PeerTIS – A Peer-to-Peer Traffic Information System

Jedrzej Rybicki Björn Scheuermann Markus Koegel Martin Mauve
Computer Networks Group
Heinrich Heine University
Düsseldorf, Germany
{rybicki, scheuermann, koegel, mauve }@cs.uni-duesseldorf.de

ABSTRACT

In this paper we propose a traffic information system based on the distribution of knowledge provided by the cars themselves. Prior work in this area attempted to realize this distribution via vehicular ad-hoc networks, i. e., by direct communication between cars. Such an approach faces serious problems due to capacity constraints, high data dissemination latencies, and limited initial deployment of the required technology. In this paper, we present a solution that is not based on ad-hoc networking, but is still fully decentralized. It establishes a peer-to-peer overlay over the Internet, using cellular Internet access. We present a structure for the overlay, a prototype implementation in a simulation environment, and results that underline the feasibility of such a system in a city scenario. We also provide an estimate of expected user benefits when our system is used for dynamic route guidance.

Categories and Subject Descriptors

C.2.1 [**Computer System Organization**]: Computer-Communication Networks—*network architecture and design, distributed networks*; H.4.3 [**Information Systems**]: [Communications Applications]; C.2.4 [**Computer Systems Organization**]: Computer-Communication Networks—*distributed systems, distributed applications*

General Terms

Algorithms, Design

Keywords

Peer-to-Peer Networks, Overlay Networks, Traffic Information Systems, Car-to-car Communication, VANET Alternatives, PeerTIS

VANET'09, September 25, 2009, Beijing, China.

1. INTRODUCTION

The vision of communicating cars has gained a lot of attention over the recent years. So far, the main effort was aimed at the development of Vehicular Ad-Hoc Networks (VANETs), where cars communicate in multihop fashion using WiFi-like links. VANETs support applications that aim to increase road safety, like emergency brake warning [31] or intersection assistance [4], as well as non-safety applications. A particularly popular representative of the latter class are *Traffic Information Systems* (TIS) [16, 30]. The main goal of these applications is to provide each car with information on the current traffic situation along its planned route. A navigation system may use this data to determine better routes. Route calculations are then no longer based on static map data only, but can be adapted to the current dynamics of road traffic or unexpected circumstances like accidents.

When such systems are realized using VANETs, they will necessarily suffer from the inherent limitations of wireless ad-hoc networks: limited network capacity [8, 23], resulting in scalability issues, and limited connectivity [12], which impacts the speed of information propagation in a VANET. Moreover, it will take significant time until the necessary market penetration of VANET technology is achieved. In order to avoid these limitations, we look at an *alternative* way of realizing such a system in this paper. Instead of using VANET technology, we will rely on infrastructure-based cellular communication, like for example UMTS. We assume that each car has Internet connectivity over such a network available (i. e., we assume that some IP-based communication channel is present between the cars). This is a reasonable hypothesis considering the already very broad availability of mobile cellular Internet access today—and continuously decreasing prices. In any case, we may expect mobile cellular Internet access to be common and inexpensive much sooner than we may expect VANET technology to be common in vehicles.

A traffic information system can be seen as a distributed database of traffic measurements, accessible for all interested cars. Each car contributes its own measurements of the road conditions by inserting them into the database. The measurements may then be used by other participants for route planning. The central benefits of using a VANET as a basis for this database result from the fact that the system is entirely user-driven and decentralized. In particular, there is no central operator that may limit access to the information, the service is free of charge and there exists no single point of failure. These benefits, however, come along with the abovementioned drawbacks regarding capacity and connectivity.

Because of these drawbacks of VANETs, we are concerned with the question how such a system can be built based on cellular communication—avoiding the problems of a VANET-based solution, but at the same time maintaining its key benefits. This can be achieved by employing a peer-to-peer Internet overlay over infrastructure-based networks. Peer-to-peer overlay networks offer robust and efficient search and retrieval of data along with redundant storage and massive scalability [14]. Therefore, a peer-to-peer overlay over cellular networks constitutes a point in the design space that offers very interesting opportunities: it exhibits the robustness and decentrality properties of a VANET-based system, but it avoids the scalability and connectivity issues of VANETs.

We put the idea of using a peer-to-peer overlay for traffic information systems forward in [22]. However, there, the general concept has only been outlined as a position statement. That paper did neither describe a complete system, nor an implementation and its evaluation. Here, we go a number of steps further. We discuss a specific peer-to-peer overlay structure tailored to the specific requirements of a traffic information system. We call our system *PeerTIS*. We provide performance measurements of PeerTIS in a large-scale simulation setup of a whole city. We assess the bandwidth cost imposed on the cellular links, and estimate the achievable user benefits.

The remainder of this paper is structured as follows. In Section 2, we review related work on traffic information systems. We then take a closer look at the application in Section 3, before we introduce and discuss the proposed overlay structure in Section 4. The evaluation of the proposed solution with regard to bandwidth usage as well as an estimation of the travel time improvements when using dynamic route guidance can be found in Section 5. We conclude our work in Section 6.

2. RELATED WORK

2.1 Dynamic Traffic Information Systems

Traffic information systems have so far primarily been discussed in the context of vehicular ad-hoc networks. In [30], Wischhof et al. present the idea of information dissemination based on periodic local broadcasts and the store-and-forward principle. Each system participant maintains a local *knowledge base* consisting of his own measurements and data received from other users. The content of the knowledge base is used to prepare beacon packets. To cope with the limited bandwidth of ad-hoc wireless networks, the authors proposed to merge the individual observations regarding the same fixed, pre-defined distant region into one value. This reduction of details on further-away regions reduces the bandwidth consumption, while at the same time the most important information (i. e., descriptions of the immediate vicinity), are still available at a high level of detail.

The idea of aggregation has also been followed by the authors of [16]. In this work, the information on individual vehicles is extensively processed (that is, compressed or filtered) in order to reduce the size of transmissions. A different approach was presented in [5], where the authors went beyond distributing raw data and instead used data clustering techniques to identify the most relevant information items. In [13], a solution for VANET-based traffic information systems is presented where information is aggregated and disseminated. In this system, the impact of road side units and their placement is assessed. Such a heterogeneous approach—VANETs supported by additional infrastructure—should provide better service quality especially during the roll-out phase of VANETs, when only a small fraction of cars is equipped with radio devices. However, it requires the deployment of additional, VANET-specific infrastructure. Our cellular communication-based peer-to-peer network, in contrast, uses existing communication networks, which, at least in principle, allows for a deployment *right now*. Furthermore, cellular networks allow to avoid store-and-forward communication and the resulting delays that are inherent to all VANET-based approaches, and therefore have the potential to offer better service to the user.

