Safe and Robust Autonomous Intersection Management Methods

Chamideh, Seyedezahra

2021

Safe and Robust Autonomous Intersection management methods

Licentiate Thesis

SeyedeZahra Chamideh

LUND UNIVERSITY

Department of Electrical and Information Technology

Lund, November 2021

SeyedeZahra Chamideh
Department of Electrical and Information Technology
Lund University
Ole Römers Väg 3, 223 63 Lund, Sweden

Series of Licentiate and Doctoral Theses
ISSN 1654-790X, No.143
ISBN 978-91-8039-112-2 (printed)
ISBN 978-91-8039-111-5 (digital)

This thesis is typeset using $\LaTeX\,2_\varepsilon$.

Printed in Sweden by Tryckeriett i E-Huset, Lund University, Lund, 2021.

Abstract

Connected Autonomous Vehicles (**AV**)s can transform urban transportation systems and have the potential to improve the safety and efficiency, since human errors and distractions are removed. However, these systems are vulnerable to model uncertainties, communication impairments associated with the wireless communication, and external disturbances. As a result, vehicles need to drive at low speed and have a large safety distance between vehicles in order to guarantee a safe traveling in the road network. In addition, intersections along the road network inherently slow down the speed of the traffic stream, which may result in congestion. However, when the traffic flow rate is high and approaches the maximum capacity of the intersection, vehicles need to fully stop for periods of time. This has a significant impact on the efficiency of the transportation system.

In the work presented in this thesis, we explore Autonomous Intersection Management (**AIM**) methods based on different control strategies with the ultimate goal to develop control methods that can be deployed in operational systems. We have mainly investigated the feasibility and implementation challenges of control strategies in a fully autonomous system in the presence of communication impairments associated with wireless channels. We design a solution, a hierarchical control strategy, which is safe and robust against uncertainties, and also works for high traffic demands and speeds.

We evaluated the robustness, scalability and performance of the investigated strategies in a realistic urban mobility simulator Simulation of Urban MObility (**SUMO**) in the presence of communication impairments associated with wireless channels.

Preface

STRUCTURE OF THE THESIS

This licentiate thesis concludes my work as a licentiate candidate. The main body of the thesis consists of two parts. The first part provides a broader and more comprehensive view on the research field in which I have been working during my licentiate. The second part is composed of three included papers that constitute my main scientific work.

INCLUDED PAPERS

The following papers are form the main body of this thesis and the respective published or draft versions are appended in the back.

Paper I: SeyedeZahra Chamideh, William Tärneberg , and Maria Kihl , "Centralized Coordination of Autonomous Vehicles at Intersections", *nstitute of Electrical and Electronics Engineers Inc.*, 28 th International Conference on Software, Telecommunications and Computer Networks (SoftCOM 2020)

Paper II: SeyedeZahra Chamideh, William Tärneberg , and Maria Kihl, "Evaluation of Decentralized Algorithms for Coordination of Autonomous Vehicles at Intersections", *IEEE* , 24th IEEE International Conference on Intelligent Transportation - ITSC2021, 2021

Paper III: SeyedeZahra Chamideh, William Tärneberg , and Maria Kihl, "A Safe and Robust Autonomous Intersection Management System using a Hierarchical Control Strategy and V2I communication", submitted Aug 2021

OTHER CONTRIBUTIONS

The following publications are not included in the thesis, but summarise work that I have been also involved in.

Paper iv: SeyedeZahra Chamideh, William Tärneberg , and Maria Kihl, "Coordination ofAutonomous Vehicles at Intersections", *in 16th Swedish National Computer Networking Workshop (SNCNW 2020), Kristianstad, Sweden, May 2020.*

Paper v: Mohammadhassan Safavi, Seyedezahra Chamideh, Emma Fitzgerald , William Tärneberg, Maria Kihl, and Björn Landfeldt, "Joint Fog Service Placement and Scheduling for 5G-Enabled IoT, a Q-Learning Approach", *IEEE Transactions on Cognitive Communications and Networking,2021,*

Acknowledgments

I would first like to thank my main supervisor, Maria Kihl, for her guidance, patience and encouragement in my work. She has always supported me during my journey of PhD and her insightful feedback pushed me to sharpen my thinking and brought my work to a higher level.

A special thanks have to go to my co-supervisor, William Tärnberg, for all his help and encouragement. He deserve a big thanks for his valuable guidance and input in my work. His expertise was invaluable in formulating the research questions and methodology.

I would like to thank all the collaborators and the ones that have helped me in my research. A big thanks to all my friends and colleagues in Networked systems lab. You have all helped to create an excellent research environment. I have enjoyed all your company and appreciate all your supports during my difficult times.

Last but not least, I want to thank my family and friends. Thank you for your unyielding support and encouragement. To my dear mom, Fatemeh, thank you so much for understanding me and supporting me for every decision I have made.

Zahra
Lund, 2021

Acronyms and Symbols

Here, important acronyms, abbreviations, and symbols are listed, which are recurring throughout the thesis.

ACRONYMS AND ABBREVIATIONS

AIM	Autonomous Intersection Management
AV	Autonomous Vehicle
BBU	BaseBand Unit
C-V2X	Cellular Vehicle-to-Everything
CAM	Cooperative Awareness Message
C-AV	Cooperative Autonomous Vehicle System
C-ITS	Cooperative Intelligent Transportation System
CPM	Cooperative Perception Message
DSRC	Dedicated Short Range Communications
FCFS	First Come First Serve
GCL	Global Centralized Layer
ICU	Intersection Coordination Unit
ITS	Intelligent Transportation System
LDL	Local Decentralized Layer
LIDAR	LIght Detection and Ranging
LP	linear programming

LV	leading vehicle
MEC	Mobile Edge Computing
MILP	mixed-integer linear programming
mMTC	massive Machine Type Communications
MPC	Model Predictive Control
OBU	On Board Unit
QoS	Quality of Service
RADAR	RAdio Detection and Ranging
SUMO	Simulation of Urban MObility
TOA	Time of Arrival
TraCI	Traffic Control Interface
URLLC	Ultra reliable Low Latency Communication
V2I	Vehicle-to-Infrastructure
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-Everything
VANET	Vehicular Ad hoc Network
VOA	Velocity of Arrival

Contents

OVERVIEW OF THE RESEARCH FIELD

1 Introduction

Intersections inherently present higher risk of conflicts between those who travel through them. Intersection collisions are very common cause of injuries and fatalities in the world. According to the European Transport Safety Council's (ETSC) report on urban transportation networks, 18,844 people lost their lives in road traffic in the EU during 2020 and about 40% of these deaths occurred on urban roads [1]. Also, the Federal Highway Administration's (FHWA's) Office of Research reported that in the United States, more than half of all car accidents leading to fatal injuries occur at or near intersections [2].

Despite the development and successful deployment of innovative signalized intersections, *i.e.* controlled by traffic lights and/or signs, to improve safety, research shows that a large number of factors that contribute to an accident are directly attributable to the drivers [3]. These factors include inattentive, fatigue, and impairment, from drugs or alcohol, driving. According to reports published by [4, 5] the following human factors are very common in accident investigations:

- Drunk and fatigue driver
- Distraction and inattention
- Impaired judgment or reduced reasoning power
- Delayed or false sensation
- Poor risk perception due to lack of experience
- Violations of traffic laws

Driver assistance technologies are introduced to address safety issues associated with traditional traffic management methods. Driver assistance technologies help to identify safety risks and act or warn the driver to avoid a crash.

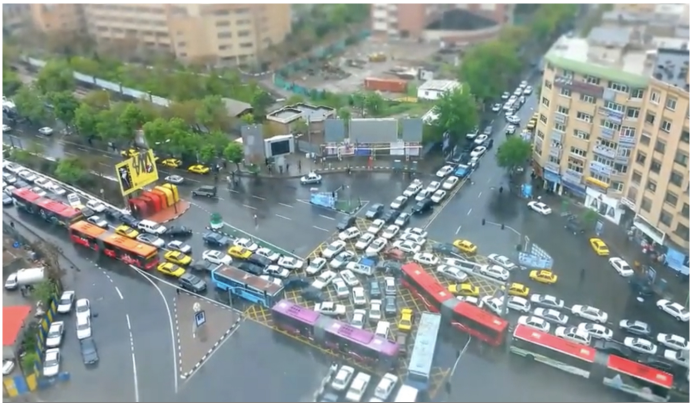

Figure 1.1: An example of signalized intersection issues

In addition to safety concerns, in urban transportation networks, traffic congestion is more likely to occur at traffic intersections. The characteristics of a signalized intersection, such as signal timing, and unnecessary braking are significant factors causing traffic congestion [6]. Traffic congestion increases the operating costs and decreases the efficiency of traffic networks by longer travel time, limited reliability of the transportation system and higher energy consumption [7].

To address the traffic congestion problem, processing and analysis of data on vehicles' mobility patterns, speed, travel time, and road network performance can improve traffic management and help to have an efficient traffic operation. Therefore, Autonomous Vehicles (**AV**s) equipped with Vehicle-to-Everything (**V2X**) communication, which allows vehicles to share road network data, have the potential to optimize the entire road network's traffic efficiency.

The continuing evolution of sensing, information processing, machine learning, control theory, and automotive technology aims to deliver even greater safety and throughput efficiency benefits. Fully automated vehicles can handle the whole task of driving and eliminate human factors.

1.1 OUTLINE

The ultimate goal of our work is to develop a safe, robust and scalable Autonomous Intersection Management (**AIM**) control strategy for a Cooperative Autonomous Vehicle System that can be deployed in operational intersection systems. To this aim, we have investigated different **AIM** approaches with different control strategies in our three papers.

This thesis is structured as follows; Chapter 2 describes the general concepts related to the research in this thesis. Section 2.1 presents required technologies for a cooperative **AIM**. Sections 2.2 and 2.3 provides our vision on the challenges of modelling the described system, and a definition of suitable performance metrics for deployment of operational **AIM** systems.

In Chapter 3, we describe our simulation framework that is used to model a close to real world environment. In Chapter 4, we provide a conclusion on how the work in each paper continued from the previous one, as well as our planned work following the research scope presented in this thesis. The second part of the thesis contains the publications.

2

Background

In this chapter, we will introduce the general concepts related to the research including required technologies for fully autonomous system, intersection modelling, control strategies. We will also present a high level discussion on challenges for implementing Autonomous Intersection Management (**AIM**) methods in real world.

2.1 REQUIRED TECHNOLOGIES FOR COOPERATIVE AUTONOMOUS INTERSECTION MANAGEMENT SYSTEM

Several key technologies are applied to serve cooperative Intelligent Transportation Systems' (**ITS**)s' objectives including sensors, wireless communication, path planning, and traffic control. In this section, we will first list communication, sensing and processing technologies essential for **AIM**. Then in the next section, we will discuss different methods to model intersections and different control strategies for **AIM**. We discuss each one of these technologies in enough detail to provide an introduction to operational requirements for **AIM**.

2.1.1 SENSING, PROCESSING AND COMMUNICATION TECHNOLOGIES FOR AUTONOMOUS INTERSECTION MANAGEMENT

New developments in software and hardware technologies have enhanced the capabilities of collecting traffic data. Vehicles are aware of their surrounding area by using built-in cameras and sensors, such as RAdio Detection and Ranging (**RADAR**) and LIght Detection and Ranging (**LIDAR**). However, to have an efficient traffic management sharing information between road network entities is crucial. Therefore, the collected data must be exchanged via a reliable wireless communication technology. Cooperative perception enables vehicles and road infrastructure to share traffic data which is built on sensing and fusion data, and communication technologies.

Generally, vehicles can be aware of their surrounding area in three different ways. Sensing can be performed by using On Board Units (**OBU**)s , such as **RADAR** and **LIDAR**. For example, Tesla equips each vehicle with eight surround cameras that provide 360 degrees of visibility around the car, and twelve updated ultrasonic sensors complement the car's visibility, allowing for detection of both hard and soft objects [8]. Also, sensing can be performed by Cooperative Awareness Message (**CAM**) where the vehicle sends out its data, e.g., position and speed, via Vehicle-to-Everything (**V2X**) communication and inform the other network entities about its situation. In addition, a combination of **OBU** sensing and **CAM** methods that is called Cooperative Perception Message (**CPM**) can be used in a Cooperative Intelligent Transportation System (**C-ITS**).

The sensor fusion is the process that combines collected data from different sources to have information with less uncertainty. Despite all advances in sensing and fusing technologies, there are still challenges such as sensor accuracy, fusion reliability, latency, and computational resources that must be considered in modelling real world systems. For sensor fusion and storage of collected data, neither on-board processing capabilities nor cloud computing will be sufficient because of the limited resource of on-board computers and the long latency and massive data transmission bottlenecks of cloud computing. To have a connected intelligent vehicles system, the storage and computing resources need to be deployed at the wireless network edge, including edge caching, edge computing using a Mobile Edge Computing (**MEC**) or Multi-access Edge Computing network architecture, running on BaseBand Unit (**BBU**) servers at base stations or radio access points along the roadside.

Various forms of wireless communications technologies have been proposed for intelligent transportation systems. For example, Intelligent Transport System G5 (ITS-G5), based on 802.11p in Europe and Dedicated Short Range Communications (**DSRC**) in the United States, is the medium range communication approach in the 5.9 GHz spectrum that is dedicated to Vehicle-to-Vehicle (**V2V**) communication. The technology enables vehicles to communicate with each other and other road users directly, without involving cellular or other infrastructures. However, current existing **ITS**s operating in the 5.9 GHz spectrum will not meet the speed and bandwidth requirements for many of the new proposed applications.

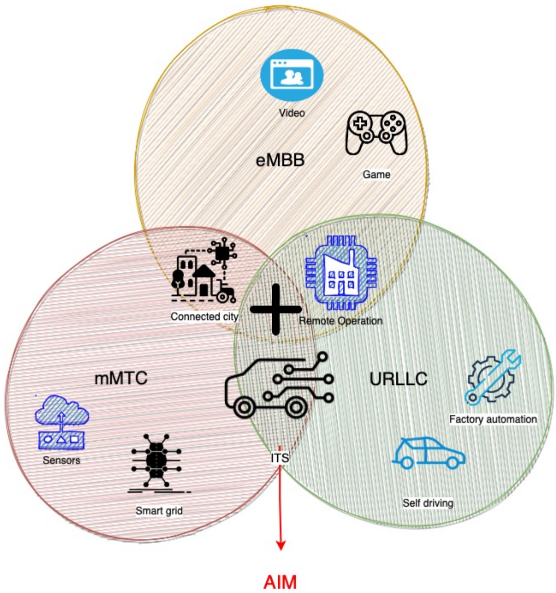

Figure 2.1: The three main 5G use cases and examples of associated applications

The essential communication in a road network may also be performed over cellular networks, such as LTE or 5G Cellular Vehicle-to-Everything (**C-V2X**) [9]. For example massive Machine Type Communications (**mMTC**) is a service category of 5G that is focused on providing connectivity to a massive number of narrow-bandwidth devices that send or receive small volumes of data such as sensors. Further, Ultra reliable Low Latency Communication (**URLLC**) is another service category of 5G aimed at mission-critical communications with a target latency of 1 millisecond and a requirement of 99.999 percent reliability [9]. In an agreement regarding the three main use cases that 5G technology must support, the communication characteristics of **ITS** and in particular **AIM** application are labelled in between the **URLLC** and **mMTC** [10], as shown in Figure 2.1. Although 5G technologies are designed for high reliability and low latency communication, The communication Quality of Service (**QoS**) highly depends on distance.

2.2 CONTROL SYSTEM

An efficient and safe management of connected Autonomous Vehicles (**AV**s obtained when all involved vehicles follow the agreed instructions. Therefore, in order to fulfill **AIM** requirements, the control system needs to be effective, safe, and robust to uncertainties. In this section we provide a methodological review on **AIM** approaches and a framework of different control strategies and intersection modelling in the literature, with critical evaluation of these works.

2.2.1 INTERSECTION MODELS

Generally, **AIM** strategies are usually based on one of the following three different approaches:

1. A reservation based approach, where the intersection is the shared resource that is being requested by vehicles, and the intersection manager's task is to make space-time reservations.
2. A trajectory planning based approach, where the vehicles' position are modelled as a function of time and the intersection manager's task is to plan collision free trajectories.
3. A collision avoidance approach in the collision region, which is a combination of reservation based and trajectory planning approaches, where the collision region of two vehicles is predictable and the intersection manager's task is to avoid that two vehicles reach their corresponding collision region simultaneously.

The reservation based approach is based on intersection discretization where discretization is a process that transfers continuous functions of space or time models into discrete counterparts [11]. By dividing the intersection into small square blocks as depicted in Figure 2.2, the small square blocks modelled as the resources must be shared between vehicles. Generally, the problem has been considered as a scheduling, resource allocation or optimization problem, where time slots and geographical space are discrete and must be allocated to passing vehicles [12]. In this model, a higher granularity, or smaller tiles, helps to model the intersection in more detail, but it introduces higher complexities for algorithm design [13].

In **AIM** methods based on trajectory planning, a vehicle's trajectory is defined as the path that the vehicle in motion follows through space as a function of time Figure 2.3. The task is to identify whether two vehicles' trajectories are in conflict or not and avoid collision by steering or acceleration controlling [14]. Several factors can have an effect on the vehicle's trajectory, and it varies with the vehicle's type, interactions between the vehicles, the road environment and the driving behavior.

However, in an intersection vehicles generally follow certain routes when passing the intersection area. This means that the path for vehicles from different directions with different intentions can be seen as *pre-defined*, as seen in Figure 2.4. Therefore, a vehicle's trajectory planning can be simplified,

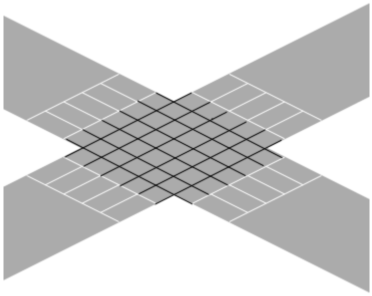

Figure 2.2: Illustration of intersection discretization

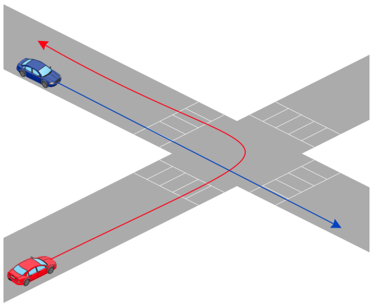

Figure 2.3: Illustration of Trajectory.

and easily be represented mathematically by a state-space model with a set of input, output and state variables. Further, the potential collision regions can be predicted by combining the above reservation based and trajectory planning approaches, and where the intersection is be modelled as a set of possible conflict points, shown in Figure 2.4. This helps to reduce the complexity of the time slots, and results in a space reservation problem [15].

2.2.2 CONTROL STRATEGIES

Papers proposing **AIM** control systems usually formulate the intersection control strategies based on three different categories *centralized*, *decentralized* or *hybrid* control strategies [13].

Centralized intersection control strategies rely on a coordination unit, which is called Intersection Coordination Unit (**ICU**) in this thesis. When a vehicle enters the coverage area of the intersection, it sends a request to the **ICU** through a wireless Vehicle-to-Infrastructure (**V2I**) communication link. The

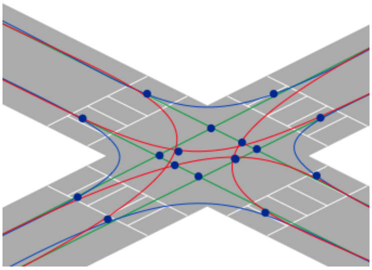

Figure 2.4: Illustration of collision regions

ICU collects information from all vehicles and make decisions based on collected data and the current road situation. The instructions are sent to each vehicle via the **V2I** communication link. [16–19].

In decentralized intersection control strategies, all vehicles communicate with each other and thereby form a Vehicular Ad hoc Network (**VANET**). Decisions are made locally by each vehicle, and the decisions are based on sensor observations and data collected through the **V2V** communication [20–22].

A hybrid intersection control strategy is a combination of centralized and decentralized control strategies [23–25]. In the hybrid control strategies, the decision can be made locally in each vehicle or globally in the **ICU**. The vehicles in the intersection area can communicate with other vehicles and the **ICU** through **C-V2X** communication links.

In this thesis, we have investigated the performance and operational issues in the **AIM** control systems for an ordinary crossroad intersection with different traffic volumes. In next section, we summarized some implementation challenges for **AIM** system with different types of control strategies.

2.3 IMPLEMENTATION CHALLENGES OF DIFFERENT CONTROL STRATEGIES

In this section, we will describe our opinions on challenges and some major drawbacks of previously proposed **AIM** methods. We discuss the operational issues of the proposed **AIM** methods that makes them insufficient to be implemented in real-world scenarios.

2.3.1 DRAWBACKS OF UNREALISTIC ASSUMPTIONS

There has been much research on **AIM** methods during the last two decades. The majority of the papers focus on a high level problem with many hypotheses to simplify the problem formulation and facilitate discussion. In most cases, the proposed **AIM** methods are focused on improving the overall performance of the intersection, for example increasing the intersection's throughput or minimizing the fuel consumption of passing vehicles. Usually, it is the optimization formulation itself that is the main research focus, not the overall system performance. Further, they are not evaluated for high vehicle densities or uncertainties caused by the wireless communication.

We believe that the essential goal of proposing an **AIM** method should be that it can be implemented in a real operational intersection. Therefore, the proposed method must guarantee a safe crossing of vehicles and make sure that all road users are protected even in places with high traffic densities and/or limited visibility. Therefore, it will be crucial to evaluate the **AIM** method in close to real world scenarios including the wireless communication.

2.3.2 CHALLENGES FOR CENTRALIZED CONTROL STRATEGIES

AIM methods based on a centralized control strategies have some major challenges. For example, the optimization problem must scale well with increasing vehicle densities. Otherwise, finding an exact solution may become intractable for realistic vehicle densities. More details can be found in our first paper (Paper I) that focus on **AIM** methods using centralized control strategies [26].

2.3.3 CHALLENGES FOR DECENTRALIZED STRATEGY

AIM methods based on decentralized control strategies have been proposed to tackle the imposed complexities of the centralized techniques [26]. However, the result may be sub-optimal due to the lack of global information. In our second paper (Paper II), we deeply discussed the implementation challenges for a decentralized **AIM**. We have shown that a decentralized control strategy is not robust to wireless channel impairments. Therefore, the solution may not be reliable in presence of packet loss and delay, that means the control system may not be able to keep a collision probability of zero.

2.3.4 CHALLENGES FOR HYBRID STRATEGY

Hybrid **AIM** methods can be developed as a solution to the challenges with the centralized and decentralized **AIM** strategies. In hybrid methods, a combination of centralized controller in the **ICU** and decentralized controller in each vehicle can be used [27]. However, in our opinion, the current published hybrid **AIM** methods have some major drawbacks. For example, The vehicle trajectory is decided by the centralized controller and then transmitted to the vehicle. This means that the local controller in each vehicle is designed to only perfectly follow commands from the central controller, which still requires a perfect centralized control strategy and a perfect wireless communication link. By not introducing any local independent intelligence in the vehicles, the vehicles cannot avoid any potential collision situations that are independent of the centralized control management, for example, situations caused by pedestrians or vehicles that do not follow the commands from the central controller.

In our third paper (Paper III), we proposed our solution to manage autonomous vehicles' movements in a safe manner at an intersection with different traffic densities based on hybrid strategy. A centralized coordinator in a higher layer that is used to balance the network load and improve the traffic efficiency, and a second layer in each vehicle with a local controller that is responsible for following the rules from the higher layer and ensure a collision free and safe crossing.

3

Simulation Framework and Evaluation Methodologies

In this chapter, we describe our simulation environment and our experiments to evaluate the performance of a cooperative autonomous intersection. Our goal is to investigate the system performance in terms of safety, scalability, and robustness. To this aim, we explain our definition of a well performed control strategy for each performance metric.

