Figure 6.9) illustrates that the curves should be used with some caution. Although SVMs perform better than the other classification techniques, regarding the ratio between precision and recall, their performance does not fully justify such high AUC. However, this is a known problem of ROC curves (Fawcett, 2006, Hajian-Tilaki, 2013, Saito and Rehmsmeier, 2015) which is due to the fact that SVMs have a very low number of false alarms combined with a high number of specificity, leading to the ROC curve being slightly biased. Regarding the rest of the classifiers it is shown that kNN perform best, while NNs and RFs follow. This is due to the fact that kNN adapt better to datasets than the rest of the classifiers.

6.6.3. *Comparison of conflict-detection classifiers*

In order to compare the performance of the classifiers for the simulated dataset regarding all the temporal aggregation intervals and the imbalanced learning techniques used comparative figures were constructed. The measures of recall (i.e. the identification of conflict-prone conditions), false alarm rate (i.e. the supplementary of specificity) and the G-means (which shows the balanced classification ability of the classifiers) were utilised. Figures 6.10 – 6.12 illustrate the performance of the classifiers regarding these three metrics.

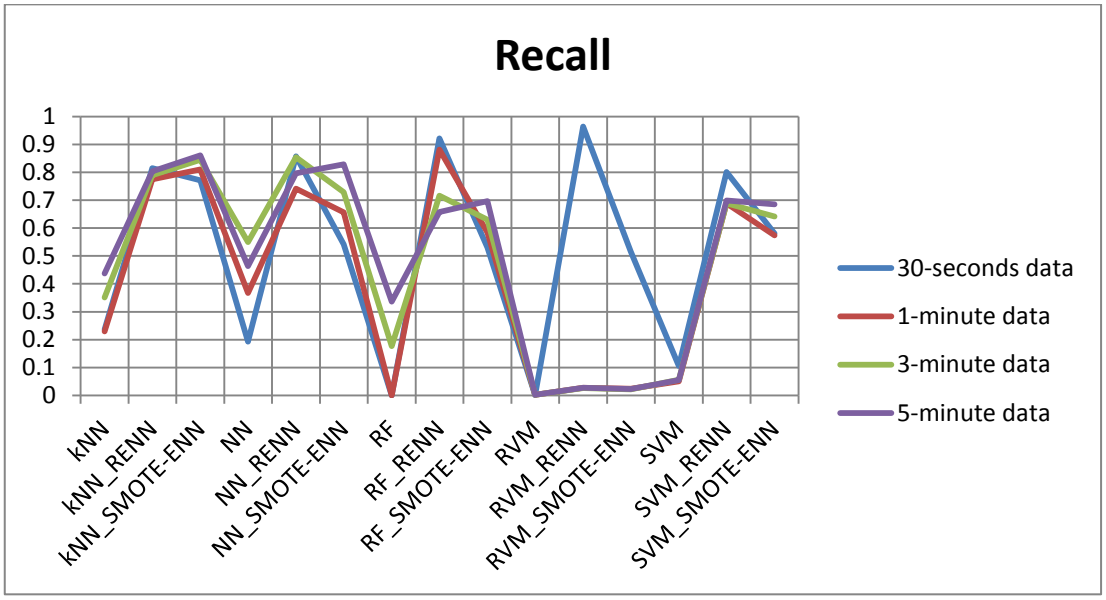


Figure 6. 10 Comparison of the recall scores between classifiers for the simulated datasets

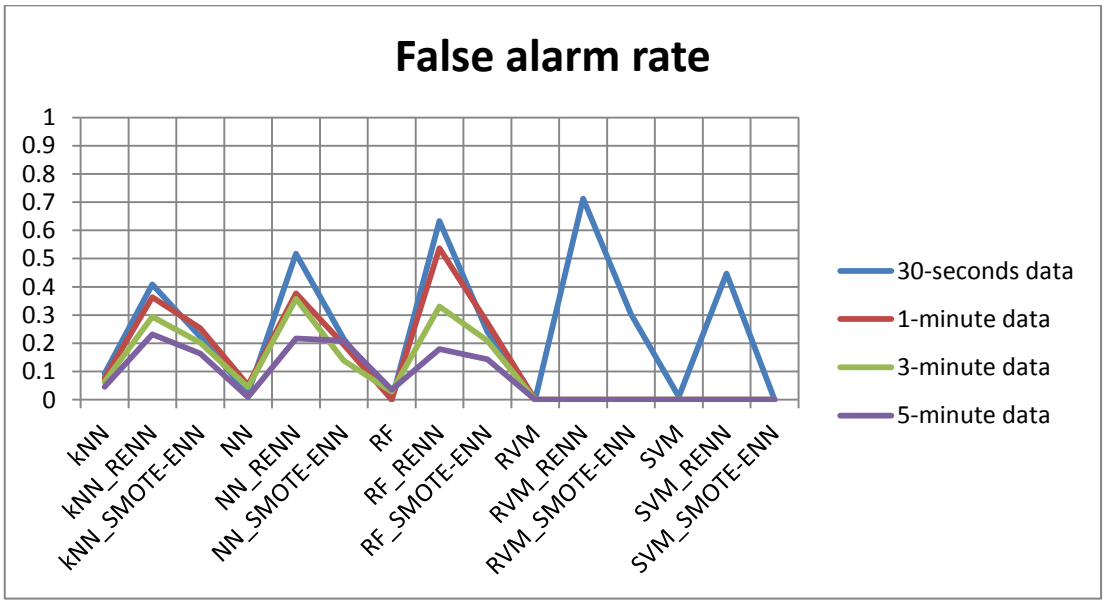


Figure 6. 11 Comparison of the false alarm scores between classifiers for the simulated datasets

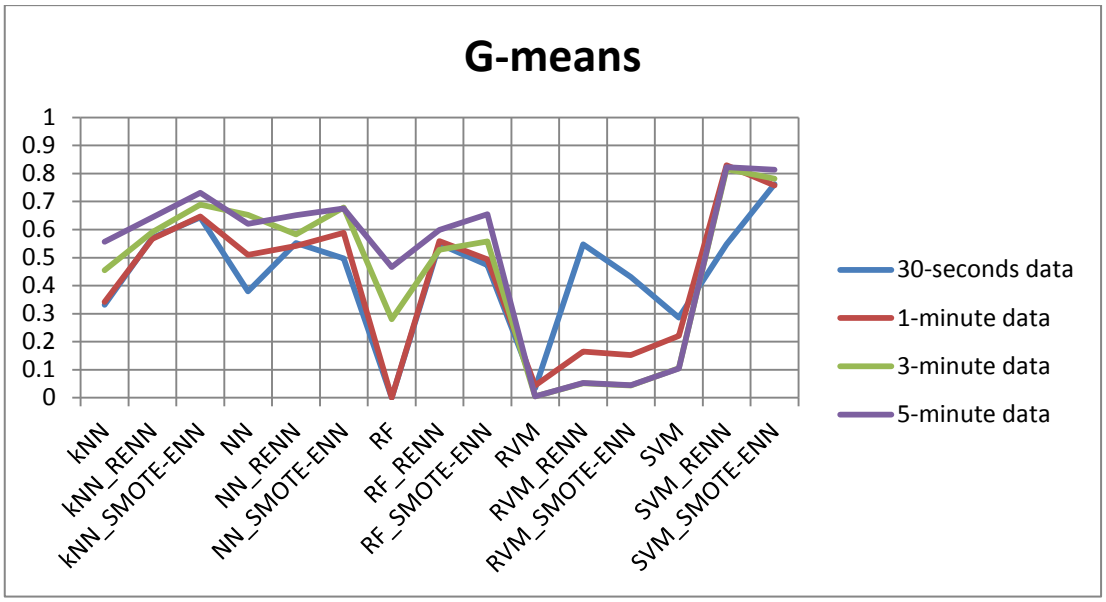


Figure 6. 12 Comparison of the G-means scores between classifiers for the simulated datasets

Figure 6.10 demonstrates that the highest recall scores when 30-second data are utilised are obtained from RVM and RF with RENN, as well as kNN and NNs utilised with SMOTE-ENN. The same algorithms, except RVMs, result in the best outcome for the 1-minute temporal aggregation equally. If higher temporal aggregation is used, undersampling utilised with NN and RF and SVMs results in the identification of more conflict-prone conditions while oversampling integrated with undersampling performs better when utilised with kNN. For the majority of the classifiers, higher temporal aggregation results in most conflicts becoming correctly classified.

However, when comparing the results from Figure 6.10 to the false alarm rates in Figure 6.11, the best classifiers become obvious through their low false alarm rates which are important for real-time applications. In general, the combination of undersampling with kNNs, RFs and NN has the best performance while oversampling integrated with undersampling works better when temporal aggregation increases and SVMs are used. Finally, looking at the G-means comparison in Figure 6.12 the superiority of SVMs regarding a balanced classification is distinct for all the temporal aggregation when imbalanced learning techniques are applied.

The results indicate the importance of data balancing techniques for acquiring better classifiers. Undersampling the majority class (i.e. safe traffic conditions) performs better than oversampling integrated with undersampling for the majority of the classifiers in identifying conflict-prone conditions and especially when higher temporal aggregation is used. On the other hand, classifiers trained with SMOTE-ENN combine good recall with low false alarm rate and are, therefore, preferable. Although more interpretable approaches such as RVMs did not yield good results, powerful classifiers such as SVMs and NNs can identify conflicts with very low false alarm rates even when raw 30-second data are provided. This is especially important for real-time and AV applications where the prediction horizon needs to be as low as possible to avoid imminent dangerous encounters.

6.6.4. *Comparison of real-time conflict detection classifiers with literature*

In order to further support the obtained classifiers which used simulated data, it was decided to compare and contrast them with NLCP models which use more aggregated data. The classifiers which are presented in Table 6.23 were compared regarding the recall and false alarm rates. These two metrics were selected because it is important for real-time collision or conflict prediction that the majority of dangerous traffic is correctly predicted without false alarms. Figure 6.13 illustrates the comparison between the literature and the developed classifiers for the simulated data.

Table 6. 23 Comparison of previous literature on NLCP with the classifiers using simulated data

Literature	Classifier	Recall	False Alarm rate
Abdel-Aty (2004)	Logistic Regression	0.69	N/A
Pande and Abdel-Aty (2006)	NN	0.57	0.29
Abdel-Aty (2008)	NN	0.61	0.21
Hossain and Muromachi (2012)	Bayesian Network	0.66	0.20
Ahmed and Abdel-Aty (2012)	Matched case-control logistic regression	0.68	0.46
Lin et al (2015)	Bayesian Network & kNN	0.61	0.38
Sun and Sun (2015)	Dynamic Bayesian Network	0.76	0.24
Aggregation	**Classifier**	**Recall**	**False Alarm rate**
30-second data	SVM_RENN	0.801	0.4475
30-second data	kNN_SMOTE-ENN	0.7714	0.2229
30-second data	SVM_SMOTE-ENN	0.5804	0

1-minute data	kNN_SMOTE-ENN	0.81	0.2535
1-minute data	SVM_RENN	0.6886	0.0004
1-minute data	RF_SMOTE-ENN	0.5845	0.2726
3-minute data	NN_RENN	0.8533	0.3577
3-minute data	kNN_SMOTE-ENN	0.8444	0.2017
3-minute data	SVM_RENN	0.6891	1E-04
5-minute data	kNN_SMOTE-ENN	0.8605	0.1638
5-minute data	NN_SMOTE-ENN	0.8285	0.2087
5-minute data	SVM_RENN	0.6992	0
5-minute data	RF_RENN	0.6576	0.1793

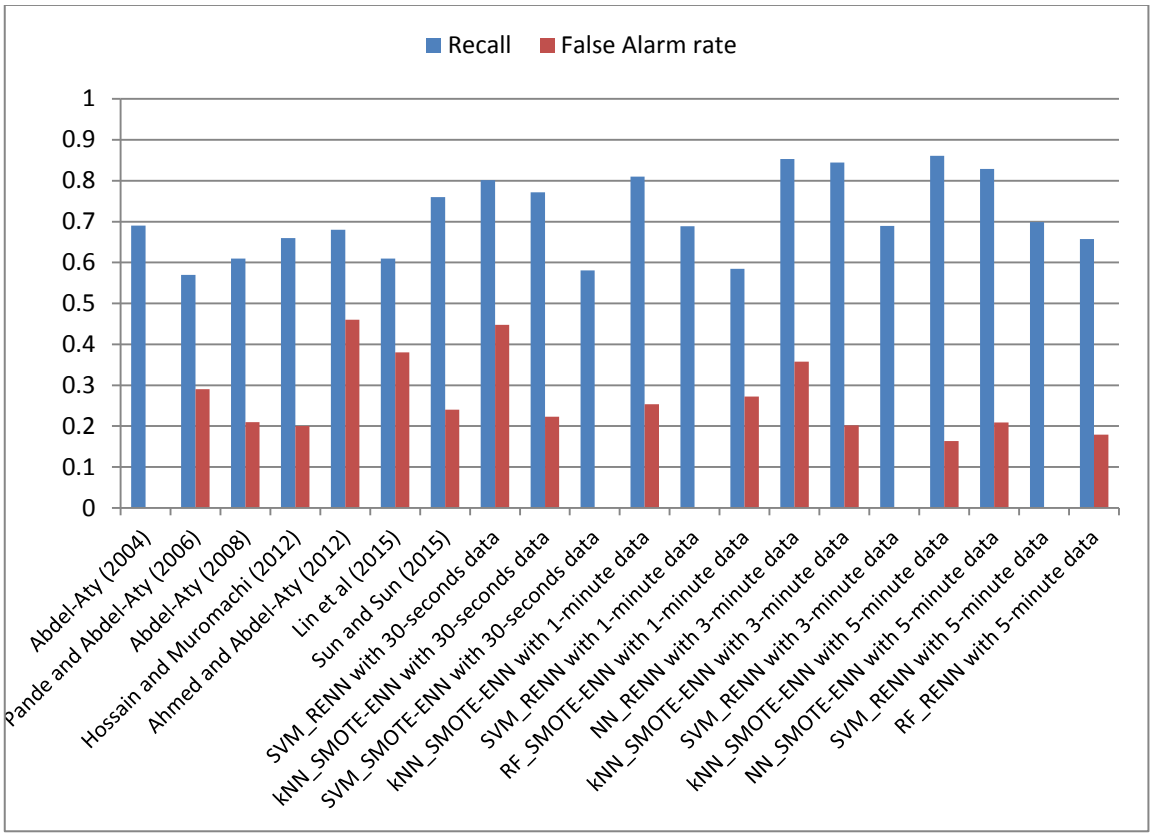


Figure 6. 13 Comparison of the recall and false alarm rates of previous literature and the best of the developed classifiers

Table 6.23 and Figure 6.13 confirm that the classifiers obtained when treating the dataset with imbalanced learning techniques outperform classifiers in the literature which used real collision data and more detailed traffic conditions datasets. The best ratio between precision and recall was found for the Dynamic Bayesian Network developed by Sun and Sun (2015). The simulation datasets without imbalanced learning treatment did not result in good classification results however it is shown that highly disaggregated traffic data can be efficiently used for classifying conflict-prone conditions with low false alarm rates. SVMs, NNs and RFs combined with

imbalanced learning and trained on simulated data are shown to predict conflict-prone traffic better than traditional techniques (e.g. Logistic Regression) and more complicated ones (e.g. static and dynamic Bayesian Networks) used in the literature. The fact that good results are obtained even when using traffic data aggregated in 30-second or 1-minute intervals and that the developed classifiers perform better than the literature exhibits two potential improvements in real-time safety studies; the potential of using highly disaggregated traffic data for collision prediction and the utilization of simulated data as a precursor for collisions.

6.7.Summary

This chapter presented the results of the classifiers that have been developed to predict collision-prone traffic conditions based on traffic characteristics. Three datasets were employed: (i) one from two UK motorways containing aggregated data, (ii) one from two urban motorways in Athens, Greece containing traffic data only 5-minutes before a collision occurrence and (iii) one from four calibrated simulations of a section of the M62 smart motorway in the UK which contained traffic aggregated in 30-second, 1-minute, 3-minute and 5-minute intervals. The corresponding conflicts for the simulated dataset were obtained using SSAM. Six classification algorithms were utilised for the task of predicting hazardous traffic conditions: k-NN, a simple data-adaptive classifier, SVMs and their Bayesian counterpart RVMs which provides probabilistic predictions, GPs another probabilistic classifier, RFs an ensemble classifier and NNs a powerful classifier which has been frequently used in real-time collision-prediction. All of the above classifiers were tested on the datasets after they had been cross-validated, keeping 2/3 of the dataset for learning and the remaining 1/3 for testing in order to avoid over-fitting. As the datasets include more normal traffic conditions compared to collision-prone ones, imbalanced learning was also utilised to improve the classification results and achieve balanced classification performance. Two imbalanced learning techniques were utilised, namely undersampling of the majority class (i.e. safe traffic conditions) and oversampling of the minority class (i.e. collision-prone traffic) integrated with undersampling. The imbalanced learning classifiers were trained using balanced datasets and were tested on the original imbalanced datasets.

The main findings for the UK dataset are the following:

- RFs showed the best performance for the original dataset and the dataset treated with imbalanced learning
- Undersampling the majority class achieved better classification performance than oversampling integrated with undersampling regarding the imbalanced learning techniques.
- Imbalanced learning achieved high classification of conflict conditions in spite of many false alarms for the majority of the classifiers.

The main findings for the Athens dataset are the following:

- It is difficult to predict collision-prone traffic just 5 minutes before a collision, however data-driven approaches such as kNN provide the best results
- Undersampling the majority class enables a much better classification performance in terms of recognising hazardous traffic without many false alarms
- SVMs and GPs showed the best classification performance among all the imbalanced learning techniques.

The main findings for the simulated datasets are:

- For all the temporal aggregation intervals, the original datasets performed worse than the datasets treated with imbalanced learning techniques.
- kNN performed better than the rest of the classifiers because its simple non-parametric nature adapts better to the applied dataset. The more robust techniques of RFs, SVMs and NNs could be utilised for every temporal aggregation interval as they perform similarly to kNN
- Oversampling the minority class works better in highly disaggregated data while undersampling the majority class resulted in better results when data were aggregated at 3-minute or 5-minute intervals.
- The higher the temporal aggregation interval, the highest the ratio between recall and false alarm rate, ensuring correct identification of conflict-prone traffic

Finally, a comparison between the models developed and classifiers already published in the literature revealed that the original datasets containing real-world traffic data 15-minute and 5-minute before a collision occurrence performed worse than the models developed in the literature. On the other hand, when imbalanced learning aids the classifiers, the developed models regarding real-world traffic data are similar or better than the literature. This way a good recall: false alarm ratio is achieved, which makes them eligible for NLCP. A comparison between the conflict prediction classifiers and previous literature in NLCP revealed that the majority of

the developed classifiers performed similarly or better than existing classifiers. This is especially important for classifiers utilizing the 30-second highly aggregated traffic data as it opens the possibility of using highly disaggregated real-world data in collision prediction as well as that of using conflicts obtained from microsimulation as a precursor of real-world collision-prone traffic.

7. Integrated risk assessment results

7.1. Introduction

The classifiers developed and presented in Chapter 6 are capable of estimating whether traffic conditions at a specific time could cause a collision or a conflict between vehicles. With the use of the proposed DBN model developed in the methodology chapter, the network-level risk assessment has been integrated with the vehicle-level risk assessment to estimate what level of impact an efficient network-level prediction model could have in distinguishing “safe” from “dangerous” traffic participants. The methodology has been implemented by using both simulated and real-world data.

7.2.The impact of NLCP on vehicle-level risk assessment

The developed DBN network which integrates network-level and vehicle-level collision prediction has been presented in Figure 4.2. The part that is of interest to this chapter and to this thesis in general is the top part of the graph as reproduced in Figure 7.1. More specifically, the estimation is correlated with the way a better collision prediction by a network-level classifier enhances or reduces the identification of a dangerous road user, given that the measurements about vehicle-level and kinematics at a previous time epoch are known.

In order to demonstrate how the network-level hint on collision risk can be employed in real-time risk assessment for autonomous driving, the vehicle-level risk in this section has been estimated with and without the network-level risk.

