Cooperative Automated Vehicle Movement Optimization at Uncontrolled Intersections using Distributed Multi-Agent System Modeling

Abdallah A. Hassan Mahmoud

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

Doctor of Philosophy

In

Computer Engineering

Hesham A. Rakha, (Chair)
Cameron Patterson
Jung-Min Park
Haibo Zeng
Hao Chen

Dec 14th 2016
Blacksburg, Virginia

Keywords: ITS, Automated Vehicles, Uncontrolled Intersections, Networked Intersections, Multi-agent Systems, Distributed Systems, Intelligent Agent Cooperation

Copyright 2016, Abdallah A. Hassan Mahmoud
Cooperative Automated Vehicle Movement Optimization at Uncontrolled Intersections using Distributed Multi-Agent System Modeling

Abdallah A. Hassan Mahmoud

ABSTRACT

Optimizing connected automated vehicle movements through roadway intersections is a challenging problem. Traditional traffic control strategies, such as traffic signals are not optimal, especially for heavy traffic. Alternatively, centralized automated vehicle control strategies are costly and not scalable given that the ability of a central controller to track and schedule the movement of hundreds of vehicles in real-time is highly questionable. In this research, a series of fully distributed heuristic algorithms are proposed where vehicles in the vicinity of an intersection continuously cooperate with each other to develop a schedule that allows them to safely proceed through the intersection while incurring minimum delays. An algorithm is proposed for the case of an isolated intersection then a number of algorithms are proposed for a network of intersections where neighboring intersections communicate directly or indirectly to help the distributed control at each intersection makes a better estimation of traffic in the whole network. An algorithm based on the Godunov scheme outperformed optimized signalized control. The simulated experiments show significant reductions in the average delays.

The base algorithm is successfully added to the INTEGRATION micro-simulation model and the results demonstrate improvements in delay, fuel consumption, and emissions when compared to roundabout, signalized, and stop sign controlled intersections. The study also shows the capability of the proposed technique to favor emergency vehicles, producing significant increases in mobility with minimum delays to the other vehicles in the network.
Cooperative Automated Vehicle Movement Optimization at Uncontrolled Intersections using Distributed Multi-Agent System Modeling

Abdallah A. Hassan Mahmoud

GENERAL AUDIENCE ABSTRACT

Intelligent self-driving cars are getting much closer to reality than fiction. Technological advances make it feasible to produce such vehicles at low affordable cost. This type of vehicles is also promising to significantly reduce car accidents saving people lives and health. Moreover, the congested roads in cities and metropolitan areas especially at rush hours can benefit from this technology to avoid or at least to reduce the delays experienced by car passengers during their trips.

One major challenge facing the operation of an intelligent self-driving car is how to pass an intersection as fast as possible without any collision with cars approaching from other directions of the intersection. The use of current traffic lights or stop signs is not the best choice to make the best use of the capabilities of future cars.

In this dissertation, the aim is to study and propose ways to make sure the future intersections are ready for such self-driving intelligent cars. Assuming that an intersection has no type of traditional controls such as traffic lights or stop signs, this research effort shows how vehicles can pass safely with minimum waiting. The proposed techniques focus on providing low-cost solutions that do not require installation of expensive devices at intersections that makes it difficult to be approved by authorities. The proposed techniques can be applied to intersections of various sizes.

The algorithms in this dissertation carefully design a way for vehicles in a network of intersections to communicate and cooperate while passing an intersection. The algorithms are extensively compared to the case of using traffic lights, stop signs, and roundabouts. Results show significant improvement in delay reduction and fuel consumption when the proposed techniques are used.
ACKNOWLEDGEMENTS

The research effort presented in this dissertation is funded by the Connected Vehicle Initiative University Transportation Center (CVI-UTC). This work could not have been completed without the help of many people in Virginia Tech and Virginia Tech Transportation Institute especially in the center for sustainable mobility.

I am so grateful to Prof. Hesham Rakha who advised this research for his generous support throughout the course of this research. I acknowledge the help provided by Yousef Bichiou in the MATLAB implementation of the vehicle dynamics model and the implementation of the Godunov scheme equations. I also acknowledge the help provided by Hao Yang in implementing the DIVC system in the INTEGRATION model. I am also grateful to Mohamed Almanna for the help with the simulation experiments of the optimized signalized control in INTEGRATION. I am also grateful to Mohammed Elhenawy for his help with the statistical analysis and interpretation of results and for his help while I was preparing some of the figures of this dissertation. I am also grateful to Ahmed Roman for help with some of the mathematical formulation used in this dissertation.
# Table of Contents

Chapter 1 .................................................................................................................. 1
  Introduction and Motivation ..................................................................................... 1
    1.1 Challenges for Intelligent Transportation Systems ........................................... 1
    1.2 Automated Vehicles ....................................................................................... 2
    1.3 Vehicle-To-Vehicle Communication ............................................................... 4
    1.4 Problem Complexity ....................................................................................... 5
Chapter 2 ..................................................................................................................... 7
  Related Work ............................................................................................................ 7
    2.1 Introduction ...................................................................................................... 7
    2.2 Centralized Methods ...................................................................................... 7
    2.3 Distributed Methods ...................................................................................... 11
    2.4 Multi-intersection Methods ........................................................................... 13
    2.5 Adaptive Signal Control ............................................................................... 14
Chapter 3 .................................................................................................................... 15
  A Fully-Distributed Heuristic Algorithm for Control of Automated Vehicle Movements at Isolated Intersections ............................................................... 15
    Abstract ................................................................................................................ 15
    3.1 Introduction ..................................................................................................... 16
    3.2 Related Work .................................................................................................. 18
    3.3 Proposed Algorithm ...................................................................................... 21
      3.3.1 Overview ................................................................................................. 21
      3.3.2 Modeling Agent States .......................................................................... 24
      3.3.3 Scheduling Algorithm .......................................................................... 26
      3.3.4 Communication Requirements .............................................................. 28
    3.4 Simulated Experiments and Results ............................................................... 29
      3.4.1 Experiment Set 1 ................................................................................... 29
      3.4.2 Experiment Set 2 ................................................................................... 31
    3.5 Conclusions and Future Recommendations ................................................... 33
Chapter 4 .................................................................................................................... 34
  Two Algorithms for Traffic Optimization of Automated Vehicles at Multi-Intersection Networks .......................................................... 34
    Abstract ................................................................................................................. 34
### 7.4.1 One-Lane All Through Traffic

84

### 7.4.2 Two-Lane All Through Traffic

89

### 7.4.3 One-Lane Through and Right Turns Allowed

92

### 7.5 Conclusions and Future Recommendations

95

### Chapter 8

97

### Conclusions and Recommendations for Future Work

97

#### 8-1 Summary of study objectives

97

#### 8-2 Conclusions

98

#### 8.3 Future Plans and Recommendations for Further Research

100

### References

103
List of Figures

Figure 1: Agent State Diagram ........................................................................................................25
Figure 2: Average Delay of Proposed Algorithm Compared to FIFO ........................................30
Figure 3: Maximum Delay of Proposed Algorithm Compared to FIFO .......................................31
Figure 4: Comparing Average Delay of Two Methods with Heavy Traffic ................................32
Figure 5: Comparing Maximum Delay of Two Methods with Heavy Traffic ..........................32
Figure 6: VEHICLE STATE DIAGRAM FOR IIZA ............................................................39
Figure 7: A 4x4 GRID NETWORK OF INTERSECTIONS ..................................................43
Figure 8: ROAD SLICES BETWEEN TWO INTERSECTIONS ..............................................45
Figure 9: NIZA COMPARED TO IIZA ...................................................................................48
Figure 10: DLA COMPARED TO IIZA ..................................................................................49
Figure 11: DLA COMPARED TO NIZA AND IIZA .............................................................49
Figure 12: AVERAGE AND MAXIMUM DELAYS OF IIZA, NIZA, AND DLA ..................50
Figure 13: Road Discretization for Godunov Scheme .........................................................58
Figure 14: Schematics of a vehicle in motion along a path ......................................................60
Figure 15: The traffic entering and leaving the network .........................................................63
Figure 16: THE TWO PHASES FOR EACH INTERSECTION ................................................65
Figure 17: Average delay VS cycle length for major and minor streets ..................................66
Figure 18: DLAG COMPARED TO SIGNALIZED CONTROL ...........................................67
Figure 19: Delay of 20% class H vehicles ..............................................................................76
Figure 20: Delay of 80% class L vehicles ..............................................................................77
Figure 21: Mean Delay for Through Traffic in One-Lane Intersection ...................................85
Figure 22: Mean Delay for Through Traffic, One Lane (No Stop Sign) .................................86
Figure 23: Mean Fuel Consumption for Through Traffic, One Lane ..................................87
Figure 24: Mean Fuel Consumption for Through Traffic, One Lane, No Stop Sign ............88
Figure 25: Mean Emissions for Through Traffic, One Lane Intersection ............................88
Figure 26: Mean Stops for Through Traffic, One Lane Intersection .....................................89
Figure 27: Mean Delay for Through Traffic, Two-Lane Intersection .....................................91
Figure 28: Mean Fuel Consumption for Through Traffic, Two-Lane Intersection ............91
Figure 29: Mean Emissions for Through Traffic, Two-Lane Intersection ...........................92
Figure 30: Mean Stops for Through Traffic, Two-Lane Intersection ....................................93
Figure 31: Mean Delay (Through and Right Traffic) (Two-Lane) ........................................94
Figure 32: Mean Fuel Consumption (Through and Right) (Two-Lane) ...............................94
Figure 33: Mean CO2 Emissions (Through and Right Traffic) (Two-Lane) ......................95
Chapter 1

Introduction and Motivation

Transportation networks and transportation means greatly affect the quality of life for people worldwide. Congested roads in big cities at rush hours make it difficult for many people to use their time efficiently without wasting their energy. The current vehicles’ increased emissions and fuel consumption lead to more damage to the environment and have a huge negative impact on health and development. The status quo in many big cities reflects what seems to be a chronic problem. Current challenges in transportation are not only threatening the quality of life, but also creating a serious issue that can be seen in the number of fatalities caused by car accidents.

1.1 Challenges for Intelligent Transportation Systems

The two major challenges for current transportation systems are mobility and safety. Current figures of fatal accidents and of losses due to congestion reflect the size of the problem. According to the World Health Organization, 1.24 million deaths are caused by car accidents annually around the world [1]. The congestion on U.S. roads accounts for an estimated cost of $121 billion every year, with a total travel delay of 5.5 billion wasted hours on congested roads, wasting 2.9 billion gallons of gas, and emitting 56 billion pounds of environmentally damaging carbon dioxide [2]. Currently, there is roughly 1 billion operating vehicles around the world [3]. This number is predicted to increase to 4 billion vehicles by the middle of the century [4]. It is clear that current transportation systems will fail unless innovative solutions are proposed to make better use of roads.
Intersections represent a major cause of both congestion and accidents. For example, in U.S., 40% of accidents occur at intersections [5]. This is one of the main motives for the research presented in this dissertation to focus on the traffic management optimization around intersections. The focus on uncontrolled intersections is based on the fact that most intersections are not signalized, and, with the advent of automated vehicles, it is not economically sound to add traffic lights to many of the currently non-signalized intersections. For example, according to the US Department of Transportation (DOT), less than 10% of intersections in the U.S. are signalized. Depending on its geometry and complexity, the average cost of signalizing an intersection is around $65000 [6]. Proposing algorithms that can control the traffic in the absence of a traffic light is much more cost effective.

This dissertation proposes a number of algorithms that assume the availability of two promising technologies: automated vehicles and vehicle-to-vehicle (V2V) communication. Since both technologies are not yet widespread, the following subsections discuss why it is believed that both technologies are quickly becoming more common.

1.2 Automated Vehicles

Automated vehicles can result in a notable improvement in transportation safety. Currently, half of the accidents in the U.S. are caused by inadequate decision-making by drivers. Globally, the International Organization for Road Accident Prevention estimates driver errors cause 90% of car accidents in the world [7]. Aside from its possible effect on traffic safety, automated vehicles can also enable drivers to manage their time more efficiently during peak hour traffic jams. More importantly, the widespread use of such vehicles opens doors for new techniques of traffic management that can make much better use of current roads and transportation systems. The
question is: Are automated vehicles really feasible? Furthermore, how long will it take for them to be commonly used by the public?

Computational capabilities of vehicles have seen a rapid increase in the past decade. A typical vehicle in the early 2000s had 20 to 100 individual networked microcontroller/PLC modules with a non-real-time operating system [8]. The current trend used in vehicle manufacturing is to use fewer, more costly microprocessor modules with hardware memory management and real-time operating systems. Newly developed embedded system platforms allow typical vehicles to run sophisticated software applications such as model-based process control, artificial intelligence applications, and ubiquitous computing [8].

Self-driving vehicles are much closer to reality than fiction. The famous Google car travelled more than 200,000 miles around California [9]. Automated vehicles are already licensed in Nevada and Florida. For a long time, one main problem for automated vehicles was the high cost of sensing devices. For example, the cost of Google’s self-driving car is around $75000 [10]. Recently, different research groups have proposed self-driving cars built using low-cost sensor devices such as 3D radars and mounted cameras. These cheaper vehicles depend mainly on artificial intelligence to detect lanes and curbs and to avoid collisions. One example is RobotCar [11], an automated vehicle built by Oxford university researchers that costs only $7500. The cost reduction associated with automated vehicles suggests that the mass production of such vehicles may emerge very quickly. A survey by McGraw Hill Financial Inc. [12] revealed that 39% of surveyed people are interested in owning an automated vehicle. In addition, 21% are still interested if the cost is $3000 or more. The survey also shows that there is increased interest in semi-automated features, such as electronic parking assistance and emergency braking and steering.
Even if mass production of automated vehicles is delayed, the distributed traffic management algorithms proposed in this dissertation can still be applicable. Vehicles equipped with Cooperative Adaptive Cruise Control (CACC) can greatly improve traffic optimization.

1.3 Vehicle-To-Vehicle Communication

Wireless communication is used to exchange information between vehicles (V2V), between vehicles and infrastructure units (road-side devices) (V2I), and between infrastructure units (I2I). UHF and VHF frequencies are widely used for short and long range communication within ITS. IEEE 802.11 protocols are used for short-range communication. Long range communication uses infrastructure networks such as WiMAX (IEEE 802.16), GSM, or 3G.

Although V2V is not yet widespread, it is regarded as the next generation of vehicle safety improvements after safety belts and airbags. It is predicted that all vehicles in the U.S. will be required to be V2V enabled within a few years. Therefore, the research proposed in this dissertation assumes that building distributed management on V2V communication is a reasonable choice. Refer to [13] for a detailed survey of inter-vehicle communication protocols.

The objectives of this research effort are to:

- Optimize automated vehicle movements through uncontrolled intersections.
- Propose a fully-distributed control relying on the cooperation between the connected vehicles around the intersection.
- Allow vehicles at an intersection to know about and make use of the traffic status at neighboring upstream intersections to better manage traffic.
- Use a heuristic approach to ensure the computational requirement of the control system fits within the limits of real-time operation.
1.4 Problem Complexity

This section demonstrates the complexity of the research problem and why a heuristic approach is reasonable if real-time operation is desired, given the extreme computational expense to search for a global optimal solution. The traffic optimization problem at an intersection can be viewed as a scheduling problem where the objective is to search for a schedule dictating the order by which existing vehicles in the neighborhood of the intersection should be permitted to pass the intersection conflict area. We can evaluate a candidate schedule based on performance objectives, including the average delay of vehicles until they pass the intersection. A derivation of the size of the solution space for a small four-lane intersection is provided in the remainder of this section.

