TRAC: An Architecture for Real-Time Dissemination of Vehicular Traffic Information

Shravan Rayanchu, Sulabh Agrawal, Arunesh Mishra, Suman Banerjee,* S. Ganguly†
{shravan, sulabh, arunesh, suman}@cs.wisc.edu, samrat@nec-labs.com

Introduction

Traffic congestion is a major cause of concern in many cities. A real-time information dissemination system announcing current traffic conditions to travelers is increasingly becoming a necessity in this context. The recent DSRC research initiative[1] envisions a deployment of DSRC access points (APs) along the road side to enable Vehicle to Roadside Communication (VRC) for this purpose. However, even if such an infrastructure is deployed on the roadside, it is not trivial to design a system that distributes on-demand traffic information. Existing solutions employ a centralized design where the traffic information is stored at a central database and the vehicles then query the database for information of their interest (e.g. what is the average speeds on a set of roads?). Such solutions are simply not scalable due to the huge volume of queries the central database would have to handle. Moreover, such a communication architecture does not take into account the fact that traffic related queries are *location sensitive* and that the sources for a particular query also tend to be *localized*.

Publish/subscribe systems are considered to be very effective for supporting data-centric, information dissemination applications including traffic notifications[2]. Under a pub/sub model, the vehicles could act as publishers as well as subscribers. In the role of publishers, the vehicles publish their current road and speed information to an AP (whenever under the range) and as subscribers they query for information about certain road segments of their interest. The APs can then form a broker network to carry out pub/sub operations. However, traditional pub/sub paradigm is inadequate for the purposes of traffic information dissemination for a number of reasons: (1) Traditional pub/sub systems assume that the number of subscribers typically outnumber the publishers. It is this assumption that justifies flooding of advertisements by each publisher. But in our system the number of publishers and subscribers are *large* and *equal* in number. (2) Since both publishers and subscribers are *mobile*, the system would be highly inefficient under traditional pub/sub semantics. (3) Given the intermittent connectivity of the publishers, the overhead of constructing a multicast tree rooted at each publisher would far outweigh any savings achieved in the delivery of publications. (4) Data published by vehicles under the range of an AP would be similar in content. Exploiting this locality of publications by reusing the multicast trees would reduce the cost of mobility, but traditional pub/sub system would be oblivious of such a property.

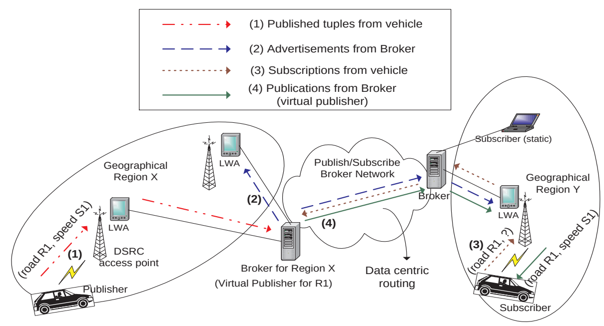

Figure 1: TRAC : Architecture

In this paper, we introduce TRAC, a scalable architecture for vehicular traffic information dissemination based on a pub/sub model. TRAC efficiently handles the above stated inadequacies using the novel concept of *virtual publisher* and optimizations like *subscription predictions*. Our objective while designing TRAC has been to distribute on-demand traffic information to highly mobile users at a minimal latency (multicast tree building time) while incurring a relatively low message overhead.

TRAC: Architecture & Algorithms

Basic Idea: The primary drawback of traditional pub/sub systems is that they store state per publisher. Even when multiple publishers publish similar information, the system constructs a tree rooted at each publisher. We propose using a *stateless* method of publishing where the system employs a *tree per each data item* rather than a tree per publisher. When a publisher connects to a broker, it simply publishes the data and disconnects. The broker then acts as a virtual publisher on behalf of the actual publisher by sending out covering advertisements, building the multicast trees and publishing the data. Since there can be multiple sources for the same data, if a tree for the data already exists in the system, a new branch created at the broker grafts onto the tree opportunistically, else a new tree for the data is constructed before sending out the publications. The concept of virtual publisher results in reduced (un)advertisement costs, as the trees are not (re)built for every movein/moveout operations of a publisher. More

*Department of Computer Science, University of Wisconsin Madison, USA
†NEC Laboratories, NJ, USA

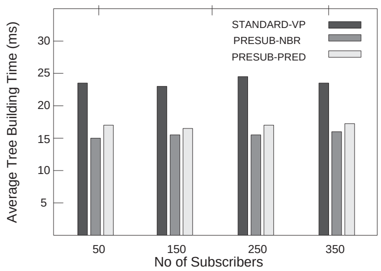

Figure 2: Avg. Tree Building Time (250 Subs)

