Caching Dynamic Information in Vehicular Ad Hoc Networks

Nicholas Loulloudes, George Pallis and Marios D. Dikaiakos

Department of Computer Science, University of Cyprus, Nicosia, CY1678, Cyprus
{loulloudes.n, gpallis, mdd}@cs.ucy.ac.cy

Abstract. Recent advances in VANET technologies have propelled the development and deployment of a wide variety of Vehicular Information Systems (VIS) to vehicles. Such systems provide real-time, crucial information, to drivers, ranging from collision avoidance warnings to the availability of road-side facilities. Despite the invaluable information provided by such systems, the imposed network overhead, due to the necessity of maintaining a global view of the road/vehicular networks and the surrounding environment, cannot be dismissed. In this work we evaluate the performance of VIS when utilizing caching techniques as means of minimizing network overhead. We present an evaluation study of our approach by conducting extensive simulations on large-scale vehicular networks under different realistic urban traffic conditions. Our evaluation results identify the critical parameters that affect information quality in VANETs as well as demonstrate the viability and effectiveness of the cache-enabled VITP.

1 Introduction

Inter-vehicle communication has emerged as a promising field of research and development, where advances in wireless and mobile ad hoc networks, global positioning systems and sensor technologies can be applied to real-life problems and result to great market potential. The uptake of these technologies by automobile manufacturers, road-network operators and the vehicular-transportation industry is expected to result in Vehicular Ad-Hoc Network (VANET) infrastructures comprising vehicles that communicate between each other and with fixed road-side units. Due to the specific characteristics of vehicular mobility, VANETs are characterized by highly dynamic topologies, short-lived links, and frequent network disconnections [1]. In addition, Vehicular Information Systems (VIS) generate a substantially high number of messages in order to obtain and maintain a global or even partial view of the prevailing conditions in the vehicular environment. This overhead imposed on the VANET can lead to the saturation of the already limited network capacity, which in turn is bound to degrade the quality of VANET services (i.e: lower response times and loss of information quality) [2]. Therefore, the provision of efficient, robust, wide-area information services over vehicular ad hoc networks remains an open challenge. In this paper, we investigate the implications of caching on the efficiency of VANETs and on the quality/accuracy of VANET-based VIS. Our study is performed in the context of *VITP*, a proactive, location-oriented protocol designed for the retrieval of dynamic vehicular information over VANET [3]. In dense networks, proactive protocols like VITP can avoid saturation caused by flooding and diffusion since information is queried on demand rather than being pushed periodically during discrete time intervals. Caching may improve further the efficiency of VITP by reducing the time to serve information requests, minimizing overall network resource consumption, and improving information accessibility in the presence of mobility constraints that result to short-lived links, network disconnections, etc. However, caching may also lead to a significant degradation of VIS quality, given the highly-dynamic nature of vehicular information. In this work, we conduct an extensive exploration of the trade-off between the efficiency and the quality of VANET-based VIS services, in the presence of caching.

In contrast to earlier research efforts [4], which have shown that caching can be beneficial when vehicles are moving on unidirectional straight roads, in our work we examine the implications of caching in a number of different vehicular traffic scenarios that are more realistic and challenging. Building upon our prior work on VITP [3], we provide answers to the key question: *Does the co-existence of a proactive, location-aware, communication protocol and caching maintain acceptable levels of vehicular information quality while sustaining network performance?* Our main contributions are summarized as follows: a) We extend the VITP architecture [3] in order to support cache-based location aware services. VITP is a proactive, location-aware, application-layer, communication protocol designed to support a distributed service infrastructure over VANETs. b) Through an extensive simulation testbed, we identify the critical parameters that affect vehicular information quality in VANETs as well as demonstrate the viability and effectiveness of the cache-enabled VITP.

The rest of the article is organized as follows: Section 2 briefly surveys the relevant work. Section 3 describes the design concepts of VITP so as to support caching. Section 4 describes the simulation set-up, while simulation results are presented and discussed in Section 5. Finally, Section 6 concludes the paper and presents future research conditions.

