IVAN: VANET Content Distribution over Named Data Network

Chih Chin Chang, Mario Gerla, Ciaran MC Goldrick
Computer Science Department
University of California Los Angeles
Los Angeles, USA

***Abstract*—Information centric network architecture offers some features that could be naturally exploited to meet the challenges posed by a vehicle ad-hoc network environment. By deploying the named data network architecture as a part of the application, location-based advertising can benefit content distribution. The main challenge of vehicular mobility in an urban environment is the fast rate of acceleration and deceleration so that a road side unit, which acts like a content distribution center, will not have enough time to deliver an entire piece of content to any given vehicle. Vehicle-to-vehicle networking must cooperate along with Infrastructure-to-Vehicle networking to forward and distribute or stream large data. This paper introduces a protocol of vehicular content distribution by delivering chunks of a file or buffer in a vehicular network prioritized by localized rarity and routed over the named data network architecture.**

Keywords—VANET; torrent; content distribution network; named data network; information centric network; location based routing; last encounter routing

I. Introduction

Content distribution and content streaming in a vehicle ad-hoc network (VANET) will become an essential component in a future environment of self-operating vehicles. The vision of efficient self-navigating automobiles in densely populated urban areas promises a highly connected network of variably accelerating mobile vehicle nodes all with some contact with some form of an infrastructure of road side units (RSU), which can become content distribution centers for the cars on the road. However, although this vision of highly connected cars with Internet access for the riders who can all enjoy data content on their daily commute, a VANET poses many challenges from the perspective of content distribution.

A. Mobility

The most apparent challenge is one of mobility. Because vehicles are expected to be moving quickly, navigating the urban environment with high efficiency, their contact with RSUs will be brief. And therefore, the problem mainly arise when content data cannot be fully delivered from the RSU to the vehicle within the limited amount of time of contact between the two. This is a problem of Infrastructure-to-Vehicle communication (I2V). Additionally, some content files are not always static. Ideally, a stream of data delivery will be available for RSUs to deliver streaming of data to the vehicles. This will be useful for extremely large files or live broadcast events such as television streaming.

B. Multicast

Ideally, in a VANET, because data delivery must be broadcasted into the ad-hoc network, multicasting would be a useful feature to minimize overhead and congestion from redundant broadcast requests for the same piece of data. Should multiple vehicles request for the same piece of data, these requests should be served by a single multicasting response.

Traditionally, a MANET routing protocol like AODV does not offer this feature. Instead, it utilizes broadcast flooding and forwarding to chain a breadcrumb of forwarding nodes going from one host Alice to another host Bob. When Bob responds, the data will be returned along the path of the broadcast. In this scheme, no information about the data is shared to the network because this information is not necessary for routing.

Named Data Network (NDN) is an implementation of the Information Centric Network (ICN) principle, which reverses the traditional paradigm. Instead of routing by hosts, which would be a natural model for communication between two nodes as is the traditional usage for networks, data packets would be routed by names. Essentially, the client-server model has been modified to the consumer-producer model. A server does not serve data to a client request, instead consumers send Interests for a particular data by name for the network to route to a producer of that named data who will respond with Data, which would be forwarded to all Interests by the forwarding routers.

C. Delay Tolerance

It is the nature of MANETs for the network to account for data loss and delivery disruption. Because nodes are mobile and location of each node change constantly, a node cannot assume that the path discovered for routing will be viable over time. Therefore, each node along a path, which potentially could be any vehicle in the VANET, must maintain some sort of cache to allow fast retransmit along the path rather than between end to end hosts.

In a traditional end-to-end protocol, such caching is not offered. Bittorrent, a peer-to-peer distribution protocol, solves this issue by its file manager which turns the file system into a sharable network cache for as long as the user agrees to share the data piece. Sharing of a data piece managed by the file manager runs into data integrity issues when an end user may make changes to the file while the file is being shared.

