AdTorrent: Digital Billboards for Vehicular Networks

Alok Nandan, Shirshanka Das, Biao Zhou, Giovanni Pau, Mario Gerla
Computer Science Department
University of California Los Angeles
Los Angeles, CA 90095-1596
{alok, shanky, zhb, gpau, gerla}@cs.ucla.edu

Abstract

One of the most important sources of revenue for big Internet-based companies are advertisements. With vehicular networks poised to become part of the Internet, this new "edge" of the Internet represents the next frontier that advertising companies will be striving to reach.

In this paper we investigate the feasibility of targeted dissemination of ad content in a car network. We present Digital Billboards, a scalable "push" model architecture for ad content delivery. We then propose, AdTorrent, an integrated system for search, ranking and content delivery in this architecture. We evaluate our design using a realistic vehicular mobility model which captures mobility characteristics such as temporal and spatial dependencies and geographic restrictions.

1 Introduction

As advertisers struggle to reach increasingly distracted and jaded American consumers, they have sought nontraditional media for their Ads, from elevators to cell phone screens.

Content-targeted advertising paradigm has proved to be a resounding success in advertising on the conventional Internet. As the Internet expands to mobile devices, even vehicular nodes are becoming a part of the "edge" of the Internet. Several interesting challenges in application design arise, while designing a targeted ad delivery mechanism for cars.

Consider this scenario; you are driving on Interstate-5 from Los Angeles to San Francisco, visiting relatives. On the way, you realize that you need to buy some gift for them. You initiate a search for "new DVD releases". The Ad software is not only keyword aware but also *location* aware. Hence the search results return not only the content or latest DVD releases but also the latest deals on those DVD releases in stores.

Imagine another similar scenario, you are traveling to Las Vegas. You are 50 miles from the city and want to get information of all hotels less than 200 dollars cost, preferably with virtual tours of the hotels.

In this paper we present Digital Billboards, an ad-service provider architecture for urban vehicular networks. We use a realistic group-mobility model to evaluate certain key parameters that guide the design of our advertisement search and delivery service. Finally, we propose *AdTorrent*, a location aware advertisement dissemination service for this architecture. The vehicular environment presents interesting challenges and constraints not encountered in content delivery on the Internet. In particular, mobility and the limited/intermittent bandwidth cause sessions to be transitory and subject to frequent disconnections.

AdTorrent seeks to provide to the user, relevant Ads guided by a particular keyword search. Ads potentially can be multimedia clips, for example, virtual tours of hotel rooms, trailers of movies in nearby theatres or a conventional television ad.

Our Contributions Vehicular Ad Hoc Networks (VANETs) present interesting challenges to protocol design. One of the key differentiating characteristics is the time-varying nature of vehicle densities and the mobility model. Mobility has a important impact on application design.

The contributions of this paper are as follows: (1) firstly, we propose a novel push-model based location-aware ad service architecture, designed for vehicular environments (2) secondly, we present a group mobility model for urban vehicular traffic, (3) thirdly, we present a peer-peer protocol that enables efficient keyword-set based search. We propose a novel search result ranking scheme that is optimized for vehicular networks. We use swarming, to quickly deliver top ranked content to the end user.

1.1 Organization of the Paper

The rest of the paper is organized as follows. Section 2.1 describes the Vehicular communication architecture. Section 3 gives an overview of the operation of the ad service in a vehicular scenario. Section 4 describes the novel mobility model we used for the purpose of evaluation and details our evaluation of the performance of our protocol using simulation. Section 5 gives a brief overview of AdTorrent, based on a push-model of content dissemination based on a popular swarming protocol. We outline the related work in section 7. Finally, Section 8 concludes the paper.

2 Preliminaries

In this section we describe the vehicular environment and the assumptions about the environment we used to design our protocol.

The network consists of a set of N nodes with same computation and transmission capabilities, communicating through bidirectional wireless links between each other, this is the infrastructure-less ad-hoc mode of operation. There are wireless gateways at regular intervals providing access to the rest of the Internet using infrastructure support(either wired or multi-hop wireless). We do not assume any routing protocol running in the underlying network. Nodes may or may not run the peer-peer application protocol. Hence, P the size of the peer-peer network is such that, $P \subset N$ and is established on top of this vehicular network.