2 Related Work

The design and development of data dissemination mechanisms for VANETs, which aim at increasing the ratio of solved queries with the minimum network overhead, while maintaining acceptable levels of information quality, has been the target of research investigations in the recent literature [4, 5]. Due to space limitation, in the following paragraphs we present the most indicative ones (a survey of recent work can be found in [6]).

Currently, carry-and-forward data dissemination schemes have been proposed, investigating the layout of road networks. An efficient such a protocol, called VADD, was proposed in [7]. VADD proposes the data forwarding using a stochastic model based on vehicular traffic statistics, such as traffic density and vehicles' speed, in order to achieve the lowest delivery delay from a moving vehicle to a static destination. More recently, a trajectory-based data forwarding scheme, called TBD, for light-traffic road networks has been proposed [8]. This scheme exploits both private trajectory information and traffic statistics.

Also, an adaptive query evaluation scheme based on the underlying road network was proposed in [5]. The proposed query scheme leverages the road-network connectivity graph to create an evaluation tree for traffic information queries. However, this method imposes a significant overhead in the network by broadcasting control messages to inform the VANET on any location changes of vehicles participating in a specific query evaluation. In large and dense VANETs this approach can lead to the saturation of the already limited network capacity, which in turn is bounded to degrade the quality of vehicular services [2].

Recently, the authors in [4] presented Hamlet, a fully-distributed caching scheme in which vehicles decide independently of each other whether to cache and for how long a piece of information. Decisions are made based on individual node's observation of the information present within its radio range. The objective of this approach is to increase content availability in the proximity of nodes and minimize network overhead. Although Hamlet demonstrates an increase in the query success rate and a decrease in the network overhead, the results are based on a confined static information data-set within a simple urban mobility scenario. However, it is important to study the effects of caching in urban environments with diverse topographical layouts using a larger information data-set, since the performance of caching in VANETs is mainly affected by the mobility scenarios.

The closest works to ours is [4]. Similarly to us, this work studies the effects of caching in VANETs. However, there are two fundamental differences. The authors do not consider a cache-based, proactive, communication protocol for the dissemination of vehicular information in their experiments, nor do they conduct experiments in dense mobility scenarios where information is generated by both mobile and stationary nodes and under the presence of unscheduled events like accidents. Our work is more general in the sense that Hamlet [4] can be used upon our cache-based VITP protocol.

3 Enabling Caching support in VITP

This section presents and defines the necessary extensions to the architecture and message syntax of the Vehicular Information Transport Protocol (VITP) in order to support caching of vehicular information.

We present an extension to the architecture of VITP so that caches can be directly accessed by VITP peers. VITP peers implement the VITP protocol and operate as clients, intermediaries, or servers in a VITP-protocol interaction. The reader can find more details concerning the main design concepts of VITP architecture in [3]. We choose to extend VITP since it allows the dissemination of query messages proactively, thus providing better control in the number of messages injected in the VANET.

According to [3], a VITP transaction consists of four phases: 1) *Dispatch-query phase*: a request Q is transported through the underlying VANET toward its target area L. Q goes through a number of intermediary VANET nodes, which push the message toward its destination using geographic routing. 2) *Virtual Ad Hoc Server (VAHS)-computation phase*: the VITP request is routed between the VITP peers of the VAHS. VAHS consists of the VITP peers that contribute to computation of the reply to a VITP query. 3) *Dispatch-reply phase*: the VITP reply is geographically routed toward source region. 4) *Reply-delivery phase*: broadcasts the VITP reply to the VANET nodes of source region, so that the reply can be received by the VITP peer that originated the transaction.

Directive	Value	Description
cacheable	Boolean	The reply can be cached.
expires	time-stamp	The time after the reply is considered expired (if cached).
private	Boolean	The cached reply can be reused only from the original peer that requested it.
public	Boolean	The cached reply can be reused from any peer.
validate-after	seconds	A peer must validate the cached reply with the target area after “n” seconds from the reply generation time.
p-validate-after	Boolean	A peer must validate the cached reply with the neighbour peers after “n” seconds since the reply generation time.
p-validate	Boolean	A peer must validate the cached reply with the neighbour peers before serving a request.
retransmit	Boolean	Serve request first from cache and retransmit it to target area also.