3.1 SIMULATION ENVIRONMENT

Vehicular Networks emerged as a specific application of Mobile Networks [28]. At the beginning, random models were used to represent vehicles' mobility and behaviour. However, it soon become clear that these model produced undesirable results and that they were not suitable for evaluating new protocols [29]. Therefore, the study of traffic simulators become an open topic of constant research. A traffic simulator aims to model the vehicular mobility and dynamics that are close to reality, where the distributions of vehicles and their speed is non-uniform.

Several simulation environments for vehicular mobility have been proposed. These simulation environments can be categorized according to their areas of application [30]. Microscopic traffic simulation has proven to be a useful tool for analysis of various traffic systems at vehicle level [30]. In microscopic models each individual vehicle's movements are modelled and also, the traffic data parameters, such as speed, travel time, fuel consumption, queues length, flow rate, and traffic density can be collected [31, 32].

In this thesis, we have developed a simulation environment based on Simulation of Urban MObility (**SUMO**) [32]. The aim is to have a simulation

Table 3.1: SUMO configuration

Parameter	Value	Description
step-length	0.05 s	Simulation time step
collision. mingap-factor	0 m	Collisions detection distance
collision.action	Warn	Collision warning is issued
collision.check-junctions	True	Check collisions between vehicles in the intersection

environment for heavily congested conditions at vehicle-level. **SUMO** is an open source, highly portable, microscopic and continuous traffic simulation package developed by the German Aerospace Centre DLR in 2001. **SUMO** supports the traffic simulation community, such as characteristics of various vehicle movements, driver behaviour, road topology and statistical data collection.

In our work, for development of new traffic control, the vehicles' behaviour must be adjusted dynamically, while the simulation is running. Dynamic control and **V2X** communication are not part of **SUMO**, but can be provided by external programs. This has been performed via the Traffic Control Interface (**TraCI**) that is an interface that allows a client program to retrieve information and influence the simulation over a network socket [33].

Also, we changed the values of some of **SUMO**'s default configuration parameters, as shown in Table 3.3. The vehicles' speed in **SUMO** are controlled by our program via **TraCI** commands setSpeed (0x40) and slowDown (0x14). However, a vehicle may drive slower or faster than this speed due to the car following model in **SUMO**. In order to force the vehicles to follow our control strategy, we disabled the behavior imposed by the car following model, by using the speed mode (0xb3) command and set all checks off. In each **SUMO** simulation time step, the vehicles' speed can be calculated based on the control strategy and perfectly followed.

In our simulations, the road network was modelled as a basic four-way intersection stored in a network XML-file. In the network file, each crossing road has two lanes. Each lane is 3.5m wide, and there is a maximum speed limit of about 72 km/h (v_{max} = 20 m/s). In addition, the intersection area is modelled as a circle with radius 150m.

Table 3.2: Vehicle Type Parameter

Parameter	Value	Describtion
guiShape	passenger	The vehicle shape for drawing.
length	4 m	The vehicle's length
width	2 m	The vehicle's width
minGap	0 m	Empty space after leader
accel	5 m/s^2	The acceleration ability of vehicles of this type
decel	5 m/s^2	The deceleration ability of vehicles of this type
emergency decel	9 m/s^2	The maximal physically possible deceleration for the vehicle
maxSpeed	200 km/h	The vehicle's maximum velocity

Further, all vehicles in our simulations have the same physical properties (summarized Table 3.2), and they arrive at the intersection according to a pre-generated traffic demand stored in a route XML-file. In the route file, all four entrance zone have similar traffic flow rates. The vehicles arriving at an intersection can turn right or left, or continue straight ahead. The vehicles' arrivals will be randomized using a Poisson distribution and an arriving vehicle is given a specific path when it arrives. The probability for receiving a specific path is the same for all paths. **SUMO** will report a collision when the physical gap between two vehicles is 0. In the simulation, the collided vehicles will immediately be removed by **TraCI** and the rest of the traffic will continue as before.

3.2 EVALUATION APPROACHES

The ultimate goal of Autonomous Intersection Management systems are to improve the safety, efficiency, and sustainability of transportation networks. This will be achieved by less traffic congestion and less number of accidents, and a decreased severity of those accidents. We believe that safety improvement is the most important potential benefit of **AIM** systems. However, the

Table 3.3: SUMO configuration

Parameter	Value	Description
step-length	0.05 s	Simulation time step
collision. mingap-factor	0 m	Collisions detection distance
collision.action	Warn	Collision warning is issued
collision.check-junctions	True	Check collisions between vehicles in the intersection

other goal, congestion reduction, may have negative impacts on safety. In this section, we describe, in our opinion, the necessary evaluation metrics to verify the performance of an intersection controlled by **AIM** in terms of safety, scalability, and robustness to uncertainties.

Since an **AIM** must be safe in all traffic conditions specially high traffic densities, we have evaluated the system performance in different traffic volume, also called *traffic flow rates*. The traffic flow rate is normally given in terms of arriving vehicles per hour per lane. All results will be compared with a conventional intersection control based on traffic lights with 90 second green phase and 90 second red phase.

The *saturation flow rate* for an intersection corresponds to the maximum achievable traffic flow rate when there is a high traffic demand. For an intersection controlled by traffic lights, the saturation flow rate depends on several factors, such as the intersection geometry, safety policy and the surrounding environment [34]. The study in [34] shows that the saturation flow rate for an intersection controlled by traffic light is almost 900 vehicles per hour per lane. The saturation flow calculation is based on a 2 seconds headway between vehicles for safety.

A saturated intersection corresponds to a situation where the maximum number of vehicles coexist inside the intersection, and the vehicles have minimum possible safe distance of $d_{min}(m)$, and assuming that the vehicles have an average speed of $\bar{V}(m/s)$. Since the intersection is full of vehicles, at least one vehicle need to leave the intersection for another vehicle to enter. The time it takes for the first vehicle at the end of an exit lane to leave the intersection zone is $t_h = d_{min}/\bar{V}(s)$. In an intersection with N_2 exit lanes and N_1 entrance lanes, the average number of N_2 vehicles can leave the intersection during period t_h simultaneously. That means that at maximum of N_2 vehicles can enter from N_1 entrance lanes, We are looking for the maximum achievable traffic flow rate for each lane. Since in each second, maximum $min(N_2, N_1)/t_h$ vehicles can enter from N_1 lanes, the maximum traffic flow rate, the system capacity, in terms of vehicles per hour, can be obtained from Equation (3.1).

$$C = \frac{3600 * min(N_2, N_1)}{t_h \cdot N_1} \tag{3.1}$$

In our papers, the traffic flow rate is divided in three different *traffic volumes*, as in [35]. The peak hour traffic flow rate for a typical intersection in an urban area is usually 450-650 vehicles/hour/lane, which is defined as *High volume traffic*. A traffic flow rate of 150-450 vehicles/hour/lane is defined as *Medium volume traffic*. Finally, a traffic flow rate of less than 150 vehicles/hour/lane is defined as *Low volume traffic*. In the result graphs, we will highlight the different traffic volumes by using different background colours: *Green* for traffic flow rates corresponding to Low volume traffic, *yellow* for Medium volume traffic and *red* for High volume traffic.

3.3 PERFORMANCE METRICS

In this thesis, we use several performance metrics for the evaluation. In this section, we describe some of these metrics and how we calculate them in our simulation environment **SUMO**. The first two performance metrics, *average speed* and *fuel consumption* are the most used performance metrics in the literature to evaluate the efficiency of **AIM** methods. We also include *safety* since it will be crucial when deploying an operational **AIM** system.

3.3.1 AVERAGE SPEED

From Equation (3.1), a higher average speed means a higher capacity for the intersection. Therefore, the system performs well when vehicles can pass the intersection with a high speed. We have evaluated the *average speed* of all vehicles in the intersection for different traffic flow rates. The average speed for vehicle i, $\bar{V}_i$ is calculated as in Equation (3.2).

$$\bar{V}_i = \frac{\int^{Tr_i} v_i(t)}{Tr_i} \tag{3.2}$$

Where Tr_i is the total time vehicle i spends on its path in intersection zone and is called *Traveling time* of vehicle i. The $v_i(t)$ denote the speed of vehicle i at time t. The average speed of all vehicles during a simulation run, $\bar{V}$, is obtained by calculating the average speed of all vehicles that have passed the intersection during the simulation, as shown in Equation (3.3).

$$\bar{V} = \frac{\sum_i^{N_l} \bar{V}_i}{N_l} \tag{3.3}$$

where N_l is the total number of vehicles that have passed the intersection during the simulation.

3.3.2 FUEL CONSUMPTION

Energy efficiency and emission reduction are other promising benefits of **ITS**. An **AIM** method performs well if it results in a low fuel consumption (or electricity for electric cars) for the vehicles. Several factors can have an effect on the fuel consumption, and it varies with the vehicle type, weather condition, driving behaviors such as rapid acceleration, and speed. The Environmental Protection Agency (EPA) study [36] shows that the acceleration rates have a significant effect on a vehicle's fuel consumption. Therefore, a smooth flow of vehicles, and thereby, a smooth change of acceleration is desired. In this thesis, we have used the *average absolute acceleration* for different traffic flow rates as a metric for fuel consumption. The average absolute acceleration for vehicle i, denoted U_i, during its traveling time Tr_i is calculated as shown in Equation (3.4).

$$U_i = \frac{\int^{Tr_i} |u_i(t)|}{Tr_i} \tag{3.4}$$

Where $u_i(t)$ is the acceleration of vehicle i at time t. The average absolute acceleration for all vehicles during a simulation, U, is obtained by calculating the average absolute acceleration of all vehicles that have passed the intersection during the simulation, as shown in Equation (3.5).

$$U = \frac{\sum_i^{N_l} U_i}{N_l} \tag{3.5}$$

3.3.3 TRAFFIC SAFETY

An operational **AIM** system needs to be totally safe. In this thesis, we have used an approximation of *average number of collisions per hour* as the main performance metric for traffic safety. As explained in Section 3.1, a collision is defined when the physical distance between two vehicles is zero. For each traffic flow rate, we ran the simulation several times with different random seeds (i.e different traffic demand profiles), where each simulation run was 1 hour. We measured the number of collisions that **SUMO** detected during each simulation run, and then calculated an average of the number of detected collisions per hour, denoted $\bar{N}_c$, as shown in Equation (3.6).

$$\bar{N}_c = \frac{\sum_k^{N_r} N_C^k}{N_r} \tag{3.6}$$

where N_r is the total number of simulation runs and N_C^k is the number of detected collisions in simulation run k. The resulting metric, $\bar{N}_c$, should be seen as an approximation, not a fact. For example, we used more simulation runs for lower traffic flow rates than for higher traffic flow rates. The main aim with this metric is to evaluate if an **AIM** method can be considered safe or not. Therefore, the absolute values of $\bar{N}_c$ are not relevant, only the comparison between methods.

3.3.4 ROBUSTNESS

Operational **AIM** systems need to be robust to uncertainties caused by the wireless communication. In this thesis, we have evaluated the effect on the safety when adding packet loss and communication delays, and used these results as a performance metric for robustness of the system.

3.3.5 SCALABILITY

Another requirement for operational systems is that an **AIM** method must be able to handle a large number of vehicle movements. Therefore, it is important to evaluate the deployed algorithms' complexity, execution time, problem size, and maximum traffic flow rate that the **AIM** method can safely control.

4

Summary and Contributions

4.1 RESEARCH CONTRIBUTIONS

The three papers included in this thesis are summarised below, which illustrates the path of our investigations on **AIM** systems. This chapter gives an overview on the content of each paper, and detail my main contributions in each work and potential future work.

We have investigated the performance and operational issues in the Autonomous Intersection Management systems for an ordinary crossroad intersection with different traffic volumes. We started with examining the feasibility of deploying **AIM** methods based on centralized control strategies (Paper I), and continued by investigating the challenges of **AIM** methods based on decentralized control strategies (Paper II). Then, we proposed a hierarchical control strategy (Paper III) based on our experiences and conclusions from Paper I and Paper II.

4.1.1 PAPER I: CENTRALIZED COORDINATION OF AUTONOMOUS VEHICLES AT INTERSECTIONS

In this paper, we compared two well-cited [16, 37] **AIM** methods based on centralized strategy in a realistic simulation environment. We investigated the safety and possibility of implementing the proposed algorithms in the real world. This side by side comparison helped us to gain insight into the strengths and limitations of these types of control systems. Our investigation verified the improvement of common performance criteria, such as energy consumption, travel time and throughput, in comparison with a conventional signalized intersection. But, our simulations show that the safety conditions are not satisfied in high traffic densities, since the collision probabilities rather quickly become larger than zero when the traffic rate increases. This means that **AIM** systems based on centralized control strategies can only can be used for low traffic rates. In addition, the problem complexity, in particular for high traffic densities, may make these type of **AIM** methods infeasible for real time traffic management scenarios.

Contribution:

I was the primary researcher in this work and my contribution is in system definition and modelling. I also created my simulation framework based on **SUMO**, which allows vehicles to follow my control rules instead of default control in Simulation of Urban MObility (**SUMO**). I used the simulation framework for performance evaluation and analysis.

For this thesis, the paper has been formatted to match the rest of the thesis.

4.1.2 PAPER II: EVALUATION OF DECENTRALIZED ALGORITHMS FOR COORDINATION OF AUTONOMOUS VEHICLES AT INTERSECTIONS

In this paper, we study the safety, scalability, and performance of **AIM** methods based on decentralized control strategies, in the presence of communication impairments associated with wireless channels. Two well-cited [38, 39] **AIM** methods are evaluated in realistic simulations in **SUMO**. We investigate the safety and feasibility of deploying this type of control strategies in the real world operational traffic systems. As for centralized control strategies, it can be shown that decentralized schemes are completely safe and well-performing for low and medium traffic flow rates. However, our results clearly show that the safety conditions are not guaranteed for high traffic densities, and therefore, **AIM** methods based on decentralized control strategies would only be usable for low traffic rates. Also, the algorithms are not robust to wireless channel impairments that results in packet loss. This means that the control system performance highly depends on wireless channel reliability.

Contribution :

I was the primary researcher in this work and my contribution is in system definition and modelling. I used my simulation framework in previous work [26] for a fair performance evaluation and comparison between different decentralized/centralized control algorithms.

For this thesis, the paper has been formatted to match the rest of the thesis.

4.1.3 PAPER III: A SAFE AND ROBUST AUTONOMOUS INTERSECTION MANAGEMENT SYSTEM USING A HIERARCHICAL CONTROL STRATEGY AND V2I COMMUNICATION

In this paper, based on our knowledge of the strengths and limitations of **AIM** methods based on centralized and decentralized control strategies, we propose a new hierarchical control strategy to manage **AV**s movement at intersections. In our proposed **AIM** method, the Intersection Coordination Unit (**ICU**) in a Global Centralized Layer (**GCL**) is responsible for assigning a safe speed to each vehicle inside the intersection, and ensure that vehicles can cross the intersection without collision, while maximizing the intersection's throughput. In the Local Decentralized Layer (**LDL**), each vehicle is responsible for tracking the reference speed assigned by the **ICU**, while avoiding collisions. In our proposed **AIM** method, each vehicle can use its own sensors to monitor its close surroundings, and thereby can take its own decisions on its movements, independent on the control decisions sent from the **ICU**. This means that our proposed **AIM** method only requires **V2I** communication, and no **V2V** communication. We investigate the safety, scalability and robustness of our proposed **AIM** method compared with two **AIM** methods based on centralized and decentralized control strategies. Our simulation results show that the proposed **AIM** method can safely handle high traffic flow rates. Also, our simulations results show the robustness of our proposed method to uncertainties caused by the wireless communication.

Contribution :

In this work, we proposed our own solution for **AIM** problem. I was the primary researcher in this work and my contribution is in system definition and modelling, solution design, simulation development, performance evaluation and analysis.

For this thesis, the paper has been formatted to match the rest of the thesis.

4.2 CONCLUSIONS AND FUTURE WORK

In this thesis, we have looked at Autonomous Intersection Management methods in the context of future Intelligent Transportation System. **AIM** has the potential to increase the road network safety and efficiency by eliminating human factors in accident investigations and balancing the roads network traffic.

In our research, we focused on control strategies for managing autonomous vehicles' movements. We have investigated implementation issues of previously proposed **AIM** control strategies in real world scenarios in our first and second papers (Paper I, II). We also proposed our hierarchical control strategy in Paper III to tackle with operational limitations imposed by centralized and decentralized systems.

Despite of all advances in technologies, it will take a long time to transition to fully autonomous systems. Therefore, intersection control mechanism for autonomous vehicles management must be compatible with human drives. In addition, pedestrians and cyclists must also be able to cross intersections in a safe manner. We will extend our work in Paper III to accommodate scenarios including humans, whether they are on a bicycle, crossing the intersection, or driving a conventional non-autonomous car. Further, a real time optimization of the vehicles' traffic flow through a network of multiple intersections can be an interesting practical challenge for our future work.

In addition, the deployment of a cooperative **ITS** is expected to lead to better traffic management by increasing efficiency and safety. Therefore, before autonomous vehicles hit the streets, it is necessary to build new technologies, testbeds and applications that will give us insight into this new technologies. We plan to build our testbed by using mini-cars, e.g. 1/10 scale model cars, to test self-driving technology and in particular **AIM** application. To build our testbed we hope to do a collaboration in a variety of research areas such as path planning, accurate positioning, sensor fusion, energy efficiency, and cyber-security.

References

[1] J. C. Dovilé Adminaité-Fodor and G. Jost, "15th road safety performance index report," The European Transport Safety Council (ETSC), Tech. Rep., June 2021.

[2] T. F. H. Administration, "Intersection safety," August 26, 2021. [Online]. Available: https://highways.dot.gov/research/research-programs/safety/intersection-safety

[3] F. Chen, M. Song, and X. Ma, "Investigation on the injury severity of drivers in rear-end collisions between cars using a random parameters bivariate ordered probit model," *International journal of environmental research and public health*, vol. 16, no. 14, p. 2632, 2019.

[4] W. H. Organisation, "Road traffic injuries," June 21,2021. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries

[5] UIIG, "Unsignalized intersection improvement guide," 2015. [Online]. Available: http://toolkits.ite.org/uiig/problems.asp

[6] S. Yuan, X. Zhao, and Y. An, "Identification and optimization of traffic bottleneck with signal timing," *Journal of Traffic and Transportation Engineering (English Edition)*, vol. 1, no. 5, pp. 353–361, 2014.

[7] J. Yang, A.-O. Purevjav, and S. Li, "The marginal cost of traffic congestion and road pricing: Evidence from a natural experiment in beijing," *American Economic Journal: Economic Policy*, vol. 12, no. 1, pp. 418–53, 2020.

[8] Tesla, "Future of driving," 2021. [Online]. Available: https://www.tesla.com/autopilot

[9] "Study on self-evaluation towards imt-2020 submission. available online:," https://www.3gpp.org/ftp//Specs/archive/37_series/37.910/, accessed: 20 January 2021.

[10] O. Teyeb, G. Wikström, M. Stattin, T. Cheng, S. Faxér, and H. Do, "Evolving lte to fit the 5g future," *Ericsson technology review*, vol. 1, pp. 1–16, 2017.

[11] K. Dresner and P. Stone, "Multiagent traffic management: A reservation-based intersection control mechanism," in *Autonomous Agents and Multiagent Systems, International Joint Conference on*, vol. 3. IEEE Computer Society, 2004, pp. 530–537.

[12] J. Lee and B. Park, "Development and evaluation of a cooperative vehicle intersection control algorithm under the connected vehicles environment," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 1, pp. 81–90, 2012.

[13] L. Chen and C. Englund, "Cooperative intersection management: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 2, pp. 570–586, 2015.

[14] J.-B. Tomas-Gabarron, E. Egea-Lopez, and J. Garcia-Haro, "Vehicular trajectory optimization for cooperative collision avoidance at high speeds," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 4, pp. 1930–1941, 2013.

[15] I. H. Zohdy, R. K. Kamalanathsharma, and H. Rakha, "Intersection management for autonomous vehicles using icacc," in *2012 15th international IEEE conference on intelligent transportation systems*. IEEE, 2012, pp. 1109–1114.

[16] M. A. S. Kamal, J.-i. Imura, T. Hayakawa, A. Ohata, and K. Aihara, "A vehicle-intersection coordination scheme for smooth flows of traffic without using traffic lights," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 3, pp. 1136–1147, 2014.

[17] C. Wuthishuwong, A. Traechtler, and T. Bruns, "Safe trajectory planning for autonomous intersection management by using vehicle to infrastructure communication," *EURASIP Journal on Wireless Communications and Networking*, vol. 2015, no. 1, pp. 1–12, 2015.

[18] Q. Ge, Q. Sun, Z. Wang, S. E. Li, Z. Gu, and S. Zheng, "Centralized coordination of connected vehicles at intersections using graphical mixed integer optimization," *arXiv preprint arXiv:2008.13081*, 2020.

[19] Y. Guan, Y. Ren, S. E. Li, Q. Sun, L. Luo, and K. Li, "Centralized cooperation for connected and automated vehicles at intersections by proximal policy optimization," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 11, pp. 12 597–12 608, 2020.

[20] Y. Wu, H. Chen, and F. Zhu, "Dcl-aim: Decentralized coordination learning of autonomous intersection management for connected and automated vehicles," *Transportation Research Part C: Emerging Technologies*, vol. 103, pp. 246–260, 2019.

[21] X. Qian, J. Gregoire, A. De La Fortelle, and F. Moutarde, "Decentralized model predictive control for smooth coordination of automated vehicles at intersection," in *2015 European control conference (ECC)*. IEEE, 2015, pp. 3452–3458.

[22] L. Makarem and D. Gillet, "Fluent coordination of autonomous vehicles at intersections," in *2012 IEEE international conference on systems, man, and cybernetics (SMC)*. IEEE, 2012, pp. 2557–2562.

[23] V. Digani, L. Sabattini, C. Secchi, and C. Fantuzzi, "Hierarchical traffic control for partially decentralized coordination of multi agv systems in industrial environments," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 6144–6149.

[24] V. Muthukumaran, R. G. Sanfelice, and G. H. Elkaim, "A hybrid control strategy for autonomous navigation while avoiding multiple obstacles at unknown locations," in *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2019, pp. 1042–1047.

[25] N. R. Kapania, V. Govindarajan, F. Borrelli, and J. C. Gerdes, "A hybrid control design for autonomous vehicles at uncontrolled crosswalks," in *2019 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2019, pp. 1604–1611.

[26] S. Chamideh, W. Tärneberg, and M. Kihl, "Centralized coordination of autonomous vehicles at intersections," in *28 th International Conference on Software, Telecommunications and Computer Networks (SoftCOM 2020)*. Institute of Electrical and Electronics Engineers Inc., 2020.

[27] R. Hult, M. Zanon, G. Frison, S. Gros, and P. Falcone, "Experimental validation of a semi-distributed sequential quadratic programming method for optimal coordination of automated vehicles at intersections," *Optimal Control Applications and Methods*, vol. 41, no. 4, pp. 1068–1096, 2020.

[28] C. K. Toh, *Ad hoc mobile wireless networks: protocols and systems*. Pearson Education, 2001.

[29] M. J. Silva, G. I. Silva, C. M. Ferreira, F. A. Teixeira, and R. A. Oliveira, "Survey of vehicular network simulators: A temporal approach," in *International Conference on Enterprise Information Systems*. Springer, 2018, pp. 173–192.

[30] M. Saidallah, A. El Fergougui, and A. E. Elalaoui, "A comparative study of urban road traffic simulators," in *MATEC Web of Conferences*, vol. 81. EDP Sciences, 2016, p. 05002.

[31] M. Fellendorf and P. Vortisch, "Microscopic traffic flow simulator vissim," in *Fundamentals of traffic simulation*. Springer, 2010, pp. 63–93.