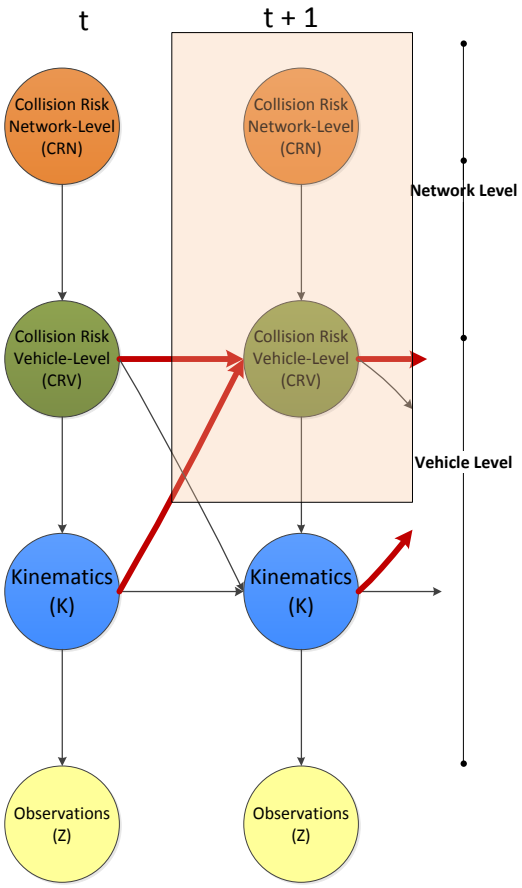


Figure 7. 1 The proposed DBN for collision risk assessment revisited (the highlighted section indicates the variables of interest)

7.2.1. Estimation of the vehicle-level risk context probability

According to equation 4.9, the probability of a vehicle-level collision risk context is given as:

$$P(\mathrm{CRV}_{\boldsymbol{n}}^{\boldsymbol{t}} = "dCP\ or\ dSA" | \mathrm{CRV}_N^{t-1} \mathrm{K}_N^{t-1} \mathrm{CRN}_n^t) = \frac{\sum_{n=1}^{N}(f_{K_n} = 1) + \sum_{n=1}^{N}(f_{CRV_n} = 1) + f_{CRN_N}}{3N}$$

where:

- $\boldsymbol{CRV_n^t}$ denotes the vehicle-level safety context of vehicle n at time t;
- $\boldsymbol{dCP, dSA}$ denote a "**d**angerous" vehicle travelling on a road segment with **C**ollision-**P**rone traffic conditions and a "**d**angerous" vehicle travelling on a road segment with **SA**fe traffic conditions respectively;

- $\boldsymbol{CRN_t}$ denotes the network-level collision risk for all the vehicles on a specific road segment;
- $f_{K_n}, f_{CRV_n}, f_{CRN_n}$ are functions which indicate the safety context of the corresponding variables (e.g. f_{CRV_n} takes the value 1 if there was a "dangerous" traffic participant in the vicinity of the ego-vehicle at the previous time moment);
- N is the number of vehicles that the ego-vehicle is sensing.

The function f_{CRN_n}, which boosts the identification of the vehicle-level safety context per the network-level risk, is given by the formula below, which considers the accuracy, recall and specificity of the network-level classifier, as well as the misclassification rates:

$$f_{CRN_n} = \begin{cases} \dfrac{Accuracy + Recall}{2} \; if \; \mathrm{CRN}_N^t = dangerous \; and \; f_{CRV_N} = 1 \\ 1 - \dfrac{Accuracy + Specificity}{2} \; if \; \mathrm{CRN}_N^t = safe \; and \; f_{CRV_N} = 0 \\ 1 - recall \; if \; \mathrm{CRN}_N^t = safe \; and \; f_{CRV_N} = 1 \\ 1 - specificity \; if \; \mathrm{CRN}_N^t = dangerous \; and \; f_{CRV_N} = 0 \end{cases}$$

Moreover, according to equations 4.13 – 4.16, the probability of a road segment having "hazardous" or "safe" traffic conditions is given by the formulas:

$P(CRN_n{}^t = "dangerous") = (\frac{Acc+Rec}{2}), \text{if CR} = 1$

$P(CRN_n{}^t = "safe") = 1 - P(CRN_n{}^t = "dangerous")$ if CR=1

$P(CRN_n{}^t = "safe") = (\frac{Acc+Spec}{2})$ *if CR=0*

$P(CRN_n{}^t = "dangerous") = 1 - P(CRN_n{}^t = "safe")$ *if CR=0*

where *CR* is the classification result for the aggregated traffic conditions in real-time (i.e. 0 or 1), *Acc* and *Rec* are the accuracy and recall of the calibrated classifier, while *Spec* is the specificity of the classifier.

7.2.2. Estimation of vehicle-level risk using simulated data

In this section, the results from two classifiers utilised in Chapter 6 are going to be utilised for the estimation of vehicle-level risk. The classifiers which are going to be initially tested are the kNN classifier under SMOTE-ENN utilised with the 30-second simulated data and the GP classifier of the 5-minute Athens dataset under NC. These classifiers are examples of the best classification results in the previous chapters and were chosen in order to estimate vehicle-level risk with as little prediction horizon as possible using disaggregated traffic data.

Assuming that vehicle-level measurements were not available, the following artificial scenarios are formulated for the estimation of the vehicle-level risk:

7.2.2.1.Traffic data aggregated at 30-second intervals

It is assumed that once traffic conditions are classified, the prediction is broadcasted for a time interval equal to the traffic data aggregation. Therefore, if the traffic data aggregation is 30-seconds, every NLCP prediction lasts for 30 seconds. In this scenario, it is assumed that traffic conditions are classified as conflict-prone and at time t_1=10seconds after the beginning of the NLCP prediction there is a traffic participant which poses a threat to the ego-vehicle. Furthermore, it is assumed that this "dangerous" vehicle has kinematics that indicate an imminent danger for the ego-vehicle. Hence, according to equations 4.6 and 4.7: ${f_{K_N}}^{t=10} = 1$ and ${f_{CRV_N}}^{t=10}$=1. It should be noted here that 10 indicates the time moment occurring ten seconds after the network-level prediction and hence 20 seconds remain for the end of the temporal aggregation interval.

The kNN classifier under SMOTE-ENN with 30-seconds temporal aggregation resulted in 77.56% accuracy, 77.14% recall and 77.71% specificity.

Scenario 1: Traffic conditions are predicted as conflict-prone

According to the formula that gives the network-level collision risk:

$$P({CRN_n}^t = "dangerous") = (\frac{Acc+Rec}{2}) = \frac{0.7756+0.7714}{2} = 0.7735 = 77.35\%$$

Furthermore, as the traffic conditions are estimated as dangerous and ${f_{CRV}}_N^{t=10}$=1, the boosting parameter for the vehicle-level safety context ${f_{CRN}}_N$ is equal to $P({CRN_n}^t = "dangerous")$. Consequently, ${f_{CRN}}_N^{t=10} = 0.7735$.

Figure 7.1. illustrates the estimation of vehicle-level risk context when the ego-vehicle is sensing 1, 3, 5 and 10 vehicles in its vicinity, with and without the network-level hint.

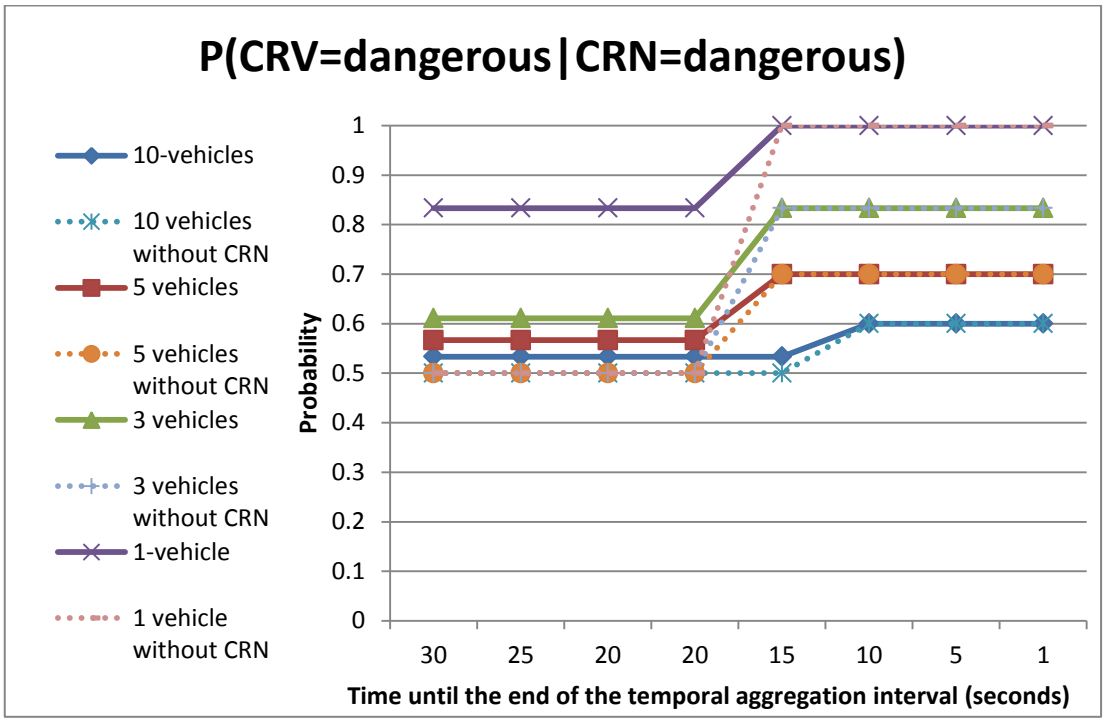


Figure 7. 2 Estimation of P(CRV=dangerous|CRN=dangerous)for a multiple vehicle scenario

From Figure 7.2 the potential enhancement of the vehicle-level safety context could be observed. First of all, if network-level safety information is available, the probability of a vehicle being considered as a threat is higher, which may be conservative as an approach but induces a hint to the ego-vehicle that a danger is imminent. Moreover, it is shown that this extra hint results in a faster increase of probability when a vehicle is sensed to be performing a dangerous manoeuvre, which could lead to the faster identification of a dangerous road user and an earlier initiation of the manoeuvre to avoid the danger. If, for example, a threshold is defined (e.g. if probability is over 65%) in order to raise a warning to the risk

assessment module the AV, then figure 7.2 demonstrates that the threshold is raised faster if network-level information is available.

To further demonstrate how vehicle-level safety is affected, a second scenario was investigated. This relates to the probability of a vehicle driving dangerously, given that the network-level collision risk is predicted as safe.

Scenario 2: Traffic conditions are predicted to be "safe"

According to the formula that gives the network-level collision risk:

$$P({CRN_n}^t = "safe") = (\frac{Acc + Spec}{2}) = \frac{0.7756 + 0.7771}{2} = 0.77635$$

Because in this scenario the traffic conditions are estimated as safe and ${f_{CRV}}_N^{t=10}$=1, the boosting parameter for the vehicle-level safety context ${f_{CRN}}_N$ is equal to:

${f_{CRN}}_N = 1 - recall$ in order to represent the false negative rate i.e. the probability that the traffic conditions are falsely identified as safe.

Hence, ${{f_{CRN}}_N}^{t=10} = 1 - recall = 1 - 0.7714 = 0.2286\%$=22.86%.

Figure 7.3 illustrates the estimation of the probability of the vehicle-level risk context being dangerous when the ego-vehicle is sensing 1, 3, 5 and 10 vehicles in its vicinity with and without the network-level hint.

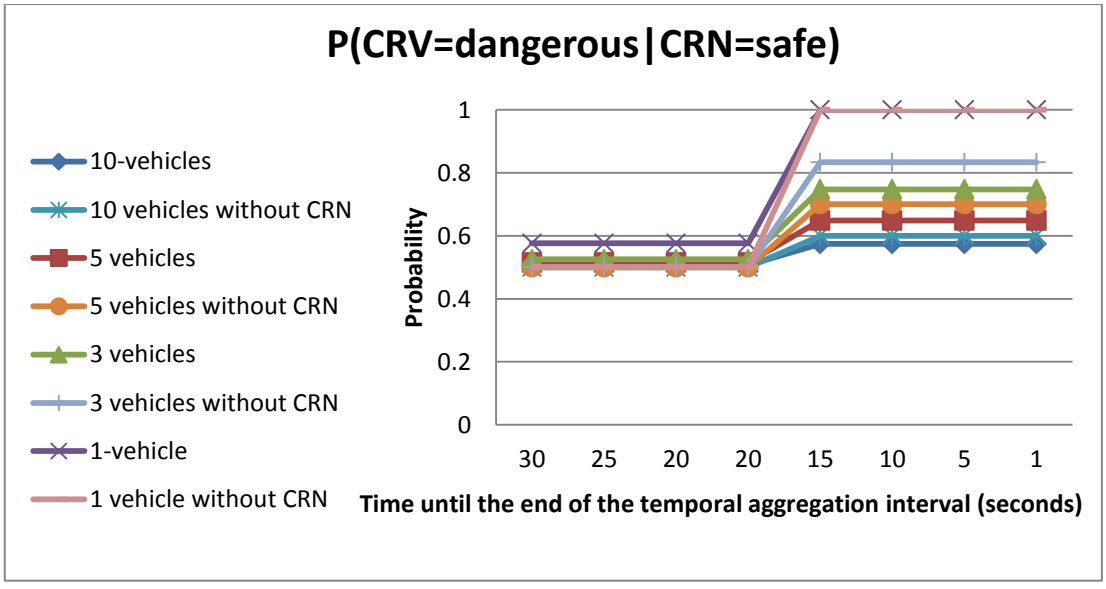


Figure 7. 3 Estimation of P(CRV=dangerous|CRN=safe) for a multiple vehicle scenario

From Figure 7.3 it is shown that the estimation of the probabilities without the network-level hint results in higher rates and a faster identification of the dangerous road user. Only when just one vehicle is in the vicinity of the ego-one and the dangerous road user is obvious, the two approaches (i.e. with and without network-level information) yield similar results. This indicates that when NLCP indicates safe traffic conditions, more trust should be given to the vehicle measurements rather than the network traffic information.

7.2.2.2. Traffic data aggregated at 5-minute intervals

In order to further test the impact of network-level collision information on vehicle-level collision risk, the classifier developed on the 5-minute aggregated data from Athens was utilised. The classifier achieved 83.95% accuracy, 91.71% specificity and 68.86% recall. For this scenario, the number of vehicles was randomly sampled for each time moment. It was also assumed that a vehicle performs dangerous manoeuvres starting from t=180 before the end of the temporal aggregation to t=100 seconds before the end of the temporal aggregation interval. Hence, ${f_K}_N^{t=180:100} = 1$ and ${f_{CRV}}_N^{t=180:100}$=1.

Scenario 1: Traffic conditions are predicted as collision-prone

According to the formula that gives the network-level collision risk:

$$P({CRN_n}^t = "dangerous") = (\frac{Acc+Rec}{2}) = \frac{0.8395+0.6886}{2} = 0.7641 = 76.41\%$$

Furthermore, for the time intervals t=300:180 and t=100:0 the traffic conditions are estimated as dangerous but there is no vehicle performing dangerous manoeuvres. Therefore, the boosting parameter for the vehicle-level safety context during these intervals is:

$${f_{CRN}}_N^{t=300:180\ \&\ t=100:0} = 1 - \frac{Accuracy+Specificity}{2} = 0.1217$$

For the time interval t=180:100 traffic conditions are estimated as collision-prone and there is only one vehicle performing a hazardous manoeuvre. Therefore, the boosting parameter for the vehicle-level safety context during these intervals is:

$${f_{CRN}}_N^{t=180:100} = \frac{Accuracy + Recall}{2} = 76.41\%$$

Figure 7.3 illustrates the estimation of the probability of a vehicle being dangerous during the 5-minute traffic data temporal aggregation interval in a multiple vehicle scenario.

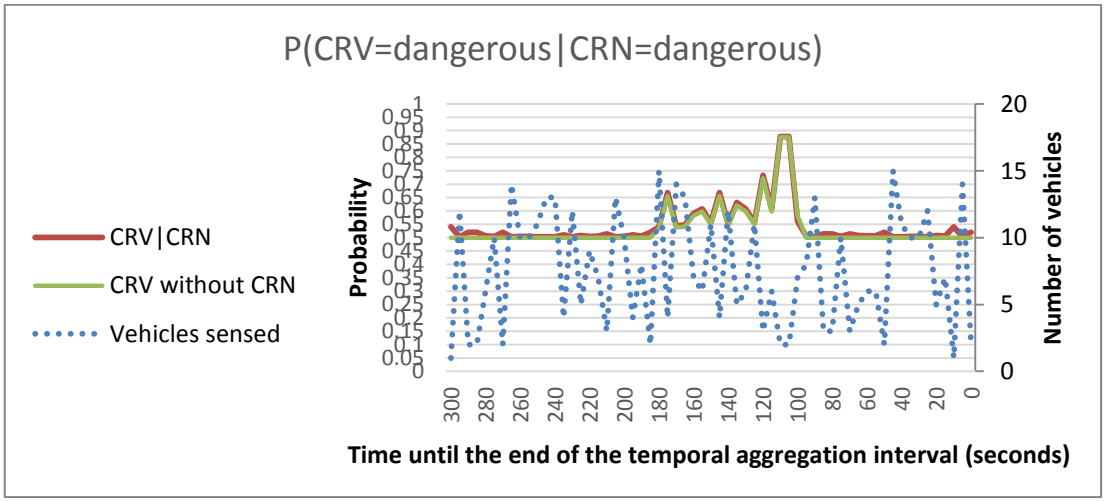


Figure 7. 4 Estimation of P(CRV=dangerous|CRN=dangerous) for a 5-minute traffic data aggregation interval

From Figure 7.4 it is further justified that knowing the NLCP estimation enhances the probability of another vehicle driving dangerously with respect to the ego-vehicle. From t=180 seconds until t=100, when a nearby vehicle is assumed to perform dangerous manoeuvres, the probability of the vehicle being dangerous given the network-level hint is higher than the corresponding probability without the network-level information. Moreover, it is demonstrated that the lower the number of vehicles, the more obvious it is to recognise the vehicle which is driving "dangerously". This is normal because with fewer vehicles, the one responsible for triggering an accident is easier to detect. Nevertheless, it is advantageous that the line representing the probability P(CRV|CRN) is above the corresponding probability graph which does not take into account network-level collision information. It is also observed that at a time moment when no danger is imminent the probability is increased, which is a potential drawback. However, this can be utilised as extra caution by an AV's planning module.

Scenario 2: Traffic conditions are predicted as safe

The classifier achieved 83.95% accuracy, 91.71% specificity and 68.86% recall. Given that the traffic conditions are predicted safe, the network-level collision risk can be estimated as:

$$P({CRN_n}^t = "dangerous") = 1 - (\frac{Acc+Spec}{2}) = 1 - \frac{0.8395+0.9171}{2} = 0.1217 = 12.17\%$$

Furthermore, for the time intervals t=300:180 and t=100:0, the traffic conditions are estimated as safe without a vehicle perceived as a threat. Therefore, during these intervals:

$$f_{CRN_N}{}^{t=300:180\ \&\ t=100:0} = P(CRN_n{}^t = \text{"}dangerous\text{"}) = 0.1217$$

For the time interval t=180:100 traffic conditions are estimated as safe but there is one vehicle performing hazardous manoeuvres. Therefore, the boosting parameter for the vehicle-level safety context during these intervals is:

$$f_{CRN_N}{}^{t=180:100} = 1 - Recall = 1 - 0.6886 = 0.3114$$

Figure 7.4 illustrates the estimation of the probability of the vehicle-level risk context being dangerous during the traffic data temporal aggregation interval and according to the vehicles sensed.

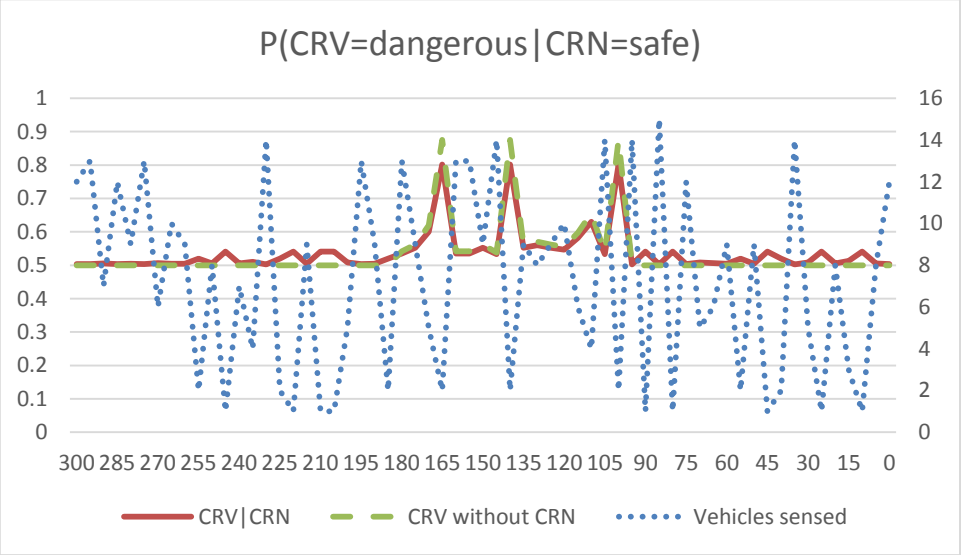


Figure 7. 5 Estimation of P(CRV=dangerous|CRN=safe)for a 5-minute traffic data aggregation interval

Like the case when traffic data were aggregated in 30-seconds intervals and the traffic conditions were assumed to be safe, Figure 7.5 illustrates that, when a danger is sensed by the ego-AV, network-level information does not contribute to the enhancement of the corresponding probability.

