Available online at www.sciencedirect.com
ScienceDirect
ELSEVIER
Ad Hoc Networks 5 (2007) 324–339
Ad Hoc Networks
www.elsevier.com/locate/adhoc

A survey on real-world implementations of mobile ad-hoc networks

Wolfgang Kiess *, Martin Mauve

Computer Networks Research Group, University of Düsseldorf, 40225 Düsseldorf, Germany

Received 3 January 2005; received in revised form 13 September 2005; accepted 6 December 2005
Available online 13 January 2006

Abstract

Simulation and emulation are valuable techniques for the evaluation of algorithms and protocols used in mobile ad-hoc networks. However, these techniques always require the simplification of real-world properties such as radio characteristics or node mobility. It has been shown that this may lead to results and conclusions which do not reflect the behavior of ad-hoc networks in the real world. Various prototype implementations demonstrate that even simple protocols such as flooding do not behave as it was predicted by earlier simulation. To overcome this problem, real-world experiments are required. In this paper, we present a survey on existing real-world implementations of mobile ad-hoc networks. We report on the technology used for the implementations as well as on key findings from experiments conducted with these implementations.
© 2005 Elsevier B.V. All rights reserved.

Keywords: Mobile ad-hoc networks; Routing; Implementation; Testbed; Real-world; Experiment; Emulation

1. Introduction

Mobile ad-hoc networks (MANETs) enable mobile users to communicate without the use of a fixed infrastructure. These networks can be used, e.g., to extend the range of access points, to allow communication in disaster areas or to realize inter-vehicle communications. There are a lot of technical challenges in designing MANETs, and for a lot of those challenges, solutions have been presented.

A central problem in this area of research is to prove that a given solution is viable and, possibly, to demonstrate its superiority in relation to other approaches. An established and widely used method for this purpose is network simulation. However, it has become apparent that simulation can only be a first step in the evaluation of algorithms and protocols for MANETs. The key reason for this is threefold:

- Simulations always require certain assumptions about the real world. These may turn out to be wrong or too coarse to capture all aspects that influence the performance of algorithms and protocols.
- Some important characteristics of MANETs, like radio propagation or energy consumption, are inherently hard to model accurately in simulators.

* Corresponding author. Tel.: +49 211 8111637.
E-mail address: kiess@cs.uni-duesseldorf.de (W. Kiess).

1570-8705/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.adhoc.2005.12.003

- Simulations do not allow the solutions to be tested in the environment they were designed for.

As a consequence, some of the algorithms and protocols for mobile ad-hoc networks have been implemented and studied in the real wold. Given the effort that is required for real-world implementations it is very surprising that there is a very large amount of duplicated work in this area. It is the key aim of this survey to increase the reuse of prior work by giving a concise summary of existing real-world implementations and pointing out the most important results that have been gained through the experimental evaluation of ad-hoc networks.

In the remainder of this paper we concentrate on the issues connected to the real-world implementation of MANETs. Nevertheless, there are other evaluation techniques such as simulation or emulation and other types of multi-hop radio networks such as mesh and sensor networks. Both will be touched upon as far as this contributes to the better understanding of real-world MANET implementations.

The paper is structured as follows: starting with an overview of the historical development of mobile ad-hoc networks in Section 2 we continue with a classification of techniques for the evaluation of MANETs in Section 3. One of these techniques, emulation, is briefly outlined in Section 4. Experiments conducted with sensor and mesh networks are examined in Section 5. Section 6 investigates real-world experiments with MANETs. The key findings of the experiments are summarized in Section 9. Section 10 highlights the advantages of integrating simulation, emulation and real-world experimentation. An outlook to future directions of research for real-world MANET implementations concludes the paper.

2. Historical development

Research on multi-hop wireless networks (which were initially called packet radio networks) started in the early 1970's. The ALOHA [1] project at the University of Hawaii was among the first demonstrations of feasibility for using packet broadcasting in a single-hop system. Based on the knowledge acquired through ALOHA, the DARPA funded PRNET project [39,36] was started in 1973. PRNET was a multihop Packet Radio NETwork system that reached a size of around 50 nodes and allowed some nodes to be mobile. It contained features still present in todays MANETs, e.g., a routing protocol employing mechanisms that are currently used by DSR and AODV. Other PRNET features were the remote debugging capability and the ability to remotely load code to the nodes. The PRNET was in daily experimental use for at least ten years [36]. An in-depth discussion of the packet radio network technology in the early to mid 80's with a special focus on findings of the PRNET project can be found in the *Proceedings of the IEEE, Special Issue on packet radio networks* [46].

The follow-up project of PRNET was SURAN (SURvivable Adaptive Networks, 1983–1990) [7] which had the goal to develop techniques enabling the operation of a packet radio network in the presence of electronic counter measures. For the SURAN project, a number of routing algorithms, an in-lab emulator and a real-world demonstrator based on custom made hardware were developed. For the demonstrator, a total of 180 custom made nodes were produced, the largest experiment with the demonstrator involved 22 nodes (some fixed, some car mounted, one airborne). The protocols developed for the SURAN project were intended for large networks with up to 10 000 nodes but these large-scale settings were not evaluated in real-world tests. The knowledge acquired during SURAN was used by the US army to enhance existing radios with packet switching capability. A survey on further MANET projects and experiments conducted by the military can be found in [67].

With the broad availability of WLAN hardware and small-scale, low-cost portable devices in the late 1990's, interest in MANETs increased dramatically. In the following we focus on results from that time up to now.

3. Overview of different evaluation techniques

After an initial theoretical analysis, algorithms and protocols for ad-hoc networks can be evaluated by simulation, emulation and real-world experiments. These methods are necessary to prove or disprove assumptions and to identify interprotocol and interlayer effects which are hard to discover when examining a protocol or an algorithm in a purely analytical way. As a general rule, the number of assumptions influences the accuracy of the results: the fewer assumptions are required by a method (i.e., the higher the degree of realism), the more can the results of the tests be trusted to represent real-world behavior. On the other hand, the increasing realism also leads to increased complexity.

Real-world parameters are often out of the experimenter's hands. This makes it difficult to repeat experiments and to fully understand and correctly interpret their results.

In a **simulation**, all of the influencing factors and also the algorithms that are to be investigated are modeled and examined in an artificial software environment with a high degree of abstraction. This allows repeatability, tight control, large scale and cost effective tests, possibly with heterogeneous operating systems and programming languages. On the downside, there is a lack of realism: as all effects must be simulated, wrong assumptions about these effects or even the lack of some effects lead to test results that do not reflect the behavior of the algorithm in a real-world implementation. Furthermore the simulator software may contain non-standard conform, simplified protocol implementations. Therefore, most results produced by simulation should be considered as qualitative assessments. Various studies have shown that the relative ranking of protocol performance can depend on the simulator or different physical layer models chosen for the study [79,49]. Examples of commonly used simulators are ns-2 [63] or GloMoSim [27]. A detailed discussion of MANET simulation is beyond the scope of this paper.