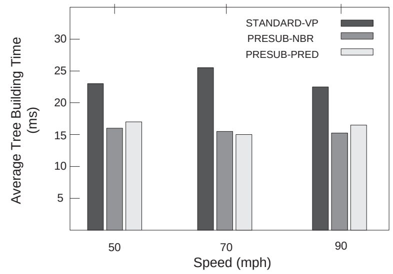

Figure 3: Avg. Tree Building Time (Speed=70mph)

Mobility Scheme	Average cost per subscription
STANDARD-VP	1
PRESUB-NBR	0.71
PRESUB-PRED	0.43

Figure 4: Average Cost per Subscription (normalized wrt STANDARD-VP)

importantly, this indirection provided by the system reduces the problem to the case of static publishers and mobile subscribers.

Architecture: In TRAC, the service area is divided into a number of local regions, each of which is assigned a broker. The vehicles are equipped with GPS, on-board diagnostics system (OBD) interface that can be used to collect data (speed of the vehicle etc) and a wireless network interface (802.11 or DSRC) to communicate with the APs. The vehicles periodically record the tupes `(road_id, speed)` and publish whenever under coverage. A Light Weight Agent (LWA) resides on the DSRC/802.11 APs to which the publishers/subscribers connect and participate in the publish /subscribe operation. The LWA is responsible for storing the connection state, caching the publications and maintaining the subscription tables. It acts as a relay agent and passes on the data to the corresponding broker for its region. As in any pub/sub system the broker is responsible for routing the advertisements and publications, subscription matching and building the multicast trees. Additionally, the broker acts as a virtual publisher by periodically publishing the aggregate information received from the LWAs in its region. We call this proposed routing scheme the STANDARD-VP scheme. Figure 1 shows the overall architecture.

Optimizations: In order to reduce the (un)subscription costs arising due to subscriber mobility we used the following schemes: (1)PRESUB-NBR: Since there is a high possibility that a subscriber after moving out of an AP connects to one of the APs in the neighboring area, we exploit this property by sending the pre-subscriptions to all the neighboring APs (LWAs) within some radius. The latency as perceived by the user would then be minimal in this case. If a subscriber does not connect to a pre-subscribed AP, unsubscriptions are issued after a timeout. (2)PRESUB-PRED: In this scheme, an LWA upon receiving an unsubscription predicts the next neighboring AP based on the road segments in the subscribed path and sends out the pre-subscription. This would require more processing on the part of an LWA, but it results in a lesser message overhead.

Simulations

We developed the TRAC Simulator by enhancing GrooveSim[3] and integrating it with a discrete event based simulator which modeled a pub/sub system. GrooveSim generates a graph based abstraction of streets for any place in the USA by importing the TIGER/Line files. Communication between the vehicles, APs and the brokers is modeled using simple message queues taking appropriate latencies ($2ms$ for each link) into account. We simulated our experiments using the map of Manhattan county (service area of 23.7 square miles divided into 16 regions with a total of 9312 road segments). A total of 92 APs (LWAs) were placed at road intersections, uniformly distributed with a mean distance of around 600 meters. We initialized 650 vehicles at random addresses in the area which subscribed to a randomly chosen path whenever under coverage. Figure 2 shows the average multicast tree building time for increasing number of subscribers. Figure 3 shows the same for increasing mobility. The STANDARD-VP scheme performs very well with a tree building time of less than 25ms. Since the average connection time even at 90mph was about 5 seconds, all the publications were delivered to the interested subscribers. The PRESUB-PRED and PRESUB-NBR schemes further reduce the tree building time. Moreover, the tree building time is impervious to increasing number of subscribers/mobility which exhibits good scalability properties of the architecture. The average message overhead for STANDARD-VP scheme was 7.24 messages/subscription. This low message overhead can be attributed to the concept of virtual publisher which results in reduced (un)advertisement costs. Figure 4 shows the normalized cost. As before, the proposed optimizations perform better than the STANDARD-VP scheme.

Conclusion and Future Work

We presented TRAC: a scalable, distributed traffic information dissemination system. TRAC uses the concept of virtual publishers and subscription predictions to completely decouple publisher mobility from the system and enable traffic information aggregation from multiple publishers. We are currently investigating the practical issues concerning the throughput and range of coverage with drive-thru experiments over 802.11b. We further plan to investigate the impact of different broker network topologies on the performance of TRAC.

References

[1] DSRC WG, http://grouper.ieee.org/groups/scc32/dsrc/index.html

[2] V.Muthusamy, A. Jacobsen, ”Effects of routing computations in content-based routing networks with mobile data sources”, MOBICOM’05

[3] R. Mangharam, R. Rajkumar, ”GrooveSim: a topography-accurate simulator for geographic routing in vehicular networks”, VANET’05