7.2.3. Estimation of vehicle-level risk using real-world data

In order to validate the credibility that network-level information has on the estimation of vehicle-level collision prediction, the vehicle-level data as described in Chapter 5 were utilised.

More specifically, the available TTC measurements were filtered in order to identify hazardous road users. According to the same principle as the one used in SSAM to derive conflicts, TTC values below 1.5 seconds were flagged as "hazardous" because 1.5 is the average human reaction time (Triggs and Harris, 1982). The number of hazardous vehicles during the trip is given in Figure 7.6.

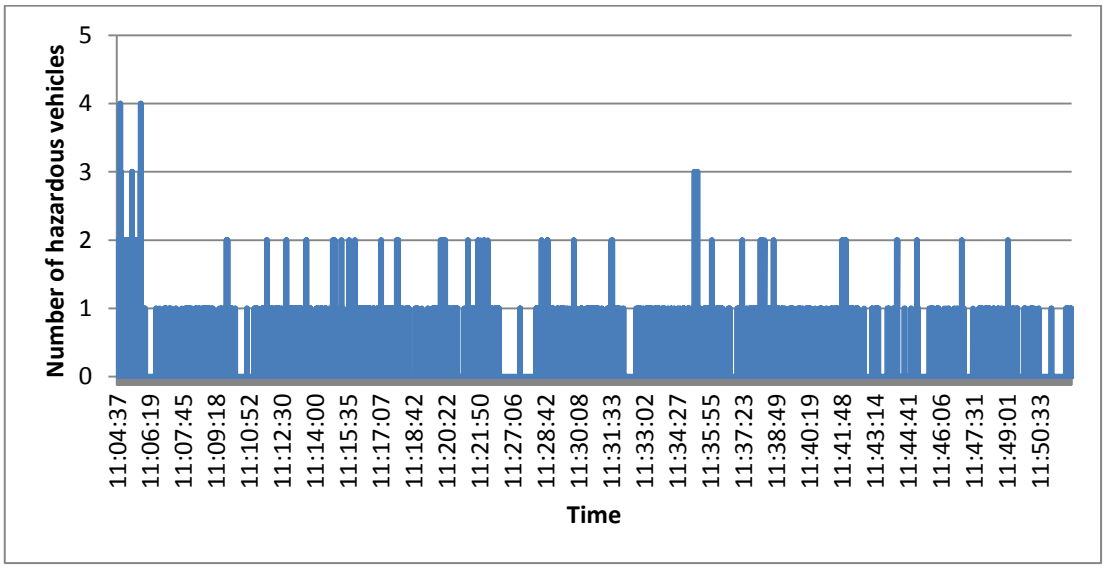


Figure 7. 6 Number of dangerous vehicles with respect to the ego-vehicle

The time interval from 11:05:37 to 11:06:25 was used in the analysis as the highest number of "hazardous" road users was observed during that one minute.

The classifiers that were tested for the estimation of CRV based on the network-level information and their characteristics are described in Table 7.1. For each of the classifiers the probability that a vehicle drives dangerously was estimated given that the NLCP points towards collision-prone and safe traffic. For the estimation of vehicle-level risk context the formulas 4.13-4.16 were used. For every vehicle with TTC<1.5 seconds it was assumed that the vehicle's kinematics were also dangerous so as to have f_{K_N}=1.

Table 7. 1 NLCP classifiers used for vehicle-level risk estimation

Traffic data aggregation	Classifier	Accuracy	Recall	Specificity
30-seconds	kNN with SMOTE-ENN	0.8395	0.6886	0.9171
1-minute	SVM with RENN	0.9219	0.6886	0.9996
3-minute	SVM with RENN	0.9222	0.6891	0.9999
5-minute	NN with SMOTE-ENN	0.8006	0.8285	0.7913

7.2.3.1.Estimation of vehicle-level risk given traffic conditions are collision-prone

Figures 7.7-7.10 illustrate the results for the probability that a vehicle poses a threat to the ego-one, given the available network-level information and the vehicle-level data.

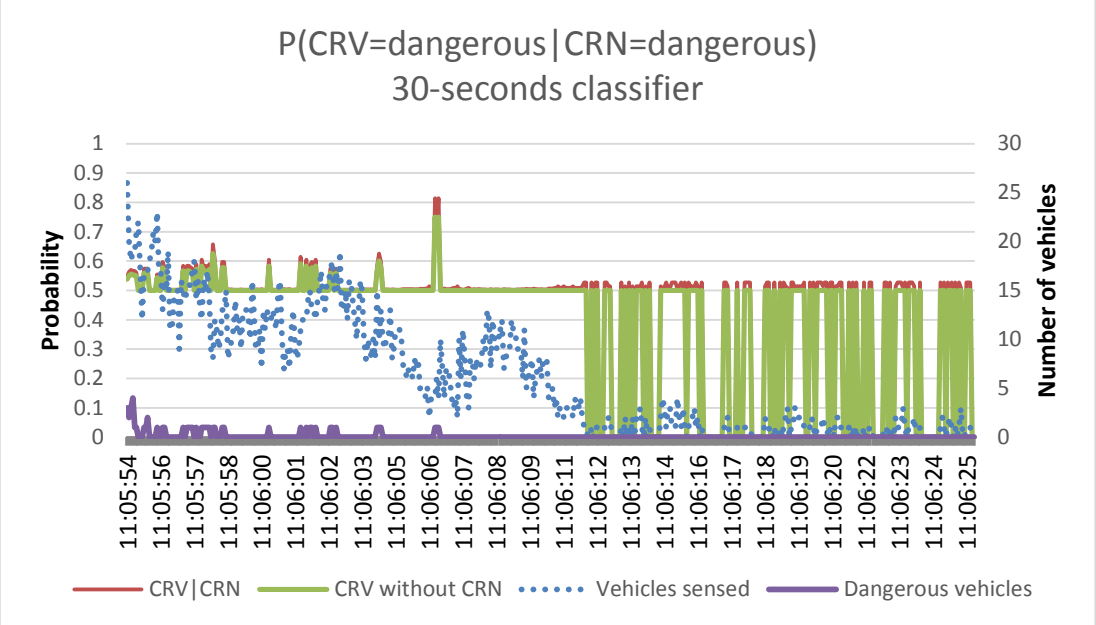


Figure 7. 7 Estimation of vehicle-level risk using 30-seconds network-level information (conflict-prone conditions)

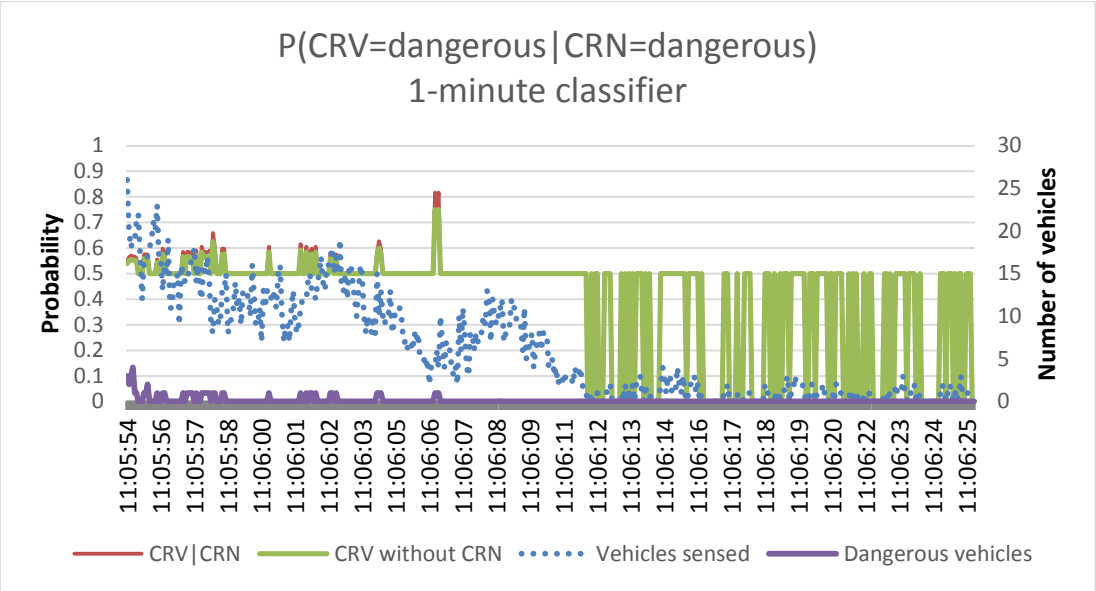


Figure 7. 8 Estimation of vehicle-level risk using 1-minute network-level information (conflict-prone conditions)

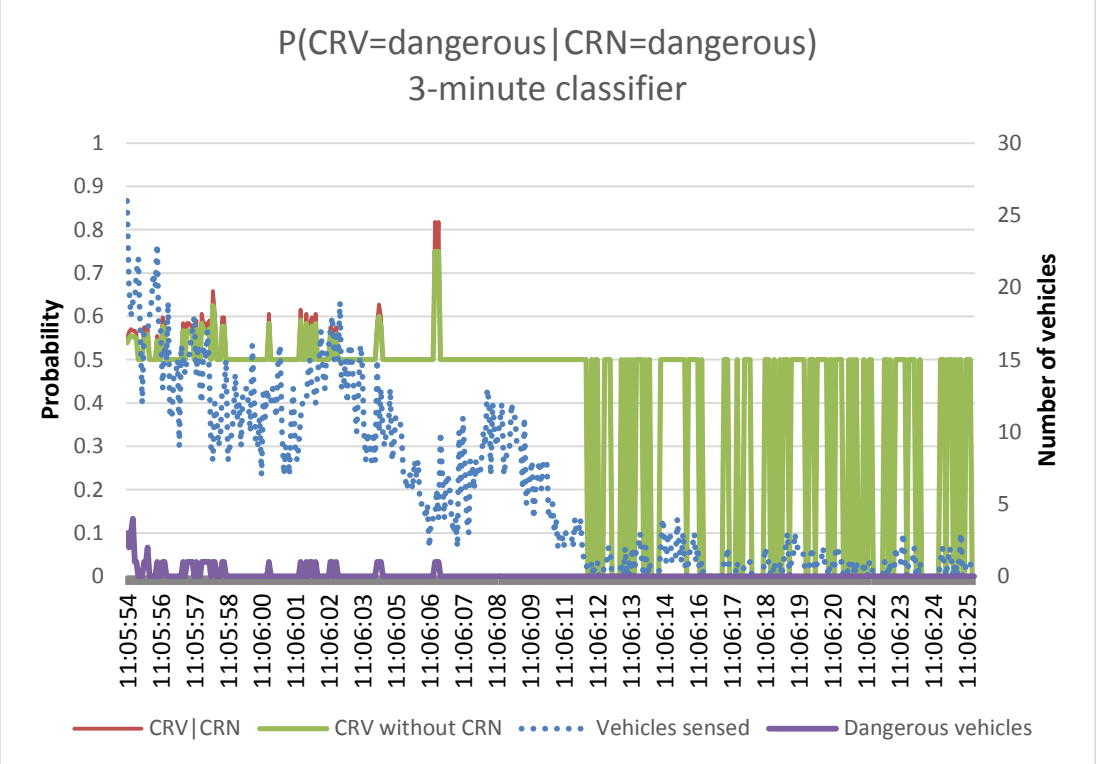


Figure 7. 9 Estimation of vehicle-level risk using 3-minute network-level information (conflict-prone conditions)

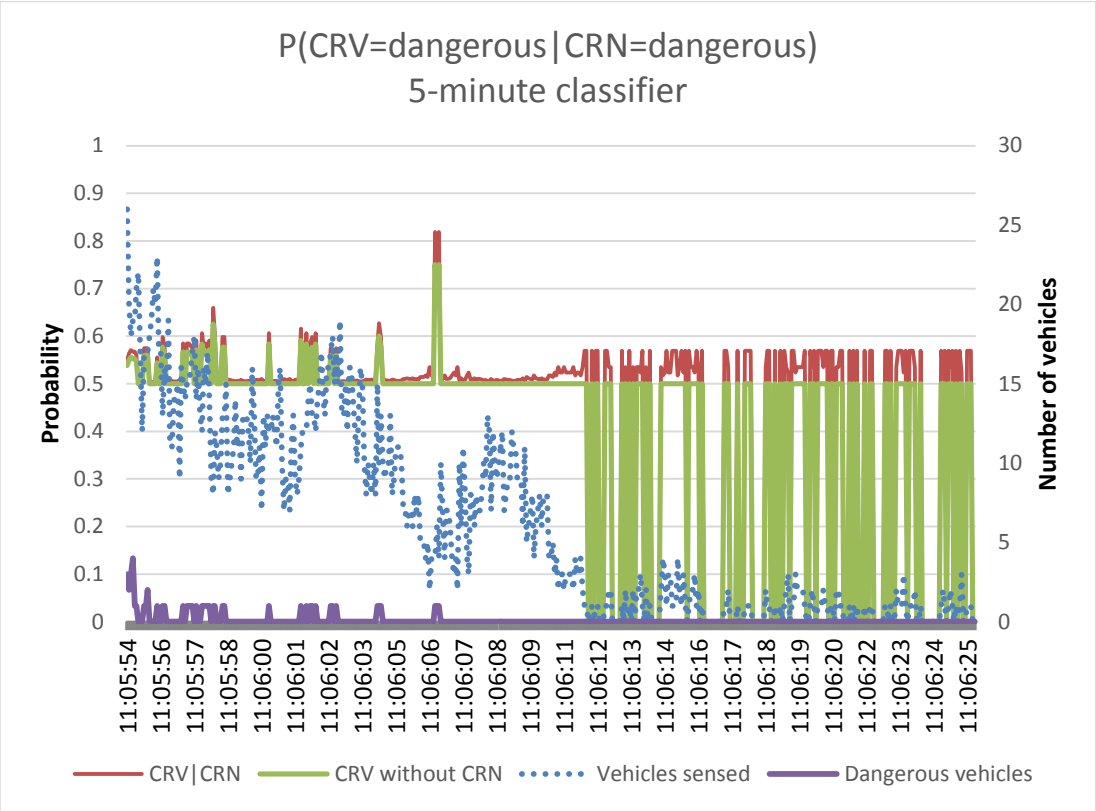


Figure 7. 10 Estimation of vehicle-level risk using 5-minute network-level information (conflict-prone conditions)

After observing Figures 7.7-7.10 it is further validated that, when traffic conditions are predicted as conflict-prone, it is easier to identify if there is an imminent danger for the ego-vehicle. Even when highly disaggregated traffic data are utilised, the probability of a dangerous vehicle being dangerous is enhanced when compared to the probability obtained only from vehicle-level measurements. When the number of vehicles sensed is high the enhancement in the probability is lower. However, the plot of CRV|CRN is always higher than the one of CRV without network-level information, assuring a greater level of safety for the ego-vehicle.

To illustrate the effect of network-level information on vehicle-level risk estimation, Figure 7.11 presents a plot of the percentage difference between the estimation of the probability that a vehicle drives in a “hazardous” way with regards to the ego-vehicle with and without NLCP.

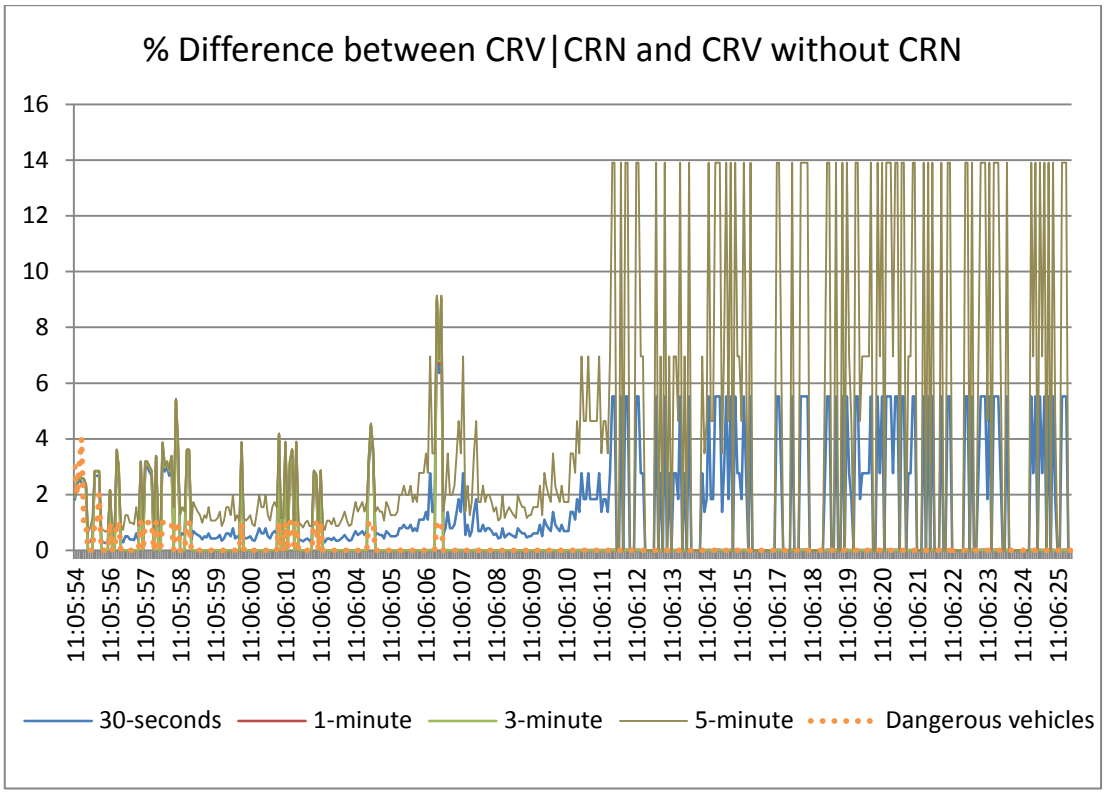


Figure 7. 11 Difference (%) between vehicle-level risk estimation with and without network-level information (conflict-prone conditions)

From Figure 7.11 it can be concluded that the greater influence came from the 5-minute classifier. This is probably due to the ability of the classifier to better detect conflict-prone and safe traffic efficiently as observed from its recall and sensitivity statistics. When there is at least one dangerous vehicle, the estimation of a dangerous vehicle-level safety context is enhanced by up to 9%, ensuring safer navigation. When no dangerous vehicles are detected, the difference can reach up to 14%. This shows that, when traffic conditions are predicted as dangerous, the ego-vehicle can adjust to a more cautious behaviour as a conflict or collision might occur.

Overall, when traffic conditions are predicted as hazardous, the ego-vehicle can better estimate if a vehicle is driving dangerously, even when highly disaggregated traffic data information is available. Furthermore, the fact that, a small probability of a dangerous vehicle is assigned even when no dangerous vehicles are around, can be exploited in an AV risk assessment module.

7.2.3.2.Estimation of vehicle-level risk given traffic conditions are safe

Figures 7.12-7.15 illustrate the results for the probability that a road user is driving dangerously towards the ego-vehicle, given the available network-level information and the vehicle-level data if the traffic conditions are indicated as safe.

