SIMPLE, EFFICIENT PEER-TO-PEER OVERLAY CLUSTERING IN MOBILE, AD-HOC NETWORKS

Thomas Zahn, Rolf Winter, Jochen Schiller
Freie Universität Berlin
Takustr. 9
14195 Berlin, Germany

Abstract - DHT-based peer-to-peer (P2P) overlays significantly reduce the overlay traffic that is needed to locate a random object on the overlay network. However, DHT-based overlays are often largely oblivious to the underlying physical network and only assign second-rate effort to the exploitation of physical proximity. Hence, a single overlay hop often amounts to an unnecessarily large number of physical hops. While this might at best be considered inefficient in stationary networks, it could prove disastrous in mobile (and wireless) networks, thus, effectively limiting the deployability of P2P overlays on top of mobile and wireless networks.

We present an approach that forms clusters in DHT-based P2P overlays based on physical proximity. By grouping physically close nodes into common overlay clusters, we can decrease the number of physical hops per overlay hop. Thus, the amount of physical traffic generated by overlays deployed on top of mobile and wireless ad-hoc networks can be reduced significantly.

1. INTRODUCTION

In a recent trend, more and more research effort has begun to be directed toward the deployment of peer-to-peer (P2P) networks in the context of mobile ad-hoc networks (MANETs). Although MANETs and P2P networks share a number of pivotal characteristics, such as self-organization, scalability, and decentralized information dissemination and discovery, many of the existing P2P architectures seem ill-suited to be deployed in MANETs without modification as P2P networks are often (deliberately) oblivious to the underlying physical network topology.

Much of the research effort on peer-to-peer (P2P) computing has been devoted to distributed hash tables (DHTs) [1,2,3,4] as those system overcome the scalability problems of first-generation P2P systems. The main advantage of DHTs is that they provide an upper bound on the number of routing hops that have to be taken to locate an object (i.e. a given key) on the P2P network. For [2, 3, 4] this bound is O(log N), where N is the number of nodes participating in the network.

However, P2P networks are overlay networks that abstract away the underlying physical network. Instead, they impose a virtual network topology that usually does not consider the underlying physical topology in its construction. This often leads to the following two situations. Firstly, a single overlay hop usually incurs many physical hops and, secondly, overlay locality is largely independent from actual, physical locality. In other words, by no means do two *overlay* neighbor nodes also have to be *physical* neighbor nodes.

Consider a DHT system with tens of thousands of participating nodes distributed over the entire globe. Suppose node A is located in New York and wants to find the participating node that is responsible for a certain key k. Node A has not yet learned which node is responsible for k, so it makes use of the DHT to route its query toward that node. The first overlay hop takes the query to node B, located in Los Angeles. Node B forwards the query to Node C, based in Sydney. Node C, then, forwards the query to Node D, located in Frankfurt. Node D finally sends the query to Node E who is responsible for key k and who also happens to be situated in New York.

This – admittedly extreme – example demonstrates that the DHT helped us efficiently locate the target node on the overlay network (with only four overlay hops). However, it also clearly shows the discrepancy between overlay hops and actual physical hops (the four overlay hops physically circled the entire globe). This discrepancy unnecessarily increases the physical network traffic and the latency. While this might at best be considered inefficient in stationary networks, it could prove disastrous in mobile (and wireless) networks. Due to limited bandwidth and transmission errors, the probability of a packet being dropped clearly increases with the path length it has to travel. Furthermore, the longer the path from source to destination, the larger the probability becomes for the previously discovered route to be broken due to mobility.

Thus, it is not sufficient to merely deploy existing P2P systems in MANETs without modifications and improvements. In this paper, we show how *Random Landmarking* can significantly decrease the physical path lengths of overlay hops. We propose a simple, yet very efficient peer-to-peer clustering scheme, based on the key characteristics of Random Landmarking, that further decrease physical network traffic in P2P networks in the context of MANETs.

The remainder of this paper is organized as follows. Section 2 discusses related work. In Section 3, we present Random Landmarking and our P2P clustering approaches in detail. Section 4 analyzes and evaluates experimental results. Section 5 concludes this paper and provides a brief outlook on our future work.