In an **emulation**, hard- and software designed for real-world deployment is modified and combined with simulation components to run under controllable laboratory conditions. The advantages are repeatability, tight control and a certain degree of realism. The costs per tested node are higher than with simulation and there are also technical scalability bounds.

In a **real-world experiment**, all parts of the system are fully functional in a real-world setting. The whole network is deployed and tested under realistic, albeit experimental conditions. Thus, no potentially wrong or inaccurate assumptions about external influences are made. Real-world experiments comprise all effects on the network and can provide feedback for simulation or emulation. Furthermore, a real-world experiment is the ultimate way to prove that an algorithm or protocol works as expected. The drawbacks of real-world experiments are the lack of repeatability and tight control as well as the limited scalability[1] mainly caused by high costs for hardware, software and manpower.

[1] The largest MANET experiment we know of comprised 72 nodes [82].

4. Emulation

An **emulator** is a combination of soft- and hardware used to mimic the behavior of a network with some of its components being implemented in the real world and others being simulated. There is a lot of published work about emulators for wired and wireless networks, here we focus on emulators for wireless ad-hoc networks. The purpose of those emulators varies, some are built to allow to test protocols on real hardware, others are used to prepare real-world experiments. In the latter case emulation is used to form a virtual dynamic topology among the nodes. This allows easy in-lab testing without moving the nodes physically around in the forefield of a full-scale experiment. Emulators can be subdivided into physical and MAC layer emulators.

In a *physical layer emulator*, all network layers except the physical layer are implemented in a real system. Physical layer emulators mangle the radio signal emitted by the wireless interfaces of the nodes to mimic the effects the radio waves would experience in a real-world setup. One possibility to do this is to attenuate the emitted signal as in the SALT/PRISM emulator [7] build for the SURAN project or as described in [38]. Here, the signals are fed with cables into programmable RF attenuators. The emulators presented in [77,14,16] also use attenuation with analogous components although these are not programmable. In the MiNT emulator [16] and the Illinois Wireless Wind Tunnel [81], the emitted signal is also attenuated to scale down the experiments. Furthermore, these two emulators allow nodes to be mobile by placing them on remote controlled vehicles. The EWANT emulator [77] also uses attenuation and emulates mobility by switching between different antennas. The emulator build by Judd and Steenkiste [37] digitizes the radio signals, feeds them into a signal processor to model the signal propagation effects and then feeds the signals back into the wireless interfaces. The ORBIT lab emulator [72] scales the radio range by transmitting at low power levels and emulates movement by switching between different nodes.

As pointed out in [16], physical layer emulators based on attenuation face some limitations: (1) in contrast to the signal that is also attenuated at the sender, interference is only attenuated at the receiver (2) receivers may be in the near-field zone of the sender (3) the small-scale fading is not realistic, being especially an issue for experiments with mobile nodes.

Inverse physical layer emulators reverse the approach of physical layer emulators: they simulate the upper parts of the network stack and transmit the packets using real hardware. This technique has been used by the sensor network software environments TOSSIM [47], EmStar/EmSim [25], EmTOS [26] and in the hybrid simulation mode of the MANET emulator MiNT [16].

In a *MAC layer emulator*, all network layers except the MAC and physical layer are implemented in a real system. MAC layer emulators simply determine the nodes that should receive a given packet: if a node is emulated to be within radio range of another node, a filter tool allows the exchange of packets between them, if the nodes are out of each others range, the respective packets are dropped. The filter tool can be either placed on a central machine or run on each participating node. In a centralized system all nodes send their packets to a central machine which then determines the nodes that should receive the packets. This can be done by using an established simulator [41] or by writing a special tool [48,22]. Decentralized emulators can be based on available network filter tools such as iptables used in [13,88,56] or specifically designed filter tools such as DSR macfilter [55], APE mackill [5] or the MAC filter presented in [32].

By dynamically adding and removing filter rules, the emulator can also create scenarios with node movement. Trace-based emulation [62,49,50] adds additional wireless effects beyond simple reachability to the filtering: the behavior of a real network is measured and then used as input for a tool which drops, corrupts and delays packets according to the real network characteristics.

The filters developed to build decentralized MAC layer emulators are also used in real-world experiments. The reason for this is that radio hardware is known to be very unreliable in terms of transmission range, leading to long-range, unstable links. These links are highly problematic for many routing protocols. To counter this, the filters are used in real-world experiments to explicitly drop packets from distant nodes.

5. Related real-world experiments

5.1. Sensor networks

Sensor networks consist of small, low-power, low-energy (stationary) nodes used for monitoring parameters such as temperature, humidity, and motion. Algorithms and protocols for these networks often focus on energy conservation and techniques for data aggregation. However, sensor networks are wireless multi-hop networks, therefore they do share some fundamental problems with mobile ad-hoc networks.

In [89] a sensor network consisting of up to 60 nodes was used to measure the packet delivery rate with respect to distance and time in office and outdoor environments. The authors identified an area in the shape of a ring close to the maximum transmission range which they call *gray area*. This area covered 20%–30% of the radio range. In this gray area, packet reception was possible but the packet loss rate had a high variance both in time and space: loss rate varied between 10% and 50%.

The behavior of basic flooding (i.e., every node rebroadcasts each packet exactly once) has been examined in detail in [23]. The authors present a sensor network with over 150 nodes deployed in a dense grid topology. Contrary to expectations, some of the nodes did not receive the flooded packet. Furthermore, a treelike representation of the flooding process reveals that the packet was received by some nodes via *backward links* (i.e., by nodes which are farther away from the source than themselves) or via *long links* (i.e., the packet is received over a distance longer than the assumed radio range). In a MANET setting, this behavior could have severe impact on routing protocols such as AODV or DSR which rely on flooding as a means to find nodes and discover routes.

The authors of [87] present a sensor network with up to 91 nodes intended to collect votes from congress participants. The measurements were performed in a laboratory environment on a grid topology. Routing was performed with a single-destination variant of DSDV: the network's traffic sink regularly flooded route requests to the network. With this, the nodes were able to select the best next hop to the sink based on a link quality metric. The authors discovered significant end-to-end loss rates over multiple hops. They implemented passive acknowledgments (i.e., a node retransmits a packet if it does not hear the same packet being forwarded by its downstream neighbor) to reduce the losses. In a small setup with 24 nodes, the passive acknowledgments decreased the loss rate. However, in larger experiments with 48 and 91 nodes the loss rate increased. The authors conclude that congestion caused by duplicate packets was responsible for this.

