

Ad Hoc Networking via Named Data

Michael Meisel
UCLA
meisel@cs.ucla.edu

Vasileios Pappas
IBM Research
vpappas@us.ibm.com

Lixia Zhang
UCLA
lixia@cs.ucla.edu

ABSTRACT

The design of the Internet protocol stack, with IP at the waist of the hourglass, mandates that packet delivery is governed by the destination IP address. This design has enabled explosive growth of the wired Internet, but faces two basic issues when applied to mobile environments. First, many mobile networks, such as mobile ad-hoc networks, are infrastructure-free, while Internet protocols are generally built with infrastructure support in mind (e.g., DHCP servers to handle IP address assignments). Second, node mobility introduces a high degree of dynamics in node interconnectivity, which defeats conventional routing protocols, originally designed for wired networks. In this paper, we argue that mobile networks can be made more effective and efficient through Named Data Networking (NDN) [4] (aka CCN).

Categories and Subject Descriptors

C.2.1 [Computer-Communication Networks]: Network Architecture and Design—*Wireless communication*

General Terms

Design

Keywords

ad-hoc networking, architecture design

1. INTRODUCTION

Over the years, many research efforts have been devoted to mobile networking, in particular ad-hoc networks. The basic goal of these networks is to deliver data to a given destination node no matter where it is; the delivery may take multiple hops through a network whose nodes and interconnectivity may change at any time.

Compared to their wired network counterparts, the basic challenge in data delivery in mobile networks is their high

degree of geographical and topological dynamics. Not only can the destination move, which defeats the routing protocols that are designed to deliver packets to stationary locations¹, but the network itself can be made of mobile nodes whose locations and interconnectivity may change at any time. This also defeats conventional routing protocols for wired networks whose designs are based on the notion that links and nodes are stationary and dynamics are due only to link/node failures and recoveries.

Years of research efforts in ad hoc networking have produced a rich literature of solutions to the above mentioned problems, however there still seems no convergence on one or a set of final standard solutions. A commonality among the existing solutions is that they typically adopted the basic models from the wired Internet protocol stack, complete with IP address assignment to each mobile node, traditional routing at the network layer and unicast support at the MAC layer to deliver data to a specific destination IP address. These solutions simply swap out wired MAC and physical layers for their wireless equivalents, and modified traditional wired routing protocols to take into account the movement of individual nodes in an ad hoc setting. It is a testament to the design of the Internet protocol stack that it can work in an environment as foreign as a multi-hop wireless network, however a working system does not necessarily serve as proof that adopting the wired solutions in a wireless, mobile environment is the best possible approach.

In this position paper, we would like to return to the drawing board and rethink how to align data communications and routing with the fundamentally broadcast nature of wireless channels. We find that this alignment can lead to an entirely different design with vastly improved performance. Our design incorporates Named Data Networking (NDN) [4] (aka CCN²), a newly proposed Internet architecture.

The paper is organized as follows. Section 2 presents a high level overview of existing solutions to data delivery in ad hoc networks. In Section 3, we introduce the new direction of networking through named data. As an existence proof, Section 4 gives a high level summary of LFBL, a new data delivery protocol designed for networking via named

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MobiArch'10, September 24, 2010, Chicago, Illinois, USA.
Copyright 2010 ACM 978-1-4503-0143-5/10/09 ...\$10.00.

¹Although all network routing protocols must handle dynamic changes due to link or node failures, dynamics of mobility is different in nature. For example, delivery to stationary nodes can use address aggregations, delivery to mobiles cannot.

²The research effort on this new direction of Internet architecture design has been renamed to NDN, although PARC continues to use the name CCN for its open source package.

data in ad-hoc networks. We conclude the paper in Section 5.

2. EXISTING SOLUTIONS FOR MOBILE NETWORKS

Research efforts on mobile ad-hoc networking started with the DARPA Packet Radio Network (PRNet) [6] in mid 70's, and became a hot research topic in recent decades. Over the years, many solutions have been developed. Until recently, these solutions have generally fallen into one of a few broad categories: proactive, reactive, and hybrid proactive-reactive.

2.1 The IP-Routing-Based Approach

Proactive protocols, such as WRP [8] and DSDV [11], are essentially a direct adaptation of routing protocols for wired networks. Like their wired counterparts, proactive ad-hoc routing protocols establish the best path to all reachable destinations and maintain consistent, up-to-date routing information at all of the nodes.

