MICHIGAN OHIO UNIVERSITY TRANSPORTATION CENTER

Alternate energy and system mobility to stimulate economic development.

Report No: MIOH UTC TS45 2012-Final

A MULTI-DIMENSIONAL MODEL for VEHICLE IMPACT on TRAFFIC SAFETY, CONGESTION, and ENVIRONMENT

Final Report

UDM Team
Nizar Al-Holou, Ph.D.
Utayba Mohammad, Dr. Eng.
Mohammad Arafat
Malok Alamir Tamer
Mohamad Abdul-Hak
Department of Electrical and Computer Engineering
University of Detroit Mercy
4001 W. McNichols Road
Detroit, MI 48221

WSU Team
Syed Masud Mahmud, Ph.D.
Electrical and Computer Engineering Department
Wayne State University
Detroit, MI 48202

Report No: MIOH UTC TS45 2012-Final

FINAL REPORT

Developed by:

Dr. Nizar Al-Holou
Principal Investigator, UDM
alholoun@udmercy.edu

Dr. Utayba Mohammad
mohammut@udmercy.edu

Graduate Research Assistants:

Mohammad Arafat
Mohammad Abdul-Hak
Malok Alamir Tamer

WSU Team:

Dr. Syed Masud Mahmud
Associate Professor
Co-Principal Investigator, WSU
smahmud@eng.wayne.edu

SPONSORS

This is a Michigan Ohio University Transportation Center project supported by the U.S. Department of Transportation, the University of Detroit Mercy and Wayne State University.

ACKNOWLEDGEMENT

This work was supported by the U.S. Department of Transportation through the University Transportation Center (UTC) program, MIOH UTC, at the University of Detroit Mercy. The authors would like to express their sincere appreciation to the project sponsors for their support.

DISCLAIMERS

A MULTI-DIMENSIONAL MODEL for VEHICLE IMPACT on TRAFFIC SAFETY, CONGESTION, and ENVIRONMENT

Executive Summary

The Intelligent Transportation System (ITS) has recently received great attention in the research community. It offers a revolutionary vision of transportation, in which a full-scale communication scheme between vehicles (V2V) and vehicles and infrastructure (V2I) is introduced. The ITS vision reflects three main areas of transportation improvement: enhancing the traffic flow and mobility of vehicular transportation, improving the active and passive safety of vehicles through V2V and V2I communication, and providing a platform that can address the environmental challenges of transportation systems from a macroscopic perspective. To implement this vision research has to be conducted on two layers: the communication layer and the application layer. The former is concerned with providing proper V2V and V2I communication at the physical, MAC, and network layers, while the latter is concerned with using the communicated data in V2V and V2I interactions to achieve the aforementioned three-point vision of ITS systems.

This research targets special cases at both the communication and application layers. In chapter 1, an adaptive traffic light application that improves mobility and safety is developed. This application adopts Webster's equation as a basis to determine the red and green time cycles. It integrates the dynamic traffic information into Webster's calculations and extends the green time or shortens the red time of traffic lights at an intersection to maximize the traffic flow at the corresponding junction. The developed system is simulated and evaluated against the conventional pre-timed traffic lights and smart pre-timed traffic lights, and the results show great improvement in controlling delay times, travel times, and traffic flow volume. In chapter 2, an environment-friendly vehicle routing application is developed. This application introduces a new methodology to collect traffic data through the ITS communication scheme, and utilize this data to route vehicles in the most collectively fuel-efficient way. The estimated fuel consumption over road segments is used as the main criteria to calculate the best route for a vehicle, and is updated continuously through ITS message exchanges. The new routing method is evaluated through simulation and is proven superior to the conventional static fastest path routing methods in terms of waiting times, travel times, fuel consumption, and CO_2 emissions. Finally, chapter 3 targets the Medium Access Control at the communication layer of ITS. It introduces an Intersection Warning Channel Access Priority (IWCAP) protocol that would guarantee warning drivers of possible collisions as they approach an intersection. This protocol utilizes one omni-directional antenna per vehicle and one DSRC channel for the intersection warning system. It provides priority and fairness for all vehicles approaching the intersection to transmit their conditions to other vehicles. The protocol analysis shows that drivers can avoid a collision if a warning message is received within 240 m from an intersection given a communication range of 500 m and a speed of 96 Km/h on a wet pavement, if the warning message is received within 0.2 s after joining the wireless network.

Table of Contents

List of Figures

List of Tables

Chapter 1: Adaptive Traffic Light Controlling Methodology under Connected Vehicles Concepts (UDM Research Team)

1.1. Introduction

The first four-way three-state traffic light was invented by police officer "William Potts" in Detroit, Michigan in 1920. Ever since, traffic lights have been one of the most important means of traffic management, proving their efficiency in organizing traffic flows and reducing traffic jams. Historically, the concepts of connected intersection control and automatic traffic lights are as old as the traffic light itself. The first interconnected traffic signal system was installed in Salt Lake City in 1917, with six connected intersections controlled simultaneously from a manual switch. The first automated control of interconnected traffic lights was then introduced in March 1922, in Houston, Texas [1]. Since then, many methods have been proposed to evolve traffic light controllers in order to achieve adaptive and dynamic functionality. Some of the proposed systems used metal detectors and computer vision to feed in traffic information to the controllers. However, those systems lacked accuracy because they were based on inaccurate sensors or unreliable computer vision algorithms, hence the collected data was inaccurate and the controller decisions were consequently inaccurate. The answer to the data collection problem at an intersection came with the introduction of the Intelligent Transportation System (ITS). In ITS vehicles can communicate with other vehicles and with roadside infrastructure. This communication scheme provides means to convey all the necessary traffic information to traffic light controllers, hence solving the data collection problem and allowing researchers to focus on the intelligent behavior of traffic lights.

The literature on real-time adaptive traffic lights can be classified in two main categories. The first is focused on the algorithms used to determine cycle length and green phase timing, and the second is focused on introducing new ways of collecting traffic data. In the data collection domain, B. Zhou et al. propose an adaptive traffic light control algorithm that uses a wireless sensor network to collect traffic data and determine the sequence and length of traffic light phases [2]. The authors assumed the intersection can only be in one of 16 cases, and they used a mathematical model to determine the intersection's next case and the period over which it should persist. On the other hand, D. T Dissanayake et al. proposed an algorithm for vehicle detection based on a Magneto-Resistive sensor [3]. Also, K. Al-Khateeb et al. proposed a real-time dynamic traffic light sequence determination algorithm, but this time using RFID technology to collect real-time traffic information [4]. In the control algorithm development domain, much research has been presented on very complicated algorithms that incorporate the learning abilities of artificial neural networks with the decision making of fuzzy expert systems [5].

Although adaptive traffic light systems have been researched rigorously, only a few papers considered the connected vehicles concepts (i.e. vehicle-to-vehicle and vehicle-to-infrastructure (V2V/V2I) communications) as the source for real-time traffic information. M. Ferreira et al. present a new concept of traffic management at intersections, using only V2V communications [6]. The proposed algorithm requires neither roadside equipment (RSE), nor a traffic light. The algorithm is only based on communication between vehicles at the same intersection, and the traffic light is replaced with an internal traffic light presented on an internal display in each vehicle. Another V2V/V2I utilization was presented by V. Gradinescu et al., in which V2V/V2I messages are used to collect real-time information about the traffic conditions around the intersection, and then an algorithm based on the well-known Webster's equation is used to determine cycle length as well as green time splitting [7].

In this chapter, an adaptive traffic light controlling methodology that uses V2V/V2I communications is developed and evaluated. Traffic information is collected from V2V/V2I message exchanges, and the cycle length and green times are determined using an algorithm based on Webster's equation. The remainder of this chapter classifies the traffic light control systems known in the literature and presents a new algorithm to determine the traffic light's cycle length and green times. A simulation model is then created, and the developed algorithm is evaluated.

1.2. Classification of Traffic Light Controlling Systems

A traffic light can be defined by three major elements which are, cycle length, green time splits, and relation to the surrounding environment. Accordingly, traffic lights can be classified into three main categories: pre-timed, actuated and adaptive.

In pre-timed traffic lights, cycle length and green time splits are pre-determined before the traffic light is put in operation. In addition, the traffic light does not respond to any sudden changes in the surrounding environment. This is the most basic and simple form of a traffic light. Further enhancements were done by defining different programs (cycle length and green times) for different times of the day, or day of the week. Historical data about traffic flow were used to find peak hours and assign suitable cycle length and green times accordingly.

Actuated traffic lights form an enhanced version of pre-timed traffic lights. In actuated traffic lights, cycle length and green times are fixed, however the traffic light's ability to respond to events in the surrounding environment is introduced by adding sensors on some, or all, of the roads controlled by that traffic light. Thus the main difference between pre-timed and actuated is the ability to respond to some events from the surrounding environment. For example, in an intersection comprised of a major road crossed by a secondary road, actuated traffic light can be used to extend green on the main road as long as no traffic is present at the secondary road; in case of incoming traffic from both directions, the traffic light will work based on the pre-timed program previously defined for that time of the day and day of the week.

Finally, adaptive traffic lights represent a revolution in the traffic management domain that is inspired by the framework of the intelligent transportation systems. In adaptive traffic lights, the cycle length and green times are calculated based on real-time traffic data from the surrounding environment. Sub-categories of the adaptive traffic lights can be further defined based on the traffic data collection techniques and the underlying algorithms that are used to determine and update the traffic light cycle. Figure 1 shows the different sub-categories of adaptive traffic lights based on data collection technique and data processing algorithm.

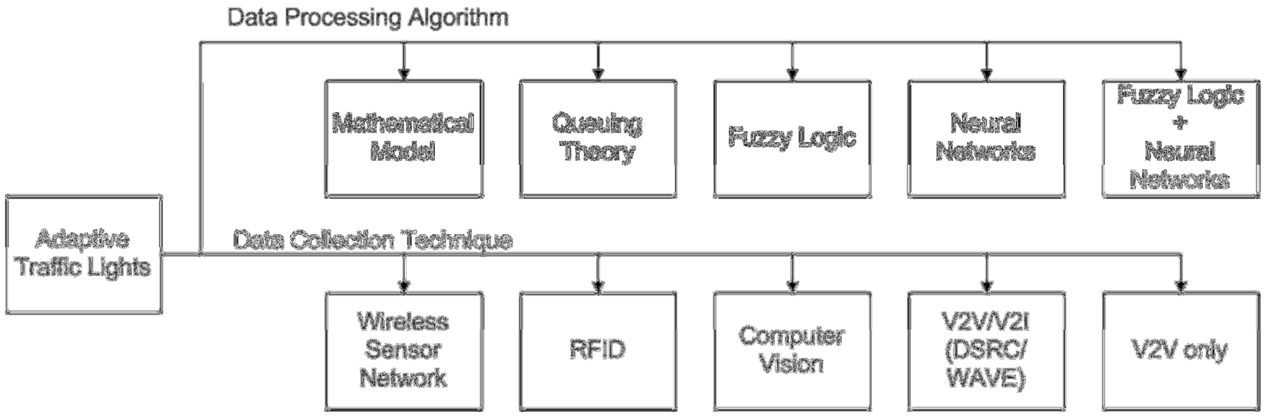

Figure 1. Different Categories of Adaptive Traffic Lights

1.3. The Adaptive Traffic Light Control Algorithm

Algorithms for controlling adaptive traffic lights can be simple (simple mathematical model), or very complicated (combination of fuzzy logic and artificial neural networks). Webster's equation is a tool used to determine the optimal cycle length for a traffic light according to traffic flow information and the lost time during yellow times and red times. Originally, the traffic flow information was based on historical data. However, since vehicles can communicate with each other and with the infrastructure using V2V/V2I communications, data can be collected from vehicles and fed into Webster's equation to come up with optimized cycle length and green times dynamically. This approach will result in optimized performance and increased throughput of controlled intersections. Equation 1.1 shows Webster's equation in which demand levels are represented by a factor called the critical flow ratio, which is the ratio of the current flow rate captured from the exchanged messages, over the road's saturation flow rate.

$$C_0 = \frac{1.5 \cdot L + 5}{1 - \sum_{i=1}^{i=n} y^i} \quad (1.1)$$

Where,

C_0 is optimal cycle length [sec].
L is lost time (yellow + all-red) [sec].
y^i is critical ratio of road segment group i.
n is number of road segment groups.

Road segment group: is a group of road segments on which incoming flow can access the intersection simultaneously.

The idea is to calculate cycle length and green time splits every cycle so that the system can dynamically adapt to changes in traffic flow. This algorithm uses information from messages exchanged between vehicles. These messages contain information about the vehicle's position, speed, acceleration, and many different parameters. Since those messages are received every simulation step, a special procedure for accumulating the information over a complete cycle length is needed. Therefore, a procedure proposed by S.-F. Cheng et al. is adopted [8].

