A Transportation Control System for Urban Environments

Catalin Fratila, Ciprian Dobre, Florin Pop, Valentin Cristea

Department of Computer Science and Engineering

University POLITEHNICA

Bucharest, Romania

E-mails: catalin.fratila@cti.pub.ro, ciprian.dobre@cs.pub.ro, florin.pop@cs.pub.ro, valentin.cristea@cs.pub.ro

Abstract— **Wireless technologies can help solve traffic congestions in urban environments, where road infrastructures develop slower than the sometimes exponential growth in the number of cars in traffic. We present a traffic control and congestion avoidance system developed over a vehicular ad-hoc network created between the cars in traffic and the road infrastructure. We propose a solution for monitoring traffic using not only sensors within the road infrastructure, but also the cars themselves acting as data collectors. The traffic control decision, provided by the road infrastructure, is scalable, load-balanced, and uses correction decisions for the route adjustment based on local areas. We present evaluation results that show the capabilities of the proposed congestion avoidance model.**

Keywords: congestion avoidance; VANETs; traffic control.

I. INTRODUCTION

Traffic congestions are realities of urban environments. The road infrastructure capacities cannot cope with the rate of increase in the number of cars. This, coupled with traffic incidents, work zones, weather conditions, make traffic congestions a major concern for municipalities and research organizations [1]. Advanced traffic control technologies may lead to more efficient use of existing road network systems, resulting in reduced traffic congestion, delays, emissions, energy consumption and improved safety. The communication capabilities provided by modern wireless technologies and mobile devices offer opportunities for the development of traffic control applications where cars and devices within the road infrastructure collaborate to solve traffic problems.

In this paper we propose a system for traffic control in large cities. A real-time model of the traffic is constructed using data collected by sensors within the road infrastructure (sensing cars passing by), and by the applications running inside cars (sensing location data). For traffic control the system assumes the existence of several servers, spread around the city, where local sensed data is aggregated. In this approach each server is responsible for a traffic area around it. It aggregates all local sensed data (from all cars and infrastructure-level sensors within the area) into a real-time model of traffic. These severs can communicate and exchange route updates. The aggregated data is further sent back (feedback) towards cars, which in turn can dynamically adjust their travelling routes.

An immediate advantage of our proposed approach is that cars become data sources, and can help better collect traffic data. The approach provides scalability in terms of computation (each server is responsible only for monitoring a small area - for big cities this means adding more servers) and communication (each car receives traffic updates only for roads within its current position). Also, the driver receives on his/her navigator feedback from the system about possible congestions and to dynamically adjust its route.

The paper is structured as follows. In Section 2 we present related work. Section 3 presents the architecture and the model used for congestion avoidance. To evaluate the capabilities of our proposed solution, we present evaluation results obtained in simulation scenarios. In Section 4 we present details for the implementation of this system as an extension of a simulator designed for VANETs, together with an analysis of the obtained experimental results. In Section 5 we give conclusions and present future work.

II. RELATED WORK

The problem of finding the optimum route for a destination in a crowded city has been previously approached in several research projects [5, 6]. The idea is to equip cars with navigators able to receive feedback from traffic and dynamically adjust (while moving) their routing paths. These navigators receive traffic updates and decide locally on the optimum path to choose – where optimality is addressed either in terms of time (compute the routes such that to decrease the time cars spend in traffic) or distance (compute the routes such that to decrease the length of the paths considering that cars pollute). We argue that a correct traffic control system should not rely on cars making the correct decision. By letting cars control and choose their routes a situation where all cars see the same congestion roads and decide on the same alternative road – which will eventually congest the alternative road – is easily encountered.

Several projects were previously proposed to solve the congestion problems in modern cities. Today many traffic lights are already equipped with sensors or cameras to detect the number of cars queuing on red light [8]. In addition, traffic lights can be augmented with sensors to collect data for traffic, and processing capabilities to aggregate traffic data and take actions such as updating the time for the change of red or green lights or to log information for an economic pricing model to control congestions. The adaptation can also use sensors deployed in key points within the road infrastructure. However, for congestion avoidance a more appropriate model should reflect the current status of all roads. A system based on sensors collecting data in key points takes decisions that reroute cars from roads where it knows of possible congestions to other roads where it has no information (and, consequently, it thinks there is no congestion).