In addition, we assume a CSMA/CA MAC layer protocol(IEEE 802.11a) that provides RTS/CTS-Data/ACK handshake sequence for each transmission. Nodes use TCP for reliable transfer of data and UDP for dissemination of index updates. It includes data packets as well as gossip messages. However, the packet, is the unit for network layer. We assume each node is reachable to every other node.

Our vehicular wireless architecture is composed of two kinds of communications, namely, vehicle-vehicle and vehicle-gateway. Dedicated Short-Range Communication(DSRC) [13] is a short to medium range communication technology operating in the 5.9 GHz range. The key characteristics of DSRC are given in the table 1. For a more detailed description, we refer the reader to [17].

2.1 Assumptions

1. *Data is not strictly real-time:* There are no real-time constraints on the data, thus in some sense, the data is delay-tolerant.

Table 1. DSRC Characteristics

Parameter	Value
Range	1000m
Frequency Band	5.9Ghz
Speed	$\leq$ 85mph
Data Rates	6-27Mbps(depending on Range)

2. *Storage of data is not constrained:* VANETs are typified by no power constraints and no data storage limitations.

3. *Data is meta-tagged:* Meta-data can be the filename, the format and/or key-words extracted from the data. For some types of data, such as text documents, metadata can be extracted manually or algorithmically. Some types of files have metadata built-in; for example, ID3 tags on MP3 music files.

4. *Communication between vehicles is over a low data rate connection:* While this constraint depends on the radio technology used. Currently, 802.11x devices will offer *goodput* of the order of a few hundred Kbps.

5. *Push model:* Data is being continually "pushed" by the access points to the nodes in the transmission range.

6. *Multi-hop delivery:* It is infeasible to transmit data to more than 2-3 hops.

3 The Digital Billboard Architecture

The digital billboard architecture serves to deliver Ads to the vehicles that pass within the range of the Access Points (APs). This architecture is:

- *Safer:* Physical billboards can be distracting for drivers

- *Aesthetic:* The skyline is not marred by unsightly boards.

- *Efficient:* With the presence of a good application on the client (vehicle) side, users will see the Ad only if they actively search for it or are interested in it.

- *Localized:* The physical wireless medium automatically induces locality characteristics into the advertisements.

Every Access Point (AP) disseminates certain sets of Ads that are relevant to the proximity of the AP deployment. This is reasonable since it is the extension of the physical billboards that we very often see lined on the streets and freeways that advertise the best offers available at the next restaurant. Our AP acts as a “digital billboard”. This model makes sense economically as well since business owners in the vicinity subscribe to this digital billboard service for a fee. The APs continually disseminate these advertisements to the vehicles that traverse the coverage area. The dissemination rate can be determined by different *levels of service* demanded and paid for by the billboard owner.

Leveraging this architecture, we want to design a location-aware distributed mechanism to search, rank and deliver content to the end-user (the vehicle). We focus not only on simple text-based Ads but also on larger multimedia Ads, for example, trailers of movies playing at the nearby theater, virtual tours of hotels in a 5 mile radius, or conventional television advertisements relevant to local businesses.

Every node that runs the application collects these advertisements and indexes the data based on certain metadata which could be keywords, location and other information associated with the data. We assume that Ads are uniquely identifiable using a document identifier (DocId).

In the next section we describe a group mobility model for an urban vehicular network. The model guides us in the design of AdTorrent, an application for advertisement search and delivery on the vehicular network.

4 Mobility Model

The mobility model is designed to describe the movement pattern of mobile users, and how their location and velocity change over time. Mobility patterns play a significant role in determining the protocol performance and thus are an important parameter to the protocol design phase. It is important for mobility models to emulate the movement pattern of targeted real life applications in a reasonable way. Otherwise, the observations made and the conclusions drawn from the simulation studies may be misleading. Thus, when evaluating our vehicular ad hoc network protocol, it is imperative to choose the proper underlying mobility model. Different application scenarios lend themselves to different mobility models. For example, a campus-wide wireless network deployment will see different mobility patterns (less constrained, more random) than an urban vehicular grid scenario (low entropy of vehicles, group mobility).