In [31], a 70 node sensor network intended for the tracking and detection of vehicles is presented. The authors discovered that asymmetric links can lead to instable reception rates and that an initial idea to overcome this, link layer handshaking, is expensive. Therefore, they used a different method to avoid asymmetric links. During the creation of a diffusion tree, spanning all nodes, the packets which allow the nodes to detect their parents were sent with a lower transmission power. Thus, the nodes selected parents which are close. The resulting link is therefore symmetric with a high probability.

In [12], experiments with up to 55 sensor nodes are performed to determine the radio characteristics of one indoor and two outdoor environments. The gray areas here span 50%–80% of the radio range. Furthermore, 5%–30% of the links were found to be asymmetric. The authors also present evidence that asymmetric links may be caused by differences in hardware calibration: when the positions of two nodes connected by such a link were swapped, the link asymmetry was inverted in 91% of the tested cases.

One of the largest sensor networks has been deployed in the context of the ExScal project [6] for intruder detection. The network consisted of more than 1000 sensor nodes and about 200 802.11b nodes that served as backbone network, creating a 2-tier network structure. Each of the backbone nodes was responsible to relay the messages of a certain number of sensor nodes to a base station. The authors discovered that the distance vector protocol initially used transported only 33.7% of the sensor nodes' messages to the next backbone node. Therefore, the authors switched to LGR (logical grid routing). LGR selects the routes according to a spanning tree that is computed during a setup phase. In combination with a custom transport protocol, 99% of the sensor nodes' packets could be delivered to the next backbone node.

Summarizing, the experiments done with communication in sensor networks show that physical layer effects must be considered when building multi-hop wireless networks. The findings on gray areas, asymmetric links and congestion are particularly interesting, since the number of nodes used in the experiments was comparatively high.

5.2. Mesh networks

The most mature wireless multi-hop networks with respect to real-world deployment are mesh networks. Mesh networks are composed of stationary nodes equipped with radio hardware and connected to the power supply system. Commonly, their aim is to provide multi-hop access to the Internet.

In the MIT roofnet project [74], two scientific mesh networks with up to 29 (indoor) and 38 (outdoor) nodes have been deployed. On the 29-node indoor network, the properties of links between 802.11b-equipped nodes were evaluated [18]: out of 124 existing links between the nodes, there were 28 links where forward and reverse delivery ratios differed by at least 25%. Furthermore, the impact of different packet sizes on the delivery ratio of single-hop transmissions has been evaluated and it could be shown that larger packets have a much lower probability of being delivered than smaller ones. It is also shown that a purely hop-count based selection of end-to-end routes often results in suboptimal routes [17,18]. Therefore, the outdoor roofnet network uses Srcr [8] as routing protocol, a variant of DSR modified to find routes with high throughput. In [2] the links in the outdoor network are examined by letting each node send a number of 1500 byte packets. The authors show that it is difficult to strictly distinguish between neighbors and non-neighbors as there are a lot of links with intermediate or high loss rates. Signal-to-noise ratio and distance exhibit only a weak correlation to the delivery rate and experiments with a physical layer emulator [37] reveal that multi-path fading may be responsible for these loss rates. The end-to-end performance of roofnet (here with 37 nodes) is evaluated in [8]. Lossy high throughput links seem to be a good choice in multi-hop paths as this provides better overall throughput than high quality links with a low bandwidth. Furthermore, short-distance, high-throughput links are preferable to long-distance, low-throughput links in this respect.

In [73] a three-node multi-radio mesh network is studied. As nodes, Linux workstations with up to four 802.11b network interfaces are used. It is shown that the throughput is reduced by up to 33% if more than two network interfaces are installed in one node (one interface transmits, the other interfaces are only switched to a passive state). The authors suspect that radiation leaking from the passive cards and board crosstalk is responsible for this. Then a two-hop experiment is described in which each hop can be performed on a different channel as the middle node uses two network interfaces. It is discovered that it is not possible to operate the two network interfaces at full capacity regardless of the channel used. This became feasible with a minimum antenna separation of at least 35 db (corresponding to 1 m of antenna separation).

The experiments presented in [19] study the impact of four different link-quality metrics on overall end-to-end TCP throughput in a 23-node indoor mesh network. The two most important of those metrics are hop count and the expected transmission count (ETX) metric. ETX uses single-hop broadcasts to determine per hop loss rates and, based on this information, calculates the path with the lowest overall number of (re)transmissions. The nodes in the examined network are Windows XP machines equipped with 802.11a radios. Routing is performed with a variant of DSR adapted to the corresponding metric. Initial baseline measurements reveal the existence of asymmetric links in the network: for about 50% of the one-hop links with two directions, the reverse and forward bandwidth differs by more than 25%.[2] In the vast majority of the presented measurements, the ETX metric achieves the best throughput, followed by the hop count metric and the other two metrics. The only exception to this is a TCP throughput measurement from a single mobile node carried around the periphery of the network to one of the static nodes. Here, the hop count metric exhibits a better performance as ETX does not adapt fast enough to link quality changes. The testbed was modified for a follow-up experiment [20] to support two network interfaces per node. Initial baseline measurements on the testbed with single-hop transmissions reveal that two radios of the same 802.11 dialect (a/b/g) in one node interfere with each other regardless of the used channel. Similar to the measurements presented above [73], throughput drops significantly due to this.

Apart from scientific approaches, there are also private mesh networks that mainly serve as access networks to the Internet. Examples include the efforts to cover the Dutch city of Leiden [86] or the city of Melbourne, Australia [57] with a mesh network. Further information on mesh network implementations can be found at [85,9].

The experiments performed on mesh networks are of particular interest since most of these networks are in real-world use. Thus very practical issues such as routing metrics and routing stability are investigated under realistic constraints. As a result the findings are useful in the context of MANETs, even though mesh networks do not include node mobility.

[2] The radios were allowed to dynamically select their data rate.

6. MANET experiments

Experiments on MANETs have been conducted on static topologies and in scenarios involving node mobility. The following two sections summarize the findings of those experiments.

6.1. Static topologies

6.1.1. Proof-of-concept implementations

Many projects concentrate on the proof-of-concept implementation of protocols and the validation in a simplified set-up. Such efforts consist mostly of the code installed on notebooks which are then used to conduct experiments. Examples of this kind of experiments can be found, e.g., in the proceedings of the REALMAN workshop [15]. While most of these experiments provide valuable proof of the performance of individual protocols, in the following we focus on those experiments that either contribute general insights or innovative methodology.