Projects such as [5] approached the problem of computing optimal adjacent routes using learning techniques. After driving on a route, the driver manually inputs the time spent in traffic, and in the future he/she will be able to select the best route depending on the average speed associated with known routes. However, such applications are limited, because they consider that traffic conditions are static. They also ignore events such as different weather conditions, accidents, etc, that are considered by many authors the roots of congestions. Alternative solutions [6] process data received from sensors within the road infrastructure (infrared cameras, asphalt integrated sensors, etc.) and compute the route dynamically, based on the traffic conditions. The authors present results showing that the alternative expert system is able to calculate a shortest path, but it also needs a large amount of data for computing the route. Also, the integration of sensors in every city intersection and/or road is costly, an aspect which can negatively influence the large-scale adoption of such a solution. At the moment, applications based on such types of sensors are used mainly on highways, and the services they provide are limited to traffic flow adjustment.

WAZE [3] is a free social traffic and navigation application that uses real-time road reports from drivers to improve the navigation decisions. Similar to our approach, WAZE uses a social layer, where drivers work together to report and receive the most relevant traffic information available at any given moment.

In this paper we propose a model which relies on the data also collected on a social basis from the traffic participants (cars travelling), but this is combined with data from sensors within the road infrastructure, to take traffic decisions. Unlike WAZE, we propose an approach where data is automatically collected by the mobile devices and aggregated, such that to take higher level decisions (such as the average speed cars are moving on a particular road segment). Also, by making use of alternative sources (sensors) and combining our approach with security solutions we previously proposed [7] we solve many of the problems that social networks have (intruders for example). We use a number of distributed servers that are in charge of different traffic zones. Such an approach has advantages on communication (the traffic data is aggregated locally), processing (data is collected from a smaller area, so the amount of data is reduced), and scalability.

III. A SYSTEM FOR TRAFFIC CONTROL

The system considers the existence of several entities (see Figure 2): cars equipped with wireless devices, traffic lights acting as communication gateways between cars (wireless communication) and a local server (wired communication). The city is divided in traffic areas called, in this paper, Traffic Zones (TZ). Each TZ is under the control of a traffic light.

We consider that cars are equipped with wireless devices that run a navigation application. This mobile application can use the device's sensors (e.g., GPS sensor) and constantly sends the sensed traffic conditions (travelling speed, position) to the server. In return, it receives information used to dynamically adjust its travelling path. The monitored data is sent to the traffic light using close-range wireless communication. When the car is not able to directly communicate with a traffic light, it can either collect data about traffic conditions that it stores locally until it is able to wirelessly send it, or it can send the monitored data using 3G or WiMAX wireless technologies. Because of communication costs, we consider that most drivers will prefer the collect-store-deliver approach.

Because they integrate communication capabilities, in the rest of the paper we refer to the traffic lights as Wireless Traffic Lights (or WTL). We also make the observation that alternative road-sided communication equipments can be used as well. Like cars, WTLs can also collect real-time data from traffic. A server connected to each WTL aggregates the data received from cars.

To support the gateway role, WTLs have to be able to communicate with the server, as well as with cars in traffic. The system would only require the addition of limited-capability processing and wireless communication equipments. The costs for such an investment could be supported by the gains in fuel consumption and street de-congestion. In addition, the inclusion of fixed-location traffic lights in the system can support the adoption of security mechanisms [4, 7]. To cope with possible attackers that might negatively influence the traffic conditions by reporting wrong data, the system can use the model for trust computation based on collaboration between cars proposed in [9]. In addition, a WTL can also communicate with other neighbour WTLs and send them its knowledge of the traffic - the costs associated with the roads it knows about (see Figure 1).

The obtained model of the traffic is used to implement a congestion control mechanism, where cars receive in support information about traffic conditions, and dynamically adapt their travelling routes. Several aspects are considered. First, the system balances traffic by sending different data to different cars. If all cars would receive the same information, for example congestion on road A and

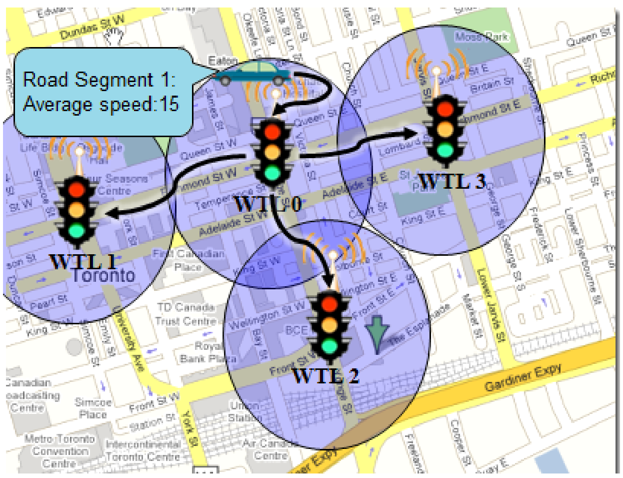

Figure 1. WTLs exchange traffic updates with cars and between themselves.