In modeling and analyzing the mobility models in a VANET, we are more interested in the movement of individual nodes at the microscopic-level, including node location and velocity relative to other nodes, because these factors directly determine when the links are formed and broken, since the communication pattern is peer-to-peer.

We used the US Census bureau data for street level maps. As a starting point we used the script from [3] paper to generate the mercator projection of the data, in our case the local map of an area around UCLA in Figure 1.

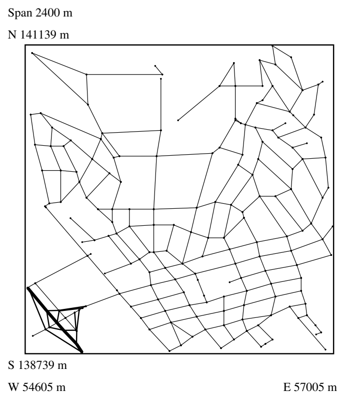

Figure 1. Local Map of Westwood, an area around UCLA

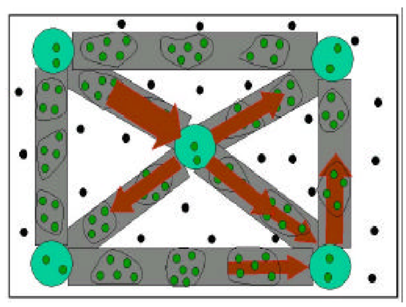

Figure 2. Overview of Real Track Based Group Mobility Model.

However, in the paper by Saha *et al* [3], the actual mobility model is quite similar to the Random Waypoint in the sense that the vehicle’ arrival and direction and speed are similar to the Random Waypoint model. Hence the results that the vehicular mobility model is very similar to the Random Waypoint model. In reality, a complex mobility behavior is observed. Some nodes move in groups; while others move individually and independently; a fraction of nodes are static. Moreover, the group affiliation is not permanent. The mobile groups can dynamically re-configure themselves triggering group split and mergence. All these different mobility behaviors coexist in vehicle or urban scenarios. We refer to the non-uniform, dynamic changing scenario described above as “heterogeneous” group mobility scenario [4]. A good realistic mobility model must capture all these mobility dynamics in order to yield realistic performance evaluation results, which, unfortunately, is not satisfactorily captured in any of the existing models.

We propose a ”real track” based group mobility model (RT model) that closely approximates the above ”heterogeneous” mobility patterns happening in the scenarios of vehicle ad hoc networks. It models various types of node mobility such as group moving nodes, individually moving nodes as well as static nodes. Moreover, the RT model not only models the group mobility, it also models the dynamics of group mobility such as group merge and split.

The key idea of our proposed model is to use some ”real tracks” to model the dynamics of group mobility. In our simulation scenarios, these ”real tracks” are represented by streets from real map. The grouped nodes must move following the constraint of the tracks. At the switch stations, which are the intersections of tracks/streets, a group can then be split into multiple smaller groups; some groups may be even merged into a bigger group. Such group dynamics happen randomly under the control of configured split and merge probabilities.

Nodes in the same group move along the same track. They also share the same group movement towards the next switch station. In addition, each group member will also have an internal random mobility within the scope of a group. The mobility speeds of these groups are randomly selected between the configured minimum and maximum mobility speeds. One can also define multiple classes of mobile nodes, such as pedestrians, and cars, etc. Each class of nodes has different requirements: such as moving speed etc. In such cases, only nodes belonging to the same class can merge into a group.

Groups split and merge happen at the switch stations. Each group is defined with a group stability threshold value. When at the switch stations, each node in the group will check whether its stability value is beyond its group stability threshold value. If it is true, this node will choose a different track from its group. A group split happens. When several groups arrive at the same station and select the same track for the next movement, naturally, they will be merged into one bigger group.