To start, we assume we have only two perpendicular lanes with \( m_1 \) and \( m_2 \) vehicles, respectively. We attempt to determine the number of ways to schedule the vehicles to pass the intersection, given that the order of the vehicles in each lane should be preserved. Suppose we have two lanes with vehicles \( x_1 \) and \( x_2 \) in the first lane, and \( y_1, y_2, y_3 \) in the second lane. We are interested in rearrangements of the type \( x_1, y_1, y_2, x_2, y_3 \) where \( x_2 \) cannot pass the intersection before \( x_1 \) and similarly for \( y_1, y_2, y_3 \). To do this, we assume an initial schedule of a total number of slots \( m_1 + m_2 \). For \( m_1 + m_2 \) vehicles, there are \( (m_1 + m_2)! \) possible rearrangements. However, because vehicles in each lane need to appear in order in the schedule, for the \( m_1 \) vehicles in the first lane, we need to divide the number of rearrangements by the number of permutations of first lane vehicles which is \( m_1! \). After placing the first lane’s vehicles in the schedule, we will have \( m_2 \) spots left for the second lane’s vehicles and there will be only one way to place them to preserve their order. Therefore, the number of possible schedules is:

\[
\binom{m_1 + m_2}{m_1}
\]
The case of three lanes with \(m_1, m_2\) and \(m_3\) vehicles respectively, is similar. There are \((m_1 + m_2 + m_3)\)! ways of ordering the vehicles. For the vehicles in the first lane, we care only about their ordered arrangement within the \(m_1 + m_2 + m_3\) spots and, therefore, we divide by their possible number of rearrangements: \(m_1\)!. This results in:

\[
\binom{m_1 + m_2 + m_3}{m_1}
\]

Next, the \(m_2\) vehicles in the second lane are included in the schedule. The number of ways to complete this step is:

\[
\binom{m_2 + m_3}{m_2}
\]

Finally, the remaining \(m_3\) vehicles are forced in the remaining \(m_3\) spots to keep their order. So, for the three-lane case, the total number of orderings is

\[
\binom{m_1 + m_2 + m_3}{m_1} \binom{m_2 + m_3}{m_2}
\]

Assuming a four-lane intersection, the same derivation leads to a total number of schedules:

\[
\binom{m_1 + m_2 + m_3 + m_4}{m_1} \binom{m_2 + m_3 + m_4}{m_2} \binom{m_3 + m_4}{m_3}
\]

For example, assuming that each lane has 40 vehicles this generates a solution space in the order of \(10^{93}\) possible schedules.
Chapter 2
Related Work

2.1 Introduction

The objective of the work presented in this dissertation is to optimize the movement of connected automated vehicles at uncontrolled intersections. Various research efforts in literature targeted this problem using different methodologies [14]. The related algorithms proposed in literature can be classified into two broad categories. One category of methods uses centralized controls requiring that the intersection be equipped with a central agent to control the movement of vehicles and to ensure that requests to pass the conflict zone be accepted in a way that avoids any possible collisions and minimizes delays.

The other category of methods uses distributed control. Vehicles around an intersection negotiate and cooperate to guarantee that they pass the intersection without any collisions and achieve some optimization function such as delay reduction and avoidance of queue formations.

2.2 Centralized Methods

Recently, several researchers adopted the proposal of techniques for the traffic management of automated vehicle traffic at uncontrolled intersections. Dresner and Stone [15, 16] are the forerunners in this field and their work in particular is a foundation for the research in this dissertation. They proposed multi-agent modeling for uncontrolled intersections where an approaching vehicle communicates with a central controller agent at the intersection and requests to pass the intersection conflicting area at some specified time. The proposed control protocol called Automated Intersection Management (AIM) is a technique that handles any conflicting requests from vehicles coming from different approaches and wishing to pass the intersection at
the same time by time reserving the conflict areas based on a First-Come-First-Serve basis. Thus, AIM assumes reliable low-latency Vehicle-to-Intersection (V2I) communications are used to reserve space and time in the intersection.

Zhu et al. [17] proposed LICP, a Look-ahead Intersection Control Policy with intelligent vehicles, which uses a centralized reservation-based protocol similar to AIM. The proposed look-ahead control works as follows: an approaching vehicle sends a request to the central controller to reserve a time slot to pass the intersection conflict zone. The controller predicts the total delay if the request is accepted and the total delay if the request is postponed. Based on this, a decision is made to accept or reject the request. This strategy resulted in a 25% improvement in performance compared to the simple FIFO scheme.

Another important extension to the AIM approach is the control protocol proposed in [18]. To our knowledge, this is the only system in literature that accommodates human-driven, semi-automated, and fully-automated vehicles. It still builds on AIM as a centralized reservation-based protocol.

Zohdy and Rakha [19] proposed a centralized heuristic algorithm that uses a custom simulator OSDI used by the central controller agent. The controller uses the simulator to continuously adjust the vehicles’ trajectories to minimize the occupancy time in conflict areas. This algorithm was tested with a four-vehicle scenario and was found to reduce the delay compared to traditional stop-sign control.

Zohdy et al. [20-22] proposed a centralized control for vehicles equipped with cooperative adaptive cruise control, CACC, using a game theory framework. Their analysis considered mixed flows of automated and regular vehicles. The algorithm achieved a significant reduction in vehicle
delays compared to the use of traffic lights and stop signs. In the same context, Elhenawy et al. [23] proposed another game-theory-based algorithm for connected vehicles at uncontrolled intersections where the authors assumed that vehicles are equipped with CACC and that they will obey the Nash equilibrium solution of the game to avoid collisions. This is another centralized approach that successfully reduced travel time but the authors only compared their algorithm to stop sign control. Another game-theory-based algorithm was proposed by Arora et al. [24] where a two-player zero-sum game was used to model the collision avoidance between two vehicles at an intersection. The control action of the first vehicle represented the first player, whereas the velocity disturbance in the second vehicle is the second player.

Colombo et al. [25] modeled the crossing of vehicles at an intersection as a scheduling problem. Their solution worked by finding the maximal controlled invariant set and checking membership in this set using an algorithm that approximated the solution. This is then used to design a controller for collision avoidance. The limitation of [25] is due to the requirement of perfect state information and the absence of any disturbances. An improvement of this method was proposed by Bruni et al. [26] where a controller was designed to deal with imperfect state information and input uncertainties.

The work of Vasirani and Ossowski [27] inspired part of the work in this dissertation. They suggested an economic approach building on the AIM [16]. In order for the driver agents to reserve time slots to pass the intersection, the traffic is modeled as a computational economy. Thus, an incoming vehicle will not simply send a request to the controller to reserve time and space to pass the conflict area, but will actually purchase a reservation and pay a fare if the controller accepts the purchase request. Now the central control agent acts as a seller following a variable price function that depends on how far in advance a reservation purchase is requested and the use of the
intersection and traffic status. The driver agent must decide, based on its wealth, which route to take for the trip, which intersection manager agent to trade, and the maximum amount of money to pay. The proposed algorithm was evaluated by simulating traffic in part of the city of Madrid, and resulted in a travel time reduction of 28% and led to a greater spread of vehicles along different routes, which results in less congestion but also longer routes as the driver agents tend to choose cheaper, lengthier routes. Part of this dissertation studies the effect of applying a possible pay for service scheme to ensure that emergency vehicles or trips of expensive delay cost find a way across the network with minimal possible delays.

Concepts such as the economic modelling in [27] are expected to attract more research efforts and become applicable in real life very soon. The motive is the different valuations of waiting-time reduction of the drivers. An interesting proposed algorithm that is valuation-aware is the Initial Time-Slot Auction algorithm [28, 29]. The intersection agent initiates an auction of the next available time slots, and the vehicle agents in the neighborhood of the intersection who are willing to take part send their bids. The highest offer wins access to the intersection at the time slot offered. A vehicle that already acquired an earlier time slot cannot participate, and also a vehicle cannot take part in the auction until all vehicles downstream have acquired a time slot. The simulated results showed significant enhancement compared to a FIFO mechanism.

The feasibility of deploying algorithms such as [27-29] is expected to raise ethical and social concerns that have begun to attract research efforts [30, 31], and more is expected to come.

A very recent interesting algorithm proposed by Ripon et al. [32] also builds on the AIM concept [16]. Optimization of traffic intersection management is achieved using either a deterministic approach or a heuristic or stochastic approach. However, in [32], the authors provided an algorithm that can be viewed as a hybrid method combining both deterministic and
heuristic approaches. The objective is to allow the intersection manager to operate in real-time. An evolutionary algorithm which searches in each time step for a speed vector (speed value for each vehicle close to the intersection) that optimizes an objective is used. The authors employed a multi-objective approach in which, at each time step, the controller optimizes one distinct objective from the set of objectives.

2.3 Distributed Methods

All previous methods are examples of centralized control. Another category of proposed research in literature relies on distributed decentralized control. Middlesworth et al. [33] proposed an interesting decentralized peer-to-peer approach that the authors suggested can be suitable for small intersections with low traffic as a replacement for stop-signs. Their algorithm relies on V2V communication and sensing. In [33], a simple broadcast communication protocol is defined for exchanging messages between connected vehicles. The communication protocol defines two messages: claim and cancel. The claim message is broadcast by an agent to announce the intention to reserve a specific time to pass the intersection conflict zone, while the cancel message releases any currently held reservation by the agent.

Zheng et al. [34] proposed a cooperative distributed control algorithm that outperformed a fixed-timing signal control for a simple intersection of one lane in each direction. Their vehicle motion control algorithm requires a vehicle close to an intersection to exchange communication with other vehicles. When a vehicle receives the state of other vehicles, it examines possible collisions and partitions all involved vehicles into collision groups. Having two vehicles in one collision group means their trajectories intersect. Vehicles in the same collision group adjust their acceleration rates until each collision group is composed of a single vehicle. The algorithm relied
completely on V2V. Quality of the algorithm was examined considering imperfect communication, which is an interesting element of this research effort.

Mladenovic and Abbas [35] proposed an interesting priority-based approach for automated vehicles at intersections using agent-modeling. They defined a priority system where each individual agent can choose for itself a specific priority level for the trip. The authors claimed their system is a step toward confronting principles of social justice, and they criticized the use of the average delay as a performance metric that requires the controller to treat a trip to a grocery store in the same manner as a trip to the emergency room. The proposed algorithm required vehicle agents within cooperative self-organizing zone of the intersection to communicate with all other vehicles that are simultaneously in the zone on all other conflicting approaches. Vehicles exchange their priority levels and information on their trajectories. Each vehicle simultaneously calculates its own arrival time and the arrival time at each of the intersection conflict cells for all other vehicles in the cooperative self-organizing zone. After this zone, the vehicle agents enter a trajectory adjustment zone divided into a decelerating section followed by an accelerating section so that vehicles safely pass the intersection conflict cells.

Makarem and Gillet [36] proposed another distributed algorithm that depends on V2V communication between vehicles to ensure their passing at the intersection is free from collisions. The algorithm works by allowing every vehicle around the intersection to define a local optimization problem that it solves. Each vehicle is required to communicate with all other vehicles in the intersection zone to exchange required information for the optimization problem to be solved.

Khoury [37] proposed a unique distributed algorithm for automated vehicles at intersections. With no reliance on wireless communication, vehicles are able to use only use
sensing information to make localized access decisions that can successfully lead to collision free traffic at the intersection. Although the algorithm is less optimal than other proposed algorithms, the possibility of applying it with no V2V or V2I communication makes it appealing.

Guangquan et al. [38] proposed a rule-based distributed control system. A predefined set of rules are used to resolve any conflicts that may occur at the intersection. These rules are known to the vehicles. Whenever a vehicle approaches the intersection, it communicates with other vehicles through V2V communication and each vehicle applies the rules to avoid any conflicts.

Carlino et al. [39] studied the possibility of running auctions at each intersection to determine the order of passing of automated vehicles’ conflicting movements of. Bidding can be done quickly and seamlessly. To ensure fairness, a benevolent system agent bid is used to maintain a reasonable travel time for vehicles with low budgets.

2.4 Multi-intersection Methods

Literature research efforts mentioned in previous sections focused only on isolated intersections. It should be noted that traffic at an intersection is affected by the state of traffic at upstream neighboring intersections. Proposing solutions that incorporate a network of connected intersections is a challenging task, but it is currently attracting growing interest by researchers due to the expected enhancement in traffic management. Several researchers already proposed algorithms to consider a network of intersections. For example, Wuthishuwong and Traechtler [40] proposed a multi-intersection algorithm where traffic information is exchanged using infrastructure to infrastructure communication I2I. Their algorithm is based on the concept of the green wave, trying to maintain the continuous driving of vehicles while passing the intersections. Yan and Dridi [41] proposed a genetic algorithm approach designed to find an optimal or near-optimal vehicle passing sequence at each intersection in a multi-intersection network.
2.5 Adaptive Signal Control

Research in this dissertation focuses on the problem of uncontrolled intersections. One important related thread of research in intelligent transportation is the adaptive signal control [42-45], which assumes signalized intersections and aims at optimizing the signal timing to consider the traffic conditions around the intersection and to sustain the mobility of vehicles. An example system is MARLIN (Multi-agent Reinforcement Learning for Integrated Network) [46, 47]. This system uses game theory and artificial intelligence to teach a network of traffic lights how to adjust to traffic patterns in real-time. This multi-agent system depends on a central agent that controls the light timings. The success of this system in achieving a 40% delay reduction using the real data of downtown Toronto rush hour traffic deserves recognition as the tested network included 60 intersections. One issue is the cost of connecting the lights to the central controller, which is estimated to be between $20,000 and $40,000 per intersection.

The reason why this dissertation focuses on uncontrolled rather than signalized intersections is the fact that more than 90% of current intersections are not signalized. With expectations of the near deployment of partially-automated and/or fully-automated vehicles, there is a real need to ensure that current intersections are ready for intelligent vehicles to use. Adaptive signal control can only be applied to less than 10% of the current intersections, so it cannot be regarded as a comprehensive approach for handling the management of traffic at all intersections in the future. It is not cost-effective to propose a solution that requires a currently non-signalized intersection to be signalized. However, given the extensive research done in adaptive signal control, it should be considered that many ideas used for adaptive signal control can be adopted for uncontrolled intersections and vice versa.
Chapter 3

A Fully-Distributed Heuristic Algorithm for Control of Automated Vehicle Movements at Isolated Intersections

Abstract

Optimizing automated vehicle movements through roadway intersections is a challenging problem. It has been demonstrated in the literature that traditional traffic control strategies, such as traffic signals and stop signs control are inefficient, especially for heavy traffic demand levels. Alternatively, centralized automated vehicle control strategies are costly and not scalable given that the ability of a central controller to track and schedule the movement of hundreds of vehicles in real-time is highly questionable. Consequently, in this chapter a fully distributed algorithm is proposed where vehicles in the vicinity of an intersection continuously cooperate with each other to develop a schedule that allows them to safely proceed through the intersection while incurring minimum delays. Unlike other distributed approaches described in the literature, the wireless communication constraints are considered in the design of the control algorithm. Specifically, the proposed algorithm requires vehicles heading into an intersection to communicate only with neighboring vehicles, while the lead vehicles on each approach lane share information to develop a complete intersection utilization schedule. The scheduling rotates between vehicles to identify higher traffic volumes and favor vehicles coming from heavier lanes to minimize the overall intersection delay. Compared to other methods reported in the literature, the simulated experiments using the proposed approach show significant reductions in the average delay and reductions in the maximum delays experienced by a vehicle, especially in cases of heavy traffic demand levels.
3.1 Introduction

Intelligent transportation systems (ITSs) play an essential role in the optimization of mobility and safety of transportation systems. Two major challenges are facing transportation engineers and ITS researchers.