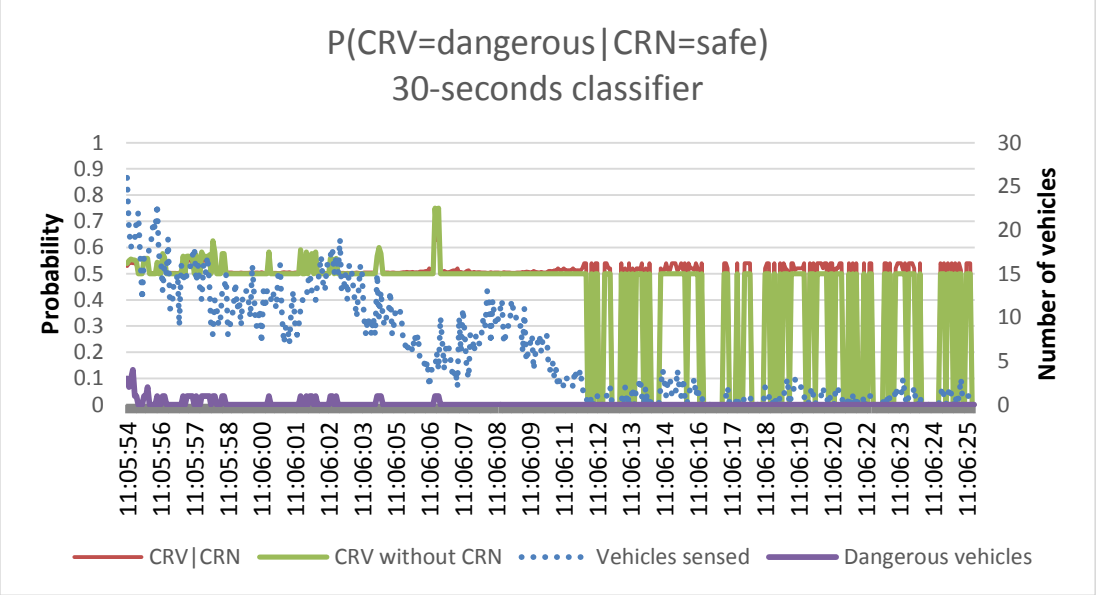


Figure 7. 12 Estimation of vehicle-level risk using 30-seconds network-level information (safe conditions)

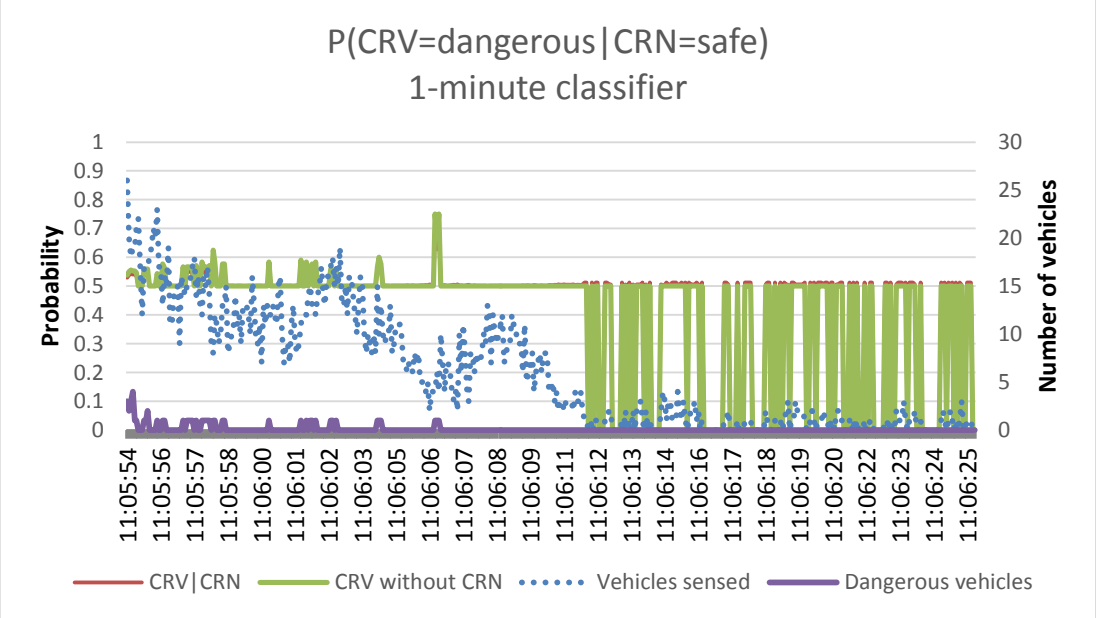


Figure 7. 13 Estimation of vehicle-level risk using 1-minute network-level information (safe conditions)

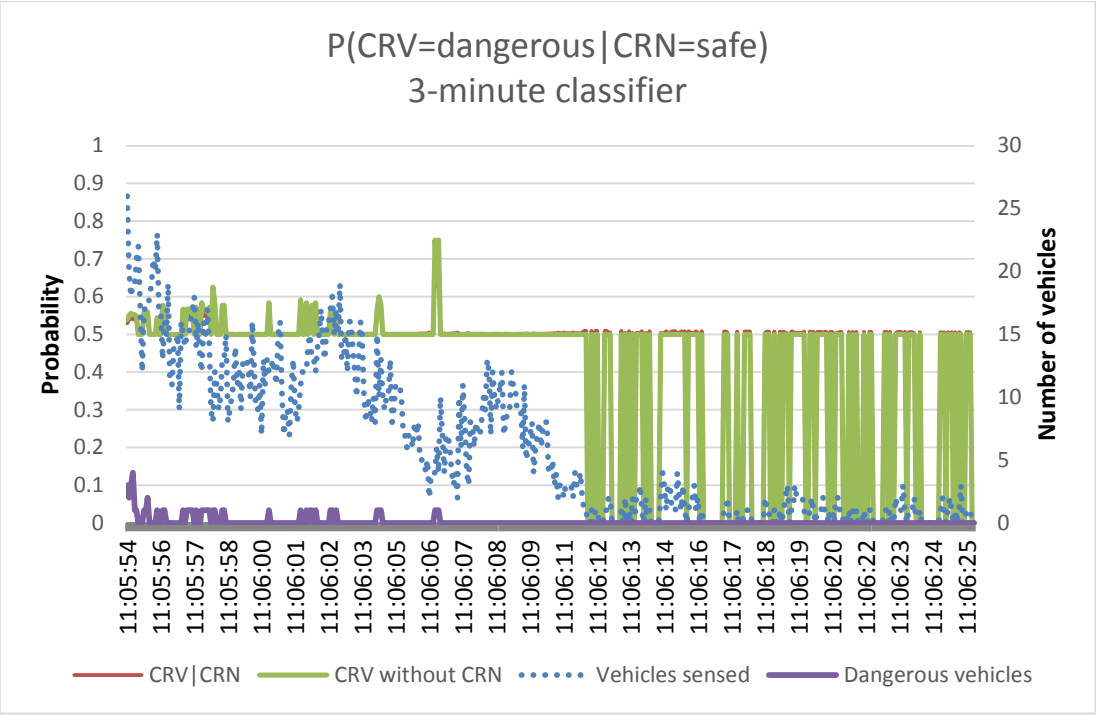


Figure 7. 14 Estimation of vehicle-level risk using 3-minute network-level information (safe conditions)

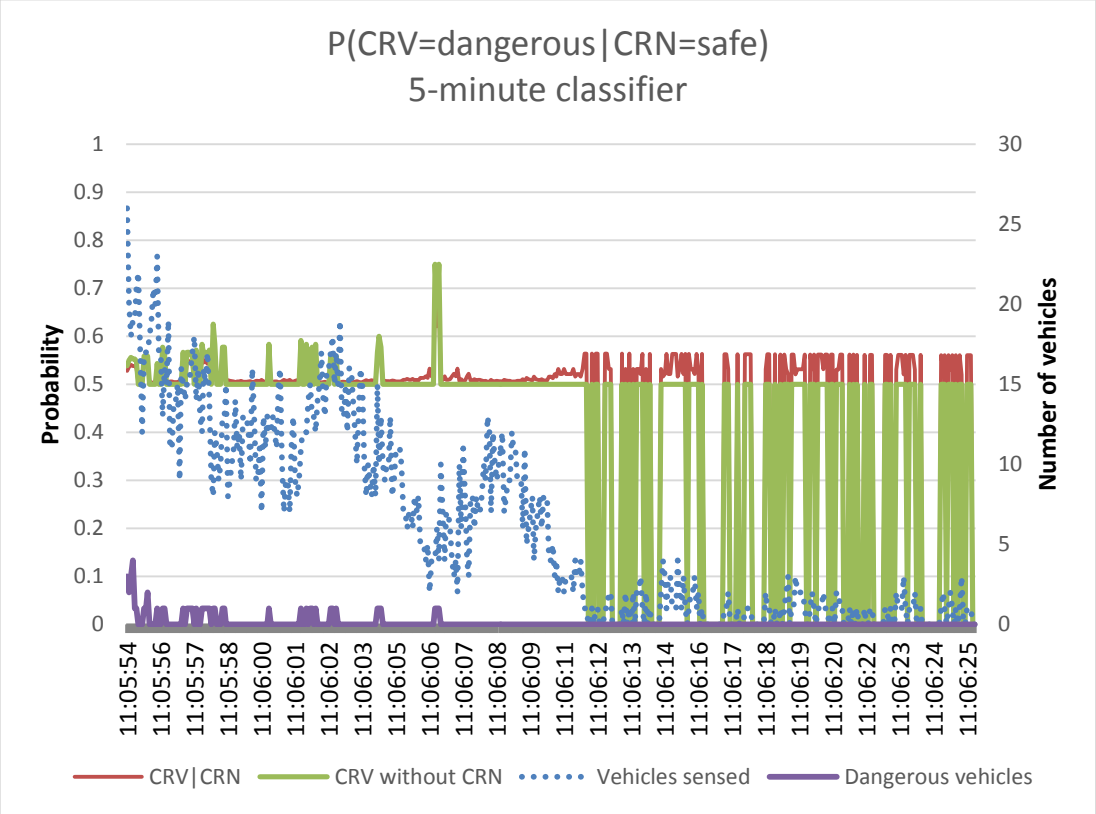


Figure 7. 15 Estimation of vehicle-level risk using 5-minute network-level information (safe conditions)

Similar to the case of simulated data, Figures 7.12-7.15 demonstrate that, if real-time network-level information points towards safe traffic conditions, then the measurements from the sensors of the ego-vehicle are more reliable to detect dangerous traffic participants. The differences between the two different ways to estimate the vehicle-level safety context probabilities are more obvious when better NLCP classifiers are used, such as the 5-minute classifier demonstrated in this chapter. Even when no dangerous vehicles are detected and traffic conditions are predicted as safe, the probability that a vehicle could be dangerous is elevated due to the possibility that the network-level information is falsely classified.

As with the conflict-prone conditions, Figure 7.16 demonstrated the percent difference between the two different approaches to estimate the probability that a vehicle is driving dangerously towards the ego-one.

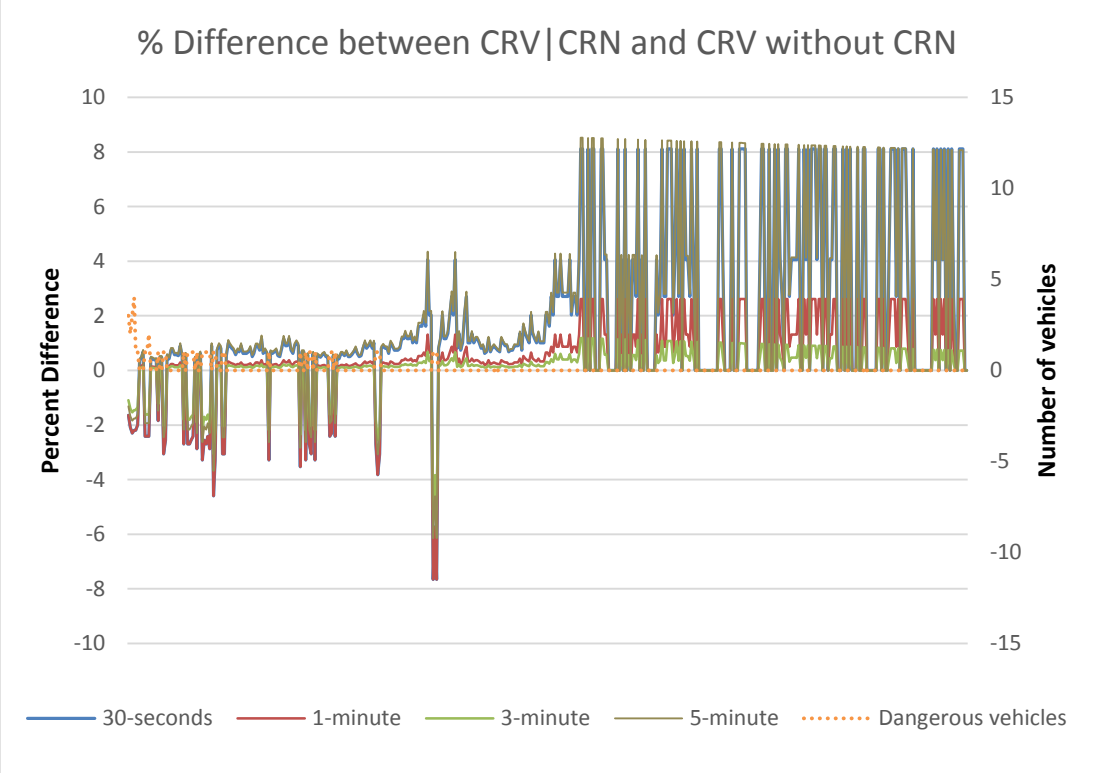


Figure 7. 16 Difference between vehicle-level risk probability with and without network-level information (safe conditions)

From Figure 7.16 it is noticeable that network-level information does not enhance AV risk assessment when traffic conditions are predicted as conflict-prone. As mentioned before, network-level information induces a slight probability that the network-level prediction is wrong when no vehicle is detected as dangerous. On the

other hand, in cases when there is an imminent danger, utilizing vehicle-level information only, results in a better hazard recognition than the proposed methodology, reaching up to 8% more confidence in estimating a dangerous traffic participant.

It should be noted that the extracted probabilities for all the scenarios are not high enough. The scenarios developed in this chapter were built on many assumptions and without highly detailed vehicle-level data. For the scenarios where traffic conditions were indicated as collision- or conflict-prone the probability of another vehicle being dangerous was higher when CRN was available, however further work is needed to calibrate the proposed DBN model in the cases when NLCP indicates safe traffic. Nevertheless, the enhanced probability for the dangerous road user when collision-prone traffic was predicted shows that the method has potential for utilization in AV risk assessment.

7.3.Summary

This chapter presented the potential impact that the network-level classifiers developed in Chapter 6 would have on the identification of "dangerous" road users using artificial data and the vehicle-level data collected for this thesis. Initially, using two of the best classifiers (i.e. the kNN classifier under SMOTE-ENN utilised with 30-second simulated data and the GP classifier of the 5-minute Athens dataset under NC) and randomly sampling a number of vehicles sensed by an ego-AV, the probability of estimating a "dangerous" road user was estimated through the DBN model formulas suggested in Chapter 4. It was shown that both in the case of 30-seconds data as well as the case of 5-minute traffic data, the probability of identifying a traffic participant was enhanced if NLCP indicated collision or conflict-prone traffic. On the other hand, when traffic conditions were indicated as safe, the prediction did not enhance the probability that a road user was a "threat" for the ego-vehicle.

The artificial data indicated the potential of using network-level information on AV risk assessment and hence the DBN was further tested using the vehicle-level data collected for the purpose of this thesis. Using real-world data and the classifiers trained on highly disaggregated traffic data it was validated that when traffic

conditions are classified as hazardous, then the identification of dangerous traffic participants is significantly enhanced. This enhancement is greater when 5-minute traffic data are utilised for predicting network-level collisions. Nevertheless, even when highly disaggregated traffic data (i.e. 30-seconds) were used, the probability of a traffic participant posing a threat to the ego-vehicle was enhanced. However, more work is needed to assure that even when NLCP indicates safe traffic, aggressively driving traffic participants are more robustly identified.

8. Conclusion and Discussion

8.1.Summary

Traffic collisions have been a significant problem for decades, because they are a major cause of deaths and injuries, as well as a cause of significant traffic delays and congestion. Although environmental and vehicle reasons have been found to contribute to collision occurrences, traffic dynamics and human error are the most dominant origins of such events.

To increase road safety, the automotive and research community has recently begun to move towards autonomous or robotic vehicles, which remove the human element from the task of driving. A safe navigation is ensured by the motion planning module which is part of an AV's software architecture and aims at providing a collision free path for the vehicle to follow. Within the motion planning module, an AV creates a trail to follow, specifies the necessary manoeuvres to efficiently follow the trail, operates among obstacles and controls its trajectory according to its dynamics. However, current AV applications cannot cope well with the complexity of the traffic environment which has led them to cause several collisions. This is because in most of the literature regarding the anticipation of risk from AVs, the traffic participants are considered to move independently and the traffic context (e.g. traffic rules or lane structure) is not considered.

Over the past decades, the advances in ITS and data collection technologies have initiated research on the identification of specific traffic conditions which potentially cause collisions in real time. The objective of real-time collision prediction is to classify current traffic into collision-prone traffic and safe traffic, based on the comparison of traffic conditions that were dominant just before historical collision occurrences and during normal operations. More specifically, ITS experts investigate real-time traffic, geometry or weather characteristics to define if the conditions on a link could potentially cause enough traffic turbulence to trigger a collision.

Potentially these network-level collision prediction (NLCP) approaches could enhance the perception and risk assessment modules of AVs. However, due to the sparse nature of collision events, models tend to over-represent safe traffic conditions

unlike collision-prone conditions and this hinders the performance of collision prediction classifiers. Additionally, the traffic conditions based on which real-time collision prediction models are developed, correspond to historical collision times which may falsely be reported. Lastly, the traffic data utilised currently for real-time prediction are aggregated usually at 5-minute intervals and make predictions for a 15-minute temporal horizon, something which does not correspond to the second-level safety analysis which needs to be undertaken by AVs.

As NLCP is believed to enhance the awareness level of AV risk assessment, this research pursued to bring together real-time collision prediction with risk assessment modules of AVs.

Initially, after reviewing current AV motion planning and risk assessment methods applied in the literature, interaction-aware models were identified as the best to accommodate network-level information, because they take context into account and are easily extendable. For the scope of this thesis, an interaction-aware model based on a Dynamic Bayesian Network was explicitly described in order to integrate NLCP with the estimation of the vehicle-level safety context, the motion properties of adjacent vehicles and an AV's sensors measurements.

The main part of this thesis was dedicated to improving network-level classifiers in order for them to contribute to the safety assessment of AVs. The problem of the asymmetry between safe and collision-prone traffic cases in collision prediction databases was tackled by introducing imbalanced learning techniques. Such techniques take into account the imbalance of the dataset and produce classifiers which can efficiently predict cases belonging to both classes (i.e. safe and collision-prone traffic). Two imbalanced learning approaches were utilised; undersampling the majority class (i.e. safe traffic) and its integration with oversampling the minority class (i.e. collision-prone traffic). An additional method to counterbalance the disproportion of safe and collision-prone traffic was the utilization of Random Forests (RFs), an ensemble classifier which literature suggested that works well with imbalanced datasets. In order to not only have individual predictions but also to make classification results more interpretable, probabilistic machine learning

classifiers such as Relevance Vector Machines (RVMs) and Gaussian Processes (GPs) were utilised. This way, a probability can be associated with each prediction.

The methods used to enhance NLCP were originally applied on three datasets. The first dataset contained highly aggregated traffic data in 15-minute intervals corresponding to historical collisions and normal traffic on two motorways (i.e. M1 and M62) and two A-roads (i.e. A3 and A12) of the Strategic Road Network of England. In order to investigate the performance of the classifiers on disaggregated data, the second dataset contained traffic data corresponding to the time interval 5 minutes before collisions occurred at two major arteries inside the metropolitan area of Athens, Greece.

To correct the reported collision time error in existing collision databases, traffic microsimulation was utilised and traffic conflicts were extracted for a segment of the M62 motorway. The traffic conditions were obtained through the VISSIM microsimulation software and the conflicts were derived from SSAM, a post-processing tool which investigated simulated vehicle trajectories and filtered them to obtain traffic conflicts. The simulation was intensively calibrated in order to represent real-world traffic, based on the traffic measurements and travel times of M62. This led not only to the acquisition of highly disaggregated traffic data which were used for traffic safety analysis, but to timestamped conflicts with highly representative traffic conditions too. The 30-second raw data obtained from microsimulation were further aggregated in 1-minute, 3-minute and 5-minute intervals, so as to investigate the effect of temporal aggregation on the classification results.

Six classification algorithms were tested in general for the task of distinguishing between hazardous and safe traffic conditions; k-Nearest Neighbours (kNN), a simple data-adaptive classifier, RVMs and GPs which provide probabilistic predictions, RFs to test ensemble learning and Support Vector Machines (SVMs) along Neural Networks (NNs) which are powerful classifiers and have been frequently used in real-time collision prediction. To avoid overfitting, all the algorithms were optimised using 10-fold cross validation before their application.

Moreover, 2/3 of the datasets were used for learning and the remaining l/3 was used for testing. The imbalanced classifiers constructed a balanced dataset i.e. containing roughly the same number of collision-prone and safe traffic conditions and were tested on the original imbalanced data.

The results indicated the importance of the data quality and the choice of temporal aggregation intervals for the correct identification of collision-prone traffic, as well as the power of imbalanced learning in acquiring better classification results.