The proposed RT model is also capable of modeling randomly and individually moving nodes as well as static nodes(such as sensors). Such non-grouped nodes are not restricted by the switch stations and real tracks. Instead, their movements are modeled as random moves in the whole field.

Fig. 2 illustrates a main idea of the proposed real track based group mobility model. In this example, group moving nodes are moving towards switch stations along the tracks. They split and merge at switch stations as shown in the figure. The black nodes in Fig. 2 represent the individually moving nodes and static nodes. They are placed and move independent of tracks and switch stations.

We evaluate the scenarios along the following metrics as defined in [3]. For brevity we present only the average connectivity duration metric, which is the most essential for protocol design in our scenario.

Average connectivity duration: This is the duration of time 2 nodes have a path between them. We further quantify this metric based on the maximum allowable hops for any path between the two nodes. This metric is relevant to our application as it justifies the usage of a swarming content delivery model in the presence of limited connectivity between the nodes.

We used a 500m transmission range for the radios. In our case we adjusted the number of nodes, to 30, 50 and 60, spread over an area of $2400m \times 2400m$. The average number of nodes in the transmission range were 4.1, 6.9 and 8.1 respectively. Each run of simulation were 900s long. Also we evaluate the scenarios are regular intervals of 10s.

We observe that the for a 4-hop limit path the connectivity duration has a almost 100% increase as opposed to a 3-hop limited path. Hence one of the design parameters of the protocol, the limit to which the search and the pieces that need to be collected is a function of the mobility pattern. For urban vehicular mobility we conclude that the incremental gain from increasing the hop limit up to 4 might be useful for increasing the robustness of protocol performance.

5 AdTorrent: Design

We outline the primary design goals of *AdTorrent* as follows:

1. location aware torrent[1] ranking algorithm;

[1] we use the terms : document and torrent interchangeably

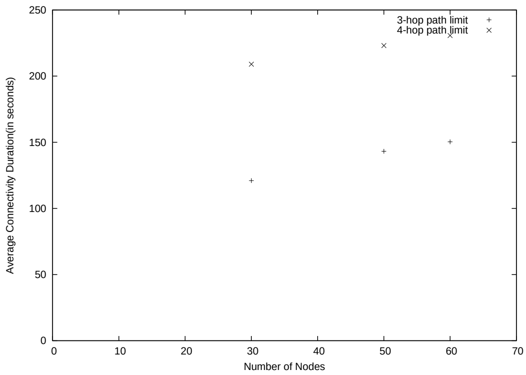

Figure 3. Average Connectivity duration

2. search should be simple and robust, in presence of node failures and departures;
3. leverage churn
4. minimal overhead of communication;

There are three main tasks performed by our application. Namely, *search* for relevant ad-content, *query dissemination* and *content delivery*. We address each of these functions in the following sections.

5.1 Search

Search involves associating keywords with document identifiers and later retrieving document identifiers that match combinations of keywords. Each file is associated with a set of metadata: the file name, its format, genre(e.g. in advertisements). For some types of data, such as text documents, metadata can be extracted manually or algorithmically. Some types of file have metadata built-in; for example, ID tags on MP3 files.

Distributed Hash Tables (DHTs) have been proposed for distributed lookups. We do not use a DHT for distributed lookup, since it is well-known that DHT's are not very stable under high churn [2]. Our query dissemination mechanism aims to achieve robustness rather than communication efficiency.

However, we introduce certain optimizations to the index information dissemination that will reduce the amount of search query communication overhead.

Indexing

Vertical partitioning divides an index across keywords. Horizontal partitioning divides an index across documents, so all the nodes have an entry for *each* keyword. Vertical partitioning minimizes the cost of searches. However, horizontal partitioned index reduces the cost of update of document. In our scenario, the number of queries will far outnumber the number of updates, since we assume the documents typically searched for, are not changing frequently. Our index is partitioned based on set of keywords. This was first introduced in KSS [18]. The motivation of using a keyword set based indexing is the reduction of overhead in terms of query data information. The downside of this approach is higher cost of insert and storage. In VANETs, we believe, storage will not be a limitation. In the AdTorrent application, one of the key characteristics will be the infrequent updates of ads, maybe once in one day or less.