Reactive routing protocols, such as DSR [5] and AODV [10], aim to address the high overhead of proactive routing approaches by establishing routing paths only when needed – that is, on an on-demand basis.

Hybrids of the two approaches, such as ZRP [3] and HARP [9], tend to use a hierarchical approach, combining proactive routing at certain levels of the hierarchy with reactive routing at others.

Although these existing solutions vary significantly from one another, all share the following basic approach:

1. Each node is assigned an IP address.
2. Applications communicate by sending data to specific destination addresses.
3. In the presence of node movement, each scheme attempts to determine a single best path to the given destination IP address, and delivers data to that IP address via that single path.
4. To cross each hop along the determined path, the sending node controls which node within the signal range should receive the data. That is, despite the fact that wireless channels are broadcast in nature, all of these types of existing designs impose point-to-point delivery across each hop.

2.2 Failings of the IP-Routing-Based Approach

The above approach is inherited from the wired protocol architecture. It has proven to be an effective design in the wired setting, but does not fit the ad hoc environment well for the following reasons.

First, assigning IP addresses to moving nodes is difficult. Because IP addresses are a very limited and precious resource, IP address management is tightly controlled and requires infrastructure support, such as a central DHCP server. This directly conflicts with ad hoc networks' desire to operate in an infrastructure-free manner.

Second, technological advances have brought us ever increasing number of computing devices, most of which are now mobile. Managing IP address assignment for all of these devices is becoming increasingly infeasible and less meaningful. In wired networks, IP addresses both represent topological locations and enable network routing to scale through

address aggregation. In an ad-hoc network where all the nodes may potentially move, nodes do not have fixed locations and IP address aggregation is not feasible. The best way to assign IP addresses to nodes in an ad-hoc networks has long been recognized as an open question [12]. In fact, the only clearly identifiable reason for assigning IP addresses to nodes in ad-hoc networks, other than as an arbitrary, temporarily unique identifier for the device, is compatibility with the existing protocol stack implementation.

Third, all communication is meant to serve the purpose of delivering data to application processes. It is the data itself that is meaningful to applications. However, because data is invisible in today's IP-centric architecture, one can only meet applications' needs by selecting a specific node to send data to, resulting in sub-optimal data delivery. When a node N receives a packet P , it forwards P to the next hop following the path given by some routing protocol, then deletes P , although the same packet could be needed again, either for the purpose of retransmission due to packet loss at the next hop, or by applications running on other nodes. N cannot save P because the data itself cannot stand alone, independent of its destination node.

Fourth, in an ad hoc network, there is an inherent trade-off between the accuracy of routing state maintained at each node and the overhead to keep this state consistent. Since data is delivered over a *single* path, the binding between a mobile node and its current IP address is critical. With either high node mobility or a large network (where even when individual nodes do not move much, there is constant movement in the aggregate), one suffers from either high overhead to keep the binding updated and/or loss of connectivity due to outdated binding information.

Finally, letting the sender determine which node will be the receiver does not utilize the broadcast nature of the wireless channel. In a broadcast channel, there are potentially multiple receiving nodes within the sender's signal range that can hear the data transmission. In general, receivers are in a better position to make forwarding decisions than senders. For example, the intended receiving node may no longer be where the sender expected, yet since the receiver is named, no other node in the range can step in to take over the forwarding task. To keep senders informed of all their neighbors' movement and connectivity changes will necessarily increase the routing update overhead.

2.3 Opportunistic Routing and DTN

More recently, a number of **opportunistic** IP-routing-based protocols have been developed, such as ExOR [1] and MORE [2]. These protocols take a step in the right direction by taking advantage of the broadcast nature of the wireless channel. As a result, they move away from items 3 and 4 of the IP-routing-based approach described above. However, they still fully adhere to items 1 and 2, and therefore suffer from all of the associated shortcomings.

