PVRP: Efficient disruption tolerant discovery in vehicular ad hoc networks

Giovanni Pau[2]*and Antony Rowstron[1]
[1]Microsoft Research, Cambridge, UK
[2]University of California at Los Angeles, California, USA.

1. INTRODUCTION

One in four vehicle owners in western European countries are estimated to own a GPS device (or personal navigation device (PND)). In the US it is only one in ten, but the popularity of these devices, in both Europe and the US, is growing. It is also likely that the manufacturers will add WiFi to the devices. Once, this happens there will be a vehicular ad hoc network formed by these vehicle mounted devices. However, the network will be frequently and dynamically partitioning, even when virtually all vehicles carry a WiFi-enabled device [16]. These networks will be characterized as having large density diversity, both spatially and temporally. Initially, density will be low as the number of WiFi-enabled devices will be small. However, over time, as popularity of the WiFI-enabled devices increases, the density will also increase. Nevertheless, it will still be the case that the density will be high in some regions, for example on freeways, while in other regions the density will be low, e.g. in suburban streets [16].

If we want to support general purpose routing in these environments, then disruption tolerant routing protocols are required that can work on networks like these: performing well in both high and low density regions. In general, routing protocols that need to maintain a path between a source and destination over a dynamically partitioning network are going to be inefficient. To address this geographic routing protocols [6, 12, 21, 14] have been proposed, and the general approach seems promising. Geographic protocols assume that each node has a static fixed identifier and a dynamic location or coordinate. Most protocols generally assume that there is a service (or usually an oracle) which provides the current location of any node, and that there are a set of connected nodes that can forward the packet from the source to the destination. A key challenge for creating disruption tolerant versions of these routing protocols is the development of disruption tolerant discovery services that can allow a node to determine the current location of a node.

The need for a discovery service has been observed by others, and there have been several proposals that work well on non-partitioned networks, for example GLS [10]. GLS is a distributed proactive service that uses the nodes in the network to store information about the current location of other nodes. It, as with several derivatives (e.g. High-Grade [20]), uses a hierarchical grid with the information stored at multiple levels of the hierarchy. In networks that are frequently and dynamically partitioning it is hard to ensure accessibility and consistency of this information. Reactive approaches have also been proposed in some geographic protocols, for example based on simple flood-based mechanisms. However, these too suffer if the network is partitioning, which can cause them to fail to find a destination, as well as generating high overhead. To address this issue we propose PVRP, a reactive disruption tolerant discovery service that is able to operate efficiently in both dense environments, but also in environments that are dynamically and frequently partitioning.

*Work started while on sabbatical at Microsoft Research Cambridge.

2. THE PVRP PROTOCOL

Each node in the network is assumed to have a static fixed identifier and a current coordinate in the form of a longitude and latitude, generated by the GPS receiver. PVRP is a disruption tolerant discovery protocol that allows one node to discover the current coordinate for a fixed identifier associated with another node or service (like Access Point or Internet access). The latency of discovery can range from milliseconds to tens of seconds.

In PVRP we assume that each node maintains a *neighbor table* consisting of the neighbors in one-hop wireless range and their last reported locations. This is maintained using a simple *hello* protocol. Each node beacons its current location every p seconds, and these are used to maintain the neighbor table, e.g. $p = 0.5$ seconds. Any entry for which no beacon is received in $p + \delta$ seconds is removed.

2.1 Maps

PVRP discovery protocol assumes that each node has access to a complete digital street-level map for its current world region. Such digital maps are commercially available from a number of suppliers, and are already used in PNDs. A digital map for the entire USA can be encoded in less than 1GB. Street maps are pre-processed such that each street segment is approximated by a straight line, and each

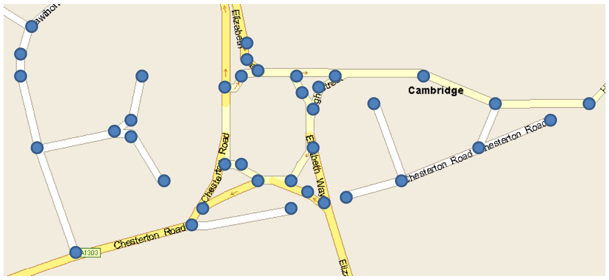

Figure 1: Rendered small street map with all junctions highlighted with a circular marker.

line terminates in a junction. Hence, the map is a graph with junctions being vertices connected by streets that are edges. Curved streets are decomposed into a sequence of straight lines connected by junctions with just two edges. For example, in Figure 1 the street at the top left corner includes two such junctions. By definition, all junctions must have at least one edge attached to them; a junction with only a single edge represents a dead end, and several examples can be seen on the middle right hand side in Figure 1. In these maps, when one street crosses another one using a bridge, it is *not* treated as a junction. The post-processed maps are readily available, and in reality this is the format that map data is usually released. For example, the USA TIGER map dataset we use, which is freely available, already comes in this format. We assume that a node can determine the street it is currently on using the GPS location information and the digital map.