Table 2. Index

Key	URL of Data	Meta-Data
Hash(AB)	131.179.168.66	Size=1MB
Hash(BC)	131.167.134.71	Genre=Ox0FAB

The index we maintain is distributed. However, every node tries to maintain information of the all the two-hop neighborhood of itself. The documents are indexed on keys. Keys consists of SHA-1 hashes of the keywords sets. Along with the keys and the URL of the data, we also store additional meta-data associated with the data. The metadata is stored in an index corresponding to each subset of at most K metadata items. KSS uses a distributed inverted index to answer queries efficiently with minimal communication overhead. Each entry of the index contains: (1) the hash of the searchable sets of keywords as the index key, (2) a pointer to the data such as the URL of the data and, (3) meta-data associated with the data, e.g. the table 3.

Placement

In a wireless scenario, it makes sense to co-locate the index and the data corresponding to the index entry. This is to reduce the overhead of data discovery latency once the index for that data has been located.

5.2 Query Data Dissemination Optimization

Each node disseminates the content availability information in the form of a bloom filter. Bloom filter [1] is an efficient method to test for set membership. In our case, the bloom filter is constructed to test the keyword membership for a particular node. A bloom filter is computed by each node based on the keywords related to the data, the node has stored. Since the data downloaded is only once every AP encounter or if the node explicitly downloads some swarming torrent, hence the updates of bloom filter and dissemination is not very frequent.

We now enumerate the basic steps of the algorithm.

Table 3. Parameters

Name	Purpose
Timers (2)	Search End and MetaData Search
k_1, k_2, m	Aggressiveness of the search, ties to the mobility , node entropy
k	Keyword Set parameter
MetaData	e.g torrentID/Multimedia

Algorithm 1 AdTorrent: Query Processing, Ranking and Content Delivery

user_input = **search("A B C")**

num_local_entries = **lookup_local_index**(hash(AB), hash(BC), hash(CA))
if (num_local_entries $> k_1$)
    goto *LookupDone*
else
    /* Found $< k_1$ local entries */
    /* not in the 2-hop neighborhood */
    num_remote_entries = **scoped_flood**(hash(XY), m)
    /* $\forall$ XY $\in$ AB,BC,CA */
    After T_1 seconds, if NO response, return NO
    If k_1 entries are found then

LookupDone:
    /* now have k_1 entries (local or remote in 1-4 hops) */
    send_udp_ctrl(Hash(XY))→METADATA(e.g. TorrentID)

    /* Collect meta_data after T_2 */
    torrent_ranking(meta_data, params)

Step Final:
    swarm(TorrentID)
    /* returns a list of Peers & HopCounts*/
    /* (this may be beyond the the scope of the search) */
    decentralized_tracker()
    /* By allowing the list of Peers beyond the k-hop scope of the search, we add some randomization */

The indexing scheme described above does not have a document ranking algorithm. The order of query results propagation and display is equally important for successful and timely dissemination in a VANET. This assumes further importance in VANET since the mobility of nodes might render some query results obsolete or irrelevant in short period of time. We incorporate a location metric in the document ranking scheme. One way to support the document ranking would be to score a document based on the following categories.

1. location
2. max # of pieces
3. stability of neighbors
4. relevance of the DocID to the Meta-Data queried

5.3 Content Delivery

Once an accurate document ranking has been performed, the actual delivery of content can be done by swarming. One of the factors that determined the ranking of a document in the query results was the number of sub-pieces of the document that were available and the location of the pieces. Thus the torrent ranking guides the system to choose documents which are more amenable to swarming downloads. The vehicle now joins the existing BitTorrent-like stream to start getting pieces of the document from neighboring nodes. We propose to do this using our earlier work in [17]. Swarming allows us to be robust to node failures (cars going out of range or powering down) and efficient in terms of delivery (the cars form a sort of end-system-multicast tree). However, the success of swarming especially in wireless ad-hoc networks depends hugely upon cooperation among vehicles at both the routing layer (forwarding packets for others) and at the application layer (sharing the advertisements that have been downloaded). In the next section we address the concerns with respect to selfish behavior on the AdTorrent network and discuss ways to mitigate it.

