ABSTRACT
The European Union recently proposed to increase Europe’s radio spectrum for technology that allows cars to communicate traffic information. By letting cars to communicate to each other, less cars will end up in a traffic jam. It is estimated that traffic jams will cost the economy around 80 billion euro by 2010. Currently, traffic news is collected by a central organ, which processes the data and broadcasts the information to navigation tools (e.g. HD traffic by TomTom), internet websites and news programs. This information flow causes the driver to receive information about a traffic jam with a delay of approximately half an hour. In this paper a solution is presented that is decentralised and which makes faster information dissemination possible. We introduce a vehicle-to-vehicle communication protocol (called SOTRIP) that works by letting cars on opposite lanes exchange information about the traffic situation on the road ahead for the receiving car. First experimental results confirm the better overall throughput, especially with heavy traffic.

Categories and Subject Descriptors
H4.3.3 [Information Systems]: Communications Applications.

General Terms

Keywords
Navigation tools, vehicle-to-vehicle communication, traffic information, communication technology, simulation.

1. INTRODUCTION
The European Union recently proposed to increase Europe’s radio spectrum for technology that allows cars to communicate traffic information. It is estimated that this will cost the economy around 80 billion euros by 2010.1 Besides the economic costs, also lives are at stake: in 2006 alone, more than 46,000 people died on Europe’s roads. Thus investment in intelligent vehicle-to-vehicle communication systems is necessary to decrease costs of traffic, both in economic terms as well as human lives.

The American counterparts of such measurements are brought together in the Intelligent transportation systems programme [1]: a collection of efforts to add information communications and technology to transport infrastructure and vehicles. The ITS effort includes technologies (e.g., wireless communication, computational technologies) as well as applications (e.g., electronic toll collection, automatic road enforcement). A promising technology delivering technology as well as application is mobile ad hoc networking, which comes from the area of networks and builds on the concepts of decentralisation and self-organisation [10] [2].

In this paper, we have taken an existing computer network communication protocol, called newscast [7], and adopted it for communication of traffic jam information in a road network. The original protocol is particularly designed to be used on very large-scale loosely coupled computer networks, is computationally very cheap (low local computational requirements) and facilitates fast propagation of information by means of gossiping. Gossip protocols enables nodes in a (typically peer-to-peer) network to communicate small pieces of information with each other according to some underlying (usually randomized) algorithm. It has been shown to deliver scalable and reliable propagation performance in distributed networks. The three properties (large-scale networks, computationally cheap and fast propagation) are interesting and appealing from an ITS point of view. Firstly, although the topologies of vehicle traffic networks and large-scale computer networks (e.g., internet) do not exactly correspond one-to-one, there are some shared properties (regular loss of connection, dynamic addition/deletion of nodes) that can be exploited in using the solution from one domain for solving a problem in another. Secondly, as we envision the nodes of the ad hoc network to be located within lightweight in-vehicle computational devices, the computational processing load should be low. Thirdly, the propagation of information should be done fast, it should only concern relevant and up-to-date information and be easily transmitted/received.

Besides these overlaps, there are some significant differences that we take into account, of which we mention the most relevant. Firstly, the geographical and physical nature of vehicle traffic requires thinking about communication in different ways than in computer networks: vehicles that pass each other on the road cannot be directly compared to computers connected by a high-bandwidth wire. Secondly, we take advantage of the fact that in the vehicle traffic domain the vehicles move in the opposite direction with respect to the information stream. In other words, the information that you are in a traffic jam now is most relevant to vehicles driving behind you (that may still consider an alternative route).
For this, there is no analogy in computer networks.

Concerning the application of gossip protocols for inter-vehicular communication, we are not the first ones. Most notably, Das et. al have presented the Spawn algorithm, which is a cooperative strategy for content delivery in vehicular networks based on the popular BitTorrent protocol [4]. However, the information provided comes from (infra-structural) gateways that are located on the roadside. In our work, we specifically aimed to let the information itself also be collected in a completely distributed way by the vehicles themselves. In other work by Wu et. al, a similar approach is taken in the development of the MDVV algorithm, a mobility-centric approach for data dissemination in vehicular networks [12]. An important difference between our work and theirs is that MDVV uses directed query-based communication (vehicles query for specific information from specific vehicles) while we use simple neighbourhood-based broadcasting: everyone can pick up the communicated information and use it, but is not obliged to.

To summarise, our research is thus aimed at providing a new protocol for vehicle-to-vehicle communication with respect to traffic jam information. The specific objectives of this research are: 1) to provide a new innovative protocol for optimizing traffic flow; 2) to show that the protocol provides a method for quick information dissemination; and 3) to show that the protocol is aimed to be better in terms of overall traffic throughput then a traffic situation without using the protocol.