1- How to manage traffic to avoid or at least to lessen congestion and minimize the wasted delay incurred by commuters? Congestion is currently a real headache in several metropolitan areas around the world. According to Texas Transportation Institute’s (TTI’s) 2012 urban mobility report [48], the congestion on U.S. roads accounted for an estimated cost of $121 billion in 2011, with a total travel delay of 5.5 billion wasted hours in congested roads, wasting 2.9 billion gallons of gas, and emitting 56 billion pounds of environmental harmful carbon dioxide. Although the accuracy of these values and figures has been severely criticized [49], there is no doubt that the congestion problem deserves more consideration and more innovative solutions to decrease the current losses in time, fuel, and environmental effects.

2- How to avoid or minimize vehicle crashes and make transportation less life threatening? According to the World Health Organization, 1.24 million deaths are caused by vehicle crashes every year. The International Organization for Road Accident Prevention estimates that 90% of accidents are due to human drivers’ mistakes. In the U.S., it is believed that more than half of car accidents are caused by inadequate decisions by drivers.

The two challenges may worsen in the near future due to the gap between the growth of the vehicle population and the expansion of infrastructure and addition of new roads or lanes. The
number of operating vehicles in the world exceeded one billion vehicles in 2010, and close to one quarter of those are running on U.S. roads [50]. Researchers and authorities should keep in mind that, due to continued population growth, economic growth, and urbanization, it is expected that the global vehicle population will increase in the middle of the century to be somewhere between 2.5 billion and 4 billion vehicles [4]. Currently, the question being asked and researched is how to enhance transportation system operations and ensure that transportation systems continue to function given that the current transportation infrastructure might not be expandable or might be expandable only at very high cost in many areas world-wide.

Traffic at intersections contributes significantly to both congestion and safety problems. The work proposed in this chapter focuses on the management of traffic at intersections. Assuming a system of automated vehicles, the proposed algorithm models the traffic as a multi-agent system using a fully-distributed protocol where vehicles coordinate to pass through the intersection without collision and with minimum delay at the intersection. This is not only a challenge for automated vehicles, but also for our current transportation system. Studies show that about 40% of car accidents in the United States are related to intersections, and about half of these accidents are caused by inadequate decision-making by drivers.

The decision to focus on algorithms based on automated vehicles is reasonable. Automated self-driving vehicles are closer to reality than to fiction. These automated cars have been demonstrated by Google and other academic institutions, such as the Technical University of Braunschweig and, more recently, the University of Oxford. It is believed that the current generation of hardware can be used to produce such automated vehicles [18]. The rapid progress in research of how to build and operate such intelligent vehicles resulted in their being licensed in the state of Nevada.
However, the researchers in this field are facing challenging issues and problems that need to be solved before such vehicles can find their way to mass production. Among the main challenges are: how to reduce the high cost of sensing devices used by these vehicles, and how to guarantee safety in any possible road, weather, or traffic conditions. Additionally, several researchers are working on an assessment of how current transportation will be enhanced or optimized when vehicles become smart and automated. Studies are evaluating various aspects, including minimizing fuel consumption, minimizing delays, and avoiding congestions.

The choice to propose a fully-distributed protocol is also reasonable. In a few years, vehicle-to-vehicle (V2V) communication will be mandated by the U.S. Department of Transportation, as it is viewed to represent the next generation of vehicle safety improvements [51]. It is considered to be the next step after safety belts and air bags. Vehicles will talk to each other and exchange trajectory data. It is expected that this will significantly reduce accidents and crashes, especially those caused by human error, as well as alleviate traffic congestion. A protocol requiring vehicles at an intersection to continuously exchange coordination messages, like the one proposed in this chapter, will soon be easily implemented in real vehicles on the roads.

3.2 Related Work

Various approaches have been used by several researchers to handle the problem of optimizing the movement of vehicles at intersections. For example, several researchers studied how to minimize the delay at intersections while keeping the current traffic light control system and human-driven vehicles. These research efforts attempted to make the traffic signals smart enough to change their timings to adapt to traffic conditions. An example of a successful algorithm is MARLIN [46, 47]. This algorithm used reinforcement learning to teach a network of traffic signals how to adapt to traffic patterns in real-time. This method was able to reduce the delay in a
network of sixty intersections in downtown Toronto by 40% at peak hours. However, it relied on a central agent connected to all traffic signals to control their timing; this makes scalability of the system questionable when dealing with larger networks. In addition, maintaining traffic signals and making them smarter is an expensive option. For example, in the case of the MARLIN system, it is estimated that additional equipment would be required at each intersection at a cost between twenty and forty thousand dollars [47].

Similar to the approach presented in this chapter, several researchers studied the traffic management at intersections for automated vehicles. Traffic signals are not used and vehicles depend solely on communication to coordinate their passing through the intersection without collisions. There are two approaches here: depending on a central controlling agent or using a distributed decentralized approach, where vehicles are the only agents and without any agents representing the infrastructure in the control system.

One of the key research efforts is the work of Dresner and Stone [16, 52] where a centralized multi-agent reservation-based intersection control protocol was presented. The protocol simply depended on a central management agent that used a First-In-First-Out (FIFO) method to reserve time slots for vehicle agents requesting to pass through the intersection. The importance of the work of Dresner and Stone is that it provided a simple feasible approach and showed how it significantly reduced the delay experienced by traditional traffic signals. Other researchers built on their work. For example, Zhu et al. [17] also used a centralized reservation-based protocol LICP, but instead of using the FIFO method, they proposed a look-ahead control policy. With this policy, when a vehicle requests to reserve a time slot, the controller agent predicts the total delay if the request is allocated, and the total delay if it is postponed, and based on these two predicted values the controller makes its decision. The use of LICP can achieve up to a 25%
performance improvement over the FIFO scheme. However, the reliance on a central agent for traffic control can create a bottleneck when the number of vehicles in the vicinity of an intersection increases.

Au et al. [18] built on the work of Dresner and Stone in a different way. Instead of limiting the system to automated vehicles, they proposed a centralized reservation-based protocol that accommodates human-driven and semi-automated vehicles in addition to fully-automated vehicles. This is the only protocol in literature that enabled smooth interaction between all types of vehicles.

The proposed algorithm in this chapter builds on the work of Zohdy et al. [19-22, 53], who used centralized multi-agent modeling and proposed and tested a number of techniques to manage the passage of automated vehicles through the intersection. In [19], a simulator (OSDI) that used a heuristic optimization algorithm that continuously adjusted the vehicles trajectories to minimize the occupancy time in areas of conflict in the intersection was built in the central controlling agent. The algorithm significantly reduced the delay for a simplified scenario of four-vehicles passing through the intersection zone. In [20-22], a game theory framework and an optimization framework was used to develop a heuristic algorithm for automated vehicles equipped with cooperative adaptive cruise control CACC, achieving a significant reduction in vehicle delays compared to traditional intersection control schemes, such as the use of traffic lights or stop signs. This analysis considered mixed flows of automated and regular vehicles and lane sharing, as well as superimposing the logic on the control of roundabouts.

The proposed algorithm in this chapter used a fully-distributed approach depending solely on V2V communication. Centralized approaches form a communication and control bottleneck
and may not prove scalable for application in real-life situations where tens or hundreds of cars may be in the vicinity of the intersection concurrently and may flood the central controller.

A distributed algorithm to address this problem has been proposed by a few research efforts, such as the work of Makarem [36], which required each vehicle to communicate with all other vehicles in the intersection. In the proposed algorithm in this chapter, the focus is on how communication exchanged in each time step can be minimized. Additionally, the heuristic approach is much easier to implement and allows for better optimization of traffic to avoid long queues of waiting cars.

3.3 Proposed Algorithm

This section provides a detailed description of the proposed intersection automated vehicle control algorithm.

3.3.1 Overview

In this chapter, a heuristic distributed coordination algorithm is proposed for the management of vehicles traversing an intersection. The algorithm exploits the fact that drivers approaching an uncontrolled intersection typically make rational decisions while proceeding or yielding without having complete information about all oncoming vehicles from all approaches. Considering an intersection with four approaches, each a single lane, the proposed algorithm assumes the four leading vehicles communicate with each other at each time step to schedule their entry time to the intersection. In each time step, one of the four vehicles will be responsible for updating the schedule if needed. If the scheduling agent identifies a conflict that may lead to a collision between two or more vehicles, the arrival times of some of the vehicles are altered by sending messages from the scheduler to update their trajectory to ensure they arrive at the correct time. Priority is always given to vehicles in more congested lanes when a conflict is identified by
the scheduler. Other than the four leading vehicles, a vehicle communicates only with its two neighboring vehicles (the one preceding and the one following it) at each time step.

The goal of the proposed system is to optimize the movement of automated vehicles traversing an intersection. The solution proposed in this chapter focuses on how to achieve this in a realistic and feasible manner. The proposed algorithm uses a fully-distributed approach: There is no central agent controlling the traffic approaching the intersection; instead each vehicle in the proximity of the intersection is modeled as an agent. All agents cooperate using a heuristic distributed coordination algorithm. A distributed algorithm seems to be more reasonable in real-world applications for a number of reasons:

a. The use of a central agent to control traffic may result in congestion within the system and thus act as a bottleneck as the volume of traffic increases in the intersection vicinity. For example, an intersection in a downtown area of a big city during the peak period may have hundreds of vehicles approaching from different directions at the same time. The ability of a central infrastructure controller to scale well and be able to communicate with all vehicles and schedule their passage in an efficient manner is highly questionable.

b. A technique requiring installing some device at each intersection to control the traffic is much more expensive compared to a distributed approach. It is reasonable to assume that state and city transportation officials are less enthusiastic about approving or testing a traffic control system that requires the installment of hardware at each intersection. Furthermore, both central and distributed approaches will require a vehicle to be equipped with very similar communication devices regardless of whether they need to communicate with an infrastructure agent or with other vehicles in close proximity to them.
c. Another important reason to favor distributed systems is related to safety issues and how the system will respond to failures in the central agent unit. Distributing the traffic control among the vehicles themselves allows for a truly fault-tolerant system.

Focusing on the wireless communication limitations due to the complexity of the traffic control at an intersection, most research efforts in this field have focused on the modeling of the control problem while ignoring the communication constraints. It is notable that several approaches implicitly assumed that communication is guaranteed to be successful and that a packet is always transferred in trivial time compared to the processing time of the control strategy. Unfortunately, the current state of wireless communication used for vehicle-to-vehicle communication does not support these assumptions. The scalability and feasibility of an approach that requires dozens of vehicles in an intersection area to exchange their states such as their positions and speeds every fraction of a second is highly questionable. In this chapter, the order was reversed. Starting from the communication restrictions, such as the transmission errors, delays, interference, and consumed power, the focus is to propose a protocol that can function well in such a non-ideal environment. Therefore, for this algorithm, trying to minimize communication between vehicles is a top priority, followed by finding a way to control the traffic within the communication limits.

By applying a priority scheme to avoid forming queues at the intersection rather than scheduling the passage of vehicles on a first-come first-serve fashion (FIFO), the proposed algorithm gives priority to vehicles that may form a longer queue if required to slow or stop at the intersection. The goal is to minimize the overall delay incurred by vehicles trying to pass the intersection. The scheduling technique will be discussed in more detail later.
3.3.2 Modeling Agent States

Every vehicle is modeled as an agent that has a role in coordinating with other agents (vehicles) to order the passage of all vehicles without collisions and with the minimum possible delay. As summarized in Figure 1, a vehicle passes through a number of states until it successfully passes the intersection safely.

1. A vehicle that is far from the intersection zone (more than 200 m away from the intersection) is in state “Out”. It simply proceeds without any actions related to intersection traffic control.

2. As the vehicle passes the edge of the intersection zone (within 200 m of the intersection), it moves to state “Last”.
   a. As soon as it enters this state, a vehicle calculates the time it could reach the intersection if it continued to travel at its current speed if not delayed by other conflicting vehicles.
   b. It sends its properties and the time it calculated in a broadcast message to cars ahead of it in the lane.
   c. The vehicle later receives a reply from the vehicle preceding it so it knows the agent it will be following. At each time step, the vehicle is updated with any change of the properties of the preceding car so it can follow without colliding.

3. A vehicle moves from “Last” state to “Mid” state when it receives a message from a vehicle announcing it is the new last in the lane. It receives the properties of the new car following it and replies with its properties so the new last can follow it appropriately. Meanwhile, the “Mid” agent continues to follow the vehicle in front of it. Thus, at every time step, every vehicle in a lane updates the vehicle immediately following it, so at each time step, each
vehicle needs to receive information from the one ahead of it and verify whether it needs to change its properties (if it needs to decelerate, for example) and then sends its possibly updated properties to the vehicle following it.

4. A vehicle enters “Head” state when it receives notification that the vehicle preceding it has passed the intersection.
   a. The new “Head” communicates with other “Head” agents in other approaches.
   b. It listens to any messages from every newly arriving “Last” agent so it knows all vehicles behind it.

![Agent State Diagram](image)

**Figure 1: Agent State Diagram**

Every time unit, one of the “Head” vehicles from the different approaches of the intersection becomes the “Scheduler” that is responsible for making sure that all vehicles in the intersection zone arrive and pass the intersection at different non-conflicting times. Only the
“Heads” communicate with each “Head” giving the most updated arrival times of vehicles following it.

3.3.3 Scheduling Algorithm

The scheduling of vehicles is designed to minimize the overall average delay of vehicles in the intersection area by avoiding forming queues of waiting vehicles. To achieve this in a distributed manner, the following algorithm is executed at every time step.

1. For each approach, the “Head” vehicle checks based on the current time, whether it is its turn to be the scheduler for this time step.

2. Each “Head” sends the partial schedule it holds for vehicles behind it in the lane. Any vehicle in a lane that alters its speed and time to reach the intersection conflict area sends a message with its new arrival time to the “Head”, so each “Head” knows the number of vehicles in its lane and holds a sorted list of times when these vehicles will reach the intersection.

3. The current scheduler receives the messages from the other “Heads”, and these messages give a complete picture of the expected times of arrival of all vehicles from all approaches.

4. The scheduler agent performs the following steps:
   a. Merge the sorted lists of various approaches into a global list of arrival times of all vehicles at the intersection
   b. Extract the traffic surges or waves; a traffic surge is a continuous flow of vehicles in the intersection conflict area from different approaches without any gaps of free traffic. The efficiency of scheduling depends on the capability of distinguishing traffic surges and identifying which approach has more vehicles that need to
proceed, delaying one vehicle in such an approach results in delaying all vehicles following it in the same lane.

c. For each traffic lane, the number of vehicles in each approach is determined; approaches with more vehicles are given higher priorities. Starting with the highest priority approach, the scheduler reserves time slots for vehicles.

d. When a vehicle from a lower priority approach conflicts with a vehicle from a higher priority approach that has already reserved time slots, the scheduler reserves the slot for the lower priority vehicle at the earliest possible time.

e. The scheduler sends to each “Head” an array with values representing how much delay is required from each vehicle in each approach. For example, the value zero for a vehicle means the scheduler does not require this vehicle to slow down as it was given priority to pass or no conflict was discovered at the time of its arrival at the intersection. On the other hand, a value of 5 for example associated with a vehicle by the scheduler means that the scheduler determined that this vehicle needs to delay its arrival to the conflict area of the intersection by 5 seconds.

f. Each “Head” receives the partial schedule sent by the scheduler for its approach, identifies if there are any nonzero values associated with any vehicles, if such values exist, and forwards the delay orders from the scheduler to the vehicles that need to delay. Each of these vehicles receives the order and decelerates to a speed that will allow it to enter the intersection at its reserved time.