2. RELATED WORK

A common concept to close the gap between physical and overlay node proximity is landmark clustering. Ratnasamy et al. [5] use landmark clustering in an approach to build a topology-aware CAN [1] overlay network. Prior to joining the overlay network, a joining node has to measure its distance to a fixed set of landmark nodes and assigns itself a point in CAN's virtual coordinate space according to its landmark distances. The intuition behind this idea is that nodes that have similar distances to all landmark nodes, are also quite likely to be close to each other topologically. However, a fixed set of landmarks renders this approach unsuitable for mobile networks. The most significant downside of this approach is that it can lead to an extremely uneven overlay ID distribution. This means that a small set of nodes could be responsible for a very large part of the ID space, essentially turning them into hot spots. Xu et al. [6] have verified this in their study presenting a fine-tuned approach.

Pastry [3, 7] uses certain heuristics to exploit physical network proximity in its overlay routing tables. Pastry does not construct its overlay structure from the underlying physical network topology. Instead, Pastry distributes its nodes evenly in the overlay ID space regardless of the actual physical topology. During its lifetime, a node periodically performs routing table maintenance and improvement by asking other nodes for "better" routing table entries. Obviously, this is a best effort approach and, therefore, Pastry does not guarantee optimal routing table states.

Existing P2P clustering techniques are often based on semantic relations [8] and group nodes sharing a common area of interest, expertise, etc. into a common overlay cluster. However, this approach is strictly based on semantic relations and entirely abstracts away from actual physical relations (such as proximity).

3. PEER-TO-PEER CLUSTER FORMATION

As previously described, it is vital to ensure that overlay hops incur as short a physical route as possible in order to successfully deploy P2P overlays on top of MANETs. However, most existing P2P overlays pay very little or no attention to actual physical proximity in the construction and maintenance of their overlay state. While at best unnecessarily inefficient in stationary, wired networks, this is a profligacy that can be ill afforded in MANETs.

We propose a clustering technique based on Random Landmarking [9] (RLM) to actively take advantage of physical proximity in DHT P2P networks. This approach significantly reduces the path lengths of overlay hops by grouping physically close nodes into common sections of the overlay ID space. Thus, two nodes that are physically close are also likely to be "close" to each other in the overlay.

Unlike semantic-based clusters [8], RLM's overlay clusters are strictly direct mappings of physical clusters, i.e. nodes in an overlay cluster are most likely physically close to one another. [8] also uses the notion of super-peers. Here, a set of overlay nodes form a cluster around another overlay node, the super-peer, that usually handles all communication from and to its cluster. In the case of RLM, however, there is no cluster "head" that would have to handle a disproportional amount of network traffic. All nodes in an RLM cluster are likely to incur an equal amount of routing traffic. This characteristic is especially important for MANETs, where nodes often possess very limited bandwidth and/or processing power and should be prevented from becoming hotspots.

Our implementation of RLM is based on Pastry [3, 7], a well-studied DHT that provides built-in locality heuristics. Pastry's overlay construction works in a top-down fashion. It randomly assigns overlay IDs regardless of the underlying topology. It, then, tries to introduce physical proximity into its overlay routing tables through table maintenance. In contrast, with RLM the overlay network is constructed in a bottom-up fashion, i.e. the overlay is built considering locality information from the underlying network. Before a node joins the overlay, it gathers information concerning its physical neighborhood and uses that information to assign itself an appropriate overlay ID.

RLM works without any fixed landmark nodes. Instead, it uses a set of *landmark keys*. A landmark key is simply an overlay ID. Rather than having dedicated landmark nodes, in RLM those nodes become temporary landmark nodes that are currently responsible for one of the landmark keys (i.e. whose own overlay IDs are currently closest to one of the landmark keys). Therefore, when one of the current landmark nodes fails or resigns, another node (that whose overlay ID is *now* closest to the landmark key) will automatically assume its role as the DHT will automatically route queries for that landmark key to the new node.

Landmark keys should be chosen so that they divide the overlay ID space into equal-sized segments. For example, in a hexadecimal-based ID space, an appropriate set of landmark keys could be: 0800...000, 1800...000, 2800...000, . . . , E800...000, F800...000.