Another way of providing description of current traffic conditions is followed in TMC [1] or Navteq Traffic RDS [18]. The data, collected mainly by road-side sensors, are broadcast by a central instance on a special RDS radio channel. The main drawbacks of this solution are the limited bandwidth of the radio channel, the cost of the road-side sensor infrastructure, and the need for central processing of the collected data. These drawbacks result in delays and a limited level of detail of the provided information.

In TomTom's HD Traffic approach [27], information about the location and the movement speed of mobile phones is used to infer the traffic situation on main artery roads. The information is collected and processed by a central instance. The results are then brought back to the users via UMTS. The measurements are indirect: speed and position are obtained by triangulation. The UMTS operator only knows the set of base stations that currently "see" a particular mobile phone, the exact position of the device is then estimated on this basis. The operator also does not know if the mobile phone is currently located in a car, on a bus, or carried by a pedestrian. All this limits the accuracy and level of detail of the information that can be inferred. What is more, like in TMC, sophisticated central processing and the need to wait for a significant number of samples before such an indirect inference approach will be successful and reliable leads to delays in information propagation. A peer-to-peer overlay with direct participation of the navigation devices will have immediate access to much more reliable data sources.

A slightly different approach is implemented in Dash Express [3], an Internet-connected navigation system. Each user anonymously sends her position and speed to the central server of the system operator. The server processes the data and sends it back to all users in the area. Like in the previously discussed systems, the central processing limits the scalability of the system.

Finally, there might also be a less technical reason why peer-to-peer sharing of cooperatively gathered traffic information is more appropriate than a server-based solution: the latter implies that a single authority may determine who may use the data. Given that this data is collected by the users themselves and given that access to this data is important to minimize the waste of resources such as fuel consumption, this does not seem to be appropriate. We believe that traffic data should be freely accessible by anyone wishing to participate in a distributed traffic information system.

An extensive comparison of the different approaches to the realization of the scalable traffic information system is provided in [11].

2.2 Peer-to-Peer Networks

The peer-to-peer communication paradigm aims for direct collaboration between network nodes, where all parties are equal in rights and duties. Peer-to-peer networks organize the system in a decentralized, distributed manner. Peers are connected to each other to form a communication overlay. Each peer shares some of its resources (bandwidth, storage, CPU power) and makes it available to the others. Peer-to-peer networks became popular because of their scalability and fault tolerance [14]. These are features highly desirable for systems directly affecting people's lives—like traffic information systems.

An interesting class of peer-to-peer networks are so-called *structured overlays*. In these systems, peers organize themselves in a predefined structure, like a ring [26], a butterfly graph [15], or a Cartesian space [20]. Content maintained in the network is then stored at specific locations in the overlay, determined by the data items. Typically, *(key, value)-pairs* are stored in so-called *distributed hash tables (DHTs)*: given the key of a data element, the distributed structure can be used to efficiently search for it in the overlay network. Essentially, this is also the functionality that is required in our system: efficient, accessible, scalable, distributed storage of traffic-related information. However, as it will soon become clear, simply applying existing DHT algorithms is not well-suited to the problem of collecting and distributing traffic information.

2.3 Security and Privacy

An important challenge in both peer-to-peer traffic information systems and VANET-based approaches is the security and privacy of users and data. A huge body of work exists on identifying possible risks [6, 24], estimating their feasibility [25], and proposing countermeasures [2] in peer-to-peer networks. VANET security issues have also received a lot of attention [7, 9, 21]. We acknowledge the importance of this work, and believe that these efforts are largely orthogonal to our work.

3. THE TIS APPLICATION

3.1 Framework and Use Case

A cooperative TIS essentially is a set of shared traffic-related information, along with mechanisms to access, use, and update it. In this paper we focus on the case where the shared information consists of travel times along road segments. We assume that all participants are equipped with a GPS receiver, a digital map, and mobile Internet access. All this can be provided by state-of-the-art cell phones, but other platforms are equally conceivable. By means of the GPS receiver, the cars are able to determine their position and speed. The digital maps are used as a basis for addressing road segments, so that each road segment can be assigned a unique ID that is known to every participant. The combination of GPS and map data allows each car to determine the ID of the road segment it is currently traveling on, and the IDs of the road segments it intends to pass along its future route. An observation of the current travel time consequently consist of a timestamp, a road segment ID, and the measured travel time.

Usually, a car's navigation system will fetch the relevant data at the beginning of the journey. This includes measurements describing the current situation along the possible routes connecting the origin and the intended destination. On the basis of this data, the navigation system will be able to choose a good route with respect to the expected travel time and potentially other aspects determined by the driver's preferences. Since the traffic conditions will change dynamically, further periodic queries will take place during the trip. This allows to update the information on the currently planned route as well as on alternate paths to the destination. Thus, it is possible to adapt the chosen route on the fly. Of course, each car will not only consume information, but it will also contribute own measurements by publishing them in the traffic information system. Furthermore, we assume that very old information is no longer relevant for the drivers and will therefore be removed after some time.

3.2 Tackling a TIS with P2P Techniques

At a first glance, the problem is amazingly simple: the data stored in the traffic information system possess natural addressing identifiers: street segment IDs. Therefore, we can use the segment IDs as keys and the available measurements on the segment as values. We can thus build the traffic information system based on a DHT. So, *in principle*, we could use any of the plethora of existing DHTs, and simply apply it to TIS data. However, we can do much better if we keep the specifics of the application in mind.

It actually turns out that the implementation of a traffic information system using a peer-to-peer overlay over cellular Internet access is far from straightforward, in several regards. First, mobile devices and cellular Internet access themselves pose challenges. Despite the rash development of infrastructure-based cellular networks, the offered service (in terms of bandwidth and latency) is still inferior in comparison to, for instance, domestic Internet access. Moreover, typical on-board devices are usually equipped with less storage and weaker CPUs. In order to deal with the limited resources of the mobile stations, the reduction of the communication burden in the overlay is necessarily a primary design goal. So far, this goal is obviously common to most peer-to-peer overlays. However, here, the considered application itself provides very interesting opportunities to achieve them—opportunities that are not present in the context of distributed hash tables in general.