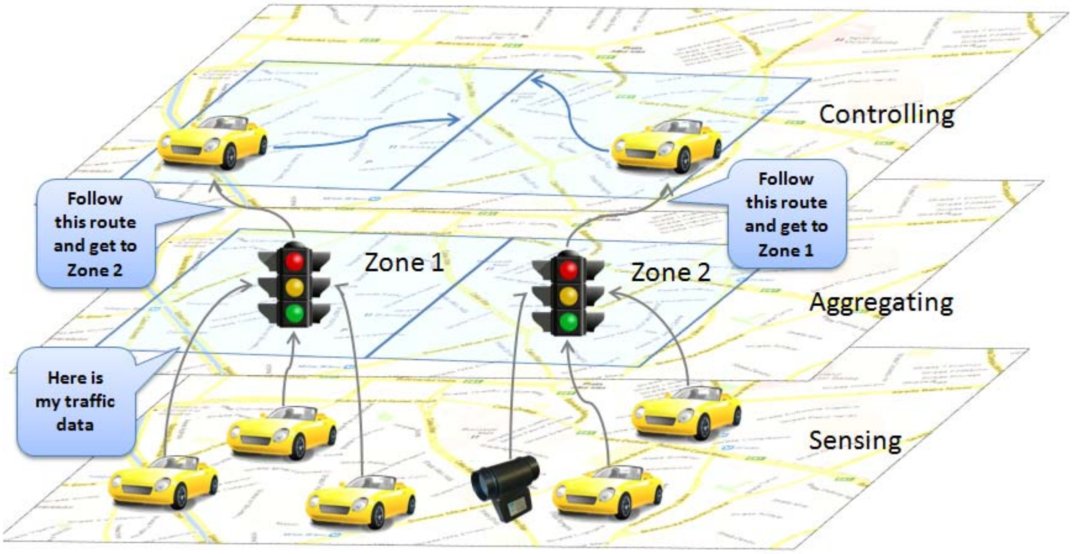

Figure 2. System architecture.

good travelling conditions on road B, then all cars would deviate their routes to road B (thus, moving congestion from A to B). Secondly, cars are equipped with devices with limited computation and communication abilities. The driver expects results to be delivered as fast as possible to support his/her decisions (if the mobile application inside the car receives information about a congestion just after the driver already enters the congested roads, the feedback will not be of much help anymore).

In summary, to support the traffic optimization, the proposed system implements three phases (see Figure 2): traffic monitoring, data aggregation and the development of traffic model, and traffic controlling using data feedback and dynamic route adaptation.

3.1. Traffic model

The local traffic model (within a TZ) is a directed weighted graph, where nodes are intersections and the edges represent road segments (we consider that a road is divided into several *segments*). The weight associated with a street segment represents the cost of travelling on that particular road. To compute this cost we start from the observations of particular cars travelling on that road segment. These observations support the computation of a *congestion degree* (Γ), having a value between 0 and 255, depending on the load of the street segment: 0 represents an empty street, and 255 represent a heavily congested street. This weight is computed as follows:

$$\Gamma_{car} = \begin{cases} 255 * \left(1 - \frac{v_m}{v_{Max}}\right), & v_m \le v_{Max} \\ 0, & v_m > v_{Max} \end{cases} \quad (1)$$

where Γ_{car} represents the congestion degree observed by a particular car, v_m is the average travelling speed of that car on that road segment, and v_{Max} represents the maximum allowed travelling speed (depending on the local legislation) on that road segment.

The average speed (v_m) is computed considering the average travelling speed until the car reaches the end of the segment, coupled with the number of times the vehicles stop in traffic while passing the segment (e.g., the number of times the car stops at the red light, considering a traffic light ending the current road segment). This is correlated with the flow of vehicles passing through the intersection situated at the end of the road segment. This is measured in number of vehicles per second (it can happen that cars have problems passing through an intersection because of other connected congested roads). The formula to compute the average speed is:

$$v_m = v_{cruise} * \left(1 - \frac{n_R}{n_{RMax}} * \frac{F_{\min}}{F}\right) \quad (2)$$

where v_{cruise} represents the average cruising speed (between two consecutive 'stops' in traffic, or until the car exists a road), n_R is the number of red light cycles the vehicle waited on, n_{RMax} is the maximum number of red light cycles a vehicle could wait on that particular road segment, F is the frequency of cars passing through the intersection, and F_{min} is the minimum frequency cars can pass by, as monitored for that intersection for the last hour.

For an intersection equipped with traffic lights, the average speed has a maximum value (v_m=v_{cruise}) when the traffic light stays green until the car crosses the intersection (n_R=0). It has a minimum value (v_m=0) when the car has to wait on many red cycles, and the frequency of cars passing through the intersection is minimum (both n_R=n_{Rmax} and F=F_{min} happen simultaneously). The average speed increases when the number of red cycles decreases, or when the frequency of cars passing through the intersection increases.

