IP Address Passing for VANETs

Todd Arnold
Department of Computer Science and Engineering
Pennsylvania State University
University Park, PA 16802
tarnold@cse.psu.edu

Wyatt Lloyd
Department of Computer Science
Princeton University
Princeton, NJ 08540-3233
wlloyd@princeton.edu

Jing Zhao, Guohong Cao
Department of Computer Science and Engineering
The Pennsylvania State University
University Park, PA 16802
{jizhao, gcao}@cse.psu.edu

Abstract

In Vehicular Ad-hoc Networks (VANETs), vehicles can gain short connections to the Internet by using wireless access points (AP). A significant part of the connection time is the time required for acquiring an IP address via Dynamic Host Configuration Protocol (DHCP). Depending on a vehicle's speed and the AP coverage area, DHCP can consume up to 100 percent of a vehicle's available connection time. We propose the IP Passing Protocol to reduce the overhead of obtaining an IP address to under one-tenth of a second. This is done without modifying either DHCP or AP software. We explore scalable implementations and describe the dynamics of the IP Passing Protocol. We also show our protocol will significantly improve efficiency, reduce latency, and increase vehicle connectivity.

1 Introduction

Wireless communication is being used to provide services for users in their vehicles, and VANETs can be leveraged to provide a wide range of services to people on the go. Traditional Internet services can be provided; combining GPS feedback from VANET users can provide up to date traffic information, or even information for vehicles to use for collision avoidance [12, 8, 13, 14]. Based on vehicular feedback, driving patterns can also be recorded for more accurate traffic analyses [5].

Integrating VANETs into current networks presents a new set of challenges. Due to the speed at which vehicles travel, they quickly move into and subsequently out of an AP's coverage range. Bychkovsky *et al.* [1] discovered that at city driving speed, after a vehicle associates with an AP and acquires an IP address, common connection times range from 5 seconds to 24 seconds. However, the WiFi DHCP often requires two or three seconds once association is complete. A vehicle's usable connection time can be considerably increased by reducing IP acquisition time and overhead.

Our solution allows vehicles to pass IP address information backwards geographically, relative to the direction of travel. For example, as node A leaves an APs coverage area, node B, who is behind node A, will reuse node A's IP address to access the Internet via the same AP. This process involves three main steps: gathering the IP information, passing the IP from node A to node B, and configuring node B's interface on the fly.

Our solution provides multiple improvements for the performance of VANETs. Based on these improvements, this paper has three main contributions:

1. We reduce the average IP acquisition latency to less than one-tenth of a second and significantly reduce the network overhead, extending the overall connectivity time by two to three seconds.

2. Our solution is backward compatible with existing infrastructure because no AP modification is required.

3. This is a novel concept because no one has examined how to extend coverage time of nodes in VANETs from a DHCP perspective.

We also propose a number of IP Passing algorithms for choosing the trailing vehicle that will receive the passed IP address. Our algorithms apply to both structured and unstructured VANETs, and these algorithms are evaluated based on efficiency, latency, and connectivity.

The rest of the paper is organized as follows. Section 2 is an overview of background material. Section 3 presents our test bed and results. The algorithms for IP Passing will be presented in Section 4. Section 5 provides some analytical results on these algorithms. Section 6 presents related work and Section VII concludes the paper.

2 Background

We make two assumptions about the equipment and capabilities of the computers in each vehicle. Our first assumption is each vehicle has at least one wireless interface

card and the interface is capable of listening in promiscuous mode. We also assume that each vehicle has a GPS receiver for identifying its own location. For our algorithms and analysis of structured VANETS we also assume a node knows its neighbors' locations. Next, we provide background information on DHCP and the Address Resolution Protocol.

DHCP provides IP addresses to nodes when they join a network. A DHCP server is responsible for maintaining a pool of addresses and issuing the IPs, through leases, to clients for a specified amount of time. The process for acquiring a DHCP lease entails four messages. The DHCP Discover message, sent as a broadcast from a client in need of a lease, is the initial contact between the client and server. When a DHCP Discover message is received, the server checks its IP pool for available addresses and issues a lease to the client with a DHCP Offer message. The client accepts the offer by sending a DHCP Request message as a response. If the offer is still valid, the server sends a DHCP ACK. The client is not able to use the lease information until it receives the DHCP ACK [2]. All client messages are broadcast messages, while the server's are unicast. When a client is prepared to leave a network, it sends a DHCP Release message to the server so the server can delete the lease information and place the IP back into its pool of available addresses [2].

ARP is responsible for mapping protocol addresses to hardware addresses and does so using a simple request–response implementation. To retrieve an unknown hardware address, a node broadcasts an ARP Request specifying the IP address it wants to contact. The other nodes in the network examine the request to determine if they are the subject of the inquiry. If a node determines it is the target, the node responds with an unicast ARP Reply, mapping the missing hardware address to its protocol address [11].

Each node maintains a list of ARP entries known as an ARP cache. It is possible for a node to update its ARP information in the ARP cache of other nodes using a Gratuitous ARP (GARP) message. The GARP message is an ARP Request where the source and destination IP are identical. Other nodes who have an ARP cache entry for that particular IP address update the entry with the MAC address included in the GARP.

3 Implementation

3.1 Overview

In order to pass an IP address, a node needs the ability to know when it should pass the IP address to another node. By setting a minimum Signal to Noise Ratio (SNR) threshold, a vehicle is able to determine whether or not it has an adequate connection to the AP. Similar to understanding when to switch access points [6], maintaining a minimum SNR threshold allows a node time to attempt passing the IP and still have time to send a DHCP Release message, if required. Whether to release or pass an IP will be discussed in detail in Section 4.