Table 1. Cache control directives used in a VITP reply message

In the cache-enabled architecture of VITP, the dispatch-query phase of VITP transaction is modified as follows: When a VITP peer receives a VITP query message, it checks whether this message can be served from its local cache. If there is a *cache hit*, the *last_accessed* attribute of the cached message is updated by the current time. In the context of VANETs, each object is characterized by the fields: message content and location-id. We consider a *cache hit* when a) there is a valid replica of the requested information in the cache and b) the location-ids of both requested information and its replica in the cache are in the same geographic location. Then, a VITP reply containing the cached message is generated and sent back to source node for the VITP transaction to be completed. If there is a *cache miss*, the query message is forwarded to other VITP peers with respect to the underlying geographic routing protocol. The dispatch-reply phase of the VITP transaction is also modified to support caching: when a VITP peer receives a reply message, it checks whether the reply is contained in its local cache. In case of a *hit*, the cache is updated to reflect the properties of the new message. In the case of a *miss*, the reply message is inserted into the cache and its parameters are populated accordingly. The cache system comprises a *Cache Replacement Module* (CRM), which detects stale information and evicts it from the cache. To this end, the CRM assigns a *Cache Utility Value* (CUV) to each cached message. The CRM can also be used in the unlikely case of a cache becoming full.

To support the cache operations described above, we extend the VITP message specification with a set of *cache-control* headers. These headers act as directives to VITP-peer caching decisions (Table 1). More details about the generic syntax of the VITP message along with examples can be found in [3].

4 Simulation Testbed Setup

For simulating the effects of vehicle movement, vehicular mobility traces were generated using the aid of TrafficModeller [9] and SUMO[1], a space-continuous microscopic

[1] SUMO - Simulation of Urban Mobility, http://sumo.sourceforge.net/

traffic simulator. For simulating the behaviour of VITP in a VANET, ns-2[2], was employed. To increase the level of realism and accuracy of results in the experimentation of this work, an extension to ns-2 was developed that enables the import of SUMO road networks in an appropriate data-structure prior the simulation run. This data-structure can be queried by vehicles at any instance of the simulation run-period in order to identify on which road/segment they are currently moving, thereby increasing accuracy in the resolution of location-aware queries.

4.1 Vehicular Mobility Generation

TrafficModeller was fed with two data sets, each one representing a region of a real city with different topographical layout. *Region 1* follows a Manhattan-like city layout where a big percentage of the road-network is comprised of long and parallel straight roads, where vehicles can accelerate to higher speeds between adjacent junctions. *Region 2* follows a more common urban layout where road segments are curved and much shorter, hence restricting vehicle acceleration. Both regions were extracted from real-world, accurate, city maps obtained from OpenStreetMap[3]. This was done in order to evaluate the efficiency of the cache-enabled VITP under real urban environments with different mobility patterns. All roads within these regions have a speed limit of 13.89 m/s. The number of junctions per squared kilometer with traffic lights is considerably much smaller in *Region 2* than in *Region 1*. It is quite important to evaluate the effectiveness of the cache-enabled VITP in road-networks with different distributions of traffic lights, since these have a direct influence in shaping the properties of vehicular traffic (i.e mean vehicle speed and density). Furthermore, a number of "hot-spots" within these regions were defined, in order to simulate the traffic conditions that arise when people drive from/to their workplaces, shopping malls, amusement centers etc. Vehicles is the simulation were set to have an acceleration of 4.5 m/s, deceleration of 2.6 m/s with their top speed bounded by the road speed limit. Driver imperfection factor, that is the ability of the driver to adapt to a desired speed, was set to 0.5 (where the value 1 indicates a perfect driver). Total simulation time was set to 1000s. These high-level abstractions were translated as low-level input specification and passed to SUMO. Vehicular traffic traces for 970 and 875 distinct vehicles that move in Region 1 and Region 2 respectively, were generated by SUMO. For both regions vehicles have a mean drive time of approximately 230s. The average vehicle speed over time for both regions is between 8 and 10 m/s. Vehicle average speeds have a large variance but follow a uniform distribution.