2.2 Discovery algorithm

PVRP treats the digital map as an overlay on the physical topology, providing a virtual overlaid namespace in which discovery is performed. The underlying motivation for this is that the physical network topology evolves rapidly with node mobility, and therefore, any discovery algorithm expressed in terms of the physical node topology will be fragile. Conversely the rate of change of the overlay namespace (the digital map) is very slow, and therefore operating in the map namespace helps to overcome the fragility introduced by the dynamic underlying physical topology.

When a node initiates a discovery, a discovery packet is generated. The discovery packet is forwarded, using unicast, through a set of nodes on a street segment towards the location of a junction. When a discovery packet reaches a junction, the packet is forked and one copy sent down all street segments leading from the junction, except for the segment it has just traversed. In low density environments, a simple store-and-forward approach is used, so if progress cannot be made towards a junction, packets are cached on nodes until a node that can make progress is found, just as in routing.

In more detail, when a node initiates a discovery, a packet is created for *each* junction attached to the street segment that the node is currently on. The packet contains the identity of the destination or service (d), a junction from the map (j), and an empty list of junctions visited (v). In the normal case where a node is mapped onto the middle of a street segment this will result in two packets being generated, one for each of the two junctions, A and B, at the end of the street segment. One discovery packet is routed towards A and the other towards B, with $j = A, v = B$ or $j = B, v = A$ being embedded in the discovery packet. If the node is at a junction, then one discovery packet is generated for each neighboring junction directly reachable by a street segment from the current junction. The post-processed map is a graph with the useful property that, for any junction (vertex), a connected vertex can be reached by greedily geographically routing along the edge towards the vertex. The discovery packets routing exploits this: when a node Y receives a packet, it first checks if it is the destination ($d = Y$). If this is the case, the discovery has been successfully delivered and the message is not forwarded. Otherwise, if d is a one-hop neighbor, Y forwards the packet to d using unicast. If neither of these two cases hold, then Y checks if it is within m meters of the destination junction, and if it is *not*, then Y simply routes the message towards the current next junction. This is achieved by Y checking if any one-hop neighbor is geographically closer to j than Y. If so, then the packet is forwarded to the selected neighbor using unicast, otherwise the packet is cached on Y. Every time the neighbor table on Y is updated, with either a new location or a new one-hop neighbor, or the location of Y changes, then Y rechecks all the cached packets to see if any should be forwarded to another node which is now closer to each packet's j. This is therefore a store-and-forward protocol for routing packets between junctions.

If Y is within m meters of the destination junction, j, then the list of visited junctions is updated to include j, and the discovery packet is split into $k - 1$ packets, where k is the number of edges out of the junction. So, there is one discovery packet for each edge leading out of the junction, where the associated junction is not in the list of visited vertices. Each of these discovery packets is then routed independently towards the next junction, using the greedy store-and-forward routing mechanism described.

A discovery message has an upper bound on its lifetime that can be expressed in terms of number of junctions traversed, wall-clock time, geographic area or number of hops. When a source generates a discovery request it can provide bounds on all or a subset of these, but we believe that normally a discovery request will be bounded on geographic area. Given the disruption tolerant nature of PVRP the discovery packets could potentially be cached on vehicles for long periods of time. There is a realistic upper bound on how long a discovery message is valid, based for example, on the speed that the source is moving. This is likely to be on the order of ten seconds to a minute. It should be noted that for all packets, we also enforce a local timeout on the time that any node will locally cache a packet, after which the packet is dropped, e.g. 20 seconds.

In general, using the street map as an overlay is good because the message overhead is $O(S)$, where S is the number of street segments in the map. In environments where the density of nodes is high, for example on a freeway, $S << N$ and a simple flood would be $O(N)$. When the density is low, $S >> N$ so PVRP will generate more overhead than simply flooding. However, in this case, PVRP is able to successfully reach the nodes when flooding would not.

We have evaluated the performance of PVRP in both large-scale simulations and on a small-scale real world testbed and across a range of experiments it performs well. While space constrains force us to omit extensive results, PVRP showed a discovery ratio ranging from 85% to 100% with density of 50% and 100% respectively. PVRP discovery overhead resulted 20 times lower than proactive location services such as GLS and 30 times lower than disruption-tolerant flood-based discovery services.

3. RELATED WORK

There have been many proposals for location discovery services, including [10, 18, 19, 3], for use with geographic routing protocols. Most of these services are proactive and attempt to maintain information distributed over a network. When the network is very sparse this is hard. To the best of our knowledge, PVRP is the first one explicitly designed to be disruption tolerant. Delay-tolerant routing protocols, e.g [2, 1, 11, 4], have been designed to support routing over networks that are partitioning over long periods of time, in some cases days. The issue of discovering the location in a DTN has been addressed, for example, in [17] they describe an architecture that uses Home Location Servers to store the mapping between users (devices) and custodians, where a custodian server effectively knows the location of the user (device). This is implemented using a DHT and assumes that source can gain direct or indirect access to the DHT. A similar idea is proposed in OLS [5], where a DHT is run across a set of APs to maintain the mapping between vehicles identities and current locations in a vehicular environment. In general, these protocols exploit opportunistic encounters, and most exploit some element of historical contact distributions to control packet forwarding. Many of these protocols are effectively scoping a flood. PVRP can also be considered as a scoped flood, where the flood is performed in the map overlay. PVRP aims to be disruption tolerant and able to work over a network that is dynamically and frequently partitioning, but where packets can be delivered on the order of seconds to tens of seconds. Due to the environment, and the number of nodes in a vehicular environment, it is difficult to maintain meaningful historical information about vehicles inter-contact times and locations.