At the next time step, another “Head” will be responsible for scheduling, and the previous steps are repeated. It may take a vehicle a few time steps to receive its final permitted time of entry to the intersection based on the traffic in all lanes.
3.3.4 Communication Requirements

Compared to the distributed approach in [36] which requires vehicles to communicate their states at every time step, the proposed algorithm in this chapter reduces the required communication in the following manner:

1. A vehicle entering the intersection zone will broadcast a message once to reach all vehicles in its lane (200 m). In this manner, the “Head” and the previous “Last” vehicles will be aware that a new “Last” is in the lane. The old “Last” will reply to inform the new “Last” of the car ahead of it. The distance between the two vehicles is typically small, and the power used for this reply message can thus also be assumed to be small.

2. At each time step, a vehicle in a lane receives only a message from the car ahead of it, and sends a message to the car behind it. In this way, the vehicles in a lane can adjust their speeds to follow each other. This minimizes interference and communication transmission power as a vehicle knows where the neighbor vehicles are located.

3. The exception from the previous point is when a vehicle is “Head” of its approach. In this case, it sends a sorted array of arrival times of vehicles in its lane to the scheduler. In the case that the scheduler wants to delay some vehicles, a reply is broadcast to other “Heads” with the vehicle ids and amount of time to slow down. In most time steps, no updates are made to the schedule or minimal changes may be expected, so even this communication is not significant.

Only when a “Head” leaves the intersection does it send a special message to the vehicle following it to become a new “Head”.

3.4 Simulated Experiments and Results

The proposed algorithm was implemented in MATLAB and the following experiments were performed.

3.4.1 Experiment Set 1

A two-lane cross-intersection was simulated. Each approach is a single lane heading toward the intersection. It was assumed that all vehicles are automated and that a vehicle entering the intersection zone will not accelerate. It either proceeds with its speed until it passes the intersection if no conflict exists from traffic of other approaches, or it slows down and may stop for some time to obey scheduling orders from peer vehicles ahead of it that are responsible for scheduling.

For this experiment, it was assumed that all vehicles enter the intersection zones with speeds varying between 30 and 40 mi/h (approximately 12 to 18 m/s). Intersection traffic was randomly generated allowing for a maximum gap of 10 seconds between two consecutive vehicles in a lane. It was assumed that all lanes have similar traffic, so the number of simulated vehicles was evenly divided among the four lanes.

The experiment was initiated by simulating four vehicles (one vehicle per lane), and running two algorithms, the distributed approach discussed in this chapter, and the FIFO centralized approach discussed in the related work section. The average delay and the maximum delay of both approaches were obtained for the four-vehicle scenario. The experiment was repeated ten times and the overall average delay and maximum delay for the two compared algorithms were recorded.

The number of vehicles considered in the simulation was increased in steps. Each step involved ten runs of randomly generated traffic of vehicles where the number of vehicles was increased by four (one added vehicle to each lane). The last ten runs included one hundred vehicles.
The results of this experiment are summarized in Figure 2 and Figure 3. Figure 2 shows how the average delay (s/veh) experienced by a vehicle decreased using the distributed prioritized scheme compared to the central FIFO approach. The reduction in the average delay increased gradually until it reached about 40% for the 100-vehicle simulations.

![Comparing the proposed algorithm with FIFO scheme](image)

**Figure 2: Average Delay of Proposed Algorithm Compared to FIFO**

Comparing the maximum delay experienced by a vehicle on the other hand did not show the same significant reduction as can be seen in Figure 3. A larger number of vehicles still shows that the distributed prioritized approach is better than the central FIFO approach.
3.4.2 Experiment Set 2

In order to assess the effect of heavier traffic on both approaches, the same set of simulated experiments was repeated with heavy traffic. Instead of allowing for a 10 second gap between two consecutive cars, the maximum allowed gap was reduced to range between one and five seconds.

The same trend was noticed for the average delays resulting from both approaches as shown in Figure 4. The maximum delays seen in this experiment are significantly reduced for the proposed distributed approach, as illustrated by Figure 5.
Figure 4: Comparing Average Delay of Two Methods with Heavy Traffic

Figure 5: Comparing Maximum Delay of Two Methods with Heavy Traffic
3.5 Conclusions and Future Recommendations

From the results shown in section 4 and the discussion of the communication protocol in section 3, the following contributions can be summarized:

1. The proposed distributed algorithm was successfully implemented and simulated for various numbers of vehicles in each lane. The rotating scheduling processing was successful without any collisions in any of the experiments.

2. The concept of prioritizing the assignment of time slots to vehicles based on identifying surges and giving preference to lanes with more crowded vehicles in each surge proved powerful when compared to the centralized FIFO method. The average delay reduction is clearer when more vehicles are simulated and when time spaces between vehicles become tighter.

3. The maximum delay experienced by a vehicle is also significantly reduced with heavier traffic compared to FIFO.

4. The communication requirements for the use of the proposed algorithm in real-life intersections are affordable. The proposed solution does not require installing any infrastructure devices.

Among the future considerations is to study the robustness of the proposed algorithm by modeling the communication limitations such as error rates and transmission delay to see the effect on the scheduling process and if it may result in any collisions, and if this is the case, how to modify the algorithm to guarantee fault-tolerance.

Another future direction is to extend the algorithm to multiple-intersections so that vehicles can better cooperate to avoid congestion.
Chapter 4

Two Algorithms for Traffic Optimization of Automated Vehicles at Multi-Intersection Networks

Abstract

Automated vehicles can now be manufactured at a reasonable cost. Managing traffic of automated vehicles at intersections using conventional traffic signals, stop signs, etc. is not optimal. Several centralized and distributed algorithms were proposed for minimizing the delay of automated vehicle traffic at isolated intersections. For a network of multiple intersections, communication between neighbor intersections can enhance mobility and decrease the average delay experienced in the network of intersections.

In this chapter, two novel multi-intersection algorithms are proposed extending a previously proposed algorithm for isolated intersections. Simulated results show how a small extension requiring no infrastructure investment reduced the average delay by 46%. This algorithm depends on indirect communication between adjacent intersections using vehicles traveling between intersections. Another promising algorithm that requires I2I communication between neighbor intersections achieved a 79% reduction in average delay. This algorithm can be achieved by modeling the system as two layers of multi-agent systems.
4.1 Introduction

More than one billion vehicles are currently operating in the world. About one quarter of those vehicles are running on US roads [50]. With continued economic growth and urbanization, the global vehicle population may reach 4 billion vehicles by 2050 [4]. In many big cities and metropolitan areas, traffic congestion at rush hours is inevitable, wasting valuable time and effort of commuters, increasing emissions polluting the environment, and increasing fuel consumption.

Traffic at cross-road intersections is a major cause of congestion. Several research efforts have been conducted and various solutions have been proposed to improve mobility and safety at intersections. With the rapid advancements in technologies used to build self-driving automated vehicles, the cost of manufacturing these vehicles has quickly fallen to a few thousand dollars. There is a need to propose efficient control algorithms to optimize traffic at intersections assuming the use of automated vehicles. Several researchers proposed ways to optimize such traffic for a single isolated intersection. More details of some of these research efforts follow in section 0.

One important way to improve traffic at an intersection is to provide the optimizer with knowledge about traffic at nearby intersections. As a result, the optimizer can prioritize conflicting requests in a way that better avoids delaying vehicles in approaches that may be congested. The hypothesis of this chapter is that augmenting an algorithm for an isolated intersection with information about its direct neighbor intersections can significantly reduce average delays experienced by vehicles proceeding through the network of intersections.

Building on the distributed algorithm for isolated intersections proposed earlier in chapter 3, two novel multi-intersection algorithms are proposed in this chapter. Simulation results show that both algorithms achieve significant reductions in both the average delay and maximum delay experienced by simulated traffic.
The rest of this chapter is organized as follows: section 2 briefly discusses some related work in literature; section 3 provides a detailed discussion of the two proposed algorithms; the results of the experimental work are shown in section 4; finally, conclusions and future work are discussed in section 5.

4.2 Related Work

There are several research efforts in the literature that addressed traffic management of automated vehicles at isolated uncontrolled intersections. Dresner and Stone proposed Automated Intersection Management (AIM) which is a simple centralized intersection control protocol for automated vehicles based on First In First Out (FIFO) priorities [15]. AIM is a multi-agent time reservation system consisting of an intersection manager (controller) and vehicle agents. When a vehicle approaches the intersection it requests a time-space slot to cross the intersection. Upon receiving the driver agent request, the controller simulates the vehicle crossing the intersection and based on the output trajectories, the controller makes decisions that avoid conflicts. Zhu et al. [17] proposed a central algorithm where the controller evaluates the total delay if a request is allocated and if it is postponed and makes the decision that results in a smaller delay. Zohdy and Rakha proposed a centralized algorithm [19] where the controller uses a custom simulator to continuously adjust vehicle trajectories to minimize occupancy time in intersection conflict zones. They also proposed a centralized algorithm using the game theory framework [21] to be used with vehicles equipped with cooperative adaptive cruise control CACC.

Some researchers proposed distributed algorithms for traffic management at uncontrolled intersections. One interesting scheme is proposed by Khoury [37], where automated vehicles at an intersection make localized access decisions based purely on sensing information rather than requiring V2V or V2I communication. The scheme avoids collisions but is less optimal than other
available algorithms. Guangquan et al. [38] proposed another distributed algorithm where passing vehicles exchange information and follow a predefined set of rules to resolve conflicts at the intersection. Other examples of proposed distributed methods are [36] and [33]. The drawback of these algorithms is the requirement that each vehicle communicate with all other vehicles at each time step.

Wuthishuwong and Traechtler [40] proposed a multi-intersection algorithm where traffic information is exchanged using infrastructure to infrastructure communication I2I. Their algorithm is based on the concept of the green wave - trying to maintain the continuous driving of vehicles while passing the intersections. Fei and Dridi [41] proposed a genetic algorithm approach designed to find an optimal or near-optimal vehicle passing sequence at each intersection in a multi-intersection network.

4.3 Proposed Algorithms

In chapter 3, a fully distributed heuristic algorithm for optimizing traffic of automated vehicles at an isolated intersection is proposed. This algorithm is a base for the algorithms proposed in this chapter. In the rest of this chapter, the algorithm proposed in chapter 3 will be referred to as IIZA (Isolated Intersection Zone Algorithm). The goal of IIZA was to minimize the delay suffered by automated vehicles close to an intersection depending only on V2V communication. In real life, such as in metropolitan and downtown areas, cross-road intersections may be located very close to each other forming a network of intersections. If the traffic control in these multiple intersections is not coordinated, leaving each intersection to control its zone as if there were no other intersections close by, vehicles may suffer accumulated delays in subsequent intersections on their routes. The traffic in such a network of intersections might not be optimal if handled as a group of isolated intersections.
The two algorithms proposed in this chapter extend IIZA such that at each intersection, the four direct neighbor intersections (west, east, north, and south) are considered while scheduling the passage of vehicles. The first proposed algorithm will be referred to as NIZA (Networked Intersection Zones Algorithm) which still depends on V2V communication and the cooperation of vehicles with no dependence on any infrastructure agents. The second algorithm, referred to as DLA (Dual-Layered Algorithm), defines two layers of traffic control: the lower layer applies IIZA at each intersection zone, while the higher layer defines a new distributed multi-agent system where each intersection in the network is an agent. Each intersection is modeled as an agent that communicates with the four direct neighbor intersections (west, east, north, and south). An intersection agent communicates with its direct neighbor intersection agents to exchange traffic states of the four approaches in the intersection zone and broadcasts what it knows about its neighbors to the vehicles in its zone. Those vehicles then make use of this information while they schedule their movements through the intersection. Thus, DLA requires V2V, V2I, and I2I communication to optimize traffic in a network of intersections. The following subsections discuss each of these algorithms, IIZA, NIZA, and DLA, in more detail.
4.3.1 Isolated Intersection Zone Algorithm IIZA

As mentioned earlier, IIZA is proposed and discussed in detail in chapter 3. It is also briefly covered in this subsection. IIZA is a fully-distributed algorithm that assumes traffic of automated intelligent vehicles with V2V communication. It allows vehicles to successfully pass the intersection without collisions and aims at optimizing the traffic in the intersection by minimizing the delay experienced by vehicles. In order to achieve this, IIZA gives priority to vehicles in approaches with heavier traffic to try to avoid forming long queues of vehicles waiting in the intersection zone. It does not require installation of any central infrastructure agent in the intersection to manage traffic. Rather, vehicles in the intersection are intelligent agents that cooperate to schedule their passage through the conflict zone using a designed communication protocol. While a vehicle is in an intersection zone, i.e. within 200 meters of the center of the
intersection, its role in IIZA depends on its state; Figure 6 shows the state diagram used by IIZA to distribute traffic control across agents (vehicles).

Assuming vehicle V1 passes the edge of the intersection zone (within 200 meters of the intersection), it moves from state “Out” to state “Last.” The vehicle sends a message that indicates its position, speed, acceleration, and requested time to pass the intersection conflict zone. This message is received by vehicle V2 ahead of V1 in the same lane. V2 goes from state “Last” to state “Mid,” sends a response to V1 acknowledging receiving V1’s message and indicating its own position, speed, and acceleration of V2. Thus, V1 can update its properties to follow V2 without a collision. V2 similarly follows its preceding vehicle, and, at each time step, each vehicle sends its current properties to its follower.

A vehicle moves from state “Mid” to state “Head” when it receives a message from its preceding vehicle that it has passed or is about to pass the intersection. The “Head” vehicle keeps track of all vehicles behind it in the lane, knows when each vehicle wants to reserve the intersection conflict zone, and computes a partial schedule to be merged with partial schedules held by “Head” vehicles of other approaches. For each time step, one of the four “Head” vehicles becomes responsible for merging the partial schedules into one schedule that all vehicles in the intersection zone should follow. To do this, the scheduler agent at a time step prioritize the lanes according to the number of vehicles in each lane. For example, if vehicles V1 and V2 from conflicting approaches are to reach the intersection at the same time, V1 will be allowed to pass first and V2 will be delayed if V1 is in a lane with more vehicles in the intersection zone than the lane of V2. The objective of this prioritization scheme is to avoid as much as possible the formation of long queues of waiting vehicles at any approach.
The scheduling algorithm continues to rotate between “Head” agents and any vehicle that is required to delay is informed by a message broadcast by the “Head” of its approach. The objective is to minimize the overall average delay of vehicles in the intersection. Simulated experiments showed that this technique is promising in reducing average delays compared to other centralized algorithms such as [15] and [21]. At the same time, IIZA tries to allow vehicles to cooperate with as few communication messages as possible. Unlike [36], where all vehicles should exchange their properties for each time step, IIZA requires a vehicle to communicate with the direct neighbors only beside the “Head” of its approach. Only the “Head” vehicle communicates with “Heads” of other approaches. This can enable IIZA to be implemented in real-life using current wireless communication protocols. Most IIZA communication goes between vehicles close to each other, reducing the power needed for wireless messages, decreasing interference, and allowing the reuse of wireless channels in different approaches.