[32] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner, "Microscopic traffic simulation using sumo," in *The 21st IEEE International Conference on Intelligent Transportation Systems*. IEEE, 2018. [Online]. Available: <https://elib.dlr.de/124092/

[33] M. Behrisch, L. Bieker, J. Erdmann, M. Knocke, D. Krajzewicz, and P. Wagner, "Evolution of sumo's simulation model," *Transportation Research Board Circular*, pp. 1–21, 2014.

[34] E. Aoyama, K. Yoshioka, S. Shimokawa, and H. Morita, "Estimating saturation flow rates at signalized intersections in japan," *Asian Transport Studies*, vol. 6, p. 100015, 2020.

[35] J. Zheng and H. X. Liu, "Estimating traffic volumes for signalized intersections using connected vehicle data," *Transportation Research Part C: Emerging Technologies*, vol. 79, pp. 347–362, 2017.

[36] R. Jones, "Quantitative effects of acceleration rate on fuel consumption. technical report," Environmental Protection Agency, Ann Arbor, MI (USA), Tech. Rep., 1980.

[37] I. H. Zohdy and H. A. Rakha, "Intersection management via vehicle connectivity: The intersection cooperative adaptive cruise control system concept," *Journal of Intelligent Transportation Systems*, vol. 20, no. 1, pp. 17–32, 2016.

[38] A. Katriniok, P. Kleibaum, and M. Joševski, "Distributed model predictive control for intersection automation using a parallelized optimization approach," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 5940–5946, 2017.

[39] G. R. de Campos, P. Falcone, R. Hult, H. Wymeersch, and J. Sjöberg, "Traffic coordination at road intersections: Autonomous decision-making algorithms using model-based heuristics," *IEEE Intelligent Transportation Systems Magazine*, vol. 9, no. 1, pp. 8–21, 2017.

PAPERS

Paper I

Paper I

Reproduced, with permission, from

SeyedeZahra Chamideh and William Tärneberg , and Maria Kihl, "Centralized Coordination of Autonomous Vehicles at Intersections," *28 th International Conference on Software, Telecommunications and Computer Networks (SoftCOM 2020)*, 2020.

I

Centralized Coordination of Autonomous Vehicles at Intersections

Recent advances in autonomous vehicles present new opportunities in Intelligent transportation systems (**ITS**)s to address urban transport challenges. Therefore, urban traffic scenarios, and in particular intersections as a bottleneck of transportation network, has received significant attention. In this paper we investigate intelligent traffic control mechanisms for autonomous vehicles at intersections as a replacement of traditional intersection control. An edge cloud controller is used to deliver services that provide traffic safety and efficiency to vehicles. Two well-cited optimization algorithms for cooperative vehicles are compared with realistic simulations in SUMO. We investigated the safety and possibility of implementing the proposed algorithms in the real world. This side by side comparison helps to gain insight into the strengths and limitations of these types of algorithms.

1 INTRODUCTION

Intelligent Transportation Systems and in particular autonomous vehicles (AVs) will likely have significant effect on future traffic management systems. As automated vehicles become more common, the traffic control strategies, for example intersection management (IM), have to be improved in order to increase the driving safety. Road intersections are currently managed by using traffic lights, which often result in many vehicles unnecessarily braking and can significantly increase travel times. By leveraging the capacities of AVs, it is possible to remove traditional intersection managers and rely on coordination among the involved vehicles at an intersection.

In cooperative intersection control, there is usually an intersection control unit that can exchange information with the vehicles. In this paper, we assume that all vehicles crossing the intersection can be manipulated by the control unit through two-way communication. Cooperative intersection control could, besides providing safe crossings for the vehicles, optimize overall costs such as travel times, traffic throughput and fuel consumption.

The idea of Autonomous Intersection Management (AIM) was first proposed by Drenser and Stone [1]. They designed a central decision maker, which manages time-space reservations in the intersection to avoid collisions. After that, a number of papers has been published on methods to design the best intersection management system as well as several metrics to evaluate such systems. For example, a vehicle scheduling problem was proposed in [2]. In their algorithm, vehicle agents are allowed to determine control actions among a set of controlled inputs. Another example is [3], where the authors proposed a decomposition scheme that gives an approximation solution to an optimal control problem. Further, a convex modelling for optimal control of autonomous vehicles at intersections was provided in [4]. Their proposed method includes problem transformation from time to space domain. Also, in [5], a communication strategy was proposed that minimized the use of communication resources for the intersection management.

In Kamal *et al.* [6] a Model Predictive Control (MPC) problem is formulated that generates feasible trajectories for autonomous vehicles. Their control algorithm optimises the control inputs of the vehicles in a given time horizon to minimise the risk of cross-collisions. They assume a constant number of vehicles approaching to the intersection and input traffic into all the sections is set at equal rate.

In Zohdy *et al.* [7], a system for intersection control is developed that optimizes vehicle trajectories within an intersection zone. In their strategy, vehicles pass the critical area with a time difference to avoid collision. They assume that vehicles approach the intersection zone with their maximum possible velocity and the proposed mechanism reduce this speed if it is required.

The overall aim of our research work is to develop a robust centralised controller in a scenario where vehicles are cooperative and connected to an edge cloud based on a 5G infrastructure. The controller should generate optimal route decisions for all vehicles based on the intersection state, and then communicate these decisions to the vehicles that follows the decisions when crossing the intersection. Obviously, an autonomous intersection control algorithm must guarantee total safety for passengers, that is, no collision can occur, which means that the collision probability must be zero for all possible scenarios. Further, the execution time of an algorithm must be limited, in order for it to be feasible to run in real-time.

In this paper, we compare the previously mentioned algorithms [6] [7] by implementing them in the realistic simulation environment SUMO [8]. We have chosen these two algorithms because they have different optimization objectives for the same type of intersection control, and they are well cited in the literature. Both algorithms have been shown to work well in numerical simulations and theoretical analysis. In this paper, we evaluate the safety and possibility of implementing the proposed algorithms in the real world. Our investigation shows the performance of the two algorithms compared with traditional signalised method. Our main conclusion is that these algorithms can only be used for low traffic densities, since the collision probability rather quickly becomes larger than zero when the traffic increases. Also, the execution times of the algorithms makes them rather infeasible for realistic traffic scenarios.

2 TARGETED SYSTEM

In urban transportation network, intersections are a bottleneck in generating traffic congestion. Traffic flow pattern in an intersection depends on its geometry, location and possible movement to and from its various lanes. Congestion wastes a massive amount of time, fuel and creates more uncertainty for traveller. Coordinated intersection traffic management is an important component of the intelligent transportation system. It enables a vehicle to communicate with roadside equipment or other vehicles, and help to improve the road traffic safety and efficiency.

We consider the problem of autonomous vehicle coordination at a crossroad intersection without traffic lights, as depicted in Figure 1, The system is composed of vehicles equipped with On Board Units (**OBU**)s, which may employ a wide range of sensors types, and an Intersection Coordination Unit (**ICU**) that will be deployed at the intersection and act as a computing

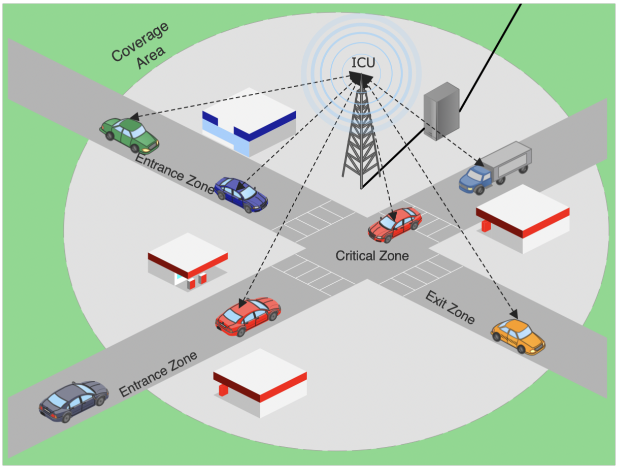

Figure 1: Targeted System and Intersection layout

resource pool. The **OBU**s and the **ICU** communicates via some type of high bandwidth radio communication, for example using 5G. However, the focus in this paper is on the control part, not the communication part, and therefore, the communication technology is not specified, and we just assume a wireless communication link with negligible packet loss and delay.

We consider a typical crossroad intersection where vehicles are allowed to make left and right in addition to through movement. We split the intersection area into three zones: the entrance zone, the critical zone and the exit zone. The entrance zone represents the area where vehicles approaches the intersection boundary. In the critical area, there is a risk of lateral vehicle collisions. The exit zone includes vehicles leaving the intersection.

Each vehicle periodically sends its status information to the **ICU**, as part of the ITS facilities layer [9]. The ICU will periodically orchestrate vehicles, aggregate data, and provide control based on the system's objectives. Also, the **ICU** will determine whether there is any danger according to the driving status of the vehicles. Collisions between two vehicles are prevented by controlling the speed of the vehicles. All vehicles are assumed to always follow the ICU's decisions.

3 INVESTIGATED ALGORITHMS

In this paper, we investigate the performance of two AIM algorithms, the *Model Predictive Control* Algorithm [6] and the *Delay Minimization* Algorithm

[7], by implementing them in the realistic simulation environment SUMO [8]. In this section, we first describe our general system model and then present high level descriptions of the two algorithms.

3.1 SYSTEM MODEL

In this section we present the system model that is used by the AIM algorithms for intersection control. A collection of $\mathcal{N} = \{1, 2, \ldots, N\}$ autonomous vehicles approaches a coordination area of an intersection with L lanes. For each $i \in \mathcal{N}$ a predetermined path is given and perfectly followed.

The vehicle dynamics are described as a second order integrator, where the vehicle is modeled as a point on the path coordinates. Given vehicle i, $p_{i,t}$ is defined to show the position (distance from the beginning of the critical zone) at time t and $v_{i,t} = \dot{p}_{i,t}$ is the speed of vehicle and $u_{i,t} = \ddot{p}_{i,t}$ the acceleration [10]. The longitudinal motion of each vehicle, $x_{i,t} = [p_{i,t}, v_{i,t}]^\intercal$, can be controlled by its acceleration. We assume the control input is updated in discrete time τ. The discrete time state model of vehicle i at time $t\tau$ is given in equation (1).

It is assumed that the vehicles will follow the acceleration decided by ICU through communication link describe in Section 2. Each vehicle will transmit its basic driving information, including current position, velocity, and destination once entered into the intersection zone to initialised the problem at the ICU.

$$x_{i,t+1} = \begin{bmatrix} 1 & -\tau \\ 0 & 1 \end{bmatrix} x_{i,t} + \begin{bmatrix} -\frac{1}{2}\tau^2 \\ \tau \end{bmatrix} u_{i,t} \tag{1}$$

We consider a limited speed and acceleration: $v_{i,t} \in V_i = [v_{min}, v_{max}]$ and $u_{i,t} \in U_i = [u_{min}, u_{max}]$. However, maximum and minimum acceleration of each vehicle depends on its current velocity and speed limitation.

3.2 MODEL PREDICTIVE CONTROL ALGORITHM

In this section, we give a high-level description of the AIM algorithm for intersection control proposed in [6]. This algorithm will in the rest of the paper be called the *Model Predictive Control (**MPC**) algorithm*. The algorithm defines an optimization problem that minimize the risk of collision between a pair of vehicles at their possible conflict point during a finite time horizon. Therefore, the algorithm introduces a risk function $\mathcal{F}^t_{i,j}$ that is used to determine whether a vehicle pair (i, j) poses a potential risk of collision at time t. The risk function is given in equation (2) below:

$$\mathcal{F}^t_{i,j} = \delta_{i,j} exp\{-\alpha_i(p_{i,t} + C_{ij})^2 - \alpha_j(p_{j,t} + C_{ji})^2\} \tag{2}$$

Here, α_i and α_j are positive constants that depend on the two vehicles' sizes. $\delta_{i,j}$ is a binary variable that states whether the pair vehicles (i, j) have the potential to collide or not. C_{ij} and C_{ji} are the distances from the conflict point of the pair vehicles (i, j) to the beginning of the critical zone at intersection for vehicle i and j respectively.

In order to avoid any rear-ends collisions, a minimum separation distance between two vehicle on the same lane, d_{min}, is defined. Since the vehicles are modeled as a point on their path, In the real world d_{min} is the minimum distance between the centre point of two vehicles. The following constraint in Equation (3) is defined to prevent rear-ends collisions between vehicles i and j at time t:

$$|p_{i,t} - p_{j,t}| \geq d_{min} \tag{3}$$

The variable $p_{i,t}$ will always have a positive value in the entrance zone. For ensuring that no collisions occur between two vehicles form different approaching lanes inside the critical zone, a linear inequality constraint is defined as in equation (4) below:

$$p_{i,t} + C_{ij} + p_{j,t} + C_{ji} \geq R_{min} \tag{4}$$

Here, R_{min} is a constant that denotes the minimum separation distance between the centers of two vehicles from different approaching path.

A Model Predictive Control (MPC) framework is used to minimize the system's cost over predefined time horizon. An MPC problem with the time horizon of T steps allows the system be optimized in current time slot, while keeping next $T - 1$ time slots in account [11], [12]. For this purpose at each time step an optimization problem is solved to drive the optimal control input for the system by predicting the system state over the defined time horizon.

The objective of the optimization is to achieve a smooth and comfortable flow of vehicles where the vehicles cross the intersection with almost constant and high speed, while minimizing the risk of collisions and energy consumption. Therefore, The system cost, J is defined as in Equation (5). One of the term in cost function attempts to minimise error between the speed of vehicle i and its desired speed, v_d^i, Minimizing the acceleration, $u_{i,t}$ and the last term related to collision avoidance risk function.

$$\begin{aligned} J = & \sum_{t=0}^{T-1} \sum_{i=1}^{N} w_{v_i} (v_{i,t+1} - v_d^i)^2 \\ & + \sum_{t=0}^{T-1} \sum_{i=1}^{N} w_{u_i} (u_{i,t})^2 + \sum_{t=0}^{T-1} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} w_f \mathcal{F}_{i,j}^t \end{aligned} \tag{5}$$

Here, w_{v_i} ,w_{u_i} and w_f are weight coefficients and J is the problem objective to be minimised, subject to the given current states of the vehicles as defined in Equation (1) and constraints as defined in Equations (3) and (4). The speed and acceleration is bounded as described in Section 3.1.

3.3 DELAY MINIMIZATION ALGORITHM

In this section, we give a high-level description of the AIM algorithm for intersection control proposed in [7]. The algorithm will in the rest of the paper be called the *Delay minimization algorithm*. In this algorithm, vehicles are assumed arrive at entrance zone with their maximum allowed speed. The ideal profile entails traveling the entire intersection zone without deceleration. This means that in the absence of obstacles, a vehicle should be able to cross the intersection at the same maximum speed. In order to avoid collisions, The algorithm adjusts the vehicles' speed, so that all vehicles can cross the intersection at their respective maximum movement speed without colliding with other vehicles.

Decisions of arrival times of each vehicle to critical zone are made by the optimization module. The objective of the algorithm is to find the optimal deceleration to minimize the total traveling time for all vehicles inside the intersection by considering the safety criteria. The minimum time for a vehicle to travel between the beginning of entrance zone and the beginning line of critical zone without deceleration is called the optimum time, denoted OT_i for vehicle i. The algorithm tries to minimise the extra delay (D_i) that is added to the optimum time in case of necessary deceleration.

In order to avoid any rear-ends collision, a minimum separation headway time of H_{min} between two vehicle on the same lane is defined.

$$|(OT_i + D_i) - (OT_j + D_j)| \geq H_{min} \tag{6}$$

To ensure that no collisions occur between a pair of vehicles form different approaching lanes inside the critical zone, the vehicles have to pass their possible conflict point with a minimum time separation $\Delta\tau$. Therefore, a linear inequality constraint is defined as in equation (7) below:

$$|(OT_i + D_i + \tau_{ij}) - (OT_j + D_j + \tau_{ji})| \geq \Delta\tau\delta_{i,j} \tag{7}$$

Here, $\delta_{i,j}$ is a binary variable that states whether the pair vehicles (i, j) have the potential to collide or not. τ_{ij} and τ_{ji} are defined as the travelling times from the conflict point of pair (i, j) to the beginning of the lane i and j respectively.

The system cost, J is defined as the sum of the required delay for all vehicles inside the intersection zone to avoid collision. J is given by Equation (8).

$$J = \sum_{i=1}^{N1} D_i \tag{8}$$

Here $\mathcal{N}_1 = \{1, 2, \ldots, N1\}$ is the set of vehicles that approaches the intersection zone in the current time step.

Also, $\mathcal{N}_0 = \{1, 2, \ldots, N0\}$ is the set of arriving vehicles at the intersection during the past time steps that are still in the entrance zone. The total number of vehicles inside the entrance zone is $N = N1 + N0$, since only vehicles in $\mathcal{N}_1$ are optimised in the current time step, The reserved time for each conflict point for vehicles in $\mathcal{N}_0$ from the previous time step is used as a new constraint for the following time step. This assumption will add the following constraint to the problem.

$$OT_i + D_i + \tau_{mn} \geq max[(OT_j + D_j + \tau_{mn}), (OT_k + D_k + \tau_{mn}))] \tag{9}$$

For all $i \in \mathcal{N}_1$ and $j, k \in \mathcal{N}_0$, τ_{mn} is an arbitrary conflict point. The algorithm defines an optimization problem to minimize the system cost, J at each time step by considering the current states of the vehicle as described in equation Equation (1) and constraints as defined in Equations (6) and (7).

4 EVALUATION

In this section, we describe our simulation environment and experiments.

4.1 EVALUATION ENVIRONMENT

To evaluate and compare control methods, a realistic simulation program based on Simulation of Urban Mobility (SUMO) [8] has been developed. SUMO is an open source, highly portable, microscopic and continuous traffic simulation that gives the user control over all aspects of the network, such as vehicle type, driver behaviour, intersection control, and statistical data collection. In our work, we have modified SUMO by allowing each vehicle's speed to be manipulated by a central controller (i.e. the ICU) instead of using their default microscopic flow algorithms in SUMO.

4.2 EXPERIMENT

In this work, we consider a four-way intersection with two lanes in each way. Each lane is 3.5m wide with a maximum speed limit of 20m/s, i.e about 70 km/h. We assume that the intersection area can be modelled as a circle with

Table I: Simulation Parameters

MPC	
v_d	16 m/s
u_{max}	5 m/s^2
u_{min}	-6 m/s^2
v_{max}	23 m/s
v_{min}	3 m/s
T	12 s
R_{min}	7 m
d_{min}	7 m

Delay minimization	
V_{max}	16 m/s
u_{max}	0 m/s^2
u_{min}	-6 m/s^2
H_{min}	2 s
$\Delta\tau$	4 s

radius 150m. Table I summarize the simulation parameters and specifications that we used for each algorithm.

The two intersection control algorithms were evaluated for different traffic flow rates. Traffic flow rate is defined as the rate at which vehicles pass a given point on the roadway, and it is normally given in terms of vehicles per hour. Based on the collected data from drivers using a navigation service in China [13] and , the the deployed intersections for a transportation project Michigan [14], we can divide the flow rate range in three different volumes. The peak hour flow rate for a typical intersection in an urban area is between 450 vehicles/hour and 600 vehicles/hour, and we define this as a *high volume traffic*. Further, a traffic rate between 150 vehicles/hour and 450 vehicles/hour is defined as a *medium volume traffic* in an urban area. Finally, a traffic rate of less than 150 vehicles/hour is defined as a *low volume traffic* in an urban area.

The maximum possible rate of vehicles crossing the counter point, that is the maximum capacity of the intersection, is defined as the *saturation flow rate*. In saturated intersection all vehicles move one after each other with minimum safe distance. In this paper, we assume that each vehicle has an average length of 4 meters. A safe gap distance of 2.5 meters is required between a pair of vehicles. Therefore, the maximum possible number of vehicles inside the intersection area is 92 vehicles that will be reached at saturation flow rate.

In this paper, we use the following performance metrics when we evaluate the two intersection control algorithms:

- Average speed of vehicles inside the intersection zone.
- Average number of vehicles inside the intersection zone.
- Collision probability.

- Execution time.

Also, all results will be compared with a standard signalised (traffic light) with 90 second green phase and 90 second red phase intersection management method.

The first two performance metrics are the most used metrics in the literature when evaluating the performance of an intersection control algorithm. Therefore, when considering the average speed and the average number of vehicles in the intersection, we expect both algorithms to improve the performance of the system when compared with the signalised method.

However, the objective of our work is to find intersection control algorithms that can be implemented in real world systems. Therefore, the third performance metric, collision probability, will be crucial, since this is the metric that checks the safety condition.

In addition, we expect that the number of objective function calculations and non-linear constraints for the MPC is higher than for the Delay minimisation algorithm, which can result in a problem for real applications due to the required processing times. Therefore, we decided to also show the resulting execution times for the algorithms as a comparison metric.

5 RESULTS AND DISCUSSION

We performed simulations to compare the two algorithms and evaluate the feasibility to deploy them in reality.

5.1 AVERAGE SPEEDS AND AVERAGE NUMBER OF VEHICLES

Figure 2 illustrates the average speed of each vehicle in one time slot for different flow rates. In the MPC algorithm, the acceleration can have either a positive or negative value. In the Delay minimization approach, the acceleration is limited to negative values, since they are always related to the maximum speed. Therefore, as expected, the system average speed for the MPC algorithm is much higher compared to the Delay minimization algorithm and the traditional signalized method.

Figure 3 shows the average number of vehicles in the intersection area for different flow rates. The signalized intersection will be saturated at a flow rate of about 700 vehicles/hour per lane, which sets an upper limit of the capacity. It is apparent from Figure 3 that for both the MPC algorithm and and the Delay minimization algorithm, there are much less vehicles inside the intersection compared to the signalized intersection. This shows that both algorithms have the potential to increase the capacity of the intersection. However, the simulations showed that when the number of vehicles in the

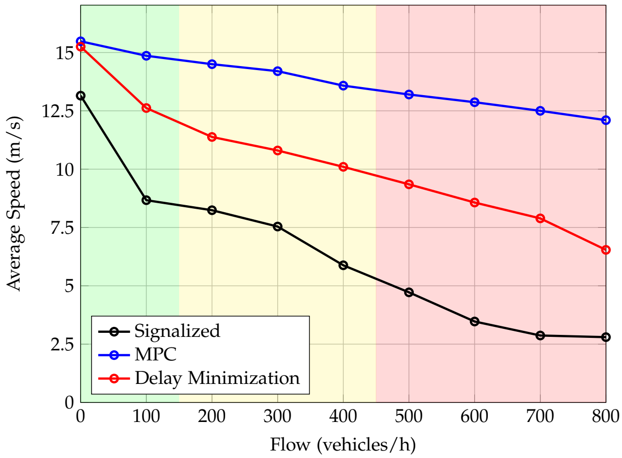

Figure 2: Average speeds for different flow rates. Flow rates are divided in three different volumes.

entrance zone increases to more than 20 vehicles at each time step, the solver may not find a feasible solution to control the vehicles' speeds in proper time. Therefore, we can not reach the maximum claimed capacity of 1600 vehicles/hour in [6].

5.2 COLLISION PROBABILITIES

Our simulation results validated the performance of the two employed methods with respect to traditional signalised approach as reported in [6,7]. The intersection capacity and traveling times (vehicle speeds) are improved with both the MPC algorithm and the Delay minimisation algorithm.

However, in order to deploy the two proposed methods in the real world, the traffic safety must be evaluated as well, since this will be crucial for operational systems.

Figure 4 shows the expected collision probabilities for different traffic flows. A signalized intersection is assumed to have zero probability of collisions for all flows, since this is the main reason for deploying traffic lights in intersections.

In the Delay minimization algorithm, the controller avoids collision between two vehicles by reserving two different time slot for crossing the intersection. However, even if this control strategy avoids collision inside the critical zone, it can not guarantee that no collisions occur in the entrance zone.