Mobility traces obtained from TrafficModeller and SUMO, were transformed to acceptable ns-2 input trace files using TraceExporter[4]. For the vehicular wireless network simulation, each vehicle is equipped with IEEE 802.11 capable communication hardware with a wireless radio coverage of 200m. In addition, the computing device of each vehicle is VITP enabled allowing it to participate in the resolution of incoming VITP requests as described in [3]. Furthermore, vehicle caches have an unlimited size.

---

[2] ns-2 - http://www.isi.edu/nsnam/ns/
[3] OpenStreetMap - http://www.openstreetmap.org
[4] TraceExporter - http://www.auto-nomos.de/

Simulation scenarios run for a period of 1000s. Vehicles are injected in the network simulation starting from $t = 1s$ to $t = 950s$. A 200s warm up period is allowed before retrieving information used in the evaluation phase. The purpose of the warm-up phase is to allow the caches to reach some level of stability. Therefore, any event prior to the above time instance is not evaluated.

4.2 Evaluation Scenarios and Query Generation

For the evaluation of the cache-based VITP for the exchange of different vehicular information under urban environments we have set up the following scenarios:

Scenario 1. Each vehicle is aware of the road-network topology through on-board digital maps and its current location through GPS. Since travel times are heavily influenced by the prevailing traffic conditions, vehicles would like to identify the road-paths towards their destination that if followed will result in reducing the travel-time. To discover the aforementioned conditions we introduce a query scheme implementing a "forward-scan-radar" traffic information system. According to this scheme, we assuming that a vehicle V turns into some *Road A* and wants to follow the fastest route to its destination D. Through its knowledge of the road network, V can calculate all possible road-paths connecting its current position to D. Upon entering *Road A* it issues *LookAhead (L)* queries, investigating dynamically the conditions along all these possible road paths. *LookAhead (L)* queries are propagated to a certain depth in the road-path and obtain the traffic conditions of the roads up to the specified depth.

Scenario 2.This scenario follows the paradigm of the first scenario. Here, vehicles issue *LookAhead* queries in the possible road-paths towards their destination, with the exception being that unscheduled events do take place (e.g., vehicle break-downs). Such events block roads in those paths and influence the normal traffic flow by causing following vehicles to slow-down or even stop. In case that vehicles stop there is a high probability of congestion build up which is further increased if the event has taken place on roads that are small in length and do not have alternative exits. In this scenario several vehicle break-downs throughout the network are simulated by vehicles that stop abruptly in the middle of the road for 100s and then resume their trip to their destination.

Scenario 3. We assume that vehicles would like to discover the availability of roadside facilities such as parking places, gas stations and restaurants. A stationary Road Side Unit (RSU) is responsible for each facility and at certain time intervals it broadcasts information concerning the facility (i.e free parking space availability). For our simulations, 5 RSUs were placed randomly on the road-network and broadcast information about their facility every 60 seconds. Vehicles are divided in equal groups with common facility interests and throughout the simulations randomly generate queries to identify the location of such RSUs and obtain the broadcasted information.

In all the above scenarios, VITP queries are issued with a default $ReturnCondition = 5$. A Return Condition as specified in [3] determines the sampling size of the requested information a query must obtain before a VITP reply can be generated and dispatched back to the originator of the request. For the first two scenarios, each vehicle issues queries with $LookAhead = 2$, to road paths from its current position to the its destination. The values for the above parameters were selected by sampling the parameter space having in mind that a high *LookAhead* value can cause the saturation

Parameter	Value
Simulation Time	1000s
Simulation Area size	$3200m * 6800m$ (Region 1) $1600m * 1400m$ (Region 2)
Number of VITP enabled Vehicles	970 (Region 1) 875 (Region 2)
Mean Vehicle Drive Time	230 s
Wireless Coverage Area	200m
VITP Query Return Condition	5
Cache *TTL* values	0s to 200s
Inter-Query Issue Time	On new road-segment entry
Look-Ahead Value	2
Number of RSUs	5
RSU Broadcast Time interval	60s
Vehicle Break-Down Duration	100s

Table 2. Simulation setup parameters

of wireless network bandwidth thereby causing a significant amount of query drops and on the other hand a high $ReturnCondition$ value increases vehicular information accuracy. This resulted in the generation of a total of 29263 queries for *Region 1* and 16557 queries for *Region 2*. For Scenario 3, 12231 and 7562 queries were generated for *Region 1* and *Region 2* respectively.