In NDN, any forwarding and consuming node will act as a data cache for the data forwarded or consumed. This mechanism assumes that a data that has been forwarded or consumed has high relevancy for a period of time along this discovered path. Intuitively, because of its multicasting nature, NDN nodes saves a copy of the data being shared to its network cache because it assumes that it is along an optimal route for the data so that another Interest would not need to be routed to the producer to return the data.

D. Locality

Range of network devices is not the only issue with mobility in a VANET. Because the data content carried by each vehicle moves, end-point locations become vital for routing purposes. Vehicle nodes are not just identified by some unique identifier, but their locations are also important especially in an end-to-end protocol.

Last Encounter Routing (LER) was an NDN routing protocol for tracking content data and its owners as the owner moves around the network. Periodically, the owner of the data would broadcast its ownership and its current location. Nearby nodes would store this location in a simplified routing table. The location would be used by each node to determine its current distance to the owner should an interest for the data be received. The interest would be forwarded by broadcast with priority given to the nodes with the shortest distance. And because distance is calculated by each node, there does not need to be any cooperation between these forwarding nodes. In such a way,

One of an interesting phenomenon occurs when the owners are static in a mobile network such is the case for RSUs in a mobile VANET. In such a situation, all of the vehicles that receive advertising of the content data will have the same location marked for the data. This is not a problem with the protocol itself because distance can still be calculated and thus priority can still be assigned. However, as vehicles move away from the RSUs, distance can become rather large and therefore backoff delay can be unnecessarily long. But as data is replicated and distributed across the network to be shared, location will resemble more like the traditional LER scenario.

E. Content Availability

Content distribution in a VANET would require cooperating forwarding and distributed content delivery amongst Vehicle-to-Vehicle (V2V) and I2V communications. Content data will need to be replicated across the network to allow vehicles who are not reachable by the RSU to receive the data requested.

To determine the priority for content delivery to maximize data distribution in the network, the bittorrent protocol introduces the priority of delivering rarest data piece first. In a VANET and other location sensitive mobile networks, data distribution is a challenge to maximize the physical coverage of the data in the network. The issue often arise when nodes become unavailable in the network, and because forwarding in a location sensitive network relies on node availability to connect physical locations, the number of data redundancy in the network is less important than the spread of data redundancy.

There also exists a balancing act between scarcity and popularity of data. A node will always prioritize the data with highest scarcity; however, data that is most popular will be most replicated and thus most readily available, which means that nodes are always prioritizing the data that would take the longest to retrieve. Under normal operations, this problem is minimal and the prioritizing heuristic rule is desirable for the location spread optimization described. However, in extreme cases where a piece of data has not been replicated at all in a local area, there will be long wait times for the nodes to retrieve the whole content being bottlenecked in search of this particular piece.

II. Survey of Related Works

Currently, the content distribution problem in a highly mobile network has been addressed beginning with a protocol, called SPAWN, for advertising "gossip" packets some set hops away from content owners. These packets behaves similarly to trackers in a distributed peer-to-peer bittorrent network. The idea is to maintain and distribute a somewhat up-to-date file that contains information on the potential owners of each piece of a file. An implementation of this protocol was later completed with an application called CarTorrent.

Cartorrent added to the SPAWN gossip protocol by introducing a file manager that was responsible for splitting a large file into pieces and maintaining the reconstruction of these pieces when their data arrives. Additionally, Cartorrent changed the way "rarity" of a piece is determined. In a traditional bittorrent protocol, a piece is shared with higher priority if it is the least owned amongst a group of peers in the peer list of the tracker file. However, because a VANET regards rarity in terms of actual physical location, a simple count of ownership does not represent the actual rarity of a piece of data. Cartorrent decides rarity by hop count.

Later implementations and improvements still fundamentally adheres to the same rules and protocol as Cartorrent and SPAWN. Codetorrent is an improvement made to Cartorrent by exploiting netcoding to piggyback data delivery. However, an end-to-end protocol is fundamentally restricted in a VANET. On the contrary, the named data network architecture allows for more straightforward and intuitive multicasting, which eliminates the need for a complex tracker file, allowing for an even more distributed and localized content distribution. Some of the challenges posed by a VANET have not been ideally addressed in current end-to-end solution.