6 Discouraging Selfishness on the AdTorrent Network

AdTorrent shares with other proposed peer to peer content delivery protocols [6, 17] and even MANET routing protocols, the implicit assumption that nodes in the network are cooperative and honest. We work with the assumption that a majority of nodes in the network will be playing the "fair" game, and raise the bar for cheating to make cooperation an evolutionary stable strategy (ESS) [21].

6.1 Adversary Model

We discuss two broad kinds of "cheating" that a node may indulge in.

Routing Layer

People have proposed selfish nodes which do not want to cooperate in the forwarding of data packets for others in the context of multi-hop ad-hoc wireless networks. These nodes are not "attacking" the network in terms of trying to stop routing, they are just trying to not forward packets for others unless compelled to. Selfish nodes in these settings are driven mainly to reduce consumptions of CPU cycles, battery life and bandwidth. All of these concerns are relevant only to power-constrained, cpu-strained devices. Another possible motivation would exist if the scenario permitted the selfish node to transmit one of its own packets rather than forward someone else's packets. We argue that in the vehicular wireless network, both these motivations do not exist. First, vehicles are not power-constrained or CPU-constrained in any way to motivate selfishness with respect to transmission of packets on the wireless interface. Secondly, in the wireless network, typically due to the broadcast nature of the medium, if the selfish node does not forward some packet, someone else within the wireless range would. This will lead to collision if the selfish node tries to sneak in its own packet, thus the selfish node doesn't benefit significantly by not forwarding traffic. There exists a feasible solution to this situation when all nodes behave selfishly but the end-result will be that no packets will be forwarded, thus it will not be a stable solution.

Application Layer

At the application layer too, we consider selfish nodes which just want to reap the benefits of the AdTorrent network without cooperating with it. This could be accomplished for instance by only turning on the AdTorrent application when the user in the car wishes to search for content and then turn it off immediately after the user is done with the search and consequent delivery of the content. This is similar to the behaviour of free-riding BitTorrent users who turn on the BitTorrent client, get the file they need and then turn it off to avoid seeding for other peers. We also want to catch nodes who lie about the content they have. Nodes could cheat by pretending to have nothing even while they continue to download content from other people. Thus for example, a selfish node could broadcast a bloom filter of an empty hash. Nodes could also broadcast false bloom filters, e.g. by generating a random bloom filter. Note that in AdTorrent cooperation is automatically built in only for the duration of the download of the file. After the file has been downloaded, the node can choose to not advertise this newly acquired content, thus denying service to neighbors who might have been interested in this content at a later time.

6.2 Securing AdTorrent

6.2.1 Detection of Cheating

This section looks at a couple of approaches that have been proposed to handle similar problems and discusses how to adapt these techniques to our particular scenario.

Routing Layer Cheating

Watchdog [22] was one of the first techniques proposed to catch malicious nodes in an multi-hop wireless network. It involves upstream nodes eavesdropping on the next hop downstream to check whether they forward packets correctly. Watchdog is a simple and effective technique to detect routing-layer cheating. There exist ways to subvert watchdog. However all of them require directional antennas and sophisticated power control algorithms. We do not believe that the simple civilian application of ad-dissemination can motivate people to engage in complicated attacks to simply free-ride on the network. Catch [21] is a recent work which addresses the problem of corner nodes in static ad-hoc wireless networks. Catch builds on top of watchdog to distinguish between a "poorly connected node" and a selfish node which just appears to be poorly connected. In vehicular networks, vehicles are constantly moving in either an infinitely straight line (freeway scenarios) or are part of an infinite grid (urban road scenario). Thus the corner node situation does not arise at all, and even when it does, the situation is transient due to the constant mobility. Thus we argue that we do not need extra techniques on top of watchdog to detect routing layer cheating. Additionally, we have discussed earlier why we believe that vehicles will have very little motivation to cheat at the routing layer.