Figure 4 shows that there is a probability of 0.2% of collisions for a flow rate of 600 vehicles/hour. This means that 1.2 collisions per hour can be

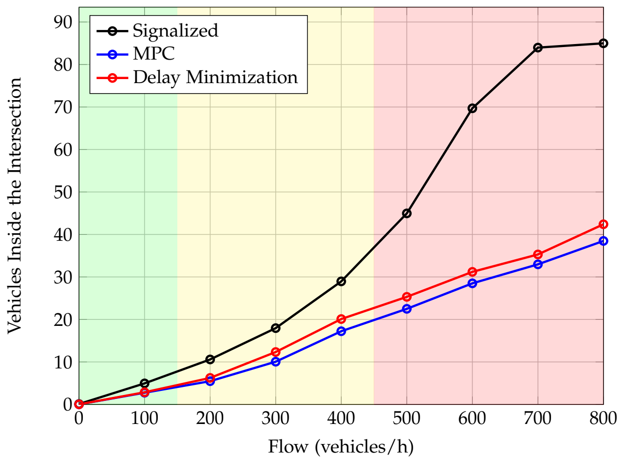

Figure 3: Average number of vehicles inside the intersection zone that is defined as a circle with radius 150m.

expected when using the Delay minimization algorithm, which of course is not an acceptable traffic safety condition.

On the other hand, the MPC algorithm prevents collisions in the whole intersection area over the given time horizon (12 sec). However, a problem occurs when the arriving flow rate increases to more than 500 vehicles/hour. At this flow rate, it can be expected that new vehicles will enter the intersection zone during the problem prediction time horizon. In the algorithm, the number of vehicles inside the entrance zone is assumed to be constant during the horizon time, and therefore, this dynamic may cause collision in intersection.

Figure 4 shows that the collision probability increases exponentially with the flow rate for the MPC algorithm, which of course is not an acceptable traffic safety condition.

5.3 EXECUTION TIMES

Another requirement for operational systems will be that an intersection control algorithm must have a low execution time in order to fulfill the extreme real-time properties required for these types of systems.

Figure 5 shows the execution times for the two algorithms with 95% confidence intervals. We used a desktop equipped with an Intel Core i7-4790K CPU @4 GHz and DDR3 RAM @1600 MT/s. The computer was configured with an Ubuntu Linux. We set the simulation step time in SUMO to 0.5 seconds.

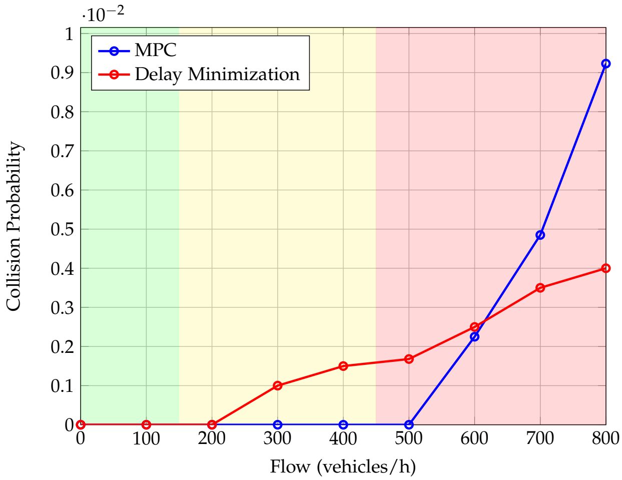

Figure 4: Collision probability

The MPC algorithm is a nonlinear optimisation approaches and it needs to predict the system state in the given prediction time. We expected a higher execution time for the MPC algorithm compared with the Delay minimization algorithm. As can be seen in Figure 5, this difference is definitely noticeable and for higher flow rates the execution time for the MPC algorithm is longer than the simulation step time, which would mean that the calculations for one time step will not be completed before the next time step begins. Therefore, the controller may not be able to make a control decision during a time step. When the number of vehicles and consequently the size of the problem increases, the decision making process time increases dramatically, and it will require more powerful computers to find the solution in proper time.

6 CONCLUSION

The objective of this paper is to evaluate two previously proposed algorithms for an autonomous intersection management [6, 7] in a realistic simulation environment, with the ultimate goal to develop control algorithms for autonomous vehicles that can be deployed in operational systems. It is observed that using these schemes improve the performance of the traditional signalized intersection. However, our simulation shows that the safety conditions are not satisfied in high traffic densities and only can be used for low traffic rate. since the collision probability rather quickly becomes larger than zero

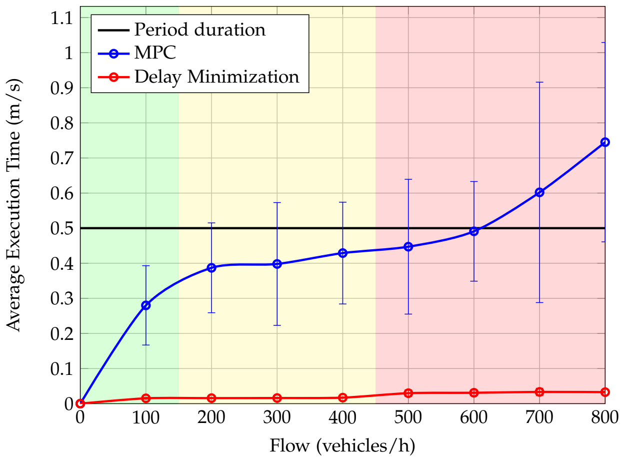

Figure 5: Execution Time

when the traffic increases. Also, the execution times of the algorithms in [6] makes the algorithm infeasible for realistic traffic scenarios.

7 FUTURE WORK

The design of real time intersection management systems is a complex task that involves many different steps. Full understanding of all different parts of the design procedure require deep knowledge of theory. This paper has just briefly described the principles of different traffic management methods. In the future, the goal is to design an efficient and scalable control method for managing vehicles at intersections.

References

[1] K. Dresner and P. Stone, "A multiagent approach to autonomous intersection management," *Journal of artificial intelligence research*, vol. 31, pp. 591–656, 2008.

[2] A. Colombo and D. Del Vecchio, "Least restrictive supervisors for intersection collision avoidance: A scheduling approach," *IEEE Transactions on Automatic Control*, vol. 60, no. 6, pp. 1515–1527, 2014.

[3] R. Hult, G. R. Campos, P. Falcone, and H. Wymeersch, "An approximate solution to the optimal coordination problem for autonomous vehicles at intersections," in *2015 American Control Conference (ACC)*. IEEE, 2015, pp. 763–768.

[4] N. Murgovski, G. R. de Campos, and J. Sjöberg, "Convex modeling of conflict resolution at traffic intersections," in *2015 54th IEEE conference on decision and control (CDC)*. IEEE, 2015, pp. 4708–4713.

[5] E. Steinmetz, R. Hult, Z. Zou, R. Emardson, F. Brännström, P. Falcone, and H. Wymeersch, "Collision-aware communication for intersection management of automated vehicles," *IEEE access*, vol. 6, pp. 77 359–77 371, 2018.

[6] M. A. S. Kamal, J.-i. Imura, T. Hayakawa, A. Ohata, and K. Aihara, "A vehicle-intersection coordination scheme for smooth flows of traffic without using traffic lights," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 3, pp. 1136–1147, 2014.

[7] I. H. Zohdy and H. A. Rakha, "Intersection management via vehicle connectivity: The intersection cooperative adaptive cruise control system concept," *Journal of Intelligent Transportation Systems*, vol. 20, no. 1, pp. 17–32, 2016.

[8] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner, "Microscopic traffic simulation using sumo," in *The 21st IEEE International Conference on Intelligent Transportation Systems*. IEEE, 2018. [Online]. Available: <https://elib.dlr.de/124092/

[9] A. Festag, "Cooperative intelligent transport systems standards in europe," *IEEE communications magazine*, vol. 52, no. 12, pp. 166–172, 2014.

[10] R. Hult, M. Zanon, S. Gros, and P. Falcone, "Optimal coordination of automated vehicles at intersections: Theory and experiments," *IEEE Transactions on Control Systems Technology*, vol. 27, no. 6, pp. 2510–2525, 2018.

[11] M. V. Kothare, V. Balakrishnan, and M. Morari, "Robust constrained model predictive control using linear matrix inequalities," *Automatica*, vol. 32, no. 10, pp. 1361–1379, 1996.

[12] L. Grüne and J. Pannek, "Nonlinear model predictive control," in *Nonlinear Model Predictive Control*. Springer, 2017, pp. 45–69.

[13] J. Zheng and H. X. Liu, "Estimating traffic volumes for signalized intersections using connected vehicle data," *Transportation Research Part C: Emerging Technologies*, vol. 79, pp. 347–362, 2017.

[14] K. Gay, V. Kniss *et al.*, "Safety pilot model deployment: lessons learned and recommendations for future connected vehicle activities." United States. Department of Transportation. Intelligent Transportation, Tech. Rep., 2015.

Paper II

Paper II

Reproduced, with permission, from

SeyedeZahra Chamideh and William Tärneberg , and Maria Kihl, "Evaluation of Decentralized Algorithms for Coordination of Autonomous Vehicles at Intersections," *24th IEEE International Conference on Intelligent Transportation - ITSC2021*, 2021.

II

Evaluation of Decentralized Algorithms for Coordination of Autonomous Vehicles at Intersections

Connected Autonomous Vehicles (**AV**)s with Vehicle-to-Vehicle (**V2V**) communication are becoming an essential component of the transportation system. Self-driving cars have the potential to optimize the roads' traffic flow, fuel consumption and remove the possibility of human error and distractions. In these systems, all involved vehicles must be fully autonomous for maximum gain. However, a fully automated system requires major updates in the transportation and network infrastructure. In this paper, we investigate intelligent traffic control mechanisms for autonomous vehicles at intersections as a replacement of traditional intersection control (i.e traffic lights). Two well-cited decentralized optimization algorithms for cooperative vehicles are compared with realistic simulations in SUMO. We investigate the safety and feasibility of deploying the proposed algorithms in the real world. Further, we study the scalability and performance of the algorithms in the presence of communication impairments associated with wireless channels. This side-by-side comparison helps to gain insight into the strengths and limitations of these types of algorithms.

1 INTRODUCTION

Intelligent Transportation Systems and in particular Autonomous Vehicle (**AV**) combined with cooperative strategies will likely have a significant effect on future traffic management systems. Currently, road intersections are controlled by traffic lights. However, these systems are burdened with several fundamental problems. For example, human recognition errors or vehicles unnecessarily braking can result in many accidents and significantly increase travel times [1, 2]. Therefore, there is a rising concern on the efficiency and safety of traditional intersection management methods. Recent advances in embedded sensors, on-board computing combining with communication technologies [3, 4], have enabled the emergence of Autonomous Intersection Management (**AIM**) strategies that can remove the need for traffic lights.

Papers proposing traffic control strategies and coordination algorithms in the context of **AV**, usually formulates the problem as an optimization problem. The main objective is to find the best trajectories and vehicles' passing sequences through negotiation and cooperation between road users and infrastructure, by considering system dynamics and collision avoidance constraints. However, imperfect system models for vehicles' dynamic and non-ideal wireless communication channel lead to uncertainties that make the design of a perfect control algorithm challenging.

Cooperative intersection management can be classified into three main categories: centralized, distributed and hybrid [5]. Centralized methods rely on a central intersection manager that gives instructions to vehicles based on the collected information from wireless communications. In distributed methods, decisions are made locally by each vehicle based on the information achieved by negotiation between vehicles and observation of the environment through local sensors. In hybrid methods, vehicles are allowed to communicate with each other and with a centralized intersection manager in order to have a more efficient intersection management.

In [6] and [7], centralized optimal intersection controllers for autonomous vehicles were proposed. Centralized approaches have been shown to work well in numerical simulations and theoretical analysis. However, our investigation in [8] shows that centralized methods are poorly scalable when the vehicle density increases and finding an exact solution becomes intractable for realistic vehicles densities. Therefore, these algorithms are practically limited to intersections with low vehicle densities, for example rural areas or city intersections during night.

In order to solve the complexity and scalability problems of the centralized approaches, distributed methods can be introduced. Here, each vehicle inside the intersection area solves a problem that is much smaller and easier to solve than in centralized approaches. In [9, 10], a distributed algorithm is proposed where vehicles sequentially solve a local optimization problem to avoid collisions. The solution is based on a receding horizon formulation with a predefined decision order. Authors complement their previous work [9], and combined the proposed optimal controller with a sequential decision making in [11]. Each vehicle's decision depends on already decided and available solutions from the preceding vehicles.

A non-linear **MPC** approach was proposed in [12]. At each optimization step, each vehicle has to find an optimal control input based on the predicted trajectories of vehicles in the intersection with the objective to minimize travel times. In [13], a linear non-convex distributed **MPC** approach was proposed, where all coordinated vehicles can solve the problem simultaneously.

The overall aim of our research is to develop robust and safe **AIM** systems for cooperative vehicles. Obviously, an fully autonomous intersection control system must guarantee total safety for passengers, that is, no collisions can occur, which means that the collision probability must be zero for all possible scenarios. Further, the computational requirement of an algorithm must be limited, in order to be feasible to run in real-time. Also, the control system must be robust to communication uncertainties, since the wireless communication will never be 100% perfect in a vehicular scenario.

In this paper, we compare the previously mentioned decentralized intersection control algorithms [11, 13] by implementing them in the realistic simulation environment **SUMO** [14]. We have chosen these two algorithms, since they have different optimization approaches for the same type of intersection control, and they are well-cited in the literature. Our investigation shows that both algorithms have the potential to improve traditional signalized methods for commonly used performance metrics such as *average traveling speed*, *fuel consumption* and *intersection throughput*.

However, in order for an **AIM** algorithm to be deployed in the real world, where vehicles cannot be allowed to collide, performance metrics as *collision probability*, *scalability*, and *robustness* will be much more important performance metrics. For these performance metrics, both algorithms perform well only for low traffic densities. When the traffic density increase, none of the algorithms will be able to keep a collision probability of zero. Also, none of the algorithms are robust to packet losses, since they both assume that all information needed for the optimization is delivered with 100% reliability. Our main conclusion is that these algorithms only work very well for low traffic densities and in order to use them in high traffic intensity they need to be modified and combined with centralized methods.

2 TARGETED SYSTEM

Intersections are the most common cause of traffic congestion in urban transportation networks [15]. Congestion wastes time, fuel and creates more uncertainty for travelers [16].

In this papers, we consider the problem of Autonomous Vehicle coordination at a crossroad intersection without traffic lights. Coordinated intersection traffic management enables a vehicle to communicate with roadside equipment or other vehicles, and thereby improves the road traffic safety and efficiency. The system is composed of vehicles equipped with On Board Unit (**OBU**), which may employ a wide range of sensors. The **OBU**s are connected to computing nodes and receive messages from other entities via wireless communication links.

The fully automated vehicles cooperate by exchanging information through 5G enabled Cellular Vehicle-to-Everything (**C-V2X**) technology where vehicle to everything (V2X) communication is enabled with both cellular network based links through the infrastructure and direct communication links between vehicles and other entities over sidelink (PC5) interfaces [17]. For 5G, 3GPP Release 16/17 has specified a reliability of 90-99.999% with a maximum end-to-end latency of 3-100 ms, when considering a data rate of 10-50 Mbps broadcast messages between vehicles within a communication range of about 360-700 meters for advance driving scenarios [18].

In our scenario, each vehicle can transmit a message informing about its characteristics, position and movement to all its neighbors within a defined range through direct inter-vehicle V2V communication links. In order to have a desired trajectory for vehicles, the longitudinal velocity of each vehicle is defined as the system output, which should be controlled by the acceleration at each time step. We consider a limited speed and acceleration, and, since the vehicles are not allowed to make U-turns at the intersection area, the minimum speed is always positive along their path. The maximum allowed speed on the road is defined by v_{max}. In addition, the maximum acceleration and deceleration are described by u_{min} and u_{max}, respectively. However, maximum and minimum acceleration of each vehicle depends on its current velocity and speed limitation.

3 INVESTIGATED ALGORITHMS

In this paper, we investigate the performance of two decentralized **AIM** algorithms. The first algorithm is the ***MPC*** algorithm [13], which be called the ***MPC*** *algorithm* in the rest of the paper. In this algorithm, all coordinated vehicles solve a local optimization problem in parallel by predicting the system model in a given finite prediction horizon. The second algorithm is the *Pure Sequential* algorithm [11]. This algorithm defines a linear fast converging optimization problem for each vehicle to solve sequentially. In the **MPC** algorithm, the objective for each vehicle is to maintain a minimum distance to the vehicle ahead. In the Pure Sequential algorithm, the goal for each vehicle is to reserve a safe time slot to cross the intersection.

We surveyed the advantages and disadvantages of the algorithms by implementing them in the realistic **SUMO** [14]. In this section, we first describe our general system model and then present high level descriptions of the two algorithms.

3.1 SYSTEM MODEL

In this section we present the system model that is used by the **AIM** algorithms. $\mathcal{N}_t = \{1, 2, \ldots, N_t\}$ **AV**s exist in a coordination area of an intersection at time t. Each vehicle $i \in \mathcal{N}_t$, has a predetermined path γ_i that is perfectly followed. $p_i(t)$ is the position (the distance from the center of the intersection) of vehicle i along its paths at time t. Similarly, $v_i(t) = \dot{p}_i(t)$ and $u_i(t) = \ddot{p}_i(t)$ denote the velocity and acceleration of vehicle i.

The longitudinal motion of each vehicle, $x_i = [p_i, v_i]^\intercal \in \mathcal{X}_i$, is defined as the system state and can be controlled by its acceleration. We assume that the control input is updated in discrete time τ. The discrete time state model of vehicle i is given in Equation (1).

$$x_{i,t+1} = \begin{bmatrix} 1 & -\tau \\ 0 & 1 \end{bmatrix} x_{i,t} + \begin{bmatrix} -\frac{1}{2}\tau^2 \\ \tau \end{bmatrix} u_{i,t} \tag{1}$$

The state space model, Equation (1)), represents the relation between the acceleration as the input of the system and the longitudinal motion of vehicle i. As described in Section 2, a limited speed and acceleration is considered for the system states, These above-mentioned limitation result in the following inequality constraints on the input and the states of the system:

$$\begin{aligned} &u_{min} \leq u_{i,t} \leq u_{max} && \forall t \geq 0 \\ &0 \leq v_{min} \leq v_{i,t} \leq v_{max} && \forall t \geq 0 \end{aligned} \tag{2}$$

3.2 DATA DISSEMINATION

Information exchange between the involved vehicles is crucial to solve the optimization problems, regardless of how the control problem is modeled. Communication between vehicles will be affected by the impairments associated with the wireless channels, which will lead to packet drops and/or random latency in packet arrivals. In this section we will describe how information is disseminated between vehicles.

In the **MPC** approach every vehicle $i \in \mathcal{N}_t$ solves its local optimization problem at time t. The optimization problem results, that is, the vehicle characteristics, position and movement, are immediately broadcast to all its neighbors within its radio range at time $t+1$. Therefore, in each time instance the total number of N_t messages are required to be exchanged between involved vehicles.

In the Pure Sequential approach, vehicles need to sequentially solve the local optimization problems. Initially, vehicles cooperatively agree on a decision order, which enables a sequential decision-making procedure. Each vehicle solves a local problem and then the expected occupancy times are broadcast over wireless communication links to the remaining vehicles. Therefore, the total number of $(N\tau + 1)N_t$ messages are required to be exchanged between the vehicles, where N is the rate of vehicle arrivals per second.

3.3 MODEL PREDICTIVE CONTROL ALGORITHM

In this section, we give a high-level description of the ***MPC*** algorithm proposed in [13].

The **MPC**s methods are generally used to represent the behavior of complex dynamic systems. An **MPC** algorithm uses the current measurements, dynamic system model and the system limitations to predict future changes and allows the current time slot to be optimized. The **MPC** is an iterative optimization model that in each time slot, compute a cost minimizing control strategy for a time horizon in the future. Only the first step of the calculated control strategy is implemented and the calculation will be repeated for the next time slots [19].

In order to use the **MPC** for our dynamic system represented in Equation (1) and then calculate the optimal control strategy, $u_{i,t}$, we need to define the system cost, which is a function of system's states $x_i = [p_{i,t}, v_{i,t}]$, and the control input $u_{i,t}$ based on the algorithm objectives. The paper [13] defines three main objectives for the intersection control algorithm along with safety constraints. To avoid any rear-end collisions and that every vehicle $i \in \mathcal{N}_t$ passes the intersection in a safe way, a minimum separation distance between two vehicles on the same lane, d_{min}, is defined. The following constraint in Equation (3) is defined to prevent rear-ends collisions between vehicle i and its leading vehicle $j \in \mathcal{L}_{i,t}$ where $\mathcal{L}_{i,t}$ is the set of vehicle's i leading vehicles.

$$p_{i,t} - p_{j,t} \geq d_{min} \quad j \in \mathcal{L}_{i,t}, \forall t \geq 0 \tag{3}$$

To ensure that no collisions occur between vehicle i and vehicle j from different approaching lanes inside the critical zone, i.e a side collision, a linear inequality constraint is defined as in Equation (4) below.

$$|p_{i,t} - p_{j,t}| \geq R_{min} \quad p_{i,t} \wedge p_{j,t} \geq 0, \gamma_j \in \Gamma_i, \forall t \geq 0 \tag{4}$$

Where Γ_i is a set of all paths that have the potential to collide with vehicle i from path γ_i. R_{min} is a constant that denotes the minimum separation distance between the centers of two vehicles. Equation (4) is a non-convex inequality, which results in a non-convex optimization problem.

Further,a problem cost function needs to be defined. The first objective is that the vehicles should cross the intersection with almost constant and high speed, which is called the desired speed (v_d). The second objective is to achieve a minimum fuel consumption, which translates into minimizing the absolute accelerations and acceleration rate. In addition, a smooth flow of vehicles, and thereby, a smooth change of acceleration is desired. Therefore, the system cost for vehicle $i \in \mathcal{N}_t$, J_i is defined as in Equation (5) where $v_{i,t}$ and v_d^i are the **MPC** algorithm control and reference variables, respectively, and $u_{i,t}$ is the manipulated variable.

$$\begin{aligned} J_i = &\sum_{t=0}^{T} (w_{v_i}(v_{i,t+1} - v_d^i)^2 + w_{u_i}(u_{i,t})^2) + \\ &\sum_{t=0}^{T-1} w_u(u_{i,t+1} - u_{i,t})^2 \end{aligned} \tag{5}$$

Here, w_{v_i} ,w_{u_i} and w_u are weight coefficients and J_i is the problem cost for vehicle i to be minimized, subject to the given current states of the vehicles as defined in Equation (1) and constraints as defined in Equations (2) to (4). Solving the optimization problem with these safety constraints is a challenging task due to its non-convex nature. To cope with non-convex limitation a semi-definite programming relaxation is applied in the paper, which is achieved by introducing a priority scheme on vehicles, such that a vehicle with lower priority must give right of way in case of a potential conflict.