A number of geographical routing protocols have been proposed for vehicular networks, and many of them explicitly use a map, e.g. [12, 9, 8, 21, 15] and some implicitly, e.g. [13, 14]. Almost all use a non-disruption tolerant flood over the physical topology to find the location of the destination, e.g. RLS [7], or simply assume that an oracle is available. *All* these would benefit from using PVRP to discover the mapping between destination identity and current location.

Acknowledgements

We thank Paolo Lutterotti for his work on porting PVRP to run on QualNet.

4. REFERENCES

[1] A. Balasubramanian, B. Levine, and A. Venkataramani. DTN routing as a resource allocation problem. *CCR*, 37(4):373–384, 2007.

[2] J. Burgess, B. Gallagher, D. Jensen, and B. Levine. Maxprop: Routing for vehicle-based disruption-tolerant networks. In *Proc. IEEE Infocom*, pages 1–11, 2006.

[3] C. Cheng, H. Lemberg, S. Philip, E. Van Den Berg, and T. Zhang. SLALoM: A scalable location management scheme for large mobile ad-hoc networks. In *WCNC2002*, volume 2, 2002.

[4] M. Demmer and K. Fall. DTLSR: delay tolerant routing for developing regions. In *Proceedings of the 2007 workshop on Networked Systems for Developing Regions*, 2007.

[5] M. Gerla, B. Zho, Y.-Z. Lee, F. Soldo, U. Lee, and G. Marfia. Vehicular grid communications: The role of the internet infrastructure. In *Wicon*, Aug. 2006.

[6] B. Karp and H. Kung. Greedy perimeter stateless routing for wireless networks. In *Mobicom'00*, Aug. 2000.

[7] M. Kasemann, H. Fubler, H. Hartenstein, and M. Mauve. A reactive location service for mobile ad hoc networks, 2002. Technical Report TR-02-014, University of Mannheim.

[8] J. LeBrun, C.-N. Chuah, D. Ghosal, and M. Zhang. Knowledge-based opportunistic forwarding in vehicular wireless ad hoc networks. In *Vehicular Technology Conference*, 2005.

[9] I. Leontiadis and C. Mascolo. Geopps: Opportunistic geographical routing for vehicular networks. In *Proceedings of the IEEE Workshop on Autonomic and Opportunistic Communications*, June 2007.

[10] J. Li, J. Jannotti, D. D. Couto, D. Karger, and R. Morris. A scalable location service for geographic ad-hoc routing. In *Mobicom*, Aug. 2000.

[11] A. Lindgreny, A. Doria, and O. Schelény. Probabilistic routing in intermittently connected networks. In *SAPIR*, Aug. 2004.

[12] C. Lochert, H. Hartenstein, J. Tian, H. Fussler, D. Hermann, and M. Mauve. A routing strategy for vehicular ad hoc networks in city environments. In *Intelligent Vehicles Symposium*, 2003.

[13] C. Lochert, M. Mauve, H. Fussler, and H. Hartenstein. Geographic routing in city scenarios. *SIGMOBILE Mob. Comput. Commun. Rev.*, 9(1), 2005.

[14] V. Naumov and T. R. Gross. Connectivity-Aware Routing (CAR) in vehicular ad hoc networks. In *IEEE Infocom*, 2007.

[15] A. Oberhoffken. MIRP: Map Information Routing Protocol for mobile ad-hoc networks, 2004.

[16] A. Rowstron and G. Pau. Characteristics of a vehicular network. Technical Report 09-0017, University of California Los Angeles, Computer Science Department, July 2009.

[17] A. Seth, D. Kroeker, M. Zaharia, S. Guo, and S. Keshav. Low-cost communication for rural internet kiosks using mechanical backhaul. In *Mobicom*, Oct. 2006.

[18] S. Woo and S. Singh. Scalable routing protocol for ad hoc networks. *Wireless Networks*, 7(5):513–529, 2001.

[19] Y. Xue, B. Li, and K. Nahrstedt. A scalable location management scheme in mobile ad-hoc networks. In *CONFERENCE ON LOCAL COMPUTER NETWORKS*, volume 26, pages 102–111, 2001.

[20] Y. Yu, G. Lu, and Z. Zhang. Enhancing location service scalability with HIGH-GRADE. In *2004 IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, pages 164–173, 2004.

[21] J. Zhao and G. Cao. VADD: Vehicle assisted data delivery in vehicular ad hoc networks. In *IEEE Infocom*, 2006.