As illustrated in Section 1, multiple pieces of information are required before node B can utilize the IP address it receives from node A. In addition to the address itself, node A must provide the subnet mask and the network's default gateway. This is the minimum amount of information required for node B to communicate with the AP.

Once associated, node B can immediately begin using the passed IP. However, the AP needs to be informed of the new MAC address which is associated with the IP address. Typically, nodes maintain an ARP cache to map IP addresses to MAC addresses [11]. A Gratuitous ARP (GARP) message can be used to update ARP cache entries in other nodes.

The next improvement is to include the AP's Extended Service Set ID (ESSID). Because several APs may operate in a given area, the inclusion of the ESSID allows for the receiver node to immediately associate with the correct AP once it is within range. To check for associativity, we operate under the assumption that the MAC address of the default gateway and the AP are the same.

We also include the GPS coordinates for the location where the passing vehicle began associating with the AP. We define this location as the *association point*. This point is required for the IP Passing algorithms. We assume that the GPS latitude and longitude information are in the format of two 32 bit integers.

Once the information is gathered, we marshal it into an IP Passing packet to forward (format shown in Figure1). We utilize the ioctl system call to retrieve and set all relevant information, so we use binary rather than ASCII for our packet format. Our packet has a payload range of 30 to 62 bytes; all our fields are a fixed length except the ESSID, which is bound by a maximum length of 32 bytes. To minimize network overhead, we use a MAC layer broadcast to avoid layer 3 and 4 headers. There is no IP or port information for distinguishing our packet from any other broadcast, because we send a MAC layer broadcast. Therefore, we include a four byte magic cookie in our packet format.

The receiver listens for broadcast packets in promiscuous mode because it may not be on the same network as the node passing the IP. When it receives a packet it parses the data of each broadcast for the magic cookie, configures each parameter, associates with the AP, and finally sends out the GARP to update the AP's ARP cache. The GARP message also acts as an acknowledgment to node A. At that point, the receiver is ready to pass traffic.

0	Bits	31
Check Sequence		
Forwarded IP Address		
Forwarded Subnet Mask		
Default Gateway IP Address		
Default Gateway MAC Address		
Default GW MAC Addr. (cont.)	GPS Information	
GPS Information (cont.)		
GPS Information (cont.)		
ESSID 0-32 bytes		

Figure 1: IP Passing Packet Format

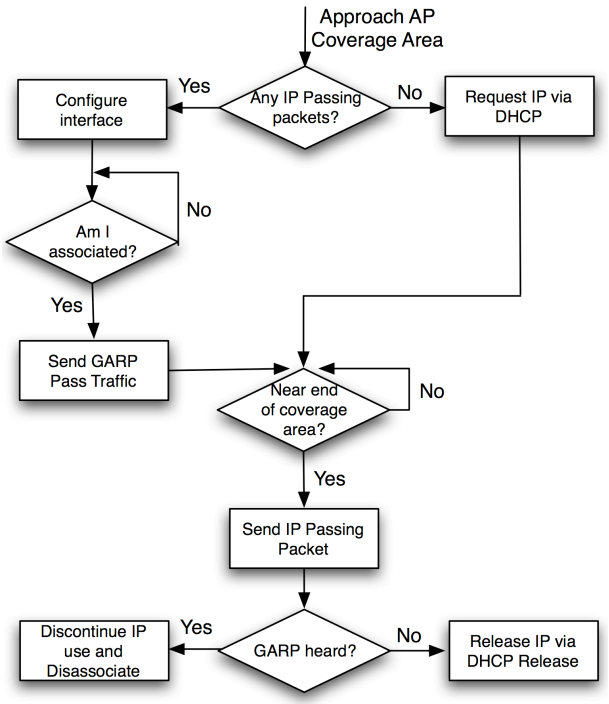

Figure 2: The basic flow of information during our distributed implementation of an IP Passing transaction. The initial vehicle entering an AP coverage area will have to request an IP address and all subsequent vehicles can share the IP. The trailing vehicle will listen for IP Passing packets before requesting a lease.

3.2 Equipment

For our experiments, we use a Linksys WRT54GL router as our AP. Our two nodes run Redhat Linux, with the 2.4.25 kernel, and network monitoring is performed by an Apple PowerBook running OS X 10.4.9. We use Ethereal to capture packets. Ethereal is an open source software package that can capture, display, and analyze all packets an interface is capable of hearing. We used Ethereal 0.10.12-1011.

By default, our AP only allowed us to view the issued DHCP leases. In order to verify our experiments, we were required to view the AP ARP cache to ensure it was properly modified following the GARP message. Although our implementation requires no modification of the AP, it was necessary to change our AP's operating system for verification purposes. We installed DD-WRTv23 SP3 on the AP to allow telnet access and to view the ARP cache.

3.3 Observations

Prior to implementing our solution, we need to determine what normal behavior is for an AP. We were intrigued by the thought an AP required two to three seconds to issue an IP address, so we wanted to view all messages passed during the transaction. We observed DHCP transactions on different APs, and found some variation in the methods for issuing leases. On our test system, the Linksys WRT54-GL, after the DHCP Discover message (Figure3 Packet 1) was received, the AP determined which IP to issue and performed three ARP requests (Figure3 Packets 2 – 4) to determine whether that IP was already in use. After the third negative response, the remaining steps of the DHCP transaction were completed (Figure3 Packets 5 – 7). The total time was approximately 2.5 seconds. This sequence of events was replicated during testing on two additional Linksys brand APs. We also looked at another brand, Apple Airport Express. Here an IP was issued after only one failed ARP request. This sequence took less than 1 second, as seen in Figure4.