The server computes the congestion cost of the different monitored (or controlled by it) local road segments, by collecting the congestion degrees for all cars passing by the traffic light. These values are used to compute an average value (after statistically eliminating possible outliers from observed data) for each road segment. These averaged values (Γ denotes the average value of monitored Γ_{car} values) are used to further compute an average congestion degree of each road segment for the current hour (denoted by Γm).

3.2. Traffic control on a global scale: update propagation between WTLs

A WTL communicates and possibly updates routes similar to a distance-vector routing (e.g., RIP) protocol [11]. It employs a distance-vector routing algorithm, where WTLs act as routers, communicating with their neighbours and offering their local knowledge about the network, as well as computing the best route for each TZ, consisting of the "next hop" - which is also a WTL and corresponds to the next traffic zone. When a vehicle asks for the best route towards a destination, the WTL offers a route necessary for the vehicle to reach a “next hop” destination (in our case, the next TZ that minimizes costs towards the car’s final destination). When the vehicle enters the next zone, it makes another inquiry to the WTL in charge of that TZ.

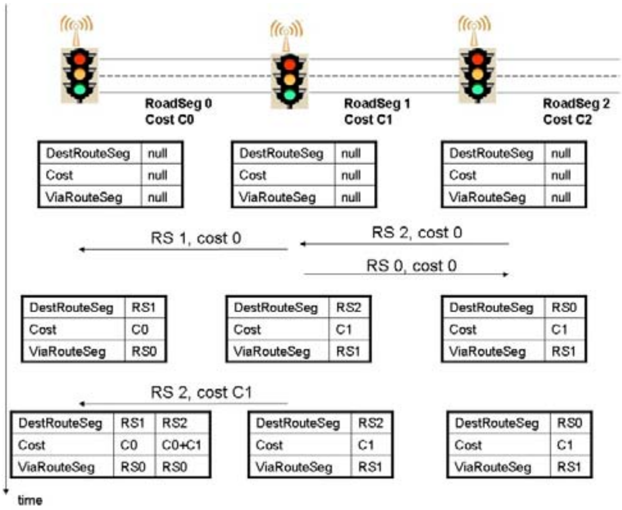

Figure 3. The distance-vector routing approach for route exchanges.f

Initially, each WTL knows only the costs to the route segments in its TZ, considering the mechanism presented in the previous Section. Also, each WTL computes the reaching its WTL neighbours. At first, this is done using a default speed (considering static road conditions), but each update from cars/sensors in traffic adjusts the costs to reach its neighbours.

For the implementation of this system, a vehicle periodically broadcasts messages containing data recorded while passing the last few road segments and the vehicle’s destination. Besides the travelling speed and number of stops, in the message that each car permanently broadcasts there are several fields that WTL interprets. A broadcast message includes the vehicle's destination and the position of the next WTL on its route that the vehicle reaches. Every time the vehicle passes a route segment, it adds to the current broadcast message information about the road segment it just passed by, and the average speed and number of stops monitored while passing. The purpose is to provide real-time feedback regarding the traffic conditions on the route segments that the vehicle has just passed (see Section 3.1). When the WTL receives a broadcast, it acknowledges the message. When the vehicle receives such an acknowledgement, it resets the list of visited road segments. The response message contains a set of route segments that WTL associated with that particular destination (an updated route to the next TZ). For the moment this is computed using an A* search applied on the traffic model (and balancing the routes between cars). This may be extended to allow the WTL to predict future congestions. In both approaches, the WTL directs the car to the next WTL, where the vehicle will be notified of next route; and so on, until it will successful reach its destination.

When the WTL receives a broadcast message, it executes the following. The data about sensed road conditions is used to update the costs associated with the associated roads. The data about other road segments (belonging to other Traffic Zones) is sent by the WTL to its neighbours. This is because with a high probability when a vehicle computes the average speed of a road segment, it will already have reached the wireless area of another WTL directly connected to the specific road segment. For example, if a vehicle runs from WTL A to WTL B, when it reaches B, it will compute the average speed of the A-B road segment. With high probability the WTL that receives the broadcast message will be B. Still, WTL will send the information to all its neighbours (in this case to WTL A).

A WTL uses three lists: 1) the known route segments, 2) the costs associated with each road segment, and 3) the road segments that a vehicle must go to for particular destinations (located on a route segment from the first list). In the beginning these lists are empty. Every WTL advertises to its neighbours that it can reach the route segments directly connected to it with a null cost. When a neighbour receives such an advertisement, it checks that the advertised route segment is not directly connected, and if it is not, it adds the route segment to the “routing table”, putting the associated cost of the route segment that must be passed to reach the neighbour who advertised the route segment. Therefore, in this step all WTLs know how to reach particular street segments from adjacent TZs.