Previously, CRoWN is a naïve application of content distribution over NDN. However, that project relies on the NDN architecture to perform all of the functions of routing, discovery, and distribution at its native network layer 3. CRoWN exhibits the viability and advantages of multicasting in a NDN VANET. However, because IVAN is an application that implements named data network, this project can use and alter parts of the NDN architecture for efficient routing and multicasting of pieces of files in a distributed content delivery network.

III. IVAN Architecture

In essence, the IVAN protocol takes advantage of the content store of the NDN design to allow forwarding vehicles become localized content delivery servers so that pieces of content become widely distributed across a network. And because these pieces are delivered by their name rather than by their endpoint location, a well maintained tracker or gossip file is not necessary.

A typical operation follows the following procedure:

1) RSUs adds files to the content store cache (CS) by splitting the file into sharable pieces

2) RSUs periodically advertise its content store with its current location

3) Nearby vehicles add advertised information to its forward interest base (FIB)

4) A random Consumer vehicle sends an Interest for a piece of data

5) Interest is forwarded to the RSU by the FIB location information

6) Forwarding nodes add Interest to Pending Interest Table (PIT)

7) RSU respond with Data

8) Forwarding nodes add Data to its CS as it hears Data and forwards depending on PIT

9) Consumer vehicle receives data, adds to file piece received list, then adds data to its CS, and determines next request by distance scarcity as determined by its FIB

10) Forwarding nodes and Consumer vehicle periodically advertise their CS with location

In addition to the NDN components (FIB, CS, and PIT), IVAN has a file manager, advertiser, and responders to manipulate some NDN functions that are specifically beneficial for this application which we will detail.

A. File Manager

The goal of the file manager (FM) is to perform bittorrent file splitting and reassembly tasks as well as determining rarity of data pieces. To this end, the FM maintains two lists, one will include names of names requested and the other will have the names of the data received. These will provide tracking function for candidate data pieces to request and also to determine if the node itself is the consumer and not a forwarder for an incoming data.

A piece of data that is missing from the data received list will be a candidate for sending the Interest request. The FM will check with the FIB to determine distance to the data piece in order to prioritize data request. If a candidate is completely missing from the FIB, its distance is the highest and should be most highly prioritized. Sending the highest priority Interest request will also add the data name to the name requested list.

Typically, because data is routed by names, naming convention for data pieces is vitally important for any application to ensure proper routing and handling by each application of each node. For IVAN (Information-centric Vehicle Ad-hoc Network), a simple generic NDN naming convention can be adopted, which is */ivan/file_name/piece_seq_num*. The file name can even be a unique vehicle id to emulate direct end-to-end communication of an IP routing scheme. Piece sequence number is important for the file manager to reconstruct the file from the pieces received.

B. Advertiser and Advertiser Responder

The Advertiser is used for maintaining the FIB by broadcasting a node's CS. Each node, including both vehicles and RSUs, will periodically advertise its CS by broadcast which will include its physical location. The advertiser is not responsible for maintaining the FIB or the CS. Its sole purpose is to advertise to nearby nodes, nodes who are within broadcast range, about the content it owns.

A design decision was made against CarTorrent's hop count method. On the one hand, by advertising to multiple k hops neighbors, more FIBs will be populated. However, the complexity of maintaining the integrity of the FIB while taking in account for hop counts make this a less desirable feature especially considering that hop counts are not considered for forwarding and routing decision.

The Advertiser Responder will populate the FIB with the most updated location information for each named data piece.

C. Interest Responder

When a vehicle or RSU receives an Interest request, it performs two tasks: check CS for data and make forwarding decision. The CS check is the first step, which checks if the receiving node already owns the data in its cache. If it does, then broadcast the Data with the name and the Interest will be dropped. Otherwise, the Interest may be forwarded.

The Interest forwarding and broadcasting is one of the primary potential for broadcast congestion in a highly dense network. The main challenge is to balance congestion with Interest routing. If forwarding is strict, the mobility of the vehicles may lose the Interest in the network and would require a resending of the interest. If the forwarding is relaxed, congestion will become more significant.