We define Traditional DHCP as the sequence of events based on the Linksys implementation of DHCP. We define Apple DHCP as the sequence of events based on the Apple implementation of DHCP. Traditional DHCP requires 2.5 seconds to complete; when a vehicle is far from the AP and does not have a reliable connection, this time could possibly increase.

3.4 IP Passing

Our IP Passing implementation simulated a distributed system where neither node had prior knowledge of its neighbors. Node A has to gather all of the necessary information, format a broadcast, and transmit the data to node B. Node B has to retrieve the information, configure its interface, insert a default route and an ARP cache entry for the default gateway, associate with the AP, and send the GARP. Each step is depicted in Figure6, and the decision making process in depicted in Figure2.

Each node's initial setup is a blank network configuration. This simulates both nodes traveling along the road, searching for an AP to associate with, and having no knowledge of each other. The first step is for node A to associate

Packet #	Elapsed Time	Source	Destination	Protocol	Bytes	Information
1	0	0.0.0.0	255.255.255.255	DHCP	428	DHCP Discover - Transaction ID 0x4c08f26e
2	0.059079	00:18:39:ea:5f:02	Broadcast	ARP	128	Who has 192.168.1.100? Tell 192.168.1.1
3	1.078637	00:18:39:ea:5f:02	Broadcast	ARP	128	Who has 192.168.1.100? Tell 192.168.1.1
4	1.999697	00:18:39:ea:5f:02	Broadcast	ARP	128	Who has 192.168.1.100? Tell 192.168.1.1
5	2.495476	192.168.1.1	192.168.1.100	DHCP	428	DHCP Offer - Transaction ID 0x4c08f26e
6	2.497751	0.0.0.0	255.255.255.255	DHCP	428	DHCP Request - Transaction ID 0x4c08f26e
7	2.504289	192.168.1.1	192.168.1.100	DHCP	428	DHCP ACK - Transaction ID 0x4c08f26e

Figure 3: Capture of a Traditional DHCP transaction on a Linksys brand AP after association has completed. We can see that once the initial Discover message is sent, it takes two and a half seconds for the DHCP server to respond with a DHCP Offer message.

Packet #	Elapsed Time	Source	Destination	Protocol	Bytes	Information
1	0	0.0.0.0	255.255.255.255	DHCP	342	DHCP Discover - Transaction ID 0xd4e6c607
2	0.114077	00:14:51:6a:be:fb	ff:ff:ff:ff:ff:ff	ARP	42	Who has 10.0.1.3? Tell 10.0.1.1
3	0.500085	10.0.1.1	10.0.1.3	DHCP	590	DHCP Offer - Transaction ID 0xd4e6c607
4	0.500987	0.0.0.0	255.255.255.255	DHCP	342	DHCP Request - Transaction ID 0xd4e6c607
5	0.502556	10.0.1.1	10.0.1.3	DHCP	590	DHCP ACK - Transaction ID 0xd4e6c607

Figure 4: Capture of Apple's DHCP transaction after association has completed, on an Airport Express. This implementation of DHCP eliminates two ARP Requests compared to the Traditional DHCP.

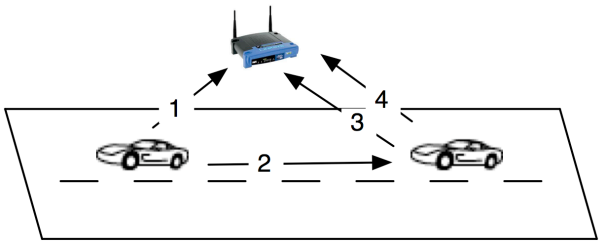

Figure 6: The four steps of an IP Passing transaction. Vehicle A obtains a DHCP address (1) and passes the IP to Vehicle B (2). Vehicle B associates with the AP (3) and broadcasts a GARP to the AP (4).

and perform a traditional DHCP request sequence. As node A continues traveling, it reaches the point where it no longer needs its IP, so it forwards the IP to node B which is just about to enter the range of the AP and is not yet associated (step 2). Node B receives the IP Passing packet and begins association with the AP (step 3). It parses the information and configures all relevant settings in preparation for when it is associated with the AP. Once Node B is associated, it sends the GARP as the final step to update the APs ARP cache. Figure5 depicts the IP Passing process from Node B's perspective. Packet 1 is a received IP Passing packet. In under one-tenth of a second from receiving the IP Passing message, node B transmits its first association request to the AP. Association requires a couple seconds, but packet 3 and 4 show node B is able to pass traffic within one-tenth of a second after association is completed.

3.5 Analyses

Table 1 provides a direct comparison of two typical DHCP implementations and IP Passing in terms of time, size in bytes, and number of messages required. Our implementation of the IP Passing Protocol reduces the amount of overhead required for acquiring an IP address from 2000 bytes to under 300.

Implementation	Time	Bytes	# of Messages
Traditional DHCP	2.5 s	2096	7
Apple DHCP	0.5 s	1906	5
IP Passing	0.09 s	296	2

Table 1: A comparison of all three test bed implementations for acquiring an IP address. The bytes for IP Passing represents the maximum possible amount.

In our implementation, the ESSID for the AP was only four bytes long. If the ESSID were the maximum length, then our message would require 296 bytes. Even at the maximum size, IP Passing results in at least an 84 percent reduction in network overhead for IP acquisition.

Figure7 shows the new acquisition time in reference to the time-line reported in [1]. We can see that the acquisition time is moved significantly to the left, allowing a longer connection time by reducing the time to acquire an IP address compared to traditional DHCP.