Once a WTL receives information about a route that was not present in its local “routing table”, it will send this information to all its neighbours through another advertisement. This action is performed several times, until the system reaches a state of convergence (see Figure 3). Furthermore, all WTLs will continue to advertise the route segments that are directly connected to it, to provide a fresh perspective to its neighbours and to force them to recalculate (if necessary) the cost associated with all the routes that required to pass through the traffic zone.

Because the WTL does not compute the entire route of the car it uses fewer CPU-intensive operations. In this case the WTL simply checks if a point on the map is located on any of the road segments that are in its local database, and if a match is found, it immediately send the response message with the corresponding entry in the database. The simplicity of the route computation is important because a car must receive a response before it passes through the intersection (in many occasions quite fast). Also, by not computing the entire route for a car the system can cope with city scenarios where traffic conditions are very dynamic and routes may change in several minutes after they were computed. It is often the case that a particular road segment gets very congested, so all vehicles passing through consequently report slow traffic speeds. This introduces a higher cost associated with the road segment, which will lead to the routing of cars on possible alternative routes. And it also happens that, after a period of time in which vehicles are not routed through that particular road segment anymore, the road segment becomes free and cars might pass very quickly, with a low cost. But because there is no car to pass through the system does not know the state of the road segment. To prevent such problems, a periodical adjustment is made to all speeds (and costs) of all road segments of the map. This adjustment tries to draw the speed of a road segment near its standard speed, creating the possibility to reduce the cost of a road segment that has developed a very high one. The frequency of this operation is set so as to not create a false cost to a road segment. Its only purpose is to help road segments with high cost regain their "credibility" (lower their cost) once no vehicles pass through.

IV. EXPERIMENTAL RESULTS

We evaluated the proposed system for traffic control in a simulated environment. For that we extended the VNSim simulator [2], a generic VANET traffic simulator that uses both the microscopic and macroscopic models in order to accurately evaluate the performance of a wide range of VANET technologies.

On top of the already provided mobility and network models [2] we added the components and mechanisms previously described. The evaluation of the implementation of this system in the simulated experiments was accomplished using a simple scenario (see Figure 4). In this scenario cars can only take one of two alternative paths. We modelled a flow of cars travelling from north to south, and another one from south to north, using a normal distribution. At each of the two intersections we introduced the proposed WTLs. In the end, we compare the results obtained in the simulation with the analytical one, which validated the model and proposed implementation.

To evaluate the performances of the model we next considered a scenario consisting of 10 roads, 25 intersections with wireless equipments, and 60 road segments (see Figure 5 – a map of Downtown Seattle). The scenario considered as main roads the ones that intersect in the centre. The main roads have three lanes, and the secondary roads only two. In the experiments cars enter at three of the ends of the main roads, and go towards the fourth (cars come from west, north, and south and go east).

Figure 4. The scenario used for performance evaluation. Real-map of UPB

Using these conditions we conducted several experiments with different traffic flows. We were particularly interested in the number of vehicles that reach their destination in a specific amount of time in the simulation, the average time needed by the vehicles to reach their destination, the average fuel consumption, in litters/100 kilometres, and finally the total emissions resulted from the engines of the cars. For the last results the VANET simulator uses the model proposed by Akcelik and Besley for fuel consumption estimation [4]. To evaluate and compare the emissions of the vehicles, we chose to study the carbon dioxide, which is one of the four categories of toxic emissions of vehicles. The carbon dioxide is stated as kilograms/hour/car.

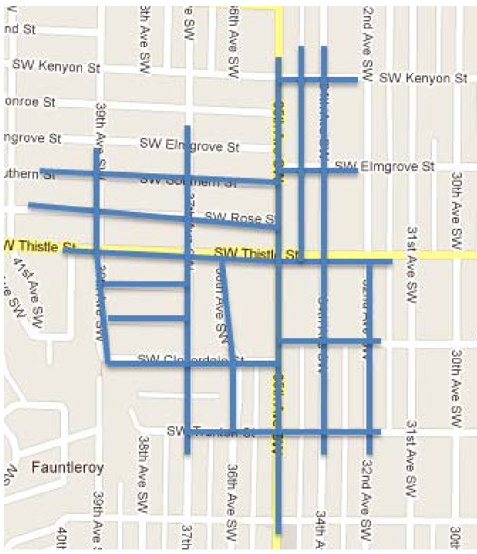

Figure 5. The scenario used in the simulation experiments. Real-map of Seattle, Washington, US.