Regarding the 15-minute UK dataset, it was found that although the original dataset could not predict collision cases efficiently, its treatment with undersampling techniques resulted in the identification of 67% of collisions with a relatively low false alarm rate when RFs were utilised. Undersampling also worked better with the 5-minute data of the Athens dataset, achieving correct identification of 70% of the collision-prone conditions with GPs with only 8% false alarms. This result demonstrated that even though traffic data before a collision occurrence include noise, the treatment with imbalanced learning can achieve excellent classification results. Comparing the results of the classifiers with the findings of existing literature, it was found that the developed classifiers when integrated with imbalanced learning techniques outperform existing approaches. Therefore, this makes them eligible for real-time collision prediction. However, as 15 minutes is a high temporal aggregation interval, the use of 15-minute data is not recommended.

The importance of imbalanced learning was further justified in the simulated dataset. The results from the simulated dataset confirmed existing literature, as the temporal aggregation interval increased the ratio between recall (i.e. the correct identification of conflict-prone conditions) and the false alarm rate increases achieving robust classification results. Nevertheless, when imbalanced learning aided the classifiers, an increase in the classification results was obvious. This was especially the case when the integration of oversampling the minority class along with undersampling the majority class was utilised. kNN, RFs and SVMs achieved the best results in recognizing traffic conditions just before conflicts and it was demonstrated that, even

when raw 30-second data were used, the classification performance was better than the state-of-the-art in the literature.
For the final part of the thesis, the results of the best network-level classifiers were imported to the recommended interaction-aware model to detect the impact on the estimation of the probability of a road user being a hazard to an AV, given the NLCP information.

Initially, two artificial scenarios with random number of vehicles were tested with two NLCP classifiers (i.e. the kNN classifier under SMOTE-ENN utilised with 30-second simulated data and the GP classifier of the 5-minute Athens dataset utilised with undersampling). It was found that, if the collision prediction indicated collision- or conflict-prone conditions, then hazardous road users were easier identified. This showed the potential of the approach in enhancing AV risk assessment in the case of "hazardous" traffic conditions.

Vehicle-level data were obtained from the Loughborough University instrumented vehicle during a one-hour driving trip on the M1 motorway (J23-J18). These data were used to validate the fact that network-level collision information can assist in identifying dangerous road users. Using only the radar data from the trip, TTC values were estimated and used to identify potential safe and dangerous road users. After importing the vehicle-level information, into the interaction-aware DBN, the probability of a dangerous road user was estimated for every measurement cycle in the cases of conflict-prone and safe traffic conditions.

Using real-world data and the classifiers trained on highly disaggregated traffic data, it was validated that when traffic conditions are classified as hazardous, then the identification of dangerous traffic participants is significantly enhanced. Even when highly disaggregated traffic data (i.e. 30-second) were utilised, the identification of a traffic participant posing a threat to the ego-vehicle was significantly assisted. However, when NLCP indicated safe traffic, the results demonstrated that dangerously driving traffic participants were identified better when network-level information was not provided.

Finally, suggestions were made on the use of network-level information to enhance safe motion planning and to assist in the case of a failure or obstruction of the sensing system of AVs, thus improving mobility and road safety in general.

8.2.Discussion

8.2.1. *Discussion framework*

This thesis attempted to integrate real-time NLCP models with collision risk assessment models for autonomous vehicles. This has been achieved by concentrating on two focal points:

- The enhancement of real-time NLCP models regarding their prediction horizon, the use of noisy and disaggregated data as well as the imbalance of current real-time NLCP datasets.
- The benefits obtained by taking real-time NLCP information into account in autonomous vehicles risk estimation models

Appropriate machine learning models were employed to distinguish between safe and collision-prone traffic conditions using a variety of traffic, collision and conflict data as well as a mixture of data aggregation intervals in Chapter 6. Next, the results from the most efficient NLCP models were input in the proposed interaction-aware model which utilises network- and vehicle-level information and the results were presented in Chapter 7. This section aims to critically synthesize and discuss the results regarding both real-time NLCP as well as its conjunction with AV risk assessment models, so as to provide a better understanding. Following the discussion on each subject, implementation recommendations are also presented.

8.2.2. *Discussion on the developed real-time collision/conflict prediction models*

The machine learning models developed in this PhD were applied on real-world datasets from the UK and Greece in Chapter 6, as well a large simulated dataset with highly disaggregated traffic and conflict data obtained from traffic microsimulation software. To overcome the imbalance problem between safe and collision-prone cases in current NLCP datasets, imbalanced learning was applied. The estimation results between the original datasets and the ones treated with imbalanced learning

were compared and contrasted so as to observe the effect of imbalanced techniques on the identification of collision-prone traffic

Taking into account that NLCP literature was found (i.e. in Chapter 3) to be moving towards machine learning approaches, six different algorithms were tested for the purposes of this thesis. In contrast with recent on real-time NLCP studies (e.g. Xu et al., 2013, Hossain and Muromachi, 2013, Lin et al., 2015; Sun and Sun, 2015;Xu et al., 2015b), which compared their proposed approaches with only one or two different methodologies, this PhD study offers a comprehensive comparison between initially used (i.e. RVMs, GPs and RFs) and previously studied methodologies in real-time NLCP (i.e. kNN, NNs and SVMs).

Although RFs were primarily utilised for variable selection in previous literature (e.g. Hossain and Muromachi, 2013), it was found that its classification performance is powerful, regardless of the temporal aggregation of traffic data in terms of identifying hazardous traffic conditions. Therefore, its utilization for real-time NLCP is suggested by the results of this study.

To address the issue of model interpretability and transferability that characterises machine learning approaches, the probabilistic machine learning techniques of RVMs and GPs were employed for the first time in real-time NLCP studies. These approaches can correlate a "stationary" prediction regarding the occurrence of collision-prone traffic conditions in real-time with the probability that the prediction is correct. Hence, predictions with larger confidence can be performed and broadcasted to traffic participants and autonomous vehicles. However, in the majority of datasets utilised by this study, these two approaches did not yield sufficient classification results. The performance of RVMs can be justified from its general instability and suboptimal learning of kernel parameters as suggested by Chen et al., (2014) as well as its poor performance on large sample sizes as suggested by Yu et al., (2004). The performance of GPs is probably a result of their computational limitations (Rasmussen, 2006) as well as their problematic conjunction with the imbalanced learning technique of SMOTE (Elrahman and Abraham, 2013).

In consistency with previous studies on real-time NLCP (Li et al., 2008, Yu and Abdel-Aty, 2013), SVMs result in good classification performance. Additionally, NNs perform well in the simulated dataset, however fail to work well with the smaller sample sizes of the UK and Athens datasets, a fact which can be found in the findings of previous studies (Karlaftis and Vlahogianni, 2011).
The most interesting findings relate to the use of imbalanced learning and traffic microsimulation for developing real-time NLCP models.

The use of microsimulation for safety studies is generally debated in the literature, because collisions cannot be obtained inside the simulation and it is difficult to correlate the obtained conflicts with actual collisions. Through intensive calibration and validation for both the traffic conditions and the obtained conflicts, it was demonstrated that datasets containing realistic conflict-prone conditions can be formulated. These datasets contain traffic conditions labelled as conflict-prone or safe and can be utilised for real-time conflict prediction in a similar procedure as real-time collision prediction datasets are employed. The benefit is that the documented conflict events are explicitly time-stamped and described as they are the output of computer software. Moreover, instead of utilizing microsimulation exclusively for before-after studies regarding the total number of collisions at a site based on the number of simulated conflicts (e.g. Shahdah et al., 2015), the identification of pre-conflict conditions is investigated.

Existing literature on real-time collision prediction suggests that traffic data should be aggregated at 5-minutes intervals and the time interval corresponding to 5-10 minutes before a collision should be employed to build NLCP models. The reason behind this suggestion is that raw traffic data from loop detectors (i.e. 20-second or 30-second data) and traffic data 0-5 minutes before a collision are noisy and do not facilitate a timely intervention from traffic management authorities. The superiority of traffic data aggregated in 5-minute intervals was validated in Chapter 6, where it was demonstrated that using that temporal aggregation interval led to the best classification results. However, the use of imbalanced learning techniques resulted in significant enhancements for the classifiers employing highly disaggregated traffic

data or traffic data obtained 0-5 minutes before collision events. As shown in Tables 6.16 and 6.23, as well as Figure 6.13, classifiers utilizing traffic data 0-5 minutes before a collision as well as simulated traffic data even at 30-second or 1-minute intervals performed better than existing literature in identifying collision or conflict-prone traffic with a low false alarm rate. Therefore, the use of imbalanced learning and especially the undersampling of the majority class, should be employed in future NLCP studies even when highly disaggregated or noisy data are available.

The results from the models developed in this thesis offer a new awareness level for the use of traffic data and their temporal aggregation. This thesis utilised real-world traffic data both highly aggregated as well as disaggregated five minutes before collision occurrences. The classifiers developed here could become a tool guide for Active Traffic Management (ATM) agencies. These agencies could apply the developed classifiers utilizing traffic data at a preferred temporal aggregation and issue warnings if needed. Specifically, if traffic conditions are classified as collision-prone, then warning messages could be presented through VMS or broadcasted to the AVs communication system, prompting the passenger to take control until the network-level prediction horizon is exceeded and safety is ensured. Moreover, as imbalanced learning and especially undersampling was found to contribute to the improvement of classification results, the use of imbalanced learning is highly suggested as collision-prone conditions detection significantly improved. The application of NLCP models is not limited to AVs only, as these predictions could initiate traffic calming schemes, such as variable speed limits (VSL). The VSL schemes work on the basis that, if external conditions (such as adverse weather or road works) exist on a motorway, then speed limits are adjusted appropriately to control for congestion and traffic violations. On the same principle, if the classifiers indicate traffic conditions which are collision-prone, then VSL could be initiated in order to decrease the general vehicle speed and thus traffic violations.

Furthermore, as simulated data were utilised in this paper, traffic agencies could benefit in areas where no data collection measuring devices exist or traffic data are sparsely collected and aggregated. As the use of microsimulation in this thesis was aimed at looking at the pre-conflict conditions, traffic environments of interest could

be simulated and validated in order to identify conflict-prone conditions. After the detection of conflict-prone traffic, ATM could issue warnings about potential conflicts, in order to enhance driver attention on the road. For instance, an early-warning for detecting conflict-prone traffic conditions could be broadcasted first and could be followed by a collision-prone traffic warning if the conflict-prone and collision-prone traffic classifiers indicate abnormal traffic.

8.2.3. *Discussion on the integrated collision risk model*

The method developed in section 4.3 of the methodology and evaluated in Chapter 7 is an initial step in the incorporation of NLCP models into risk assessment modules for AVs. The evaluation in Chapter 7 demonstrated that, when "dangerous" (i.e. conflict or collision-prone) traffic conditions were detected by the network-level classifiers, then the probability of detecting a vehicle-level "dangerous" traffic user was enhanced. On the other hand, in cases when traffic conditions are deemed safe, network-level information did not provide assistance in identifying hazardous traffic participants. As the model is constructed in order to resemble human-like driving and perception, safe traffic conditions lead the model to be biased towards safe traffic participants. As a result, because the vehicle-level safety context is dependent on the network-level collision risk, the identification of dangerous vehicles is hindered. More sophisticated functions which can be learned from data so as to take into account network-level information could result in better probability estimation for the cases when traffic conditions are classified as safe. As real-world vehicle-level data were obtained from only one driving trip, a larger dataset would further enhance the performance of the model.

The model was not compared with other state-of-the-art algorithms in this thesis. As also indicated by Lefèvre, (2012), such collision risk assessment models need to be evaluated on the same dataset. Although there are publicly available datasets such as the NGSIM dataset (FHWA, 2006) or the Warrigal dataset (Ward et al., 2014b), these do not meet the requirements for the model proposed in this PhD. This is due to the fact, that NGSIM contain network- and vehicle-level data from only 15 minutes and the Warrigal dataset contains vehicle-level information from a mining site.

As the literature review on motion planning methods revealed in Chapter 2, dealing with complex traffic scenarios and multiple obstacles is troublesome for AVs, the next paragraphs will describe scenarios where the methods presented in this thesis could become of assistance.

Using NLCP information, AVs could adjust their motion planning routine. Risk assessment is part of the manoeuvre planning routines as it was found in Chapter 2 and as a result the most prominent advantages relate to adjusting or changing the current manoeuvre planned by an AV. For example, AVs could slow down when entering road segments which are predicted to have collision-prone traffic. This would benefit them in two ways: i) by slowing down, a safer journey would be assured and ii) AVs would have more time to detect and avoid dangerous traffic users with the enhanced vehicle-level safety context probability. Moreover, when network-level information indicated that the traffic conditions at a road segment could cause a collision, an alternative route or lane could be chosen to avoid hazardous encounters.

Autonomous lane changing (Lefèvre,2012; Thrun 2010) is an important aspect of autonomous vehicles that potentially could prevent a number of collisions by changing the trajectory that a vehicle is following in order for it not to collide with an obstacle. This aspect of AVs was not taken into account when developing the DBN, as this work's primary contribution was a methodology which brings together traffic engineering and AV risk assessment. However, autonomous lane changing usually requires harsh accelerations or decelerations which would not provide a sufficient level of comfort to the passengers. NLCP information could enable slower speeds and hence slower accelerations as discussed in the previous paragraph. In that way the comfort of passengers during these harsh acceleration or deceleration events would be enhanced.

AVs require a lot of information from multiple sensor platforms (Polychronopoulos et al., 2007, Huang, E. et al., 2013). Most of AVs utilise cameras (Bertozzi et al., 2000) and laser scanners (Jiménez et al., 2012, Mertz et al., 2013) to scan the surroundings and estimate a safe path for the vehicle. However, it is unknown how

AVs are going to cope with system failure (Koopman and Wagner, 2016, Dixit et al., 2016). In that perspective, the method developed in this project and the incorporation of network-level information in general could be advantageous. As NLCP utilises more macroscopic data compared to the data received by the sensor systems of AVs and have sub-second frequency, the network-level prediction will be known a-priori for specific time periods. Hence, if the majority of the sensing systems fail, then, according to the network-level information, the AV can slow down, as in the case of collision-prone traffic conditions indicated in the previous paragraph, until it reaches a safe point or the system error is fixed. This applies also in cases where the sensor system and especially the vision-based systems become obstructed (e.g. from a big truck in front of the vehicle or from adverse weather conditions). Consequently, NLCP could assist not only the identification of "dangerous" road users but could act as a safety net for all the motion planning levels i.e. from routing to manoeuvre planning.

8.2.4. *Discussion synopsis*

This section critically discussed the results presented in Chapters 6 and 7 of this thesis. Regarding the NLCP models, the most interesting findings relate to the power of imbalanced learning, the employment of RFs as well as the use of highly disaggregated and simulated data. As traffic data were aggregated in different temporal aggregation intervals and real-time conflict detection classifiers were developed, ATM systems could benefit in terms of issuing warnings for conflict and collision-prone traffic, as well as ensuring proper traffic flow through variable speed limits according to the road safety level at a specific time moment. The discussion of the integrated collision risk model for AVs, demonstrated that the findings of this thesis could enhance AV motion planning and general road safety. Network-level predictions utilise aggregated data at higher temporal interval than the frequency of the sensors of an AV and hence provide a broader perception horizon. If NLCP is available, then AVs could reduce speeds, change routes or prompt a passenger to take control in order to ensure a safe journey, even when other sensor systems fail.

8.3.Contribution to knowledge

This research has provided new methodological and quantitative outcomes which could enhance future safety analyses. The main contributions to knowledge are:

1. The integration of NLCP along vehicle-level risk assessment by AVs

Perhaps the most interesting contribution of the present PhD thesis is the proposed interaction-aware model, which integrates network-level and vehicle-level collision prediction. A part of this thesis is dedicated to estimating the probabilities of vehicle-level collision risk, given NLCP and it was demonstrated that a quicker identification of dangerous traffic participants is possible. As also described in the methodology chapter, the way in which the DBN model is constructed, makes it easily extensible to accommodate more sophisticated probability functions. These functions could explicitly consider the dynamics and the behaviour of traffic participants, as long as they are appropriately defined. The fact that the estimated model is mathematically sound supports the argument that it could become a tool which combines motion prediction and risk assessment by AVs with real-time NLCP as studied in the past decades by traffic engineers. The integrated approach could enhance the perception horizon and safety assessment by AVs, as pointed out in previous sections of this thesis.

2. The effect of imbalanced learning in tackling the corresponding problem of existing collision-traffic databases

This research extensively utilised imbalanced learning techniques to obtain classifiers which perform well, both in the task of identifying dangerous traffic conditions as well as in the identification of normal traffic operations. Imbalanced learning achieved high classification rates with a few false alarms, even when highly aggregated or highly disaggregated traffic data were utilised. This is especially important for real-time safety assessment, as traffic management agencies need to timely identify hazardous or safe traffic in a timely manner and perform the necessary actions to ensure safe and smooth traffic flow. Although the initial approach of safety experts should be to build classifiers using the original datasets, the aiding nature of imbalanced learning could cope with inherent difficulties in

acquiring balanced classification results. As it was demonstrated in this thesis, even when 15-minute traffic data were utilised, some of the classifiers integrated with imbalanced learning managed to detect the majority of the collision-prone traffic conditions.

3. The utilization of simulated traffic and conflict data for use in real-time safety assessment

One of the datasets presented in the present thesis was acquired from traffic microsimulation. Although the use of simulation in safety analysis has been generally debated, the present study utilised it in a different way than the literature has. Usually, simulated traffic and the obtained conflicts from SSAM are employed for before-after analyses of interventions at intersections or motorways, as well as the estimation of the number of collisions based on the number of simulated conflicts. In the present thesis, the traffic conditions before each conflict event were sampled in order to develop the classifiers. More specifically, the classifiers aimed at identifying conflict-prone conditions within the simulation model to enhance understanding of the precise traffic conditions that trigger conflict events. Thus, the scope of microsimulation was not the estimation of collisions or the validation of conflicts happening at a specific location, but rather the acquisition of disaggregated traffic data at a very high level and the tackling of the misreported collision time in collision databases. This enables traffic data collection in cases where traffic measurements are unavailable or highly aggregated (such as the 15-minutes data utilised in this work). Moreover, it enables the use of microsimulation software for real-time safety assessment as the results from the classifiers could be used in order to test if real-world conflict-prone conditions are correctly identified.

4. The utilization of disaggregated traffic data close to the collision event time

Most of the classifiers developed and presented in this thesis made use of traffic data just before a collision or a conflict. In the Athens dataset, traffic data were aggregated in 5-minute intervals but the data corresponding to just 5 minutes before collisions were employed. This is not the first time that traffic data have been aggregated at 5-minute intervals, and used in real-time collision prediction. However,

the vast majority utilises traffic data 5-10 minutes prior to the collision events. Moreover, even when 30-second or 1-minute simulated traffic data were utilised, it was demonstrated that traffic conflicts could also be efficiently detected if the dataset was treated with imbalanced learning techniques. Traffic data from a time interval close to the collision or conflict event include random noise. However, it is possible to build effective classifiers which can distinguish between collision events and normal traffic operations. Of course, predictions with such prediction horizon (e.g. 30 seconds to 5 minutes before collisions) might not give enough time to traffic management agencies to intervene and prevent collisions nowadays. However, with the advances in communication technologies, except for the AV application which was demonstrated in this work, vehicle-to-vehicle (V2V) or vehicle-to-infrastructure (V2I) communication applications could bring about the application of such limited time predictions.

8.4.Study limitations

The research presented in this thesis is not without shortcomings. It includes limitations, the most important of which are outlined below:

- **Collision time inaccuracies for the real-world datasets**: The 15-minute UK dataset and the Athens dataset incorporate the error regarding the exact time of the collision. As a result, the traffic measurements which were used as collision-prone might not be as representative of the traffic conditions leading to the collision as possible.

- **Traffic data aggregation:** Traffic data for the UK dataset as well as the Athens dataset were provided in 15-minute and 5-minute averages. Such data are not detailed enough to describe the pre-collision conditions. Furthermore, more detailed data could lead to a better calibrated simulation model to obtain the conflict-prone conditions used in this work.

- **On-site conflicts validation:** The traffic microsimulation model was intensively calibrated regarding the traffic conditions (i.e. with the use of the GEH-statistic) the travel times corresponding to the real-world traffic

measurements and the crash potential index of the NGSIM data. However, the traffic conditions leading to conflicts were not validated using conflict observations from the same site that was simulated. This could lead to discarding several traffic conditions and the corresponding "conflicts", which are not consistent with real-world observations.

- **Omitted variables:** The developed NLCP models did not consider some significant factors which have been found to contribute to collision occurrences, such as weather and lighting conditions, time of the day and day of the week, road pavement conditions and others. The inclusion of these variables in the analysis would describe collision and normal traffic more explicitly and hence may have improved the classification results.

- **Spatial and temporal transferability:** The classifiers developed in this work were tested in a part of the original dataset. Testing the learned classifiers of the obtained datasets on different motorways and during different time periods could enhance the results and give a clearer picture of their performance.