This is because, beyond the challenges of the technological platform, a quite unique feature of a TIS is the very specific usage pattern. As noted above, a navigation system will typically fetch all the data relevant for the planned and alternative routes, and will refresh the necessary information periodically. It will also contribute own measurements whenever a road segment has been passed. First, we observe that the lookups at the beginning and upon refresh operations have a very bursty structure: in most cases, many road segments will be looked up at once. Furthermore—and this is the key point—there is an inherent structure in the accessed data elements: they are not random access operations to independent keys, as they are typically assumed (implicitly or explicitly) when DHTs are designed and analyzed. Instead, the usage pattern has interesting *locality properties*: Since information about long, contiguous routes through the road network is needed, a car will issue requests about many segments that are geographically close together. Moreover, if a car contributes own observations for a road segment, it will typically have requested information about this road segment at an earlier time—it will usually only pass road segments that lie on its planned route. We show in our work how these interrelations between the user's actions can be used to improve the performance of our system.

4. A P2P OVERLAY FOR TRAFFIC INFORMATION SYSTEMS

In this section, we explain the details of our realization of a peer-to-peer overlay based traffic information system, called PeerTIS. The main question to be answered in this context is: what kind of overlay structure is able to provide efficient means for storing and retrieving highly dynamic traffic information in a mobile environment, taking the previously discussed locality of requests and updates into account? As it turns out, the locality property shifts some trade-offs in a direction that is very different from what is commonly discussed in the context of peer-to-peer overlays.

We will start this section with a naive approach: we simply use an existing, well-known DHT, the Content Addressable Network (CAN) by Ratnasamy et al. [20]. We will then, step by step, point out how is is possible to substantially improve on that, until, finally, we will arrive at the full PeerTIS design.

4.1 Naive Approach

The Content Addressable Network, a well-known structured peer-to-peer overlay, was one of the very first DHTs. It offers the usual DHT interface: data in form of (key,value)-pairs may be stored in the system, and can later on be retrieved given the key. The keys stored in CAN are mapped to points in a d-dimensional space, the *key space*. This mapping is usually performed by a predefined hash function. The key space is divided into zones, one zone per peer, so that each point in the key space is assigned to one zone and therefore also to a peer. Each peer is responsible for storing the information assigned to all keys that fall into its zone. When a new peer joins, the zone of one existing peer is split in half, the new peer takes over one half, including the corresponding (key,value)-pairs. When peers leave, zones are merged and respective data are handed over. Further mechanisms exist to react in case of peer failures and other adversarial events.

The communication overlay is formed by connecting the peers that share a common edge between their zones in the key space. Each node maintains a small routing table containing the IP addresses and ports of peers responsible for the neighboring zones. Store and lookup operations can then be routed through the overlay by greedily forwarding them in the "direction" of their destination in the key space: a peer that performs a lookup checks in his routing table which neighbor offers the most progress in the key space towards the hashed coordinates and passes the look-up message to that neighbor. This peer is either responsible for the zone containing the sought-after key, or it forwards the request to one of its own neighbors offering further progress towards the destination in the key space. This greedy forwarding continues until the message eventually reaches the peer responsible for the zone containing the point. For a much more in-depth description of the CAN overlay, we refer the reader to [20].

In a naive application of CAN to a cooperative traffic information system, one might proceed as follows. In order to access the data for route planning, a participating car will first identify the IDs of the relevant road segments. It will then use the hash function to determine the position of these IDs in the key space, and locate the peers responsible for these segments—one after the other. Note that due to the hashing of the keys, the positions of distinct road segment IDs in the ID space are entirely independent.

The process of accessing the data is depicted in Figure 1. This figure shows an example road and a (two-dimensional) CAN key space, that is subdivided into 18 zones (corresponding to 18 cars currently participating in the system). The dashed arrows show how the segments of a single road are mapped into the structure. The long solid arrow indicates that the car shown in the figure is responsible for the top-left zone in the CAN key space. Note that this has nothing to do with the physical position of the car: the responsibility zone is assigned randomly when the car joins the overlay (e. g., when the navigation device is switched on), and may later be adjusted through splitting and merging as other peers join and leave. It does, however, not relate to the position or movement of the car itself. If the car intends to look up the currently available information about the three road segments ahead of it, it needs to generate three separate lookup requests, which will follow the CAN overlay paths as indicated by the solid black arrows.

4.2 Improving the Lookup Performance

In many peer-to-peer applications (e. g., file-sharing systems), queries are largely independent from each other. As pointed out above, in a traffic information system, in contrast, they are *not* independent. The key reason is the inherent structure of the stored data, resulting from the road segment neighborhood relation in the road network. In the naive approach outlined above, however, this structure is lost due to hashing. We therefore propose to keep the adjacent segments of the streets "close" to each other in the peer-to-peer structure. In other words: we aim to preserve the topology of the street network within the overlay network.

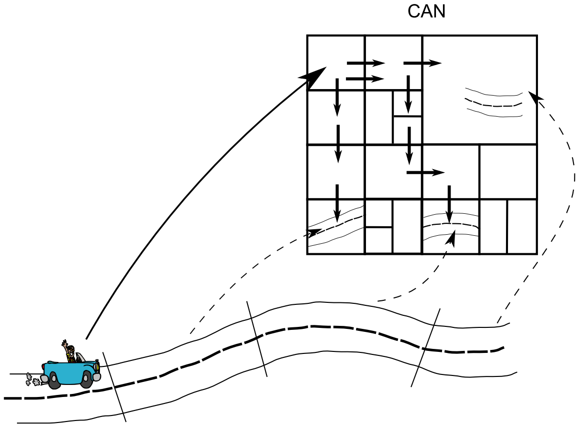


Figure 1: Naive application of CAN to a TIS. Dashed arrows show how the road segments are mapped into the structure. Bold solid arrows indicate the hops in the overlay necessary, for the peer in the left top zone, to fetch the data.

This aim can be achieved by leaving out hashing altogether. We may leave the general structure of the CAN overlay as it is, but instead of hashing road segment IDs to d-dimensional coordinates in an artificial coordinate space, we use a two-dimensional space, and use the *geographical coordinates* of each road segment in the real world as key coordinates. In such a system, the peers are no longer responsible for zones in a virtual multi-dimensional key space, but for geographical regions and the traffic measurements that fall within this region.

By doing so, we may reasonably expect to reduce the total effort that is necessary when a burst of lookups for all road segments along one or more alternative routes occurs: the looked-up road segments are contiguous in the road network, neighboring segments are geographically close together. Consequently, if their location in the CAN space corresponds to the position in the real world, they will be managed by peers that are close together in the overlay structure. Therefore, much more efficient lookups become possible. To retrieve the information on a whole route, it is then sufficient to locate the peer responsible for the *one* segment of this route, and then to simply follow the route in the peer-to-peer overlay. One single "multi-hop lookup", forwarded along the requested route in the overlay, can collect all the required data.