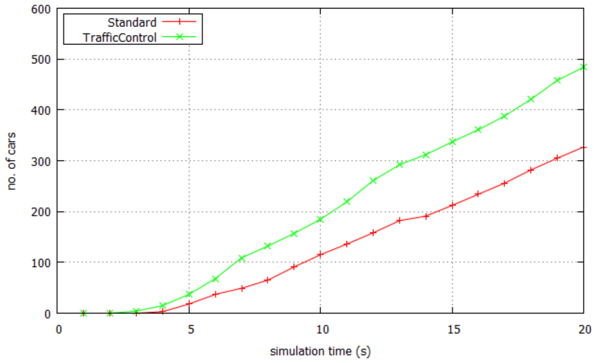

Figure 6a. Number of cars reaching their destination, 150 vehicles / lane / hour, non-adaptive traffic lights.

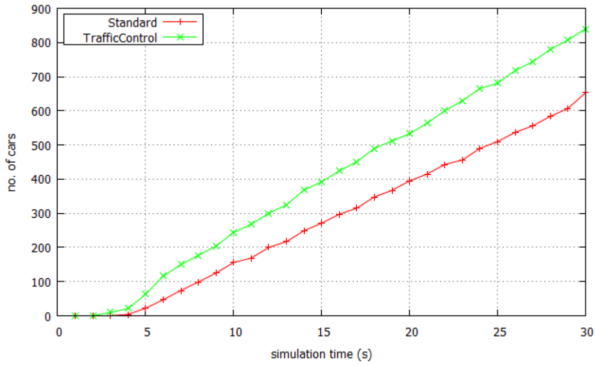

Figure 6b. Number of cars reaching their destination, 150 vehicles / lane / hour, adaptive traffic lights.

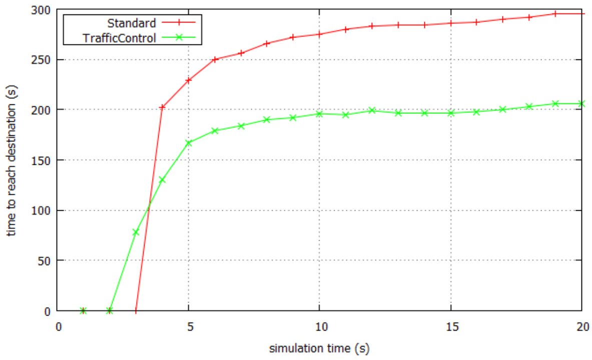

Figure 7a. Average time to reach destination, 150 vehicles / lane / hour, non-adaptive traffic lights.

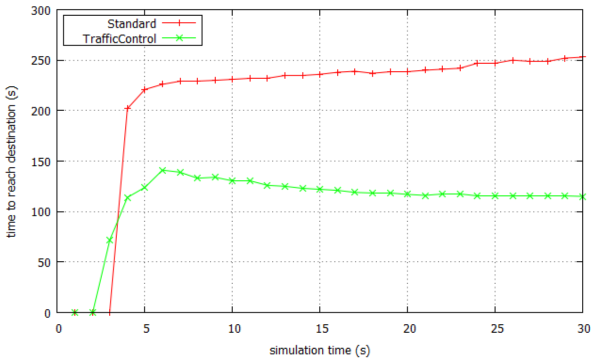

Figure 7b. Average time to reach destination, 150 vehicles / lane / hour, adaptive traffic lights.

Using this scenario, we conducted several experiments. We gradually increased the flow of cars, and conducted experiments where cars choose the shortest path to destination (the Standard series), and then using the proposed traffic control system (the TrafficControl series). We also conducted several experiments where WTLs can adapt their red-green cycle timing phases, depending on the queue length of cars waiting on the semaphore [4]. Results show that when using a high traffic flow, the adaptive traffic light system offered better results, while our proposed solution provided better results for lower traffic flows of vehicles. We also tried to use the two systems together and reached much better results than using them separately. These models, running concurrently, provided an increase in the total number of vehicles that reached the destination before the end of the simulation. The result doubled compared to the scenarios involving a high flow of vehicles.

Figures 6 (a, b) show the results obtained for the number of cars reaching their destination, for different scenarios. In all these experiments the proposed system manages to correct traffic conditions and led to an increase in the total number of cars reaching their destination (thus, cars that are not waiting in traffic anymore). In Figure 6b we demonstrate that by also adding the adaptive traffic lights into the simulation scenario the number of cars reaching their destination actually increase from 600 to almost 900 cars (considering the same amount of time).

Similar results were obtained for the average time vehicles need to reach their destinations (Figure 7a,b), the average fuel consumed by cars per time unit (Figure 8a,b), and the average emissions a car give into the atmosphere (Figure 9a,b), in several scenarios.