- **Variable selection:** For the task of classification, none of the available variables were excluded. If variable or feature selection was utilised, the classification results might have been improved.

- **Limited vehicle-level data:** The influence of NLCP on vehicle-level risk assessment was estimated using artificial data and data from one driving trip. For the full validation of the model, data from a variety of sensors need to be employed in order to investigate a more realistic impact of real-time collision prediction on vehicle-level risk. Moreover, if a large vehicle-level dataset is available, then the functions which indicate a dangerous road user or dangerous vehicle kinematics could be learned from the data. The indicative functions used in this thesis are a simple approach in order to demonstrate the potential of such a model. Data driven functions based on gap-acceptance

models or vehicle trajectories would definitively enhance the quality of the DBN outcomes and consequently vehicle-level safety assessment.

- **Limited traffic information provided to the DBN:** NLCP was demonstrated to assist vehicle-level risk assessment, especially when traffic conditions were deemed as collision-prone. However, it is not known if information only on occupancy, flow, and average speeds without supplementary variables such as weather conditions or road geometry, can provide assistance online to AV motion planning routines.

- **Limited testing of the DBN:** Only a relative comparison of the estimated probability of vehicle-level risk with and without the NLCP information was performed in this thesis. In order to demonstrate the full potential of the proposed DBN methodology the full network needs to be calibrated and utilized online for AV motion planning using a test vehicle.

- **Omission of traffic rules and signs in the DBN:** The proposed DBN model characterises the vehicle-level safety context based only on the kinematic properties of the motion of vehicles. Nevertheless, in everyday traffic, users are frequently described as both safe and dangerous, according to the obedience to traffic rules and traffic signs.

- **Online calibration of the DBN:** The probabilities from the proposed DBN model were estimated after the driving session. In order to test its capabilities for real-time collision risk assessment, its online application should be tested.

8.5.Extensions and suggestions for future research

The work that has been presented in this thesis, both in terms of the NLCP as well as its integration with vehicle-level risk assessment in the interaction-aware DBN model, can easily be extended and transferred to other study areas. Building more robust network-level classifiers, as well as expanding and enhancing the DBN model, can increase the accuracy and safety perception of AV planning modules. Considering the limitations of the current study as described in the previous sub-section, several improvements can be made.

Regarding NLCP the current work utilised six classifiers (i.e. kNN, RVMs, SVMs, GPs, RFs and NNs) which are all frequently used in the machine learning literature. An extension to this part could be to incorporate alternative machine learning and data mining techniques. More efficient and newer techniques such as Deep Learning, Deep Neural Networks, ensembles of SVMs, information-theoretic clustering and feature selection can be tested and compared with the classifiers presented in this work. Furthermore, kernel-based methods (such as RVMs and SVMs) can be further enhanced with transport-related kernel functions in order to obtain better classification results. Moreover, it should be interesting to explore the use of raw traffic data to detect collision-prone conditions as the utilised classifiers using 30-seconds simulated data resulted in good classification rates after the treatment with imbalanced learning. The incorporation of real-time weather characteristics or pavement conditions could also be taken into consideration in future work to explicitly describe the conditions before a collision. The application of the cost-sensitive classification, which was an imbalanced learning technique not utilised in this study, should also be an interesting extension of the current work.

As far as the simulation modelling is concerned, simulation of autonomous or connected vehicles could be incorporated in the models along with their sensor systems. This will result in conflict-based models which could be easily applied into the interaction-aware model because they would be based on representative traffic conditions in the presence of automated vehicles.

Regarding the incorporation of NLCP into AV risk assessment modules, an obvious extension is the utilization of vehicle-level data acquired online, so as to have information on all four layers of the proposed DBN and enable a ground truth validation of the DBN results, thus enhancing the robustness of the model. As mentioned in the limitations section, the functions indicating dangerous road users according to their kinematics should be learned from sensor data and vehicle trajectories. Moreover, the obedience to traffic rules, lane markings and traffic signals or signs should be researched in order to cope with the traffic environment in every detail.

Finally, as risk assessment is a part of AVs motion planning routine, the incorporation of NLCP in path or trajectory planning could be investigated. For example, imagine a Rapidly exploring Random Tree (RRT) which grows edges according to the network and vehicle-level collision risks or a costmap built according to different NLCP risk levels. The potential of collaboration between NLCP and AV risk assessment methods was slightly shown in this thesis, but relevant research can certainly lead to enhanced safety in a highly automated traffic environment in the future.

References

Abdel-Aty, M. a. & Pemmanaboina, R. (2006) Calibrating a Real-Time Traffic Crash-Prediction Model Using Archived Weather and ITS Traffic Data. *IEEE Transactions on Intelligent Transportation Systems*. 7 (2), 167–174.

Abdel-aty, M. & Pande, A. (2005) 'The viability of real-time prediction and prevention of traffic accidents', in Al-Qadi, Tarek Sayed, Alnuaimi, & Massad (eds.) *Efficient Transportation and Pavement Systems*. Taylor & Francis Group, London. pp. 215–226.

Abdel-Aty, M. & Pande, A. (2007) Crash data analysis: collective vs. individual crash level approach. *Journal of safety research*. 38 (5), 581–7.

Abdel-Aty, M. & Pande, A. (2005) Identifying crash propensity using specific traffic speed conditions. *Journal of safety research*. 36 (1), 97–108.

Abdel-Aty, M., Pande, A., Das, A. & Knibbe, W.J. (2008) Assessing safety on Dutch freeways with data from infrastructure-based intelligent transportation systems. *Transportation Research Record*. 2083 (2083), 153–161.

Abdel-Aty, M., Pande, A. & Hsia, L. (2010) The concept of proactive traffic management for enhancing freeway safety and operation. *ITE Journal (Institute of Transportation Engineers)*. 80 (4), 34–41.

Abdel-Aty, M., Pande, A., Hsia, L.Y. & Abdalla, F. (2005) 'The Potential for Real-Time Traffic Crash Prediction', in *ITE Journal on the web*. 2005 pp. 69–75.

Abdel-Aty, M., Uddin, N., Pande, A., Abdalla, F. & Hsia, L. (2004) Predicting Freeway Crashes from Loop Detector Data by Matched Case-Control Logistic Regression. *Transportation Research Record*. 1897 (1), 88–95.

Ackerman, E. (2016) Fatal Tesla Self-Driving Car Crash Reminds Us That Robots Aren’t Perfect -. IEEE Spectrum

Agamennoni, G., Nieto, J.I., Nebot, E.M. & Member, S. (2012) Estimation of Multivehicle Dynamics by Considering Contextual Information. *IEEE Transactions on robotics*. 28 (4), 855–870.

Ahmed, M. & Abdel-Aty, M. (2013) A data fusion framework for real-time risk assessment on freeways. *Transportation Research Part C: Emerging Technologies*. 26, 203–213.

Ahmed, M., Abdel-Aty, M. & Yu, R. (2012a) A Bayesian Updating Approach for Real-Time Safety Evaluation using AVI Data. *Transportation Research Record,Journal of the Transportation Research Board*. 2450.

Ahmed, M., Abdel-Aty, M. & Yu, R. (2012b) Assessment of the Interaction between Crash Occurrence , Mountainous Freeway Geometry , Real-Time Weather and AVI Traffic Data. *TRB 2012 Annual Meeting*. 2450.

Ahmed, M.M. & Abdel-Aty, M. (2012) The Viability of Using Automatic Vehicle Identification Data for Real-Time Crash Prediction. *IEEE Transactions on Intelligent Transportation Systems*. 13 (2), 459–468.

Alin, A., Butz, M. V. & Fritsch, J. (2012) 'Incorporating environmental knowledge into Bayesian filtering using attractor functions', in *IEEE Intelligent Vehicles Symposium*. [Online]. 2012 pp. 476–481.

Amundsen, F. & Hyden, C. (1977) Proceeding of First Workshop on Traffic Conflicts. First Workshop on Traffic Conflicts.

Aoude, G., Luders, B.D., Levine, D.S. & How, J.P. (2010a) 'Threat-aware path planning in uncertain urban environments', in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. October 2010 pp. 6058–6063.

Aoude, G.S., Luders, B.D., How, J.P. & Pilutti, T.E. (2010b) 'Sampling-based threat assessment algorithms for intersection collisions involving errant drivers', in *IIFAC*

Symposium on Intelligent Autonomous Vehicles. 2010.

Archer, J. (2005) *Indicators for traffic safety assessment and prediction and their application in micro-simulation modelling: A study of urban and suburban intersections*. Doctoral Thesis, KTH Royal Institute of Technology, Sweden.

Archer, J. & Kosonen, I. (2000) The potential of micro-simulation modelling in relation to traffic safety assessment. *ESS conference proceedings*.

Archer, J. & Young, W. (2010) The measurement and modelling of proximal safety measures. *Proceedings of the Institution of Civil Engineers*. Transport (November TR4), 191–201.

Armand, A., Filliat, D. & Ibanez-Guzman, J. (2014) 'Ontology-based context awareness for driving assistance systems', in *IEEE Intelligent Vehicles Symposium*. 2014 pp. 227–233.

Asimov, I. (1950) *I , Robot*. Spectra.

Astarita, V., Giofre, V., Guido, G. & Vitale, A. (2011) Investigating road safety issues through a microsimulation model. *Procedia - Social and Behavioral Sciences*. 20226–235.

Bacha, A., Bauman, C., Faruque, R., Fleming, M., Terwelp, C., Hong, D., Wicks, A., Alberi, T., Anderson, D., Cacciola, S., Currier, P., Dalton, A., Farmer, J., Hurdus, J., Kimmel, S., King, P., Taylor, A., Covern, D. Van & Webster, M. (2008) Odin : Team VictorTango ' s Entry in the DARPA Urban. *Journal of Field Robotics*. 25 (8), 467–492.

Bahram, M., Hubmann, C., Lawitzky, A., Aeberhard, M. & Wollherr, D. (2016) A Combined Model- and Learning-Based Framework for Interaction-Aware Maneuver Prediction. *IEEE Transactions on Intelligent Transportation Systems*. 17 (6), 1538–1550.

Bandyopadhyay, T., Won, K.S., Frazzoli, E. & Hsu, D. (2012) 'Intention-Aware Motion Planning', in *Proc. of the 10th Workshop on the Algorithmic Foundations of Robotics*. [Online]. 2012 pp. 475–491
.
Barcelo, J. (2011) NULL. *Fundamentals of Traffic Simulation*. Vol. 108 Springer-Verlag New York.
Batista, G.E.A.P.A., Prati, R.C. & Monard, M.C. (2004) A Study of the Behavior of Several Methods for Balancing Machine Learning Training Data. *SIGKDD Explor. Newsl.* 6 (1), 20–29.

Ben-Hur, A. & Weston, J. (2010) A user's guide to support vector machines. *Methods in molecular biology (Clifton, N.J.)*. 609,223–239.

Bertozzi, M., Broggi, A. & Fascioli, A. (2000) Vision-based intelligent vehicles : State of the art and perspectives ☪. *Robotics and Autonomous Systems*. 321–16.

Bessiere, P., Mazer, E., Ahuactzin, J.-M. & Mekhnacha, K. (2013) *Bayesian Programming*,CRC Press .

Bishop, C.M. (2006) *Pattern Recognition and Machine Learning*. Springer New York.

Bohren, J., Foote, T., Keller, J., Kushleyev, A., Lee, D., Stewart, A., Vernaza, P. & Satterfield, B. (2008) Little Ben : The Ben Franklin Racing Team ' s Entry in the 2007 DARPA Urban Challenge. *Journal of Field Robotics*. 25 (9), 598–614.

Bonsall, P., Liu, R. & Young, W. (2005) Modelling safety-related driving behaviour - Impact of parameter values. *Transportation Research Part A: Policy and Practice*. 39 (5), 425–444.

Brand, M., Oliver, N. & Pentland, A. (1997) Coupled hiddenMarkov models for complex action recognition. *International Conference on Computer Vision and*

Pattern Recognition. 1–6.

Brechtel, S., Gindele, T. & Dillmann, R. (2014) 'Probabilistic Decision-Making under Uncertainty for Autonomous Driving using Continuous POMDPs', in *2014 IEEE 17th International Conference on Intelligent Transportation Systems (ITSC)*. 2014 Qingdao, China: . pp. 392–399.

Breiman, L. (1996) Bagging Predictors. *Machine Learning*. 24 (421), 123–140.

Breiman, L. (2001) Random Forests. *Machine learning*. 45.15–32.

Broggi, A., Medici, P., Zani, P., Coati, A. & Panciroli, M. (2012) Autonomous vehicles control in the VisLab Intercontinental Autonomous Challenge. *Annual Reviews in Control*. 36 (1), 161–171.

Brown, M., Funke, J., Erlien, S. & Gerdes, J.C. (2017) Control Engineering Practice Safe driving envelopes for path tracking in autonomous vehicles. *Control Engineering Practice*. 61,307–316.

Brown, T.L. (2005) Adjusted Minimum Time-To-Collision (TTC): A Robust Approach to Evaluating Crash Scenarios. *DSC North America*. 40–48.

Burns, L.D. (2013) A vision of our transport future. *Nature*. 497, 181–182.

Carin, L. & Dobeck, G.J. (2003) Relevance vector machine feature selection and classification for underwater targets. *Oceans 2003. Celebrating the Past ... Teaming Toward the Future (IEEE Cat. No.03CH37492)*.

.

Chawla, N. V., Bowyer, K.W., Hall, L.O. & Kegelmeyer, W.P. (2002) SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*. 16,321–357.

Chen, H., Tino, P. & Yao, X. (2014) Efficient probabilistic classification vector

machine with incremental basis function selection. *IEEE Transactions on Neural Networks and Learning Systems*. 25 (2), 356–369.

Chen, S., Gunn, S.R. & Harris, C.J. (2001) The relevance vector machine technique for channel equalization application. *IEEE Transactions on Neural Networks*. 12 (6), 1529–1532.

Commission, E. & Sheet, F. (2016) *2015 road safety statistics : What is behind the figures ?* (March), 2016–2019.

Cunto, F. (2008) Assessing Safety Performance of Transportation Systems using Microscopic Simulation. *Doctoral Thesis*,University of Waterloo, Ontario, Canada.

Cunto, F. & Saccomanno, F.F. (2008) Calibration and validation of simulated vehicle safety performance at signalised intersections. *Accident Analysis and Prevention*. 40 (3), 1171–1179.

Demir, B. & Ertürk, S. (2007) Hyperspectral Image Classification Using Relevance Vector Machines. *2007 IEEE International Geoscience and Remote Sensing Symposium*. 4 (4), 586–590.

Department for Transport (2012) *GB Road Traffic Counts - Datasets - DGU*. [Online] [online]. Available from: http://data.gov.uk/dataset/gb-road-traffic-counts.

Department for Transport (2016) *Reported Road Casualties Great Britain: 2015 Annual Report*. (September), 1–409.

Di, A. & Sacco, N. (2016) Open problems in transportation engineering with connected and autonomous vehicles. *Transportation Research Procedia*. 14,2255–2264.

Dijkstra, A. (2013) Assessing the safety of routes in a regional network. *Transportation Research Part C: Emerging Technologies*. 32,103–115.

Dijkstra, A., Marchesini, P., Bijleveld, F., Kars, V., Drolenga, H. & van Maarseveen, M. (2010) Do Calculated Conflicts in Microsimulation Model Predict Number of Crashes? *Transportation Research Record: Journal of the Transportation Research Board*. 2147 (January 2016), 105–112.

Dixit, V. V, Chand, S. & Nair, D.J. (2016) *Autonomous Vehicles : Disengagements , Accidents and Reaction Times*. PLoS ONE 11(12): e0168054 1–14.

Dolgov, D., Thrun, S., Montemerlo, M. & Diebel, J. (2010) Path Planning for Autonomous Vehicles in Unknown Semi-structured Environments. *The International Journal of Robotics Research*. 29 (5), 485–501.

Dowling, R., Skabardonis, A. & Alexiadis, V. (2004) Traffic Analysis Toolbox Volume III : Guidelines for Applying Traffic Microsimulation Modeling Software. *Report. No. FHWA-HRT-04-040, U.S. DOT, Federal Highway Administration, Washington, D.C.* III (July), 146.

Dreiseitl, S. & Ohno-Machado, L. (2002) Logistic regression and artificial neural network classification models: A methodology review. *Journal of Biomedical Informatics*. 35 (5–6), 352–359.

El-Basyouny, K. & Sayed, T. (2013) Safety performance functions using traffic conflicts. *Safety Science*. 51 (1), 160–164.

Elrahman, S.M.A. & Abraham, A. (2013) A Review of Class Imbalance Problem. *Network and Innovative Computing*. 1,332–340.

Eskandarian, A. (2012) *Handbook of Intelligent Vehicles*. Azim Eskandarian (ed.). Springer London.

Essa, M. & Sayed, T. (2015a) Simulated Traffic Conflicts Do They Accurately Represent Field-Measured Conflicts? *Transportation Research Record: Journal of the Traportation Research Board*. 2514.

Essa, M. & Sayed, T. (2015b) Transferability of calibrated microsimulation model parameters for safety assessment using simulated conflicts. *Accident Analysis & Prevention*. 84,41–53.

Fagnant, D.J. & Kockelman, K. (2015) Preparing a nation for autonomous vehicles : opportunities , barriers and policy recommendations. *Transportation Research Part A*. 77,167–181.

Fan, R., Wang, W., Liu, P. & Yu, H. (2013) Using VISSIM simulation model and Surrogate Safety Assessment Model for estimating field measured traffic conflicts at freeway merge areas. *IET Intelligent Transport Systems*. 7 (1), 68–77.

Fang, S., Xie, W., Wang, J. & Ragland, D.R. (2016) Utilizing the eigenvectors of freeway loop data spatiotemporal schematic for real time crash prediction. *Accident Analysis and Prevention*. 94,59–64.

Fawcett, T. (2006) An introduction to ROC analysis. *Pattern Recognition Letters*. 27 (6), 861–874.

Ferguson, D. & Likhachev, M. (2008) 'Efficiently Using Cost Maps For Planning Complex Maneuvers', in *International Conference on Robotics and Automation Workshop on Planning with Cost Maps*. 2008.

Fernandez, C., Dominguez, R., Fernandez-Llorca, D., Alonso, J. & Sotelo, M.A. (2013) Autonomous Navigation and Obstacle Avoidance of a Micro-bus. *International Journal of Advanced Robotic Systems*. 10 (212), 1–9.

FHWA, Halkias, J. & Colyar, J. (2006) *NGSIM Overview*. Federal Highway Administration, Technical Report.

Fletcher, L., Teller, S., Olson, E., Moore, D., Kuwata, Y., Leonard, J., Miller, I., Campbell, M., Nathan, A. & Kline, F. (2008) The MIT – Cornell Collision and Why It Happened. *Journal of Field Robotics*. 25 (10), 775–807.

Forrest, A. & Konca, M. (2007) *Autonomous Cars and Society*. Report, Worcester Polytechnic Institute.

Furda, A. & Vlacic, L. (2011) Enabling Safe Autonomous Driving in Real-World City Traffic Using Multiple Criteria Decision Making. *IEEE Intelligent Transportation Systems Magazine*. 3 (1), 4–17.

Gadepally, V.N. (2013) *Estimation of Driver Behavior for Autonomous Vehicle Applications*.Doctoral Thesis, Ohio State University

Galar, M., Fernandez, A., Barrenechea, E., Bustince, H. & Herrera, F. (2012) A review on ensembles for the class imbalance problem: Bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*. 42 (4), 463–484.

Gettman, D. & Head, L. (2003) Surrogate Safety Measures from Traffic Simulation Models. *Transportation Research Record: Journal of the Traportation Research Board*. 104–115.

Gettman, D., Pu, L., Sayed, T. & Shelby, S.G. (2008) *Surrogate Safety Assessment Model and Validation : Final Report*. (June), FHWA-HRT-08-051.

Ghahramani, Z. (2001) An introduction to Hidden Markov Models and Bayesian Networks. *International Journal of Pattern Recognition and Artificial Intelligence*. 15 (1), 9–42.

Gindele, T., Brechtel, S. & Dillmann, R. (2015) Learning Driver Behavior Models from Traffic Observations for Decision Making and Planning. *IEEE Intelligent Transporation Systems Magazine*. 7 (1), 69–79.

Golob, T.F., Recker, W. & Pavlis, Y. (2008) Probabilistic models of freeway safety performance using traffic flow data as predictors. *Safety Science*. 46 (9), 1306–1333.

Golob, T.F. & Recker, W.W. (2004) A method for relating type of crash to traffic flow characteristics on urban freeways. *Transportation Research Part A: Policy and Practice*. 38 (1), 53–80.