We have implemented a forwarding strategy that is not absolutely strict, which is to forward only if the Interest is new or incurs and update on the PIT. The PIT will be updated only if the the Interest received is a new Interest depending on name that has never been requested by anybody on this receiving node or if the Interest received has already been requested by another node, the PIT entry of that request will be updated by adding the requester to the list associated with the data name. This means that only newly received Interest requests are forwarded. In an extreme case where no FIBs have been populated, then this scheme will naturally resort to broadcast flooding, which is the same as AODV route finding.

D. Data Responder

When a data has been received, the receiver will first determine if it is the requester for the data by checking its data request list. If it is the data requester, the data will be passed to the file manager to add to data responded list and returned to the reconstruction of the file. In both cases if the receiver is or isn't the requester, the data will be saved to the receiver's CS.

Saving to the CS regardless of PIT entry is a deviation from NDN protocol. However, this is to improve data spread in the network at the expense of storage, which can be assumed to be inexpensive in a VANET, and data spread is the essential challenge to content distribution in a VANET. If the receiver is not the requester, then it will check its PIT. If there exist an entry in the PIT, then the receiver is along the breadcrumb path of the requested Interest. The receiver will then remove the entire entry of the PIT and broadcast the data. If there does not exist an entry in the PIT then the data is dropped to reduce congestion and flooding.

E. *Content Store*

The Content Store is a table data structure of names associated with their data. This table represents all of the data that is currently owned and shared by the node. By separating content store from the reconstructed file, a user can potentially manipulate the pieces of data while sharing the file to other users. Data integrity becomes much more robust in the network because revisions do not need to be tracked and utility of the file is improved since pieces are usable as they are received.

F. *Forward Interest Base*

The Forward Interest Base is populated when a node responds to an advertise. It will have the most updated location of a data piece by mapping the piece name to the location last encountered. It is used for Interest forwarding strategy by the Interest Responder and the location data is used to determine distance, which guides priority of Interest delivery for both Interest forwarding and Interest requesting.

G. *Pending Interest Table*

The Pending Interest Table is a map of a data name and a list of its associated requesters. This list is to prevent Interest forwarding loop, which occurs when Interests are broadcasted and overheard amongst a group of nodes.

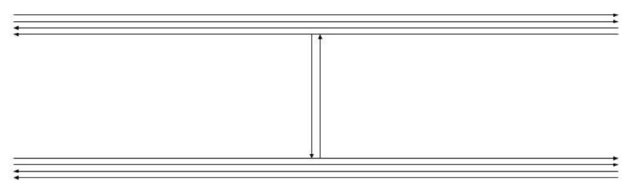
IV. EVALUATION

A. *Simulation Setup*

IVAN was implemented as an application in NS-3 (network simulator). Mobility traces were constructed by the SUMO simulator with 324 mobile nodes of variable random acceleration and deceleration rates. Mobile nodes follow usual traffic behavior such as collision avoidance and right of way. There is 1 immobile node acting as the RSU at the lower T-intersection. Mobile nodes start at the four corners of the roads. The roadmap is shown in Figure 1.

The RSU is assumed to hold the necessary data and data gathering mechanism that is unrelated to the VANET content distribution mechanism. The RSU may hold the entire file that is to be distributed in the VANET or the RSU may be able to generate or request the file from some other infrastructure. Each file or buffer to be sent from the RSU is assumed to be too large for a trivial single transmission. And such a file is then broken into ten pieces by the file manager. The size and number of pieces can and should change depending on a rate adaptation and rate optimization mechanism that are beyond the scope of this paper.

Fig. 1. Vehicles start at the corners of the map moving towards the center. Random vehicles may turn towards the other street.



16 random vehicles were selected to request the file from the RSU even though all the vehicles have the IVAN application installed and can act as data and interest forwarding nodes. 16 were selected to test differences between popular files and unpopular files since some vehicles selected are closer together while others are further apart with some being part of isolated platoons.