6.1.2. ABR at Georgia Institute of Technology

The ABR protocol was studied in a static four node network using a chain topology in [80]. ABR belongs to the class of reactive routing protocols. It uses flooding for route discovery and beacons for route maintenance. Amongst other results, it was shown that different beacon intervals had very little influence on ABR route discovery time. In order to improve comparability, the authors also defined a systematic procedure executed at the beginning of each experiment. Before any measurements were taken, all links were tested with ping sessions to ensure that the conditions had not changed significantly since the previous experiment.

6.1.3. AODV at University of California, Santa Barbara

The work of Royer et al. [75] concentrates on real-world implementations of AODV. The authors took several real-world related issues into account, including: (1) The loss of state information during reboot of a node which may result in routing loops (discovered by Bhargavan et al.). (2) A harmful interaction between AODV and TCP: if route requests are answered by an intermediate node instead of the destination, the destination has no route to the source. This causes problems with the transmission of TCP ACKs. (3) If AODV is used by a node with multiple network interfaces, the node must be able to distinguish the different networks associated with each interface. Therefore the authors added an interface field to the routing table entries. (4) Finally, packet buffering was introduced during route discovery to reduce packet loss.

The DAMON tool [69], developed by the same group, is intended for the monitoring of mobile networks. In the course of an experiment to validate this tool the reception rate of unicast and broadcast packets has been measured. The measurements have been conducted in a noisy congress environment on a number of 802.11b-equipped nodes for both data and AODV management traffic. It has been discovered that there was no correlation between the loss rates of the unicast and broadcast packets, and the authors argue that it is therefore difficult to use broadcast packets for route discovery if the route is later on used by unicast packets.

6.1.4. OLSR at INRIA Rocquencourt

An evaluation of the OLSR protocol under Linux with 802.11b network interfaces can be found in [45]. A test on a four-hop string topology revealed the existence of *fluctuating links*, i.e., links with a range longer than the specified radio range, a poor quality and a high variance both in time and space.[3] To overcome these fluctuating links, two predefined signal strength thresholds were defined. A node was accepted as neighbor if the signal strength was higher than the high threshold and removed from the neighbor list if the signal strength dropped below the low threshold. Experiments in a static 20 node indoor setup without any data traffic revealed that most of the control packets lost were lost on links with a low signal strength. Although the above presented signal strength method worked quite well, there were still some links with a low delivery ratio.

In a second set of experiments, TCP and UDP performance was tested in 1:N and N:1 scenarios, i.e., one source communicates with N sinks and vice versa. In the 1:4 TCP experiment, each of the data streams got its share of the available bandwidth although two one-hop connections received significantly more bandwidth than the two two-hop connections. The trend of uneven bandwidth distribution intensified in the 4:1 TCP scenario where the two sources in one-hop distance captured the whole bandwidth while the other two sources were unable to transmit any significant amount of data at all. For the UDP test in the same setup, the bandwidth per connection was limited to 1 Mb/s. In the 1:4 UDP scenario, each connection got an equal share of the bandwidth. Although this changed slightly for the 4:1 UDP scenario where the sources in one-hop distance received more than their fair share, the other sources were still able to deliver data.

[3] In fact, fluctuating links and gray areas as presented above are two names for the same phenomenon.

6.1.5. PARO at the National Autonomous University of Mexico/IBM T.J. Watson Research Center, Hawthorne

In [28], the power-aware PARO protocol is evaluated in a test with three Linux notebooks with 802.11 adapters. PARO introduces additional hops between nodes that can otherwise communicate directly to minimize overall energy consumption. To exchange packets with nodes that cannot communicate directly, the authors suggest to combine PARO with a classical multi-hop routing protocol. The three nodes were positioned on a line such that the outer nodes could just communicate at the maximum transmission energy level (100 mW). The node in the middle was then moved from one side to the other and the achievable energy savings with PARO were measured. Already in this small setup, some problems were discovered which did not show up in the preceding simulations:

- As the power levels of the wireless interfaces could only be regulated to be 1, 5, 20, 50 and 100 mW, the overall consumption of the system could not be reduced in one scenario in contrast to an expected saving of 50%.
- PARO requires that at least 802.11 RTS/CTS traffic is sent at the maximum power. The authors have discovered that their radios (Aironet PC4800) need ≈7 ms to switch signal levels which is longer than the spacing between RTS/CTS and Data/ACKs. This means that PARO does not work with standard 802.11 hardware.

Another interesting issue was the overall energy consumption: the card transmitted with a maximum power of 100 mW but needed about 1400 mW to do so. Thus, transmission power is not (yet) the bottleneck.

6.1.6. Radio characterization at Pervasive Computing and Networking Laboratory/University of Pisa/IIT Institute Pisa, Italy

Although the experiments presented in [4] are performed in single-hop settings, they contain some information relevant for multi-hop network design. Interesting are the measurements of the communication distance with respect to parameters such as data rate and ground height with 802.11 network interfaces. The authors determine the communication distances[4] for their 802.11 equipped laptops at different data rates in an outdoor setting under optimal conditions: 30 m at 11 Mb/s, 70 m at 5.5 Mb/s, 90–100 m at 2 Mb/s and 110–130 m at 1 Mb/s. It is furthermore shown that the distance of the node from the ground influences packet delivery. In the respective experiment, the 802.11 radios are set to 11 Mb/s and the nodes are placed 30 m apart. The packet delivery ratio in this setting varies between 85% and 98% for node heights between 0.4 m and 1.6 m.

6.2. Experiments including node mobility

6.2.1. DSR at Carnegie Mellon University

The work on the DSR prototype [55,54] started in 1998 at the Carnegie Mellon University. It comprised five mobile nodes installed in cars moving at top speeds of 40 km/h, a mobile node connected via mobile IP and two stationary nodes which were installed 671 m apart at opposite ends of the course traveled by the mobile nodes. The nodes were equipped with 900 MHz WaveLAN-I radios with a nominal range of 250 m and GPS for tracking purposes, routing was performed with DSR. To overcome the missing link layer acknowledgments of the WaveLAN-I radios, acknowledgments on the routing layer were implemented lowering the per-hop loss rate from 11% to 5% by means of retransmissions.

The designers of the DSR prototype identify several tools and utilities which have proven to be valuable for the analysis and debugging of the prototype [55]:

- A GPS receiver at each mobile node enabling the tracking of individual nodes.
- A visualization tool that displays the status of the nodes and allows a birds view on the experiment.
- Tcpdump to track all packets for a detailed post-run analysis.[5]
- A per-packet signal-strength recording.
- A per-packet state-tracing which recorded the internal states of the used protocols, namely TCP and DSR.
- A macfilter which allowed the emulation of movement without actually moving the nodes.

In an initial test of the DSR prototype ping packets were sent from the first stationary node to the second stationary node via the five nodes circling between them. With a loss rate of about 5% for the first hop, the overall end-to-end loss rate is reported to be 10%. About 90% of the packets used two and three-hop routes. Due to the variability in the environment, roughly 10% of the ping packets were exchanged directly between the two nodes over a distance of 671 m producing a loss rate of 22.3%.