IIZA for isolated intersections has two advantages:

1- It is fully-distributed depending only on passing vehicles to optimize traffic at an intersection with no need for infrastructure agents, which can make it feasible for implementation with reasonable cost.

2- It attempts to minimize the V2V communication required to achieve successful traffic control and optimization. It also uses a heuristic algorithm for scheduling vehicles that is not computationally demanding.

One major disadvantage is that IIZA only considers the traffic around the intersection in a circle with a radius of 200 meters. This limitation is due to the limits of current wireless technology used in vehicular networks. In this way, IIZA uses incomplete information for the optimization of the traffic at the intersection. This leaves room for more optimization and delay reduction if additional
information can be fed to the vehicles while they optimize their movement through the intersection. The following two subsections discuss two ways to overcome this disadvantage by incorporating information from neighbor intersections.

4.3.2 Networked Intersection Zones Algorithm NIZA

NIZA extends IIZA to allow vehicles at an intersection to use the history information of neighboring intersection zones to estimate the traffic state beyond the 200 meter limit. The main objective of NIZA is to achieve this without losing the two advantages of IIZA stated in the previous subsection. NIZA extends IIZA with no increase in communication requirements, and continues to depend solely on the vehicles to manage their movements without any need for any infrastructure agents installed at the intersections.

For the purpose of this chapter, a 4x4 grid network of intersections has been used. Figure 7 shows the network of intersections, where four horizontal major roads intersect with four vertical minor roads. The network is composed of 16 intersections, labeled 1 to 16 in Figure 7.
NIZA extends IIZA in the following manner:

When a “Head” vehicle of an approach passes an intersection conflict area, it stores the number of vehicles behind it in the same approach for later use. When the vehicle enters the next intersection zone on its route through the network, it will be in state “Last”, and it provides the “Head” vehicle of its approach to the intersection zone with the history value it stores for the number of vehicles that were in the previous intersection when it left. This value is used by NIZA as an estimate of the traffic in the additional 200 meters behind the “Last” vehicle in the approach. The goal here is to double the radius of the zone considered for traffic optimization at an intersection. Information about traffic within 200 meters from the center of the intersection is now augmented with an estimate of the traffic level in the following 200 meters.

For example, assume vehicle V1 proceeds through intersections 1 and 2 in Figure 7. Assume V1 left intersection 1 at time t, and was at the edge of the intersection zone of intersection 2 at time t.
+ d. At time t, V1 definitely knows the size of the following traffic in a window of 200 meters (the intersection zone of intersection 1). Assume this value is \( C_t \). At time \( t + d \), V1 does not know how many vehicles follow in the next 200 meters, but NIZA uses \( C_t \) as an estimator of \( C_{t+d} \). Of course, the true value of \( C_{t+d} \) will in many cases differ from \( C_t \), but it can roughly indicate how heavy the following traffic is, and can be a good estimator especially when the two intersections are not far from each other.

For each approach, the traffic volume is the sum of the number of vehicles inside the intersection zone for this approach and the estimated number of vehicles in the following 200 meters. Therefore, the scheduling algorithm uses estimates of traffic of the four approaches and assigns priorities based on it.

The difference between IIZA and NIZA can be understood by thinking of IIZA as an algorithm that can only see the traffic in a narrow 200-meter slice of the road (the intersection zone). NIZA can see an additional 200-meter slice of traffic, so its estimation of more congested approaches should be more accurate than IIZA. However, NIZA still uses incomplete information, and even if the estimates for 400 meters are accurate, there is still unseen traffic between the intersections that is not captured by the values transferred between intersections. Nevertheless, it is interesting that NIZA still works without requiring the installation of any central infrastructure agents at the intersections. The processing is still fully distributed, and NIZA computational and communication requirements are the same as those for IIZA. This means that any improvement in delay reduction achieved by NIZA is achieved for free.

4.3.3 Dual-Layered Algorithm DLA
Instead of considering only two 200-meter slices of the road as NIZA does. DLA is a multi-intersection algorithm that uses the history of traffic at an intersection at carefully chosen times to estimate current traffic in the whole road section between the two intersections.

For example, consider Figure 8, which shows the road section between two intersections as five 200-meter slices assuming the distance between intersections 1 and 2 is 1 km.

![Figure 8: ROAD SLICES BETWEEN TWO INTERSECTIONS](image)

Considering the through traffic from intersection 1 towards intersection 2, vehicles around intersection 2 knows the traffic state in slice A only out of the five slices in figure 3. Using IIZA, they use only this limited knowledge while optimizing the traffic. In the case of NIZA, the vehicles at intersection 2 add an estimate of traffic in slice B to what they know about the traffic in slice A. Using DLA, the vehicles at intersection 2 will be supplied with more estimates of traffic in remaining slices C, D, and E. Estimated traffic at road slices far from intersection 2 can be achieved by using the history of through traffic in the same direction at the intersection zone of intersection 1.

Assuming vehicles proceed between intersections 1 and 2 with an average speed $s_{av}$, the number of vehicles in the intersection zone of intersection 1 heading to slice E at time t can be used to estimate the number of vehicles in slice E at time $t + 200/s_{av}$. It can also estimate the number of vehicles in slice D at time $2(t + 200/s_{av})$ and so on.

Vehicles at intersection 2 need the values of traffic at neighbor intersection 1 at these times in the past to estimate the current traffic in each of the slices between the two intersections.
DLA assumes there are two layers of multi-agent systems working together. The lower layer is composed of the vehicles in an intersection zone applying IIZA and is responsible for managing traffic locally at an intersection.

The upper layer is a multi-agent system of intersection agents. For example, for the network in figure 2, there will be 16 agents.

DLA requires each intersection agent do the following for every time step:

Storing the traffic volume in each direction within its intersection zone for later use by adjacent intersections. This can easily be accomplished simply by listening to the exchanged messages between the “Head” vehicles working using a modified version of IIZA.

1- Exchanging traffic information in its zone with the four direct neighbor intersection agents.

2- Broadcasting historic values of neighbor intersections traffic that represent current estimates of traffic beyond the intersection zone. Vehicle agents of lower level can use these broadcasted estimates to optimize the traffic through the intersection.
4.4 Simulated Experiments and Results

The proposed algorithms discussed in section 3 are implemented using MATLAB. A 4x4 grid network of intersections is used, similar to the one in Figure 7. Simulations are run to compare the algorithms and compute the delays in the network. The distance between any two adjacent intersections is 1 km. The horizontal roads represent major roads with heavier traffic, while the vertical roads are minor roads. Vehicles are allowed onto the major roads at double the rate (veh/h) of the arrival of vehicles on minor roads. Origin-Destination demands for an hour of traffic (5040 vehicles) are submitted to the INTEGRATION simulation program. The vehicle initial conditions (initial speed, time, and coordinates when entered network) and vehicle routes generated by INTEGRATION are used as inputs to the MATLAB simulator built to test the algorithms proposed in this chapter.

Figure 9, figure 10, and Figure 11 show the resulting delays for each vehicle in the network during the simulated one-hour traffic using IIZA, NIZA, and DLA. Figure 9 illustrates the improvement NIZA achieved compared to IIZA. Figure 10 illustrates the clear reduction in delays when using DLA compared to IIZA. Figure 11 puts the three algorithms into one figure to make it clear that, as expected, DLA is more optimized than NIZA, yet it should be noted that it might be more costly to implement.

The delay of a vehicle in an intersection zone is the difference between the time it actually spent in the intersection zone and the time it needs to pass the intersection zone if it is the only car present. The delay of a vehicle shown in the figures represents the sum of its delay times in all network intersections on the vehicle’s route.
Figure 12 shows the average and maximum delays found for simulated vehicles when using each of the three tested algorithms. NIZA is found to achieve a 46% reduction in the average delay of vehicles compared to IIZA. DLA achieved a 79% reduction in the average delay compared to IIZA. The maximum delay experienced by vehicles is reduced by 32% when using NIZA and by 58% when using DLA.
Figure 10: DLA COMPARED TO IIZA

Figure 11: DLA COMPARED TO NIZA AND IIZA
4.5 Conclusions and Future Recommendations

This chapter proposes two novel algorithms for traffic control of automated vehicles at interconnected intersections. Even a small extension of an algorithm for isolated intersections reduced the delay significantly. The NIZA algorithm is less optimal than the DLA algorithm; however, it allows indirect communication between neighbor intersections by enabling a vehicle to store traffic when it is at the first intersection and use it later when it enters the next intersection zone. It is easy to implement and the computational and communication requirements are reasonable. Installing infrastructure intersection agents to exchange I2I traffic information proves to be efficient and produced the smallest delay. The DLA algorithm shows how intersection zone traffic can be used to estimate traffic in other road segments away from the intersection. The presence of infrastructure agents can also enhance other transportation functions, such as vehicle routing to avoid congested intersections.
Future extensions to the proposed work include further testing of the multi-intersection algorithms in various scenarios with different intersection network sizes and topologies and tuning the algorithms accordingly. Using the framework of DLA for re-routing the vehicles in the network for enhancement of network mobility is another objective.
Chapter 5

Dual-Layered Algorithm using Godunov Scheme Estimation for Multi-Intersection Networks

Abstract

The price of Automated Vehicles (AVs) is expected to decrease in the very near future, thus making these vehicles a reality. Managing the movement of AVs through roadway intersections using conventional traffic control devices (traffic signals and stop signs) is not optimal. Consequently, several centralized and distributed algorithms have been proposed to minimize the total intersection delay. For a network of multiple intersections, communication between neighboring intersections can enhance the corridor mobility and decrease the network-wide delay. In this chapter, a novel multi-intersection algorithm is proposed, extending a previously proposed algorithm for isolated intersections. The proposed algorithm requires Infrastructure-to-Infrastructure (I2I) communication between neighboring intersections. This algorithm models the system as a two-layer multi-agent system. The algorithm is compared to an optimized signalized intersection controller, and is demonstrated to reduce the network-wide mean travel time by 40%.
5.1 Introduction

The previous chapters discussed how roadway intersections constitute bottlenecks within transportation networks resulting in major congestion. A number of heuristic distributed algorithms are proposed in earlier chapters to increase mobility of automated vehicles at intersections. The proposed algorithms; IIZA, NIZA, and DLA can also be applied for non-automated vehicles equipped with communication capabilities, by allowing versions of iCACC running one or more of the proposed algorithms and taking control of the vehicle when it enters an intersection zone.

NIZA and DLA algorithms demonstrated the power of involving information from neighboring intersections. It has been shown in chapter 4 that the networked-intersections approach reduced the delay compared to the isolated intersection algorithm IIZA. The DLA algorithm works by dividing the distance between two intersections into a number of zones (of length 200 meters each) and estimating the traffic volume (number of vehicles) in each zone. The main hypothesis is that by giving priority to vehicles in more populated lanes, the odds of forming long waiting queues in these higher traffic lanes will decrease, leading to increased mobility and decreased overall delay. Assuming that this hypothesis is valid, the question becomes how to enhance the DLA algorithm estimates of traffic out of the intersection zones. The objective of this chapter is to propose an enhanced algorithm DLAG that extends the DLA algorithm by calculating more accurate estimates of the traffic between two neighboring intersections. DLAG uses the first order Godunov scheme [54, 55] to accurately estimate the number of vehicles along the road section between two intersections, so the scheduling algorithm assignment of priority will reflect the true state of the traffic and should be expected to achieve more reductions in the overall delay.
In order to evaluate the proposed DLAG algorithm, simulated experiments are designed to compare DLAG to the optimized signalized intersection control.

5.2 Motives for proposing a new distributed algorithm

This section briefly discusses the reasons why the authors believe it is important to manage traffic at intersections using a distributed rather than a centralized approach, and why a new distributed algorithm is inevitable. One major driving motivation for the proposed algorithms in this chapter is the need for cost-effective techniques to handle the growing challenge of intersection traffic management.

Enhancing mobility and safety at intersections is a challenge given that, currently 40% of accidents in the U.S. occur at intersections. According to the U.S. Department of Transportation, less than 10% of the intersections are currently signalized. Depending on its geometry and complexity, the average cost of signalizing an intersection is around $65,000. With more than three million intersections in the U.S., it is clear that developing non-signalized intersection control algorithms is much more cost effective. However, this will be true only if these alternative algorithms are feasible without requiring the installation of new infrastructure equipment at the intersections that may be comparable to the cost of signalizing the intersections.

In terms of cost, a well-designed fully-distributed approach can be much more feasible than a centralized approach. For example, one promising system, MARLIN [46, 47], required that traffic lights be connected to a central controller. The cost of this infrastructure addition is estimated to be between $20,000 and $40,000 per intersection. Applying such an approach for U.S. intersections could require a budget of 90 billion dollars.
Can a distributed approach really avoid this cost problem? The DLAG algorithm proposed in this chapter assumes the use of automated vehicles equipped with V2V capability. It is reasonable to assume that this is still too expensive to be feasible for real life application. However, it is important to consider the fact that V2V communication is expected to be mandated in a few years in the U.S. for safety reasons. In fact, V2V communication is considered the next generation of vehicle safety improvements after safety belts and airbags. If vehicles have to be equipped with V2V anyway, then making use of this capability for traffic management will in fact be economical.

Current distributed techniques for intersection management have two main drawbacks:

- They require a vehicle to broadcast to and receive from all vehicles in the intersection zone at every cycle. Using currently available wireless technology, this may result in severe interference and retransmissions and delays that cause the algorithm to fail.
- They only focus on isolated intersections where vehicles within the vicinity of an intersection have no knowledge about the traffic approaching from nearby intersections. The lack of knowledge about the traffic outside the intersection zone can limit the capabilities of a control algorithm to enhance the mobility of vehicles at the intersection.

The algorithms proposed in this chapter are designed to realistically compensate for these drawbacks.

5.3 Proposed Algorithm

As previously discussed, IIZA assumes vehicles communicate in a narrow 200-meter window (the intersection zone) due to limitations in the wireless communication technology. For IIZA or any isolated intersection algorithm, the scheduling algorithm is fed with partial incomplete
information, namely, the traffic in the 200-meter intersection zone. NIZA and DLA extend IIZA by providing additional estimates of traffic in the unseen road sections outside of the intersection zones. NIZA still uses incomplete information because there is still unseen traffic between intersections that is not captured by the values transferred between intersections. However, it is interesting that NIZA still works without the requirement of installing any central infrastructure agents at the intersections. The processing is still fully distributed, and NIZA’s computational and communication requirements are the same as those for IIZA. Thus, any improvement in delay reduction is achieved by NIZA at no extra cost. DLA requires an intersection to have an agent monitoring the traffic volume in each lane in the intersection zone. This intersection agent will send this information to the corresponding intersection agents of the direct neighboring intersections at every time step. The control is still distributed among the vehicle agents, who still make the scheduling decisions in light of the history information received by the intersection agents and broadcast to the vehicles in the intersection zone.