A joining node asks its bootstrap node to locate the current landmark nodes by simply using the DHT to route to the nodes responsible for the landmark keys and measures its distance (hops, RTT) to each of them. It will then assign itself an overlay ID that shares a prefix with the landmark node it is physically closest to. The remainder of its ID will be assigned randomly. A node also periodically re-measures its distances to the current landmark nodes to check whether it has moved out of its old cluster and into a new one. In that case, it will assign itself a new overlay ID based on the new cluster's landmark node's ID and rejoin the network under the new ID.

RLM has the following effects. First of all, it leads to physically close nodes forming overlay regions, or clusters, with common ID prefixes, which means that these nodes are also likely to be numerically close to each other in the overlay ID space. Furthermore, since the last overlay routing step in DHT systems is the numerically closest, with RLM the last overlay routing step also tends to be the physically closest, whereas with Pastry the opposite is the case [3, 7].

In wireless, mobile ad-hoc networks one should strive to exploit locality for node communication. It would certainly be advantageous if nodes could interact with physically close nodes as much as possible and with remote nodes as little as possible. A network with such local communication characteristics is not only more realistic than a completely random one (people are more likely to communicate with people in their vicinity than with people far away) but also more efficient as shorter routing paths decrease the likelihood of transmission errors and packet losses.

However, as previously described, this is usually not the case when peer-to-peer networks are deployed on top of MANETs. P2P systems are chiefly concerned with overlay routing and only marginally with efficient physical routing. We, therefore, propose the following simple, yet efficient overlay clustering scheme based on RLM to exploit physical proximity in DHT-based overlay routing. RLM creates overlay clusters (i.e. groups of overlay nodes sharing a common ID prefix) by grouping nodes based on their physical neighborhood. This leads to the following (heuristic) properties of intra-cluster communication with RLM:

1. Since nodes in an RLM cluster share a common overlay ID prefix, intra-cluster communication will likely involve only *very* few overlay routing steps – in fact, it will often be done with only one overlay hop as many of its fellow cluster members will be covered by a node's leaf set.
2. Since RLM clusters are formed based on physical proximity, intra-cluster communication will likely incur short physical routes.

Therefore, nodes should favor their own cluster when looking up objects on the P2P overlay (such as files, services, etc.). For this purpose, a node uses two different keys to insert its objects into the overlay. The first one is a global key as produced by a conventional hash function and the object is stored on a random (as determined by the hash function) node on the overlay. The object (or a reference) is also stored under a local key, which is produced by replacing the global key's prefix with its owner's ID prefix. When an overlay nodes looks for an object, it will now first query its own cluster (by replacing the prefix of the query key with its own prefix). Only if that query fails, will the node start a global query.

The next section will present simulation results that demonstrate that RLM-based clustering can significantly, and efficiently reduce physical path lengths in overlay networks.

4. NETWORK EVALUATION AND ANALYSIS

To evaluate and analyze the performance of RLM clusters, we implemented RLM and a Pastry reference application using the ns2 network simulator. Due to the complexity of ns2, we used in all simulation scenarios networks of 250 physical nodes, all of which were participating in a Pastry / RLM overlay. Each simulation run lasted one (simulated) hour, with each node issuing one overlay request per minute.

First, we evaluated mobile networks with and without churn, assuming an ideal physical network – i.e. there are no packet losses, physical routes between two arbitrary nodes are always known, and an overlay hop always takes the shortest physical route. In the second set of simulations, we employed a "real" physical, wireless network with an 802.11 MAC layer and AODV as its underlying ad-hoc routing protocol.

We considered three different churn rates for our simulations:

- No churn: No node failures, nodes live throughout a complete simulation run.
- Node lifespan of [60, 3600]s: Each node has a random lifespan between 60s and 3600s. A failed node is immediately replaced by a random, new node.
- Node lifespan of [60, 600]s: Each node has a random lifespan between 60s and 600s. A failed node is immediately replaced by a random, new node.

We also considered two different overlay table maintenance rates to take care of deteriorating effects:

- (300, 60)s: Each node performs maintenance on its overlay routing table (as defined by Pastry) every 300s. Each node maintains its leaf set every 60s.
- (60, 10)s: Each node performs maintenance on its overlay routing table every 60s. Each node maintains its leaf set every 10s.

Also, two different network traffic patterns were used:

- Random lookups: Each nodes issues an overlay request for a random key once per minute.
- 90% local lookups: Every node issues one request per minute, but we assume that 90% of the time overlay nodes request keys that are close to themselves on the overlay (i.e. that share the same prefix as the requestor's ID, or in other words local requests) and only 10% of all lookups are random (global requests).