During the evaluation of a TCP transfer in a static two-hop scenario [55,54], fluctuating links led to poor performance: three nodes were set up in a chain topology, with the two outer nodes being positioned such that they were as far away from the middle node as possible but still able to successfully transmit ping packets to the middle node. Temporarily, the two outer nodes were able to communicate directly leading to a significant amount of packet loss. The use of a macfilter prohibiting the use of this one-hop route improved the throughput by 30%. Therefore, the authors emphasize the necessity of a mechanism to prevent the use of fluctuating links.

The authors of [55] also mention some additional general lessons learned:

- Packets controlling the routing protocol should be delivered with high priority (e.g., by implementing multi-level priority queues).
- Management of human experiment participants is difficult and time consuming.
- Wireless signal propagation is highly variable.

The DSR prototype implementation was extended to support real-time traffic such as audio and video [34]. In a network consisting of one mobile and seven fixed nodes with 802.11 Lucent WaveLAN adapters the mobile node transmitted

[4] The maximum communication distance is defined as the point where the packet reception probability drops below 85%.

[5] The authors emphasize that the additional processing time due to the usage of tcpdump has influenced the results of some experiments as this delayed acknowledgments.

an audio and a video stream over up to three hops to one of the fixed nodes. The experiment showed that the transmission of real-time traffic over an ad-hoc network is possible if the routing protocol is adapted to the specific scenario.

6.2.2. AODV/DSDV at Sydney Networks and Communications Lab

An experiment conducted with implementations of AODV and DSDV is described in [14]. The two routing protocols were tested in a scenario with four fixed and one mobile node: the fixed nodes were set up in a chain topology, the mobile node passed this chain from one end to the other. For the experiment Linux PCs and notebooks with 802.11b adapters were used. The maximum transmission rate was limited to 1 Mb/s to avoid automatic rate changes by the 802.11 b adapters. Furthermore, the adapters were wrapped with metallic anti-static bags to limit the transmission range to 5 m thus allowing in-lab testing.[6] Two tests were performed in this setup, sending UDP packets from the mobile node to one of the fixed nodes at the end of the chain and transferring a file with FTP in the other direction. It was discovered that both routing protocols frequently selected very unreliable links which resulted in poor performance. The reason for this problem was that both routing protocols prefer routes with a low hop count. Implicitly this leads to a preference for unreliable long range links. The DSDV implementation did not suffer as much as AODV as it used a handshake before accepting a link. To overcome the unreliable links, the *powerwave* tool was implemented as a sub-layer below the routing layer: nodes regularly exchange echo packets with each other to filter those links with a bad signal-to-noise ratio. The authors have detected two shortcomings in their tool: the high network load due to echo packets and the insufficient interaction with the routing protocols as they are not informed about link breakage but have to detect this situation by employing their own timers.

6.2.3. Centibots at Artificial Intelligence Center SRI International, Menlo Park/University of Washington, Seattle/Stanford University

One of the largest MANETs was deployed within the scope of the centibots project [43]. The goal of the project was to deploy a team consisting of 100 autonomous robots for the surveying of an indoor area. The robots used 802.11b network interfaces, routing was performed with TBRPF, a pro-active link-state routing protocol. The largest number of robots running at the same time were 72 with a maximum route length of five hops and a throughput of about 1 Mb/s [82]. The robots were moving at 30 cm/s in an area of 650 m^2. When the experimenters tried to run all robots at once, the network broke down. The problem was solved by bringing 10–18 nodes up at a time. The final reason for this problem was not fully identified, the experimenters name three potential sources: the network interfaces, TBRPF and the TBRPF implementation they were using.

6.2.4. GPSR at University of Mannheim

The Fleetnet Router [30,58,59] implements the greedy forwarding strategy of the position-based routing protocol GPSR, i.e., a node selects the neighbor closest to the target as next hop. The target's position is discovered by flooding a position request. On reception of the request, the target sends a reply containing its position. Nodes are installed in cars and have the following components: a Windows-based application PC, a Linux-based 802.11b router, onboard GPS and GPRS to monitor the internal state of the node. Furthermore, packets received from nodes farther away than 220 m are dropped to avoid the fluctuating link problem. In a static three-hop experiment with the fleetnet router [59], it was discovered that the maximum achievable throughput of 400 kb/s depends on the size of the packets as smaller packets lead to more collisions. In the same setup with mobile nodes, it has become evident that unacknowledged broadcasts are often lost. Thus, flooding used to discover the target's position took a long time to reach all nodes. Furthermore, the lack of feedback from the MAC layer about broken links was an issue. In [58], the experimenters evaluated the router in a static three-hop setup with notebooks without the cars and the application PC and found some additional problems. During one of the test runs, a bursty loss occurred blocking nearly all packets. The authors suspect interference and attenuation by large objects between sender and receiver to be responsible for this. Furthermore, high round trip times occurred for the first packet of each test run. This was due to process scheduling of Linux.

[6] Due to the different transmission range, the AODV timers had to be adapted.

6.2.5. Routing protocol evaluation at Dartmouth College/Colorado School of Mines/University of Illinois at Urbana–Champaign/Bucknell University, Lewisburg

In [29], an experimental comparison of four MANET routing protocols (APRL, AODV, ODMRP and STARA) can be found. The network consisted of 40 laptops equipped with 802.11b cards running at a fixed rate of 2 Mb/s. Nodes had GPS receivers attached to track their position for later emulation and simulation. The nodes flooded beacons with their own position and timestamped positions of other nodes to the whole network. Emulation was performed by placing all nodes in the same room and using packet filtering to emulate a dynamic topology. The experiment itself was conducted on a rectangular athletic field of size 225×365 m on which the notebooks where carried around by the participants in a random fashion. The results only take 33 of the 40 nodes into account as seven nodes did not work correctly. The outdoor experiment revealed that the two reactive protocols AODV and ODMRP deliver much more messages than the two proactive protocols. However, even those protocols produced a high overhead and achieved a low absolute delivery rate. The repetition of the experiment by means of simulation showed large differences to the real experiment [49]. This has been extended in [50] to examine how different radio layer models affect the simulation results. A simple stochastic RF model with standard outdoor parameters produces results that are closest to the real experiment while the other models (also those enhanced with the connectivity information from the experiments) differed more.