Application Layer Cheating

- Advertising false content: Neighbors can monitor the node and detect when it is downloading something. They can independently compute the new bloom filter that the node should broadcast after the download is done. If the new bloom filter does not match the broadcasted hash, then the nodes can blacklist the neighbor.
- Not responding to queries: When their is a match on the bloom filter for a particular keyword search, the "requester" sends a message asking the node for some data. The "requested" can send back a "Didn't match" or a "Busy" response. Neighbors of the "requested" node can monitor the node to see if it is truly busy or not. If it is sending back "Busy" responses when it is not, then the node is blacklisted. If the node sends back more than one "Didn't match" response to different nodes, then the node will be blacklisted.

6.2.2 Propagation of Cheat Information

URSA [20] is a robust access control mechanism for mobile ad-hoc networks. Given a detection mechanism for detecting cheating nodes in the neighborhood, URSA does a good job of propagating this information, and resisting individual and collaborative attacks against the access control. URSA assumes that there is a system-wide public and private key. Each node in the system v_i has partial knowledge of the private key. This mechanism could be implemented easily in AdTorrent during the software installation. Vehicles receive partial tickets, k of which can be combined to construct a valid ticket with a certain lifetime. When vehicles are detected to be cheating, the accuser sends out a signed scoped broadcast of the accusation. k accusations result in the node being refused access rights to the ad-hoc network.

We propose to use URSA for the propagation of cheating information and the access control for the network. As discussed earlier, we use our application-specific mechanisms to detect application-layer cheating.

7 Related Work

This section summarizes previous work related to cooperative data transfer protocols for the wired settings as well as gossiping mechanisms used in ad hoc routing. BitTorrent is a popular [5] file-sharing tool, accounting for a significant proportion of Internet traffic. There are two other peer-peer bulk transfer protocols namely, Slurpie and CoopNet. CoopNet uses HTTP redirect messages sent by the server to enable cooperative downloads. CarTorrent [17] is a recent work that extends the BitTorrent protocol to the vehicular networks scenarios addressing issues such as intelligent peer and piece selection given the intermittent connectivity and limited bandwidth of the wireless medium.

The *Infostations* model [8] of wireless ad-hoc networks aims to provide trade-offs between delay and capacity of these networks by providing geographically intermittent connectivity. It has been observed that the *Infostations* model trade *connectivity* for *capacity*, by exploiting the mobility of the nodes.

8 Conclusion

In this paper we presented a novel application involving search and location aware content delivery(in our case advertisements/deals) to the nodes. We proposed a efficient keyword search on this content overlay. We have proposed a new mobility model for our protocol design and analysis. In particular, our new model uses *real* street map data, modeling vehicles traveling on these streets. We analyzed the properties of this mobility model that affect protocol design. The mobility model shows that the scope of epidemic dissemination of search and index information beyond a certain

Our work is a first step in the design and eventually the implementation of AdTorrent application.

References

[1] B. Bloom, Space/time tradeoffs in in hash coding with allowable errors. CACM, 13(7):422-426, 1970.

[2] S. Rhea, D. Geels, T. Roscoe, and J. Kubiatowicz, Handling Churn in a DHT, USENIX 2005

[3] A. Saha and D. Johnson, *Modeling mobility for vehicular ad-hoc networks* ACM VANET 2004

[4] B. Zhou, K. Xu and M. Gerla, *Group and swarm mobility models for ad hoc network scenarios using virtual tracks,* In Proceedings of MILCOM 2004

[5] B. Cohen, *Incentives Build Robustness in BitTorrent,* IPTPS 2003.

[6] R. Sherwood, R. Braud and B. Bhattacharjee, *Slurpie: A Co-operative Bulk Data Transfer Protocol,* In Proc. of IEEE INFOCOM 2004.