Daly Tolerant Networking (DTN) has also attracted many research efforts recently. Although there seem to be some apparent similarities between DTN and NDN, there exist fundamental differences between the two. In DTN, whole application data units are framed within bundles (which are usually considerably larger than packets). Though DTN performs late binding of data names to node IP addresses, each bundle is still ultimately delivered by following the traditional addressing scheme of senders and receivers. In con-

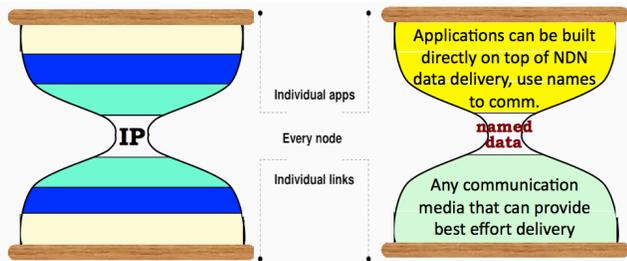


Figure 1: NDN moves the universal component in the Internet protocol stack from IP to named data.

trast, packets in NDN have unique names that are used directly for delivery. As a new Internet architecture model, NDN has no dependency on IP, although it can fully utilize IP connectivity wherever it is available. Furthermore, NDN automatically embraces ad-hoc networking and delay-tolerant data delivery, rather than having these functions through separate additional modules as in today’s Internet architecture.

3. A NEW DIRECTION FOR MOBILE NETWORKS

In a recent paper [4], Jacobson *et al.* suggest a new direction, dubbed *named data networking* (NDN), to evolve the Internet protocol architecture. As shown in Figure 1, NDN replaces IP with *named data* as the thin waist in the Internet hourglass architecture. That is, instead of sending data packets to a given destination IP address, hosts communicate in an NDN network by requesting desired data. Instead of IP nodes, data becomes the first class entity in an NDN network.

In this section we explain why communicating by named data not only helps the global Internet in general, but also enables simpler and more efficient ad-hoc/mobile networking in particular.

3.1 Named Data Networking (NDN)

At a high level, applications in an NDN network communicate using the following 3-way exchange. First, applications (running on some nodes) with content that may be interesting to others announce the content names, or the prefixes of the names, in a way similar to IP routing announcements. Second, applications that are interested in a particular content send out *Interest* packets which contains the names of the requested data. Depending on specific network settings, these Interest packets may propagate along multiple paths towards potential locations of the data, and pull the data down to the requesting nodes. Along the way, intermediate forwarding nodes may cache the data packets to meet potential future requests for the same data. Unlike IP forwarding, where a packet is solely for delivery to a specific destination address and the router removes it from memory as soon as it finishes forwarding, named data is meaningful in its own right, independent from its consumers. Thus, routers can cache a packet for later transmission when receiving the same interest request.

From the above brief description, we see that an NDN network changes the communication semantics from “where” to “what”. This is particularly advantageous for ad-hoc net-

works. Generally speaking, application data have relatively stable structures compared to the dynamic topological connectivity in an ad-hoc network. The IP address of a mobile node may be subject to change when its current location changes, however the application data names carried on the node do not necessarily change. In fact, both landmark-based and geo-based routing solutions for ad-hoc networks can be viewed as the thinking along a similar direction of using more stable/scalable names, although each of these solutions picked a specific name space with specific definitions. In some sense, they can be considered as specific sub-cases of the more general NDN architecture.

Making data the first class entity in the architecture also greatly facilitates security development. When the communication paradigm is delivering data to specific locations, one has to worry about securing the communication channels and all of the boxes along the way. This is a difficult problem, as evidenced by the ever-increasing security threats on the Internet, despite many diligent efforts to secure it. In an NDN network, one only needs to worry about the security of the data itself, “have I received the correct data sent by the intended party as I requested?” The task of securing the data can be accomplished by end-to-end cryptographic signatures and encryption (when data secrecy is needed), leaving open only the task of key management among the data sending/receiving parties, but not any channel or boxes in the middle of the data delivery paths.

3.2 NDN for Ad-Hoc Networking

The NDN approach can be particularly beneficial in an ad-hoc networking environment. Mobile nodes can communicate based on what data they need, instead of computing a specific path to reach a specific node. This can greatly simplify the implementation for the following reasons.

First, one no longer needs to assign IP addresses to each node; instead the nodes can use application data names directly to forward Interests and data packets among each other.

Second, because this data-name-based design does not have routing loops [4], Interest packets can be forwarded along *multiple* paths towards potential data locations; in case more than one direction returns the requested data, a node can evaluate which path gives the best performance and send future Interest for the same data source in that direction only. This multipath approach is particularly beneficial in ad-hoc networks because the use of multiple paths removes the critical dependency on precomputed single paths, hence relaxed the stringent requirements on the timeliness of routing updates and routing state consistency among all the nodes.