6.2.6. DSR at Rice University, Houston

In [76] the authors use unmodified DSR routing code from the ns-2 simulator in a real network. In order to achieve this, the code is encapsulated in a user-level process that provides a simulator/real-world packet format converter. Using this technique the ability to handle real-time video traffic over a mobile ad-hoc network is investigated. The network used in the experiments consists of four stationary and two remotely controlled mobile nodes. The communication at each node is performed over 802.11b equipped Linux notebooks that use DSR for routing, the mobile nodes have an additional Windows notebook that handles the live video. The average packet delivery ratio during the demonstration is above 95% at an overall latency of about 30 ms, thus validating the presented implementation technique.

6.2.7. DSR at University of Colorado, Boulder

The MANET examined in [35] is composed of 10 nodes out of which some are mounted on remote-controlled miniature airplanes. The nodes are composed of single board computers equipped with 802.11b network interfaces and GPS, routing is performed with DSR. The authors demonstrate in this work that it is possible to combine airborne and ground nodes in a MANET. They achieve a throughput of about 250 kb/s at a latency of 30 ms over up to three hops.

6.2.8. Ad-hoc networking with directional antennas at BBN Technologies, Cambridge

In [71], a system for ad-hoc networking with directional antennas is described. The implementation of this system used the same routing code for the real experiment as for the simulation. For routing, the link-state routing protocol HSLS was used. An experiment was conducted with 20 nodes (cars) that drove around a 4×3 km area. Each car was equipped with four directional antennas selectable on a packet-per-packet base and 802.11b as physical layer. According to the authors, their system outperformed a similar setup (20 cars but with omnidirectional antennas and OLSR) although more details on this are not available. In a second experiment, a helicopter was added as aerial node.

7. Testbeds

A *testbed* is a framework which supports testing, comparing and evaluating algorithms and protocols in the real world.

The only existing testbed for mobile ad-hoc networks used to a larger extent is the ad-hoc protocol evaluation testbed (APE) [52]. APE is a Linux distribution which can be booted directly from CD on regular notebooks. Each experiment participant is instructed to move according to a choreography script. To a certain extent this makes experiments repeatable. Furthermore, the authors have integrated tools to collect traces about the experiments and to upload these traces at the end of an experiment to a central computer. They have also developed the *virtual mobility metric* based on measured signal quality. The idea is to use per packet signal quality to compute virtual distances between the nodes. These distances describe the topology of the network as it is perceived by the nodes and are used to determine how similar two repetitions of and experiment are with respect to connectivity.

The indoor experiments with APE presented in [52] were conducted with 9–37 nodes. The nodes were divided in at most four independently moving groups which split up and reunited in the course of the experiment. The authors ran several experiments with OLSR and AODV. By comparing the virtual mobility graphs of the distinct experiments the authors conclude that the choreographed approach is suitable to produce comparable test runs.

A four node experiment with APE [53] revealed the existence of *communication gray zones*[7] in 802.11b based ad-hoc networks. A node X is said to be in the communication gray zone of a node Y if it is listed in the neighbor table of Y but Y cannot forward any data traffic over X. The reason for this lies in the different reception characteristics of broadcasted beacons used for neighbor discovery and unicast data packets in 802.11b-based ad-hoc networks: (1) 802.11b broadcast packets are normally sent at a lower bit rate than unicast packets, thus they can be received over greater distances. (2) Broadcast packets are not acknowledged and can thus be transmitted over unidirectional links. (3) The small size of beacons results in fewer packet losses due to bit errors and collisions. (4) Fluctuating links lead to entries in neighbor tables about nodes which are only occasionally reachable. The authors also evaluated the impact of three different strategies to overcome gray zones, exchanging neighbor tables, accepting a neighbor only after the reception of three beacons and discarding beacons received with a low signal quality. They show that all three strategies improve the packet delivery rate significantly.

The ORBIT Testbed [65,72] is under development and will consist of a 400-node indoor radio layer emulator and a 50-node outdoor, full-scale network. The indoor network described in [72] consists of 64 static nodes with 802.11a/b/g network interfaces in a grid layout.

All other testbeds are still in a conceptual state. There is a plan to extend netbed [83] and there is also work on WHYNET [84] in the context of the NSF network research testbed program. An overview of existing (wired and wireless) testbeds and recommendations for future work on testbeds can be found in the 2002 NSF testbed workshop report [64].

[7] This is different form the problem of gray areas/fluctuating links.

8. Software tools

Software tools are intended to ease the task of implementing and evaluating algorithms and protocols for ad-hoc networks. They can be roughly divided into:

- frameworks,
- monitoring tools,
- performance metrics.

Frameworks support the task of implementing MANETs. The PICA API [10] and the "user level framework for ad-hoc routing" [3] shadow the calls to operation system specific functions. With this, a protocol can be developed once and used on different operating systems without porting the implementation. The MANET routing framework [60], FRANC [11] and the ad-hoc support library [40] extend this approach. Besides allowing platform independent implementations, they also offer some common services needed by a lot of algorithms and protocols. The idea is to implement services such as flooding, neighbor discovery, packet buffering during reactive route discovery, reliable unicast and broadcast, queues, timers, packet sniffing or network emulation for testing purposes in the framework. This allows the implementer to concentrate on the specifics of the individual algorithm or protocol.

The click modular router [42] provides a script language allowing the combination of simple modules which have tasks like decrementing the TTL or recalculating the checksum of a packet to a router. Modules can be easily written and there is a whole library of modules available. This approach accelerates the protocol development as it fosters reusability. Click has been developed for routing in fixed networks but has also been used to implement routing protocols for MANETs.

Monitoring tools collect information, such as battery state, traffic statistics and link quality and transmit it to one or several sinks. The collected information is then used for analytical or management purposes. Monitoring information can be transmitted either in-band, i.e., over the experimental network itself [39,55,35] or out-of-band, i.e., over an additional network [30,68,26,78].

The PRNET network monitoring [39] was intensively used during the whole project for debugging, to alter operating parameters of the radios and for remote software updates. As each station in the network registered the neighbors' beacons, the exact time of failure could be determined in case of an error. The SURAN automated network manager [7] was mainly used for network visualization. The CMU position and communication tracking daemon [55] provided the information for a highly developed visualization tool. This tool allowed a "bird's eye view of the network". The authors claim that this was crucial to explain the network to others and have also used the tool for debugging. The information acquired with DAMON [69] is used to visualize a network and for troubleshooting. Monitoring support is also integrated into the nodes used for the MANET experiments of the University of Colorado, Boulder [35] and the Fleetnet Router [30]. The ORBIT measurements framework and library (OML) can be used to steer the experiment as shown in [78]. Here, a traffic source increases the data rate until the monitoring reports that loss exceeds a certain threshold. Furthermore, OML is used to monitor the nodes' hardware status.

MERIT [21] is a framework to assess the performance of a routing protocol. The routes an optimal routing protocol with global knowledge would have chosen are therein compared to those routes actually chosen by the routing protocol. Currently this approach has not yet been fully implemented.