The procedure calculates the estimated flow for each road segment group as follows:

$$v^i = f^i + 4q^i \qquad (1.2)$$

Where,
v^i is the estimated flow for road segment group i.
f^i is the exponentially smoothed average incoming flow on road segment group *i.*
q^i is the exponentially smoothed average size of the standing queue on road segment group *i*.
Both the average incoming flows (f^i) and the average size of the standing queue (q^i) of road segment group *i* are obtained by periodically performing the following exponential smoothing updates:

$$f^i := 0.75f^i + 0.25f^i_{in} \qquad (1.3)$$

$$q^i := 0.9q^i + 0.1\hat{q}^i \qquad (1.4)$$

Where,
f^i_{in} is the number of vehicles flowing into road segment group *i* during the interval between the averaging updates (one simulation step).
$\hat{q}^i$ is the size of standing queue on road segment group *i* during the same interval.

v^i, f *i*, and q^i are used as a measure to represent the relative congestion of each road segment group, and thus help calculate the cycle length and green times. The critical flow ratios are then calculated for every road segment group *i* as the ratio between the estimated flow and the saturation flow rate:

$$v^i = \frac{v^i}{m \times s^i} \qquad (1.5)$$

Where,
v^i is the estimated flow calculated in equation (1.2).
s^i is the saturation flow rate of one road segment.
m is the number of road segments inside of a road segment group *i*.

Finally, in order to use Webster's equation to calculate cycle length, the lost times must be defined. In this research, lost times are limited only to yellow times, and hence two yellow phases are considered for the intersection in Figure 2. To determine the time in each of those yellow phases, a well-known rule in traffic design was used. This rule determines yellow time for an approach of to intersection (in seconds) according to the speed limit, in miles per hour (mph), on that approach. According to this rule, 1 second of yellow time should be scheduled for every 10 mph of the maximum speed allowed. In this research, the speed limit on all approaches is assumed to be 40 mph, and thus each phase of yellow time is set to 4 sec. In conclusion, the lost time (*L*) in Webster's equation will be replaced by 8 seconds.

Now, Webster's equation can be used to calculate the optimal cycle length C_0 according to the estimated flows. Also green times can be calculated according to critical ratios in Equation (1.5) as follows:

$$g^i = \frac{y^i}{Y}(C_0 - L) \quad (1.6)$$

Where,
g^i is the green time that should be associated with the road segment group i. Y is the summation of all yi at the intersection. Some limitations should be introduced to the algorithm to ensure comfortable driving and accommodate a diverse pedestrian population. For example, minimum red time should be respected; this time is required by a pedestrian to cross the road segment with an average speed of (4ft /s). Also, a minimum and a maximum cycle length should be respected, where the minimum cycle length is the sum of two minimal green times with two yellow times, and the maximum cycle length is normally 1.5 C_0 for an optimal cycle length in moderate traffic conditions.

Webster's algorithm has also been extended to deal with special traffic cases. Two cases are covered in this research: a) eliminating tedious waiting times on red by switching the traffic light automatically to green when the opposite direction has no demand; and b) extending a finished green phase when the opposite direction has no demand.

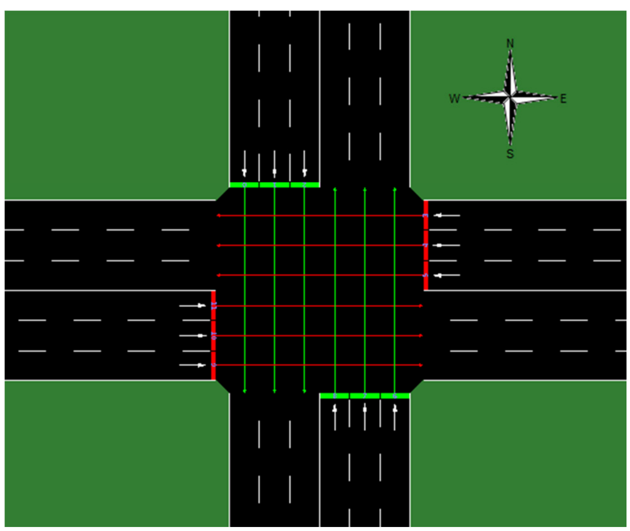

Figure 2. Proposed Intersection

The simulation scenario that will be used to validate the concept has to be as realistic as possible. To build such a scenario, the following parameters and models should be defined carefully: a) the road network; b) traffic flows that will run through the network; and c) the routing algorithm used to route vehicles between their respective origin and destination. Since this simulation scenario will be used to evaluate the performance of adaptive traffic light algorithms, only a single intersection is needed, and the routing algorithm is thus eliminated.

The road network used for this scenario is an isolated 4-way intersection, where isolated means that the effect of adjacent intersections is not considered. Each approach at the intersection has two directions, and each direction has three lanes. No left or right turns are allowed. The speed limit on all lanes is 40 mph ≈ 17.89 m/s.

Table 1. Vehicles Types in the Simulation

Vehicle Type	Max Speed [m/s]	Acceleration [m/s2]	Length [m]	Probability [%]
Typical	70	2.68	7.5	49
Fast	80	3.83	7	19
Slow	60	1.92	6.5	22
Van	60	2.44	10	10

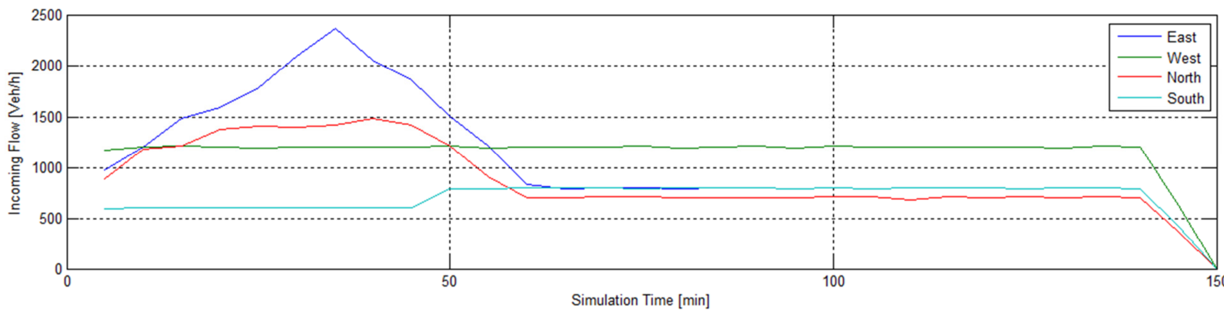

Figure 3. Incoming Flow over Simulation Time

Vehicles participating in the scenario were assumed to be of four types: Typical, Fast, Slow and Van. Table 1 specifies the different parameters and distribution for each vehicle type.

The input flow of vehicles into the intersection is defined carefully to represent situations such as peak hour demand. For this evaluation process, the input flow definition was inspired by the one used by V. Gradinescu et al. in [7]. It assumes the simulation covers almost three hours of real life, during which a peak demand occurs. The input flow on different approaches and their variations over the simulation time are illustrated in Figure 3.

A pre-timed traffic light cycle of 31 simulation seconds of green time and 4 simulation seconds of yellow time per approach is used, as described in the Simulation of Urban Mobility (SUMO) simulator.

1.4. Evaluation and Validation

The simulation scenario simulates three hours of real life during which a peak demand will occur, and then the demand will go down to a normal level. The measure of effectiveness (MOE) used to evaluate the performance of the algorithm is basically the average control delay which is defined as the difference in travel time for vehicles when driving down a road with a traffic light controller and when driving over the same road but without a traffic light controller.

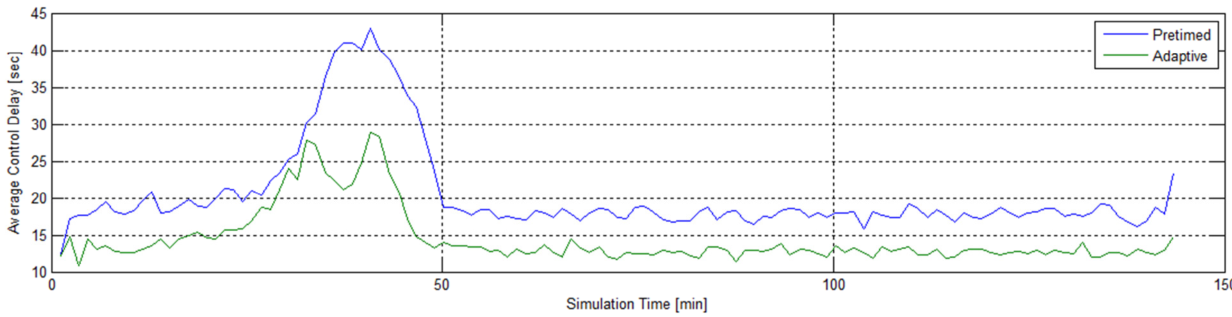

Figure 4. Average Control Delay of the Pre-timed and Adaptive Algorithms

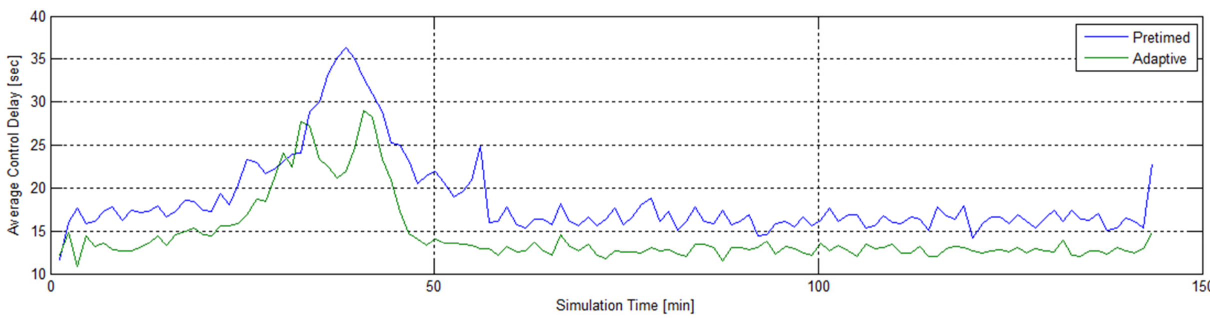

Figure 5. Average Control Delay of the "Smart" Pre-timed and Adaptive Algorithms

As illustrated in Figure 4, the simulation shows that the algorithm used outperforms the pre-timed traffic light over the entire simulation time. In addition, the algorithm improves the recovery time of the intersection after a peak demand.

Moreover, after evaluating different cycle lengths for the pre-timed program, it was observed that long cycle lengths are suitable for high demand situations, and short cycle lengths are suitable for low demand ones, thus a smart pre-timed program was evaluated. The traffic light starts with a short cycle length of 60 minutes, however after 25 minutes it switches to a longer cycle length of 80 minutes with fixed yellow times. Then, when the simulation reaches minute 55, the traffic light switches back to the old timing cycle. The results for this smart pre-timed program are shown in Figure 5. Although the "smart" pre-timed traffic light outperforms the traditional pre-timed traffic light, the adaptive algorithm still outperforms the "smart" pre-timed program and provides lower control delays over the entire simulation time.

Other MOEs have been evaluated such as queue length, waiting time, travel time, and of course emissions and fuel consumption. Two of the important aspects of ITS applications are the reduction in emissions and fuel consumption.

Figure 6 shows the results of evaluating queue length in front of the traffic light. Although the adaptive algorithm increased queue length on the northbound segment during a portion of the peak demand period, the algorithm succeeded in reducing queue length in front of the traffic light over the entire simulation time. This indicates that the adaptive algorithm has succeeded in reducing congestion at the intersection.

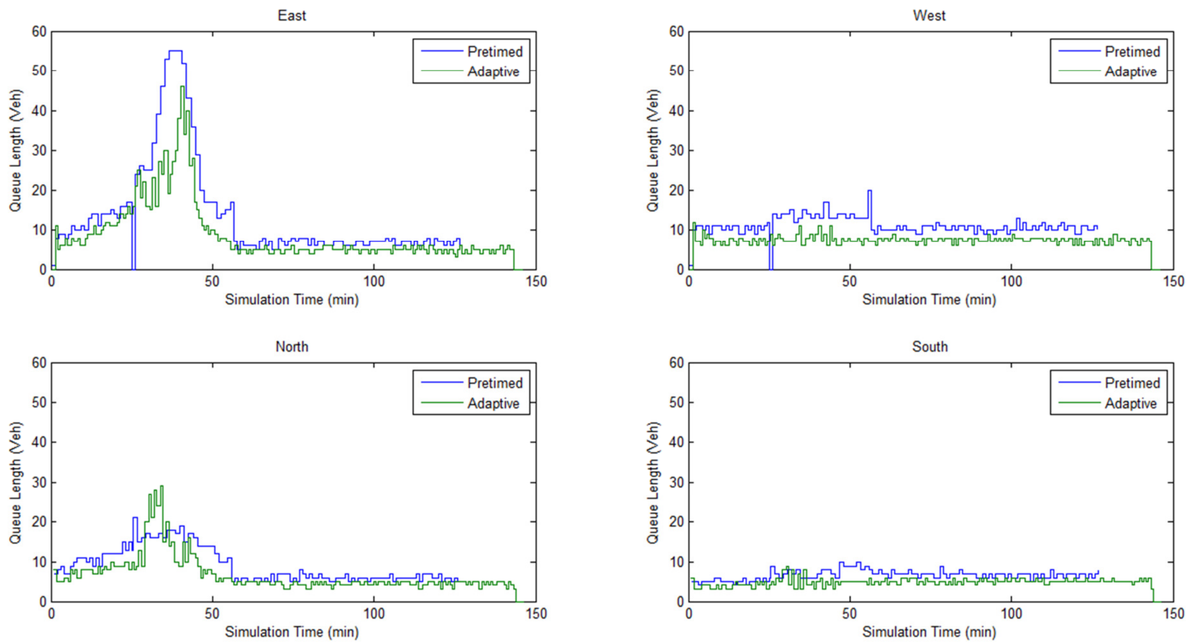

Figure 6. Queue Length Variations in front of the Intersection on all Bounds

In terms of waiting and travel times, the adaptive algorithm introduces small delays on the northbound segment while reducing them on the eastbound lanes and improving the overall performance at the intersection. Figures 7 and 8 show the simulation results of evaluating those parameters on every bound.