3.4 PURE SEQUENTIAL ALGORITHM

In this section, we give a high-level description of the Pure Sequential algorithm proposed in [11]. This collision avoidance solution relies on the design of a controller that prevents the system of reaching a given configuration. The paper [11] defines, for each vehicle i, the critical set C_i as the set of all displacements along its path leading to a potential collision. Thus, C_i can be defined as:

$$C_i \triangleq \{x_{i,t} \in \mathcal{X}_i | p_{i,t} \in [L, H]\} \tag{6}$$

Where $[L, H]$ describes the junction boundary. Therefore, the set of all conflicting configurations that result in collision at intersection can be represented as in Equation (7).

$$\mathcal{S} = \{x \in R^n : \exists (i, j) \in \mathcal{N}_t, (x_{i,t} \in C_i \wedge x_{j,t} \in C_j)\} \tag{7}$$

Safety is ensured if only one vehicle, i, is allowed to cross the critical area at time t. To avoid collision, vehicle $i \in \mathcal{N}_t$ needs to reserve a time slot to cross the critical zone. Therefore, the safe state set for vehicle i can be defined as follow:

$$C_i^{safe} = \{x_{i,t} \in \mathcal{X}_i | p_{i,t} \in [L, H], t \in \mathcal{K}_i\} \tag{8}$$

Where $\mathcal{K}_i$ is a unique time slot, $\mathcal{K}_i \neq \mathcal{K}_j$, when only vehicle i is allowed to cross the intersection. The objective of each vehicle i is to find the $\mathcal{K}_i$. For this aim, the already reserved time slots need to be known as a constraint, in order to solve the local control problem. Therefore, it is necessary to define a decision order set for the vehicles in $\mathcal{N}_t$. The decision order $\mathcal{O}$ is a permutation of the indices in $\mathcal{N}_t$, where vehicles will solve the optimization problem sequentially based on this decision order set. Let $\mathcal{O}_i^b$ and $\mathcal{O}_i^a$ be the sets having the indices of all vehicles $j \neq i$ appearing before and after i in $\mathcal{O}$ respectively. For simplicity, in this paper, a first-come-first-served protocol is used for the decision order policy. This means that if vehicle j arrives earlier than vehicle$i$, then $j \in \mathcal{O}_b^i$.

Vehicle i in the decision order will solve the two sub-problem explained as follows.

Problem A: Finding the optimal control policy such that vehicle i enters the intersection only after all preceding vehicle(s) $j \in \mathcal{O}_i^b$ have crossed the intersection.

Problem B: Finding the optimal control policy such that vehicle i exits the intersection only before any preceding vehicle(s) $j \in \mathcal{O}_i^b$ enters the intersection.

For each vehicle $i \in \mathcal{N}_t$, the expected occupancy time of the intersection at time t can be expressed as Equation (9), and the sum of the occupancy times of all preceding vehicles of vehicle i is shown in equation Equation (9a)

$$\mathcal{I}_i = \{k \in R : x_{i,t+k} \in C_i\} \tag{9}$$

$$\mathcal{T}_i = \sum_{j \in \mathcal{O}_b^i} \mathcal{I}_j \tag{9a}$$

Therefore, for vehicle i in *Problem A*, the earliest intersection entry time is $T_{max} = \max\{\mathcal{T}_i\}$ and the latest intersection exit time in *Problem B* is $T_{min} = \min\{\mathcal{T}_i\}$. The Problems A and B can then be formulated as the following constrained linear quadratic regulator (LQR) programs in Equation (10).

$$\min_{u_{i,t}} \quad J(x_{i,t}, u_{i,t}) \tag{10}$$

$$s.t \quad (1), (2) \tag{10a}$$

$$p_{i,T_{max}} < L \quad \text{If Problem A} \tag{10b}$$

$$p_{i,T_{min}} < H \quad \text{If Problem B} \tag{10c}$$

The constraints Equations (10b) and (10c) are sufficient to ensure that vehicle i is outside the critical area during $\mathcal{T}_i$. Both problems A and B are the combination of two optimization sub-problems. First, a finite time solution optimization problem defines a collision free trajectory up to time T_{max} (T_{min} for problem B). Second, an infinite optimization problem defines the trajectory for all times after T_{max} (T_{min} for problem B).

The algorithm considers the same cost function for all vehicles and given as Equation (11) for finite sum-problem A:

$$J_i = \sum_{t=0}^{T_{max}} w_{v_i}(v_{i,t+1} - v_d^i)^2 + w_{u_i}(u_{i,t})^2 \tag{11}$$

The same holds for the cost function of the finite sub-problem for B, if T_{max} is replaced by T_{min} in Equation (11). For the second sub-problem, it is assumed that the only objective is to minimize the deviation of the vehicle's speed from the desired value. Therefore, the infinite optimization problem can be defined as $\sum_{t=0}^{T}(v_{i,t+1} - v_d^i)^2$ subject to Equations (1) and (2) where T is an arbitrary time step.

4 EVALUATION

In this section, we describe our simulation environment and experiments with the objective to evaluate the performance of the proposed decentralized algorithms, and investigate the safety, scalability and feasibility of implementing them in the real world.

4.1 EVALUATION ENVIRONMENT

To evaluate and compare the control methods, a realistic simulation program based on **SUMO** [14] has been developed. SUMO is an open source, highly portable, microscopic traffic simulator that gives the user control over all aspects of the network. In our work, we have modified SUMO by allowing each vehicle's movement to be controlled by proposed algorithm control strategy instead of using the default microscopic flow algorithms.

In our simulation all vehicles have the same physical properties and they arrive to the intersection according to a pre-generated traffic demand stored in a route XML-file. In route file the vehicles' arrivals are randomized based on a Poisson distribution and the probability that trips will start/end at the different entrance/exit lanes are the same.

4.2 EXPERIMENTS

In this work, we consider a four-way intersection with two lanes in each way. Each lane is 3.5m wide with a maximum speed limit of 20m/s, i.e., about 70 km/h. We assume that the vehicles enter the intersection with an arbitrary initial speed close to desired speed. In addition, the intersection area is modelled as a circle with radius 150m. Table I summarizes the simulation parameters and specifications that we used in our simulation program.

The scalability and the feasibility of the control algorithms were evaluated for different traffic flow rates, i.e. the rate at which vehicles pass a given point on the roadway.

4.3 EVALUATION METRICS

In this paper, we use the following performance metrics when we evaluate the algorithms:

- **Average speed** of vehicles. An algorithm is well performed in term of speed when all vehicles travel at similar velocity and the relative movement is smooth.
- **Fuel consumption** is a form of thermal efficiency, and an algorithm with lower fuel consumption is desired.
- **Throughput** An intersection can be modeled as a queuing system where the maximum throughput of an intersection can be calculated as either the maximum number of vehicles that can coexist in an intersection, or the minimum time a vehicle spends in the intersection.
- **Safety**: **AIM** methods usually perform well for low traffic density. However, for high flow rates, a feasible solution with safety constraints, may be harder to find. An infeasible solution may lead to car crashes. Therefore, the expected collision probability increases when the algorithm can not find a solution for the optimisation problem within a sampling period.
- **Scalability**: **AIM** methods must be able to handle a large number of vehicle movements. Therefore, it is important to evaluate the algorithms'

Table I: Simulation Parameters

General Parameters		
v_d^i	16 m/s	Targeted Speed
u_{max}	5 m/s^2	Maximum acceleration
u_{min}	-6 m/s^2	Maximum deceleration
v_{max}	20 m/s	Maximum speed
v_{min}	0 m/s	Minimum speed
MPC		
T	10 s	Prediction horizon
R_{min}	7 m	Minimum separation distance between vehicles from different approaching lane
d_{min}	7 m	Minimum separation distance between vehicles in the same lane
Pure Sequential		
L	7 m	Junction lower bound
H	-7 m	Junction upper bound

complexity, and problem size, when the traffic flow rate increases. In this paper the maximum scalability of an **AIM** algorithm translates to a minimum increase in the required iterations to solve the optimization problem when the flow rate increases.

- **Robustness**: An **AIM** method is robust to the limitations set by the wireless channel such as packet loss and latency, when these limitations have a minimum impact on the algorithm's performance. The maximum robustness is calculated as the maximum probability of a reliable solution in the presence of wireless channel impairments.

The first three performance metrics are the most used metrics in the literature when evaluating the performance of intersection control algorithms. However, the objective of our work is to find intersection control algorithms that can be implemented in the real world. Therefore, we have added three more performance metrics in our investigation, that is collision probability, scalability and robustness. These performance metrics will be crucial when deploying an algorithm in an operational systems.

In addition, all results will be compared with a standard signalized, controlled by traffic light, intersection with 90 second green phase and 90 second red phase.

5 RESULTS AND DISCUSSION

We performed simulations to compare the two algorithms and evaluate them according to the performance metrics described above. All results are compared with the results that would be achieved in an intersection controlled by traffic lights, assuming perfect driving behavior for all vehicles.

5.1 AVERAGE SPEED AND FUEL CONSUMPTION

Figure 1.a illustrates the average speed of each vehicle during one time slot for different flow rates. It is apparent that both the **MPC** and the Pure Sequential perform better compared to an intersection controlled by traffic lights.

Since in the **MPC** algorithm, one objective of the optimization problem is to drive in a smooth and comfortable acceleration, the vehicles controlled by this algorithm avoid high acceleration changes. Therefore, in comparison with the Pure Sequential algorithm, where a given vehicle can reach the maximum speed faster, the **MPC** has a lower average speed.

Several factors can have an effect on the fuel consumption, and it varies with the vehicle type, weather condition, driving behaviors such as rapid acceleration, speed. The Environmental Protection Agency (EPA) study [20] shows that the acceleration rates have a significant effect on a vehicle's fuel consumption. In our experiments, we assumed that all vehicles are of the same type. Therefore, the main factor in fuel efficiency will be the driving behavior.

Figure 1.b shows an estimation of the average fuel consumption for each vehicle during one time slot. Figure 1.b shows that the **MPC** algorithm results in the lowest fuel consumption, and this is due to that the minimum acceleration and the acceleration rate are defined as costs of the optimization problem. As expected, intersections controlled by traffic lights, where many vehicles unnecessarily brake, will have the maximum average fuel consumption for each vehicle.

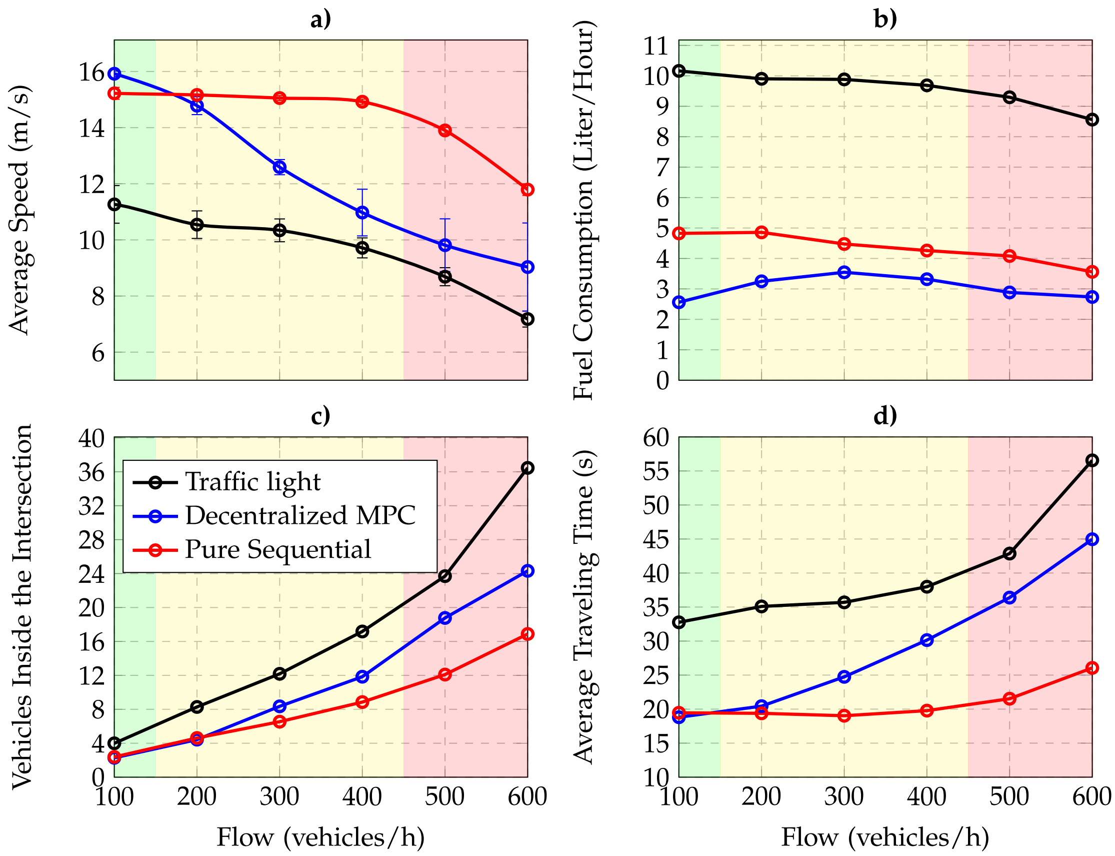

Figure 1: a) The average speed of each vehicle inside the intersection. b) The average fuel consumption for each vehicle. c) The average number of vehicles that coexist inside the intersection. d) The average traveling time for each vehicle when crossing the intersection. The intersection is defined as a circle with radius 150m.

5.2 THROUGHPUT

Figures 1.c and 1.d illustrate the average number of vehicles that coexist in the intersection, and the average time a vehicle spends in the system, respectively. Both algorithms improve the intersection throughput compared with an intersection controlled by traffic lights. As can be seen in Figure 1.d, the average traveling time for vehicles controlled by the Pure Sequential algorithm at flow rate 500 vehicles/hour is about 50% (21 sec) of the traveling time in an intersection controlled by traffic lights. The same holds for the average number of vehicles inside the intersection. It is clear that, for both the **MPC** algorithm and Pure Sequential algorithm, there are much less vehicles inside the intersection, and the vehicles have lower traveling time compared to the signalized intersection. This result shows that the algorithms have the potential to increase the capacity of the intersection.

5.3 SAFETY

In order to evaluate the traffic safety, we have also estimated the collision probabilities for each algorithm and traffic flow rate, as shown in Figure 2. The collision probability is calculated as the expected number of collisions/hour/vehicle. For example, a collision probability of 0.2% for a flow rate of 500 vehicles/hour correspond to, in average, that about one collision per hour can be expected in the intersection. An intersection controlled by traffic light is assumed to have zero probability of collisions for all flows, since this is the main reason for deploying traffic lights in intersections. The Pure Sequential algorithm performs safe for flow rates less than 200 vehicles/hour. However, for arrival rate higher than 300 vehicles/hour the collision probability increases exponentially. In the Pure Sequential only one vehicle is allowed to exist in the critical area at each time instance. For traffic intensity higher than 400 vehicles/hour/lane the arrival rate to critical area is higher than 1600 vehicles/hour (2 sec/vehicle). Therefore, the probability of a feasible solution for optimization problem decreases with increasing flow rate.

In the **MPC** algorithm, several vehicles are allowed to coexist in the critical area at the same time. The algorithm prevents collisions in the whole intersection area over the given time horizon (10 sec) by keeping vehicles at a safe distance to each other. However, a problem occurs when the arriving flow rate increases to more than 400 vehicles/hour. At this flow rate, the vehicles' initial speed may stop the optimizer from finding a feasible solution that keeps a safe distance to all other vehicles and thereby avoids collisions in a specific time slot. The decentralized **MPC** performs almost the same as the centralized **MPC** algorithm [8]. However, by increasing flow rate

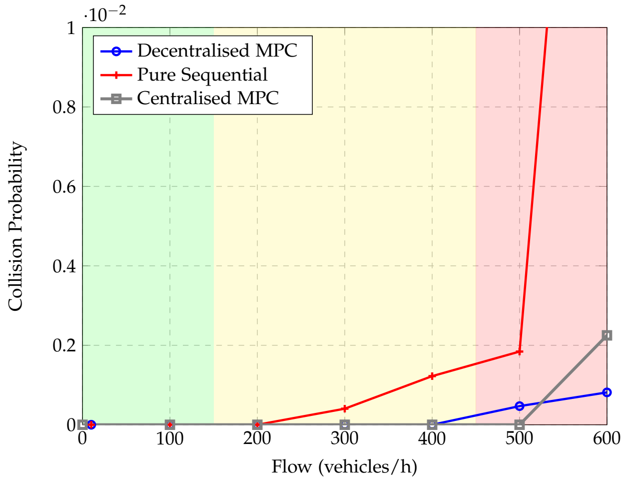

Figure 2: Collision probability in different traffic intensity

and consequently increasing the problem size finding the exact solutions for centralized algorithm become intractable.

5.4 SCALABILITY

Scalability is another crucial requirement for operational systems to fulfill the extreme real-time properties of these types of systems. An intersection control algorithm must be scalable in order to handle a large number of vehicles inside an intersection in a real-world scenario.

To evaluate and compare the algorithms' scalability, the average number of iterations that each optimization solver needs to have in order to find a feasible solution for increasing flow rates, are shown in Figure 3. The **MPC** algorithm needs to predict the system states of all vehicles involved in the intersection for the prediction horizon. Therefore, by increasing the flow rate, thereby the problem size, the number of required iterations increases exponentially. On the other hand, the Pure Sequential is a linear convex optimization problem and the other vehicles' occupancy time is the only constraint imposed from other entities. Therefore, the problem size and the number of required iterations is almost constant when increasing the number of vehicles involved in the control problem.

As it is shown in Figure 3, when the flow rate is 600 vehicles/hour/lane (that is an average number of 24 vehicles within the intersection, see Figure 1.c, the **MPC** algorithm needs on average 32 iterations, while the Pure Sequential algorithm needs only 13 iterations to find a feasible solution.

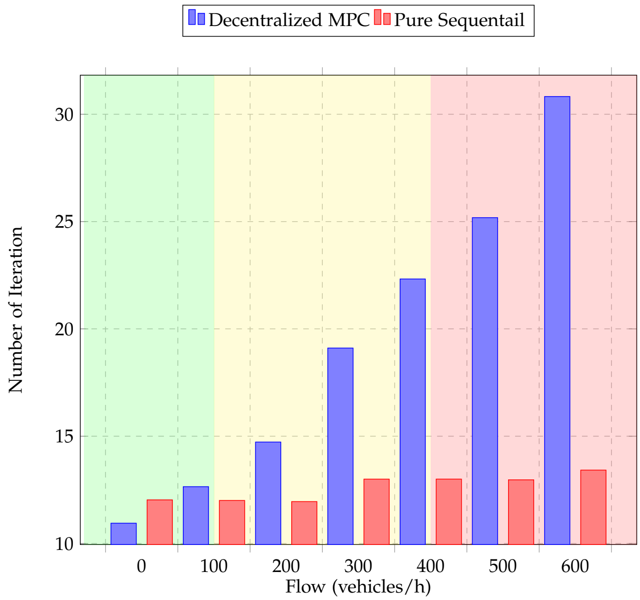

Figure 3: Average required iteration for feasible solution

5.5 ROBUSTNESS

Figure 4.a illustrates the maximum required number of messages that needs to be transmitted at each time instance for each control algorithm. These messages include the occupancy time for Pure Sequential and initial states for **MPC**. In most evaluations of **AIM** algorithms, it is assumed that the limitations set by the wireless communication, that is the packet losses and delay, are negligible. However, this is not a correct assumption for real world scenarios with current technologies. The measurements in [21] for **C-V2X** show that the average end-to-end latency in PC5-based communication is 30ms with almost 2.5% packet loss (i.e. 97.5% reliability).

For the Pure Sequential algorithm, each vehicle will start its local optimization problem when it has collected all message(s) from all preceding vehicle(s) Therefore, if only one message is dropped, the optimization problem cannot be formulated and the algorithm fails.

For the **MPC** algorithm, the vehicles work in parallel, and each vehicle can solve its local problem regardless of the other vehicles' problem formulation. Therefore, packet loss for one vehicle has no effect on the other vehicle(s) local optimization problems. However, an unreliable solution for one vehicle can increase the collision probability of the entire system. Therefore, it is crucial to evaluate how robust the algorithms are to channel impairment.

Figure 4.b shows the probability that each vehicle receives all messages it needs to solve its local optimization problem for a specific time slot. We provide the results when the wireless channel has the reliability of 99.999%, which is the standard for **C-V2X** 5G, Ultra Reliable Low Latency Communication (URLLC), in Release 16, and the reliability of 97.5% that was measured in [21] for existing technologies. The simulation results confirm that both algorithms perform well in an URLLC wireless channel. However, a packet loss rate of 2.5% can reduce the reliability if the algorithms significantly. For instance, in Figure 4.b for a flow rate of 600 vehicles/hour, when deploying the **MPC** algorithms, vehicles are not aware of 30% of the other vehicles. For the Pure Sequential algorithm, there is, for the same flow rate, a probability of 40% that a vehicle will not receive all messages from preceding vehicle(s) and, therefore, the optimization problem will fail in this time step.

In order to have a safe intersection *all* vehicles inside the intersection need to find a reliable solution during each time step. Therefore, Figure 4.c shows the average probability that all vehicle(s) involved in the intersection receive their messages in each time step. From Figure 4.c it is apparent that **MPC** algorithm performs better than the Pure Sequential algorithm for low flow rates. For example, with a flow rate of 300 vehicles/hour, the **MPC** algorithm will have a probability of 62% that all vehicle(s) receive the messages from other vehicles. For the Pure Sequential, this probability is only 41%. However, for high flow rates both algorithms perform poor with regards to packet loss, and the probability of a completely safe intersection is almost zero. This means that the algorithms' performance highly depends on the reliability of the wireless communication channel and the algorithms are, therefore, not robust to packet loss.

6 CONCLUSION

The objective of this paper is to evaluate two previously proposed and well-cited algorithms for autonomous intersection management [13] [11] in a realistic simulation environment.

It is seen that using these schemes improve the performance for low vehicle flow rates compared with intersections controlled with traffic lights. However, our results clearly show that the safety conditions are guaranteed for high traffic densities, and therefore, the algorithms would only be usable for low traffic rate. When increasing the flow rate, the collision probability rather quickly becomes larger than zero. Also, the algorithms are not robust to wireless channel impairments that results in packet loss. So to summarize, new types of Autonomous Intersection Management algorithms are required in order to fulfil the visions of completely autonomous and cooperated vehicles.

7 FUTURE WORK

The design of real time intersection management systems is a complex task that involves many different steps. Full understanding of all distinct parts of the design procedure requires deep knowledge of theory. In this paper and our previous work [8], we briefly describe the principles of different centralized and decentralized traffic management methods. In the future, the goal is to design an efficient, scalable and robust control method for managing vehicles at intersections.

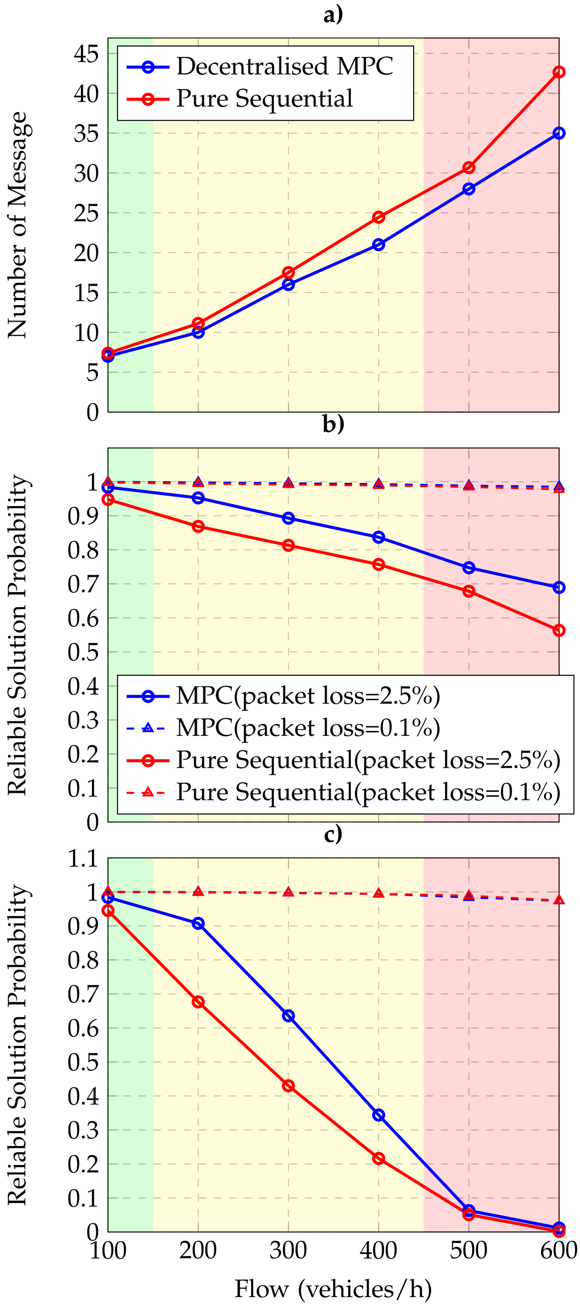

Figure 4: a) Maximum number of required messages in each time step b) The average probability that a vehicle receives the required information from all other vehicle(s) inside the intersection. c) The average probability that all vehicles receive the required information during each time step. All results are shown with packet dropping rates of 0.1% and 2.5%

References

[1] D. Shinar, *Traffic safety and human behavior*. Emerald Group Publishing, 2017.