The results show an increase in the number of vehicles reaching the destination before the end of the simulation. For example, for the case of 150 vehicles / lane / hour, we noticed an increase of 45% in performance, from 327 to 484 cars. The results also show the average time cars need to reach their destination decreases by approximately 30%. We made comparisons between sets of experiments conducted without the use of the proposed model (the standard series) and the same experimental scenarios conducted using the traffic control model proposed in this paper.

Cars with petrol driven internal combustion engines are a constant source of air pollution. Road transportation is responsible for the emission of carbon monoxide, carbon dioxide, hydrocarbons, nitrogen dioxide, metals and many other organic compounds into the environment. As presented, the proposed traffic model also has the potential to reduce fuel consumption (as well as air pollution). We obtained, for example, a decrease of up to 14% in the emissions, from 1931 to 1646 kilograms of carbon dioxide/hour – when there are more cars in traffic then the road infrastructure can cope with, as in the case of 150 cars / lane / hour.

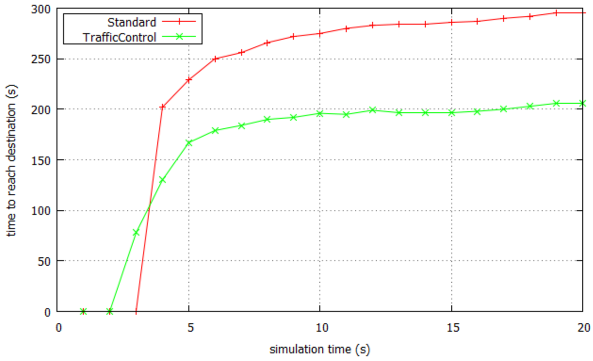

Figure 8a. Average fuel consumption, 150 vehicles / lane / hour, non-adaptive traffic lights

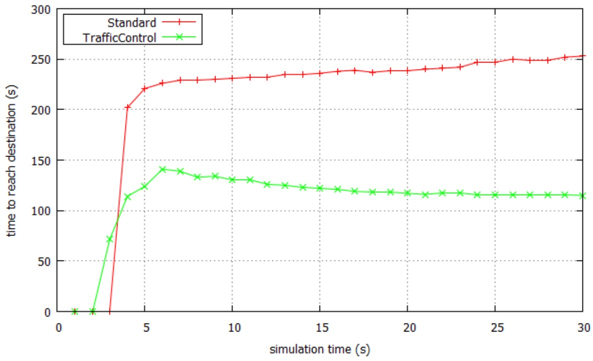

Figure 9b. Average fuel consumption, 150 vehicles / lane / hour, adaptive traffic lights

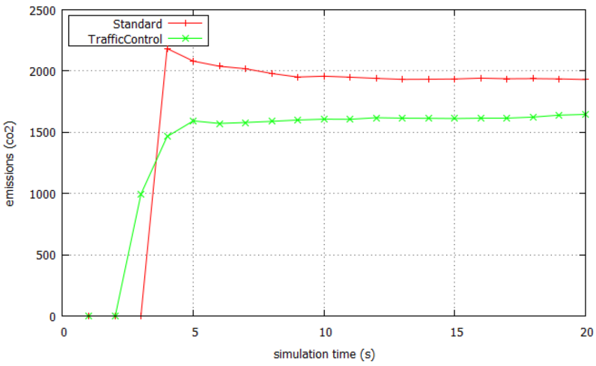

Figure 9a. Average emissions, 150 vehicles / lane / hour, non-adaptive traffic lights

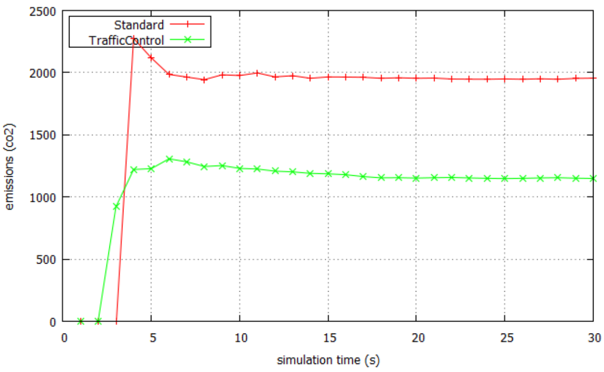

Figure 9b. Average emissions, 150 vehicles / lane / hour, adaptive traffic lights

Scenarios involving a lower traffic flow provided results that show an increase in the total number of vehicles that reach their destination before the end of the simulation as high as 15%, and also a shorter average time in which cars reach their destination, with a decrease of over 40%. The fuel consumption registers a slight decrease, which can prove to be very significant, considering that these scenarios involve a large number of cars. Regarding the average emissions, a decrease of about 20% has been registered.