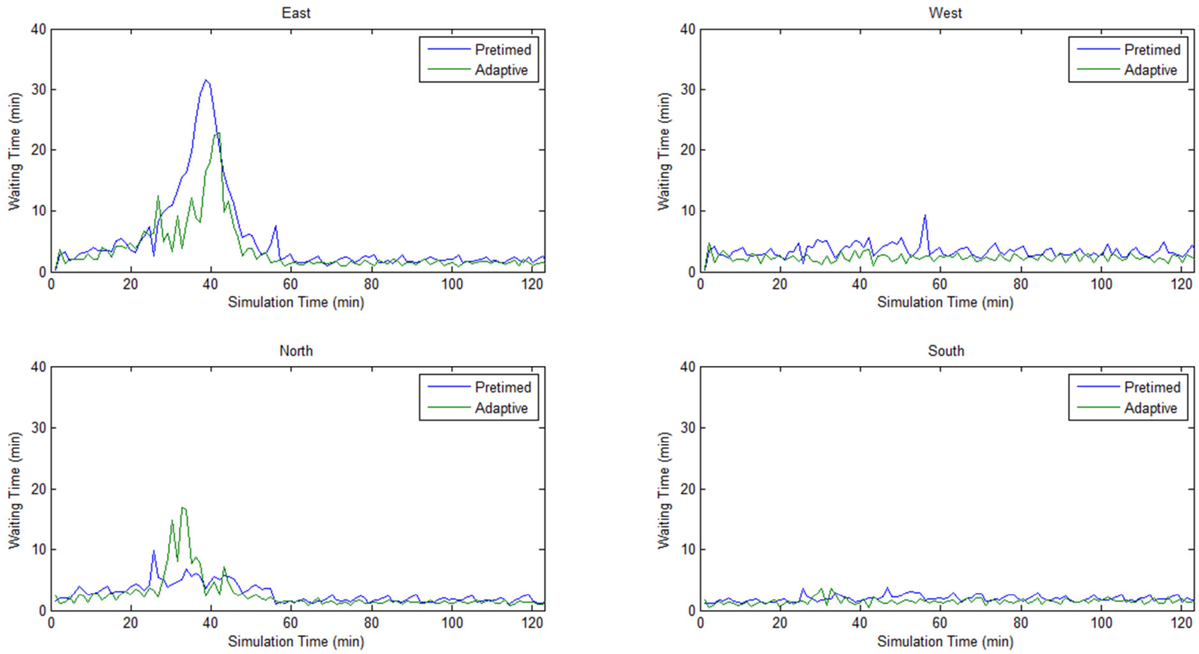

Figure 7. Waiting Time Variations over Simulation Time on all Bounds

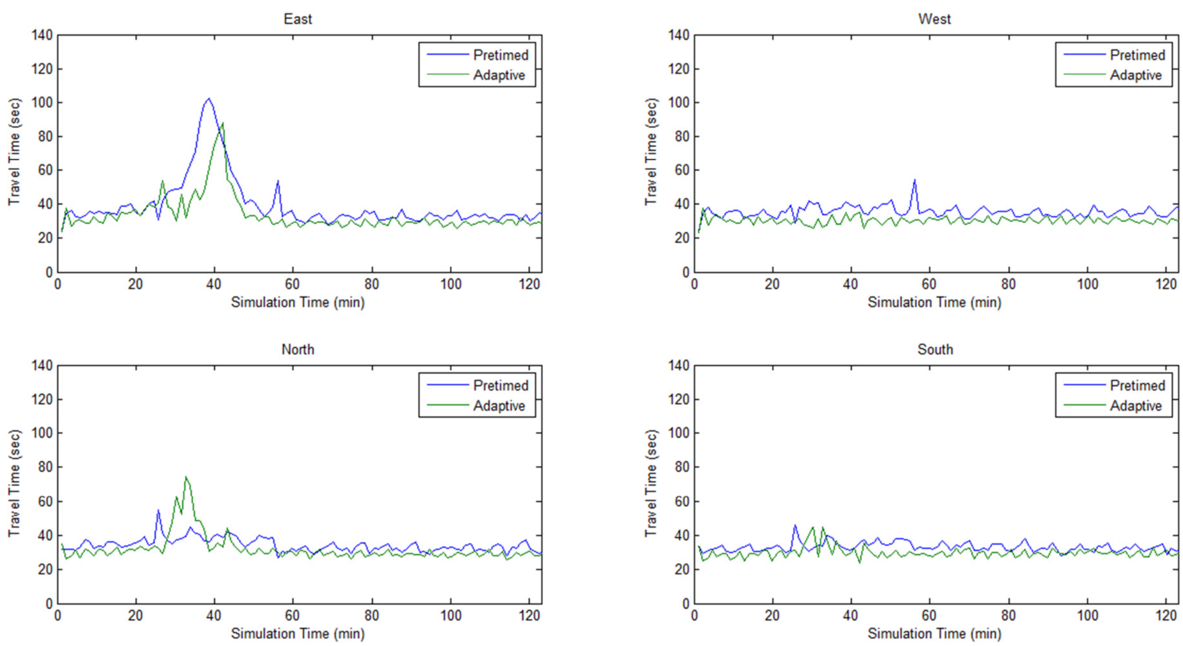

Figure 8. Travel Time Variations over Simulation Time on all Bounds

1.5. Conclusion

In this section, a methodology for adaptively controlling traffic lights in a connected vehicle environment was proposed and evaluated.

A new algorithm based on Webster's equation was developed. This algorithm used message exchanges between vehicles and the traffic light (V2I) to estimate the demand level on every road segment and thus calculate suitable cycle length and green time splits. Many parameters were considered in the algorithm such as minimum and maximum cycle length, and minimum green time for one direction. Furthermore, the algorithm was able to react to special events at the intersection such as switching the traffic light to green for a direction waiting on red when there is no demand in the opposite direction, and extending the green phase beyond the calculated value for one direction as long as the opposite direction has no demand while maintaining a maximum limit on acceptable waiting times for pedestrians.

Chapter 2: Evaluate Different Roadway Routing Algorithms (UDM Research Team)

2.1. Introduction

Transportation activities are the fastest growing source of U.S. greenhouse gas emissions, accounting for 33.1% of the total energy-related CO_2 emissions in 2008 0. Given the significant impact of motor vehicle emissions on the environment and the demand to reduce the dependency on oil, the Intelligent Transportation Systems (ITS) technologies have recently focused on enabling innovative eco-friendly solutions such as “green driving”. One potential solution includes a routing advisory system that would guide drivers through a minimized fuel consumption travel route: “Eco-Routing.” Recently, some car navigation systems have integrated the Eco-Routing service into their systems, utilizing different estimated emission models.

However, these existing models are calculated on the basis of either predetermined or pre-measured road link attributes, such as traffic density and average speed. This approach does not take into consideration the dynamic traffic conditions. Real–time communication between vehicles (V2V) and between vehicle and infrastructure (V2I) is an essential functional requirement to achieve improved energy and emission estimates. The stochastic nature of traffic conditions requires us to employ a fast speed communication technology. The newly developed Dedicated Short-Range Communications (DSRC) provides vehicles with the capability of exchanging real-time traffic information, a necessity for proper estimation of traffic congestion.

This report is focused on the core concept of *evaluating eco-routing algorithms*. Section 2.2 presents an overview of the evolution of navigation systems. Sections 2.3 and 2.4 classify navigation systems and discuss their different route optimization parameters. Section 2.5 presents the recent developments in ITS-based eco-routing systems and defines the scope for this research. Section 2.6 introduces a new system for adopting ITS technology in eco-routing systems. Section 2.7 describes the iTETRIS simulation environment as a platform to evaluate the performance of the new navigation methodology. Section 2.8 presents a case study of a real world road network, and compares the performance of the new routing methodology against the conventional routing methods over this network via simulation. Finally, section 2.9 concludes the chapter and discusses its findings.

2.2. Background

Finding the optimal route in a road network from a current start location to a given destination location is an everyday problem that most drivers have to tackle when planning a trip for a new target point. Many applications were designed to provide route search service using different platforms, ranging from separate car navigation devices, to cell phones, to web-based navigational maps. The term “optimal”, in a routing algorithm, may refer to a range of objectives that end-users can choose from to optimize the route, such as fastest route, shortest route, fastest route given a preference to various road characteristics, or most fuel-efficient route. These navigation algorithms also differ in the way they deal with the changing traffic conditions over time, and they can be divided into three main categories: *static, deterministic time-dependent and dynamic* planning algorithms. In the static planning model, all travel times and traffic conditions are considered constant over time, resulting in less realistic driving costs for road segments. The static model was improved on in the extended deterministic model, in which certain properties of road networks are considered to change as a function of the time of day, week or even season (e.g. some roads may be closed during specific time periods) thus the accuracy of the route cost estimation is increased. The extended deterministic planning model is currently used by the most of the commercial navigation systems. However, more accurate representation of the traffic flow, and hence responsive routing, can be achieved with a dynamic routing model. In dynamic routing, real-time traffic information is integrated into the planning model. The optimal route to the destination is calculated before the start of a trip and it is updated dynamically to adapt to new traffic conditions during the travel time. Theoretically, the resulting optimal route of the dynamic routing approach outperforms the static and time-dependent routing approaches in terms of accuracy and efficiency. However, in order to compare different route-planning methodologies, a clear classification of those methodologies is needed.

2.3. Route Planning Classifications

In this section we present three road network models. First, we discuss the model that we use to plan routes in a road network if all driving times are constant over time. This is the basic model. We then extend this model to incorporate the influences of real-life on the quality of a planned route. This model is used to take daily traffic condition patterns, such as traffic jams, into account when planning a route. Finally, we extend the time-dependent model so that uncertainty in travel times and costs are also incorporated.

2.3.1. Time-Independent Planning

This is the basic model that is used to plan routes in a road network if all driving times and road conditions are constant over time. Costs in this model do not depend on the time of day.

Under time-independent routing, a road network is modeled in graph theory with a road-graph tuple $G= (N, E, w_e, w_r)$. The route cost of a route is represented by w_p: $P(G) \rightarrow R_0^+ \cup \{\infty\}$ υ

$$w_p(p) = \sum\nolimits_{e \in E(p)} w_e\,(e) + \sum_{i=1}^{l-1} w_r\,(e_i, e_{i+1}) \qquad (2.1)$$

Where $P = (e_1,\dots,e_m)$, a route from start node to destination node, consists of a sequence of edges with $e_i \times E$. $w_e : E \rightarrow R_0^+$ is the non-negative cost of edge e (length of the road segment/Travel Time over the road segment). $w_r : P2(G) \rightarrow R_0^+ \cup \{\infty\}$ υ is the cost function to model the traffic rules by on pairs of adjacent edges. If $w_r\,(e_1, e_2) > 0$, then there exists a rule from edge e_1 to e_2 with rule cost $w_r\,(e_1, e_2)$.

2.3.2. Deterministic Time-Dependent Planning

This is an extended version of the previous model. It takes into account certain properties of a road network that change over time and assumes that all edge weights are stochastic. For example, a cost function can take into account different weightings of travel time, distance, scenic value, and fuel consumption.

The road graph in this model is a tuple $Gt = (N, E, w_e^t, w_r^t, t_e^t, t_r^t)$.

- $t_e^t : E \times R_0^+ \rightarrow R_0^+ \cup \{\infty\}$. The driving time function that gives for every edge the time it takes to traverse that edge.
- $t_r^t : P2(G) \times R_0^+ \rightarrow R_0^+ \cup \{\infty\}$ The turning time function that gives for every rule the time it takes to make the corresponding turn.
- Time-route is given by a tuple $\pi = (p,t)$. Given a departure time t_1, let $\pi = (p, t_1)$ be a time-route, The total route costs of time-route π are given by:

$$w_p(\pi) = w_e^t(e_1, t_1) + \sum_{i=2}^{l} w_e^t(e_i, t_i + t_r^t(e_{i-1}, e_i, t_i)) + \sum_{i=2}^{l} w_r^t(e_{i-1}, e_i, t_i), \qquad (2.2)$$

With $t_2 = t_1 + t_e^t(e_1, t_1)$ and $t_{i+1} = t_i + t_r^t(e_{i-1}, e_i, t_i) + t_e^t(e_i, t_i + t_r^t(e_{i-1}, e_i, t_i))$ for i = 1, 2, ...