[2] K. D. Kennedy and J. P. Bliss, “Inattentional blindness in a simulated driving task,” in *proceedings of the human factors and ergonomics society annual meeting*, vol. 57, no. 1. SAGE Publications Sage CA: Los Angeles, CA, 2013, pp. 1899–1903.

[3] S. Chen, J. Hu, Y. Shi, Y. Peng, J. Fang, R. Zhao, and L. Zhao, “Vehicle-to-everything (v2x) services supported by lte-based systems and 5g,” *IEEE Communications Standards Magazine*, vol. 1, no. 2, pp. 70–76, 2017.

[4] H. Wymeersch, G. R. de Campos, P. Falcone, L. Svensson, and E. G. Ström, “Challenges for cooperative its: Improving road safety through the integration of wireless communications, control, and positioning,” in *2015 International Conference on Computing, Networking and Communications (ICNC)*. IEEE, 2015, pp. 573–578.

[5] L. Chen and C. Englund, “Cooperative intersection management: A survey,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 2, pp. 570–586, 2015.

[6] R. Hult, G. R. Campos, P. Falcone, and H. Wymeersch, “An approximate solution to the optimal coordination problem for autonomous vehicles at intersections,” in *2015 American Control Conference (ACC)*. IEEE, 2015, pp. 763–768.

[7] C. Bali and A. Richards, “Merging vehicles at junctions using mixed-integer model predictive control,” in *2018 European Control Conference (ECC)*. IEEE, 2018, pp. 1740–1745.

[8] S. Chamideh, W. Tärneberg, and M. Kihl, "Centralized coordination of autonomous vehicles at intersections," in *28 th International Conference on Software, Telecommunications and Computer Networks (SoftCOM 2020)*. Institute of Electrical and Electronics Engineers Inc., 2020.

[9] G. R. de Campos, P. Falcone, and J. Sjöberg, "Autonomous cooperative driving: a velocity-based negotiation approach for intersection crossing," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. IEEE, 2013, pp. 1456–1461.

[10] G. R. Campos, P. Falcone, H. Wymeersch, R. Hult, and J. Sjöberg, "Cooperative receding horizon conflict resolution at traffic intersections," in *53rd IEEE Conference on Decision and Control*. IEEE, 2014, pp. 2932–2937.

[11] G. R. de Campos, P. Falcone, R. Hult, H. Wymeersch, and J. Sjöberg, "Traffic coordination at road intersections: Autonomous decision-making algorithms using model-based heuristics," *IEEE Intelligent Transportation Systems Magazine*, vol. 9, no. 1, pp. 8–21, 2017.

[12] A. Britzelmeier and M. Gerdts, "Non-linear model predictive control of connected, automatic cars in a road network using optimal control methods," *IFAC-PapersOnLine*, vol. 51, no. 2, pp. 168–173, 2018.

[13] A. Katriniok, P. Kleibaum, and M. Joševski, "Distributed model predictive control for intersection automation using a parallelized optimization approach," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 5940–5946, 2017.

[14] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner, "Microscopic traffic simulation using sumo," in *The 21st IEEE International Conference on Intelligent Transportation Systems*. IEEE, 2018. [Online]. Available: <https://elib.dlr.de/124092/

[15] T. Afrin and N. Yodo, "A survey of road traffic congestion measures towards a sustainable and resilient transportation system," *Sustainability*, vol. 12, no. 11, p. 4660, 2020.

[16] D. Schrank, B. Eisele, and T. Lomax, "2019 urban mobility report. technical report," *Texas A&M Transportation Institute*, 2019.

[17] A. Ghosh, A. Maeder, M. Baker, and D. Chandramouli, "5g evolution: A view on 5g cellular technology beyond 3gpp release 15," *IEEE Access*, vol. 7, pp. 127 639–127 651, 2019.

[18] "Study on self-evaluation towards imt-2020 submission. available online:," https://www.3gpp.org/ftp//Specs/archive/37_series/37.910/, accessed: 20 January 2021.

[19] S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control engineering practice*, vol. 11, no. 7, pp. 733–764, 2003.

[20] R. Jones, "Quantitative effects of acceleration rate on fuel consumption. technical report," Environmental Protection Agency, Ann Arbor, MI (USA), Tech. Rep., 1980.

[21] L. Miao, J. J. Virtusio, and K.-L. Hua, "Pc5-based cellular-v2x evolution and deployment," *Sensors*, vol. 21, no. 3, p. 843, 2021.

Paper III

Paper III

Reproduced, with permission, from

SeyedeZahra Chamideh and William Tärneberg , and Maria Kihl, "A Safe and Robust Autonomous Intersection Management System using a Hierarchical Control Strategy and V2I communication,"*Submitted*,2021.

III

A Safe and Robust Autonomous Intersection Management System using a Hierarchical Control Strategy and V2I communication

Connected Autonomous Vehicle can significantly improve the safety and throughput of urban transportation systems. However, these systems are vulnerable to model uncertainties, wireless communication impairments, and external disturbances. In this paper, we investigate an Autonomous Intersection Management (**AIM**) system based on a hierarchical control strategy. In our proposed method, the Intersection Coordination Unit (**ICU**) in a Global Centralized Layer is responsible for assigning a safe speed to each vehicle while minimizing the system's cost. In the Local Decentralized Layer, each vehicle is responsible for tracking the reference speed assigned by the **ICU**, while avoiding collisions. In our method, each vehicle can use its own sensors to monitor its close surroundings, and take its own decisions on its movements, independent on the control decisions sent from the **ICU**. We investigate the safety, scalability and robustness of proposed method compared with two **AIM** methods based on centralized and decentralized control strategies. For the evaluation, we use the realistic urban mobility simulator **SUMO**. Further, we study the scalability and performance of the algorithms in the presence of communication impairments associated with wireless channels. Our simulation results show the proposed method can safely handle high traffic flow rates. Also, our method is robust to uncertainties caused by the wireless communication.

1 INTRODUCTION

According to the European Transport Safety Council's (ETSC) report on urban transportation networks, 18,844 people lost their lives in road traffic in the EU during 2020, despite the Covid-19 restrictions on travels. About 40% of these deaths occurred on urban roads [1]. Therefore, safe urban traffic management methods, and in particular intersection management methods, are crucial. Recent advances in intelligent transportation systems, in particular Autonomous Vehicle (**AV**) combined with Vehicle-to-Everything (**V2X**) technologies, and the introduction of *cooperative Autonomous Vehicle* present new opportunities to address urban transport challenges. By taking advantage of Vehicle-to-Everything (**V2X**) communication, Autonomous Intersection Management (**AIM**) systems have the potential to remove traditional intersection managers (i.e traffic lights), provide a safe crossing for vehicles, and optimize overall costs such as travel times, traffic throughput and fuel consumption [2].

Papers proposing **AIM** systems and methods usually formulate the intersection control problem as an optimization problem. The aim of the optimization is usually to find the vehicles' optimal *trajectories*. Here, a vehicle's trajectory is a time function representing a vector of the vehicle's spatial locations. The optimization is based on some objectives, considering system dynamics, and collision avoidance constraints. The required information is collected through negotiation and cooperation between road users and the infrastructure. An **AIM** method is generally based on *centralized*, *decentralized* or *hybrid* control strategies [3]. Centralized control strategies rely on a central Intersection Coordination Unit (**ICU**) that orchestrates the intersection and gives specific movement instructions to each vehicle. Centralized control strategies are always based on information collected from all vehicles in the intersection. Several papers have proposed and evaluated **AIM** methods based on centralized control strategies. For example, Drenser and Stone [4] designed an **AIM** method, which is based on time-space reservations in the intersection to avoid collisions. Lee and Park [5] proposed an **AIM** method based on a nonlinear constrained optimization problem with the objective to enhance the throughput performance of the intersection. The authors of [6] proposed a platoon-based **AIM** method, where vehicles in one platoon can accelerate or brake simultaneously. In [7], an **AIM** method is proposed with aim to minimise the total travel time using a mixed-integer linear programming (**MILP**) and a linear programming (**LP**) model.

However, **AIM** methods based on centralized control strategies have some major drawbacks. For example, the optimization problem will not scale well with increasing vehicle densities. Finding an exact solution becomes intractable for realistic vehicle densities, and thereby, the collision probability will be non-zero for high traffic volumes, which we have shown in [8]. In

[8], we compare two well-cited **AIM** methods based on centralized control strategies. Our conclusion in [8] is that **AIM** methods based on centralized control strategies are unsuitable for deployment in operational systems.

AIM methods based on decentralized control strategies have been proposed to tackle the imposed complexities of the centralized techniques [8]. In **AIM** methods based on decentralized control strategies, decisions are made locally by each vehicle. The local decisions are based on information collected from other vehicles and/or observations of the environment through local sensors. Several papers have proposed and evaluated **AIM** methods based on decentralized control strategies. For example, in [9] an **AIM** method is proposed that dynamically adjusts the vehicles' speed in order to provide both safety and comfort of the drivers. In [10], a decentralized optimization model is formulated with objective to find a collision-free trajectory for each vehicle. In [11], an **AIM** method based on a decentralized non-linear Model Predictive Control (**MPC**) approach was proposed, with the objective to minimize travel times. In [12], an **AIM** method based on a linear non-convex **MPC** approach was proposed.

However, also **AIM** methods based on decentralized control strategies have major drawbacks, in particular for high vehicle densities. In [13], we compare two well-cited **AIM** methods based on decentralized control strategies. Our conclusion is that the safety conditions, particularly in large-scale problems, may not be satisfied. Further, the solution may be sub-optimal due to the lack of global information.

Hybrid **AIM** methods can be developed as a solution to the crucial drawbacks with centralized and decentralized **AIM** methods. In hybrid methods, a combination of centralized control in the **ICU** and decentralized control in each vehicle can be used. Vehicles can communicate both with the **ICU** and/or with other vehicles, in order to achieve a more efficient intersection management [14]. Hybrid **AIM** methods usually include two layers of control: a centralized coordinator in a higher layer that is used to balance the network load and improve the traffic, and a second layer in each vehicle with a local controller that is responsible for following the rules from the higher layer and ensure a collision free crossing. For example, in the proposed hybrid **AIM** method [15], a centralized coordinator is responsible to assign a safe Time of Arrival and Velocity of Arrival for each approaching vehicle. Then, in each vehicle, a PID controller is responsible for keeping the trajectories obtained from the centralized coordinator. In [16], a bi-level **MPC** is proposed, where a centralized intersection manager allocates a time slot to each vehicle, and in each vehicle, the control commands for this vehicle are computed.

However, the current published hybrid **AIM** methods have some major drawbacks. For example, the lower layer control in a vehicle is only used for following this specific vehicle's trajectory. The vehicle trajectory is decided by the higher layer controller and then transmitted to the vehicle. This means that the local controller in each vehicle only follows commands from the central controller, which requires a perfect centralized control strategy and a perfect wireless communication link. By not introducing any local independent intelligence in the vehicles, the vehicles cannot avoid any potential collision situations that are independent of the centralized control management, for example, situations caused by pedestrians or vehicles that do not follow the commands from the central controller.

In order for an **AIM** method to be deployable in real operational systems (i.e real intersections), the **AIM** method needs to guarantee that the intersection is totally safe in all situations. For example, the control strategy used in the **AIM** method must be *scalable* in order to handle a large number of vehicles without compromising safety. Further, there will always be packet losses and delays in the wireless communication. This means that the wireless communication will lead to uncertainties that will cause many challenges when designing control strategies for **AIM** systems. Therefore, an **AIM** method must be *robust*, in order to guarantee safety when there are uncertainties in the system. In, for example [17, 18], measurements in operational autonomous vehicle systems showed delays of several milliseconds due to the wireless communication network and packet losses of about 3% due to disturbances on the wireless links. However, most current papers do not consider any uncertainties. For example, most papers do not model the wireless channel at all. Further, the proposed methods are usually evaluated only with numerical methods or simple simulation models, not by using well-known realistic simulation environments or testbeds. The optimization problem itself is usually the main focus, not the system performance. For example, most papers do not consider collision probability as an important performance metric.

In this paper, we propose a both safe and robust **AIM** system. Our **AIM** method is based on a hierarchical control strategy consisting of two control layers: a Global Centralized Layer (**GCL**) placed in the **ICU** and a Local Decentralized Layer (**LDL**) placed in each vehicle. Each vehicle can use its own sensors and local information to control its own movements, as modern vehicles equipped with advanced collision avoidance functions or self-driving capabilities. Each vehicle disseminates its local information to the **ICU**. The **GCL** calculates the optimal speed for all vehicles in the intersection using an **MPC** controller, and sends a reference speed to each vehicle. A local fast **MPC** controller in a vehicle tracks its assigned reference speed and ensure collision avoidance. By dividing the optimization control problem into two layers, and tackling a simpler optimization control problem with lower dimension

Figure 1: Targeted System and Intersection layout

on each layer, the computational complexity of the optimization problem is reduced.

We evaluate our **AIM** method in the well-known realistic simulation environment **SUMO** [19]. We show that our proposed method can guarantee a collision-free intersection for all vehicle densities, and significantly improve the intersection throughput and vehicles' fuel consumption. Further, we show that our method is robust against uncertainties imposed by impairments in the wireless communication channels.

2 TARGETED SYSTEM

In this paper, we consider the problem of Autonomous Vehicle coordination at a crossroad intersection without traffic lights, as depicted in Figure 1. The intersection's traffic flow pattern mainly depends on the intersection's geometry, location, and vehicle movements to and from the various lanes. We split the intersection area into three zones: the entrance zone, the critical zone, and the exit zone. The entrance zone is the area where vehicles approach the intersection boundary. In the critical area, there is a risk of collisions and coordination is crucial. The exit zone includes all vehicles that have passed the intersection, and thereby are leaving the system.

The intersection control is performed in an **ICU** and in all vehicles. The vehicles are equipped with On Board Unit (**OBU**), which employ a wide range of sensors. Further, each vehicle has a local computing node connected to the **OBU**, and can use optimization algorithms to avoid dangerous situations. The **OBU**s and the **ICU** exchange data through, for example, 5G-enabled Cellular Vehicle-to-Everything (**C-V2X**) technology.

Each vehicle periodically sends an information message to the **ICU** informing about its characteristics, position, and movement. Information messages are sent over Vehicle-to-Infrastructure (**V2I**) links provided by the wireless network. A modern vehicle with advanced collision avoidance functions, has the power of perception. For example, vehicles can detect other vehicles in front, by using cameras or radar sensors, and use this information to avoid collisions. This capability extends to a 360-degree field around the vehicle, enabling the car to detect and track all moving and static objects in its surroundings. Each vehicle can see both other vehicles in its surroundings and road obstacles with the help of its sensors. Therefore, in our proposed method there is no need for **V2V** communication.

The **ICU** will orchestrate vehicles, aggregate data, and provide a reference speed to each vehicle, based on the system's objectives. The **ICU** will predict and avoid dangerous situation according to the driving status of the vehicles. The system objectives are to balance the entire intersection traffic density, and obtain a smooth flow of vehicles with optimal system's speed and fuel consumption performance.

The **OBU**s use the received reference speeds in a local fast optimization algorithm, which controls the vehicle's movements so that the reference speed is maintained without causing collisions with other vehicles. An **OBU** can perform its optimization algorithm independent from the **ICU**, in order to avoid any collisions due to lost or delayed messages from the **ICU**.

The **OBU** should provide also the ordinary collision avoidance functions for the vehicle, in order for the system to be safe also for unforeseen potential collision situations, for example due to pedestrians or uncontrolled vehicles. However, these situations are not in the focus of this paper. Therefore, these types of events are not part of our optimization algorithms. However, our proposed **AIM** method can easily be extended to also include unforeseen potential collision situations caused by, for example, pedestrians, cyclists, or uncontrolled vehicles. Basically, we only need to include these events in the system model and the optimization algorithm in the **OBU**.

3 PROPOSED ALGORITHM

In this paper, we propose a new **AIM** method based on a hierarchical control strategy, which can guarantee all safety requirements of an operational **AIM** system, in particular for high vehicle densities and uncertainties caused by the wireless communication links. In this section, we first discuss some, in our opinion, major drawbacks of previously proposed **AIM** methods, and describe the approach we have used in order to solve the major challenges. Then, we describe an overview of our **AIM** method. After this, we proceed by describing the system model we have used, and finally we present high level descriptions of the algorithms used in the **AIM** method.

3.1 PREVIOUS RESEARCH AND OUR APPROACH

There has been much research on **AIM** methods. In most cases, proposed **AIM** methods are focused on improving the overall performance of the intersection, for example increasing the intersection's throughput or minimizing the fuel consumption of passing vehicles. Usually, it is the optimization formulation itself that is the main research focus, not the overall system performance. Further, most proposed methods are not evaluated for high vehicle densities or uncertainties caused by the wireless communication.

This means that previously proposed **AIM** methods skip all difficult scenarios, where collision avoidance may be a major challenge. Also, collision probability is seldom considered as a performance metric. Further, the proposed methods are usually only evaluated with numerical approaches or simple simulation models. Often, realistic system models are not used.

This means, in our opinion, that previous papers proposing **AIM** methods cannot guarantee a safe intersection in an operational system.

We believe that the essential goal of proposing an **AIM** method should be that it some day can be implemented in a real operational intersection. Therefore, the proposed method must guarantee a safe crossing of vehicles and make sure that all road users are protected even in places with high traffic densities and/or limited visibility. Therefore, it will be crucial to evaluate the **AIM** method in close to real world scenarios including the wireless communication.

Our proposed **AIM** method builds on a two-layer **MPC** controller, and we have used the following approach:

- We consider safety as the main performance metric. The expected number of collisions is calculated for all scenarios, in order to validate that our proposed **AIM** method can protect the users.

- We evaluate our proposed **AIM** method for different vehicle densities, in order to validate that our method can safely manage also high vehicle densities, corresponding to rush hour scenarios.
- We evaluate how packet loss and communication delay impact the system performance, in order to validate that our proposed **AIM** method is robust to uncertainties caused by the wireless communication.
- In our proposed **AIM** method, the local **MPC** in each vehicle can take independent decisions about its movements, as modern vehicles with advanced collision avoidance functions or self-driving capabilities, in order to always guarantee a safe driving.
- We evaluate our proposed method in real world scenarios using Simulation of Urban MObility (**SUMO**), which is a microscopic traffic simulation package designed to handle large networks.

3.2 OVERVIEW OF OUR PROPOSED AUTONOMOUS INTERSECTION MANAGEMENT METHOD

Our proposed **AIM** method includes two layers: a *Global Centralized Layer* placed in the **ICU** and a *Local Decentralized Layer* placed in each vehicle. The main uniqueness of our proposed **AIM** method compared with **AIM** methods based on hybrid control strategies is that the local controller placed in each vehicle can take its own intelligent decisions, independent of the global controller placed in the **ICU**. In other propsed **AIM** methods, the local controller only follows the commands coming from the global controller.

Our proposed **AIM** method employs Model Predictive Control (**MPC**) algorithms in each layer in order to control the vehicles' movements. **MPC** is an iterative optimization model that is computed in sampling intervals. For each sampling interval, a cost minimizing control strategy is computed. The optimization is performed both for the current sampling interval, and for future sampling intervals, called the time horizon. Only the first step of the calculated control strategy is implemented, and the calculation will be repeated in the next sampling interval [20].

The first layer, the *Global Centralized Layer*, employs an **MPC** algorithm that is based on data from all vehicles in the intersection. The overall objective of the Global Centralized Layer is to control the speed of each vehicle involved in the intersection, in order provide a smooth flow of vehicles without collisions. This optimal speed for each vehicle, called the *reference speed*, is then sent to each vehicle.

The reference speed is used in the second layer, the *Local Decentralized Layer*, where each vehicle performs a local optimization based on the system's objectives. The main goal of the Local Decentralized Layer is to avoid collisions, and follow the reference speed obtain from the **GCL**. To ensure a minimum distance to the vehicle ahead, all vehicles solve a local **MPC** optimization problem. The **MPC** algorithm is based on the reference speed received from the **ICU** and the collected data from the on-board sensors. Each vehicle periodically sends information messages to the **ICU** containing its current position, speed and acceleration. This local control is performed with shorter sampling intervals compared with the centralized control, which means that the vehicle quickly can react to changes in its close surroundings.

In perfect situations, the two control layers will function as described above. However, there will always be situations that are not ideal. For example, in some situations, the **GCL** may not find a feasible solution for all vehicles [8]. Also, the information messages sent between the **ICU** and vehicles may be delayed or dropped, due to uncertainties in the wireless communication. In these situations, the Local Decentralized Layer will make its own decisions, based only on its local data, until a reference speed from the **ICU** is received.

3.3 SYSTEM MODEL

In this section we present the general system model that is used by our proposed **AIM** method.

3.3.1 Intersection

The intersection is modeled as a circle with radius R, which is divided into three zones: the entrance zone, the critical zone, and the exit zone. The entrance zone is the area where vehicles approach the intersection boundary, and it contains one or several lanes, denoted entrance lanes. We denote all entrance lanes with a set $\mathcal{E} = \{1, 2, \ldots, E\}$. The exit zone includes all lanes, denoted exit lanes, where the vehicles leave the intersection. We denote all exit lanes with a set $\mathcal{K} = \{1, 2, \ldots, K\}$. In the critical zone, there is a risk of side collisions. The entrance lanes and exit lanes are connected through internal lanes inside the critical zone. We denote all internal lanes with a set $\mathcal{I} = \{I = e + k | e \in \mathcal{E}, k \in \mathcal{K}\}$, where $e + k$ implies that lane e is connected to lane k.

A vehicle that enters the intersection will follow a *path*, which starts from one entrance lane, cross one internal lane and end in one exit lane. We describe all possible paths in the intersection with the set $\Gamma = \{(e, k) | e \in \mathcal{E}, e \in \mathcal{K}\}$. Since the intersection is modeled as a circle, all different paths have almost the same length.

Two different paths may intersect inside the critical zone. The vehicles on these paths while crossing their corresponding internal lanes have a potential risk of collision in this specific intersect point, called the *conflict point (CP)* of the corresponding paths. We describe all possible conflict points in the critical zone with the set $\mathcal{CP} = \{CP(\gamma_i, \gamma_j) | \gamma_i, \gamma_j \in \Gamma, \gamma_i \perp \gamma_j, i \neq j\}$ where $\gamma_i \perp \gamma_j$ implies that γ_i and γ_j are intersecting paths.

3.3.2 Vehicles

A set of $\mathcal{N}_t = \{1, 2, \ldots, N_t\}$ **AV** exist in area of the intersection at time t. Each vehicle $i \in \mathcal{N}_t$ has a predetermined path $\gamma_i = (e_i, k_i) \in \Gamma$ and the path is perfectly followed. A vehicle will not change its path while crossing the intersection. The total time vehicle i spends on its path is called *Traveling time* and denoted by Tr_i.

A vehicle is modeled as a point on its path. Within the intersection area, a vehicle i has a *position*, $p_i(t)$, at time t, on a lane $l_i \in \mathcal{E} \oplus \mathcal{K} \oplus \mathcal{I}$. The vehicle's position corresponds to the remaining distance from the center of the corresponding path. Note that the sign of $p_i(t)$ changes when the vehicle cross the center of I_i. Therefore, $p_i(t)$ always has positive values when the vehicle is on e_i and negative values when the vehicle is on k_i. Similarly, $v_i(t) = \dot{p}_i(t)$ and $u_i(t) = \ddot{p}_i(t)$ denote the speed and acceleration of vehicle i at time t. The maximum deceleration (i.e the minimum acceleration) and maximum acceleration are described by u_{min} and u_{max}, respectively. Vehicles are not allowed to make U-turns in the intersection area, which means that the minimum speed is always positive along the trajectory of a vehicle. There is a maximum speed limit in the intersection, which is denoted v_{max}.