Consequently, the look-up messages differ from the ones used by the original CAN in an important aspect: they contain not one single key to be looked up, but the whole sequence of road segments for which information is requested. Because the peers in CAN are connected to their direct neighbors, the process of "following" can take place without further look-ups by simply handing the request over from peer to peer. Each peers forwarding a request will simply provide the information it is responsible for, before passing the request further on to the next peer. This is a substantial benefit over the situation in the naive approach, where individual and independent lookups to many independent positions scattered all through the overlay were necessary.

In Figure 2, we show what would happen in the same example as in Figure 1 above. The desired information can now easily be accessed using *one* multi-hop lookup.

4.3 Load Distribution

The substitution of hash coordinates with geographical ones allows, as noted above, to save significant effort when peers issue

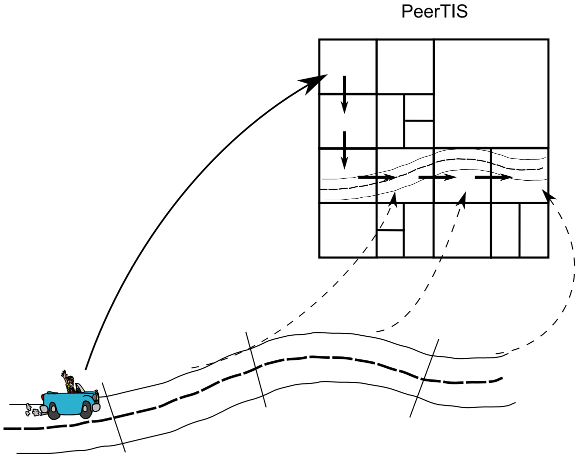


Figure 2: Preservation of the road network structure in the PeerTIS overlay.

bursts of lookups for consecutive groups of road segments. However, it comes at a costs of its own: in contrast to de-facto random hash coordinates, the geographical coordinates of road segments are not uniformly distributed over the key space. This has implications for the load distribution and fairness in the overlay. For instance, peers responsible for a zone in the city center, with many roads and lots of traffic, will experience a higher load than peers that manage a rural area.

We propose to tackle this problem by adjusting the zone sizes. In the original CAN, a joining peer would pick a random position and locate the peer currently responsible for this position. This zone would then be split, the new peer taking over one half of it. Because larger zones are more likely to be "hit" than smaller ones by a newly joining peer, this results in relatively homogeneous zone sizes. This is fine when keys are uniformly distributed over the key space. In our case, we must accept a less uniform key distribution, because we want to preserve the locality relations between road segments in the overlay. Therefore, we apply a modification that leads to smaller zone sizes in areas with more data. Referring to the example above, the subdivision into zones should be much finer in the city center than in the rural area. As a consequence, the workload imbalance is reduced.

This can be achieved by taking into account the physical position of cars at the time when they join the network. Instead of joining at a random position in the key space, peers join in the area containing their *own current location*. It is easily possible with PeerTIS because it natively supports the geographical addressing, meaning that the zone containing the geographical positions can be found as explained earlier. Consequently, there will be more peers joining in areas with higher density of vehicles and data, and less in the "unpopular" ones.[1]

A very pleasant side effect of these *geographical joins* is a further improvement in the look-up performance. In most cases, directly after joining the network, the new peer will fetch all the data needed to make its own routing decision. Since the planned route will start at the current position of car, and the new peer joined in the overlay at just this position, the number of hops needed for the first look-up will be very small.

4.4 Exploiting Temporal Correlations

Besides the *topological* dependencies between the queries that we have used above to improve the performance of bursts of lookups for geographically connected road segments, there are also *temporal* correlations between the users' actions. In particular: subsequent requests and updates by the same peer are very likely to refer to the same geographic area, or even to the same road segments. When, while driving, the information on the planned route is refreshed, a very similar set of road segments will be looked up. Furthermore, when a car produces own traffic measurements, this will typically happen for segments on its route—i. e., for segments which it has looked up previously.

This peculiarity opens interesting avenues for further improvements. We suggest that each peer should maintain, apart from the standard routing table, a cache of contact data to peers that are responsible for the road segments on its own planned route. This cache can be used to improve both the periodical update requests and the upload of own measurements. It is quite likely that still the same peers will be responsible for the data. Using the information from the cache, they can be contacted without any new lookup traffic.

Due to the limited resources of the mobile devices, it is highly desirable to avoid excessive node state. However, the cached information comes at a very small cost: it is a side-effect of the previous look-ups, and, apart from a few kilobytes of memory for storing the cache, it does not require any additional effort for maintenance. Of course, not actively maintaining the cached information means that it may be outdated when it is used later on.

But even if the cached information is not perfectly up to date, it may be helpful: even if the peer previously responsible for a given segment meanwhile accepted a join request and handed over the data to a new peer, it will still be very "close" to the desired information. It will therefore be able to pass the request on to the correct peer with very little effort. So, once again, we take advantage of preserving the inherent structure of the data in the overlay.

If a cached entry happens to be completely invalid—for instance, when the peer has meanwhile left the network—there is still the option of doing a regular lookup in the overlay; so, in this case, nothing is lost.

5. EVALUATION

In the previous section, we have shown how PeerTIS is an adaptation of CAN to the very specific structure of traffic data. We will now evaluate this approach, using a prototype implementation of the system in a simulation framework. This includes an estimation of the bandwidth usage, as well as the expected latency and the quality of the provided information. Finally, we assess the potential user benefit in a typical usage scenario.

5.1 Simulation Setup

An important aspect of simulation studies of systems that are related to vehicular road traffic is realistic car movement. For the simulations in this paper, we used the VISSIM [19] microscopic traffic simulator to generate car movements. In a scenario modeled after a medium-sized real city, with more than 500 km of roads, VISSIM simulated up to 10 000 vehicles.

Existing simulation frameworks like [10, 28], coupling network and traffic simulators, are tailored for VANET-style communication, not for cellular Internet access as we need it here. Recall that one of our design assumptions was to use existing and *already*

---

[1] It should be noted here that the original hash-based CAN, just like any other DHT, also suffers from unbalanced load: the uniform distribution of keys does not necessary guarantee a uniform distribution of workload, since some keys (so-called "hot spots") are more popular than others.

working IP-based cellular communication technology as our basis. Furthermore, we are primarily interested in application-layer metrics like the number and size of messages or the travel time savings. For these reasons, modeling the lower layers in abundant detail does not seem appropriate, and is not worth the significant effort in terms of simulation time. Hence, for our purposes, we decided against building a full-blown integrated simulation solution (or heavily adapting one to support cellular communication).