2.3.3. Dynamic Time-Dependent Planning

Since initial travel speeds over different road segments are not equal to those observed by the driver during his/her trip, the static and time deterministic models introduce uncertainty to deal with unexpected events, such as traffic jams, accidents, road conditions…etc. This adaptation does not provide accurate estimates of the travel time over different segments of the road along the trip route. To address this problem, dynamic time-dependent planning models combine the route planning system with GPS receivers, digital maps, and ITS-based wireless technologies to aggregate real-time traffic information and generate a real time updated edge weight/cost throughout the planned trip. Wu, Chien and Sung presented a dynamic navigation algorithm 0 with the assumption that vehicles are equipped with a wireless network interface to exchange real-time traffic information with neighboring vehicles (V2V). The navigating vehicle broadcasts a road query message whenever a predefined incremental distance (e.g. 2 Km) is traveled. The query message inquires other driving vehicles, within a specific range, about their individual travel speed, and upon receiving sufficient responses the weights of the surrounding road segments will be updated. This methodology improves the responsiveness of the routing algorithm and is expected to provide better routes for the navigating vehicle, however it has to deal with a fundamental communication problem, and that is network overloading. The network loading has two sources: the interest region of the inquiry message and the inquiry decimation algorithm. The interest region problem is addressed by tuning the query decimation distance according to the average vehicle speeds along the navigating vehicle road. While the query decimation is treated by using a rebroadcast block - based algorithm, in which the number of rebroadcasting nodes is limited to those located in a rebroadcast block of dynamic size set to adapt to the traffic density, i.e. a multiple of the average inter-vehicle distance.

2.4. Routing Algorithm Criteria

This chapter focuses on the route planning functionality of an automobile navigation system. Navigation systems usually provide the option of optimizing the vehicle route according to different criteria. In general, the driver can choose between planning the fastest route, the shortest route, or the fastest route while giving preference to highways.

These various objectives mainly differ in their calculated cost function:

a) *Fastest Path Routing Algorithms:* The cost function/edge weight = Travel time.

b) *Shortest Path Routing Algorithms:* The cost function/ edge weight = Length of route.

c) *Multi-Objective Routing Algorithms:*

The cost function/edge weight = Travel time + Amenities of driving/Ease to drive.

d) *Green Eco Routing Algorithms:*

The cost function/ edge weight is a function of the following road factors: slopes, start and stop sequences/traffic lights and stop signs, cross roads, curves, railroad crossings and speed changes. These factors are known to impact fuel efficiency, and for each segment of road, the eco navigation system defines a fuel consumption index based on both these road attributes as well as engine characteristics. This index used in combination with distance and estimated travel time can provide a calculation of the path of minimum fuel consumption -- the "green" route. Fuel is wasted by actions such as accelerating just before a curve, a roundabout or a limited speed zone such as a city center or school neighborhood, or by climbing a slope in the wrong gear. Digital maps can be used to anticipate of these conditions and minimize the speed and gear changes on a trip.

2.5. Related Work

There have been several recent studies aimed at presenting vehicle navigation systems that reduce fuel consumption. One methodology is eco-driving; this approach is very effective in reducing fuel consumption. However, eco-driving systems mainly involve advising drivers of fuel-efficient driving techniques. The eco-driving system is rendered impractical if not combined with a fuel-efficient route selection. A navigation system that searches for a fuel-consumption optimized route based on predicted fuel consumption calculations, guiding drivers through fuel efficient routes is key to providing true eco driving. Kono and Fushiki propose an ecological route search, which generates routes of optimized fuel consumption using several factors such as traffic information, geographic information, and vehicle parameters 0. They report the fuel consumption prediction model and the results of comparative driving experiments using their ecological route search and conventional time priority route search methods. Raghu, Ganti, and Nam Pham developed a navigation service called "GreenGPS" with a prediction model for fuel consumption that reads the engine parameters through an On-Board Diagnostic (OBD-II) interface and utilizes participatory sensing data (voluntary data collection shared through a web-based community) to map fuel consumption on city streets 0. The experimental evaluation of this green navigation system showed an average fuel consumption reduction of 6% over the shortest route and 13% over the fastest based on a prediction accuracy of 1%. The participatory sensing framework developed in this service runs in a client/server environment. PoolView, a central web-based server works as a data collection application used to aggregate the client-side data uploaded through a scanner device, DashDyno, interfaced to both inter-vehicle ODB-II and a Garmin GPS. These raw data are then used to build a new predictive consumption model or to refine the accuracy of an existing one. However, these approaches using participatory sensing applications usually suffer, specifically over the short-term of service, from the lack of the widespread participants, i.e. the member users who own the ODB-II scanner with the service installed on it. Hence, the contributed data would be sparse and not adequate to build an accurate model.

As a solution to this challenge, namely, the generalization from sparse multidimensional data sets, GreenGPS proposed a clustering methodology by deploying a three-dimensional data cube framework: make, year, and class of a particular vehicle. As a result, the aggregated data can be grouped into eight clusters; each cluster represents one or a combination of the previous dimensions, such that a separate predictive consumption model can be built for each group.

The proof-of-concept experiment of this study involved driving sixteen sedan cars of different make, year and class over the course of three months and a total of approximately 1000 miles. Because of this limited data set, the experimental results showed that the (make,year) cluster is the lowest cumulative error cluster scheme, followed be either the (make) cluster or the (year) cluster. Finally, if a navigated car matches none of the previous clusters, then the most common computed model will be used to predict the fuel consumption of this car resulting in a low accuracy outcome. The model structure used by this study to estimate fuel consumption relies on simple calculations involving only the main factors that affect the estimation results such as the physical and the geographical road characteristics and the vehicle attributes. However, this calculated model has two main drawbacks:

a) First, it doesn't consider the factors related to driver behavior since the experimental samples were held for only sixteen different drivers.

b) Second, it doesn't take into account real-time traffic information such as the congestion conditions. Instead, it uses the predetermined-time traffic information which is computed based on the Time-Of-Day historical data.

Jian-Da Wu and Jun-Ching Liu proposed a fuel consumption predictive model utilizing the adaptive learning capability of a back-propagation neural network to refine accuracy and hence minimize the forecasting errors 0. The proposed predictive process goes through two main phases: data acquisition and training of the predictive model's neural network. In the first phase, the system acquires data and information regarding the key factors that impact fuel consumption in automobiles. The chosen factors include: make, weight and type of car, engine style and gear transmission type. These factors are fed into the following phase of creating and training the predictive model. The proposed model accounts only for the resistance that the car experiences while it's driving, the more tractive force is exerted the more fuel is consumed. For instance, the weight of the car is proportional to the rolling and gradient resistances, while the aerodynamic resistance is dependent on the frontal area (car's type) and vehicle's velocity (gear transmission). The following equation is used to specify the previous relationships:

$$F_T = D + R + m\frac{dV}{dt} + mg\sin\alpha \qquad (2.3)$$

Where F_T is the tractive force, m is the vehicle's mass, V is the road speed, g is the acceleration of gravity, and α is the inclination angle of the road.
The $m\frac{dV}{dt}$ is called acceleration resistance and $mg\,sin\,\alpha$ is climbing resistance. D is the aerodynamic drag which is given by the equation: $D = \frac{1}{2}\rho C_d A V^2$, where ρ is the air density, C_d is the coefficient of aerodynamic drag and A is the vehicle's frontal area. R is the tire rolling resistance which is expressed by this equation: $R = f_r.G$, where f_r is the coefficient of rolling resistance, $G = mg$ is the force that the vehicle exerts on the ground. The second phase goes through two sub phases: the input parameters initialization required to operate the predictive system's neural network, and the learning algorithm of that network. The impact factors and the information of fuel consumption are obtained from the FTP-75 (USA) consumption test method for 14 chosen automobile brands. A total of 124 data points (seven domestic car manufacturers) and 217 data points (seven domestic and seven imported car manufacturers) are stored in the prospective database. For the proposed neural network basic parameters (BP), the number of BP network neurons chosen is 250. The hidden layer's learning rate and training epochs are 0.55 and 1000, respectively. Finally, the percentage of accuracy (PA), which represents the work efficiency for the prediction performance of vehicles' fuel consumption system, is defined as: $PA(\%) = 1 - \left|\frac{T-A}{T}\right| \times 100\%$ Where T means the target fuel consumption in the auto energy website and A represents the fuel consumption from the proposed neural network system. The BP learning algorithm includes iterative modification of the synaptic weights and bias. It can formally map between the input parameters and the desired output variables. The familiar weight and bias initialization function are generated randomly between -1 and 1. The training stage was implemented for three different database quantities, one, a half, and one-third of the total database size respectively. Finally, the whole database is tested to evaluate the performance of the fuel consumption prediction. The testing results shows a high quality prediction performance of approximately 98% for 124 training samples and around 94% for 42 training samples out of 124.

2.6. Design and Implementation

In section 2.5 we presented the various environmentally friendly routing algorithms. The main goal of our research is to further enhance green routing and develop a novel algorithm that optimizes fuel consumption by detecting traffic congestion in real-time and applying dynamic routing. Traffic congestion is identified from the speed of vehicles, and it is associated with a congestion index that reflects the degree of congestion along a road segment. Based on the calculated traffic density degree, a decision on a new eco-route may be taken in order to avoid congestion and thus reduce the waiting time, which could result from adhering to a previously selected static route.

Our proposed detection and sensing methodology depends on the exchange of real-time traffic information with neighboring vehicles. The navigation device will collect the travel time information as it moves through a road network, this information is then filtered based on its relevance. The novel routing algorithm then implements an innovative eco-routing algorithm to determine the least cost route to the destination dynamically as traffic conditions change. Next we describe the most important components of the solution.

a) Travel Time Measurement

The first step in determining the most fuel efficient route is estimating the real travel time for each upcoming edge along the whole trip; traffic congestion is calculated based on the received traffic travel time messages from neighboring vehicles. This process is iterated at a predetermined time interval whether or not a new eco-route has been decided.

To enhance the accuracy of the proposed predictive travel time model, not only are the current driving vehicles on the next upcoming edge considered, but so are vehicles that are heading towards this edge and will affect the traffic conditions for the navigated vehicle in the future. The estimated travel time is calculated based on two main weighted components: the vehicle's instantaneous and predicted speed profiles.

The flow chart in Figure 9 shows the main processes that are involved in this start-up phase.

The distance vector routing algorithm has a key advantage over the link state routing algorithm in our application, because its routers only communicate with their directly connected neighbors. This will allow the physical implementation of a solution where nodes communicate directly using an ad-hoc vehicle-to-vehicle network, without the support of fixed infrastructure or using a CALM network, since communication will take place at most over the length of a single section of road. Also the amount of data traffic is much lower with distance vector routing because of fewer control messages being sent and because a router does not have to communicate with all other routers in the network, but only with its neighbors.

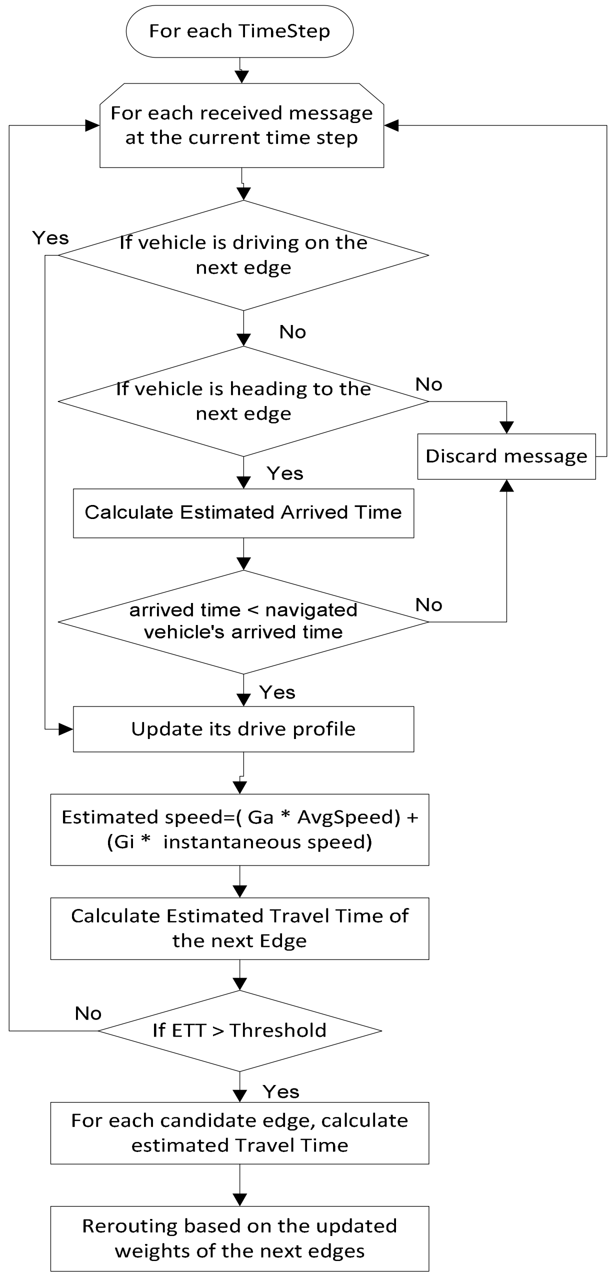

Figure 9. Predictive Travel Time Model Flow Chart

b) Routing Algorithm

Routing is the core functionality of the proposed dynamic eco-routing solution. The navigated vehicle starts its trip by following a static eco-route calculated on predetermined historical traffic information. The proposed routing algorithm will exploit all available road segments connecting to the destination and evaluate and compare the associated traffic congestion indexes; here we propose an iterative approach for calculating a traffic congestion index assessing roadway traffic conditions.