4 Algorithms for Passing IPs

The following subsections present algorithms for choosing whom the IP address is passed to and when to release it. We assume that if a node does not receive an IP address through IP Passing, that node will attempt to use DHCP for obtaining one. This is necessary when a node has no geographically forward neighbors or when its geographically forward neighbors do not pass it an IP promptly.

We present two optimal algorithms based on awareness of immediate neighbor topology first, followed by more realistic distributed algorithms without knowledge of the neighbor topology. We define all of our parameters in Table

Packet #	Elapsed Time	Source	Destination	Protocol	Bytes	Information
1	0.000000	Agere_b6:34:9e	Broadcast	IEEE 802.3	128	Source port: picknfs Destination port: picknfs
2	0.078179	D-Link_d5:a9:dc	Broadcast	IEEE 802.11	104	Probe Request SSID: "598b[Malformed Packet]"
...						Association Process
3	1.997938	00:18:39:ea:5f:04	D-Link_d5:a9:dc	IEEE 802.11	122	Association Response[Malformed Packet]
4	2.013008	D-Link_d5:a9:dc	Broadcast	ARP	160	Who has 192.168.1.122? Gratuitous ARP

Figure 5: The packet capture for an IP Passing transaction. The trail vehicle receives an IP Passing packet, Packet 1, and immediately attempts to associate with the AP. Almost immediately after associating, the vehicle is able to pass traffic. The number of non-association related packets is reduced from 7 to 2, with a significant reduction in overhead.

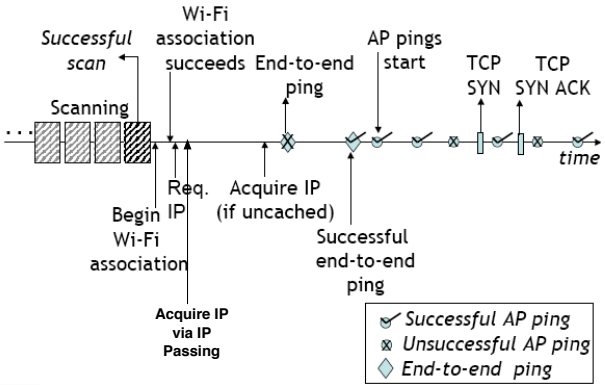

Figure 7: The time-line for of network activity, with the new IP acquisition time via IP Passing depicted.

Param	Description
r	The communication range of vehicles
R	The length of road covered by an AP
d	The physical distance between any two cars
D	The distance between the vehicle that is forwarding an IP and the receiving vehicle
v	The velocity of the cars
t_{DHCP}	The time required to obtain an IP address via DHCP
t_{pass}	The time required to send a passing message
t_{GARP}	The time required to send a GARP
$t_{inrange}$	The time a node is in range of the AP before it is passed an IP address
t_{expire}	The time it takes for a DHCP lease to expire

Table 2: IP Passing parameters determining the efficiency of the algorithms.

2.

The main parameter of concern is D, which we define as the physical distance between the vehicle currently in possession of the IP address and the vehicle that will receive the IP for later use. We are also concerned with d, which is the distance between any two vehicles.

4.1 Algorithms with Neighbor Topology Awareness

A car benefits the most from IP passing when it receives the IP before being associated with the AP. So the proposed algorithms aim at delivering the IP address to a car before it is associated and with minimum overhead. A car knows the location where cars behind itself will be associating with the AP, because this can be inferred from where it became associated (*association point*). So, a car leaving the AP area would deliver its IP to the cars near the association point and moving in the same direction.

4.1.1 One-hop IP Passing

With the knowledge of the one-hop neighbor topology, a node knows its neighbors' locations and directions of travel. In addition, a node knows a unique name for all its neighbors and includes this name explicitly in the passing message. We first consider the case where the association point is within one-hop coverage of a car leaving the AP, so IP passing is limited to one hop. At the end of this subsection we relax the one-hop assumption. The distance between the passing car and the receiving car, D, is important for evaluating the algorithms and is examined with each algorithm.

Farthest Neighbor: A simple solution would be to pass the IP address to the one-hop neighbor farthest from itself (FN). The idea is that the farthest neighbor is most likely outside the AP coverage area and would benefit from receiving an IP address earlier.

To compute the distance between two cars involved in IP passing, assume the passing node has n neighbors and the distance between itself and neighbor i is d_i, let r be the range of the node, then

$$D_{FN} = \max_{1 \le i \le n} \left\lfloor \frac{r}{d_i} \right\rfloor * d_i \quad (1)$$

Nearest Neighbor behind Association Point: The Nearest Neighbor behind Association Point (NNb) algorithm selects the neighbor nearest the association point of the AP without having entered the coverage area. The idea is to pass the IP address to a node immediately prior to the node associating with the AP, so the node can use the IP address immediately. Nodes have knowledge of their neighbors, and this includes the location of their neighbors.

Nodes also know where they associated with the AP and take that location as the association point.

For this algorithm, the distance between passing cars is the distance from the node in possession of the IP to the neighbor closest to the association point without passing that point. Let l_p be the location where a car starts to pass its IP, and l_a is the location where a car starts to associate with the AP.

$$D_{NNb} = \min_{1 \le i \le n} \left\lceil \frac{|l_p - l_a|}{d_i} \right\rceil * d_i \quad (2)$$

4.1.2 Releasing

When there are no nodes in need of an IP address, a node releases the address, otherwise, the AP will be unable to issue the address again until a very long (typically hours or days) timeout has occurred. Because the neighbor information is known, a passing node will always know whether it should pass the IP address to a neighbor. If the leaving node cannot find a neighbor to benefit from the IP Passing, it releases the IP so the AP is able to reissue it.