We developed a simulator that models an IP communication channel, including models for network congestion and packet loss. The traffic information system was implemented as an application on top of this. The two central aims were to keep simulation times reasonable despite the very high number of nodes, and to keep the implementation of the peer-to-peer system itself as generic as possible, so that it can be re-used in future field tests of our system. The penetration ratio for the evaluation was set to 20 %, unless noted otherwise. To underline the scalability of our solution we also conducted simulations with different penetration ratios.

When cars enter the simulation, they join the peer-to-peer overlay and request information on their planned route. To join the network the participant has to contact a peer which is already in the network. For that purpose for instance the original CAN has an associated DNS domain name which is resolved into IP address of one or more CAN bootstrap peers (which maintains a partial list of CAN peers) [20]. In our setup we emulate the presence of such a bootstrap DNS server. Each active peer sends periodic updates to the server. The server keeps a short list of some active peers. A joining peer can request that list and contact one of the peers. Note that such a server is only one possible solution, and that it does not imply a central authority: the server is not involved in the peer-to-peer network operation itself, and many bootstrap servers (or even multiple different bootstrapping mechanisms) can be used independently in parallel.

After joining the network, peers upload their own observations into the overlay each time they passed a street segment. Furthermore, each car periodically refreshes its information on the segments on its intended traveling route by issuing a new request every 30 seconds.

At the beginning of the simulation, all initially present cars will join the overlay almost at the same time. Such a situation will obviously not occur in the real world, where cars join and leave continuously. We therefore removed the first 200 seconds of simulation time from our evaluations, giving the system time to reach a stable state. Stable state does not mean that the set of participating peers is constant, but rather that the overall number of peers stays more or less the same: new peers join the network and others leave the network as soon as they reach the planned destination. The overhead caused by these changes in the structure of the overlay is of course included in the evaluation.

5.2 Feasibility

In this first part of the evaluation, we concentrate on the feasibility of the proposed system in terms of bandwidth and latency demands. Figure 3 shows the average bandwidth usage. For each simulation second, we sum up the sizes of all transmitted messages. This includes the maintenance overhead for the peer-to-peer structure as well as traffic caused by the insertion of new data and queries. The figure shows the average value for all active users along with 95 % confidence intervals. The values are very small (note that the y-axis is bytes/s, not kB/s). They easily fit the theoretical bounds of the UMTS networks (384 kB/s for upload). Older cellular systems, like EDGE (123 kB/s) or GSM/GPRS (42 kB/s), would likewise be able to handle such traffic amounts. The values obtained in a field tests [29] suggest that a throughput of ca. 40 kB/s between mobile nodes is feasible with UMTS. We compare the naive application of the original CAN overlay as described in Sec. 4.1 to our modified overlay, PeerTIS. As expected due to the reduced number of look-ups, PeerTIS results in a substantially lower bandwidth consumption.

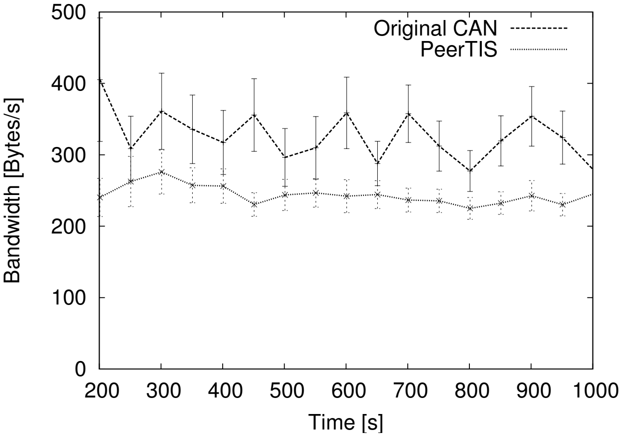


Figure 3: Average peer load with 20 % penetration ratio.

Table 1: Average peer load at different penetration ratios.

Penetration ratio	Avg. bandwidth usage [Bytes/s/peer]
10 %	244
20 %	245
30 %	248
40 %	251

The values measured in the initiation period (not presented on the plot) are about twice as high as in the stable state. Meaning that even if such artificial conditions (when the whole network is built in about 100 s) the bandwidth usage should not be critically high.

To assess the scalability of the system, we tested it in simulations with increasing penetration ratios. As can be seen in Table 1, the impact of varying penetration ratios on the average bandwidth usage per peer is negligible. This is a result of the self-scalability of the peer-to-peer system: each new peer increases the total load in the system on the one hand, but it compensates this by also contributing resources to the system on the other hand.

Although the average values of bandwidth usage are much smaller than the bandwidth offered by typical mobile Internet access, it might still happen that the network is locally overloaded by many peers concentrated in one cell of the cellular network. Thus, we determined another set of values in order to look at the load per UMTS cell. However, neither information on the positions of UMTS base stations nor on the shapes of the respective cells were available to us, therefore we use an approximation. We divided the scenario into grid of square cells, sized 500×500 m each, and measured the total amount of traffic transferred in each cell during an interval of five seconds length in the middle of the simulation. The snapshot of the network load is visualized in Figure 4, where the load of each cell in the simulation area is shown. The obtained values for the cells with the highest load added up to about 25 kB over five seconds, suggesting that the local network overload is not a problem. One may expect an even better situation in the real-world, if the load is divided between multiple service providers—our system

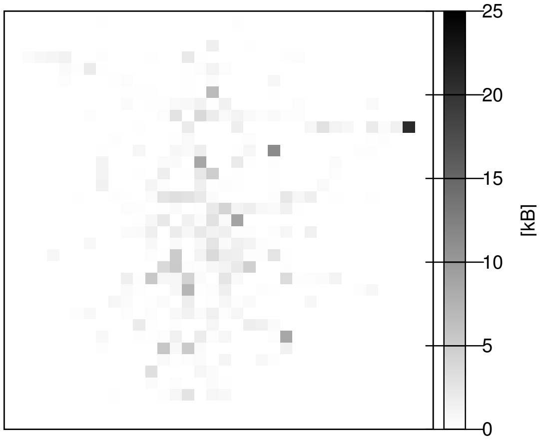


Figure 4: Per-cell network usage in 5 s time slots.