In the algorithm illustrated in the pseudo code below we extend Dijkstra's algorithm to calculate and select the least cost path, thus achieving the dynamic programming solution to optimize vehicle routing. The proposed methodology is evaluated using an open source simulation tool described in details in section 2.5.

A. Preprocessing the network map:

  1. Input: OSM OpenStreetMap xml file, Geographical Information database.

B. Implementation:

  1. Upload the following Geographical Information**:**

    1.1 Number of stop signs.

    2.1 Number of traffic lights.

    3.1 Road Gradient.

C. Integrate it into the OSM Network map:

  1. Output: Geo-Characterized OSM xml file.

  2. Calculate an initial Eco-Route for the navigated vehicle.

  3. Input:

    a. Source address -> (altitude, longitude).

    b. Destination address -> (altitude, longitude).

    c. Vehicle parameters (Weight, Frontal Area, Make, Model, Year).

    d. Depart time.

    e. Geo-Characterized OSM xml file G(V, E).

D. Implementation:

  1. Call and Execute Dijkstra's algorithm:

If ({Make, Model, Year} exists)

Extract the corresponding coefficients form look up table

Else

Set the coefficients to default values

For each edge e ϵ E Determine its weight in Gallon/Mile:

$$\mathbf{gpm = k_1 m v^2 \frac{(ST + vTL)}{\Delta d} + k_2 m \frac{v^2}{\Delta d} + k_3 m \cos(\theta) + k_4 A v^2 + k_5 m \sin(\theta)}$$

Where v is the maximum allowed speed of edge e.

ST: number of stop signs.

TL: number of traffic lights.

Δd: Length of edge e.

A: Frontal Area of the navigated vehicle.

m: Mass of the navigated vehicle.

k1, k2, k3, k4, k5: variable coefficients.

2. Determine the optimal fuel efficient cost road for travelling.

Output: Set of edges that forms the optimal Eco-Route.

3. Request the real-time traffic information

4. Set Ds = 2 km; // periodic search distance for updating the vehicle's local traffic table.

5. Do {

Choosing the candidate road edges CE={e1, e2,..en};

Receiving broadcasted messages from neighbor vehicles;

Msg_ID	Timestamp	V_ID	Speed	Positon(x,y)	Current_Edge	Next_Edge
1	0.02	4	3 m/s	(200, 467)	e1	e3

6. **If** (receiving message for edge i ϵ CE && Msg_ID != received Msg_ID) {

Update the Local Traffic Table;

// Calculate the density of the next planned edge enext at the current time.

7. ρcurrent = Calculate_Density(enext, Local_Traffic_Table, Current_time);

Step#1 // Calculate the density of the next planned edge at the expected arrival time.

8. ρexpected = Calculate_Density(enext, Local_Traffic_Table, Expected_Arrival_time);

Step#2 // Determine Edge weight based on its traffic density

9. ρAverage = (ρexpected + ρcurrent)/2 ;

Step#3 // Compute the Speed and Distance threshold for the next edge enext

1. SpeedTH = $\frac{\mathbf{Speed_{max} + Speed_{min}}}{\mathbf{2}}$;

DistanceTH is determined from the map based on the road class

V < SpeedTH	True	True	False	False
D < DistanceTH	True	False	True	False
Priority	1	2	3	4
Weight	1	2	4	8

2. **If** (Weight(enext) > 2) { For (e=1 ; e<N ; e++){ **Repeat** Step#1, Step#2, and Step#3 for each candidate edge; }

Select the less weight || priority edge; Recalculate the ECO-Route with the new edge as the start point;}

3. **While** (the navigated vehicle is Dr Km far away from a destination point)

Function Calculate_Density(edge e, Local_Traffic_Table, time t)

{

// Sorting vehicles according to its straight distance from the navigated vehicle.

4. **For** (v =1, v <=N; v++) {

// Compute the straight distance between the navigated vehicle and the next vehicle

5. $\mathbf{D_{s,v}^{staright}} = \sqrt{(\mathbf{x_s} - \mathbf{x_n})^2 + (\mathbf{y_s} - \mathbf{y_n})^2}$ }

6. **Sort** (D1, D2….,Dn);

// Compute the average vehicle distance of edge e

7. $\mathbf{D_e} = \frac{\sum_{i=1}^{N-1} \mathbf{D_{i,i+1}}}{\mathbf{N-1}}$

// Compute the Average Vehicle Speed

8. $\mathbf{V_e} = \frac{\sum_{i=1}^{N} \mathbf{V_i}}{\mathbf{N}}$

9. *Return the density ρr = (De, Ve)*}

2.7. Simulation Tool (ITETRIS)

In order to evaluate the presented eco-routing algorithm, two simulators are needed: a traffic simulator and a communication network simulator. The traffic simulator should be microscopic, i.e. able to provide information about the individual vehicles in the simulation, such as location, speed, acceleration, route and travel time. Also, it should be able to import real-world maps, such as TIGER or OSM, in order to simulate realistic traffic scenarios and gain information about the impact of particular algorithms on real world traffic. Moreover, the traffic simulator should incorporate a fuel consumption and pollutants emissions model, which provides the opportunity to simulate the Eco-friendly routing algorithm presented in section 2.6. With respect to the network simulator, it should support wireless ad-hoc networks with dynamic topologies. Although different network simulators can provide those capabilities, it is also important that the network simulator supports new protocols that are used in the vehicular networks for exchanging different traffic information. Those protocols are the Dedicated Short Range Communications (DSRC) in the US, and Wireless Access in Vehicular environments (WAVE) in Europe. After researching and evaluating several computer-based simulators, iTETRIS was identified as the most flexible, capable, and fully integrated simulation environment; hence it was selected to evaluate the new echo-routing algorithm. iTETRIS is based on two basic simulators, a traffic simulator, SUMO, and a network simulator, ns-3. However, iTETRIS is not limited to the features offered by ns-3 and SUMO; rather, it extends its functionality through a central block, iCS, that provides the ability to simulate traffic management strategies that utilize the novel cooperative V2X communication systems. Figure 10 illustrates iTETRIS software architecture.

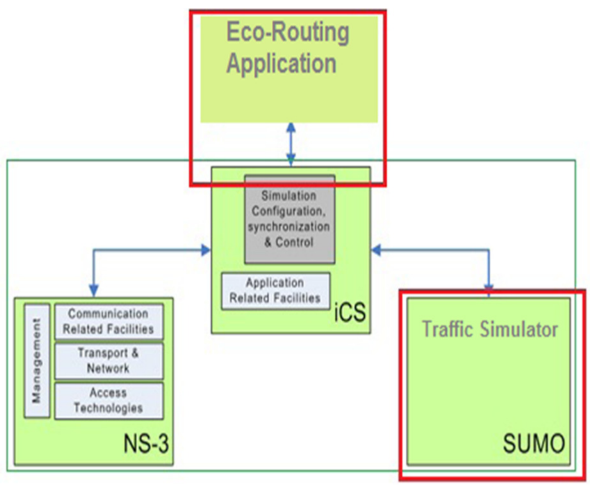

Figure 10. iTETRIS Architecture

2.8. Case Study

In order to test the new routing algorithm, a real map of the road network at Eichstätt in Bavaria, Germany, was imported in SUMO as shown in Figure 11. Traffic in the city was simulated over one hour of simulation time, during which a set of trips (source/destination edges) was generated with a uniform random distribution to cover the simulated area. Those trips were assigned to random vehicles, and the vehicles were then dispatched evenly throughout the simulation at an emission rate of 4 vehicles per 25 seconds.

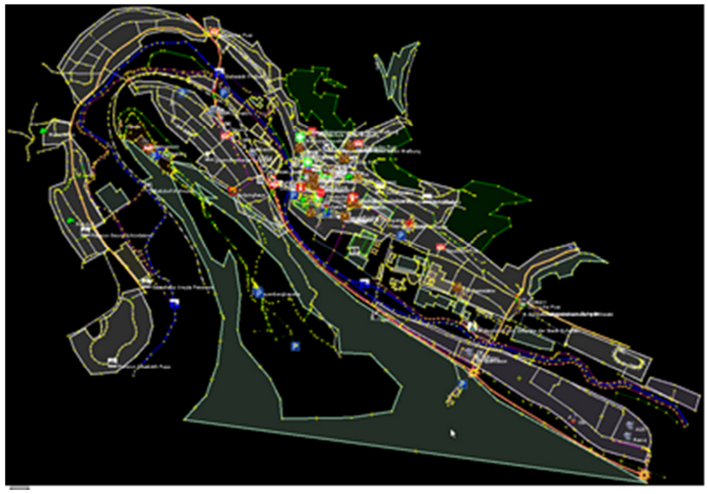

Figure 11. Road Network for Case Study

Two routing algorithms were evaluated and compared: the first optimized vehicles' routes for the shortest travel time, and the second optimized the routes for the most efficient fuel consumption. Each vehicle in the simulation was assigned a recording device that captures and stores the vehicle's instantaneous trip information such as CO_2 emissions and fuel consumption and other static information such as the route length and travel time. The trip information of all vehicles was then processed, and the following global performance metrics were calculated:

- Total route length (meters): the total distance traveled by all vehicles to get to their destinations.
- Total waiting time (seconds): the total time spent waiting at traffic lights by all vehicles.
- Total travel time (seconds): the total travel time spent by all vehicles to get to their destinations.
- Total fuel consumption (liter/second): The total average fuel consumption rate by all vehicles.
- Total CO_2 emissions (gram/second): The total CO_2 emission rate by all vehicles.

Figures 12 – 15 illustrate the ecological advantages of dynamic fuel consumption-optimized routing with respect to the traditional static travel-time optimized routes. In addition to the environmental advantages, even the driving experience, presented by the travel distance and time, is improved. The travel time and distance improvements result from the dynamic traffic status updates provided by the V2V and V2I communication. The ecological improvements, on the other hand, result from the dynamic traffic updates as well as the modified road costs that reflected the fuel consumption and CO_2 emissions associated with these roads.

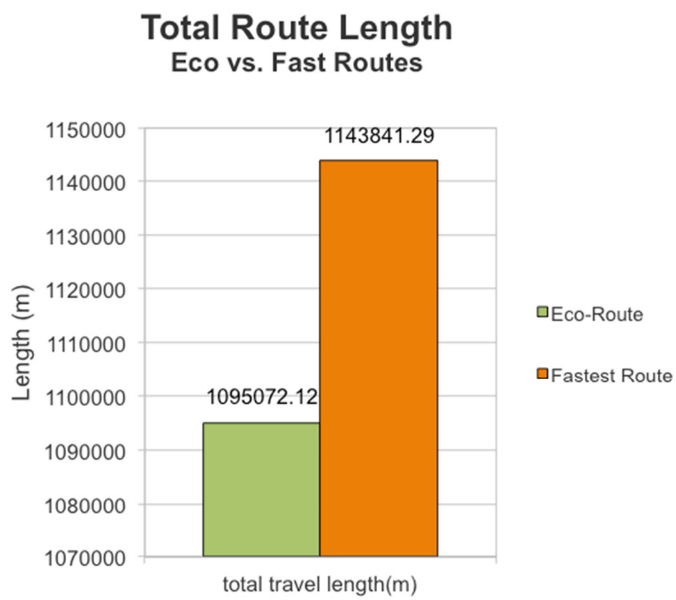

Figure 12. Total Travel Length of Vehicles (Meter)

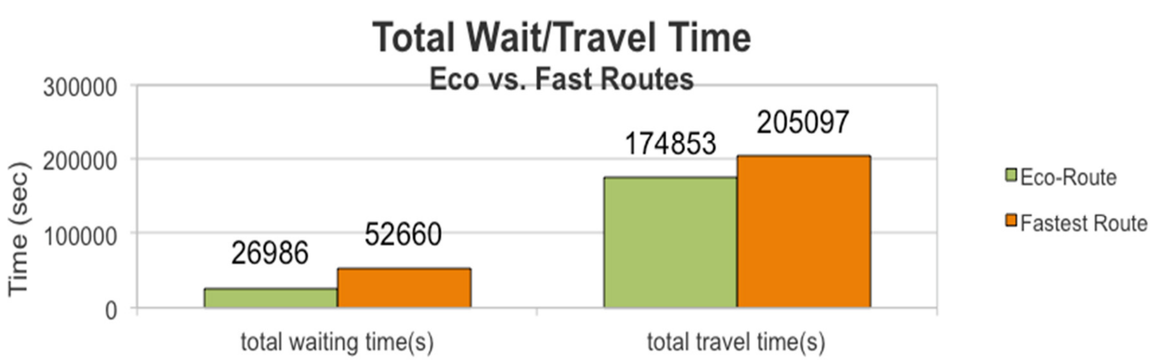

Figure 13. Total Travel/ Waiting Time (Second)