Releasing the IP does not occur instantaneously, it requires the node to send a DHCP Release message before it leaves the AP's range. Let $t_{release}$ be the time it takes for the node to successfully send the DHCP release message. The node must start sending the message $t_{release}$ seconds or $d_{release} = t_{release} * v$ meters before it exits the range of the AP. The node's current position, v, $t_{release}$, and the edge of the AP's range are known, so $d_{release}$ can be calculated.

4.1.3 Multi-Hop IP Passing

If the association point is beyond one-hop distance of the leaving node, which is likely if the AP is deployed near the road, then the algorithms would have to select one or more intermediate nodes to forward the IP. With the assumption that a node knows its neighbor information, the IP address can be passed through multiple hops with the help of the two algorithms we discussed above.

When a node determines that the association point is outside its communication range, the IP address may be passed to a node farther than its immediate neighbors. The leaving node cannot know who will eventually receive the IP. Therefore, instead of specifying the destination node ID, it only specifies the intermediate node in the message header. It uses the FN algorithm to pick the farthest neighbor as the next hop, and specifies that node as the intermediate node to forward the message. The intermediate node, upon receiving the message, will check whether the association point is within its communication range. If true, it can use either FN or NNb to deliver the message as the final step; otherwise, it continues to forward the message to the next hop using the above protocol.

If any intermediate node cannot find a proper neighbor as the next hop, it will release the IP address to the AP.

4.2 Distributed Algorithms without Neighbor Topology Awareness

For a more realistic VANET environment, we assume a node knows nothing about the surrounding network topology. It may have many neighbors of which it is unaware and cannot choose the neighbor to pass the IP address to. Thus, in this section we provide a method to approximate the algorithms used in the previous section in a distributed manner. The method for approximating the algorithms within one hop are presented first, then a discussion on how to extend to multiple hops is provided.

4.2.1 One-hop IP Passing

The basic idea of the distributed algorithms is the passing node sends the reference position of a node to pass to, either explicitly or implicitly. Nodes in the network then wait an amount of time proportional to their distance from this reference point before broadcasting a GARP. While waiting to send the GARP, nodes listen promiscuously for other nodes sending a GARP to claim the IP address. If a node hears another node claim the IP address it does not send a GARP.

When nodes send a GARP, they must listen for collisions. If a collision occurs the nodes use exponential backoff.

It is possible for one node to claim an IP address without another node overhearing the claim. This could occur when the claiming node sends the GARP and there is a collision at the other node. Solving this problem is part of future work. A potential solution is when a collision occurs between the passing message and when a node wants to claim an IP address it must first ARP for that address.

Farthest Neighbor:

The FN without topology information (FN_u) algorithm emulates FN by basing the waiting time on the distance from the passing node. This could be accomplished by the passing node including its current location in the passing message rather than the association point. This incurs no additional overhead. The distance could be implicitly determined based on the strength of the signal of the passing message; however, this solution adds some complexity and ensuring these signal strength values from wireless cards of different model to be comparable is not always possible. So we add the location of the passing node, the *reference position*, in the passing message.

The passing node broadcasts the IP Passing message when it is about to leave the AP coverage area. All nodes who hear the message but do not yet have an IP address will

broadcast a GARP message after a specific delay $\frac{\delta}{|l_r - l_i|}$, where l_r is the reference position, l_i is the location of the ith neighbor, and δ is a constant to adjust the delay to a more reasonable value. With this formula, the node farther away from the passing node will broadcast the GARP message earlier. The GARP message confirms that its sender has taken the IP address. So other nodes who overhear the GARP messages will abort their own GARP broadcast. This process ensures the farthest neighbor will eventually takes the IP address passed by the leaving node.

Nearest Neighbor behind Association Point:

Similar to the FN_u, the distributed equivalent of the NNb algorithm (NNb_u) bases the waiting time on distance from the association point. The location of the association point is written in the passing message field by the leaving node, and only the nodes behind the association point are allowed to claim the IP address. The protocol to claim the IP is similar to FN_u, except the reference position included in the passing message is the association point, and the nearest neighbor to that position has the shortest waiting time, so it is the first to reply.

4.2.2 Releasing

In the distributed approach, if a node simply sends a passing message, there is no guarantee the corresponding IP address will be used. For instance, this happens when a passing node has no neighbors or no neighbors that need an IP address. If a node blindly passes its IP address, the pool of available addresses at the AP will quickly dwindle to zero. To prevent this, an ACK message is required to tell the leaving node whether the passing IP is taken. But because a GARP message is broadcast from whoever takes the IP address, this message can serve as the ACK, adding no extra overhead.

However, as an ACK is required, a passing node need to wait for an ACK message or a timeout if no one replies. The passing node must attempt to pass its IP address earlier than in the previous section. It must ensure that after a timeout, it can still communicate with the AP and release the IP address. The length of the timeout, $t_{timeout}$, should be long enough that all nodes have a chance to claim the IP address.

To compute the IP passing distance (the distance between two nodes passing the IP address) for the distributed algorithms (D_u), we should take the above factors into account. So D_u can be computed by revising the distance formulas of the equivalent algorithms with the neighbor topology awareness as below:

$$D_u = D - (t_{release} + t_{timeout}) * v \tag{3}$$

4.2.3 Multi-Hop IP Passing without Neighbor Topology Awareness

The multi-hop IP passing protocol discussed in the previous section can be easily adapted to fit in the distributed approach, because of its distributed nature.

Without neighbor topology information, the distributed algorithm can simply use FN_u and NNb_u wherever FN and NNb were used in the previous section.