The remainder of this paper is structured as follows. In Section 2 we give an overview of related vehicle-to-infrastructure and vehicle-to-vehicle traffic control. Section 3 describes the SOTRIP vehicle-to-vehicle protocol. Section 4 we demonstrate the effectiveness of the protocol by means of simulation. Finally, in Section 5 we conclude and provide pointers for future work.

2. BACKGROUND

From the well-known variants of vehicle-to-X (where X is vehicle, infrastructure, roadside or mobile) communication networks, we focus on, firstly, vehicle-to-infrastructure, where the taken measurements are along the road – for example, dynamics road signs and traffic lights; and, secondly, vehicle-to-vehicle, where vehicles communicate information to each other with traffic information and react to that information.

2.1 Vehicle-to-Infrastructure (V2I)

As the urban traffic jams form an increasing problem, there is a need for an innovative system that can optimize the signal plans of traffic signals. An exemplar approach for solving this problem is by using a “green wave”, where adjacent junctions with traffic lights are synchronised. This results in a pack of cars that can drive on in one direction without having to stop at the junctions. This solution works fine for controlling rush hour traffic, but it fails when confronted with an unexpected or incidental increase of traffic (e.g., additional traffic when people are leaving a stadium) – we need methods that can adapt and are not as strict as the “green wave” method.

De Oliveira et. al have looked at swarm intelligence for traffic lights control [5]. Here, traffic lights act as social insects. According to the number of cars waiting and passing the traffic lights, the ‘insects’ change traffic signal plans. In another work, Bazzan suggests the use of evolutionary game theory techniques [2] for traffic signal control: expectations of agents concerning intentions converge to a pattern by feedback, enabling a stable equilibrium.

In other work, Gershenson [6] has proposed a method where agents act as traffic lights. By combining the number of cars waiting and the time that they are waiting, a preference is given for a specific direction. The method enables convoys of cars to emerge which encounters only green lights even before stopping at a junction, i.e., a dynamic “green wave”.

2.2 Vehicle-to-Vehicle (V2V)

The Trafficopeter system [8] is a distributed traffic data collection system, where vehicles function as agents and can broadcast messages to other agents. These messages contain the vehicle’s position, speed and direction. The information received by other vehicles is stored on a grid, where the vehicle itself is placed at the centre of the grid. So-called centroid messages are created by combining messages received by others that are driving in the same direction as the vehicle itself. When querying for information up ahead, a car propagates it request to cars in front of it. This continues till the query is received by a car that is on the location on which the query is about. Then the information is propagated backwards, to the car that requested the information.

The SOTIS system [11] is a traffic information system that distributes traffic data. Each vehicle that is equipped with the system sends information (position, road, time, speed) to all other surrounding vehicles (equipped with the system). After receiving different messages, the system checks the messages for their time stamp and updates its knowledge base. Based on the average velocity of all vehicles on a specific road segment, each vehicle calculates the current traffic condition. Using specific time schedules, different messages are sent to surrounding cars. Cars preferably send traffic news to cars driving in the other direction, so the news will be spread more efficient.

2.3 Summary

We observe some potentially problematic issues with the presented systems with respect to our specific objectives. In Trafficopeter, information is only propagated to vehicles driving in the same direction, thus the time before a car receives an answer to their query can be quite long. In SOTIS, there are three issues. Firstly, if there are many vehicles on the road, this could lead to overloading because all vehicles continuously broadcast messages. Secondly, much data has to be stored, also about roads where a receiving vehicle is not going to go. Finally, messages are not deleted when outdated, and the system only decreases detail when a specific road segment is further away.

We kept these issues in mind while developing SOTRIP: 1) vehicles on opposite lanes communicate with each other, 2) information is only sent when considered useful for others, and 3) information ‘evaporates’ over time (old information is deleted) and space (other roads).

3. THE SOTRIP PROTOCOL

In this Section, we describe the SOTRIP protocol. For using SOTRIP, a vehicle must have a GPS navigation device (e.g., a ‘traditional’ TomTom or Garmin), a processing unit (e.g., for updating messages) and a communication device (for short-range communication).

Table 1 shows the algorithm that each vehicle continuously executes. We describe the protocol in three simple (chronological) steps.