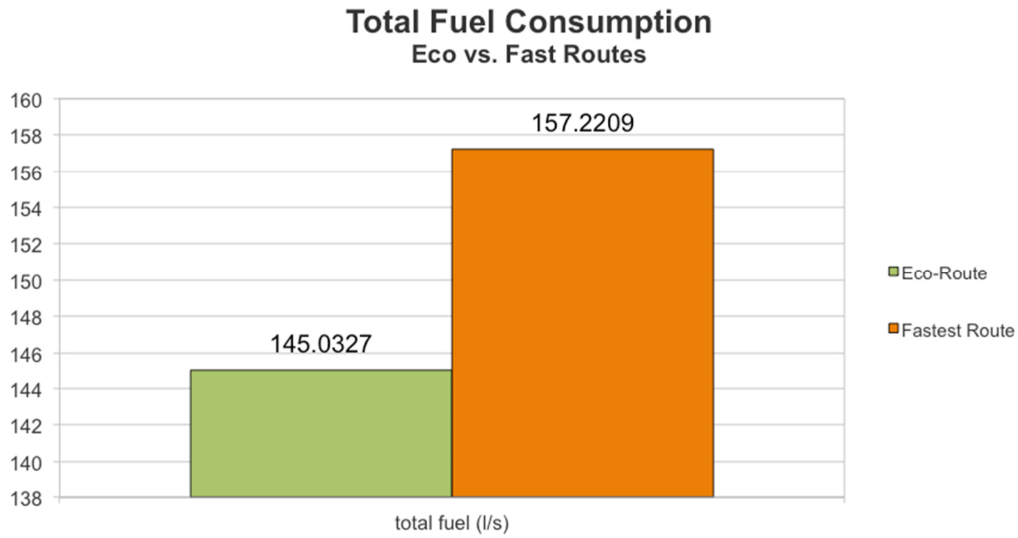

Figure 14. Total Consumed Fuel (Liter/Second)

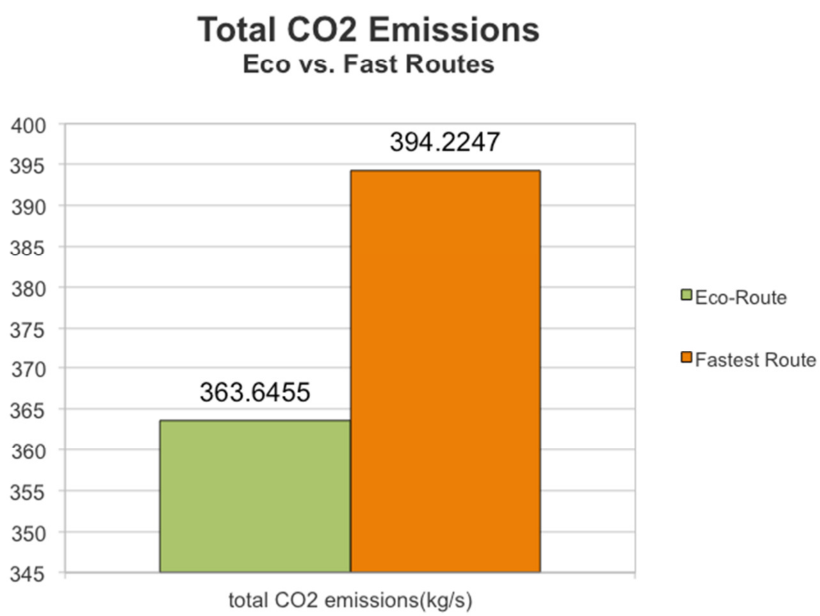

Figure 15. Total Generated CO2 Emissions (Grams/Second)

2.9. Conclusion

Optimal route search is one of the primary functions of car navigation devices. Several researchers have studied the methods of finding the optimal route considering multiple criteria to find the route that the users want. Most of the optimal route search algorithms focus on static route computations and optimize routes for either the shortest distance or the fastest travel time. With the development of intelligent transportation systems (ITS), car navigation devices are capable of receiving real-time information for traffic conditions. In this chapter navigation systems are classified according to the traffic information collection methodology and the calculated road costs for the route optimization parameters. In addition, a travel time-based eco-friendly routing algorithm was presented and evaluated by simulation. The simulation results showed that the eco-routing calculation allows for a decently improved performance compared to the conventional static fastest travel time routing approach.

Chapter 3: Development of an Efficient Media Access Protocol for Smart Intersections (WSU Research Team)

3.1. Introduction

Recent statistics show that in the United States, motor vehicle accidents cause 9117 fatalities at intersections [17]. The development and deployment of Intelligent Transportation Systems (ITS) technologies will reduce the risk of accidents and fatalities, improve safety, and solve traffic problems [18][19]. The rapid evolution of wireless technologies, especially, the new Dedicated Short Range Communications (DSRC) at 5.9 GHz will support ITS systems and provide wireless data communications between vehicles, and between vehicles and infrastructure. According to one report [20], a comprehensive list of vehicle safety applications enabled by DSRC was compiled. More than 75 application scenarios were identified and analyzed. One of the identified safety applications is intersection collision warning. The report suggested the use of infrastructure sensors to determine the locations of vehicles and then transmit this information to other vehicles approaching the intersection using DSRC wireless technology. Researchers have proposed the use of TDMA and IEEE 802.11 CSMA MAC protocol for intersection collision warning systems [21]. However, the medium access mechanism in 802.11b/a MAC protocol is based on a random back-off timer. Therefore, a mechanism must be developed to provide fairness [22] among vehicles that contend for the channel. According to the DSRC specifications [23], a system of priority access may be essential as a channel access strategy to support the dynamic environment of vehicle safety applications.

In this paper, we propose an Intersection Warning Channel Access Priority (IWCAP) protocol. The IWCAP protocol can be used either as a new MAC protocol to support this application or as an add-on component to the IEEE 802.11a MAC protocol. The application may have an option to choose between the back-off timer and IWCAP. The paper is organized as follows: in Section II, we describe the intersection warning system architecture and the required technologies to build this system; in Section III, we describe in detail our proposed protocol; then, in Section IV, we analyze our protocol and indicate how the IWCAP can be used as an add-on component to the IEEE 802.11a. Finally, we conclude this paper in Section V with future work.

3.2. System Architecture

3.2.1. Dedicated Short-Range Communication (DSRC)

In North America, 5.9 GHz Dedicated Short-Range Communications (DSRC) systems are being developed to support a wide range of roadside-to-vehicle and vehicle-to-vehicle safety applications. There are seven non-overlapping 10MHz channels in the 5.850-5.925 GHz band. DSRC also supports data rates of 6, 9, 12, 18, 24, 36, 48, and 54 Mbps. In this paper, we assume that one of the channels is used for an intersection-collision warning system. In addition, the MAC protocol that supports DSRC systems is the IEEE 802.11a. In this paper, we use the IEEE 802.11b/a term slot time to represent the time a vehicle has to sense or burst the channel.

3.2.2. Measuring Distance to an Intersection

Our proposed protocol relies on the vehicle's speed and on measuring the distance between an intersection and vehicles approaching the intersection. There are several techniques that were proposed to measure the distance to an intersection [24] [25]. For example, a GPS device can be installed at the intersection that continuously transmits its coordinates to incoming vehicles. Vehicles, which will also be provided with GPS devices, receive the coordinates from the device and calculate the distance to the intersection. Another approach is by using magnetic strips or markers on the road that are placed at a predefined distance from an intersection. These magnetic strips may carry pieces of information including the distance to intersection, lane number, and direction. Another similar approach is to use RF transmitters that are embedded on the road. Vehicles approaching an intersection detect a signal from an RF transmitter and then receive a message that contains information about the distance to the intersection. In this paper, we assume that one of these techniques provide vehicles with information about the distance to an intersection, the lane number and the intersection leg that a vehicle is traveling. We will also assume that these techniques trigger the vehicles to start our proposed protocol at a predefined distance from an intersection.

3.3. IWCAP Protocol

The proposed protocol operates on three phases: the *synchronization phase*, the *contention phase*, and the *transmission phase*, as shown in Figure 16. In the *synchronization phase*, vehicles approaching the intersection sense the channel for an idle status for T_{sync} slot times. If the channel is sensed idle during this period of time, then vehicles can start the next phase: the *contention phase*.

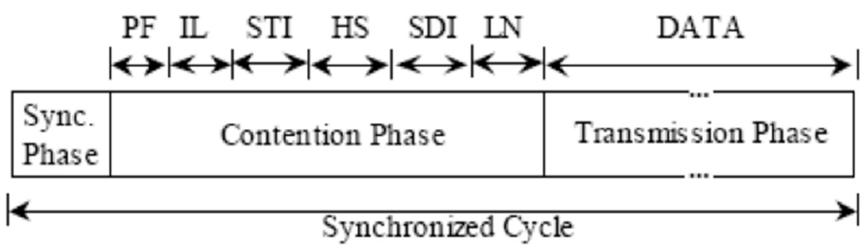

Figure 16. The intersection warning channel access priority (IWCAP) protocol with three phases: synchronization, contention, and transmission

In the *contention phase*, vehicles contend for the channel using six variables. We assume that each binary bit in a variable is represented by one IEEE 802.11a slot time. Figure 17 describes an algorithm on how vehicles contend for their assigned channel. Vehicles contend for the channel one bit at a time, MSB first. A contending vehicle will listen to the channel for a slot time if the value of its bit is 1, and will burst the channel for a slot time if the value of its bit is 0, as shown in Lines 10-13 of Figure 17. If a contending vehicle senses the channel is busy, this vehicle will drop from contending for the channel. If a contending vehicle senses the channel idle for all its 1 bits, then this vehicle continues to contend for the assigned channel. In the *contention phase*, vehicles approaching the intersection contend for the channel based on the following six sub phases:

3.3.1. Priority Flag (PF)

The objective of this sub phase (in collaboration with the next sub phase) is to provide fairness for all vehicles approaching the intersection from all intersection legs to transmit their data. By default, the Priority Flag for each vehicle is set to 0. Vehicles that lose the next sub phase set its Priority Flag to 1, as explained in the next sub phase. Setting the Priority Flag to 1 gives those vehicles a higher priority to win the channel on the next contention cycle.

3.3.2. Intersection Leg (IL)

In our system architecture, we assume that each intersection leg has a unique value. The value of an Intersection Leg is assumed to have a length of Li bits. The objective of this sub phase is to allow vehicles traveling from one of the legs to continue contending for the channel using the next sub phases. Other vehicles traveling from the other legs lose the contention for the channel. If a contending vehicle senses the channel busy, this vehicle will drop from contending for the channel and set the Priority Flag to 1, as shown in Line 17. Setting the Priority Flag to 1 gives those lost vehicles a higher priority to win the channel on the next contention cycle. The vehicles that win this sub phase will not be able to contend on next cycle since their Priority Flag is 0. In addition, when a vehicle wins the channel and transmits its data, this vehicle updates its Priority Flag to 0, as shown in Line 21.

```
ContentionSubPhases(ContentionStatus)
1  M = [~PriorityFlag,
2        IntersectionLeg,
3        ShortestTimeToIntersection,
4        ~HighestSpeed,
5        ShortestDistanceToIntersection
6        LaneNumber]
7  while (ContentionStatus = Unknown)
8    x= GetNextSubPhase(M)
9     for every bit B in x
10        if(B=0)
11            then burst the channel for a time slot
12        if(B=1)
13           then listen to the channel for a time slot
14               if (Channel=Busy)
15                   then ContentionStatus = Lost
16                       if (x= IntersectionLeg)
17                          then PriorityFlag = 1
18                       return ContentionStatus, PriorityFlag
19 if (ContentionStatus = Unknown)
20     then ContentionStatus = Won
21         PriorityFlag = 0
22         TransmissionPhase()
23 return
```

Figure 17. Describes an Algorithm on how Vehicles Contend for their Assigned Channel

3.3.3. Shortest Time to Intersection (STI):

In this sub phase, vehicles approaching the intersection from one leg contend for the channel. A vehicle that reaches the intersection first has a higher priority than other vehicles to win the channel and transmit its data. Vehicles calculate the time to an intersection from their speed and from the distance to the intersection. The value of the calculated time is assumed to be of type integer. The length of this measurement is *sti L* bits. Figure 18 and Figure 19 show six vehicles approaching an intersection and contending for the channel.

We assume that the length of Lsti is eight bits in this example. Notice that Vehicles V2, V3, V4, and V6 win this sub phase. These four vehicles reach the intersection in two seconds.