Fundamentally, an ad hoc network itself can be so dynamic that spending limited resources to chase down accurate interconnectivity information at every moment can be too costly. In an NDN-based ad hoc network design, there is no need to assign either IP addresses or identifiers to each node. When a node *A* has something worth sending, as determined by a specific application, it advertises the data using an application-level data name. When a node *R* needs some data, it simply sends an Interest packet using the corresponding application data name. Other nodes which receive the Interest will decide either to forward it towards *A* (the data name advertiser), or to broadcast the Interest. This decision is governed by local control settings and policies.

One common question regarding the NDN design is its use of names. NDN requires that all nodes communicate using application-level names that can be associated with the data being transmitted. We note that, in fact, all data *is* associated with high level names which simply have not been used in networking level data delivery; instead network delivery has been location specific and used addresses. In the next section, we use a specific design to illustrate the benefits of address-free data delivery in ad-hoc networks.

Another benefit of NDN for mobile networks is the ability to uniquely address and cache fragments of application data. Traditional caching approaches, which cache objects rather than fragments of objects, work poorly or not at all in mobile environments. The reason is that each cached object has to be retrieved in its entirety from the same caching node, assuming a stateful transport protocol, like TCP. Given that most cacheable objects, e.g. images, audios, videos, cannot fit within one packet, the implementation of any transparent caching techniques in mobile networks becomes challenging and highly ineffective. In essence, transparent caching techniques work only in static networks, assuming that routing paths remain stable and always go through the same transparent cache. Other caching techniques, such as byte caching, that work at the network layer rather than the transport layer, suffer from the same limitations. They can work only when routing paths are stable and byte caching nodes know the exact path that the packet will follow, a requirement that cannot be satisfied in mobile environments.

In contrast, NDN enables caching even when paths are neither stable nor predictable. Given that any fragment of information is uniquely identified, it can be cached in any node that forwards it and it can be reused if other nodes request it. For example, if one was to download a lengthy cacheable MPEG file over HTTP within a mobile network, none of the intermediate nodes in the mobile network will be able to cache the whole file, assuming that packets will follow different paths. With NDN, the different data chunks of the same MPEG file are assigned a unique name and thus, when a node requests the file, the intermediate nodes can store any parts of the MPEG file that goes through them. Subsequent requests for the same MPEG file, or requests for retransmission of some parts of the file, can then be served by the copies of the various parts of the file that have been opportunistically stored in the intermediate nodes.

Finally, NDN is perfectly suitable for enabling communication on top of any type of network, a requirement so crucial in mobile networks where the network connectivity state can range from almost always connected to intermittently connected. In essence, NDN enables the implementation of a large gamut of protocols, ranging from real-time to delay-tolerant protocols, within the same framework. With NDN, mobile nodes do not need to sense their connectivity state and consequently adjust their communication modality based on the current network conditions. For example, a mobile node that was originally connected through a 3G cellular network can transition to peer-to-peer, delay-tolerant communication over Bluetooth once it moves out of the 3G network range, without the need to change its communication modality. In contrast, in a traditional, IP-based communication environment, the node would have to transition from a TCP-based application protocol working over 3G to possibly a UDP-based, delay-tolerant protocol.

4. DESIGN EXAMPLE: LFBL

Listen First, Broadcast Later (LFBL) [7] is a new forwarding protocol for wireless ad-hoc networks, designed from the ground up with named data in mind. LFBL uses a variation of NDN's 3-way exchange (name prefix announcements, Interest forwarding, and data return) meant for a mobile, ad-hoc environment. A node N floods a request that carries the name of the requested application data. Any node or nodes that happen to have that named data send a response packet, which is forwarded to N using information gleaned by the intermediate nodes during the request phase. Finally, N sends an acknowledgment as feedback to encourage or discourage these responses. NDN broadcasts name prefixes (as all routing protocols do), while LFBL saves the prefix propagation by paying the cost of flooding requests.

At intermediate nodes, the responsibility for forwarding decisions is placed squarely in the hands of the receiver, rather than the sender. After receiving a packet, a potential forwarder pauses to listen to the channel, waiting to see if a more optimal node forwards the packet first. Otherwise, it forwards the packet itself. While the idea of receiver-based forwarding is not new in and of itself, its full potential in the context of highly dynamic mobile/ad-hoc networks, particularly when combined with named data, has not yet been explored.