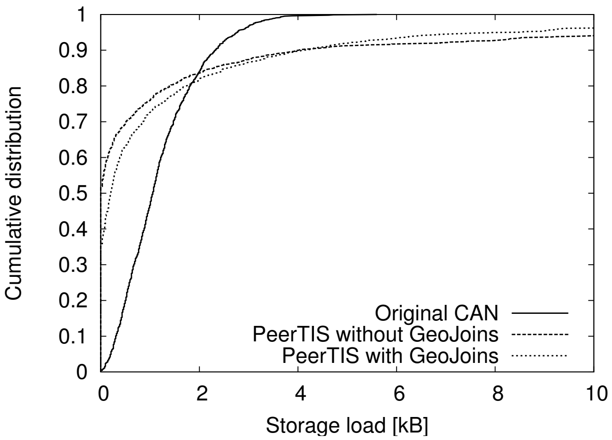


Figure 5: Load distribution (CDF).

does at no point assume that all navigation systems communicate using the same service provider.

The distribution of the load reflects the topology of the city we used for the simulation. The most loaded cell (in the north-east of the city, near the upper right corner of the figure) is a simulation artifact: in this cell, a freeway in the simulated city scenario crosses the boundary of the simulation area. Many new cars are generated at this point by the traffic simulator, which will then join the overlay. Many other cars leave the simulation area here; when they leave the overlay, they hand over their data to other nodes. Both these processes cause comparatively high artificial load.

The peer-to-peer overlay without hashing is susceptible to an unbalanced load distribution. Figure 5 shows the cumulative distribution of storage load on the peers. From the figure it becomes clear that 50 % of the participants do not store any data. The countermeasure proposed in Section 4.3 (geographical joins) brings some improvement here.

An unbalance can also be observed in the bandwidth usage of single peers. We identified the busiest peers by adding the sizes of all packets sent and received over the whole time span of a car's participation in the network. For 20 % penetration ratio (see Figure 6(a)) the busiest peer was in the unlucky position of being responsible for a busy junction in the city center. The bandwidth consumption history of the busiest peer in the 40 % penetration ratio scenario is depicted in Figure 6(b). That peer joined in a popular area and had to take over a lot of data at the very beginning of its network presence. Its area was then split few times, causing significant peaks in its bandwidth usage history. Recall that each such split (or merge) includes a hand-over of the data stored in the affected zone. Although the values are higher than the average presented above—up to 10 kB/s—they are still perfectly feasible with a GPRS or UMTS mobile Internet connection. Hence, we may conclude that despite the imperfectly balanced load distribution, no peer has to spend an excessive amount of resources.

5.3 Performance

The previously presented results show that, from a technical point of view, the proposed system is feasible. From a practical perspective, however, it is important to see if it can provide a quality of service that matches the demands of the end users. In the context of this question, we analyzed the average response time of the system. Since the responses for the queries are exchanged directly between the peers, the most important factor influencing the response time is the number of hops in the overlay. Figure 7 shows the cumulative distribution function of the hop counts.

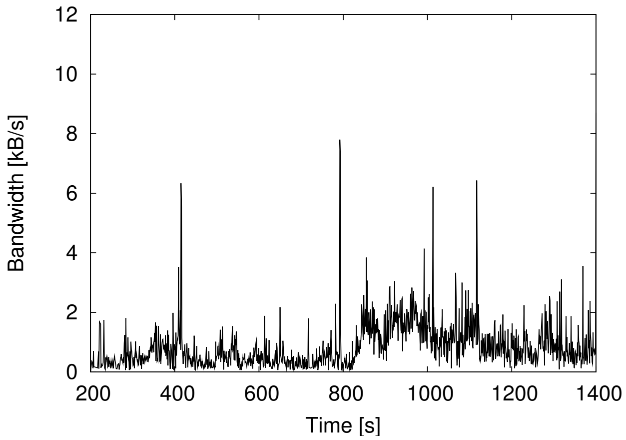
It turns out that the proposed adjustments of the peer-to-peer structure result in a situation where 90 % of all lookups required at most two hops in the overlay to reach their destination. It shows that the mechanisms proposed for achieving shorter lookup paths in the overlay—preserving the structure of the road network and caching contact data to responsible peers—play very well together. It is worth noting that caching contact data alone (that is, in an otherwise unmodified CAN overlay) already provides substantial improvement. This supports our assumptions about the correlations of user actions. Nevertheless, only in combination with the non-hashed, structure-preserving PeerTIS overlay it shows its full potential: the situation is again significantly improved.

The number of hops in the overlay can easily be translated into latency times by multiplying the number of hops by the measured real UMTS latency. In real world experiments, a latency of about 450 ms has been measured for two mobile nodes using an UMTS network [29] (with no HSDPA or HSUPA).

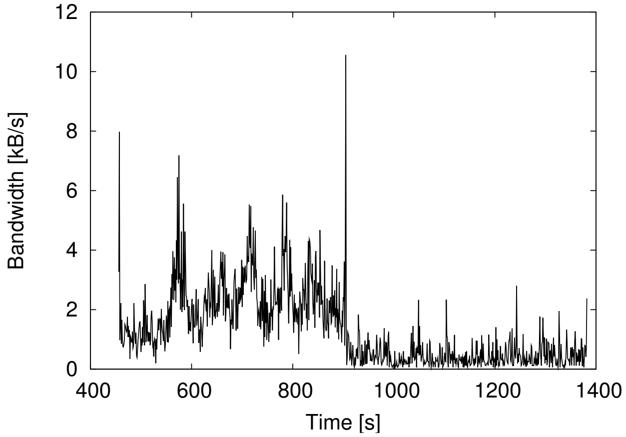
5.4 Potential User Benefits

In this section we assess the potential profits that a traffic information system can offer to the driver. Conventional, non-dynamic road navigation devices identify a route based on the travel times in the static map data. Typically, however, there will be deviations from these travel times, caused, e. g., by traffic jams. Such delays are often hard to predict, and are not considered in the routing process of such static devices. A networked TIS that participates in our peer-to-peer system can, in contrast, make use of the dynamic information provided by other participants. In order to see how much this could help the drivers to arrive more quickly at their destination, we conducted the following tests. Methodically they follow the lines of [13], where the user benefit in a VANET-based system has been evaluated in a very similar way.

We used an open source car navigation system, Navit [17], as a basis. We modified it to take the data available via PeerTIS into account for routing decisions. For the evaluation, we generated 700 start/destination pairs at random. For each pair we calculate three routes. The first route was the one provided by the unmodified version of Navit, based on static navigation data. For the calculation of the second route, we used the information available in the Peer-



(a) 20 % penetration ratio.