Google (2015) *Google Self-Driving Car Project Monthly Report (May 2015).*

Guan, D., Yuan, W., Lee, Y.K. & Lee, S. (2009) Nearest neighbor editing aided by unlabeled data. *Information Sciences*. 179 (13), 2273–2282.

Guido, G., Vitale, A., Astarita, V., Saccomanno, F., Giofré, V.P. & Gallelli, V. (2012) Estimation of Safety Performance Measures from Smartphone Sensors. *Procedia - Social and Behavioral Sciences*. 54,1095–1103.

Gurney, J.K. (2013) *Sue My Car Not Me : Products Liability and Accidents Involving Autonomous Vehicles JOURNAL OF LAW, TECHNOLOGY & POLICY,2013*.

Guyon, I. & Elisseeff, A. (2003) An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research (JMLR)*. 3 (3), 1157–1182.

Haboucha, C.J., Ishaq, R. & Shiftan, Y. (2017) User preferences regarding autonomous vehicles. *Transportation Research Part C*. 78,37–49.

Habtemichael, F.G. & De Picado Santos, L. (2014) Crash risk evaluation of aggressive driving on motorways: Microscopic traffic simulation approach. *Transportation Research Part F: Traffic Psychology and Behaviour*. 23,101–112.

Hajian-Tilaki, K. (2013) Receiver operating characteristic (ROC) curve analysis for medical diagnostic test evaluation. *Caspian Journal of Internal Medicine*. 4 (2), 627–635.

Hardy, J. & Campbell, M. (2013) Contingency Planning over Probabilistic Obstacle Predictions for Autonomous Road Vehicles. *IEEE Transactions on Robotics*. 29 (4),

913–929.

Hassan, H.M. & Abdel-Aty, M. a (2013) Predicting reduced visibility related crashes on freeways using real-time traffic flow data. *Journal of safety research*. 4529–36.

Hayward, J.C. (1972) Near-miss determination through use of a scale of danger. *Highway Research Record*. 38, 424–34.

He, H. & Garcia, E.A. (2009) Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*. 21 (9), 1263–1284.

Helbing, D., Hennecke, A., Shvetsov, V. & Treiber, M. (2002) Micro-and Macro-Simulation of Freeway Traffic. *PERGAMON Mathematical and Computer Modelling*. 35 (13), 517–547.

Herbrish, R. (2002) *Learning Kernel classifier, theory and algorithms*.The MIT Press

Highways Agency & Robert Goodwill (2014) *New generation of motorway opens on M25 - Press release GOV*. [Online]. Available from: https://www.gov.uk/government/news/new-generation-of-motorway-opens-on-m25.

Highways England (2017) *Roads managed by Highways England*. [Online] [online]. Available from: https://www.gov.uk/government/publications/roads-managed-by-the-highways-agency.

Himmelsbach, M., Hundelshausen, F. Von, Luttel, T., Manz, M., Muller, A., Schneider, S. & Wunsche, H.-J. (2009) 'Team MuCAR-3 at C-ELROB 2009', in *Proceedings of 1st workshop on field robotics, civilian European land robot trial, University of Oulu, Oulu, Finland*. 2009 pp. 978–951.

Ho, T.K. (1998) The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 20 (8), 832–844.

Holland, J.H. (1992) Genetic Algorithms - Computer programs that ‘evolve’ in ways that resemble natural selection can solve complex problems even their creators do not fully understand. Scientific American p.66–72.

Hossain, M. (2011) *Development of a real-time proactive road safety management system for urban expressways*. Doctoral Thesis. Tokyo Institue of Technology.

Hossain, M. & Muromachi, Y. (2012) A Bayesian network based framework for real-time crash prediction on the basic freeway segments of urban expressways. *Accident; analysis and prevention*. 45,373–81.

Hossain, M. & Muromachi, Y. (2013) A real-time crash prediction model for the ramp vicinities of urban expressways. *IATSS Research*. 37 (1), 68–79.

Hou, J., List, G.F. & Guo, X. (2014) New Algorithms for Computing the Time-to-Collision in Freeway Traffic Simulation Models. *Computational Intelligence and Neuroscience*. 2014.

Howard, T.M. (2009) *Adaptive model-predictive motion planning for navigation in complex environments*. Doctoral Thesis. Carnegie Mellon University.

Huang, E., Antoniou, C., Wen, Y., Ben-Akiva, M., Lopez, J. & Bento, L.C. (2013) 'Real-Time Multi-Sensor Multi-Source Network Data Fusion Using Dynamic Traffic Assignment Models', in *Proceedings of the 12th International IEEE Conference on Intelligent Transportation Systems, St. Louis, MO, USA, October 3-7, 2009*. [Online]. 2013

Huang, F., Liu, P., Yu, H. & Wang, W. (2013) Identifying if VISSIM simulation model and SSAM provide reasonable estimates for field measured traffic conflicts at signalised intersections. *Accident Analysis and Prevention*. 50,1014–1024.

Huguenin, F., Torday, A. & Dumont, A. (2005) 'Evaluation of traffic safety using microsimulation', in *5th Swiss Transport Research Conference*. 2005.

Hundelshausen, F. Von, Himmelsbach, M., Hecker, F., Mueller, A. & Wuensche, H. (2008) Driving with Tentacles : Integral Structures for Sensing. *Journal of field Robotics*. 25 (9), 640–673.

Hyden, C. (1987) The Development of a Method for Traffic Safety Evaluation: the Swedish Traffic Conflict Technique. *Bulletin 70, Lund University of Technology, Lund*. 1–8.

Ikeda, H., Expressway, M., Corporation, P. & Kaneko, Y. (1999) Abnormal Incident Detection System Employing Image Processing Technology. *October*. 748–752.

Imprialou, M.-I. (2015) *Developing accident-speed relationships using a new modelling approach,*Doctoral Thesis, Loughborough University.

Imprialou, M.M., Quddus, M. & Pitfield, D. (2015) 'Exploring the Role of Speed in Highway Crashes : Pre-Crash-Condition-Based Multivariate Bayesian Modelling', in *Transportation Research Board 94th Annual Meeting*. 2015 pp. 1–17.

Jeon, J., Cowlagi, R. V, Peters, S.C., Karaman, S., Frazzoli, E., Tsiotras, P. & Iagnemma, K. (2013) 'Optimal Motion Planning with the Half-Car Dynamical Model for Autonomous High-Speed Driving', in *American Control Conference*. 2013 pp. 188–193.

Jiménez, F., Naranjo, J.E. & Gómez, O. (2012) Autonomous manoeuvring systems for collision avoidance on single carriageway roads. *Sensors* 12 (12), 16,498–521.

Kala, R. & Warwick, K. (2013) Motion planning of autonomous vehicles in a non-autonomous vehicle environment without speed lanes. *Engineering Applications of Artificial Intelligence*. 26 (5–6), 1588–1601.

Kammel, S., Ziegler, J., Pitzer, B., Gindele, T., Jagzent, D., Schr, J. & Hundelshausen, F. Von (2008) Team AnnieWAY ' s Autonomous System for the 2007 DARPA Urban Challenge. *Journal of Field Robotics*. 25 (9), 615–639.

Karlaftis, M.G. & Vlahogianni, E.I. (2011) Statistical methods versus neural networks in transportation research: Differences, similarities and some insights. *Transportation Research Part C: Emerging Technologies*. 19 (3), 387–399.

Katrakazas, C., Quddus, M., Chen, W.-H. & Deka, L. (2015) Real-time motion planning methods for autonomous on-road driving : State-of-the-art and future research directions. *Transportation Research Part C: Emerging Technologies*, 60, 416-442.

Kessler, A.M. (2015) Elon Musk Says Self-Driving Tesla Cars Will Be in the U . S . by Summer, *New York Times*.

Kockelman, K.M. & Ma, J. (2007) Freeway speeds and speed variations preceding crashes withing and across lanes. *Journal of Transportation Research Forum*. 46 (1964), 43–61.

Koller, D. & Friedman, N. (2009) Probabilistic Graphical Models: Principles and Techniques. Vol. 53. The MIT Press.

Kolski, S., Ferguson, D., Bellino, M. & Siegwart, R. (2006) 'Autonomous Driving in Structured and Unstructured Environments', in *2006 IEEE Intelligent Vehicles Symposium*. pp. 558–563.

Koopman, P. & Wagner, M. (2016) Challenges in Autonomous Vehicle Testing and Validation Driver Out of the Loop, in *2016 SAE World Congress* .

Kosonen, I. (1996). Hutsim: Simulation Tool for Traffic Signal Control Planning, Doctoral Thesis, Helsinki University of Technology.

Kuhnt, F., Kohlhaas, R., Schamm, T. & Marius, J.Z. (2015) Towards a Unified Traffic Situation Estimation Model – Street-dependent Behaviour and Motion Models –. *International Conference on Information Fusion*. 1223–1229.

Kushleyev, A. & Likhachev, M. (2009) 'Time-bounded lattice for efficient planning in dynamic environments', in *2009 IEEE International Conference on Robotics and Automation*. [Online]. May 2009 pp. 1662–1668.

Kyriakidis, M., Happee, R. & Winter, J.C.F. De (2015) Public opinion on automated driving : Results of an international questionnaire among 5000 respondents. *Transportation Research Part F: Psychology and Behaviour*. 32,127–140.

Laumond, J. (1998) *Robot Motion Planning and Control*. Jean-paul Laumond (ed.). Vol. 229. Springer-Verlag Berlin Heidelberg.

Laurikkala, J. (2001) Improving identification of difficult small classes by balancing class distribution. *Proceedings of the 8th Conference on AI in Medicine in Europe: Artificial Intelligence Medicine*. 63–66.

LaValle, S.M. (2006) *Planning Algorithms*. Cambridge University Press.

Lee, C., Hellinga, B. & Saccomanno, F. (2003) 'Real-time crash prediction model for application to crash prevention in freeway traffic', in *Transportation Research Board 82nd Annual Meeting*. 2003.

Lee, C., Saccomanno, F. & Hellinga, B. (2002) Analysis of Crash Precursors on Instrumented Freeways. *Transportation Research Record*. 1784 (1), 1–8.

Lee, U. & Vasseur, P. (2014) 'Local path planning in a complex environment for self-driving car', in *4th Annual IEEE International Conference on Cyber Technologoy in Automation, Control and Intelligent Systems*. 2014 Hong Cong, China: . pp. 445–450.

Lefèvre, S. (2012) *Risk Estimation at Road Intersections for Connected Vehicle Safety Applications*. Doctoral Thesis, INP Grenoble and INRIA.

Lefèvre, S., Laugier, C. & Ibañez-Guzmán, J. (2012) 'Risk assessment at road

intersections: Comparing intention and expectation', in *IEEE Intelligent Vehicles Symposium, Proceedings*.

Lefèvre, S., Vasquez, D. & Laugier, C. (2014) A survey on motion prediction and risk assessment for intelligent vehicles. *ROBOMECH Journal*. 1 (1), 1–14.

Lemaitre, G., Nogueira, F. & Aridas, C.K. (2016) Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning. *CoRR*. abs/1609.01–5.

Leonard, J., How, J., Teller, S., Berger, M., Campbell, S., Fiore, G., Fletcher, L., Frazzoli, E., Huang, A., Karaman, S., Koch, O., Moore, D., Olson, E., Peters, S., Teo, J., Truax, R., Walter, M., Barrett, D., Epstein, A., et al. (2008) A Perception-Driven Autonomous Urban Vehicle. *Journal of Field Robotics*. 25 (February), 727–774.

Li, X., Lord, D., Zhang, Y. & Xie, Y. (2008) Predicting motor vehicle crashes using Support Vector Machine models. *Accident Analysis and Prevention*. 40 (4), 1611–1618.

Li, Z., Chitturi, M., Noyce, D., Ran, B., Chitturi, M. V & Noyce, D. (2013) Development of Next Generation Intersection Control,*Technical Report, CFIRE 04-18, National Center for Freight & Infrastructure Research & Education Department of Civil and Environmental Engineering College of Engineering University of Wisconsin–Madison*.

Lin, L., Wang, Q. & Sadek, A.W. (2015) A novel variable selection method based on frequent pattern tree for real-time traffic accident risk prediction. *Transportation Research Part C: Emerging Technologies*. 55,444–459.

Lin, L., Yeh, Y. & Chu, T. (2014) *Feature Selection Algorithm for ECG Signals and Its Application on Heartbeat Case Determining*. 16 (4), 483–496.

Lin, P. (2014) *What If Your Autonomous Car Keeps Routing You Past Krispy Kreme?*

- *Yahoo Finance*. [Online] [online]. Available from: http://finance.yahoo.com/news/autonomous-car-keeps-routing-past-130800241.html;_ylt=A2KJ3CUL199SkjsAexPQtDMD (Accessed 1 September 2014).

Litman, T. (2014) 'Autonomous Vehicle Implementation Predictions Implications for Transport Planning', in *Transportation Research Board Annual Meeting 2014*.

López, V., Fernández, A., García, S., Palade, V. & Herrera, F. (2013) An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. *Information Sciences*. 250,113–141.

Lv, Y., Tang, S. & Zhao, H. (2009) Real-time highway traffic accident prediction based on the k-nearest neighbor method. *2009 International Conference on Measuring Technology and Mechatronics Automation, ICMTMA 2009*. 3547–550.

Macek, K., Becker, M. & Siegwart, R. (2006) 'Motion Planning for Car-Like Vehicles in Dynamic Urban Scenarios', in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. October 2006 pp. 4375–4380.

Martin, A. (2013) *Interactive Motion Prediction using Game Theory*. Master Thesis, University of Padova.

Martinez-gomez, L. & Fraichard, T. (2009) Benchmarking Collision Avoidance Schemes for Dynamic Environments, ICRA Workshop on Safe Navigation in Open and Dynamic Environments.

Mathworks (2016) *MATLAB*.

McNaughton, M., Urmson, C., Dolan, J.M. & Lee, J.-W. (2011) 'Motion planning for autonomous driving with a conformal spatiotemporal lattice', in *2011 IEEE International Conference on Robotics and Automation*. May 2011 pp. 4889–4895.

Mertz, C., Navarro-serment, L.E., Maclachlan, R., Rybski, P., Steinfeld, A., Supp, A., Urmson, C., Vandapel, N., Hebert, M., Thorpe, C., Duggins, D. & Gowdy, J. (2013) Moving Object Detection with Laser Scanners. Journal of Field Robotics 30(1), 17–43.

Merwe, R. van der, Doucet, A., Freitas, N. de & Wan, E. (2000) The Unscented Particle Filter, *Technical Report CUED/F-INFENG/TR 380, Cambridge University*.

Michael E. Tipping (2009) An Efficient Matlab Implementation of the Sparse Bayesian Modelling Algorithm (Version 2.0). *Signal Processing, IEEE Transactions on*. 57 (6), 0–14.

Minderhoud, M.M. & Bovy, P.H. (2001) Extended time-to-collision measures for road traffic safety assessment. *Accident; analysis and prevention*. 33 (1), 89–97.

Mui, C. (2013) *Will The Google Car Force A Choice Between Lives And Jobs?* [Online] [online]. Available from: http://www.forbes.com/sites/chunkamui/2013/12/19/will-the-google-car-force-a-choice-between-lives-and-jobs/?ss=future-tech (Accessed 1 September 2014).

Murphy, K. (2002) *Dynamic Bayesian Networks: Representation, Inference and Learning*.. Doctoral Thesis, University of California, Berkeley.

Murphy, K.P. (2012) Machine Learning: A Probabilistic Perspective, The MIT Press.

Murphy, L. & Newman, P. (2011) 'Risky planning: Path planning over costmaps with a probabilistically bounded speed-accuracy tradeoff', in *2011 IEEE International Conference on Robotics and Automation*.. May 2011 pp. 3727–3732.

National Highway Traffic Safety Administration (2015) 2014 Crash Data Key Findings. *Traffic Safety Facts*. Technical Report (November), 1–2.

Ntousakis, I.A., Nikolos, I.K. & Papageorgiou, M. (2016) Optimal vehicle trajectory planning in the context of cooperative merging on highways. *Transportation Research Part C*. 71,464–488.

Oh, C., Oh, J.-S., Ritchie, S.G. & Chang, M. (2001) Real-Time Estimation of Freeway Accident Likelihood, *Report UCI-ITS-TS-WP-00-8, Department of Civil and Environmental Engineering and Institute of Transportation Studies, University of California, Irvine*.

Ong, S.C.W., Shao Wei Png, Hsu, D. & Wee Sun Lee (2010) Planning under Uncertainty for Robotic Tasks with Mixed Observability. *The International Journal of Robotics Research*. 29 (8), 1053–1068.

OpenStreetMap® (2016) *OpenStreetMap*.

Ozbay, K., Yang, H., Bartin, B. & Mudigonda, S. (2008) Derivation and Validation of a New Simulation-based Surrogate Safety Measure Kaan. *Transportation Research Record,Journal of the Transportation Research Board*. 1–19.

Paden, B., Cap, M., Yong, S.Z., Yershov, D. & Frazzoli, E. (2016) A Survey of Motion Planning and Control Techniques for Self-driving Urban Vehicles. *IEEE Transactions on Intelligent Vehicles*, 1(1), pp. 33-55, 2016.

Pande, A. (2005) *Estimation of Hybrid Models for Real-Time Crash Risk Assessment on Freeways*,Doctoral Thesis, University of Central Florida.

Pande, A. & Abdel-Aty, M. (2005) A Freeway Safety Strategy for Advanced Proactive Traffic Management. *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*. 9 (3), 145–158.

Pande, A. & Abdel-Aty, M. (2006) Assessment of freeway traffic parameters leading to lane-change related collisions. *Accident Analysis and Prevention*. 38 (5), 936–948.

Park, H. & Haghani, A. (2016) Real-time prediction of secondary incident occurrences using vehicle probe data. *Transportation Research Part C: Emerging Technologies*. 70,69–85.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. & Duchesnay, É. (2012) Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*. 12,2825–2830.

Peng, Y., Abdel-Aty, M., Shi, Q. & Yu, R. (2017) Assessing the impact of reduced visibility on traffic crash risk using microscopic data and surrogate safety measures. *Transportation Research Part C: Emerging Technologies*. 74 (January 2008), 295–305.

Perkins, S.R. & Harris, J.I. (1967) Criteria for Traffic Conflict Characteristics, *HRB Rec. 225. Highway Research. Boardd., 35-44, 1967*.

Phillips, C.L., Bruno, M.A., Maquet, P., Boly, M., Noirhomme, Q., Schnakers, C., Vanhaudenhuyse, A., Bonjean, M., Hustinx, R., Moonen, G., Luxen, A. & Laureys, S. (2011) 'Relevance vector machine' consciousness classifier applied to cerebral metabolism of vegetative and locked-in patients. *NeuroImage*. 56 (2), 797–808.

Pivtoraiko, M. & Kelly, A. (2005) 'Efficient Constrained Path Planning via Search in State Lattices', in *International Symposium on Artificial Intelligence, Robotics, and Automation in Space*. 2005.

Pivtoraiko, M. & Kelly, A. (2009) 'Fast and feasible deliberative motion planner for dynamic environments', in *Proceedings of the 2009 ICRA Workshop on Safe Navigation in Open and Dynamic Environments*.

Pivtoraiko, M., Knepper, R.A. & Kelly, A. (2009) Differentially Constrained Mobile Robot Motion Planning in State Spaces. *Journal of Field Robotics*. 26 (3), 308–333.
Polychronopoulos, a., Tsogas, M., Amditis, a. J. & Andreone, L. (2007) Sensor Fusion for Predicting Vehicles' Path for Collision Avoidance Systems. *IEEE Transactions on Intelligent Transportation Systems*. 8 (3), 549–562.

Powers, D. (2011) Evaluation : From Precision , Recall And F-Measure To ROC , Informedness , Markedness & Correlation. *Journal of Machine Learning Technologies*. 2 (1), 37–63.

Press, W.H., Teukolsky, S.A., Vetterling, W.T. & Flannery, B.P. (1993) *Numerical recipes in Fortran (The art of scientific computing)*. Vol. 35.

PTV Planung Trasport Verkehr AG (2013) *PTV VISSIM 6 User Manual*.

Pu, L. & Joshi, R. (2008) *Surrogate Safety Assessment Model (SSAM): Software User Manual*.

Quddus, M. A., Wang, C. & Ison, S.G. (2010) Road Traffic Congestion and Crash Severity: Econometric Analysis Using Ordered Response Models. *Journal of Transportation Engineering*. 136 (5), 424–435.

Rasmussen, C.E. (2006) Gaussian processes for machine learning. *International journal of neural systems*. 14 (2), 69–106.

Rauskolb, F.W., Berger, K., Lipski, C., Magnor, M., Cornelsen, K., Effertz, J., Form, T., Graefe, F., Ohl, S., Doering, M., Homeier, K., Morgenroth, J., Wolf, L., Basarke, C. & Berger, C. (2008) Caroline : An Autonomously Driving Vehicle for Urban Environments. *Journal of Field Robotics*. 25 (January), 674–724.