9. Summary of results

Even though the existing experiences with real-world implementations of mobile ad-hoc networks are quite heterogeneous, there are several observations that can be generalized:

- A lot of available links in a wireless network are asymmetric. This has been shown for sensor networks [12,31], mesh networks [17–19] and MANETs [44].
- In has been shown for sensor networks that the direction of an asymmetric link can be switched by switching the positions of the two affected nodes [12].
- Distance may only exhibit a weak correlation to the packet reception rate. In sensor networks, this is known as *gray areas* [12,89], in MANETs as *fluctuating links* [55,54,53,45,59]. The problem has also been verified for mesh networks [2]. The emulator experiments in [2] suggest that this may be an effect of multi-path fading. The size of such gray areas depends on the environment [12].
- Even simple flooding does not behave as expected [23].
- Experiments are time-consuming and expensive [55,51,50].
- 802.11 radio interfaces have a circular *gray zone* at the border of the transmission range in which broadcasts can be received but unicasts cannot [53].
- Current simulators are not accurate because the assumptions on which simulators are built are too simple, therefore simulation results can differ significantly from real-world experiments [29,44].
- Packet delivery is influenced by the distance of the nodes from the ground [4].
- If multiple TCP connections from one source or to one sink are present, one-hop connections capture nearly all of the available bandwidth [45].
- Switching on all nodes in an ad-hoc network at the same time can overload the network [82].
- Battery power and wireline power supply are a bottleneck during experiments [51].
- Emulation tools like macfilter are essential to save time during the preparation of an experiment [55]. The importance of this is underlined by the number of implementations existing under different names: powerwave [14], APE mackill [5], MobiEmu [88], fleetnet packet suppression mechanism [59], FRANC virtual networks [11] and the MAC filter used in [32].
- Every tool which is used should be tested for its influence on the experiment, e.g., tcpdump is reported to consume lots of resources and may have an impact on the performance of the investigated routing protocol [55].
- Packets for the control of the routing protocol should be delivered with high priority which can be achieved by implementing multi-level priority queues [55].
- Current 802.11 drivers do not report broken links to upper layers: to use link-layer acknowledgments on higher layers, the driver needs to be patched [34,59].
- A routing protocol that uses hop count as route metric may select suboptimal routes. In particular this has been investigated for mesh networks [17,14,18–20] but also shown for sensor networks [87].

- Two network interfaces of the same type integrated close to each other in one computer interfere regardless of the used channel [20,73].
- Draves et al. [19]: "... static and mobile wireless networks can present two very different sets of challenges, and solutions that work well in one setting are not guaranteed to work just as well in another".

10. Integrating simulation, emulation and real-world experimentation

Cost[8] increases from simulation over emulation to real-world experimentation. If protocols were implemented in a way that allows simulation as well as emulation and real-world experimentation with the same code basis, the advantages can be combined while the disadvantages are avoided: (1) Without the need to reimplement the same code for each step, the work load and also the number of errors is lower. (2) The test setup can be validated before moving on to more expensive tests. (3) Previously unknown effects which occur during the test can be integrated in the preceding steps. (4) As it is a step by step approach, wrong assumptions and unknown effects can be isolated easier. (5) The approach provides full realism at low overall costs while it still enables the rapid testing of an idea in a simulator.

We call this *SER integration* (simulation, emulation, real-world integration). Existing SER integration approaches can be classified as follows:

1. Run encapsulated code and either use a packet converter between real world and simulation format [76] or encapsulate the packets [16].
2. Write the code by using an API available in the simulator as well as in reality:
   (a) Integrate the API in an existing simulator: nsclick [61], GEA [33].
   (b) Write a custom-made simulator that supports the API: SURAN [7], Rooftop CPT (used by WINGS [24] and GloMo DAWN [70]), "user level framework for ad hoc routing" [3], the routing protocol evaluation presented in [29,49], TOSSIM [47], EmStar/EmSim [25], EmTOS [26]
3. Port the code manually: [75,49,66,71].

[8] Costs in this context include costs for soft- and hardware as well as time for coding, porting software and human resources.

The approaches (1) and (2a) seem the most promising as they allow to use a well established network simulator that normally contains a variety of protocols and radio layer models without making changes to the code.

11. Conclusions and outlook

The wealth of unanticipated results and information gained through real-world experiments shows that protocols and algorithms for mobile ad-hoc networks *must* be evaluated in real-world settings. Simulation and emulation are valuable tools but they cannot replace experiments.

At the same time experimentation is not yet a mature methodology in the context of ad-hoc networks: results are often non reproducible and hard to explain. In most cases it is nearly impossible to validate the measurements and to isolate external influences from the actual behavior of the investigated algorithm. Furthermore there are no benchmark settings and there exists no "best-practice" for conducting experiments. This makes it very hard to compare the results of experiments from different research groups.

A significant effort to solve these problems is necessary to provide credible and comparable results and to encourage researchers to validate their ideas in real-world settings. This will certainly include testbeds such as APE that should furthermore support SER integration but it will also require research on how to conduct experiments. Most likely the ad-hoc network community could learn a lot about how to address these issues from the natural sciences where a very established methodology for conducting and evaluating experiments has been developed for a long time.

References

[1] N. Abramson, The ALOHA system—another alternative for computer communications, in: Proc. of the Fall 1970 AFIPS Computer Conference, 1970, pp. 281–285.

[2] D. Aguayo, J. Bicket, S. Biswas, G. Judd, R. Morris, Link-level measurements from an 802.11b mesh network, SIGCOMM Comput. Commun. Rev. (CCR) 34 (4) (2004) 121–132.

[3] J. Allard, P. Gonin, M. Singh, G.G. Richard, A user level framework for ad hoc routing, in: Proc. of the 27th Annual IEEE Conf. on Local Computer Networks (LCN), 2002, pp. 13–19.

[4] G. Anastasi, E. Borgia, M. Conti, E. Gregori, A. Passarella, Understanding the real behavior of mote and 802.11 ad hoc

networks: an experimental approach, Pervasive Mobile Comput. 1 (2) (2005) 237–256.

[5] How to build, install and run the APE testbed. TeX-file in the APE distribution. Available from: <http://apetestbed.sourceforge.net/>.

[6] A. Arora, R.R. et al. ExScal: Elements of an extreme scale wireless sensor network, in: Proc. of the 11th Int. Conf. on Embedded and Real-Time Computing Systems and Applications (RTCSA), 2005.

[7] D.A. Beyer, Accomplishments of the DARPA SURAN program, in Proc. of the Military Communications Conference (MILCOM), September 1990.