[7] K.Tang, M. Gerla and R. Bagrodia, *TCP Performance in Wireless Multi-hop networks,* Proceedings of the Second IEEE Workshop on Mobile Computer Systems and Applications, 1999.

[8] T. Small and Z.J. Haas, *The Shared Infostation Model - A New Ad Hoc Networking Paradigm (or Where there's a Whale, there's a Way),* In the Proc. of ACM MOBIHOC 2003

[9] Z.J. Haas, J.Y. Halpern and L.Li, *Gossip based ad hoc routing* , In the Proc. of IEEE INFOCOM 2002

[10] *QualNet user manual* www.scalable-networks.com

[11] K. Birmanm M. Hayden, O. Ozkasap, Z. Xiao, M. Budiu and Y. Minsky, *Bimodal Multicast,* ACM Trans on Computer Systems, vol 17, no.2 pp 41-88, 1999

[12] P. Eugster, S. Handurukande, R. Guerraoui, A.M.Kermarrec and P. Kouznetsov, *Lightweight probabilistic broadcast,* in Proc. of IEEE International Conference on Dependable Systems and Networks(DSN 2001)

[13] *Dedicated Short Range Communication Architecture,* www.astm.org/SNEWS/MAY_2004/dsrc_may04.html

[14] F. Bai, N. Sadagopan, and A. Helmy, *Important: a framework to systematically analyze the impact of mobility on performance of routing protocols for ad hoc networks,* in 1. A Survey of MobIlity Models 29 Proceedings of IEEE Information Communications Conference (INFOCOM 2003), San Francisco, Apr. 2003.

[15] T. Camp, J. Boleng, and V. Davies, *A Survey of Mobility Models for Ad Hoc Network Research,* in Wireless Communication and Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications, vol. 2, no. 5, pp. 483-502, 2002.

[16] C. Bettstetter, H. Hartenstein, and X. Perez-Costa, *Stochastic Properties of the Random Waypoint Mobility Model,* in ACM/Kluwer Wireless Networks, Special Issue on Modeling and Analysis of Mobile Networks, vol. 10, no. 5, Sept 2004.

[17] A. Nandan, S. Das, G. Pau, M.Y. Sanadidi and M. Gerla, *Cooperative Downloading in Vehicular Wireless Ad Hoc Networks,* In Proceedings of Wireless On-Demand Networks and Services, St. Moritz, Switzerland, Jan 2005.

[18] Omprakash Gnawali, *A Keyword-Set Search System for Peer-to-Peer Networks,* Masters Thesis, Massachusetts Institute of Technology, 2002.

[19] Agilent Technologies, Inc. www.agilent.com

[20] H. Luo, J. Kong, P. Zerfos, S. Lu, L. Zhang *URSA: Ubiquitous and Robust Access Control for Mobile Ad-Hoc Networks* In IEEE/ACM Transactions on Networking, October 2004

[21] R. Mahajan, M. Rodrig, D. Wetherall, and J. Zahorjan, *Sustaining Cooperation in Multi-hop Wireless Networks* To appear at Networked Systems Design and Implementation (NSDI), May 2005

[22] S. Marti, T.J. Giuli, K. Lai, and M. Baker *Mitigating Routing Misbehavior in Mobile Ad Hoc Networks*

[23] X. Hong, M. Gerla, G. Pei, and C.-C. Chiang, *A group mobility model for ad hoc wireless networks,* in Proc. ACM Intern. Workshop on Modeling, Analysis, and Simulation of Wireless and Mobile Systems (MSWiM), August 1999. 30 Chapter 1

[24] A. Jardosh, E. M. Belding-Royer, K. C. Almeroth, and S. Suri. *Towards Realistic Mobility Models for Mobile Ad hoc Networks,* in Proceedings of Ninth Annual International Conference on Mobile Computing and Networking (MobiCom 2003), San Diego, CA, pp. 217-229, September 2003.

[25] J. Yoon, M. Liu and B. Noble, *Sound Mobility Models,* in Proceedings of Ninth Annual International Conference on Mobile Computing and Networking (MobiCom 2003) September 2003, San Diego, CA.