4.1. Ideal Mobile Networks

In a first set of simulations, we compare the performance of Pastry to overlays with RLM-based clusters in mobile networks, assuming an ideal underlying physical network. Nodes move around according to the Random Waypoint Model at a constant speed of 0.6 m/s and a pause time of 30s.

Fig. 1 compares the average number of physical hops per overlay hop as generated by Pastry and RLM in various simulation settings. These figures include *all successful* overlay hops (lookup hops, routing table maintenance, leaf set maintenance, join hops, etc.). RLM achieves shorter physical paths per overlay hops than Pastry does in all but one scenario. When there is no churn, RLM achieves a ratio of 3.03 physical hops per overlay hop with a routing table maintenance period of 60s and a leaf set maintenance period of 10s as opposed to Pastry's ratio of 4.21. With longer maintenance periods of 300s and 60s, respectively, RLM still achieves a ratio of 3.27 compared to Pastry's ratio of 4.25. Even in the presence of moderate churn (node lifetime between 60s and 3600s), RLM outperforms Pastry significantly. Only when the network topology becomes too volatile, i.e. when there is high churn in the network (node lifetimes between 60s and 600s), will it take table maintenance periods of 60s and 10s, respectively, for RLM to maintain a better ratio than Pastry does. With such high churn, table maintenance periods of 300s and 60s are not sufficient to keep up with the frequent topology changes to maintain RLM's locality properties.

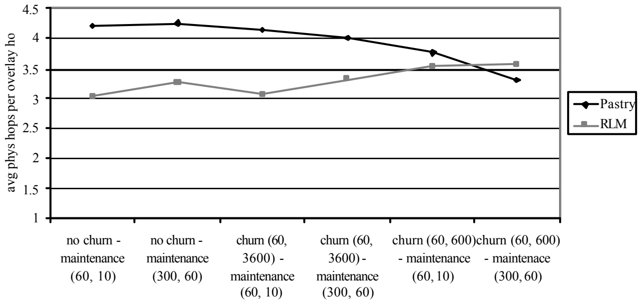

Figure 1. Average physical hops per overlay hop (including all overlay message types).

A peculiar trend can be observed with Pastry's curve. Pastry's ratio seems to be improving with higher churn rates and less frequent maintenance periods. However, this is *merely* due to the fact that with increasing churn and longer maintenance periods the number of stale routing table entries increases. It is important to bear in mind that maintenance traffic dominates the overall traffic in our simulations. There are two reasons for Pastry's curve to fall in that case: i) in Pastry, low-level routing tables entries (row 0, 1, …) tend to be physically closer than higher level entries and leaf set entries, and ii) there are exponentially less entries per routing table row (see [3, 7]). Thus, routing table maintenance can often be executed only on the lower routing table rows as there will simply be no valid entries left in the high level rows. Hence, only the fraction of shorter, lower level routing table maintenance in the overall traffic grows. Thus, Pastry's ratio seems to improve. Fig. 4 clearly demonstrates that this trend largely abates when only the overlay *lookup* traffic is considered since in the lookup process lower routing table entries, higher level entries and leaf set entries are more evenly involved.

Fig. 2 indicates another interesting characteristic to notice. RLM generates slightly more overall *overlay* messages than Pastry does regardless of the churn rate. This is due to RLM's landmark re-measuring efforts and cluster reorganization. However, as Fig. 3 shows, due to RLM's better ratio of physical hops per overlay hop, RLM always produces less *actual, physical* messages in all scenarios compared to Pastry. It is important to bear in mind that overlay traffic as such does not exist *physically*. It is merely a virtual construct. Rather, it is the actual, physical traffic produced by the overlay that has effects on network parameters such as bandwidth and power consumption. Therefore, despite the fact that RLM creates slightly more overlay traffic than Pastry does during an average simulation run, RLM will reduce the actual bandwidth consumption. This is especially crucial when peer-to-peer networks are deployed on top of MANETs