5.3.1 Dual-Layered Algorithm using Godunov Scheme DLAG

The problem with the DLA algorithm is the inaccuracy of the estimates of traffic in unseen road segments due to a number of non-realistic assumptions implied in its operation. While estimating traffic along road segments far from the intersection, DLA simply assumes that traffic at an upstream intersection zone flows as one unit until it reaches the downstream intersection zone. In other words, the exact number of vehicles on one approach is assumed to proceed at a constant forward speed and ignores the fact that queues also form and propagate backwards. It is unrealistic to assume that such an assumption will hold in reality. The promising results of the DLA algorithm discussed in chapter 4 compared to NIZA and IIZA are the driving factor for the authors to find a way to extend the DLA and to design an algorithm that makes accurate estimates of traffic along
the road beyond the intersection zone. DLAG is still a dual-layered algorithm that assumes there are two layers of multi-agent systems working together. The lower layer is composed of the vehicles in an intersection zone applying an enhanced version of IIZA and is responsible for managing traffic locally at an intersection, while the upper layer is a multi-agent system of intersection agents.

The main difference between DLA and DLAG lies in the type of information that is exchanged by intersection agents for each time step.

The governing equation of the flow of vehicles along a roadway is the same as the continuity equation for a fluid and is given by

\[
\frac{\partial k}{\partial t} + \frac{\partial q}{\partial x} = 0 \quad (1)
\]

Where \( k \) is the traffic stream density (i.e. number of vehicles per unit length) and \( q = k u(k) \) is the flux of vehicles (i.e. number of vehicles entering the road per unit time). The value of the traffic stream velocity \( u \) depends on the density \( k \). This relation is given by the fundamental diagram:

\[
k(u) = \frac{1}{C_1 + \frac{C_2}{u_f - u} + C_3 u} \quad (2)
\]

Where \( C_1 = \frac{u_f}{k_j u_c^2} (2 u_c - u_f) \), \( C_2 = \frac{u_f}{k_j u_c^2} (u_f - u_c)^2 \) and \( C_3 = \left( \frac{1}{q_c} - \frac{u_f}{k_j u_c^2} \right) \)

Equation 1 can be re-written as:

\[
\frac{\partial k}{\partial t} + u(k) \frac{\partial k}{\partial x} = 0 \quad (3)
\]
Equation 3 can be solved numerically using the Godunov scheme [55]. The road is discretized in small elements (cells) as shown in Figure 13.

![Figure 13: Road Discretization for Godunov Scheme](image)

Where, \( q_0 \) is the input flux and \( q_f \) is the output flux. It is noted that \( q_f \) is only given at the initial time step as the initial condition. Later values for \( q_f \) are determined using the scheme where \( q_{i \pm \frac{1}{2}} \) are the fluxes at the edge of cell \( i \).

The updated value of the density \( k_i \) at time \((n + 1)\Delta T\) is given by

\[
k_i^{n+1} = k_i^n - \frac{\Delta t}{\Delta x} \left( H_{i+\frac{1}{2}}^n - H_{i-\frac{1}{2}}^n \right) \tag{4}
\]

Where \( H_{i+\frac{1}{2}}^n \) is the numerical flux and is given by

\[
H_{i+\frac{1}{2}}^n = \frac{1}{2}(q_{i+1}^n + q_i^n) - \frac{1}{2}\left|q_{i+1}^n - q_i^n\right|\left(k_{i+1}^n - k_i^n\right) \tag{5}
\]

This expression is computed after solving a Riemann problem [56] and using the Roe approximate Riemann solver [57].

Equation 5 can be simplified as follows

\[
H_{i+\frac{1}{2}}^n = \begin{cases} 
\min(q_{i+1}^n, q_i^n) & \text{if } k_{i+1}^n > k_i^n \\
\max(q_{i+1}^n, q_i^n) & \text{if } k_{i+1}^n < k_i^n 
\end{cases}
\]

It is also noted that for the scheme to be stable we need to ensure the condition

\[
\]
\[
\frac{\Delta t}{\Delta x} \leq \frac{1}{u_f}
\]

\(u_f\) : is the free-flow velocity.

Applying this to a road section between two intersections, the road is divided into small cells of 20 meters in length. The first cell starts at the center of the upstream intersection, whose agent keeps track of the value of the input flux to the first cell at every time step and sends it to neighboring intersections. This allows the downstream intersection vehicles to use the algorithm to estimate the densities in each cell. This is how the DLAG algorithm works.

5.3.2 DLAG Implementation

The DLAG algorithm has been implemented and simulated using MATLAB. The MATLAB code simulates the vehicle movements and dynamics in the transportation network and also simulates all communication needed between the agents during the operation of the algorithm.

The following subsections discuss the dynamic model and the car-following models used for the implementation of DLAG.

Dynamic Model

For a vehicle moving on the road in a straight line, the corresponding equations of motion are:

\[
\begin{align*}
\dot{x} &= v \\
\dot{v} &= a
\end{align*}
\]

These equations of motion are depicted in Figure 14, where \(x\) is the position along the road, \(v\) is the corresponding velocity and \(a\) is the tangential acceleration of the vehicle.
For a typical vehicle, the maximum acceleration that could be experienced is

\[ a_{\text{max}} \leq \frac{F - R}{m} \]

Where, \( F \) is the traction force, \( R = R_a + R_r + R_g \) is the resistive force sum of the aerodynamic resistance \( R_a \), the rolling resistance \( R_r \) and the grade resistance \( R_g \) and \( m \) is the mass of the vehicle. The vehicle tractive force is defined by

\[ F = \min \left( 3600 \eta_a \frac{P}{\varphi}, m_t \alpha g \mu \right) \]
Where, $\eta_d$ is the drive-line efficiency, $P$ is the vehicle power, $v$ is the vehicle velocity, $m_{ta}$ is the mass of the vehicle on the tractile axle, $\mu$ is the coefficient of road adhesion and $g$ is the gravitational acceleration.

The aerodynamic resistance force is defined by

$$R_a = \frac{\rho}{25.92} C_d C_h A_f v^2$$

Where, $\rho$ is the fluid density, $C_d$ is the car drag coefficient, $C_h$ is the altitude correction factor, and $A_f$ is the frontal area of the vehicle subjected to the flow.

The rolling resistance force is defined by

$$R_r = m g \frac{c_{r0}}{1000} (c_{r1} v + c_{r2})$$

Where, $c_{r0}$, $c_{r1}$, and $c_{r2}$ are the rolling constants.

The grade resistance force is defined by

$$g = m g G$$

Where, $G$ is the roadway grade. In this work, we assume that the roadway is flat (i.e. $G = 0$).

The maximum deceleration that can be experienced by a vehicle in this work is $-5 \text{ m/s}^{-2}$. Therefore, the constraint subjected to the tangential acceleration is given by

$$-5 \leq a \leq \frac{F - R}{m}$$
Car-Following Model

In order to determine the behavior of a vehicle with respect to its peers on a lane, we make use of the Rakha-Pasumarthy-Adjerid (RPA) car-following model that includes the Van Aerde steady-state car-following model [58], the vehicle’s dynamics model described earlier and a collision avoidance model. Both models predict the proper velocity a vehicle should adopt in order to remain within a safe distance with respect to the vehicle in front.

Given two vehicles indexed by ( )n, and ( )n − 1, the expected velocity of vehicle (n) at time t + ∆t is given by [59].

\[
u_n(t + \Delta t) = \min \left\{\frac{u_n(t) + \frac{F - R}{m} \Delta t}{-c_1 + c_3 u_f + s_n(t + \Delta t) - \sqrt{A}} \right\}
\]

Where, \(u_f\) is the free-flow speed, \(u_c\) is the speed-at-capacity, \(k_j\) is the jam density, \(q_c\) is the lane capacity,

\[s_n(t + \Delta t) = s_n(t) + (u_{n-1}(t) - u_n(t)) \Delta t + \frac{1}{2} a_{n-1}(t) \Delta t^2\]

is the vehicle spacing,

\[A = \left[c_1 + c_3 u_f - s_n(t + \Delta t)\right]^2 - 4 c_3 [s_n(t + \Delta t) u_f - c_1 u_f - c_2] \Delta t\]

\(d_{max}\) is the maximum deceleration that could be experienced by the vehicle,

\[c_1 = \frac{u_f}{k_j u_c^2} (2 u_c - u_f), \quad c_2 = \frac{u_f}{k_j u_c^2} (u_f - u_c)^2 \quad \text{and} \quad c_3 = \left(\frac{1}{q_c} - \frac{u_f}{k_j u_c^2}\right)\]

are the steady-state car-following model parameters.

The car-following model could also be expressed in terms of the distance that the vehicle attempts to maintain with respect to the car in front.
\[ \bar{d} \geq \min \left\{ \max \left\{ \frac{c_1}{u_f - v} + \frac{c_2}{u_f} \frac{v^2 - \bar{v}^2}{2 d_{\text{max}}} + c_3 \right\} \right\} \]

Where, \( v \) is the velocity of the vehicle in consideration and \( \bar{v} \) is the velocity of the vehicle in front.

### 5.4 Simulated Experiments and Results

A 4x4 grid network of intersections is used similar to the one in Figure 7. Simulations are run to compare the algorithms and compute the delays in the network. The distance between any two adjacent intersections is 1 km. The horizontal roads represent major roads with heavier traffic while the vertical roads are minor roads as shown in Figure 15.

![Figure 15: The traffic entering and leaving the network](image-url)
Vehicles are allowed onto the major roads at double the rate (veh/h) of arrival of vehicles on the minor roads. The Origin-Destination (O-D) demand for one half of an hour of traffic (5040 vehicles) is submitted to the INTEGRATION simulation program as presented in Table 1. The vehicle initial conditions (initial speed, time, and coordinates when entered network) and vehicle routes generated by INTEGRATION are used as inputs for the MATLAB simulator built to test the DLAG algorithm.

<table>
<thead>
<tr>
<th>Origin</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>Sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>200</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>420</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>-</td>
<td>0</td>
<td>0</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>200</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>420</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>-</td>
<td>0</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>200</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>420</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>200</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>420</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>400</td>
<td>840</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>400</td>
<td>840</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>840</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>400</td>
<td>840</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>200</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>200</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>420</td>
</tr>
<tr>
<td>10</td>
<td>20</td>
<td>20</td>
<td>200</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>200</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>420</td>
</tr>
<tr>
<td>11</td>
<td>20</td>
<td>200</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>420</td>
</tr>
<tr>
<td>12</td>
<td>200</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>420</td>
</tr>
<tr>
<td>13</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>400</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>400</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>840</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>400</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>400</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>840</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>400</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>400</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>840</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>400</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>400</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>840</td>
<td></td>
</tr>
</tbody>
</table>

Table 1: O-D Demand for the network

5.4.1 Comparison with Optimized Signalized Intersection Control

To assess the effectiveness of the DLAG algorithm, the travel time for all vehicles while applying the DLAG algorithm is compared to the travel time considering the traffic signal control. The INTEGRATION software was used to simulate the network with the use of optimal signal timing for each traffic signal. As we only have one lane for each approach, each traffic signal has two phases allowing for permissive left turns, as shown in Figure 16.
At the first step, the major streets were given phase lengths, specifically 67% of the cycle length, while the minor streets were given 33% of the cycle length. These percentages were calculated based on the ratio of the volume to the unopposed saturation flow rate (i.e. 1800 veh/h). The lost time for the major and minor streets (i.e. the time for yellow and all red lights) was 4 seconds each.

Figure 17 illustrates the variation in the average delay as a function of the cycle length for the major and minor streets. The optimal cycle length was found to be between 20 and 40 seconds, so we selected a cycle length of 40 seconds as it is a more reasonable value and provides sufficient time for pedestrians to cross the street. The sub-optimal cycle length (i.e. 40 seconds) corresponds to a sub-optimum delay of 171.73 seconds for each vehicle (i.e. 2.86 minutes).
Figure 17: Average delay VS cycle length for major and minor streets

Given a cycle length of 40 seconds, INTEGRATION optimized the green time for each intersection and provided the optimized green split. This operation was performed every 240 seconds (15 times) during a simulation time of 3600 seconds.

The INTEGRATION simulation model provides different routing approaches. The Time-Dependent Dynamic Traffic Assignment (DTA) was used to simulate the network. Based on the expected volume and queue sizes in each link, the DTA anticipates the travel time and computes the minimum path for every scheduled vehicle departure. INTEGRATION provides us with the path used by each vehicle once it enters the network. The same routes were used to run DLAG to compare with the signalized control.

Figure 18 shows the histogram of the travel time of all vehicles for both DLAG and the optimized signalized control. The DLAG controller significantly reduced the travel time where the mean travel time for DLAG is 663 seconds compared to a mean travel time of 812 seconds using optimized signalized intersections. The mean travel time is reduced by 18.42%. In order to test the statistical significance of the achieved reduction in travel time, a pooled t-test is used to compare the two means of travel time for both control mechanisms. The null and alternative hypotheses are:
\[ H_0: \text{Mean}_{\text{Signalized}} \leq \text{Mean}_{\text{DLAG}} \]

\[ H_a: \text{Mean}_{\text{Signalized}} > \text{Mean}_{\text{DLAG}} \]

The p-value is found to be less than 0.0001, and we reject the null hypothesis; therefore, we conclude that the mean travel time for DLAG is significantly less than the mean travel time of the optimized signal control. The maximum travel time for DLAG also experienced a significant reduction of 23% compared to the maximum travel time seen using the optimized signalized control. It is also clear in Figure 10 that the uncertainty (spread) in travel time for the DLAG is much lower than the uncertainty when using the optimized signal control strategy since the standard deviation with DLAG control is found to be 115.2 sec compared to a standard deviation of 157.1 sec for the signalized control.

![DLAG versus Optimized Signal Control](image)

**Figure 18: DLAG COMPARED TO SIGNALIZED CONTROL**
5.5 Conclusions and Future Recommendations

This chapter proposes a novel connected automated vehicle control algorithm at interconnected intersections entitled DLAG. The proposed DLAG algorithm predicts traffic arrivals at intersections using departures from upstream intersections. The DLAG controller produced reductions in the mean travel time of 18.42% and reductions in the maximum travel time in the range of 23% compared to the traffic signal control scenario.

Future extensions to the proposed work include further testing of the multi-intersection algorithms in various scenarios with different intersection network sizes and topologies and tuning the algorithms accordingly. Using the framework of DLA for re-routing the vehicles in the network for enhancement of network mobility is another objective.
Chapter 6
Quality of Service for emergency and prioritized classes of vehicles

Abstract
The cost of delaying a vehicle to avoid a conflict at an intersection is not constant. For some vehicles, the cost of the delay is expensive whereas for others it may be cheap or even costless. It is desired to allow the control strategy to compute the delay cost and manage traffic accordingly. In this chapter, a simple assignment of vehicle to one of two discrete classes is tried to evaluate the effect of such a technique on the quality of service provided to the more critical vehicles such as emergency vehicles. The DLAG guided by the type of vehicle as class H or class L (for high cost and low cost respectively) achieved better quality of service for the critical class of vehicles with average delay reduction of 35% for class H vehicles, while the class L vehicles are not significantly affected. Class L vehicles observed around 8% increase in the average delay. For the whole system, the overall average delay is reduced by 14%.
6.1 Introduction

In chapter 5, DLAG, a distributed multi-agent approach for control of traffic of automated vehicles passing a network of intersections is successfully provided to reduce average and maximum delays. As discussed in previous chapters, minimizing the delay is a challenging task. Realistically, vehicles passing an intersection will have inevitable conflicting requests and the control scheme will have to delay some vehicles anyway. Although it is highly desired to achieve a schedule that results in zero delay for all vehicles, yet, this scenario is not expected in a real intersection unless if the flow rate of traffic is extremely low. If this is the case in real life, the question is whether the cost of delay of unit time is really the same for all vehicles competing to pass the intersection. Some trips are clearly more important than others, is it fair to assume that the loss of a passenger due to some delay is a constant? If it is not, and it is definitely not, how can a control system handle such a concept; variability of cost of delay?