For all the scenarios the existence of an underlying greedy geographic routing layer that forwards VITP messages from the source node towards the destination area is assumed. After each issued VITP query is satisfied by a VAHS, the generated reply is again routed geographically towards the source node. Each reply is generated with *cache-control* header = *[cache-control: cacheable, expires=t, public]*. Consequently, the information contained in each VITP reply message is cached to all intermediary nodes on its way towards the source node. Furthermore, we consider that all nodes use a *TTL-based cache replacement policy*. According to this policy, messages are removed from the cache as soon as their *TTL* (Time to Live) value expires. A *TTL* value specifies the maximum time for which a cached copy should be considered valid. Finally, Table 2 presents an overview of the simulation setup parameters.

5 Evaluation

To describe the performance of cache-enabled VITP in inter-vehicular networks, we employ the following metrics which are considered to be the most indicative:

- **Query Recall:** *the number of replies received while issuing queries towards a specific location of interest, over the number of replies that should have been received from that location*.
- **Response Time:** *is the average Round Trip Time (RTT) of a successful VITP transaction*.
- **Information Accuracy:** *measures how close the received value describing some vehicular information is to the actual value at the location of interest*.

– **Number of Exchanged Messages:** *is defined as the total number of exchanged messages, including geographic routing messages and VITP query resolution messages throughout the whole simulation period.*

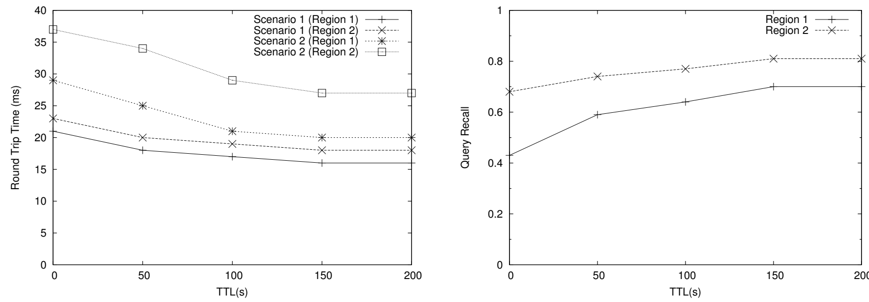
5.1 Caching Evaluation - Querying Road Traffic Conditions

For the three aforementioned scenarios we evaluate the performance of the cache-based VITP on both regions and under different *TTL* values assigned to traffic query replies. A $VITP_{TTL=0}$ emulates the original VITP where information caching is not supported on VITP peers. The maximum value of $VITP_{TTL=200}$ denotes that traffic information is cached on VITP peers for the whole drive time duration and this value was selected since it approximates the average run-time of vehicles in all the scenarios. We begin our evaluation by investigating the performance of the cache-based VITP for Scenario 1 and Scenario 2. We measure the response time of the *LookAhead* traffic queries by varying the *TTL* values of cached reply messages. According to *TTL*-based replacement policy, when the *TTL* of a message expires, this cached message is discarded by all vehicles' caches. The results are reported in Fig. 1(a) where the x-axis represents the *TTL* values in seconds and the y-axis represents the response time in milliseconds.

We observe that the average response time for Scenario 1 in both regions decreases with the increments in *TTL*. It is evident that longer *TTL*'s result to a better diffusion of information throughout the vehicles' caches and, consequently, to an increased probability that the requested information is found nearby the requesting vehicle. Queries exhibit the lowest response time when $VITP_{TTL=200}$, with an improvement of 24% for Region 1 and 22% for Region 2 in comparison to the original VITP (no caching). It is interesting to observe that the RTT remains constant when $VITP_{TTL\geq 150}$ for all the scenarios examined in this section. This occurs because vehicle queries present high temporal locality of reference. As we referred in section 4, all vehicles in the simulation issue VITP queries in the road-paths towards their destination area in order to discover the prevailing traffic conditions. Since it is not possible to change the mobility of vehicle at runtime and alter their initial route chosen by SUMO, the queries remain constant over the whole simulation time.