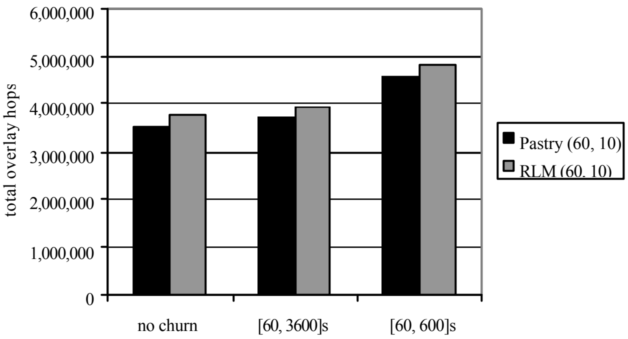

Figure 2. Total number of overlay hops (incl. all types of overlay messages) during an average simulation run in three different simulation scenarios: no churn, moderate churn, and high churn. Routing table and leaf set maintenance periods: 60s and 10s.

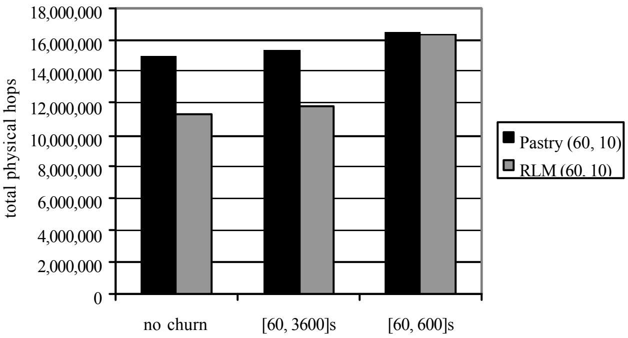

Figure 3. Total number of physical hops during an average simulation run in three different simulation scenarios: no churn, moderate churn, and high churn. Routing table and leaf set maintenance periods: 60s and 10s.

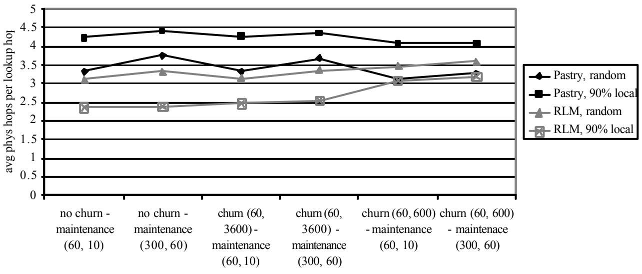

Figure 4. Average physical hops per overlay lookup hop when lookups are either totally random or 90% of the lookups are intra-cluster lookups.

Thus far, we have only considered the overall overlay traffic. Next, we will take a look at the impact of RLM on the overlay lookup process. Fig. 4 depicts the average number of physical hops per overlay lookup hop. When all lookups are completely random, RLM outperforms Pastry in networks with no churn and moderate churn - as was the case with the overall overlay traffic. In networks with high churn, topology changes occur too frequently so that RLM nodes have to constantly reassign their IDs. Thus RLM encounters stale entries more often and has to resort to less optimal table entries, which leads to ratios slightly larger than Pastry's.

Most importantly, however, when a communication pattern is applied where nodes look up objects from their own cluster in 90% of the cases and only have to resort to global lookups in the other 10% of the cases, Fig. 4 shows that RLM *significantly* outperforms Pastry in *all* churn and maintenance scenarios. In networks with no or moderate churn, Pastry lookups travel physical paths that are around 80% longer than RLM's. Even with high churn, Pastry lookups still produce physical paths that are around 25% longer than RLM's. This is because local lookups can be processed with very few overlay lookups – often with only one as the target is frequently directly covered by the originator's leaf set. These local overlay hops tend to generate short physical routes as RLM's overlay clusters are based on physical proximity.

4.2. Wireless Networks

In the second set of simulations, we employ a "real" physical, wireless network with an 802.11 MAC layer and AODV [10] as its underlying ad-hoc routing protocol. Due to increased simulation complexity, we simulate networks of 150 mobile nodes. In the case of RLM, 8 landmark keys are used. Nodes move around according to the Random Waypoint Model at a constant speed of 0.6 m/s and a pause time of 30s. We use a request rate of one lookup per minute per node.