Standard wifi ad-hoc network devices were installed into each node to perform broadcasting and default wifi collision detection and avoidance are followed. Propagation delay is a constant set to range so that no data disruption will occur within the range and all broadcast are successful. Also environmental factors do not affect data delivery.

B. *Simulation Result*

Simulation result is gathered per piece of data for each requesting vehicle and is measured by the difference between the time in milliseconds of request and reception.

TABLE I. NS-3 SIMULATION RESULT (IN MILLISECONDS)

Vehicle ID	Piece 1	Piece 2	Piece 3	Piece 4	Piece 5	Piece 6	Piece 7	Piece 8	Piece 9	Piece 10	Average
9	7	512	525	1693	512	2006	1010	1004	2999	501	1076.9
13	1	507	525	1693	512	1506	1503	1004	2999	501	1075.1
30	495	502	525	227	38	15	0	1033	509	7511	371.555556
25	501	507	525	1693	512	1006	2001	506	3002	502	1075.5
27	11007	10	1	52	2	11	3	29	63	3	19.3333333
41	5	17	466	212	11	7	502	474	1033	11359	303
37	11510	2022	279	9	363	7	1	5	2	376	340.444444
45	14632	889	268	455	451	0	2	8	2	1	230.666667
14	11007	1002	201	6419	2	11	84	3033	1063	5	1313.33333
53	0	5	1	11	1	2	6	502	4	25501	59.1111111
75	30500	1	1	5	0	20	66	2001	5	57	239.555556
87	11	4	3	10	2	492	500	3001	3998	34073	891.222222
90	41995	2999	142	1010	10	18	11	8	2	2	466.888889
134	60509	552	6	6	3	1	501	2	5	7	120.333333
133	60000	1000	999	1003	2	5	1500	6	3	3	502.333333
77	4020	7999	38527	7	0	11	39	46	29	73029	5630.88889

Outliers can be seen that are above ten seconds between request and response. These outliers are requests made by vehicles who are not connected within range to the VANET that has the available data piece. In other words, these vehicle nodes have no neighbors who have any routing information for the piece requested. But because of mobility, eventually these vehicles will come into contact with others who can respond with the data. These outliers are not counted in the average because they do not represent the performance of the protocol, but rather they are bottlenecked by the mobility itself. A common contributor to this phenomenon is the extreme case in the simulation in the beginning when no nodes have made contact with the RSU and therefore no advertising broadcast has been heard.

Outside of the extreme outliers, the results show that there are essentially two modes of forwarding performance expectations. The slower mode, which can be seen as pieces taking about two or three seconds, are requests made by nodes who are first establishing the forwarding routes in the VANET. This is the case where there exists a possible route for the interest and data packet. And because of both the distance and the lack of FIB forwarding strategy due to the content store provider being outside of immediate advertising range, essentially the trivial force flood of broadcasting must be performed en route from the consumer vehicle to the producing node. While taking seconds to receive a data piece, the data returned from this first establishment of route provides massive gain in performance for subsequent requests. Also, in an urban environment, the situation where a vehicle is the only node that is the first to request a piece of data should be extremely rare.

The second mode can be seen to be within the tens of milliseconds and some even falling below a single millisecond. This phenomenon occurs when a piece of data has already been requested by another vehicle within a platoon or within very few platoons. When an interest request is routed to a node with the data available in the content store, the data is returned to the requester immediately so that the request does not need to be routed to a sharing vehicle or RSU as would be in the traditional torrent protocol. Content store is essentially a network cache that shares any passing piece of data regardless of the application usage. In this way, nodes that may not have ever requested the data will still participate in sharing of the data.

Therefore, the remarkable phenomenon occurs when only a handful few vehicles need to request for data for that content to have wide distribution in the network. This addresses the issue of popularity in traditional bittorrent scheme where only files that are popular in a network will be widely shared. However, because in a VANET and other mobile networks, popularity depends on location and not based on a simple count on the tracker. Therefore, in accordance with the basic bittorrent principle of sharing the rarest first, it is desirable for a network to have maximized distribution despite a low count in popularity.