Rokach, L. (2010) Ensemble-based classifiers. *Artificial Intelligence Review*. 33 (1–2), 1–39.

Roshandel, S., Zheng, Z. & Washington, S. (2015) Impact of real-time traffic characteristics on freeway crash occurrence: Systematic review and meta-analysis. *Accident Analysis and Prevention*. 79,198–211.

Ross, P.E. (2014) *A Cloud-Connected Car Is a Hackable Car, Worries Microsoft - IEEE Spectrum*. [Online] [online]. Available from: http://spectrum.ieee.org/tech-talk/transportation/advanced-cars/a-connected-car-is-a-hackable-car (Accessed 1

September 2014).

Saeys, Y., Inza, I. & Larra??aga, P. (2007) A review of feature selection techniques in bioinformatics. *Bioinformatics*. 23 (19), 2507–2517.

Saffarzadeh, M., Nadimi, N., Naseralavi, S. & Mamdoohi, A.R. (2013) A general formulation for time-to-collision safety indicator. *Proceedings of the Institution of Civil Engineers - Transport*. 166 (5), 294–304.

Saidallah, M., Fergougui, A. El & Elalaoui, A.E. (2016) *A Comparative Study of Urban Road Traffic Simulators*. MATEC Web Conference, 81(2016)

Saito, T. & Rehmsmeier, M. (2015) The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PLoS ONE*. 10 (3), 1–21.

Sargent, D.J. (2001) Comparison of artificial neural networks with other statistical approaches. *Cancer*. 91 (S8), 1636–1642.

Sayed, T. & Zein, S. (1999) Traffic conflict standards for intersections. *Transportation Planning and Technology*. 22(4) (October 2014), 309–323.

Scanlon, J.M., Sherony, R. & Gabler, H.C. (2016) Preliminary potential crash prevention estimates for an Intersection Advanced Driver Assistance System in straight crossing path crashes. *IEEE Intelligent Vehicles Symposium, Proceedings*. 2016–Augus (Iv), 1135–1140.

Scheuer, A. & Fraichard, T. (1997) 'Collision-Free and Continuous-Curvat ure Path Planning for Car-Like Robots', in *1997 IEEE International Conference on Robotics And Automation*. [Online]. 1997 pp. 867–873.

Schnieder, M. (2017) Development of an improved time-to-collision algorithm - *P 16CVC002_2 - European Short Research Project*,Loughborough University.

Schröder, J., Gindele, T., Jagszent, D. & Dillmann, R. (2008) 'Path Planning for Cognitive Vehicles using Risk Maps', in *IEEE Intelligent Vehicles Symposium, 2008*. [Online]. 2008 pp. 1119–1124.

Schuster, W. (2015) Trajectory prediction for future air traffic management – complex manoeuvres and taxiing. *Aeronautical Journal*. 119,121–143.

Shahdah, U., Saccomanno, F. & Persaud, B. (2015) Application of traffic microsimulation for evaluating safety performance of urban signalised intersections. *Transportation Research Part C: Emerging Technologies*. 6096–104.

Shahdah, U., Saccomanno, F. & Persaud, B. (2014) Integrated traffic conflict model for estimating crash modification factors. *Accident Analysis and Prevention*. 71,228–235.

Shahdah, U.E. (2014) Integrating Observational and Microscopic Simulation Models for Traffic Safety Analysis,Doctoral Thesis, University of Waterloo,Ontario, Canada.

Shariff, A. & Rahwan, I. (2016) The social dilemma of autonomous vehicles. *Science* 352 (6293), 1573–1576.

Sharma, A. & Collins, E.G. (2014) *Robust Sampling-Based Trajectory Tracking for Autonomous Vehicles*. 3446–3451.

Shaumyan, A. (2016) *sklearn-bayes* Python Package.

Shew, C., Pande, A. & Nuworsoo, C. (2013) Transferability and robustness of real-time freeway crash risk assessment. *Journal of safety research*. 46,83–90.

Shi, Q. & Abdel-Aty, M. (2015) Big Data applications in real-time traffic operation and safety monitoring and improvement on urban expressways. *Transportation Research Part C: Emerging Technologies*. 58,380–394.

Siegwart, R., Nourbakhsh, I.R. & Scaramuzza, D. (2011) *Introduction to Autonomous Mobile Robots 2nd edition*. MIT Press.

Singh, S. (2015) Critical reasons for crashes investigated in the National Motor Vehicle Crash Causation Survey. *National Highway Traffic Safety Administration*. (February), 1–2.

Snider, J.M. (2009) Automatic Steering Methods for Autonomous Automobile Path Tracking. Technical Report CMU-RI-TR-09-08, Robotics Institute, Carnegie Mellon University (February).

Sobhani, A., Young, W. & Sarvi, M. (2013) A simulation based approach to assess the safety performance of road locations. *Transportation Research Part C: Emerging Technologies*. 32,144–158.

Sousanis, J. (2011) *World Vehicle Population Tops 1 Billion Units*. [Online]. Available from: http://wardsauto.com/ar/world_vehicle_population_110815 (Accessed 1 September 2014).

Staubach, M. (2009) Factors correlated with traffic accidents as a basis for evaluating Advanced Driver Assistance Systems. *Accident; analysis and prevention*. 41 (5), 1025–33.

Strobl, C., Boulesteix, A.-L., Zeileis, A. & Hothorn, T. (2007) Bias in random forest variable importance measures: illustrations, sources and a solution. *BMC Bioinformatics*. 825.

Sun, J. & Sun, J. (2015) A dynamic Bayesian network model for real-time crash prediction using traffic speed conditions data. *Transportation Research Part C: Emerging Technologies*. 54176–186.

Sun, Y., Wong, A.K.C. & Kamel, M.S. (2009) Classification of Imbalanced Data : a Review. *International Journal of Pattern Recognition and Artificial Intelligence*. 23

(4), 687–719.

SYSTRA Limited (2009) S-Paramics Principles. Paramics Microsimulation SYSTRA Limited, Edinburgh

Takahashi, O. & Schilling, R.J. (1989) Motion Planning in a Plane Using Generalised Voronoi Diagrams. *IEEE Transactions on Robotics and Automation*. 5 (2), 143–150.

Theofilatos, A. (2015) *An advanced multi-faceted statistical analysis of accident probability and severity exploiting high resolution traffic and weather data*. Doctoral Thesis, National Technical University of Athens, Greece.

Thorpe, C. & Durrant-Whyte, H. (2009) *The DARPA Urban Challenge*. Martin Buehler, Karl Iagnemma, & Sanjiv Singh (eds.). Vol. 56. Springer Berlin Heidelberg.

Thrun, S. (2010) Toward Robotic Cars. *Communications of the ACM*. 53 (4 (April)), 99–106.

Thrun, S., Montemerlo, M., Dahlkamp, H., Stavens, D., Aron, A., Diebel, J., Fong, P., Gale, J., Halpenny, M., Hoffman, G., Lau, K., Oakley, C., Palatucci, M., Pratt, V., Stang, P., Strohband, S., Dupont, C., Jendrossek, L.-E., Koelen, C., et al. (2006) Stanley: The robot that won the DARPA Grand Challenge. *Journal of field Robotics*. 23 (4), 661–692.

Tipping, M. (2001) Sparse Bayesian Learning and the Relevance Vector Mach. *Journal of Machine Learning Research*. 1,211–244.

Tipping, M.E. (2009) A Baseline Matlab Implementation of ‘ Sparse Bayesian ’ Model Estimation. SparseBayes Manual. 2–6.

Tipping, M.E. & Faul, A.C. (2003) Fast Marginal Likelihood Maximisation for Sparse Bayesian Models. *Ninth International Workshop on Aritficial Intelligence and*

Statistics. 1–13.

Toledo, T., Koutsopoulos, H.N. & Ben-Akiva, M. (2003) Modeling Integrated Lane-Changing Behavior. *Transportation Research Record*. 1857 (1), 30–38.

Tomek, I. (1976) An Experiment with the Edited Nearest-Neighbor Rule. *IEEE Transactions on Systems, Man, and Cybernetics*. 6 (6), 448–452.

Transport For London (2010) *Traffic Modelling Guidelines TfL Traffic Manager and*.
WDOT (2014) Wisconsin Department of Transportation, *Model Calibration - Traffic Analysis and Microsimulation*. [Online] [online]. Available from: http://www.wisdot.info/microsimulation/index.php?title=Model_Calibration.

Treat, J.R., Tumbas, N.S., Mcdonald, S.T., Hume, R.D., Mayer, R.E., Stansifer, R. & Castellan, N.J. (1979) *TRI-LEVEL STUDY OF THE CAUSES OF TRAFFIC ACCIDENTS Executive Summary*, Technical Report.

Triggs, T.J. & Harris, W.G. (1982) Reaction Time of Drivers to Road Stimuli. *Medicinski Pregled*. 62 (June 1982), 114–9.

Tsui, K.L., So, F.L., Sze, N.N., Wong, S.C. & Leung, T.F. (2009) Misclassification of injury severity among road casualties in police reports. *Accident Analysis and Prevention*. 41 (1), 84–89.

Varaiya, P. (1993) Smart cars on smart roads. Problems of control. *IEEE Transactions on Automatic Control*. 38 (2), 195–207.

Verikas, A., Gelzinis, A. & Bacauskiene, M. (2011) Mining data with random forests: A survey and results of new tests. *Pattern Recognition*. 44 (2), 330–349.

Le Vine, S., Zolfaghari, A. & Polak, J. (2015) Autonomous cars: The tension between occupant experience and intersection capacity. *Transportation Research Part C: Emerging Technologies*. 521–14.

Vogt, A. & Bared, J.G. (2008) *Accident Models for Two-Lane Rural Roads: Segment and Intersections* - Technical Report FHWA-RD-98-133. (October 1998).

Wang, C., Quddus, M. a. & Ison, S.G. (2013) The effect of traffic and road characteristics on road safety: A review and future research direction. *Safety Science*. 57264–275.

Wang, L., Abdel-Aty, M., Shi, Q. & Park, J. (2015) Real-time crash prediction for expressway weaving segments. *Transportation Research Part C: Emerging Technologies*. 611–10.

Wang, W., Qu, X., Wang, W. & Liu, P. (2013) Real-time freeway sideswipe crash prediction by support vector machine. *IET Intelligent Transport Systems*. 7 (4), 445–453.

Ward, J., Agamennoni, G., Worrall, S. & Nebot, E. (2014) 'Vehicle collision probability calculation for general traffic scenarios under uncertainty', in *IEEE Intelligent Vehicles Symposium, Proceedings*. [Online]. 2014 Dearborn, Michigan, USA: . pp. 986–992.

Ward, J., Worrall, S., Agamennoni, G. & Nebot, E. (2014) The warrigal dataset: Multi-vehicle trajectories and V2V communications. *IEEE Intelligent Transportation Systems Magazine*. 6 (3), 109–117.

Ward, J.R., Agamennoni, G., Worrall, S., Bender, A. & Nebot, E. (2015) Extending Time to Collision for probabilistic reasoning in general traffic scenarios. *Transportation Research Part C: Emerging Technologies*. 51,66–82.

Wei, J., Snider, J.M., Gu, T., Dolan, J.M. & Litkouhi, B. (2014) 'A behavioral planning framework for autonomous driving', in *IEEE Intelligent Vehicles Symposium, Proceedings*. [Online]. 2014 pp. 458–464.

Wei, L., Yang, Y., Nishikawa, R.M., Wernick, M.N. & Edwards, A. (2005)

Relevance vector machine for automatic detection a of clustered microcalcifications. *IEEE Transactions on Medical Imaging*. 24 (10), 1278–1285.

Wei Miao Yu, Tiehua Du & Kah Bin Lim (2004) Comparison of the support vector machine and relevant vector machine in regression and classification problems. *ICARCV 2004 8th Control, Automation, Robotics and Vision Conference, 2004.* 2 (December), 1309–1314.
WHO (2015) *Global Status Report on Road*. 340.

Wille, J.M. & Form, T. (2008) 'Realizing complex autonomous driving maneuvers: The approach taken by team CarOLO at the DARPA urban challenge', in *2008 IEEE International Conference on Vehicular Electronics and Safety*. September 2008 pp. 232–236.

Wille, J.M., Saust, F. & Maurer, M. (2010a) 'Comprehensive treated sections in a trajectory planner for realizing autonomous driving in Braunschweig'S urban traffic', in *13th International IEEE Conference on Intelligent Transportation Systems*. September 2010 pp. 647–652.

Wille, J.M., Saust, F. & Maurer, M. (2010b) 'Stadtpilot: Driving autonomously on Braunschweig's inner ring road', in *2010 IEEE Intelligent Vehicles Symposium*. June 2010 Ieee. pp. 506–511.

Williams, C.K.I. & Barber, D. (1998) Bayesian classification with Gaussian processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 20 (12), 1342–1351.

Wilson, D.L. (1972) Asymptotic Properties of Nearest Neighbor Rules Using Edited Data. *IEEE Transactions on Systems, Man and Cybernetics*. 2 (3), 408–421.

Wu, Y., Nakamura, H. & Asano, M. (2013) A Crash Risk Estimation Model for Urban Expressway Basic Segments Considering Geometry , Traffic Flow and Ambient Conditions. *Eastern Asia Society for Transportation Studies*. 9.

Xu, C., Liu, P., and Wang, W. (2016a) ‘Evaluation of the Predictability of Real-Time Crash Risk Models’. *Accident Analysis and Prevention* 94, 207–215

Xu, C., Liu, P., Wang, W., and Jiang, X. (2013a) ‘Development of a Crash Risk Index to Identify Real Time Crash Risks on Freeways’. *KSCE Journal of Civil Engineering* 17 (7), 1788–1797.

Xu, C., Liu, P., Wang, W., and Li, Z. (2012) ‘Evaluation of the Impacts of Traffic States on Crash Risks on Freeways.’ *Accident; analysis and prevention* 47, 162–71.

Xu, C., Liu, P., Yang, B., and Wang, W. (2016b) ‘Real-Time Estimation of Secondary Crash Likelihood on Freeways Using High-Resolution Loop Detector Data’. *Transportation Research Part C: Emerging Technologies* 71, 406–418.

Xu, C., Wang, W., and Liu, P. (2013b) ‘A Genetic Programming Model for Real-Time Crash Prediction on Freeways’. *IEEE Transactions on Intelligent Transportation Systems* 14 (2), 574–586.

Xu, C., Wang, W., Liu, P., and Li, Z. (2015a) ‘Calibration of Crash Risk Models on Freeways with Limited Real-Time Traffic Data Using Bayesian Meta-Analysis and Bayesian Inference Approach’. *Accident Analysis and Prevention*, 85, 207–218.

Xu, C., Wang, W., Liu, P., and Zhang, F. (2015b) ‘Development of a Real-Time Crash Risk Prediction Model Incorporating the Various Crash Mechanisms across Different Traffic States.’ *Traffic injury prevention* ,16 (1), 28–35.

Xu, W., Pan, J., Wei, J., and Dolan, J.M. (2014) ‘Motion Planning under Uncertainty for On-Road Autonomous Driving’. in *2014 IEEE International Conference on Robotics and Automation (ICRA),* 2061–2067

Yamamoto, T., Hashiji, J., and Shankar, V.N. (2008) ‘Underreporting in Traffic Accident Data, Bias in Parameters and the Structure of Injury Severity Models’.

Accident Analysis and Prevention 40 (4), 1320–1329

Yang, H. (2012) *Simulation-Based Evaluation of Traffic Safety Performance Using Surrogate Safety Measures*,Doctoral Thesis, Rutgers University, New Jersey.

Young, W., Sobhani, A., Lenne, M.G., and Sarvi, M. (2014) 'Simulation of Safety: A Review of the State of the Art in Road Safety Simulation Modelling'. *Accident Analysis and Prevention* 66, 89–103

Yu, R. (2013) *Real-Time Traffic Safety Evaluation Models and Their Application for Variable Speed Limits*.Doctoral Thesis, University of Central Florida.

Yu, R. and Abdel-Aty, M. (2014) 'Analyzing Crash Injury Severity for a Mountainous Freeway Incorporating Real-Time Traffic and Weather Data'. *Safety Science* 63, 50–56.

Yu, R. and Abdel-Aty, M. (2013a) 'Using Hierarchical Bayesian Binary Probit Models to Analyze Crash Injury Severity on High Speed Facilities with Real-Time Traffic Data.' *Accident; analysis and prevention* 62, 161–7.

Yu, R. and Abdel-Aty, M. (2013b) 'Utilizing Support Vector Machine in Real-Time Crash Risk Evaluation.' *Accident; analysis and prevention* 51, 252–9.

Yu, R. and Abdel-Aty, M. (2013c) 'Multi-Level Bayesian Analyses for Single- and Multi-Vehicle Freeway Crashes.' *Accident; analysis and prevention* 58, 97–105

Yu, R., Abdel-Aty, M., and Ahmed, M. (2013) 'Bayesian Random Effect Models Incorporating Real-Time Weather and Traffic Data to Investigate Mountainous Freeway Hazardous Factors.' *Accident; analysis and prevention* 50, 371–6.

Zhang, S., Deng, W., Zhao, Q., Sun, H., and Litkouhi, B. (2013) 'Dynamic Trajectory Planning for Vehicle Autonomous Driving'. in *2013 IEEE International Conference on Systems, Man, and Cybernetics* October 2013.4161–4166.

Zhao, P., Chen, J., Mei, T., and Liang, H. (2011) 'Dynamic Motion Planning for Autonomous Vehicle in Unknown Environments'. in *2011 IEEE Intelligent Vehicles Symposium (IV)* June 2011. 284–289.

Zhu, R.M.H., Zhang, L., and Chen, A. (2006) 'A New Method to Assist Small Data Set Neural Network Learning'. *Sixth International Conference on Intelligent Systems Design and Applications, 2006. ISDA '06.* 1, 17–22

Ziegler, J., Bender, P., Dang, T., Stiller, C., and Preliminaries, A. (2014a) 'Trajectory Planning for BERTHA - a Local , Continuous Method'. in *2014 IEEE Intelligent Vehicles Symposium (IV), June 8-11, Dearborn, Michigan, USA*. 450–457

Ziegler, J., Bender, P., Schreiber, M., Lategahn, H., Strauss, T., Stiller, C., Dang, T., Franke, U., Appenrodt, N., Keller, C.G., Kaus, E., Herrtwich, R.G., Rabe, C., Pfeiffer, D., Lindner, F., Stein, F., Erbs, F., Enzweiler, M., Knoppel, C., Hipp, J., Haueis, M., Trepte, M., Brenk, C., Tamke, A., Ghanaat, M., Braun, M., Joos, A., Fritz, H., Mock, H., Hein, M., and Zeeb, E. (2014b) 'Making Bertha Drive-an Autonomous Journey on a Historic Route'. *IEEE Intelligent Transportation Systems Magazine* 6 (2), 8–20.

Ziegler, J. and Stiller, C. (2009) 'Spatiotemporal State Lattices for Fast Trajectory Planning in Dynamic on-Road Driving Scenarios'. in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems* held October 2009. 1879–1884.

Appendix

Publications related to this thesis

A number of publications have been published in peer-reviewed journals or presented at conferences as a result of this research. These are the following:

Journal papers:

- Christos Katrakazas, Mohammed Quddus, Wen-Hua Chen (2017), *A study to predict real-time conflict conditions using highly disaggregated data,* In IEEE Transactions on Intelligent Transportation Systems (In Press)

- Christos Katrakazas, Mohammed Quddus, Wen-Hua Chen, Lipika Deka (2015), *Real-time motion planning methods for autonomous on-road driving: State-of-the-art and future research directions*, Transportation Research Part C: Emerging Technologies, 60, November 2015, Pages 416-442

Conferences

- Christos Katrakazas, Mohammed Quddus, Wen-Hua Chen (2017*) A Simulation Study of Predicting Conflict-prone Traffic Conditions in Real-time,* in: Transportation Research Board 96th Annual Meeting, Washington D.C., USA, January 2017

- Christos Katrakazas, Mohammed Quddus, Wen-Hua Chen (2017) *A new methodology for collision risk assessment of autonomous vehicles*, in: Transportation Research Board 96th Annual Meeting, Washington D.C., USA, January 2017

- Christos Katrakazas (2017), *A new methodology for real- time risk assessment of autonomous vehicles*, in 49th Annual Universities Transport Studies Group Conference, Dublin, Ireland.

- Christos Katrakazas, Mohammed Quddus, Wen-Hua Chen (2016) *Real-time Classification of Aggregated Traffic Conditions using Relevance Vector Machines*, in: Transportation Research Board 95th Annual Meeting, Washington D.C., USA, January 2016.