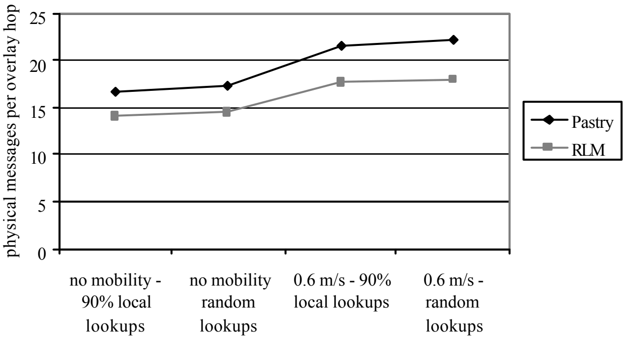

Figure 5. Physical messages per overlay hop.

Fig. 5 shows the number of physical messages, i.e. AODV packets, that are sent out to perform an average overlay hop. Again, these figures include all (lookups, maintenance, etc.) overlay hops. In both stationary and mobile wireless networks, RLM generates around 20% less AODV packets for the execution of an overlay hop than Pastry does when the same communication patterns are assumed. If one compared a conventional Pastry overlay (random traffic) to a RLM cluster-based one (90% of lookups can be answered locally), this difference exceeds 25%. With a request rate of one request per minute per node, maintenance traffic dominates lookup traffic in our simulations. Thus, the difference would clearly further widen with an increasing request rate as RLM's local lookups would reduce the need for AODV to (re-) discover long routes.

5. Conclusion

Existing P2P overlays often neglect to consider physical locality or consider it only marginally. Thus, overlay routing often incurs unnecessarily long physical routes, which limits the deployability of such systems on top of MANETs.

In this paper, we have presented and analyzed a P2P overlay clustering technique based on Random Landmarking. RLM groups nodes that are physically close to each other into common regions of the overlay ID space. Thus, nodes that are close to each other in the overlay network are also likely to be physically close to one another.

Simulation results show that the exploitation of RLM clusters significantly reduces the average physical path lengths of overlay hops. Furthermore, when objects are stored both globally and inside their owners' overlay clusters so that lookup can often be serviced locally, the performance improvements become even larger. Our simulation results further indicate that RLM clusters can reduce the number of physical packets exchanged to perform an overlay hop in an AODV-based MANET by up to 25%. Therefore, RLM-based P2P overlay clustering considerably improves the deployability of P2P overlays on top of mobile ad-hoc networks.

6. References

[1] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker. "A Scalable Content-Addressable Network". In *Proc. of ACM SIGCOMM*, Aug. 2001.

[2] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan. "Chord: A scalable peer-to-peer lookup service for internet applications". In *Proc. of ACM SIGCOMM*, Aug. 2001.

[3] A. Rowstron and P. Druschel. "Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems". In *International Conference on Distributed Systems Platforms (Middleware 2001)*. Nov. 2001.

[4] B. Y. Zhao, J. D. Kubiatowicz, and A. D. Joseph. "Tapestry: An infrastructure for fault-resilient wide-area location and routing". Technical Report UCB//CSD-01-1141, U.C. Berkeley, April 2001.

[5] S. Ratnasamy, M. Handley, R. Karp, and S. Shenker. "Topologically-Aware Overlay Construction and Server Selection". In *IEEE Infocom 2002.*

[6] Z. Xu, C. Tang, and Z. Zhang. "Building Topology-Aware Overlays using Global Soft-State". In *ICDSC'2003,* May 2003.

[7] M. Castro, P. Druschel, Y. C. Hu, and A. Rowstron. "Exploiting Network Proximity in Peer-to-Peer Overlay Networks". *International Workshop on Future Directions in Distributed Computing (FuDiCo)*, June 2002.

[8] W. Nejdl, M. Wolpers, W. Siberski, C. Schmitz, M. Schlosser, I. Brunkhorst, and A. Löser. "Super-Peer-Based Routing and Clustering Strategies for RDF-Based Peer-to-Peer Networks". *In Proc. of the 12th International World Wide Web Conference*, May 2003.

[9] R. Winter, T. Zahn, and J. Schiller. "Random Landmarking in Mobile, Topology-Aware Peer-to-Peer Networks". *10th International Workshop on Future Trends in Distributed Computing Systems (FTDCS 2004)*, May 2004.

[10] Ad hoc On-Demand Distance Vector Routing. http://www.ietf.org