In this chapter, the focus is to provide an answer to the question: In case of applying the proposed distributed control scheme in a network of intersections, can this provide more capability to give priority to special classes of emergency, service, or “willing to pay” vehicles that for some reason will need to pass the network of intersections with the minimum possible delay.

If two vehicles $V_1$ and $V_2$ are to pass a network of intersections and experience the same amount of delay, for example $d$ seconds, this may seem to be fair enough and acceptable. However, in reality this ignores a number of facts, namely:

- **Variable cost of delay unit of time**: The cost of the delay of $d$ seconds may, and most probably will, be different for $V_1$ and $V_2$. If one of the vehicles is an ambulance car holding a critically injured patient, the delay will be extremely more
expensive than the case of a vehicle of an individual or a family heading to a grocery store for shopping. It does not need to have the example of the ambulance to show this fact of the variable cost of the delay, even for the same vehicle carrying the same individual passing the same network of intersection at different times, if the vehicle start the trip, for example heading to work in the morning early enough, the cost of a delay of $d$ seconds will be much lower than if the same passenger overslept and starts his trip to work later than his usual trip start time. Putting this fact into consideration, the real objective would be to **minimize the cost of delay rather than minimizing the delay itself**.

- **Capability of control scheme to handle and communicate with each individual vehicle in the network**: Traditional intersection control techniques such as traffic signals assume that control of the traffic cannot be done at the level of individual vehicles. This is not true for the case of automated vehicles with communication and cooperation capabilities in the distributed manner proposed in this dissertation. It is feasible to give advantage to certain classes of vehicles when needed. For example, passengers can get much more encouraged to use public transportation buses when it becomes clear that the control at intersections can identify this class of vehicles and make a reasonable estimate of the cost of the delay, based on the number of passengers and their identities. Even if we have no knowledge of each individual passenger trip details in the bus, like her destination, whether or not she is late, what is her profession, etc., still the number of passengers riding the bus can give a good estimate of the increased cost of delay for the bus riders compared to a typical private vehicle holding one or two passengers.
6.2 Prioritized Vehicle Classes

The cost of delay for a vehicle can be viewed as a continuous function determined by a number of independent variables such as:

- Amount of delay in seconds
- Number of passengers
- Type of the trip (to work, to home, leisure, etc.)
- Expected arrival time compared to appointment time if any
- Passengers’ value of time
- Etc.

Since it is extremely difficult to get full knowledge of the circumstances of all trips and passengers to suggest accurate measures of the delay cost, there is a need to find ways to add flexibility to the network to provide priority to vehicles that are assigned to be among the class of vehicles suffering from expensive cost of delay. The assignment of vehicles in such a priority class can be done manually and made permanent after studying the case of the vehicle and passengers, or can be driven from known data in real-time; such as applying machine learning regression to estimate the delay cost.

One possible approach is to define a number of classes of vehicles to capture the variability of cost. Each class of vehicles is assumed to have the same cost of delay for the unit of time. This is analogous to converting the cost of delay from a continuous value that is difficult to estimate into a discrete value representing one class of vehicles. In this chapter, for simplicity, we study the case of having the minimum possible number of distinct classes, namely two classes: class H (for High cost) representing expensive delays, such as emergency vehicles, and class L (for Low cost)
for vehicles that can be considered less sensitive to delays and thus having a lower delay cost. Adding more than two classes of vehicles with similar delay costs should be straightforward, and is in fact one of the planned extensions of this research in the future.

The 4x4 grid network previously used in chapters 4 and 5 is used with the same traffic levels. The same setup used for DLAG experiments in chapter 5 is used to study the effect of giving priority based on the classes of vehicles.

The objective is to study:

- Is the DLAG algorithm able to remarkably enhance the quality of service for vehicles of class H that are more sensitive to delays?
- What is the average amount of savings of delays for class H vehicles?
- If quality is improved or guaranteed for class H vehicles, how are the vehicles of the lower class L affected?

One possible and expected way to assign vehicles to classes H and L in the near future can be expected to occur by allowing passengers pay more when the cost of their delay is higher. A passenger who is late or needs to guarantee his arrival time can pay a fixed fee to rise from class L to class H.

Currently, in many transportation networks, use of some shorter routes in congested cities may require the payment of a toll fee, which is similar to the idea of having different classes of vehicles depending on the delay cost. The main difference is that the introduction of more intelligence to the vehicles and the networks should trigger the move away from enforcing tolls for the use of specific physical resources like a less congested lane or road section like a bridge, and toward the use of paid better service within the same network.
without requiring different routes. Rather, the goal will be achieved through an intelligent
control scheme that can incorporate the various classes of vehicles while scheduling
vehicles’ passage through the intersection and resolving potential conflicts accordingly.

6.3 Simulated Experiments and Results

The objective of this section is to study the potential power of a distributed control strategy
such as the DLAG algorithm to handle the two vehicle classes with different priorities. In this
experiment, the vehicles allowed during the simulation are assigned randomly so 80% of the
vehicles are labeled as class L vehicles while the lower percentage of 20% of vehicles are labeled
as class H vehicles.

The same 4x4 grid network of intersection as in Figure 7 is used. The traffic is allowed with
the same O-D levels as in Table 1. A total of 5040 vehicles are allowed into the network in the first
half of the hour, and the simulation continues for one more half of the hour to ensure all vehicles
exited the entire network. First the DLAG algorithm is used with all vehicles as one class as usual,
then the same experiment is simulated again with the 80% class L and 20% class H applied. The
scheduling in the latter case is done in a preemptive manner where the lanes holding more vehicles
are given priority. When no vehicles of class H are in the intersection zone, or when the number
of class H vehicles is the same in each direction, the regular prioritization based on the Godunov
estimates is solely used.

The results demonstrated notable enhancement for the delay of vehicles of class H
compared to the case when all vehicles are treated equally. The delay reduction experienced by
class H vehicles in this case is reduced by 35% on average. This is shown in Figure 19.
In order to test the statistical significance of the achieved reduction in delay for class H vehicles, a pooled t-test is used to compare the two means of the delay before and after giving the priority to class H vehicles. The null and alternative hypotheses are:

\[ H_0: \text{Mean}_{\text{without QoS}} \leq \text{Mean}_{\text{with QoS}} \]

\[ H_a: \text{Mean}_{\text{without QoS}} > \text{Mean}_{\text{with QoS}} \]

The p-value is found to be less than 0.0001, and we reject the null hypothesis; therefore, we conclude that the mean delay after applying the quality of service for class H vehicles is significantly less than the mean delay of DLAG when all vehicles are treated equally.

For the majority of vehicles (80% class L vehicles), it is interesting to note that the average delay is almost the same before and after applying the modified algorithm as shown in Figure 20. Considering the class L vehicles, the average delay is increased by around 8% when the class H vehicles are favored when passing the conflict area of the intersections. The retardation is less serious when considering the enhancement for more critical vehicles. When combining the delay experienced by all vehicles of both classes, it is evident that the overall mean delay is reduced by 14% after applying the quality of service compared to the delay of DLAG with all vehicles treated equally.
Figure 19: Delay of 20% class H vehicles
Conclusions and Future Recommendations

The results of labeling the vehicle according to the cost of the delay are promising and suggest that DLAG can be very powerful in scheduling vehicles at the intersection based on the actual cost of a delay. The vehicles of lower cost of delay, class L, are not remarkably retarded by the priority given to their class H counterparts.

The planned extension to the work in this chapter includes the study of the effect of giving the priority using other schemes. For example, it is planned to study the case of prioritizing based on the number of passengers in the vehicle so that congested cities can encourage public transportation or carpooling.
Another important extension is to design an algorithm that can automatically predict the cost of the delay based on available information about the vehicle, trip, and passenger, in addition to the traffic history information.

The use of more than two classes and running more extensive experiments is also planned.
Chapter 7

Distributed Intersection Vehicle Control in the INTEGRATION Micro-Simulation Model

Abstract

The DIVC system, a version of the IIZA algorithm for isolated intersections, is incorporated into the INTEGRATION microscopic traffic assignment and simulation software. Adding the proposed algorithm to INTEGRATION makes it possible for future researchers in the field to easily run, evaluate, and compare the proposed algorithms to alternative intersection control strategies. DIVC is compared to signalized control, roundabouts, and stop signs for an intersection at various levels of congestion. The results show that the DIVC control outperformed traditional traffic signal control strategies by reducing delays by about 20%, fuel consumption by 4.4%, and CO₂ emissions by 4.2%. DIVC also slightly outperformed roundabouts for higher demand levels by reducing delays by 5%, fuel consumption by 1.7%, and CO₂ emissions by 7.6%.
7.1 Introduction

The algorithms proposed earlier in this dissertation have been implemented in MATLAB. All simulated experiments to compare the proposed algorithms are also coded using MATLAB. Algorithms in this dissertation can be implemented using any platform chosen by a researcher. However, the authors are keen to implement the proposed algorithms as added functionality in the INTEGRATION micro-simulation model to ensure that other researchers in the field are capable of using and testing them. This step allows the research community to have easy access to both isolated intersection and networked intersection algorithms, with the hope that more researchers can contribute to enhancing and extending the algorithms or provide their own novel techniques to solve this challenging problem.

7.2 INTEGRATION Micro-Simulation Model

The INTEGRATION model is a fully microscopic traffic simulation model. The INTEGRATION model is a trip-based model that is able to trace individual vehicle movements (both lateral and longitudinal movements) from a vehicle’s origin to its destination at a level of resolution of one status update every 0.1 seconds. This software tool has been used by transportation researchers and traffic engineers for almost three decades since its development in the mid-eighties.

INTEGRATION provides modeling of various features, such as vehicle dynamics, car-following, and collision avoidance, in addition to many advanced unique features such as modeling of traffic signal controls, eco-routing strategies, and emission dispersion.

The use of this powerful model to simulate traffic in a network is fairly easy. The model requires the user to provide five fundamental data input files. These input files describe the characteristics of the network nodes, links, traffic signals if any, incidents if any, and traffic
demands in the form of O-D departure rates for each possible O-D pair within the network. The input files are text files that can be entered easily through any text editor. Along with the fundamental input files, INTEGRATION allows the user to provide other optional input files for advanced features such as modeling of Intelligent Transportation Systems, advanced signal timing options, and overriding vehicle dynamic characteristics. INTEGRATION provides on-screen animation concurrent to the execution of actual model logic.

7.3 DIVC Implementation in INTEGRATION

The distributed intersection vehicle control (DIVC) is a version of the IIZA algorithm. DIVC has been implemented using FORTRAN and added as a function in INTEGRATION. The networked intersection algorithms proposed, NIZA, DLA, and DLAG, are expected to be added to INTEGRATION in the future. This step is very important for a number of reasons, including:

- Capability of comparing the proposed algorithms in this dissertation to several control mechanisms that are built into INTEGRATION.
- Ease of extracting useful outputs such as the emissions, fuel consumption, etc.
- Allowing future researchers to use the proposed algorithms, extend and enhance them, and compare them to their own proposed algorithms in order to reach better solutions to the increasingly challenging problem of traffic at intersections.

The first step is the implementation of a version of the proposed algorithm for isolated intersections IIZA. In INTEGRATION, the implemented version of the algorithm has been named DIVC which stands for Distributed Intersection Vehicle Control. For users of INTEGRATION who need to run this algorithm, a new optional input file divc_input.dat is to be used.
The file “divc_input.dat” is an optional file that permits the user to activate the distributed intersection vehicle control system to allow connected or automated vehicles to pass an intersection without signals. This file, if it is to be used, should be presented in the same input folder as all other input files. The format of the file is illustrated in Table 2. Table 3 shows one sample input file.

The distributed intersection vehicle control (DIVC) system is designed to allocate time slots for connected and automated vehicles to pass intersections without regulation by signals. The system requires that all vehicles be equipped with wireless communications, and they are able to coordinate with each other to search for the best time to pass the intersection and minimize travel delays. This system is a fundamental platform to develop a fully connected and automated vehicle system in the near future.

- Upstream and downstream links: The links are defined to identify the approaches of one intersection (up to 12 approaches). The priority of vehicles passing the intersection is given based on the approaches.
- Passing time of movements: The passing time of the movements (up to 3) is defined to set the weight for different movements (left, right and through). Generally, the passing time of left or right turns is larger than the through traffic.
Table 2: Description of the file 'divc_input.dat'

<table>
<thead>
<tr>
<th>Line 1</th>
<th>Column #</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>File title</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>Number of intersections controlled by the DIVC system</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>Flag indicating whether the DIVC system is applied 1: activated; 0: not</td>
</tr>
<tr>
<td>3+</td>
<td>1</td>
<td>Line identification number</td>
</tr>
<tr>
<td>2 - 5</td>
<td></td>
<td>Upstream link numbers of the intersection 2: West, 3: South, 4: East, 5: North</td>
</tr>
<tr>
<td>6 - 9</td>
<td></td>
<td>Downstream link number of the intersection 6: West, 7: South, 8: East, 9: North</td>
</tr>
<tr>
<td>10 - 13</td>
<td></td>
<td>Passing time for different movements (seconds) 10: Left turn, 11: Right turn, 12: Through</td>
</tr>
<tr>
<td>14</td>
<td></td>
<td>Updating interval of the DIVC system (second)</td>
</tr>
</tbody>
</table>

Table 3: Sample DIVC system input file

```
Distributed intersection vehicle control system
1 1
1 2 14 6 10 7 11 3 15 1 1 1 10
```

7.4 Simulated Experiments and Results

To study the DIVC system, various simulations are performed for an intersection considering various levels of congestion. In these experiments, the DIVC system is compared to the three most widely used intersection control strategies, namely traffic signals, roundabouts, and stop signs. The intersection has a major roadway and a minor roadway, where the arrival rates on the minor roadway are always half the flow of the major roadway. The flows along the major road range from 500 to 1600 veh/h. The base saturation flow rate of the roadway is 1700 veh/h.
The results of running the DIVC system are compared to the results when the intersection is controlled by each of the three other control strategies: traffic signals, roundabouts, and stop signs. In the case of traffic signals, it has been noted that the signal performs better when using two phases rather than three phases, so two phases are used for the experiments in this chapter. For each experiment, the same input traffic is introduced once to each of the four compared strategies, and INTEGRATION output file 15 is used to extract and compare the delay, fuel consumption, CO₂ emissions, and number of stops at the intersection for each scenario using one of the tested control strategies.