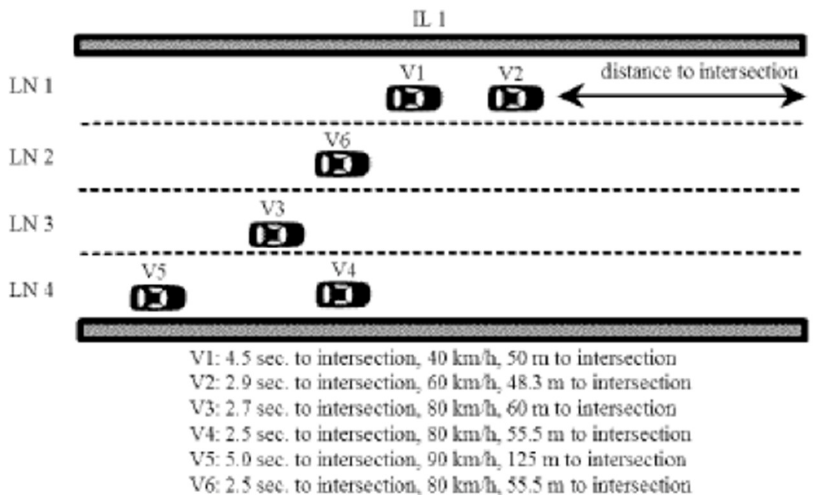

Figure 18. Six Vehicles that won the first 2 sub phases are contending for the channel using the next 4 sub phases

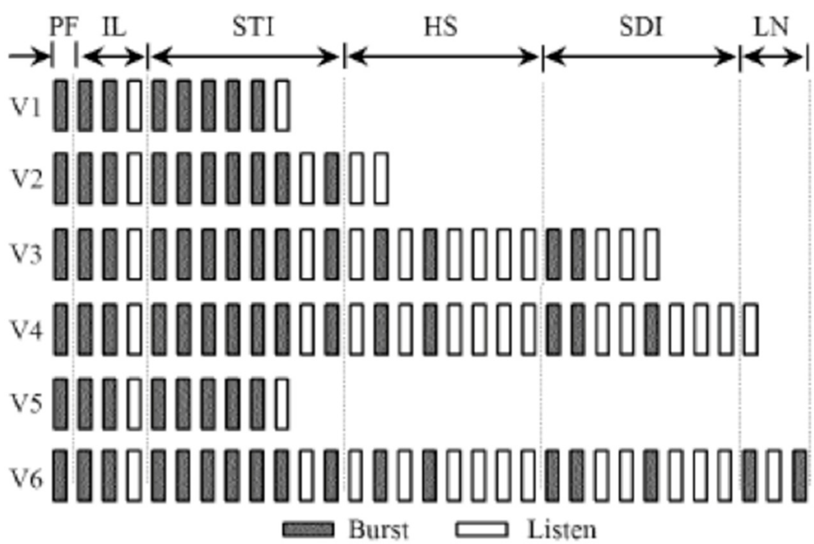

Figure 19. An illustration that shows slot times are used either to sense or burst the channel

3.3.4. Highest Speed (HS):

Since the value of the Shortest Time to Intersection is of type integer, vehicles with different speed may have an equal value of Shortest Time to Intersection. Since the impact of collision is greater with high speed vehicles, we assume that a vehicle with the highest speed has a higher priority than other vehicles to win the channel and transmit its condition. The value of the calculated speed is assumed to be of type integer. The length of this measurement is *hs L* bits. As shown in Figure 18 and 19, Vehicles V3, V4, and V6 win this sub phase since these vehicles have the highest speed of 80 km/h.

3.3.5. Shortest Distance to Intersection (SDI):

Vehicles with an equal speed value and an equal time to intersection value may have different value of distance to intersection. The reason is that we assumed the value of the Shortest Time to Intersection and the value of Highest Speed are of type integer. Therefore, we assume that a vehicle with the Shortest Distance to Intersection has a higher priority than other vehicles to win the channel and to transmit its condition. We assume that the value of the calculated distance is of type integer. The length of this measurement is *sdi L* bits. As shown in Figure 18 and 19, Vehicles V4 and V6 win this sub phase since these two vehicles have the shortest distance to the intersection, 55 m.

3.3.6. Lane Number (LN):

As shown in Figure 18, Vehicles V4 and V6 have the same Shortest Time to Intersection, same Highest Speed, and same Shortest Distance to Intersection. One of these two vehicles should transmit its conditions to avoid any data collision. To resolve this contention, we will use the Lane Number. The length of this measurement is l *L* bits. As shown in Figures 18 and 19, Vehicle V6 wins the *contention phase* and immediately starts the *transmission phase*.

In the *transmission phase*, a vehicle that wins the *contention phase* transmits its condition to other vehicles approaching the intersection from other legs. The winning vehicle transmits its data using the same omni directional antenna that is used in the *contention phase*. We assume the following will be transmitted: vehicle speed (vs), vehicle heading (vh), vehicle position (vp), and a checksum (CRC). Table 3 shows our proposed settings for IWCAP protocol.

Table 2. The proposed IWCAP parameters

IWCAP Parameter	*Value*	*Range*
T_{sync}	10 slot times	--
T_p	1 slot time	--
L_i	3 bits	1-7 legs
L_{sti}	8 bits	1-254 seconds
L_{hs}	8 bits	1-255 km/h
L_{sdi}	8 bits	1-254 meters
L_l	3 bits	1-7 lanes
vs	8 bits	1-255 km/h
vh	9 bits	1-360 degrees
vp	64 bits	--
crc	16 bits	--

3.4. Protocol Analysis

3.4.1. Synchronization Phase

To participate in this protocol, a vehicle has to sense an idle channel for T_{sync}. We set T_{sync} to 10 slot times since the maximum number of idle slot times is 9 (when IL=3 and STI=254). On the other hand, if a vehicle wins the *contention phase*, then this vehicle will not contend for the channel for the next *n* cycles. This is to allow other vehicles to transmit their conditions. The number of cycles, *n*, depends on the number of lanes, the number of legs, the wireless range of the omni directional antenna, and vehicles' length. To increase the bandwidth per vehicle, a vehicle that wins the *contention phase* will not contend for the channel for the next *n* cycles or if it senses an idle channel for duration of $T_{sync} + \Delta t$. If Δt is zero, then this vehicle will always detect T_{sync} on the next cycle and participate again in the *contention phase*. Therefore, Δt is used to detect an idle channel where no other vehicle traveling on the same leg exists to contend for the channel. The value of Δt must be at least equal to T_{sync}.

3.4.2. DSRC PHY and MAC Layers:

Our proposed protocol can be used as an add-on to the IEEE 802.11a MAC protocol. Instead of using the random back-off algorithm in 802.11a, the distributed coordination function (DCF), the intersection warning system can use the *contention phase* in our proposed protocol. In addition, the same DSRC PHY layer can be used in our proposed protocol.

3.4.3. Communication Range and Distance to Intersection Analysis:

In our analysis, we study the required communication range and the distance to an intersection to avoid a collision. The distance to an intersection is given by

$$d = \frac{v^2}{2 \cdot g \cdot (f + G)} + d_p + d_b \qquad \textbf{(3.1)}$$

where v is the velocity of a vehicle, g is the acceleration due to gravity (9.81 m/s2), f is the coefficient of friction, G is the grade (slope) of a road, d_p is the distance a vehicle has traveled after processing the protocol, and d_b is the distance traveled during the brake reaction time. The brake reaction time is the time between recognizing the warning and applying the brake. We assume that the brake reaction time is 2.5 s [26] if drivers manually apply the brake and a 0.1 s if an automated braking system is developed to react when a warning is received. The frictional force between tires and the road is variable and depends on the pavement and tire conditions. We will assume that f varies between 0.29 (wet pavement [26]) and 0.8 for 96.5 Km/h (60 MPH). The grade of a road is assumed to be -7%.

To calculate d_p, the protocol has 31 slot times in the *contention phase*, and 10 slot times in the *synchronization phase*. According to the DSRC specification [23], a slot time is 15×10^{-6} s. Therefore, our proposed protocol will spend 615×10^{-6} s in the *synchronization and contention phases.* Further, for our analysis, a vehicle transmits 36 bytes, including the IEEE 802.11a frame header, using one of the DSRC data rates. We also make the following assumptions. There is a four-leg intersection and four lanes per leg. The width of a lane is 3.7 m (12 ft.). There is a divisional island between two opposite directions. The width of the island is 3.7m.

Therefore, the dimension of an intersection zone is 34m×34m, and the maximum communication range equals $34 + 2 \cdot d$ m. We vary the number of vehicles approaching an intersection from 1 to 600 vehicles. Each of these vehicles has a chance to win the channel and transmit its condition to other vehicles. Figure 20 and Figure 21 show the maximum distance, *d*, to an intersection for break reaction times of 2.5 s and 0.1 s respectively. Figure 22 shows the vehicle's required communication range for two DSRC data rates (6 Mbps and 54 Mbps), under different friction coefficients, and a brake reaction time of 2.5 s. Figure 23 shows the vehicle's required communication range if the brake reaction time is 0.1 s. For a dry pavement (f=0.8), the required communication range is ~270 m as shown in Figure 22. If we consider wet pavement (f=0.29) in designing the intersection collision warning system, then the required communication range is ~500 m. The communication range can be reduced by ~100 m if an automatic braking system is used to slow down the vehicle. In addition, as shown in Figure 22 and Figure 23, the DSRC data rates have minimal effect on the required communication range.

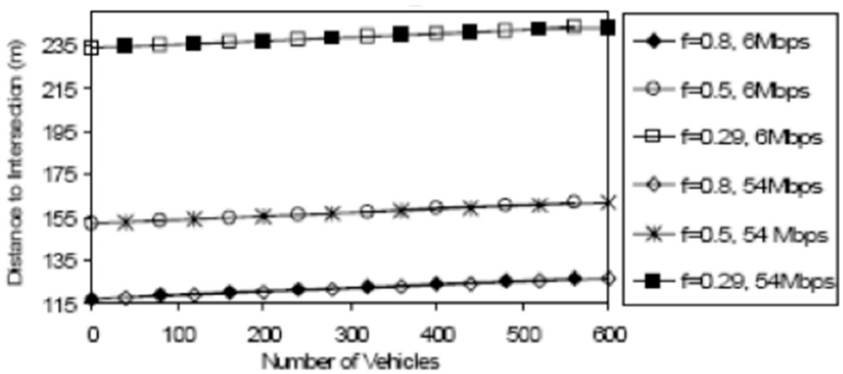

Figure 20. Required distance to an intersection with a brake reaction time of 2.5 s and a range of friction coefficients

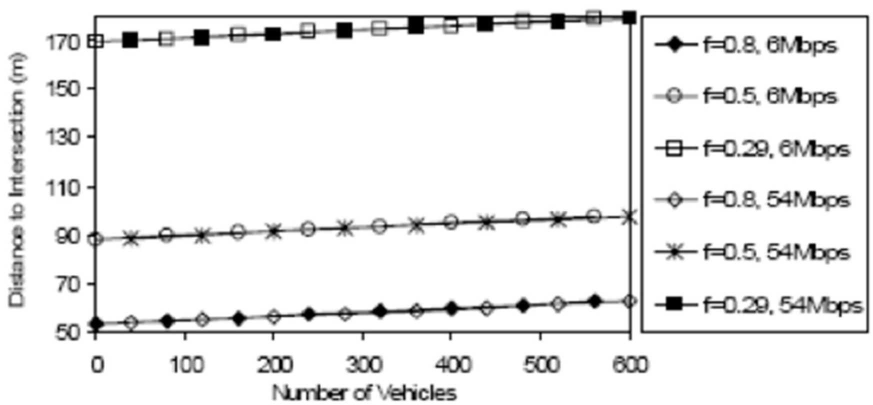

Figure 21. Required distance to an intersection with a brake reaction time of 0.1 s and a range of friction coefficients

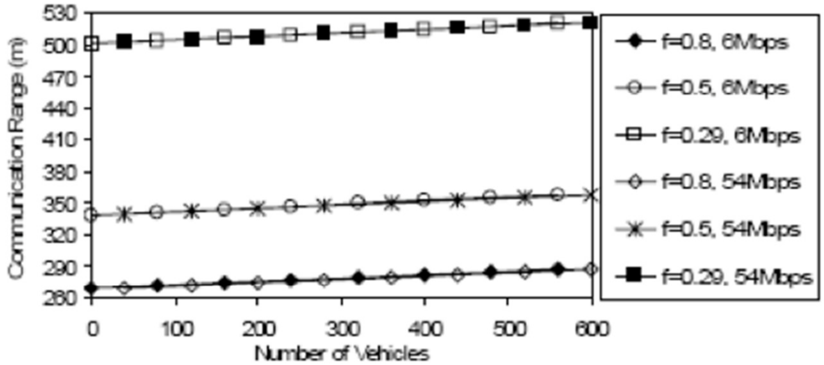

Figure 22. Required communication range with a brake reaction time of 2.5 s and a range of friction coefficients.