The overall aim of the control algorithm is to find an optimal safe (i.e collision-free) trajectory for each vehicle, where a vehicle's trajectory is defined as a time function representing the vehicle's position at any time [21]. A safe trajectory for vehicle i is achieved by controlling the motion of the vehicles along their paths such that only one vehicle can access a specific conflict point at a specific time.

Since vehicles' positions depend on their speeds and accelerations, we need to define a system model, which is a function of position, speed and acceleration. We have decided to use a discrete system model. The discrete longitudinal motion of the vehicle, denoted by $x_{i,n} = [p_{i,n}, v_{i,n}]^\intercal$, is the system state. The system state can be controlled by its acceleration $u_{i,n}$. $p_{i,n}, v_{i,n}$ and $u_{i,n}$ are the position, speed and acceleration of vehicle i at time $t = n\tau$, where τ is the model's sampling interval. As described later, the **GCL** and the **LDL** use this system model with different sampling intervals.

Since vehicles have a limited maximum speed and acceleration, there are the following inequality constraints on the vehicle's movement parameters:

$$\begin{aligned} &u_{min} \leq u_{i,n} \leq u_{max} \qquad \forall n \geq 0 \\ &0 \leq v_{min} \leq v_{i,n} \leq v_{max} \qquad \forall n \geq 0 \end{aligned} \tag{1}$$

3.4 GLOBAL CENTRALIZED LAYER

In this section, we give a high-level description of the **MPC** algorithm used in the Global Centralized Layer.

The main objective of the Global Centralized Layer optimization is to achieve a safe intersection, with a smooth and comfortable flow of all vehicles. The optimization is performed every sampling interval τ_c, for a predefined time horizon T_c. The output of the optimization is a reference speed for each vehicle, which is then sent to each vehicle, and used in the Local Decentralized Layer.

3.4.1 Space state model

The control system is updated in discrete sampling intervals τ_c. The discrete space state model of vehicles $i \in \mathcal{N}_t$ in sampling interval n is given in Equation (2).

$$x_{i,n+1} = \begin{bmatrix} 1 & -\tau_c \\ 0 & 1 \end{bmatrix} x_{i,n} + \begin{bmatrix} -\frac{1}{2}\tau_c^2 \\ \tau_c \end{bmatrix} u_{i,n} \qquad 0 \leq n \leq T_c, \forall i \in \mathcal{N}_t \tag{2}$$

The initial value, $x_{i,0}$, to start the algorithm is obtained from the *latest* information message collected from vehicle i. This means that the **GCL** algorithm is robust to uncertainties in the wireless communication. Some information messages can be lost, and the **GCL** algorithm will still be able to function properly.

3.4.2 Constraints

The objective of the constraints is to avoid collisions by keeping a safe distance between all vehicles. The constraints are a function of all vehicles states $x_{i,n} = [p_{i,n}, v_{i,n}]$, and the control input $u_{i,n}$.

To avoid rear-end collisions, a minimum separation distance between two vehicles on the same lane, d_{min}, is defined. d_{min} corresponds to the minimum distance between the center point of two vehicles, since vehicles are modeled as a point on the road. The following constraint in Equation (3) is defined to prevent rear-end collisions between vehicle i and its leading vehicle j, where $j \in \mathcal{N}_t$ is the vehicle in front of vehicle's i, *leading vehicle (**LV**)*, at step $n = 0$.

$$p_{i,n} - p_{j,n} \geq d_{min} \qquad \forall i \in \mathcal{N}_t, j = LV(i), 1 \leq n \leq T_c + 1 \tag{3}$$

To ensure that no collisions occur between vehicle i and vehicle j within the critical zone, i.e to ensure that only one vehicle enters a conflict point

$CP(\gamma_i, \gamma_j)$ at a specific time, a linear inequality constraint is defined as in Equation (4) below.

$$|p_{i,n} - C_{i,j} - p_{j,n} + C_{j,i}| \geq R_{min} \qquad \begin{aligned} &l_i, l_j \in \mathcal{E} \oplus \mathcal{I}, \gamma_j \in \Gamma_i, \\ &1 \leq n \leq T_c + 1 \end{aligned} \tag{4}$$

where $C_{i,j}$ and $C_{j,i}$ are the remaining distance from the center of the corresponding path, γ_i and γ_j, to the conflict point $CP(\gamma_i, \gamma_j)$ of vehicles i and j, inside the critical zone, as described in Section 3.3. Γ_i is a set of all paths, γ_j, which have the potential to collide with vehicle i from path γ_i inside the critical zone. The constraint $l_i, l_j \in \mathcal{E} \oplus \mathcal{I}$ ensures that neither vehicle i nor j already have crossed the intersection. R_{min} is a constant that denotes the minimum separation distance between two vehicles within the critical zone. Equation (4) is a non-convex inequality, which leads to a non-convex optimization problem.

To cope with the non-convex limitation, a semi-definite programming relaxation is applied, which is achieved by introducing a priority scheme on vehicles, such that a vehicle with lower priority must give right of way in case of a potential conflict. In this paper, we use a simple First-Come-First-Serve priority scheme with a basic right of way rule, as commonly used in many uncontrolled intersections. Therefore, the constraint Equation (4) can be modified to the following convex constraint Equation (5).

$$p_{i,n} - p_{j,n} \geq R_{min} + r_{i,j} \qquad \begin{aligned} &j \in \mathcal{P}_i, r_{i,j} = C_{i,j} - C_{j,i} \\ &l_i, l_j \in \mathcal{E} \oplus \mathcal{I} \\ &|p_{i,0} - p_{j,0} - r_{i,j}| \leq R_{min} \\ &1 \leq n \leq T_c + 1 \end{aligned} \tag{5}$$

Where $\mathcal{P}_i$ is a set of all vehicles j from $\gamma_j \in \Gamma_i$, which have priority on vehicle i. The constraint $|p_{i,0} - p_{j,0} - r_{i,j}| \leq R_{min}$ ensures that the vehicles i and j have almost the same distance from their corresponding conflict point at the first sampling interval.

3.4.3 System cost and optimization

The **MPC** requires a system cost function that should be minimized. In this paper, we have based the system cost on the vehicles' speed and fuel consumption, however, the system cost can of course be based on other metrics as well.

Th first objective of the control system is that vehicles should drive with as high speed as possible. For this objective, we use a so called *Target speed*, which is the reference speed value that the optimization algorithm should aim for. The Target speed should of course be less than or equal to the maximum speed limit, however, the strategy for setting the Target speed is out of the scope of this paper. We denote the Target speed with a time varying variable v_d^t. Therefore, the first objective of the control system is that the vehicles' speed should be as close as possible to the current Target speed, v_d^t.

The second objective of the control function is to achieve a minimum fuel consumption. Several factors can have an effect on the fuel consumption, and it varies with the vehicle type, weather condition, driving behaviors such as rapid acceleration, speed. The Environmental Protection Agency (EPA) study [22] shows that the acceleration rates have a significant effect on a vehicle's fuel consumption. Therefore, minimum fuel consumption translates into *minimizing the absolute accelerations and the acceleration rates*. In addition, a smooth flow of vehicles, and thereby, a smooth change of acceleration is desired.

Therefore, the system cost, J is defined as in Equation (6). where $v_{i,n}$ (the speed of vehicle i at time interval n) and v_d^t (the Target speed at time t) are the **MPC** control and reference variables, respectively, and $u_{i,n}$ (the acceleration of vehicle i at time interval n) is the manipulated variable.

$$\begin{aligned} J = & \sum_{n=0}^{T_c} \sum_{i=1}^{N} (w_{v_i}(v_{i,n+1} - v_d^t)^2 + w_{u_i}(u_{i,n})^2) \\ & + \sum_{n=0}^{T-1} \sum_{i=1}^{N-1} w'_{u_i}(u_{i,n+1} - u_{i,n})^2 \end{aligned} \tag{6}$$

Here, w_{v_i} ,w_{u_i} and w'_{u_i} are weight coefficients, and J is the problem cost to be minimized, subject to the given current states of the vehicles as defined in Equation (2) and constraints as defined in Equations (1), (3) and (5).

The **ICU** will calculate the optimal speed for each vehicles involved in the intersection and send the results to the corresponding vehicles, which will be used as a reference speed for the Local Decentralized Layer control problem in the vehicle during the next sampling interval τ_c. The Algorithm 1 summarizes the method used in the Global Centralized Layer.

Since the Local Decentralized Layer can guarantee a safe movement, independent from the Global Centralized Layer results, the sampling interval in Equation (2) , τ_c, can have a greater value compared with conventional centralized **MPC** methods. Therefore, the method complexity is reduced.

3.5 LOCAL DECENTRALIZED LAYER

In this section, we give a high-level description of the **MPC** algorithm used in the Local Decentralized Layer. The main objective of the **LDL** optimization

Algorithm 1 Global Centralized Layer, **ICU** level

1: **for** Every centralized sampling interval τ_c **do**
2: Collect $p_{i,0}$, $v_{i,0}$ and γ_i from vehicles $i \in \mathcal{N}_t$ as to form initial states.
3: Estimate vehicles' future states by Equation (2) $\forall i \in \mathcal{N}_t$.
4: Calculate the targeted acceleration for each vehicle ,$u_{i,0}$, to minimize cost Equation (6) subject to constraints Equations (1), (3) and (5) and system model Equation (2).
5: Calculate the reference speed for each vehicle to be used for second layer control algorithm, $v_{i,1}$, from Equation (2).
6: Send $\hat{v}_i^t = v_{i,1}$ to vehicle $i \in \mathcal{N}_t$ as reference speed.
7: **end for**

is to determine the vehicle's movements, according to the vehicle's path γ_i, aiming for its reference speed, using the latest $\hat{v}_i^t$ received from the **ICU**, while avoiding other vehicles detected by the sensors.

3.5.1 Space state model

The optimization is performed every sampling interval τ_d, for a predefined time horizon T_d. The discrete space state model of vehicle i is given in Equation (7).

$$x_{i,n+1} = \begin{bmatrix} 1 & -\tau_d \\ 0 & 1 \end{bmatrix} x_{i,n} + \begin{bmatrix} -\frac{1}{2}\tau_d^2 \\ \tau_d \end{bmatrix} u_{i,n} \qquad 0 \leq n \leq T_d \tag{7}$$

Where $x_{i,0}$ is the vehicle's current state (position and speed).

3.5.2 Constraints

The **GCL** control algorithm should have calculated a reference speed that guarantees that a vehicle has exclusive access to each conflict point on its path. However, to ensure robustness, the **LDL** control algorithm will not blindly follow the reference speed coming from the **GCL**. The objective of the constraints in the **LDL** control algorithm is to continuously ensure a safe distance to surrounding vehicles, which can be seen by the on-board sensors, and thereby avoid unforeseen collisions. To this aim, the following constraint in equation Equation (8) is defined.

$$p_{i,n} - \hat{p}_{j,n} \geq d_{min} \qquad i \in \mathcal{N}_t, \;\; j = \mathcal{O}(i), \;\; 1 \leq n \leq T_d + 1 \tag{8}$$

where d_{min} is the minimum separation distance between two vehicles and $\mathcal{O}(i)$ is the leading vehicle of vehicle i.

3.5.3 System cost and optimization

The general objective of the control system is to achieve a safe and smooth flow of vehicles. Therefore, as in Section 3.4, the system cost is defined as a function of the vehicle's speed and acceleration. The algorithm's reference speed, $\hat{v}_i^t$ at time t, is received from the latest Global Centralized Layer optimization results.

$$\begin{aligned} J_i = & \sum_{n=0}^{T_d} (w_{v_i}\delta_i(v_{i,n+1} - \hat{v}_i^t)^2 + w_{u_i}(u_{i,n})^2) \\ & + \sum_{n=0}^{T_d-1} w'_{u_i}(u_{i,n+1} - u_{i,n})^2 \end{aligned} \tag{9}$$

Here, w_{v_i} ,w_{u_i} and w_u are the same weight coefficients as in Equation (6), and δ_i is a binary variable that states whether $\hat{v}_i^t$ has been received from the **ICU** or not. The algorithm will ensure collision free movements during the **MPC** prediction horizon T_d even if the reference speed has not been received.

J_i is the system cost for vehicle i, which is to be minimized for the time horizon T_d, subject to the given current states of the vehicles as defined in Equation (7) and constraints as defined in Equation (8). The Algorithm 2 summarizes the control algorithm used in the Global Centralized Layer.

Algorithm 2 Local Decentralized Layer, vehicle level

for all $i \in \mathcal{N}_t$ in parallel **do**
  for Every sampling interval τ_d **do**
    Received $\hat{v}_i^t$ from **ICU**
    if Vehicle i has not received $\hat{v}_i^t$ **then**
      $\delta_i = 0$ in Equation (9)
    end if
    Estimate the vehicle's future states by Equation (7).
    Calculate the optimal acceleration ,$u_{i,0}$, to minimize cost Equation (9) subject to constraints Equation (8).
    Apply the obtained acceleration.
    Send the updated state to **ICU** through the wireless network.
  end for
end for

4 EVALUATION

In this section, we describe our simulation environment and our experiments. The main objective of our experiments is to evaluate the performance of an intersection controlled by our proposed **AIM** method. We have investigated the system performance in terms of safety, scalability, and robustness.

4.1 SIMULATION ENVIRONMENT

We have developed a simulation environment based on Simulation of Urban MObility (**SUMO**) [19]. **SUMO** is an open source, highly portable, microscopic and continuous traffic simulation package that gives the user control over all aspects of the networked system, such as road topology, vehicle type, driver behaviour, intersection control, and statistical data collection. In our work, we have modified **SUMO** so that, instead of using the default microscopic flow algorithms in **SUMO**, the speed of each vehicle can be manipulated by our control strategy. This has been performed via the Traffic Control Interface (**TraCI**) in **SUMO**. Also, we changed the values of some of **SUMO**'s default configuration parameters, as shown in Table I.

The vehicles' speed in **SUMO** can be controlled by **TraCI** with commands setSpeed (0x40) and slowDown (0x14). However, a vehicle may drive slower or faster than this speed due to the car following model in **SUMO**. In order to force the vehicles to follow our control strategy, we disabled the behavior imposed by the car following model by using the speed mode (0xb3) command and set all checks off. In each **SUMO** simulation time step, the vehicles' speed can be calculated based on the control strategy calculated in Equation (9) and perfectly followed.

In our simulations, all vehicles have the same physical properties and they arrive to the intersection according to a pre-generated traffic demand stored in a route XML-file. In the route file, the vehicles' arrivals will be randomized using a Poisson distribution and an arriving vehicle is given a specific path when it arrives. The probability for receiving a specific path is the same for all paths. **SUMO** will report a collision when the physical gap between two vehicles is 0. In the simulation, the collided vehicles will immediately be removed by TracI.

4.2 EXPERIMENTS

In this paper, we consider a basic four-way intersection where each crossing road has two lanes. Each lane is 3.5m wide, and there is a maximum speed limit of about 72 km/h (v_{max} = 20 m/s). We assume that the vehicles enter the intersection with an initial speed that is slightly lower than the maximum speed limit. In addition, the intersection area is modelled as a circle with radius 150m. In our simulations, all four entrance zone have similar traffic flow rates. In our simulations, we used a fixed Target speed of 16 m/s (about 57 km/h), however, we will in the results also discuss the effects of

Table I: SUMO configuration

Parameter	Value	Description
step-length	0.05 s	Simulation time step
collision. mingap-factor	0 m	Collisions detection distance
collision.action	Warn	Collision warning is issued
collision.check-junctions	True	Check collisions between vehicles in the intersection

a dynamic Target speed. Table II summarizes the simulation parameters and the specifications that we used in our simulations.

We have evaluated our proposed method for different traffic volume, also called *traffic flow rates*. The traffic flow rate is normally given in terms of arriving vehicles per hour per lane.

All results will be compared with 1) a conventional intersection control based on traffic lights with 90 second green phase and 90 second red phase; 2) an **AIM** method based on a centralized control strategy using an **MPC** algorithm, and 3) an **AIM** method based on a decentralized control strategy using an **MPC** algorithm. The **AIM** methods we used in the comparison are published in well-cited papers, and they have previously been evaluated in [8] and [13]. The **MPC** algorithm in these **AIM** methods has the objective to prevent collisions in the whole intersection area over a given time horizon of 10 seconds with a sampling interval of 1 second, by keeping vehicles at a safe distance to each other. To ensure a fair comparison, the traffic demands and initial speed of each vehicle are the same for all evaluated **AIM** methods.

The *saturation flow rate* for an intersection corresponds to the maximum achievable traffic flow rate when there is a high traffic demand. For an intersection controlled by traffic lights, the saturation flow rate depends on several factors, such as the intersection geometry, safety policy and the surrounding environment [23]. The study in [23] shows that the saturation flow rate for an intersection controlled by traffic light is almost 900 vehicles per hour per lane. The saturation flow calculation is based on a 2 seconds headway between vehicles for safety.

In this paper, a saturated intersection corresponds to a situation where the maximum number of vehicles coexist inside the intersection, and the

Table II: Simulation Parameters

Intersection parameters				
R	150 m	The radius of intersection zone		
u_{max}	5 m/s^2	Maximum allowed acceleration		
u_{min}	-6 m/s^2	Maximum allowed deceleration		
v_{max}	20 m/s	Maximum allowed speed		
v_{min}	0 m/s	Minimum allowed speed		
System Parameters				
R_{min}	7 m	Minimum separation distance between the centre of two vehicles from different approaching lanes		
d_{min}	6 m	Minimum separation distance between the centre of two vehicles in the same lane		
v_d^t	16 m/s	Target speed at intersection $\forall t$		
w_{v_i}	1	Weighting coefficient $\forall i$		
w_{u_i}	0.1	Weighting coefficient $\forall i$		
w'_{u_i}	0.5	Weighting coefficient $\forall i$		
GCL Parameters		**LDL** Parameters		
T_c	10 s	T_d	2 s	**MPC** prediction horizon
τ_c	1 s	τ_d	0.1 s	**MPC** sampling interval

vehicles have a separation distance of d_{min}. From the optimisation problem Equation (6), we can expect that the vehicles' average speed will be very close to v_d^t. Since the intersection is full of vehicles, at least one vehicle need to leave the intersection for another vehicle to enter in each road. The time it takes for the first vehicle at the end of an exit lane to leave the intersection zone is $t_h = d_{min}/v_d^t$. In an intersection with N_2 exit lanes and N_1 entrance lanes, the average number of N_2 vehicles can leave the intersection during period t_h simultaneously. That means that at maximum of N_2 vehicles can enter from N_1 entrance lanes, We are looking for the maximum achievable traffic flow rate for each lane. Since in each second, maximum $min(N_2, N_1)/t_h$ vehicles can enter from N_1 lanes, the maximum traffic flow rate in terms of vehicles per hour, can be obtained from Equation (10).

$$C = \frac{3600 * min(N_2, N_1)}{t_h \cdot N_1} \tag{10}$$

In this paper, the traffic flow rate is divided in three different *traffic volumes*, as in [24]. The peak hour traffic flow rate for a typical intersection in an urban area is usually 450-650 vehicles/hour/lane, which is defined as *High volume traffic*. A traffic flow rate of 150-450 vehicles/hour/lane is defined as *Medium volume traffic*. Finally, a traffic flow rate of less than 150 vehicles/hour/lane is defined as *Low volume traffic*. In the result graphs, we will highlight the different traffic volumes by using different background colours: *Green* for traffic flow rates corresponding to Low volume traffic, *yellow* for Medium volume traffic and *red* for High volume traffic.

4.3 PERFORMANCE METRICS

In this paper, we use several performance metrics in the evaluation. The first two performance metrics, *average speed* and *fuel consumption*, are the most used performance metrics in the literature, when evaluating the performance of **AIM** methods. However, we also include *safety*, *scalability* and *robustness*, since these performance metrics will be crucial when deploying an operational **AIM** system.

4.3.1 Average speed

The system performs well when vehicles can pass the intersection with a high speed. Therefore, we have evaluated the *average speed* of all vehicles in the intersection for different traffic flow rates. The average speed for vehicle i, $\bar{V_i}$ is calculated as in Equation (11).

$$\bar{V_i} = \frac{\int^{Tr_i} v_i(t)}{Tr_i} \tag{11}$$

where Tr_i is the Traveling time of vehicle i. The average speed of all vehicles during a simulation run, $\bar{V}$, is obtained by calculating the average speed of all vehicles that have passed the intersection during the simulation, shown in Equation (12).

$$\bar{V} = \frac{\sum_i^{N_l} \bar{V_i}}{N_l} \tag{12}$$

where N_l is the number of vehicles that have passed the intersection when the simulation ends.

4.3.2 Fuel consumption

An algorithm performs well if it results in a low fuel consumption (or electricity for electric cars) for the vehicles. As described in section 3.4, the acceleration can have a significant effect on a vehicle's fuel consumption. Therefore, we have used the *average absolute acceleration* for different traffic flow rates as a metric for fuel consumption. The average absolute acceleration for vehicle i, denoted U_i, during its traveling time Tr_i is calculated as shown in Equation (13).

$$U_i = \frac{\int^{Tr_i} |u_i(t)|}{Tr_i} \tag{13}$$

The average absolute acceleration for all vehicles during a simulation, U, is obtained by calculating the average absolute acceleration of all vehicles that have passed the intersection during the simulation, as shown in Equation (14).

$$U = \frac{\sum_i^{N_l} U_i}{N_l} \tag{14}$$

4.3.3 Traffic safety

An operational **AIM** system needs to be totally safe. In this paper, we have used an approximation of *average number of collisions per hour* as the main performance metric for traffic safety. As explained in Section 4.1 a collision is defined when the physical distance between two vehicles is zero. For each traffic flow rate, we ran the simulation several times with different random seeds (i.e different traffic demand profiles), where each simulation run was 1 hour. We measured the number of collisions that **SUMO** detected during each simulation run, and then calculated an average of the number of detected collisions per hour, denoted $\bar{N}_c$, as shown in Equation (15).

$$\bar{N}_c = \frac{\sum_k^{N_r} N_C^k}{N_r} \tag{15}$$

where N_r is the total number of simulation runs and N_C^k is the the number of detected collisions in simulation run k. The resulting metric, $\bar{N}_c$, should be seen as an approximation, not a fact. For example, we used more simulation runs for lower traffic flow rates than for higher traffic flow rates. The main aim with this metric is to evaluate if an **AIM** method can be considered safe or not. Therefore, the absolute values of $\bar{N}_c$ are not relevant, only the comparison between methods.

4.3.4 Scalability

An operational **AIM** system must be scalable in order to safely control the intersection also for high traffic flow rates. In this paper, we have used the *maximum traffic flow rate that the **AIM** method can safely control* as the main performance metric for scalability. An intersection is considered safe if the average number of collisions per hour (calculated as above) in the intersection is less than 0.2 collisions/hour. As above, the absolute values are not important, only the comparison.

4.3.5 Robustness

Operational **AIM** systems need to be robust to uncertainties caused by the wireless communication. In this paper, we have evaluated the effect on the safety when adding packet loss and communication delays, and used these results as a performance metric for robustness of the system.

5 RESULTS AND DISSUASION

In this section, we show and discuss the results from our evaluation.

5.1 SPEED

Figure 2 illustrates the average speed of each vehicle with 95% confidence intervals for different traffic flow rates. It is apparent that the **AIM** methods perform better compared to an intersection controlled by traffic lights. The resulting average speed for all traffic flow rates is similar to or slightly below the Target speed. Also, our proposed **AIM** method results in similar average speed as the **AIM** methods based on centralized and decentralized control strategies. This is expected, since the overall system objectives for all three **AIM** methods are the same.