We would like to highlight LFBL's use of named data, otherwise known as an `appID`. First, communication by data names helps support logical mobility, where the requested data may either move between nodes, or is present at multiple nodes at the same time. An example of the former can be multiple moving nodes passing by a traffic accident, each taking a snapshot when they pass; an example of the latter is retrieving a popular media file that many other nodes have downloaded. In LFBL, any node can respond to a request as long as that node can provide the requested `appID`. If the requester receives more than one response to its request, it can pick the one it prefers by choosing which to send an acknowledgment to. A responder which ceases to receive acknowledgements for its responses will eventually give up.

Second, because forwarding decision is made by the receiving end of each hop, LFBL achieves robust data delivery in presence of high node mobility. As nodes move, a node originally on the best path may have moved away from its original spot, while a nearby node may have become the best candidate for forwarding, all without the sender's awareness. The listen-first nature of the scheme allows each node makes the best decision about its role in data forwarding. Thus, despite node movement, data can continue to flow towards the requester, finding any available path along the way.

To evaluate the performance of LFBL, we implemented LFBL in the QualNet network simulator³. We experimentally evaluated LFBL and compared it against AODV, a representative wireless ad-hoc routing protocol. LFBL outperforms AODV using four different metrics and in a diverse set of simulation scenarios. Our results show that, under high dynamics, LFBL delivers nearly 5 times more data compared to AODV, while having comparable overhead, introduced mainly due to redundant packets. (See Figure 2.) This and other results validate our design choices in terms of the capability of LFBL to deal with highly dynamic environments, caused due to physical mobility of nodes and

³<http://www.scalable-networks.com/>

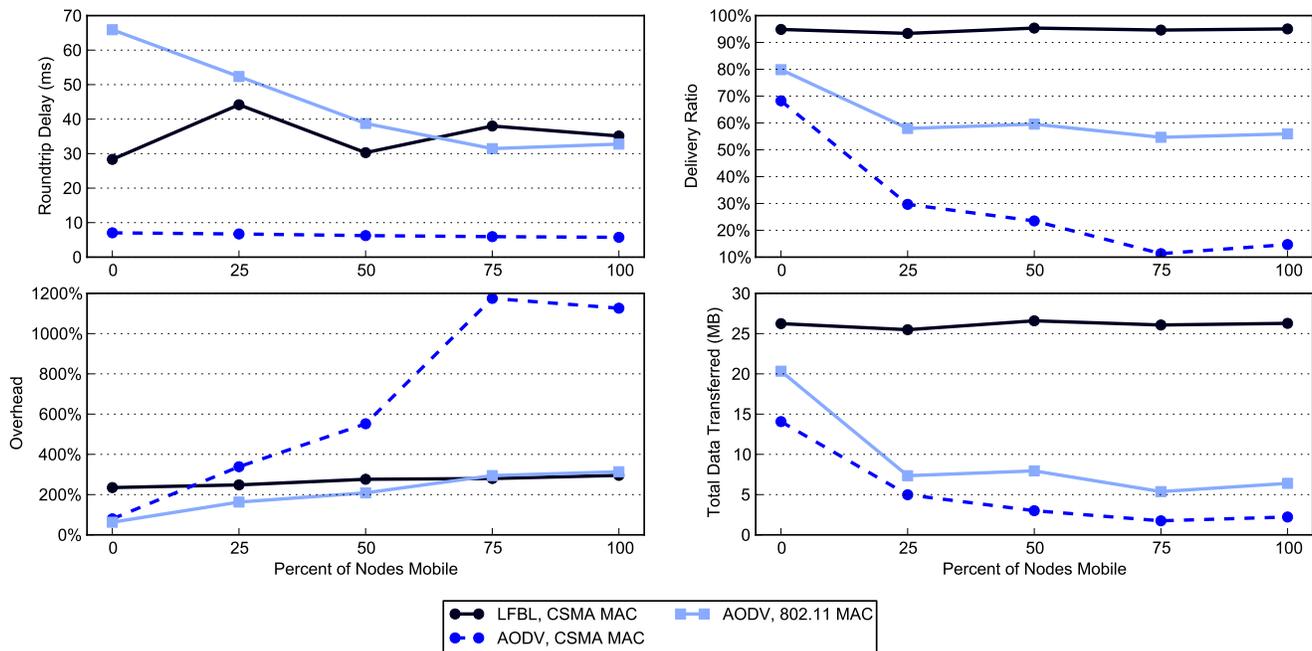


Figure 2: Some characteristic results for LFBL vs. AODV. This scenario explores the effect of the percentage of mobile nodes. The nodes that are mobile move at a fixed rate of 30 meters per second using a random waypoint mobility model.

logical mobility of application identifiers. Interested readers can find more details in the LFBL paper [7].