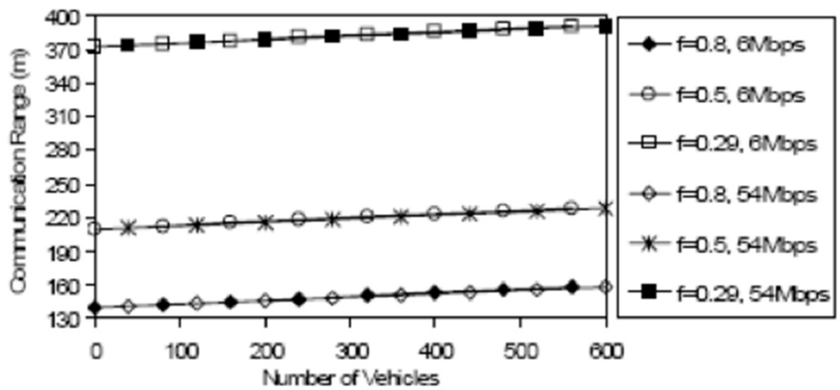

Figure 23. Required communication range with a brake reaction time of 0.1 s and a range of friction coefficients.

3.4.4. Bandwidth and Latency Analysis:

The bandwidth and latency of our proposed protocol depend on the number of intersection legs, number of lanes, number of vehicles approaching the intersection, and the processing time of the protocol. The bandwidth required per vehicle is given by

$$BW = \frac{36}{p \cdot \sum_{n=1}^{n=l} V_n} \text{ bytes/sec} \tag{3.2}$$

and the latency is given by $p \cdot \sum_{n=1}^{n=l} V_n$, where p is the processing time of the *synchronization*, *contention and transmission phases*, l is the number of legs, and V_n is the number of vehicles within distance, d, per leg l.

The value of V_n is given by

$$V_n = \frac{d \cdot N}{L_v + (V_s \cdot t_h)} \tag{3.3}$$

Where N is the number of lanes per leg l, L_v is the average length of a vehicle, V_s is the speed of the vehicle, and t_h is the time headway of the vehicle. We assume that the average length of a vehicle is 5 m and the required distance, d, to an intersection is 245 m (from Figure 3.5). To study the average latency and bandwidth, a range of headways between 0.5 and 2 s are used with two velocities of 64 Km/h (40 MPH) and 96 Km/h (60 MPH). We assumed that vehicles approach an intersection from three legs with the same speed, and the vehicles at the fourth leg are not moving with a gap of 1 m between vehicles. Each leg has four lanes. Each of these vehicles has a chance to win the channel and transmit its condition to other vehicles. As shown in Figure 24, the latency increases with the increase in the number of vehicles approaching the intersection and decrease in the speed of vehicles. The maximum latency in our scenario is 0.2 s when vehicles are traveling at 64 Km/h with 0.5 s headway. Consequently, the total required bandwidth decreases with the increase in the number of vehicles and decrease in the speed of vehicles, as shown in Figure 25. The minimum bandwidth in our scenario is 0.19 Kbytes/s. In normal driving behavior, drivers maintain an average headway of 1.5 s and a speed less than 96 Km/h towards intersections. Therefore, a bandwidth of 0.19 Kbytes/s is sufficient under such normal driving conditions.

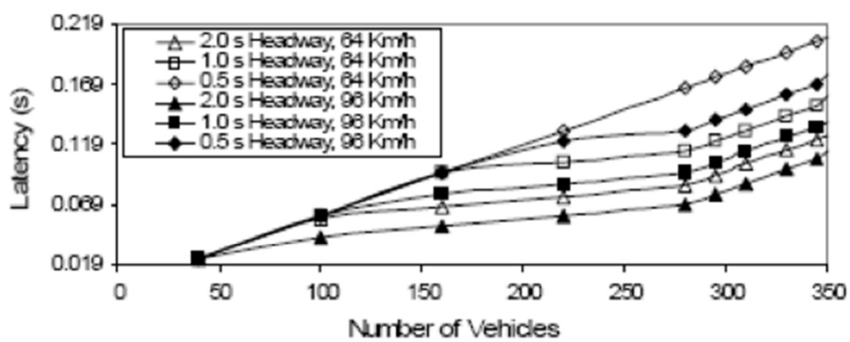

Figure 24. Protocol latency for a range of headways and two vehicle's speed

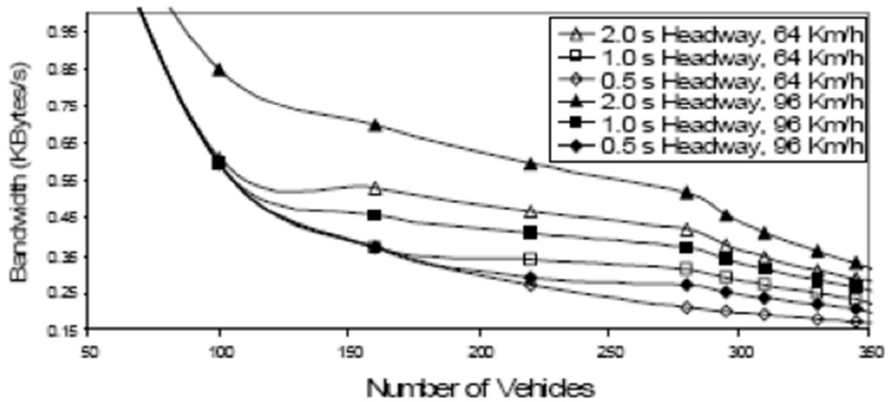

Figure 25. Vehicle bandwidth for a range of headways and two vehicle's speed

3.5. Conclusion

In this chapter, we proposed an intersection warning channel access priority (IWCAP) protocol to warn drivers of a possible collision when they approach an intersection. Our protocol utilizes one omni-directional antenna per vehicle and one DSRC channel for the intersection warning system. Our protocol provides priority and fairness for all vehicles approaching the intersection to transmit their conditions to other vehicles. The analysis shows that drivers can avoid a collision if a warning message is received within 240 m to an intersection, with a communication range of 500 m, a speed of 96 Km/h on a wet pavement, and if the warning message is received within 0.2 s after joining the wireless network. We are currently developing a simulator for several traffic scenarios on different intersection geometric designs.

Glossary of Acronyms

AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
COTS	Commercial Off-the-Shelf
DSRC	Dedicated Short-Range Communications
E_b/N_0	Energy of Bit-to-Noise Ratio
FCC	Federal Communications Commission
FHWA	Federal Highway Administration
FPGA	Field Programmable Gate Array
Gbps	Gigabit Per Second
GHz	Giga-Hertz
GPS	Global Positioning System
IP	Internet Protocol
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
ISO	International Standards Organization
IVPS	In-Vehicle Payment Service
IVTP	In-Vehicle Toll Processing
ITS	Intelligent Transportation Systems
MAC	Medium Access Control
MDOT	Michigan Department of Transportation
MHz	Mega-Hertz
MTU	Maximum Transmission Unit
OBU	On-Board Unit
OEM	Original Equipment Manufacturer
OFDM	Orthogonal Frequency Division Multiplexing
OS	Operating System
PC	Personal Computer
PER	Packer Error Rate
RF	Radio Frequency
RITA	Research and Innovative Technology Administration
RSU	Roadside Unit
SNR	Signal to Noise Ratio
TCP/IP	Transmission Control Protocol/ Internet Protocol
UDP	Universal Datagram Protocol
URL	Uniform Resource Locator
USDOT	United States Department of Transportation
V2I	Vehicle to Infrastructure
V2V	Vehicle to Vehicle
WAVE	Wireless Access in Vehicular Environments

Resulting Publications

1. M. Arafat, U. Mohammad, N. Al-Holou, Ph.D., M. A. Tamer, and M. Abdul-Hak, “Adaptive Traffic Light Controlling Methodology Under Connected Vehicles Concepts,” WORLDCOMP'12, Proceedings of the 2012 International Conference on Information and Knowledge Engineering (IKE'12), July 15-20, 2012, Las Vegas, Nevada.
2. Mohamad Abdul-Hak, Malok Alamir Tamer, Muhammad Arafat, Utayba Mohammad, Nizar Al-Holou, Youssef Bazzi, "Developing a Traffic Network Based on Wireless Communication to Reduce Vehicle Energy Consumption and Emission" Poster received “Bronze Paper Award,” at the May 2012 ITS-MI Annual meeting.
3. Malok Alamir Tamer, Mohamad Abdul-Hak. Muhammad Arafat, Utayba Mohammad, Nizar Al-Holou, “ITS-based Eco-Routing For Vehicle Navigation System,” Poster received “Silver Paper Award,” at the 2011 ITS-MI Annual meeting.

References

[1] “Traffic Control Systems Handbook”, Federal Highway Administration Report FHWA-HOP-06-006, October 2005, section 1.5: “System Evolution.”

[2] B. Zhou, J. Cao, X. Zeng, and H. Wu, "Adaptive Traffic Light Control in Wireless Sensor Network-Based Intelligent Transportation System", in Proc. VTC Fall, 2010, pp.1-5.

[3] Dissanayake, D.T.; Senanayake, S.M.R.; Divarathne, H.K.D.W.M. and Samaranayake, B.G.L.T.; "Real-Time Dynamic Traffic Light Timing Adaptation Algorithm and Simulation Software" in Industrial and Information Systems (ICIIS), 2009 International Conference, Dec 2009, pp.563-567.

[4] Al-Khateeb, K. and Johari, J.A.Y.; "Intelligent Dynamic Traffic Light Sequence Using RFID", in Computer and Communication Engineering, 2008. ICCCE 2008, May 2008, pp.1367-1372.

[5] Patel, M. and Ranganathan, N., "IDUTC: An Intelligent Decision-Making System for Urban Traffic-Control Applications", in Vehicular Technology, IEEE Transactions, VOL. 50, No. 3, May 2001, pp.816-829.

[6] Michel Ferreira , Ricardo Fernandes , Hugo Conceição , Wantanee Viriyasitavat , Ozan K. Tonguz, Self-organized traffic control, Proceedings of the seventh ACM international workshop on Vehicular InterNETworking, September 24-24, 2010, Chicago, Illinois, USA.

[7] V. Gradinescu, C. Gorgorin, R. Diaconescu, V. Cristea and L. Iftode, "Adaptive Traffic Lights Using Car-to-Car Communication", Vehicular Technology Conference, 2007. VTC2007-Spring. IEEE 65th In Vehicular Technology Conference, 2007. VTC2007- Spring. IEEE 65th (2007), pp. 21-25.

[8] Shih-Fen Cheng, Epelman, M.A., Smith, R.L., "CoSIGN: A Parallel Algorithm for Coordinated Traffic Signal Control", VOL. 7, No. 4, Dec 2006, pp.551 - 564.

[9] US EIA 2009, www.eia.doe.gov

[10] U.S Department of Transportation Research and Innovative Technology Administration http://www.its.dot.gov/

[11] Yan-Jing Wu, Shih Chien and Wei-Cheng Sung [2009], A Dynamic Navigation Scheme for Vehicular Ad Hoc Networks.

[12] Toshiaki Kono, Takumi Fushiki [2009], FUEL CONSUMPTION ANALYSIS AND PREDICTION MODEL FOR “ECO” ROUTE SEARCH.

[13] Raghu K. Ganti, Nam Pham, Hossein Ahmadi, Saurabh Nangia, and Tarek F. Abdelzaher[2010], GreenGPS: A Participatory Sensing Fuel-Efficient Maps Application.

[14] Jian-Da Wu, Jun-Ching Liu [2010], Development of a predictive system for car fuel consumption using an artificial neural network.

[15] http://ict-itetris.eu/

[16] www.nhtsa.com

[17] http://www.itsa.org/

[18] http://www.its.dot.gov/

[19] U.S Department of Transportation, National Highway Traffic Safety Administration, Vehicle Safety Communications Project; Task 3 Final Report; Identify Intelligent Vehicle Safety Applications Enabled by DSRC, March 2005.

[20] A. Dogan, G. Korkmaz, Y. Liu, F. Ozguner, U. Ozguner, K. Redmill, O. Takeshita, and K. Tokuda, “Evaluation of intersection collision warning system using an inter-vehicle communication simulator,” in Proc. of the 7th International IEEE Conference on Intelligent Transportation Systems, 3-6 Oct. 2004, pp. 1103 – 1108.

[21] R. Naumann, R. Rasche, J. Tacken, and C. Tahedi, “Validation and simulation of a decentralized intersection collision avoidance algorithm,” IEEE Conference on Intelligent Transportation System, 9-12 Nov. 1997, pp. 818 – 823.

[22] Standard Specification for Telecommunications and Information Exchange Between Roadside and Vehicle Systems — 5 GHz Band Dedicated Short Range Communications (DSRC) Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Sept. 2003.

[23] S. V. Bana and P. Varaiya, "Space division multiple access (SDMA) for robust ad hoc vehicle communication networks," in Proc. of the 4th IEEE International Conference on Intelligent Transportation Systems, Oakland, CA, 2001.

[24] S. R. Mosra, S. Shanker and S. M. Mahmud, “An intelligent architecture for issuing intersection collision warnings,” in Proc. of the 4th Intelligent Vehicle Systems Symposium of National Defense Industries Association (NDIA), National Automotive Center and Vectronics Technology, June 22 –24, 2004, Traverse City, Michigan, pp. 176-183.

[25] American Association of State Highway and Transportation Officials (AASHTO), A policy on geometric design of highways and streets,1990.