For Scenario 2, the average RTT for all *TTL* values in both regions is higher than in the case of Scenario 1. Unscheduled events like vehicle break-downs cause congestion which increases the geographic distance between querying vehicles and the query target location. Remember that when $VITP_{TTL=0}$ vehicles obtain the information only from the target location. Due to the use of geographic routing, queries have to traverse a greater number of hops in order to overcome the break-down and reach the target location. This increase in the number of hops consequently leads to an increase in query round trip time. When caching is enabled, the lowest response time is encountered for $VITP_{TTL=200}$ where there is an improvement of 31% for Region 1 and 27% for Region 2.

Fig. 1(b) denotes the traffic information query recall for Scenario 1 under both regions, where the x-axis represents the *TTL* values and the y-axis represents the query recall percentage. The general trend is that increasing the *TTL* value increases the query recall for both Region 1 and Region 2. The lowest recall for both regions is when $VITP_{TTL=0}$, where no information is cached in the network and a reply to a given



(a) Query Response Time (RTT) over TTL (b) Query Recall over TTL

Fig. 1. Response Time and Query Recall vs. TTL

traffic information query can be answered only from the source location. On the other hand, the best recall is achieved when $VITP_{TTL \geq 150}$ with 70% and 81% for Region 1 and Region 2 respectively. As the *TTL* value increases, the number of replicated information in the network increases and a query can be served not only at its target location but also from information stored in other vehicles cache. This allows for vehicles to obtain the desired information faster and in addition reduces the amount of queries that should be re-generated in order to obtain information for the target locations of previous unresolved queries. The above observation is also reflected by examining the number and geographic distribution of replicated objects in the network (Figure 2). Each subfigure illustrates a combined snap-shot of the road and wireless network. The x and y-axis denote each zone boundary in the road network while the z-axis (on the right) denotes the percentage of information completeness at each zone. Information completeness is calculated by taking into account all the distinct information replicas in a zone to the total distinct information in the VANET. It is worth noting here that replication, RTT and information accuracy are interrelated. Specifically, we observe that the pattern of dependence among them follows the rule that the increased replication results in reduced response times but also results in reduced information accuracy.

The trend of traffic information query accuracy for both scenarios is given on Fig. 3(a). The x-axis represents the *TTL* values, whereas, the y-axis represents the accuracy of the information that was retrieved through the VITP queries. For Scenario 1, the highest accuracy 86% and 85% for Region 1 and Region 2 respectively, is obtained while traffic information is retrieved directly from the query target location. Information accuracy drops as the value of *TTL* increases denoting that there is high probability that VITP

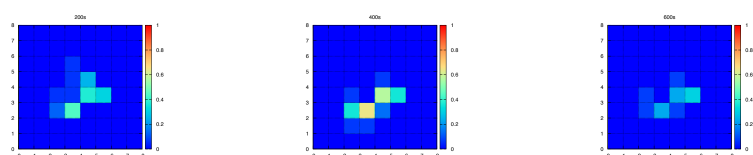
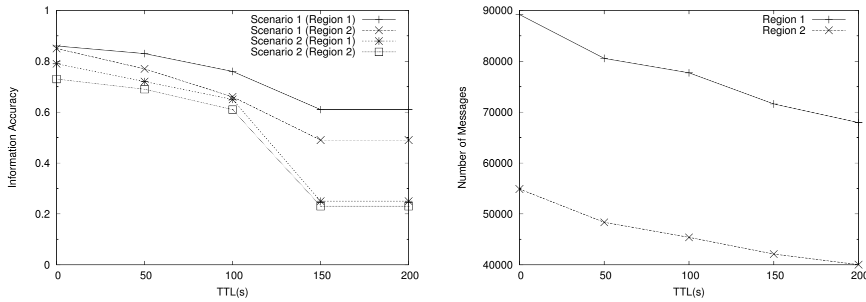