The only problem for implementing this solution is the intermediate node does not broadcast GARP message as the final destination. The intermediate node can still acknowledge the receipt of the IP passing message implicitly because it will rebroadcast the message. This does not add extra message overhead, nor force the passing node to start passing the IP earlier than one-hop passing.

5 Algorithm Analyses

IP address passing should efficiently use IP addresses and decrease the latency to connectivity for vehicles in a VANET. The use fraction and average distance used metrics evaluate the efficiency of the algorithms. The average latency to connectivity metric is used to compare the latency of the algorithms.

In addition, IP address passing should scale well and allow as many cars as possible to connect to the Internet. The denied request fraction metric is used to evaluate the scalability of the algorithm.

The following analysis assumes cars are evenly spaced d meters apart.

5.1 Use Fraction

The use fraction (u) is the fraction of the time an issued IP address is utilized. The higher the use fraction the more efficient the use of the address. When only using DHCP, the use fraction is the time that a DHCP lease is used divided by the time until it expires, t_{expire}.

$$u_{DHCP} = \frac{\frac{R}{v} - t_{DHCP}}{t_{expire}} \tag{4}$$

With a car traveling at a reasonable speed and t_{expire} often in hours or days, u_{DHCP} is very low. For an extreme example, consider a car traveling 5 meters a second (about 10 MPH), an AP with road range of 200 meters, a t_{DHCP} of 4 seconds, and a t_{expire} of 1 hour. Then $u_{DHCP} = .001$. In a real scenario a car would likely be traveling faster and t_{expire} would be much longer.

With all IP Passing algorithms, the use fraction is based on the distance between passing nodes, D.

$$u = \begin{cases} \frac{R}{D} & D > R \\ 1 & D \leq R \end{cases} \tag{5}$$

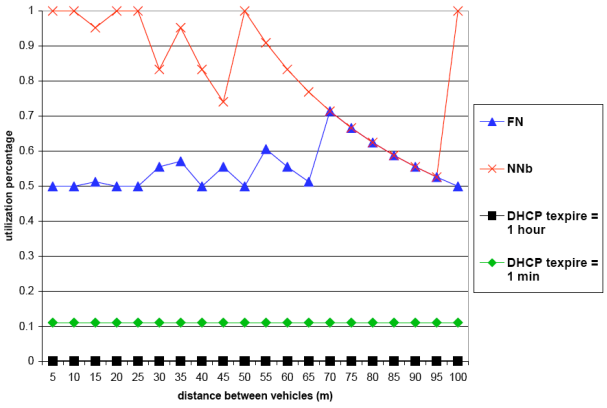

Figure 8: The use fraction depicts the percentage of time an IP is used before the lease times out. For Traditional DHCP, leases are very long and the use fraction is very low. With IP Passing, the use fraction can be dramatically improved.

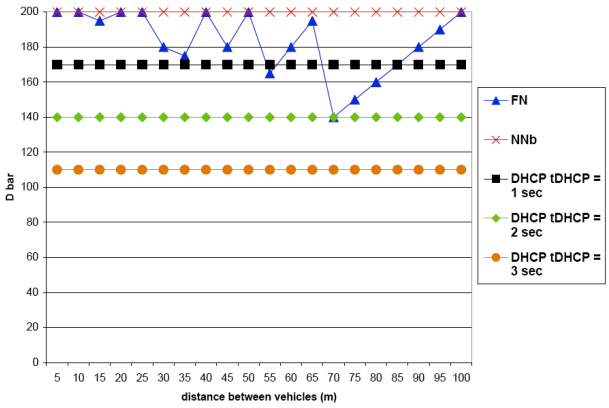

Figure 9: D bar ($\bar{D}$) is the distance a node covers while using an IP within range of an AP. DHCP cannot achieve the full coverage area due to the time required to obtain a DHCP lease.

Figure 8 lets $R = 100m$, $r = 200m$, $v = 30m/s$, and $t_{DHCP} = 0$; these parameter values are all realistic or biased in favor of pure DHCP. It is clear all the algorithms perform significantly better than pure DHCP, particularly when d is small. Overall, the NNb algorithm has the highest use fraction because the goal is to pass to a node that is close to the range of the AP. The FN algorithm also performs particularly well, significantly higher than DHCP with very low expiration leases.

5.2 Average Distance Used

The average distance used, $\bar{D}$, is the distance traveled by a node with a usable IP address, while within range of the AP. Distance covered while in possession of an IP but out of range the AP is not counted.

When only using DHCP, the average distance used is simply the range of the AP less the distance needed to obtain the DHCP lease.

$$\bar{D}_{DHCP} = R - t_{DHCP} * v \tag{6}$$

For the IP Passing algorithms, the average distance used is the same as D except it can be no greater than the range of the AP R.

$$\bar{D} = \begin{cases} R & D > R \\ D & D \leq R \end{cases} \tag{7}$$

Figure 9 lets $R = 200m$, $r = 200m$, $v = 30m/s$. When the distance between cars is small all the algorithms perform significantly better than pure DHCP. For this metric, the NNb algorithm performed the best.

Whenever $D > R - t_{DHCP} * v$ the IP passing algorithm will be more efficient in terms of average distance used than pure DHCP.

Equation 7 shows that $\bar{D}$ is maximized when $D \geq R$ and equation 5 shows that u is maximized when $D \leq R$. Thus the overall efficiency is maximized when $D = R$. The algorithm with D closest to R would maximize both metrics of efficiency.

We define efficiency as $u * \bar{D}$. Given a realistic scenario where $R = 200m$, $r = 200m$, $d = 5m$, $v = 30m/s$, $t_{DHCP} = 2s$, and $t_{expire} = 1hour$, both algorithms are over 700 times more efficient than traditional DHCP.