V. Conclusions

We presented a model for traffic control and congestion avoidance in urban environments. Various studies [1] previously showed the important role played by the transport infrastructure in modern economies. It is estimated that only in Europe traffic congestion affect approximately 10% of the existing transport network, with significant financial implications. Such reports recognize that the development of new infrastructures is not a reliable solution as compared to investing in technologies based on intelligent transport systems (ITS) and Vehicular Ad-Hoc Networks (VANETs).

In this paper we proposed a traffic system designed to solve traffic congestions by collecting traffic data from the road, aggregating it and providing feedback to cars similar to ideas from networking protocols. It uses cars to collect traffic data and several WTLs that are able to aggregate and take decisions as to how to influence the routes the cars are driving. Whenever a road segment starts to provide lower average speeds for vehicles passing through, a routing algorithm provides alternatives routes, less congested and providing better times to reach destinations.

We evaluated this approach in a series of simulation experiments. We demonstrated that the average time needed for vehicles to reach their destinations registered a significant decrease of up to 40% compared with the time needed for vehicles to reach their destination using predefined static routes. The average fuel consumption also registers economies up to 1 litter per 100 kilometres, which is a major advantage considering the fact that in a city with millions of vehicles the fuel saving can be quite high. The total emissions also show a high decrease because of less fuel consumption and the fewer accelerations and brakes that vehicles need to apply. The emissions' decrease varies from 14% to 40%.

VNSim's official site and extended details on these experiments are available at http://cipsm.hpc.pub.ro/vanet.

Acknowledgement

This work was partially supported by project "ERRIC - Empowering Romanian Research on Intelligent Information Technologies/FP7-REGPOT-2010-1", ID: 264207. The research was co-funded by national project TRANSYS CNCSIS-PN-II-RU-PD ID: 238, and by the Sectoral Operational Programme Human Resources Development 2007-2013 of the Romanian Ministry of Labour, Family and Social Protection through the Financial Agreement POSDRU/89/1.5/S/62557. All authors have equal contributions to the article.

References

[1] US Department of Transportation, "Traffic Congestion and Reliability: Trends and Advanced Strategies for Congestion Mitigation", Final Report, September 1, 2005, accessed February 2nd, 2012, from http://ops.fhwa.dot.gov/congestion_report.

[2] A. Gainaru, C. Dobre, V. Cristea, A Realistic Mobility Model Based on Social Networks for the Simulation of VANETs. In: *Proc. of the IEEE 69th Vehicular Technical Conf.* (VTC Spring 2009). Barcelona, Spain, pp. 1-5, 2009.

[3] I. Lequerica, M. García Longaron, P.M. Ruiz, Drive and share: efficient provisioning of social networks in vehicular scenarios, *IEEE Communications Magazine*, 48(11), 2010, pp. 90-97.

[4] S. Mazilu, M. Teler, C. Dobre, Securing Vehicular Networks Based on Data-Trust Computation. In: Proc. of *3PGCIC 2011*, Barcelona Spain, pp. 51-58, 2011.

[5] B. Liu, Using Knowledge to Isolate Search in Route Finding, In: *Proc. of 14th International Joint Conference on Artificial Intelligence (IJCAI-95)*, Montréal, Québec, Canada, 1995, pp. 119-124.

[6] G. Eggenkamp, L.J.M. Rothkrantz, Intelligent dynamic route planning, In: *Proc. of the 13th Belgium-Netherlands Conference on Artificial Intelligence*, Amsterdam, 2001, pp. 381-388.

[7] C. Gosman, C. Dobre, V. Cristea, A Security Protocol for Vehicular Distributed Systems. In: Proc. of *SYNASC 2010*, Timisoara, Romania, pp. 321-327, 2010.

[8] Kehse, U. 2010. Sustainable Mobility. Intelligent Traffic Management. Siemens, *Pictures of the Future, Magazine for Research and Innovations*, Fall 2010.

[9] Raya, M., P. Papadimitratos, V. Gligor, and J. Hubaux, 2008. On data-centric trust establishment in ephemeral ad hoc networks. In: *Proc. of IEEE Infocom*, Phoenix, Arizona, pp. 1238-1246.

[10] Hutchins, J., A.T. Ihler, P. Smyth, Probabilistic Analysis of a Large-Scale Urban Traffic Sensor Data Set, KDD Workshop on Knowledge Discovery from Sensor Data Las Vegas, Nevada, USA, pp. 94-114, 2008.

[11] Malkin, G. 1998. RFC 2453, RIP Version 2. *Request for Comments 2453*, The Internet Society, November 1998.