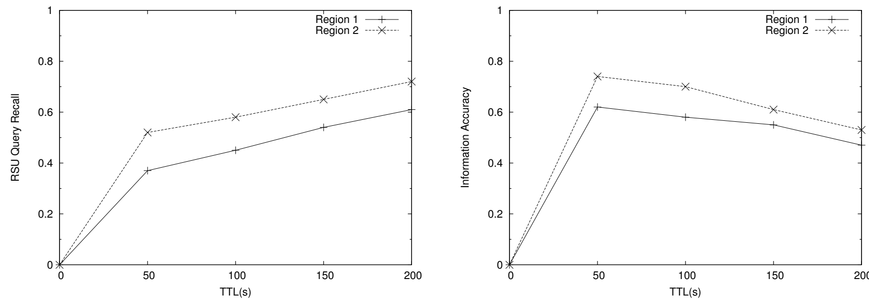
Fig. 2. Geographic distribution of Replicas in respect to simulation time



(a) Information Accuracy over TTL (b) Number of Messages over TTL

Fig. 3.

queries will be answered from other vehicle's cache instead of the information source location. The drop in information accuracy is expected since cached information does not accurately reflect the traffic conditions on the queried roads. Despite this, *the cache-based VITP manages to maintain a high level of information accuracy (83% and 77% for Region 1 and Region 2 respectively) when $VITP_{TTL=50}$, while at the same as seen from Fig. 3(b) can reduce the number of messages injected in the network by 11% and 12% respectively*. The noticeable difference in the rate of change of information accuracy between the two regions is due to the difference in the road-network layout. The road-network of Region 1 is comprised of several straight roads that enable vehicles to maintain a relative constant speed over a larger period of time. Therefore, in the absence of unexpected traffic events such as vehicle break-downs, cached information can maintain and report a relatively "better" representation of the prevailing road conditions of Region 1 in contrast to Region 2. The general observation here is that the TTL value of the cached information is directly influenced by the rate of change of traffic information in the roads to be queried. This rate of change is influenced by the road length, vehicle density and existence of traffic lights. Therefore, caching policies should be adaptive and take into consideration these factors in order to adjust TTL values of traffic information. In addition the results depicted on Fig. 3(a) denote that information accuracy is heavily influenced by the presence of vehicle break-downs (Scenario 2) in the road-network. For $VITP_{TTL=0}$ queries manage to capture the conditions of the road-network with relatively high accuracy (79% and 73% of the actual values for Region 1 and Region 2), but as seen previously on Fig. 1(a), with a notably higher round-trip time. As the *TTL* value increases, there is an extensive drop in information accuracy and this can be justified as follows. For low *TTL* values ($VITP_{TTL<=100}$), the accuracy levels remain close to the ones in the original VITP. This is due to the fact that the specific TTL time interval is smaller than the duration of instability in the road-network (about 100s) caused by vehicle break-downs. Although in this interval several queries will be resolved from vehicles caches, a high percentage of the remaining queries will be resolved at the query target location thus obtaining a more accurate view of the prevailing road conditions. For $VITP_{TTL>100}$, the majority of queries are being resolved from information in a vehicle's cache. The low information accuracy provides evidence that cached information does not reflect correctly the conditions in the road-network.



(a) RSU Query Recall over TTL (b) RSU Information Accuracy over TTL

Fig. 4. Caching implications on RSU discovery.

Finally, Fig. 3(b) depicts the total number of exchanged VITP messages for the resolution of traffic information queries with respect to *TTL*. This figure illustrates the reduction in the network overhead achieved through the cache-based VITP. We intentionally skip the evaluation for Scenario 2 in this figure, since the observed reduction of messages that would otherwise be present, would be misleading to the reader. The reduced number of messages would not be an aftermath of caching but simply because vehicles do to complete their whole routes because of break-downs and thereby issue less *LookAhead* queries. In this figure, the x-axis represents the *TTL* values, whereas, the y-axis represents the total number of VITP messages exchanged over the whole simulation period. We observe that caching in VANETs decreases the traffic in the network about 21% for Region 1 and about 27% for Region 2. This reduction in the number of exchanged messages results in increasing the network's reliability; fewer exchanged messages lead to fewer network failures.