5.3 Average Latency to Connectivity

The average latency to connectivity, lat, is the amount of time from when a node enters the range of the AP until it has a usable IP address. When using only DHCP $lat = t_{DHCP}$. The lat for the IP passing algorithms is equal to the time it takes to pass the IP address, t_{pass}, send the GARP, t_{GARP}, and the time the node is in range before it receives the passed IP address, $t_{inrange}$. Note that $t_{inrange}$ can be negative, but can only offset t_{pass} because the GARP must be sent while in the AP's range. Thus $t_{inrange} = max(-t_{pass}, \frac{R-D}{v})$.

$$lat = t_{pass} + t_{GARP} + t_{inrange} \tag{8}$$

Figure 10 lets $R = 200m$, $v = 30m/s$, $t_{pass} = t_{inrange} = 100\mu s$; these parameter values are all realistic. It is important to note the IP Passing algorithm would effectively be limited by the latency of pure DHCP, because nodes use DHCP to obtain an address when no address is passed to them.

Both the algorithms perform significantly better than pure DHCP when d is small. The NNb algorithm performs the best, since it attempts to pass IPs to vehicles before they enter the coverage area. This effectively shows our algorithms can dramatically reduce the latency to connectivity.

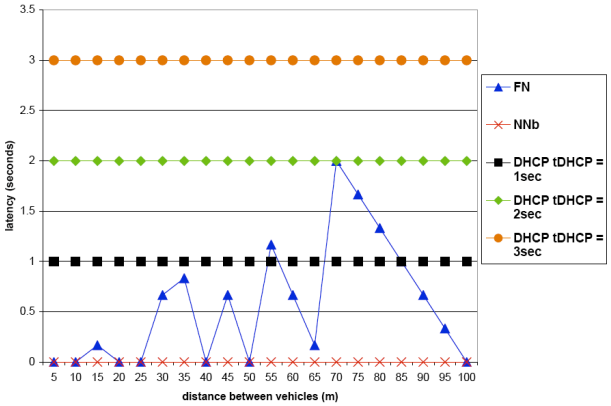

Figure 10: The average time from when a node enters the range of an AP until it has a usable IP address. DHCP is always a fixed amount; while IP Passing can vary, it can obtain an IP address in under one-tenth of a second.

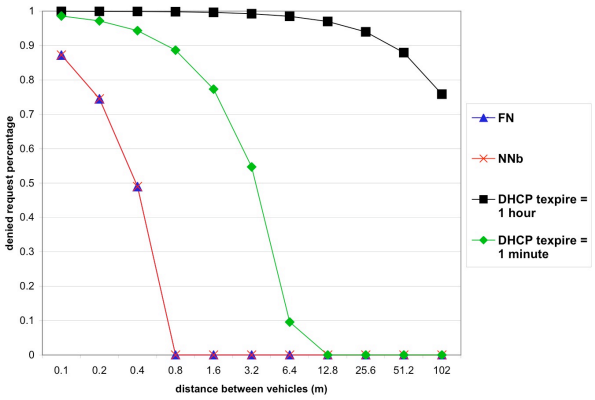

Figure 11: Denied request fraction is the percentage of DHCP requests an AP cannot serve. Due to the length of DHCP leases, a DHCP pool can become saturated very quickly. With IP Passing, only the minimum amount of IPs required to satisfy all vehicles passing through the AP coverage area will be used.

This is especially important in scenarios where connectivity is very brief, such as accessing home WiFi networks from the highway.

5.4 Denied Request Fraction

The denied request fraction, dr, is the fraction of DHCP Requests an AP can not serve because it has issued its entire pool of addresses. This can occur when there are more nodes in range of the AP than IP addresses. This can also occur when nodes leaving the range of the AP do not pass or release the IP address, as in [1].

When only using DHCP, the number of requests for an IP per t_{expire} is the number of cars that enter the range of the AP. Let n_{pool} be the total number of leases an AP has to distribute. Thus for only DHCP the dr is given by equation 9.

$$dr = max\left(0, \frac{\left\lceil \frac{v*t_{expire}}{d} \right\rceil - n_{pool}}{\left\lceil \frac{v*t_{expire}}{d} \right\rceil}\right) \quad (9)$$

For the IP address passing algorithm, the denied request fraction is zero whenever the number of IP addresses available is greater than the number of cars within range of the AP. Thus for the IP passing algorithms the dr is given by equation 10.

$$dr = max\left(0, \frac{\left\lceil \frac{max(R,D)}{d} \right\rceil - n_{pool}}{\left\lceil \frac{max(R,D)}{d} \right\rceil}\right) \quad (10)$$

Let $R = 200m$, $r = 200m$, $v = 30m/s$, $n_{pool} = 255$. In Figure 11, the IP passing algorithms mirror one another. When d is very small, $D \approx R$, so $max(R, D) \approx R$ for all the algorithms.

The IP Passing algorithms dramatically outperform even pure DHCP with a lease time of one minute. When compared to a more realistic pure DHCP with a lease time of one hour all the algorithms provide over 1000 times more connectivity to passing vehicles.

6 Related Work

The CarTel paper series discusses the usage of in-situ networks, DHCP connection times, and DHCP lease caching [5] [1]. During their data collection phase, Bychkovsky et al. discovered that DHCP leases require on average 2.5 seconds to obtain after a node is associated with an AP [1]. Their solution was to cache IP address leases for reuse when accessing the AP at a later time [1]. However, several issues arise when caching leases. For example, most commercial APs have limited IP addresses. Caching may not always be possible, because the cached lease may expire before the vehicle returns several hours later. Furthermore, this is of no help to vehicles passing an AP for the first time. IP Passing will provide substantial gains in performance by providing AP coverage area information for association as well as eliminating the overhead of DHCP, after the initial acquisition.