5. CONCLUSION

In today’s networks, many devices can become truly mobile, with adequate resources and wireless communication capacity. However, mobile device communications are often marked by frustration and inconvenience. Users are forced to be highly aware of their connectivity environment, with many applications only working when networking infrastructure is available. The simplest ad-hoc network is that of two people with laptops sitting next to each other, who cannot email a file they wish to share because infrastructure is either unavailable, not working properly, or too costly to use. Most often, people simply fall back on the use of USB flash drives. The billion US dollar market for flash memory is, in part, a testament to the failure of the mobile networking research community to provide a network architecture that supports truly ad-hoc mobile communications.

In this paper, we made an observation that the solutions to ad-hoc networking typically adopted the basic models from the existing wired Internet protocol stack, which do not meet the need for ad-hoc communications. Our work in LFBL was motivated by our surprise that dealing with high dynamics in general was still an elusive goal. We attributed this failure of existing routing protocols mainly to their dependence on network topology. While current ad-hoc networks were designed to deal with changes in the network topology, their main pitfall in dealing with those dynamics was that they required the full or partial network topology for the computation of best routes. As such, frequent changes in the topology had a direct impact on the performance of those protocols. Based on the above observations, we set out to de-

sign a new *forwarding* protocol for highly dynamic multi-hop wireless networks, capable of dealing both with physical mobility of nodes and logical mobility of application data. The result is Listen First, Broadcast Later (LFBL), a new multi-hop wireless protocol comprising of a distributed *forwarding* capability with essentially no *routing* protocol. LFBL’s use of data names for communication is a powerful primitive, which it has yet to be fully exploited.

We advocate that the named data networking approach, as sketched out by Jacobson *et al.* [4], represents a promising architectural direction to develop effective and efficient solutions for ad-hoc networks.

6. ACKNOWLEDGEMENTS

This research was sponsored by the U.S. Army Research Laboratory and the U.K. Ministry of Defence and was accomplished under Agreement Number W911NF-06-3-0001. The views and conclusions contained in this document are those of the author(s) and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

7. REFERENCES

- [1] S. Biswas and R. Morris. ExOR: opportunistic multi-hop routing for wireless networks. In *SIGCOMM 2005*. ACM.
- [2] S. Chachulski, M. Jennings, S. Katti, and D. Katabi. Trading structure for randomness in wireless opportunistic routing. In *SIGCOMM 2007*.

- [3] Z. J. Haas, M. R. Pearlman, and P. Samar. The zone routing protocol (zrp) for ad hoc networks. In <http://www.ietf.org/proceedings/55/I-D/draft-ietf-manet-zone-zrp-04.txt>.
- [4] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard. Networking named content. In *Proceedings of the 5th ACM International Conference on Emerging Networking Experiments and Technologies*, 2009.
- [5] D. Johnson and D. A. Maltz. Dynamic source routing in ad hoc wireless networks. In *in Mobile Computing, Kluwer Acad. Publ., 1996*.
- [6] J. Jubin and J. D. Tornow. The darpa packet radio network protocols. In *Proceedings of the IEEE, 75(1): 21-32, January 1987*.
- [7] M. Meisel, V. Pappas, and L. Zhang. Listen first, broadcast later: Topology-agnostic forwarding under high dynamics. Technical Report 100021, UCLA Computer Science Department, 2010.
- [8] S. Murthy and J. Garcia-Luna-Aceves. An efficient routing protocol for wireless networks. In *ACM Mobile Networks and App. J., Special Issue on Routing in Mobile Communication Networks, Oct. 1996, pp. 183-97*.
- [9] N. Nikaein, C. Bonnet, and N. Nikaein. Harp - hybrid ad hoc routing protocol. In *International Symposium on Telecommunications 2001*.
- [10] C. E. Perkins and P. Bhagwat. Ad hoc on demand distance vector routing, mobile computing systems and applications. In *Second IEEE Workshop on Mobile Computing Systems and Applications, 1999*.
- [11] C. E. Perkins and P. Bhagwat. Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers. In *ACM SIGCOMM'94*.
- [12] N. Vaidya. Open problems in mobile ad hoc networking. In *Keynote talk presented at the Workshop on Local Area Networks 2001*.