5.2 Caching Evaluation - Querying Road-Side Facilities Availability

Fig. 4(a) depicts the query recall of VITP messages for the discovery of road-side facilities. From the results, it is evident that in the lack of any information diffusion mechanism, as is the case of the original VITP ($VITP_{TTL=0}$), there is a very low probability that vehicles in the simulation will locate and retrieve information broadcasted by any RSU. In order for a vehicle to retrieve such information it is necessary for it to be located within the wireless coverage area of the RSU during a specific broadcast interval and in addition to be interested for the specific information broadcasted. On the other hand, we observe that as the *TTL* value increases ($VITP_{TTL>0}$), the query recall increases, indicating that the utilization of caching indeed allows the diffusion of information broadcasted by the RSUs and consequently increases the probability that road-side facility related queries will be successfully answered from other vehicles' caches. Even for low TTL values ($VITP_{TTL=50}$), query recall up to 52% is achieved while broadcasted information is captured with an accuracy up to 74%. The best-case scenario for query recall is when ($VITP_{TTL=200}$), meaning that vehicles cache information broadcasted by road-side units for all their drive time duration. On the other hand, as Fig. 4(b) denotes, increasing *TTL* values has a negative effect on the accuracy of the information received by querying vehicles. While ($VITP_{TTL=200}$) might give 61% and 72% query recall for Region 1 and Region 2 respectively, it only manages to reflect 47% and 53% of the information accuracy.

6 Conclusions

VANETs have been envisioned to be useful in VIS. In this work, we evaluate the performance of VIS when utilizing caching techniques to minimize the network overhead imposed by such services. In particular, we explore the utilization of caching in VANETs by extending the architecture of VITP in order to support cache-based location aware services. Simulation results have shown that the use of a TTL-based cache replacement policy in urban environments, can achieve significant improvements under both normal traffic conditions and unscheduled traffic events. In addition, results have shown that caching allows vehicles to locate and collect information from fixed RSUs that otherwise would be impossible to do so. Moreover, simulation results have also shown that the utilization of caching in VITP reduces significantly the overhead imposed on the network by minimizing the total number of exchanged messages among vehicles while requesting time-sensitive information.

References

1. Pallis, G., Katsaros, D., Dikaiakos, M.D., Loulloudes, N., Tassiulas, L.: On the structure and evolution of vehicular networks. In: Proceedings of 17th Annual Meeting of the International Symposium on Modelling, Analysis and Simulation of Computer and Telecommunication Systems. (2009) 502–511
2. Scheuermann, B., Lochert, C., Rybicki, J., Mauve, M.: A fundamental scalability criterion for data aggregation in vanets. In: Proceedings of the 15th Annual International Conference on Mobile Computing and Networking. (2009) 285–296
3. Dikaiakos, M.D., Florides, A., Nadeem, T., Iftode, L.: Location-aware services over vehicular ad-hoc networks using car-to-car communication. IEEE Journal on Selected Areas in Communications **25**(8) (2007) 1590–1602
4. Fiore, M., Mininni, F., Casetti, C., Chiasserini, C.F.: To cache or not to cache? In: Proceedings of the 28th IEEE Conference on Computer Communications. (2009)
5. Gao, J., Han, J., Yang, D., Wang, T.: Road network based adaptive query evaluation in vanet. Proceedings of the the 9th IEEE International Conference on Mobile Data Management (2008) 49–56
6. Loulloudes, N., Pallis, G., Dikaiakos, M.D.: Information dissemination in mobile cdns. Content Delivery Networks: Principles and Paradigms (Springer, 2008) 343–366
7. Zhao, J., Cao, G.: Vadd: Vehicle-assisted data delivery in vehicular ad hoc networks. IEEE Transactions on Vehicular Technology **57**(3) (2008) 1910–1922
8. Jeong, J., Guo, S., Gu, Y., He, T., Du, D.: Tbd: Trajectory-based data forwarding for light-traffic vehicular networks. In: Proceedings of the International Conference on Distributed Computing Systems, Montreal, Canada (2009) 231–238
9. Papaleontiou, L., Dikaiakos, M.D.: Trafficmodeler: A graphical tool for programming microscopic traffic simulators through high-level abstractions. In: Proceedings of the 69th IEEE Vehicular Technology Conference. (2009)