Most work on DHCP for VANETs focuses on the distribution of IPs amongst the vehicles, not between the vehicles and the AP [10, 9, 7, 3, 15].

Hadaller et al. [4] describe a method to overcome a performance anomaly within 802.11. In any 802.11 network, the worst performing device will degrade the network performance for every other device. VANETs operate in a broadcast based medium, so only one device can be transmitting within a given transmission radius. Within this medium, a device with a poor connection will transmit at

a slower rate or suffer transmission errors, causing retransmissions, thereby degrading overall performance. Hadaller et al. proposes to allocate more transmission time to devices closer to the AP. Combining this concept with IP Passing, which reduces the number of messages required for establishing communications, would be a very beneficial combination for reducing overhead and increasing the performance of VANETs.

7 Conclusions and Future Work

We have shown that by leveraging existing technologies, without AP modification, we can reduce the amount of network traffic and overhead required for a node to connect to the Internet in a VANET. The DHCP request process consists of four 428 bytes messages and multiple 128 byte ARP requests. The IP Passing protocol lowers the network overhead to two packets with a combined maximum size of 296 bytes, which is a factor of 6 better than Traditional DHCP. Additionally, the overhead for implementing IP Passing can be further reduced by piggy-backing the IP Passing messages with vehicle-to-vehicle updates for position information or other messages in the underlying network architecture.

It is possible to implement IP Passing in both neighbor-aware and neighbor-unaware networks and the neighbor unaware protocol is only slightly less optimal. The broadcast based neighbor-unaware protocol's performance is reduced due to the required delay to avoid collisions and contention for passed IPs. We expect that this small drop in performance will be compensated for by eliminating the need to maintain network structure. We plan to run simulations to determine if our intuition is correct.

We have currently only examined the benefits of passing IPs between vehicles with the same direction of travel. An interesting area of future work is determining the benefits of bidirectional passing. We also plan on exploring the effects of non-uniform speeds, transmission ranges, and APs.

Acknowledgment

This work was supported in part by the National Science Foundation (CNS-0092770, CNS-0519460, CNS-0721479).

References

[1] V. Bychkovsky, B. Hull, A. Miu, H. Balakrishnan, and S. Madden. A measurement study of vehicular Internet access using in situ wi-fi networks. In *ACM MobiCom'06*, pages 50–61, 2006.

[2] R. Droms. Dynamic Host Configuration Protocol. RFC 2131.

[3] M. Fazio, C. E. Palazzi, S. Das, and M. Gerla. Automatic IP address configuration in VANETs. In *Proceedings of the 3rd international workshop on Vehicular ad hoc networks (VANET'06)*, pages 100–101, 2006.

[4] D. Hadaller, S. Keshav, and T. Brecht. MV-MAX: Improving Wireless Infrastructure Access for Multi-vehicular Communication. In *The 2006 SIGCOMM Workshop on Challenged Networks (CHANTS'06)*, pages 269–276, 2006.

[5] B. Hull, V. Bychkovsky, Y. Zhang, K. Chen, M. Goraczko, A. Miu, E. Shih, H. Balakrishnan, and S. Madden. Cartel: a distributed mobile sensor computing system. In *ACM SenSys'06*, pages 125–138, 2006.

[6] A. Mishra, M. Shin, and W. Arbaugh. An Empirical Analysis of the IEEE 802.11 MAC Layer Handoff Process. *SIGCOMM Comput. Commun. Rev.*, 33(2):93–102, 2003.

[7] M. Mohsin and R. Prakash. IP address assignment in a mobile ad-hoc network. *MILCOM 2002. Proceedings*, 2:856–861.

[8] T. Nadeem, S. Dashtinezhad, C. Liao, and L. Iftode. TrafficView: traffic data dissemination using car-to-car communication. *SIGMOBILE Mob. Comput. Commun. Rev.*, 8(3):6–19, 2004.

[9] S. Nesargi and R. Prakash. MANETconf: Configuration of Hosts in a Mobile Ad Hoc Network. In *IEEE INFOCOM 2002*, volume 2, pages 1059–1068.

[10] C. N. Ojeda-Guerra, C. Ley-Bosch, and I. Alonso-González. Using an updating of DHCP in mobile ad-hoc networks. In *The 24th IASTED International Conference on Parallel and Distributed Computing and Networks (PDCN'06)*, pages 58–63, 2006.

[11] D. C. Plummer. An Ethernet Address Resolution Protocol. RFC 826.

[12] X. Yang, J. Liu, F. Zhao and N. Vaidya. A Vehicle-to-Vehicle Communication Protocol for Cooperative Collision Warning. *Int'l Conf. on Mobile and Ubiquitous Systems: Networking and Services (MobiQuitous 2004)*, Aug. 2004.

[13] J. Zhao and G. Cao. VADD: Vehicle-Assisted Data Delivery in Vehicular Ad Hoc Networks. *IEEE INFOCOM*, 2006.

[14] J. Zhao, Y. Zhang, and G. Cao. Data Pouring and Buffering on The Road: A New Data Dissemination Paradigm for Vehicular Ad Hoc Networks. *IEEE Transactions on Vehicular Technology*, 56(6):3266–3277, November 2007.

[15] H. Zhou, M. W. Mutka, and L. Ni. IP Address Handoff in the MANET. In *IEEE INFOCOM 2004*, volume 4, pages 2425–2433.