Step 1 – When a SOTRIP vehicle drives on the road, it continuously checks the number of vehicles within a certain perimeter. This number represents the (local) traffic load. If the number of same-direction vehicles exceeds a threshold, the system sends a message to the opposite-direction vehicles within the perimeter. The direc-

\[We provide more detail (and pseudocode) of SOTRIP in [9].\]
Check whether CURRENT NEWS needs to be updated
2. Check whether JAMS AHEAD needs to be updated
3. Handle incoming messages
4. Check whether the car is in a traffic jam, if so, report the jam
5. If the car has traffic news, it sends it to oncoming cars

Step 1 – The vehicle starts sending this information.
Step 2 – Vehicles continuously receive messages. If a vehicle receives a message that he already received before, he updates this information by refreshing the time stamp and length of the jam. The length of the jam is increased with the speed of the vehicle carrying the information for every time step: the length of the jam is thus known when the vehicle does not receive any more messages. At that moment, it starts to increment the distance attribute of the message by adding the current speed per time step (which is used for approximating the location of the jam). After some (fixed) time, the vehicle starts sending this information.

Step 3 – Vehicles that have stored information about a traffic jam on their own lane will slightly change this information each time step: the distance attribute is decreased by the speed of the vehicle (thus the distance to the jam is updated while getting closer to it). This information is needed later for distinguishing different messages about traffic jams. Receiving vehicles check whether the message is useful in terms of their route and already received information. By comparing the message distance attributes of the stored and received messages, a vehicle determines whether this information is about the same jam or another one. When it is considered a new jam, this information will be added to the list of stored jams; if not, the stored information will be updated with the newly received information (i.e., length, time and distance).

4. EXPERIMENTS

We conducted a series of simulation experiments to test SOTRIP. In this Section, we present a description of the case study, the experimental design and setup, results and analysis.

4.1 Description

We created a simulation in the MASON multi-agent simulation environment. In here, many different scenarios can thus be tested by adding different road layouts. The particular scenario used for the experiments, consists of one highway, a ramp to get on the highway and an exit to leave the highway. A traffic jam is created by a truck that has been turned over, which is also removed again after some (fixed) time.

4.2 Hypotheses

The effectiveness of SOTRIP will be checked by investigating four hypotheses, concerning traffic throughput, inflow and protocol parameters. These hypotheses are based on the stated objectives in the introduction.

1. Hypothesis 1: The total number of cars have reached a destination faster then when not using SOTRIP.
2. Hypothesis 2: If the inflow of traffic is increased, the relative overall trip time is decreased.
3. Hypothesis 3: If the threshold for reporting a jam is decreased, the number of informed cars before the exit increases.
4. Hypothesis 4: If the activity range is increased, the number of informed cars before the exit increases.

4.3 Design and Setup

The experimental design is defined of 3 dimensions: the input-interval $\in \{10, 25, 50, 100, 200\}$ represents the traffic density, and is implemented by the number of timesteps a car is spawned, e.g., each 10 or 25 timesteps; the broadcast threshold $\in \{5, 10, 15\}$ is the number of observed vehicles after each other to classify as a traffic jam; and, finally, the activity range $\in \{15, 25, 35\}$ represents, firstly, the size of the beforementioned perimeter, and, secondly, the maximum distance that a vehicle is able to receive messages from another vehicle. In addition to these 3 dimensions, we conducted another set of experiments (for benchmarking) in which SOTRIP was not used.

The measured variables are: the number of informed cars – the number of cars that have received a message about a traffic jam up ahead (either before or after the exit); and the number of cars that have reached their destination – the cars that drove from the left side of the screen to the right side; and the total trip time is the time that each car has spent to drive from its spawn point to one of the two destinations.

We conduct 10 simulation runs for each experimental setting; a run continues until all cars spawned at the left side, have reached a destination on the right side.

4.4 Results

The overall results for the average trip time can be found in Figure[1] The horizontal labels represent the different experimental settings, the first number is the input interval, the second the broadcast threshold and the last is the activity range. Two experiments have been included to give more detail, these can be found in Figure[2] and in Figure[3]

4.5 Analysis

In this Section, we briefly describe the verification of the before-mentioned hypotheses and report on some additional observations.

4.5.1 Verification

We conducted statistical significance tests (two-sample $t$-tests) in order to verify the four hypotheses. For reasons of space, we left out the statistical calculations themselves (but which can be found in[4]) and report on the conclusions with respect to the hypothesis verification, drawn on the basis of the statistical tests.

Hypothesis 1 – We compared the no-system run with all other runs, for 13 out of 16 comparisons the hypothesis holds: the SOTRIP system outperforms the no-system. In two of the other three cases, the fact that the no-system performs better can be attributed to the ‘disastrous’ interaction of the two protocol parameters (threshold and activity range, which are in both cases 15). In these cases, the number of cars needed to be classified as a jam, does not “fit into” the activity range. Thus the 15 cars threshold is never met because the activity range is too small to ‘see’ 15 cars.

Hypothesis 2 – For the input intervals 10 and 25, the trip time significantly decreased. For larger input intervals (less traffic), we did not see significant improvements in the trip time.