7.4.1 One-Lane All Through Traffic

Assuming an intersection with a single lane in each approach, each of the four control strategies is run twelve times, where the flow is increased on the major road by 100 vehicles every run (vehicles are admitted in the first half of the hour, and the second half of the hour of simulation is reserved to ensure all vehicles exit the transportation network). The minor road runs half the number of vehicles running on the major flow for each run. Figure 21 shows the mean delay for different traffic volumes using each of the four strategies. The DIVC system achieves the best reduction in the mean delay compared to the other control strategies, especially for heavier traffic scenarios. It is clear from Figure 21 that using stop signs is remarkably inefficient even with an intersection of only one lane. To better demonstrate the comparison between the three more efficient strategies, the following figures will focus only on them, but all the experiments presented in this chapter included the four control strategies. Figure 22 provides a better demonstration of DIVC compared to roundabouts and signalized control, especially for higher flows. For the simulated scenario with maximum traffic, the DIVC achieved a 5% average reduction in delays compared to the roundabout strategy, and a 20.5% average reduction in delays compared to the
signalized control. It is also noted in Figure 22 that the mean delay is almost the same for both DIVC and the roundabout control, and the slight reduction achieved by DIVC is more evident as the traffic through the intersection increases.

![Graph showing Mean Delay for Through Traffic in One-Lane Intersection](image)

**Figure 21: Mean Delay for Through Traffic in One-Lane Intersection**

One of the benefits of running the simulations in INTEGRATION is the important byproducts provided, such as fuel consumption, CO\(_2\) emissions, and number of stops. Figure 23 shows the fuel consumption (in L) for each of the four control strategies. Again, the results from the stop sign control are omitted to better demonstrate the DIVC fuel consumption compared to the roundabout and signalized control intersections, as shown in Figure 24. Considering the highest flow scenario, the use of DIVC slightly reduces the average fuel consumption by 1.7% compared to the use of roundabouts, whereas it achieves an average reduction of 4.4% compared to the signalized control.
for the same scenario. It is also clear in Figure 24 that the three strategies result in similar fuel consumption for low traffic demand levels, and the curves start to separate as the traffic flow increases.

![Figure 24: Comparison of Fuel Consumption for Different Strategies](image)

> **Figure 24**: Mean Delay for Through Traffic, One Lane (No Stop Sign)

As for the CO$_2$ emissions, the same trend is observed when comparing the DIVC to roundabouts and to signalized controlled intersections, as shown in Figure 25. For the maximum traffic scenario, DIVC reduced the mean CO$_2$ emissions by 7.6% when compared to the use of a roundabout, and the average emissions were reduced by 42% when compared to the signalized...
control scenario. It is also clear how the differences between the three strategies increase as the traffic demand increases.

Figure 23: Mean Fuel Consumption for Through Traffic, One Lane

Finally, Figure 26 shows the mean stops for vehicles using each of the four control strategies. We notice that the increase in the mean stops for DIVC as the flow increases is less than the increase in the mean stops for both the roundabout and the signalized control. The mean stops for the roundabout is less than the DIVC for the lower traffic scenarios, whereas the DIVC mean stops becomes less than the roundabout for the higher flow scenarios.

In summary, the results suggest a general improvement when the DIVC system is used for all the measured quantities, especially for higher traffic demand levels.
Figure 24: Mean Fuel Consumption for Through Traffic, One Lane, No Stop Sign

Figure 25: Mean Emissions for Through Traffic, One Lane Intersection
Figure 26: Mean Stops for Through Traffic, One Lane Intersection

7.4.2 Two-Lane All Through Traffic

All the experiments simulated for the one lane intersection are repeated for a bigger intersection with two lanes in each direction. It is still assumed that there is only through traffic with no right or left turns. The twelve scenarios for each control strategy are run with the increased flow rate as discussed in the previous subsection. Figure 27 presents the mean delay for the DIVC, roundabout, and signalized control. It is clear that the mean delay for the two-lane intersection is very similar to the case of the one-lane intersection as depicted in Figure 22. The mean delay is reduced when using DIVC compared to the roundabout and signalized control. The reduction in the delay is more pronounced for the signalized control scenario. Compared to the roundabout, the two strategies seem to result in almost the same average delay for the lower traffic demand levels,
and, as the traffic demand increases, the DIVC outperforms the roundabout in terms of delay reduction. For example, for the maximum experimented flow scenario, the DIVC achieved a mean delay reduction of 4.8% compared to the roundabout, and a mean delay reduction of 21.8% compared to the signalized control. These figures are very similar to the results of the single lane case.

Similarly, the trends for fuel consumption, CO$_2$ emissions, and mean stops are very similar to the one-lane intersection case as long as the traffic is restricted to only through traffic. Figure 28 shows the mean fuel consumption for the DIVC, roundabout, and signalized control for the two-lane intersection. For the maximum flow scenario, the DIVC achieves an average reduction of 1.7% compared to the use of roundabouts, and an average reduction of 4.4% compared to the signalized control, which are the same values depicted earlier in Figure 24.
Similarly, Figure 29 shows the mean CO₂ emissions in grams for the three control strategies. For the maximum flow scenario, DIVC reduces the average emissions by 12.3% compared to the roundabout and reduces the average emissions by 42% compared to the signalized control. Again, these figures are very close to the one-lane case. Also, Figure 30 demonstrates the mean stops for the four control strategies and all flow rates when using the two-lane intersection. Again, the same conclusions of the one-lane case can be seen; the DIVC is better than the other three strategies for higher flow rates, and the increase in the mean stops with the increase in the flow rate is less for the DIVC than the roundabout and signalized control.
Figure 29: Mean Emissions for Through Traffic, Two-Lane Intersection

7.4.3 One-Lane Through and Right Turns Allowed

All experiments discussed in earlier subsections are run again allowing both through traffic and right turns. For each experiment, 75% of the traffic is through, while the remaining 25% is made to turn right. First, the one-lane intersection is used, and then all the experiments are run using the two-lane intersection.

Considering the mean delay, the DIVC outperformed both the roundabout and signalized control as presented in Figure 31, especially for high traffic demand levels. For the maximum flow scenario, the DIVC achieved a reduction of approximately 14% compared to the roundabout in mean delay, and a reduction of about 30% compared to the signalized control.
Figure 30: Mean Stops for Through Traffic, Two-Lane Intersection

Similarly, Figure 32 shows the enhancement in mean fuel consumption caused by the use of the DIVC. For the maximum flow scenario, the DIVC reduces the mean fuel consumption by 0.72% compared to the roundabout, and by 4.76% compared to the signalized control. Figure 33 presents the mean CO₂ emissions, for the maximum flow scenario; DIVC reduces the mean emissions by 1.1% compared to the roundabout and by 5.15% compared to the signalized control.
Figure 31: Mean delay (Through and Right Traffic) (Two-Lane)

Figure 32: Mean Fuel Consumption (Through and Right) (Two-Lane)
7.5 Conclusions and Future Recommendations

The proposed DIVC controller was successfully implemented in the INTEGRATION software and its performance was compared to three common intersection control strategies, namely: traffic signal controls, roundabouts and all-way stop controls. The testing entailed loading traffic demands ranging from 500 to 1600 veh/h along the major road and half that demand on the minor road (the demand was doubled for the dual-lane approaches). The base saturation flow rate for the approaches was 1700 veh/h/lane. The tested scenarios included single lane through traffic, dual lane through traffic, and dual lane through and right turning traffic. The results demonstrate that the proposed DIVC algorithm produces significant benefits compared to the signalized control
and all-way stop-sign control scenarios. Compared to the roundabout, the DIVC outperformed in delays, fuel consumption, and emissions for higher traffic demand levels. For lower traffic demand levels, however, both strategies produced similar trends. Generally, the results of the performed simulations suggest a general improvement when the DIVC system is used for all the measured quantities, especially for high traffic demand levels.

For single lane intersections allowing only through traffic, the DIVC algorithm achieved a 5% average reduction in delay compared to roundabout control and a 20.5% average reduction in delay compared to a signalized controlled intersection. The DIVC algorithm slightly reduced the average fuel consumption by 1.7% compared to the roundabout control considering the highest flow scenario, whereas it achieved an average reduction of 4.4% compared to the signalized control. The DIVC algorithm reduced the mean CO₂ emissions by 7.6% when compared to a roundabout and the average emissions were reduced by 42% when compared to a signalized controlled intersection. Similar results were observed for the dual-lane intersection approaches and when right turning traffic was introduced.

The future plan includes the full implementation of the DIVC algorithm in the INTEGRATION software to work for all types of movements including through, right, and left turning vehicles. The DIVC is to implement the connected intersections algorithms proposed in earlier chapters. The implementation of the DLAG algorithm in INTEGRATION is a priority to allow researchers in the field to use the technique and further the research on how to improve it.
Chapter 8
Conclusions and Recommendations for Future Work

8-1 Summary of study objectives

The dissertation develops and tests a number of fully-distributed heuristic connected automated vehicle control algorithms at roadway intersections. The proposed algorithms have the following objectives:

- Minimize the delay experienced by CAVs proceeding through intersections by predicting the formation of queues and managing traffic to balance queues.
- Ensure the safety of CAVs by preventing the occurrence of vehicle crashes. This is a major concern with the increased percentage of car accidents caused by human driver errors occurring at intersections. This should accelerate either the use of fully automated vehicles, or at least intersection cooperative cruise control iCACC to take control around intersections assuming vehicles are capable of exchanging communication reliably.
- Ensure that the system operates in real-time. Several research efforts have taken place in the past few years. One of the major objectives during the design of the control strategies proposed in this dissertation is to ensure that the operation of the intersection control is realistic and that the computational complexity and communication overhead are feasible for future on-road applications in real-time.
- Develop a system that requires minimal infrastructure investment. Currently most of the existing intersections are non-signalized making the proposed system a very viable solution.
• Allow the traffic control system to target vehicles around the intersection that are critical and thus require some guaranteed level of quality of service, such as ambulances, where the delay of arrival is more costly.

8-2 Conclusions

The designed fully-distributed and heuristic multi-agent systems and algorithms proposed in this dissertation have been implemented, and several experiments were simulated, as discussed throughout the dissertation, to evaluate the success of the algorithms in managing traffic to maximize mobility and avoid vehicle collisions, and to compare the novel techniques to alternative intersection control strategies. The following are some of the conclusions of the dissertation:

• The Isolated Intersection Zone Algorithm IIZA successfully allowed for the management of traffic and prioritized passage in case of a possible conflict based on the queue length. Compared to a FIFO scheme that reserves time in the same order the controller receives a request, IIZA achieved delay reductions of up to 40%. The maximum delay experienced by vehicles in the experiments is also significantly reduced compared to a FIFO scheme.

• NIZA and DLA are two proposed algorithms for connected intersections that demonstrated the added value of using knowledge about traffic at neighboring intersections to improve the performance of a subject intersection. NIZA is found to achieve a 46% reduction in the average delay of vehicles compared to IIZA. DLA achieved a 79% reduction in average delay compared to IIZA. The maximum delay experienced by vehicles is reduced by 32% when using NIZA and by 58% when using DLA.

• DLAG proved to be efficient at reducing delays in a network of intersections with better estimations of traffic in unseen sections of the road (sections where the intersection control does not require vehicle communication). DLAG significantly outperformed optimized
signalized control with a significant reduction in travel time. Specifically, the mean travel time was reduced by 18.42%. The maximum travel time for DLAG also experienced a 23% reduction compared to the maximum travel time seen using the optimized signalized control.

- A modified version of DLAG aimed at enhancing the quality of service for emergency vehicles by classifying vehicles based on the cost of the delay was implemented and tested. Results of labeling the vehicle according to the cost of the delay are promising and suggest that DLAG can be very powerful in scheduling vehicles at the intersection based on the actual cost of delay.

- The addition of the proposed intersection control to the INTEGRATION model was successful by implementing the DIVC, which can easily be used to simulate the isolated intersection algorithm, in the hope of completing the implementation and adding the networked intersections functionality in the near future.
  - Generally, the results of the performed simulations suggest an improvement when the DIVC system is used for all the measured quantities especially when the traffic demand increases.
  - For single lane intersections allowing only through traffic, the DIVC algorithm achieved a 5% average reduction in delay compared to roundabout control, and a 20.5% average reduction in delay compared to a signalized controlled intersection. The DIVC algorithm slightly reduced the average fuel consumption by 1.7% compared to the roundabout control considering the highest flow scenario, whereas it achieved an average reduction of 4.4% compared to the signalized control. The DIVC algorithm reduced the mean CO₂ emissions by 7.6% when compared to a
roundabout and the average emissions were reduced by 42% when compared to a signalized controlled intersection. Similar results were observed for the dual-lane intersection approaches and when right turning traffic was introduced.

8.3 Future Plans and Recommendations for Further Research

Several paths can be followed to extend the work proposed in this dissertation.

- A crucial step in near future is the complete addition in INTEGRATIONS software of all the proposed algorithms, IIZA, NIZA, DLA, DLAG, and the combined DLAG with quality of service. This will increase the possibility of continuous study of the algorithms and open many doors for researchers to compare the algorithms to their proposed systems.
- The main objective function in the proposed algorithms is to minimize vehicle delays. The implementation in INTEGRATION will allow for a more sophisticated optimization function that incorporates more parameters within the optimization framework. One important goal will be to minimize fuel consumption and CO₂ emissions for eco-driving.
- Another interesting extension of the proposed algorithms is the study of designing a priority algorithm based on the number of passengers; thus, a bus will be given a priority of several multiples of a regular vehicle, and carpooling will have double or triple the priority of a single passenger vehicle. Comparing this to other techniques in the literature that encourage public transportation or carpooling may be beneficial.
- Most algorithms in the literature making use of vehicle communication assume perfect communication which is not typical for current wireless communication technology with possibilities of delays in the communication networks, interference and collisions of messages and need for retransmission, etc. One important extension entails testing the applicability of the proposed algorithms for real-life traffic to include realistic conditions.
of the wireless communication network such as packet loss, noise, interference, message delays, etc. It is important to determine if any deteriorated conditions of the communication network can lead to vehicle collisions or non-optimized traffic. As part of this study, the communication parameters can be quantified such as power need for communication messages, minimum values for communication network parameters (packet loss, delay, etc.) to keep the transportation algorithm working correctly.

- For the algorithms that require intersection agents; DLA and DLAG, one useful extension could be to make use of the layer of networked intersections to run an algorithm to be designed to dynamically re-route the vehicles to pass through less-congested intersections on route to their destination. One possible technique is that an intersection periodically receives traffic states of other intersections in the network and broadcasts this image to the vehicles in its zone. Upon getting the network traffic image, a vehicle can evaluate an estimated delay for a number of different possible routes and, with some probability, changes its route to a less congested one.

- One important issue that need to be studied is lag impacts. The simulated studies in the dissertation did not consider the possibility of delays in communication or lost information. One way this can be studied is through running simulations that enforce these lags and determine if this may lead to any unsafe states.

- One important extension is to perform an extensive formal verification of the proposed algorithms. Model checking [60] of the state machines representing an intersection controlled by the proposed algorithms is important to prove that the system is collision-free. The use of any of the available tools [61] for model checking is very useful to verify the safety and liveness [62] of the control strategies proposed in the dissertation.
In this dissertation, the Godunov scheme has been used to formalize the estimation of density at road sections around intersections. One possible alternative that may be studied is the use of queueing theory to formalize the calculation of queue lengths at various approaches of an intersection and study if this can lead to better performance in terms of delay reduction.
References


[10] I. A. Budisteanu, "Using Artificial Intelligence to create a low cost self-driving car."

[12] P. Bansal and K. M. Kockelman, "ARE AMERICANS READY TO EMBRACE CONNECTED AND SELF-DRIVING VEHICLES? A CASE STUDY OF TEXANS."


[26] L. Bruni, A. Colombo, and D. Del Vecchio, "Robust multi-agent collision avoidance through scheduling."