(b) 40 % penetration ratio.

Figure 6: Bandwidth usage of the busiest peers.

TIS system; i. e., it uses the information that is provided by other cars participating in the system. Consequently, this information will become more accurate with increasing penetration ratio. The third route is a globally optimal one: it is obtained by using perfect, global knowledge of the current traffic situation. For each of these routes, we compute the total travel time along this route based on the current traffic situation. We use this time as a means to quantify how good the routing decision is. Comparing the first and second route allows us to draw conclusions on the benefit of PeerTIS over static navigation systems. This third route allows us to estimate an upper bound of what could *generally* be possible with any arbitrary mechanism to distribute traffic information data. Comparing the second and third route therefore shows how close to this theoretical optimum we get with our system.

For each start/destination sample, we divide the travel times of the static (first) and the PeerTIS (second) routes by the travel time along the optimal (third) route. This *relative travel time* quantifies how much longer than optimal the chosen route is. For instance, a relative travel time of 1.2 for the static road data route means that, when following this route, a driver would be underway 20 % longer than if he had perfect knowledge and chose the currently optimal way to his destination. Admittedly, this methodology is not perfectly realistic, since we use static car traces and implicitly assume that better knowledge of the current situation will result in better routes regardless of the future development. However, an evaluation that takes into account all the dynamics and interrelations is significantly beyond the scope of this paper—our primary aim is to give an indication of the achievable benefits.

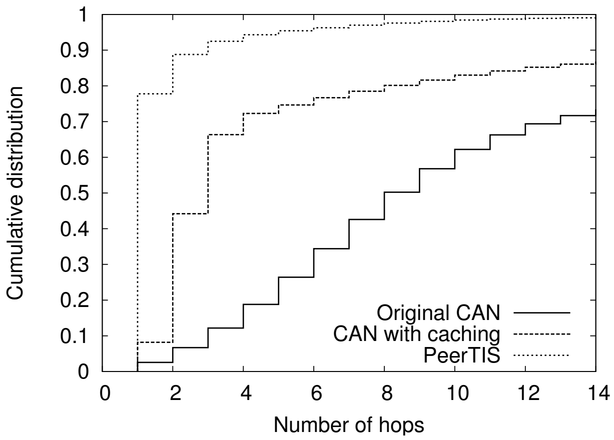


Figure 7: Look-up hop count (CDF).

In Figure 8, the results of our evaluation are depicted. We need not distinguish between PeerTIS, unmodified CAN, or other overlay structures. They differ in the way the data is organized in the overlay, but not in terms of the information available to the driver. Hence, from the perspective of the quality of the resulting routing decisions, they are all equal.

For obvious reasons, the optimal route always has a relative travel time of 1. A static navigation system will choose the optimal route in about 50 % of all cases, but may often make quite sub-optimal decisions. In about 10 % of all cases, the route is more than 20 % slower than the optimal route.

For PeerTIS, two lines are shown, for two different penetration ratios, when 5 % and 20 % of the vehicles participate in the system, respectively. In both cases, it results in substantially better routing decisions than the static navigation system, which becomes clear from the significantly lower relative travel times. Even though the routes are not always optimal with PeerTIS, there are far less extremely bad cases. The small difference between the quality of the predictions made with 20 % and 5 % participating cars suggest that PeerTIS is able to provide a high quality of service already very early on.

6. CONCLUSIONS

In this paper we presented PeerTIS, a dynamic, cooperative traffic information system using an Internet-based peer-to-peer overlay. Existing cellular Internet access is used as its foundation. This is combined with a peer-to-peer overlay tailored to the specific properties of the application: spatial and temporal correlations between users' actions, topological dependencies among the data that is stored in the system, and awareness of the current user position.

We have performed simulations on the basis of a prototype implementation. The results underline the feasibility of such a system. They also show that a large reduction of required bandwidth at the peers and an even more significant reduction of the necessary number of hops per lookup can be achieved by tailoring the overlay structure to the specific properties of the application. Finally, we have assessed the user benefit that can be expected from a peer-to-peer traffic information system, in terms of the travel time savings.

In our future work we will focus on further improvements of the load distribution in the overlay. We believe that this can be achieved

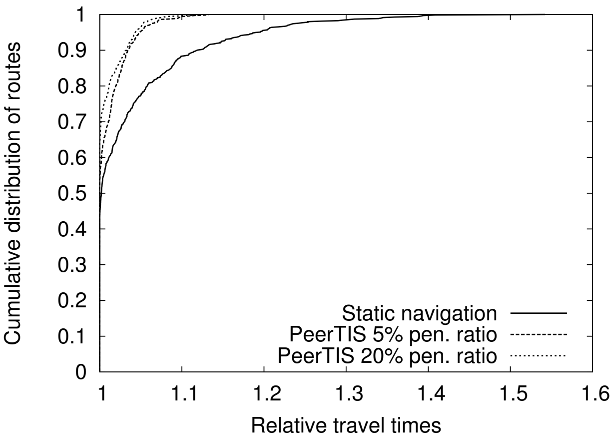


Figure 8: Relative travel times: static road navigation and PeerTIS in relation to optimal routing choices with perfect global knowledge.

by dynamic resource management: peers that currently maintain empty zones should help out by adding redundancy to more heavily loaded areas. Furthermore, we plan to assess the user benefits in more detail. For this purpose, a tighter feedback loop between the peer-to-peer overlay simulator and the vehicular movement in the traffic simulator is necessary. We will build upon existing solutions for VANETs, like [10, 28]. Finally, we intend to proceed further towards testing our implementation in the real-world.

Acknowledgements

This work was supported and made possible through a grant by the Deutsche Forschungsgemeinschaft (DFG). We thank Daniel Cagara for his technical help in the simulation study. We would also like to thank the anonymous reviewers for very detailed and constructive comments on our work.

7. REFERENCES

[1] Traffic message channel TMCProFM. http://www.t-traffic.de/.

[2] M. Castro, P. Druschel, A. Ganesh, A. Rowstron, and D. S. Wallach. Secure routing for structured peer-to-peer overlay networks. In *OSDI '02: Proceedings of the 5th Symposium on Operating systems design and implementation*, pages 299–314, New York, NY, USA, 2002. ACM.

[3] Dash Express. , http://www.dash.net/.

[4] F. Doetzer, F. Kohlmayer, T. Kosch, and M. Strassberger. Secure communication for intersection assistance. In *WIT '05: Proceedings of the 2nd International Workshop on Intelligent Transportation*, May 2005.