V. Conclusion

IVAN is a novel protocol for vehicular content distribution over RSUs and VANET. This protocol uses distributed peer-to-peer bittorrent data piece sharing to enable the delivery of potentially large data files in a disruptive mobile network. To optimize multicasting and routing, IVAN uses NDN over LER at the application level, exploiting and manipulating NDN components to alter design choices most suitable for meeting VANET challenges. IVAN's practice can be generalized to other applications because NDN and other strategies that may be hidden in different layers may be useful at the application layer. However, this practice has the potential danger of a networking layer overstepping its theoretical ideal allowed functionality and may break the ideal separation of powers in each layer.

Aside from contribution to general practices, this protocol is a demonstration of applying information centric networking principles to solve VANET content distribution problems. The delay tolerance problem is resolved by the NDN strategy of local caching during forwarding. Even though this project does not explicitly demonstrate disruption tolerance in its simulation, the NDN caching along the return route of the data serves as delay tolerant forward routers. Should disruption occur, a retransmission timer or NACK can be demonstrated in each forwarding node.

Additionally, by distributing data pieces across the network by the NDN content store, mobility and locality of content distribution is mitigated also by advertising of piece availability by location. Forwarding is also optimized by location information and PIT entry, however, flooding and interest looping is minimized by checking uniqueness. Also because of the multicasting nature of NDN, redundant requests per node is eliminated. All neighboring nodes who wants the same data will be able to receive the same data with a single request.

Interesting future research can involve overlaying SDN over RSUs to coordinate data spread in an urban environment. For example, RSUs can cooperate for maximizing content spread in the network and even improving the rarest-piece-first heuristic.

Data eavesdropping may also reduce congestion at the cost of CS storage space. Nodes that overhear data broadcast can decide to add the data to the CS regardless of PIT or any other factors. It reduces path discovery for new and sparsely distributed networks and potentially new vehicle platoons at the cost of storage.

IVAN is a demonstration of implementing NDN and LER routing at the application layer to allow fine-tuned manipulation to address specific challenges of content distribution in a VANET.

References

[1] J. s. Park, M. Gerla, D. S. Lun, Y. Yi and M. Medard, "Codecast: a network-coding-based ad hoc multicast protocol," in *IEEE Wireless Communications*, vol. 13, no. 5, pp. 76-81, October 2006.

[2] K. C. Lee, S. Lee, R. Cheung, U. Lee, and M. Gerla, "First Experience with CarTorrent in a Real Vehicular Ad Hoc Network Testbed," In Proceedings of 2007 Mobile Networking for Vehicular Environments, pp. 109-114, Anchorage, AK, May 2007.

[3] Mario Gerla, Chuchu Wu, Giovanni Pau, Xiaoqing Zhu, Content distribution in VANETs, Vehicular Communications, Volume 1, Issue 1, January 2014, Pages 3-12, ISSN 2214-2096, http://dx.doi.org/10.1016/j.vehcom.2013.11.001.

[4] M. Amadeo, C. Campolo and A. Molinaro, "CRoWN: Content-Centric Networking in Vehicular Ad Hoc Networks," in *IEEE Communications Letters*, vol. 16, no. 9, pp. 1380-1383, September 2012.

[5] Shirshanka Das, Alok Nandan, and Giovanni Pau. 2004. SPAWN: a swarming protocol for vehicular ad-hoc wireless networks. In *Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks* (VANET '04). ACM, New York, NY, USA, 93-94.

[6] U. Lee, J. S. Park, S. H. Lee, W. W. Ro, G. Pau and M. Gerla, "Efficient peer-to-peer file sharing using network coding in MANET," in *Journal of Communications and Networks*, vol. 10, no. 4, pp. 422-429, Dec. 2008.

[7] Y.-T. Yu, "Scalable Opportunistic VANET Content Routing with Encounter Information", *IEEE Int'l Conf Network Protocols*, pp. 1-6.