Figure 3 shows the number of vehicles that coexist within the intersection area with a traffic flow rate of 700 vehicles/hour/lane during 10 minutes, starting after 30 minutes since the beginning of the simulation (in order to have a steady-state situation). The simulation results clearly show that the steady-state average of the number of vehicles inside an intersection controlled by our proposed **AIM** method is almost 5 times less than for an intersection controlled by traffic lights. Vehicles cross the intersection quickly, since they do not have to stop for traffic lights. This results in an urban traffic system without traffic queues.

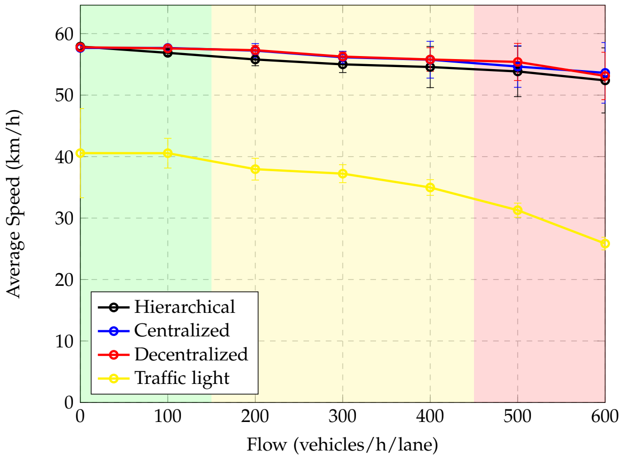

Figure 2: The average speed of vehicles while crossing the intersection.

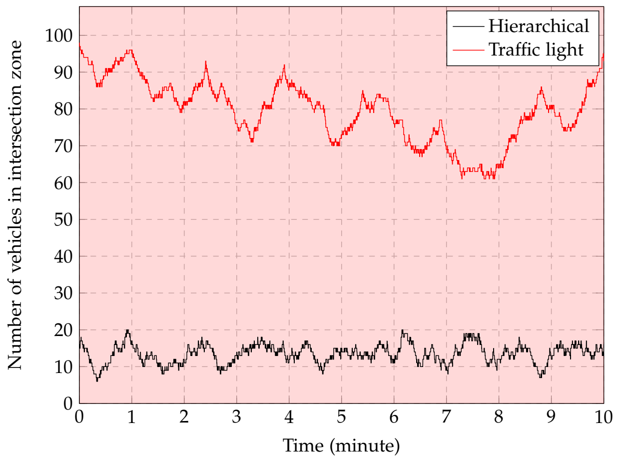

Figure 3: The number of vehicles that coexist inside the intersection for a traffic flow rate of 700 vehicles/hour/lane.

5.2 FUEL CONSUMPTION

Both layers of our proposed **AIM** method include an optimization objective that the vehicle should move with a smooth and comfortable speed. Therefore, vehicles will avoid high acceleration changes, which can result in a lower fuel consumption. Figure 4 shows the average absolute acceleration for the different **AIM** methods with 95% confidence intervals. As expected, intersections controlled by traffic lights, where many vehicles unnecessarily have to brake due to red lights, will have the highest average absolute acceleration. The three **AIM** methods will result in rather similar average absolute acceleration for vehicles crossing the intersection.

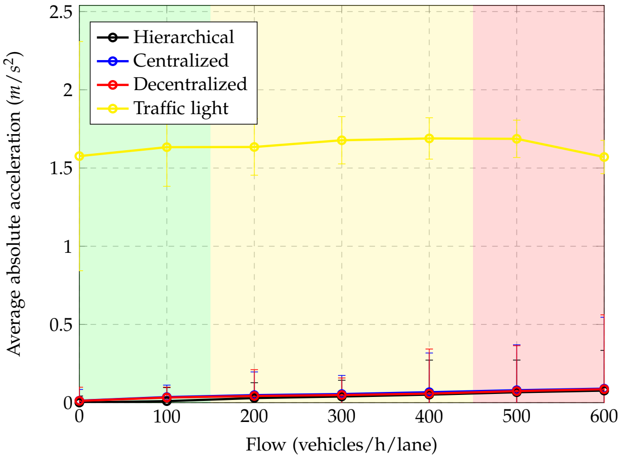

Figure 4: The average absolute acceleration for vehicles crossing the intersection.

5.3 TRAFFIC SAFETY

Our results regarding speed and fuel consumption validate the typical performance metrics of **AIM** methods in comparison with intersections controlled by traffic lights. The average speed when crossing the intersection and the vehicles' expected fuel consumption are improved using **AIM** methods, either based on centralized, decentralized or hybrid control methods. However, in order to deploy an **AIM** method in an operational system, the traffic safety must be evaluated as well, since this will be crucial for operational systems.

In order to evaluate the traffic safety, we have made an approximation of the average number of collisions per hour, as explained in Section 4.3, for different traffic flow rates, as shown in Figure 5. An intersection controlled by traffic lights is assumed to have zero collisions per hour for all traffic flow rates, since this is the main reason for deploying traffic lights in intersections. In the simulations shown here, the wireless communication links are assumed to be perfect, which means that there are no packet losses or communication delays.

However, this results in major problems for the **AIM** methods based on centralized and decentralized control strategies when the traffic flow rate is higher than 400 vehicles/hour/lane. For these traffic flow rate, the control

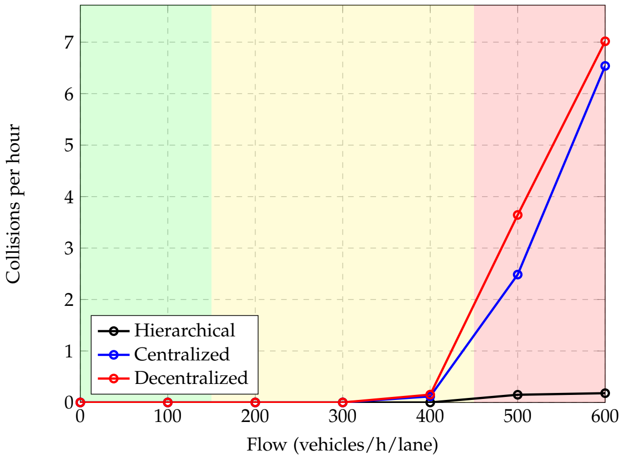

Figure 5: Average number of collisions per hour for different traffic flow rates

algorithms have problems to find solutions that ensure that all vehicles can keep a safe distance to all other vehicles, and thereby avoid collisions. For centralized control strategies, the main problem is due to the complexity of the optimization problem. For decentralized control strategies, the main problem is that each vehicle does not have full knowledge of the global situation. These problems result in a non-zero expected number of collisions in the intersection, also for rather low traffic flow rates, corresponding to Medium traffic volumes, and perfect wireless communication channels.

In our proposed **AIM** method, the Local Decentralized Layer can avoid collisions by keeping a safe distance to other vehicles in its close surroundings, also when the **GCL** has not been able to find a feasible solution. Therefore, our **AIM** method has a much improved traffic safety performance in comparison with the **AIM** methods based on centralized and decentralized control strategies, as shown in Figure 5. For example, with a traffic flow rate of 600 vehicles/hour/lane, 2400 vehicles/hour, the average number of collisions is almost 6.5 collisions/hour for the compared **AIM** methods, while the average number of collisions is only 0.18 collisions/hour for our proposed method.

5.4 ROBUSTNESS

In most evaluations of **AIM** methods, perfect wireless communication is assumed, or not taken into consideration. This means that packet loss and communication delay are not included in the evaluation. However, this is of course not a correct assumption for real world scenarios with current communication technologies. For example, the measurements in [17] for

C-V2X show that the average practical end-to-end communication delay in current 5G technologies is 10ms with almost 2.5% packet loss (i.e. 97.5% reliability) for packet transmission when the vehicle speed is around 18 km/h. In addition, Ford in partnership with Qualcomm (in US) and Datang (in China) has been testing C-V2X devices since 2017 [25, 26]. The test results are encouraging, however, they also confirm that the reliability of the wireless links depends on, for example, the vehicles' speed, the communication range, the weather, and environment noise.

Non-ideal wireless communication, i.e wireless communication links with packet loss and/or communication delays, have different impact on **AIM** methods based on centralized and decentralized control strategies. In **AIM** methods based on centralized control strategies, the global optimization algorithm is usually only reliable and can avoid collisions if *all* messages from vehicles in the system are received correctly and in due time (i.e before the next sampling intervals where the vehicle data is used). Therefore, if only one message is dropped, the algorithm may fail to find a safe trajectory for all vehicles. The same is valid for the local optimization algorithm in **AIM** methods based on decentralized control strategies. However, in decentralized control strategies, the vehicles work in parallel, and each vehicle can solve its own local optimization problem regardless of other vehicles' problem formulations. Therefore, packet loss or communication delay for one vehicle have no effect on the other vehicles' local optimization problems. However, an unreliable solution for one vehicle can of course increase the collision probability of the entire system.

Therefore, it is crucial to evaluate how robust the **AIM** methods are to non-ideal wireless communication links. We performed simulations for two scenarios with non-ideal wireless communication links, in the following called *Scenario 1* and *Scenario 2*. In Scenario 1, the wireless links have a reliability of 99.999% (i.e. a packet loss probability of 0.001%) and no communication delays. In Scenario 2, the wireless links have a reliability of 97.5% and an average communication delay of 20 ms. Scenario 1 corresponds to the reliability for **C-V2X** 5G, Ultra Reliable Low Latency Communication (URLLC) [27], but ignoring the communication delays. Scenario 2 corresponds to the measurements in [18].

Figure 6 shows the traffic safety in terms of average number of collisions per hour (calculated as before) for the different **AIM** methods in these two scenarios. Our results confirm that non-ideal wireless communication links can significantly reduce the reliability of **AIM** methods based on centralized and decentralized control strategies. However, in our proposed **AIM** method based on a hierarchical control strategy, the Local Decentralized Layer can always rely on its own local data if it has not received a new reference

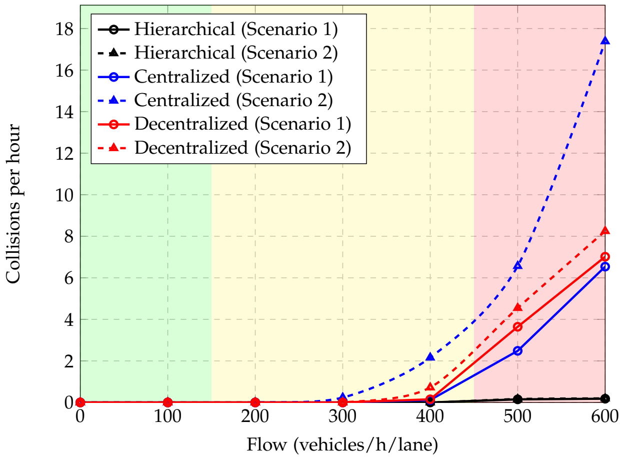

Figure 6: Average number of collisions per hour for different traffic flow rates using non-ideal wireless communication links

speed from the **GCL**. This means that the **LDL** can find its own collision-free movements in case of lost messages. Therefore, reasonable packet loss and communication delays, due to the characteristics of the wireless communication links in vehicular environments, have minimum effects on the traffic safety of our proposed method.

5.5 SCALABILITY

Scalability is another crucial requirement for operational **AIM** systems. An **AIM** method must be scalable in order to safely handle a large number of vehicles crossing the intersection during, for example, peak hour.

To evaluate and compare the **AIM** methods' scalability, Figure 7 shows the maximum traffic flow rate that each **AIM** method can safely handle. The figure shows the maximum traffic flow rate where the average number of collisions per hour is less than 0.2. As before, the calculation of the average number of collisions per hour is an approximation, and it is the comparison that is interesting, not the absolute values.

As shown in Figure 7, our proposed **AIM** method is safe also for High traffic volumes. However, the **AIM** methods based on centralized and decentralized control strategies can ensure a safe intersection only for Low and Medium traffic volumes.

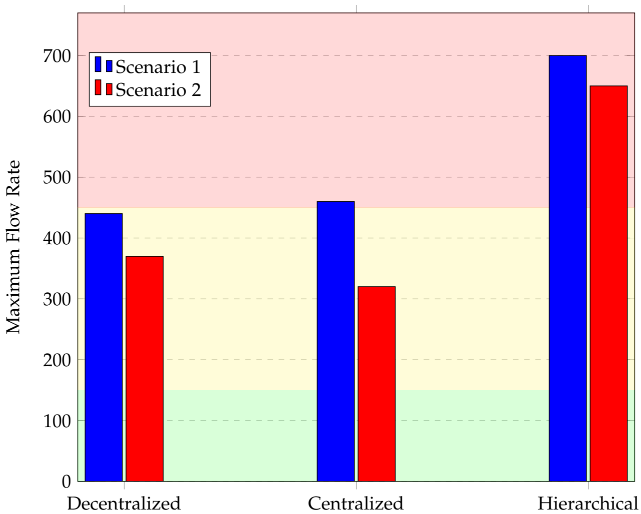

Figure 7: Maximum traffic flow rate in order to have less than 0.2 collisions per hour

5.6 EFFECT OF SYSTEM PARAMETERS

In our evaluations, we have used some parameters, Table II, which is common in **AIM** methods based on **MPC** [28], for example, a Target speed, v_d^t, of 16 m/s and minimum separation distance, that is the minimum space between two vehicles, R_{min} and d_{min}, of around 6-7 m. However, there are not really any good arguments for why these parameters should have exactly these values. Therefore, in this section we evaluate the system performance when varying the Target speed and the minimum separation distance between vehicles.

5.6.1 Target speed

First, we evaluated the system performance when varying the Target speed. We used three different Target speeds: (1) 30 km/h, representing a low Target speed; (2) 57 km/h (the Target speed we used in our experiments above), representing a medium Target speed; and (3) 72 km/h, representing a high Target speed (the speed limit of the intersection). In all three cases, the traffic flow rate was set to 700 vehicles/hour/lane.

The results of our simulations are summarized in Table III. As shown in Table III, the average number of vehicles that coexist within the intersection, $\bar{N}$, is reduced with a higher Target speed, since vehicles can cross the intersection quicker. So, a higher Target speed means a higher system capacity. However, the safety will decrease when the Target speed increases. This is mainly

Table III: Effect of Target speed

v_d^t	$\bar{N}$	$\bar{N}_c$
$30km/h$	30	0
$57km/h$	14	0.2
$70km/h$	10	1

due to the safety distance between vehicles that is used in the optimization algorithms, and the controlled variable, the acceleration. The safety distance is set in meters, which corresponds to a shorter time period for higher speeds. An error in the control algorithm will therefore with a higher probability result in a collision. Also, the controlled variable, the acceleration, will have the same maximum and minimum values, irrespective of the Target Speed. This means that for higher speeds, the ability for the algorithm to affect the speed of vehicles by controlling the acceleration will be reduced, which also may introduce errors.

5.6.2 Minimum separation distance

Second, we evaluated the system performance with larger minimum separation distance in order to increase the vehicles' reaction time in higher speed. We repeated the simulation with a Target speed of 57 km/h and a traffic flow rate of 700 vehicles/hour/lane. However, in this case the minimum separation distance in the **GCL** was set to 15 m, while the **LDL** still used a minimum separation distance of $6m$. We compared the safety and speed changes of our proposed **AIM** method with the **AIM** using centralized and decentralized control strategies for Scenario 1 in Section 5.4.

The results of our simulations are summarized in Table IV. From Equation (10), the saturation flow rate in the intersection can be calculated to around 38000 vehicles/hour/lane, which is much higher than 700 vehicles/hour/lane, the traffic flow rate used in the simulation. Therefore, we expect a smooth flow of vehicles without congestion. As shown in Table IV, the average number of vehicles that coexist within the intersection, $\bar{N}$, increased compared to the simulations with a lower minimum separation distance (about 14 vehicles), but it is still much lower than number of vehicles in a saturated intersection, around 80 for a minimum separation distance of $15m$, and also for a traditional intersection controlled by traffic lights, Figure 3. As expected, the safety is increased for all three **AIM** methods. A higher minimum separation distance corresponds to a longer time period for the algorithm to avoid collisions. The collisions in the **AIM** methods with

Table IV: Effect of minimum separation distance

AIM method	$\bar{N}$	$\bar{N}_c$
Hierarchical	17	0
Centralized	17	3
Decentralized	17	3.5

centralized and decentralized control strategies are mainly due to that the controller cannot always find a solution that keeps a minimum separation distance of $15m$ for all vehicles inside the intersection, and therefore, the algorithms fail in some cases. But in our proposed **AIM** method based on a hierarchical control strategy, the **LDL** can avoid collisions, even when the **GCL** cannot find a solution.

6 CONVOLUTION

In this paper we propose a new **AIM** method based on a hierarchical control strategy for optimal coordination of automated vehicles at intersections. Our proposed **AIM** method consists of a Global Centralized Layer where a centralized control algorithm allocates an reference speed to each vehicle by solving an **MPC** optimization problem, and a Local Decentralized Layer where a decentralized control algorithm is responsible for following the allocated speed and avoid collisions. In our proposed **AIM** method, each vehicle uses its own sensors to get local information from its close surroundings. Therefore, our proposed **AIM** method will not require any **V2V** communication, and each vehicle can take its own decisions on its movements without receiving control directions from the **GCL**. Our simulation results, performed in the realistic simulation environment **SUMO**, clearly shows that our proposed **AIM** method is safe, scalable, and robust to packet loss and delay caused by wireless communication.

Further, we showed the effects of changing the values of two important system parameters, Target speed and minimum separation distance. We showed that the values of these two parameters can have a large effect on the safety of the system. Therefore, one way to increase the safety and robustness of the **AIM** system, is to include a mechanism that selects optimal Target speed and minimum separation distance, based on the current traffic flow rate and the current environment (for example, the weather). For example, in situations with low traffic flow rates and good weather, the Target speed can be set higher than in situations with high traffic flow rate is and bad weather. An adaptive system will probably be crucial to guarantee a totally safe **AIM** system with maximum capacity. However, this topic will remain for future research.

References

[1] J. C. Dovilé Adminaité-Fodor and G. Jost, “15th road safety performance index report,” The European Transport Safety Council (ETSC), Tech. Rep., June 2021.

[2] R. Hult, G. R. Campos, E. Steinmetz, L. Hammarstrand, P. Falcone, and H. Wymeersch, “Coordination of cooperative autonomous vehicles: Toward safer and more efficient road transportation,” *IEEE Signal Processing Magazine*, vol. 33, no. 6, pp. 74–84, 2016.

[3] L. Chen and C. Englund, “Cooperative intersection management: A survey,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 2, pp. 570–586, 2015.

[4] K. Dresner and P. Stone, “A multiagent approach to autonomous intersection management,” *Journal of artificial intelligence research*, vol. 31, pp. 591–656, 2008.

[5] J. Lee and B. Park, “Development and evaluation of a cooperative vehicle intersection control algorithm under the connected vehicles environment,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 1, pp. 81–90, 2012.

[6] M. Bashiri and C. H. Fleming, “A platoon-based intersection management system for autonomous vehicles,” in *2017 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2017, pp. 667–672.

[7] W. Zhao, R. Liu, and D. Ngoduy, “A bilevel programming model for autonomous intersection control and trajectory planning,” *Transportmetrica A: transport science*, vol. 17, no. 1, pp. 34–58, 2021.

[8] S. Chamideh, W. Tärneberg, and M. Kihl, "Centralized coordination of autonomous vehicles at intersections," in *28 th International Conference on Software, Telecommunications and Computer Networks (SoftCOM 2020)*. Institute of Electrical and Electronics Engineers Inc., 2020.

[9] X. Liang, T. Yan, J. Lee, and G. Wang, "A distributed intersection management protocol for safety, efficiency, and driver's comfort," *IEEE internet of things journal*, vol. 5, no. 3, pp. 1924–1935, 2018.

[10] A. Mirheli, M. Tajalli, L. Hajibabai, and A. Hajbabaie, "A consensus-based distributed trajectory control in a signal-free intersection," *Transportation research part C: emerging technologies*, vol. 100, pp. 161–176, 2019.

[11] A. Britzelmeier and M. Gerdts, "Non-linear model predictive control of connected, automatic cars in a road network using optimal control methods," *IFAC-PapersOnLine*, vol. 51, no. 2, pp. 168–173, 2018.

[12] A. Katriniok, P. Kleibaum, and M. Joševski, "Distributed model predictive control for intersection automation using a parallelized optimization approach," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 5940–5946, 2017.

[13] S. Chamideh, W. Tärneberg, and M. Kihl, "Evaluation of decentralized algorithms for coordination of autonomous vehicles at intersections [unpublished]," in *24th IEEE International Conference on Intelligent Transportation (ITSC2021)*. IEEE, 2021.

[14] R. Hult, M. Zanon, G. Frison, S. Gros, and P. Falcone, "Experimental validation of a semi-distributed sequential quadratic programming method for optimal coordination of automated vehicles at intersections," *Optimal Control Applications and Methods*, vol. 41, no. 4, pp. 1068–1096, 2020.

[15] M. Khayatian, M. Mehrabian, and A. Shrivastava, "Rim: Robust intersection management for connected autonomous vehicles," in *2018 IEEE Real-Time Systems Symposium (RTSS)*. IEEE, 2018, pp. 35–44.

[16] R. Hult, M. Zanon, S. Gros, and P. Falcone, "Optimal coordination of automated vehicles at intersections: Theory and experiments," *IEEE Transactions on Control Systems Technology*, vol. 27, no. 6, pp. 2510–2525, 2018.

[17] "Driving 5g innovation for urban public transport," September, 2020. [Online]. Available: https://www.ericsson.com/en/cases/2021/5g-ride-driving-innovation-for-urban-public-transport

[18] L. Miao, J. J. Virtusio, and K.-L. Hua, "Pc5-based cellular-v2x evolution and deployment," *Sensors*, vol. 21, no. 3, p. 843, 2021.

[19] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner, "Microscopic traffic simulation using sumo," in *The 21st IEEE International Conference on Intelligent Transportation Systems*. IEEE, 2018. [Online]. Available: <https://elib.dlr.de/124092/

[20] S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control engineering practice*, vol. 11, no. 7, pp. 733–764, 2003.

[21] B. Jacob and E. Violette, "Vehicle trajectory analysis: an advanced tool for road safety," *Procedia-Social and Behavioral Sciences*, vol. 48, pp. 1805–1814, 2012.

[22] R. Jones, "Quantitative effects of acceleration rate on fuel consumption. technical report," Environmental Protection Agency, Ann Arbor, MI (USA), Tech. Rep., 1980.

[23] E. Aoyama, K. Yoshioka, S. Shimokawa, and H. Morita, "Estimating saturation flow rates at signalized intersections in japan," *Asian Transport Studies*, vol. 6, p. 100015, 2020.

[24] J. Zheng and H. X. Liu, "Estimating traffic volumes for signalized intersections using connected vehicle data," *Transportation Research Part C: Emerging Technologies*, vol. 79, pp. 347–362, 2017.

[25] R. Weber, J. Misener, and V. Park, "C-v2x-a communication technology for cooperative, connected and automated mobility," in *Mobile Communication-Technologies and Applications; 24. ITG-Symposium*. VDE, 2019, pp. 1–6.

[26] J. Zagajac, "The c-v2x proposition," in *5GAA Washington DC Workshop*, 2018.

[27] D. Garcia-Roger, E. E. González, D. Martín-Sacristán, and J. F. Monserrat, "V2x support in 3gpp specifications: From 4g to 5g and beyond," *IEEE Access*, vol. 8, pp. 190 946–190 963, 2020.

[28] M. A. S. Kamal, J.-i. Imura, T. Hayakawa, A. Ohata, and K. Aihara, "A vehicle-intersection coordination scheme for smooth flows of traffic without using traffic lights," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 3, pp. 1136–1147, 2014.