Hypothesis 3 – If the input interval is 10, in most cases, more cars are informed before the exit then after the exit if the threshold parameter is decreased. The exceptional cases are the same as mentioned with hypothesis 1 (both protocol parameters set to 15). If the input interval is 25, we see no significant difference between
Figure 1: Overall trip time

Figure 2: Numbers of cars over time, where input = 10, threshold = 15 and activity range = 25.

Figure 3: Numbers of cars over time, where input = 25, threshold = 5 and activity range = 25.

informed cars before and after the exit. We conclude that for heavy traffic situations (input interval 10), the increase of the threshold has a positive impact on the number of informed cars; but for situations with less traffic (input interval ≥ 25) we see the opposite.

Hypothesis 4 – Overall, we observe no specific relationship between the activity range parameter and the number of informed cars before the exit. However, for heavier traffic situation, we think (but this is not statistical significant) that an increase in the range has negative effect on the number of informed cars. For situations with less traffic, an increase in the activity range has a more or less positive impact on the number of informed cars.

4.5.2 Further observations

In addition to the hypothesis verification, we report on some additional observations on the basis of obtained results relating to the protocol parameters.

Firstly, although the two protocol parameters (threshold and activity range) significantly impact the effectiveness of SOTRIP, we cannot conclude on a setting of both parameters that is best in all situations. The performance of the protocol seems to be more affected by the external factors (i.e., input interval) than the protocol parameters. We can still say that in heavy traffic, an increase in the threshold has a positive effect and vice versa. Conversely, in heavy traffic, an increase in the activity range has a negative impact; a positive influence can be seen with less heavy traffic.

Secondly, a potential problem in our experimental design is that the two protocol parameters are not completely independent (while design-wise we treat them as such). The reason for this dependence is that, technically, both are accomplished by the same thing: the range of the communication device. This makes it difficult to calculate the effect for one of the two variables on the increase or decrease in informed cars. In order to untie the correlation, some inner changes should be made to SOTRIP and we plan to do so in further developments of the protocol.
5. CONCLUSIONS

We presented a new vehicle-to-vehicle communication protocol, called SOTRIP, with which we aim for improving road traffic flow. The protocol is based on the principles of gossip enabling fast, cheap and up-to-date dissemination of information. The protocol consists of three simple steps: 1) when a vehicle is in a traffic jam, it sends a message to a second car driving in the opposite direction; 2) the receiving vehicle stores the message until it has passed the traffic jam, after which it starts propagating the message to vehicles driving in the other direction; and 3) the receiving cars have information about the situation in front of them and can react to that information.

We conducted a series of simulation experiments, demonstrating that with SOTRIP, the overall travel time of vehicles is decreased. Additional experiments were performed to test the impact of the two main protocol parameters on the information dissemination, but no general conclusion could be drawn by looking at these parameters; different situations require different settings for the best dissemination.

With respect to the objectives of this study (as formulated in the introduction), we can say the following. Regarding the first objective, we have shown by means of a small literature study that there exist techniques with the same aim as SOTRIP but these methods have some disadvantages that we try to solve by developing SOTRIP. For the second objective, we have shown that SOTRIP can disseminate information in a distributed fashion and that the dissemination is affected by the traffic situation and some protocol settings. However, we can not draw definite conclusions about the speed of dissemination. Regarding the third objective, we have shown that SOTRIP enables faster traffic throughput.

In summary, the contributions of this study are threefold. Firstly, we have delivered a novel, gossip-based vehicle-to-vehicle communication protocol. Secondly, the experiments have resulted in both simulation software for programming traffic scenarios and data of one particular traffic scenario. Finally, the experimental design of this study can be used for further experimentation for multiple and more complex scenarios.

For future work, we consider a number of different directions this research can be heading. First of all, regarding the protocol itself, we have to make some parts adaptive. One of the main conclusions that we draw is that there is no single optimal setting of the protocol parameters that works best for all traffic situations. Thus it must be researched how to let vehicles estimate the traffic situation (for example, how heavy the traffic is) and adapt the protocol parameters according to these measurements. Another adaption to the protocol is to enhance the algorithm for broadcasting messages when the car is in a traffic jam. This could decrease the waste of bandwidth with even further in comparison with existing protocols. Secondly, with respect to further experimentation, we plan to test multiple and more complex traffic scenarios. We also plan to perform experiments where the propagation of information is also done on the congested lane (backwards). Thirdly, also regarding experiments, we want to vary the ratio of equipped cars and measure the influence on the traffic flow. For now, we assumed that all vehicles are equipped with SOTRIP but it is more realistic to only assume a portion of the vehicles has SOTRIP. Finally, we want to further test the reactions of vehicles to the received information as an integral part of a navigation device (e.g., TomTom or Garmin) that many cars have on board these days.

6. REFERENCES