[5] S. Dornbush and A. Joshi. StreetSmart Traffic: Discovering and disseminating automobile congestion using VANETs. In *VTC '07-Spring: Proceedings of the 65th IEEE Vehicular Technology Conference*, Apr. 2007.

[6] J. R. Douceur. The sybil attack. In *IPTPS '02: Proceedings of 1st International Workshop on Peer-to-Peer Systems*, pages 251–260, Mar. 2002.

[7] P. Golle, D. Greene, and J. Staddon. Detecting and correcting malicious data in VANETs. In *VANET '04: Proceedings of the 1st ACM International Workshop on Vehicular Ad Hoc Networks*, pages 29–37, Oct. 2004.

[8] P. Gupta and P. R. Kumar. The Capacity of Wireless Networks. *IEEE Transactions on Information Theory*, IT-46(2):388–404, Mar. 2000.

[9] J.-P. Hubaux, S. Capkun, and J. Luo. The security and privacy of smart vehicles. *IEEE Security and Privacy*, 2(3):49–55, 2004.

[10] C. Lochert, M. Caliskan, B. Scheuermann, A. Barthels, A. Cervantes, and M. Mauve. Multiple Simulator Interlinking Environment for Inter Vehicle Communication. In *VANET '05: Proceedings of the 2nd ACM International Workshop on Vehicular Ad Hoc Networks*, pages 87–88, Sept. 2005.

[11] C. Lochert, J. Rybicki, B. Scheuermann, and M. Mauve. Scalable data dissemination for inter-vehicle-communication: Aggregation versus peer-to-peer. *Oldenbourg it – Information Technology*, 50(4):237–242, Aug. 2008.

[12] C. Lochert, B. Scheuermann, M. Caliskan, and M. Mauve. The feasibility of information dissemination in vehicular ad-hoc networks. In *WONS '07: Proceedings of the 4th Annual Conference on Wireless On-demand Network Systems and Services*, pages 92–99, Jan. 2007.

[13] C. Lochert, B. Scheuermann, C. Wewetzer, A. Luebke, and M. Mauve. Data Aggregation and Roadside Unit Placement for a VANET Traffic Information System. In *VANET 2008: Proceedings of the Fifth ACM International Workshop on VehiculAr Inter-NETworking*, Sept. 2008.

[14] E. K. Lua, J. Crowcroft, M. Pias, R. Sharma, and S. Lim. A survey and comparison of peer-to-peer overlay network schemes. *IEEE Communications Surveys and Tutorials*, 7(2):72–93, Mar. 2005.

[15] D. Malkhi, M. Naor, and D. Ratajczak. Viceroy: a scalable and dynamic emulation of the butterfly. In *PODC '02: Proceedings of the 21st annual symposium on Principles of distributed computing*, pages 183–192, New York, NY, USA, 2002. ACM.

[16] T. Nadeem, S. Dashtinezhad, C. Liao, and L. Iftode. TrafficView: A scalable traffic monitoring system. In *MDM '04: Proceedings of the 5th IEEE International Conference on Mobile Data Management*, pages 13–26, Jan. 2004.

[17] Navit. http://www.navit–project.org/.

[18] NAVTEQ Traffic. , http://navteq.com/about/advanced_traffic.html.

[19] PTV AG. The traffic simulator VISSIM. http://www.english.ptv.de/cgi-bin/traffic/traf_vissim.pl.

[20] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker. A scalable content-addressable network. In *SIGCOMM '01: Proceedings of the 2001 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, pages 161–172, Aug. 2001.

[21] M. Raya, A. Aziz, and J.-P. Hubaux. Efficient secure aggregation in VANETs. In *VANET '06: Proceedings of the 3rd ACM International Workshop on Vehicular Ad Hoc Networks*, pages 67–75, Sept. 2006.

[22] J. Rybicki, B. Scheuermann, W. Kiess, C. Lochert, P. Fallahi, and M. Mauve. Challenge: Peers on wheels – a road to new traffic information systems. In *MobiCom '07: Proceedings of the 13th Annual ACM International Conference on Mobile Computing and Networking*, pages 215–221, Sept. 2007.

[23] B. Scheuermann, C. Lochert, J. Rybicki, and M. Mauve. A Fundamental Scalabiliy Criterion for Data Aggregation in

VANETs. In *MobiCom 2009: Proceedings of the Fifteenth ACM SIGMOBILE International Conference on Mobile Computing and Networking*, Sept. 2009.

[24] E. Sit and R. Morris. Security considerations for peer-to-peer distributed hash tables. In *IPTPS '02: Proceedings of 1st International Workshop on Peer-to-Peer Systems*, pages 261–269, Mar. 2002.

[25] M. Srivatsa and L. Liu. Vulnerabilities and security threats in structured overlay networks: A quantitative analysis. In *ACSAC '04: Proceedings of the 20th Annual Computer Security Applications Conference*, pages 252–261, Washington, DC, USA, 2004. IEEE Computer Society.

[26] I. Stoica, R. Morris, D. Liben-Nowell, D. R. Karger, M. F. Kaashoek, F. Dabek, and H. Balakrishnan. Chord: a scalable peer-to-peer lookup protocol for Internet applications. *IEEE/ACM Transactions on Networking*, 11(1):17–32, 2003.

[27] TomTom. HD Traffic. http://www.tomtom.com/hdtraffic.

[28] A. Wegener, M. Piórkowski, M. Raya, H. Hellbrück, S. Fischer, and J.-P. Hubaux. TraCI: an interface for coupling road traffic and network simulators. In *CNS '08: Proceedings of the 11th Communications and Networking Simulation Symposium*, pages 155–163, Apr. 2008.

[29] C. Wewetzer, M. Caliskan, K. Meier, and A. Luebke. Experimental Evaluation of UMTS and Wireless LAN for Inter-Vehicle Communication. In *ITST '07: Proceedings of the 7th International Conference on ITS Telecommunications*, pages 287–292, June 2007.

[30] L. Wischhof, A. Ebner, H. Rohling, M. Lott, and R. Halfmann. SOTIS – a self-organizing traffic information system. In *VTC '03-Spring: Proceedings of the 57th IEEE Vehicular Technology Conference*, pages 2442–2446, Apr. 2003.

[31] Q. Xu, T. Mak, J. Ko, and R. Sengupta. Vehicle-to-vehicle safety messaging in DSRC. In *VANET '04: Proceedings of the 1st ACM International Workshop on Vehicular Ad Hoc Networks*, pages 19–28, Oct. 2004.