[8] J. Bicket, D. Aguayo, S. Biswas, R. Morris, Architecture and evaluation of an unplanned 802.11b mesh network, in: Proc. of the Eleventh Annual Int. Conf. on Mobile Computing and Networking (MobiCom), ACM Press, 2005, pp. 31–42.

[9] R. Bruno, M. Conti, E. Gregori, Mesh networks: Commodity multi-hop ad hoc networks, IEEE Commun. Mag. 43 (3) (2005) 123–131.

[10] C.T. Calafate, R.G. Garcia, P. Manzoni, Optimizing the implementation of a MANET routing protocol in a heterogeneous environment, in: Proc. of the Eighth IEEE Symp. on Computers and Communications (ISCC), June 2003.

[11] D. Cavin, Y. Sasson, A. Schiper, FRANC: A lightweight java framework for wireless multihop communication, Technical report, EPFL, Lausanne, 2003.

[12] A. Cerpa, N. Busek, D. Estrin, SCALE: A tool for Simple Connectivity Assessment in Lossy Environments, Technical Report 21, Center for Embedded Networked Sensing, University of California, Los Angeles (UCLA), September 2003.

[13] W. Chao, J.P. Macker, J.W. Weston, Naval research lab's mobile network emulator, January 2003. Avaialble from: <http://downloads.pf.itd.nrl.navy.mil/proteantools/MNE-1.5.pdf>.

[14] K.-W. Chin, J. Judge, A. Williams, R. Kermode, Implementation experience with MANET routing protocols, SIGCOMM Comput. Commun. Rev. (CCR) 32 (5) (2002) 49–59.

[15] J. Crowcroft, M. Conti, A. Passarella, (Eds.), in: Proc. of the IEEE ICPS Workshop on Multi-hop Ad hoc Networks: from theory to reality (REALMAN), July 2005.

[16] P. De, A. Raniwala, S. Sharma, T. Chiueh, MiNT: A miniaturized network testbed for mobile wireless research, in: Proc. of the IEEE INFOCOM, 2005.

[17] D.S.J. De Couto, D. Aguayo, B.A. Chambers, R. Morris, Performance of multihop wireless networks: Shortest path is not enough, in: Proc. of the First Workshop on Hot Topics in Networks (HotNets-I), October 2002.

[18] D.S.J. DeCouto, D. Aguayo, J. Bicket, R. Morris, A high-throughput path metric for multi-hop wireless routing, in: Proc. of the Ninth Annual International Conference on Mobile Computing and Networking (MobiCom), September 2003, pp. 134–146.

[19] R. Draves, J. Padhye, B. Zill, Comparison of routing metrics for static multi-hop wireless networks, in: Proc. of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM), 2004, pp. 133–144.

[20] R. Draves, J. Padhye, B. Zill, Routing in multi-radio, multi-hop wireless mesh networks, in: Proc. of the Tenth Annual International Conference on Mobile Computing and Networking (MobiCom), 2004, pp. 114–128.

[21] A. Farago, V.R. Syrotiuk, Merit: A unified framework for routing protocol assessment in mobile ad hoc networks, in: Proc. of the Seventh Annual International Conference on Mobile Computing and Networking (MobiCom), 2001, pp. 53–60.

[22] J. Flynn, H. Tewari, D. O'Mahony, JEmu: A real time emulation system for mobile ad hoc networks, in: Proc. of the First Joint IEI/IEE Symposium on Telecommunications Systems Research, November 2001.

[23] D. Ganesan, B. Krishnamachari, A. Woo, D. Culler, D. Estrin, S. Wicker, Complex behavior at scale: An experimental study of low-power wireless sensor networks, Technical Report UCLACSD-TR 02-0013, Computer Science Department, UCLA, 2002.

[24] J. Garcia-Luna-Aceves, Wireless Internet gateways (WINGs) for the Internet, Technical report, University of California, Santa Cruz, 2001.

[25] L. Girod, J. Elson, A. Cerpa, T. Stathopoulos, N. Ramanathan, D. Estrin, EmStar: a Software Environment for Developing and Deploying Wireless Sensor Networks, in: Proc. of the 2004 USENIX Technical Conference, 2004.

[26] L. Girod, T. Stathopoulos, N. Ramanathan, J. Elson, D. Estrin, E. Osterweil, T. Schoellhammer, A system for simulation, emulation, and deployment of heterogeneous sensor networks, in: Proc. of the 2nd international conference on Embedded networked sensor systems (SenSys), 2004, pp. 201–213.

[27] The GloMoSim simulation environment. Available from: <http://pcl.cs.ucla.edu/projects/glomosim/>.

[28] J. Gomez, A.T. Campbell, M. Naghshineh, C. Bisdikian, PARO: supporting dynamic power controlled routing in wireless ad hoc networks, Wireless Networks 9 (5) (2003) 443–460.

[29] R.S. Gray, D. Kotz, C. Newport, N. Dubrovsky, A. Fiske, J. Liu, C. Masone, S. McGrath, Y. Yuan, Outdoor experimental comparison of four ad hoc routing algorithms, in: Proc. of the 7th ACM Int. Symp. on Modeling, analysis and simulation of wireless and mobile systems (MSWiM), 2004, pp. 220–229.

[30] H. Hartenstein, H. Füßler, M. Mauve, W. Franz, Simulation Results and Proof-of-Concept Implementation of the FleetNet Position-Based Router, in: Proc. of the IFIP-TC6 8th Int. Conf. on Personal Wireless Communications (PWC), September 2003, pp. 192–197.

[31] T. He, S. Krishnamurthy, J.A. Stankovic, T. Abdelzaher, L. Luo, R. Stoleru, T. Yan, L. Gu, J. Hui, B. Krogh, Energy-efficient surveillance system using wireless sensor networks, in: Proc. of the Second Int. Conf. on Mobile systems, applications, and services (MobiSys), 2004, pp. 270–283.

[32] M. Heissenbüttel, T. Braun, T. Roth, T. Bernoulli, GNU/Linux Implementation of a Position-based Routing Protocol, in: Proc. of the IEEE ICPS Workshop on Multi-hop Ad hoc Networks: from theory to reality (REALMAN), July 2005.

[33] A. Herms, D. Mahrenholz, GEA: A Method for Implementing and Testing of Event-driven Protocols, in: Proc. of the First Int. Workshop on Wireless Mesh Networks (MeshNets), July 2005.

[34] Y.-C. Hu, D.B. Johnson, Design and demonstration of live audio and video over multihop wireless ad hoc networks, in: Proc. of the Military Communications Conference (MILCOM), October 2002.

[35] S. Jadhav, T. Brown, S. Doshi, D. Henkel, R. Thekkekunnel, Lessons learned constructing a wireless ad hoc network test bed, in: Proc. of the 1st Workshop on Wireless Network Measurements (WINMee), April 2005.

[36] J. Jubin, J.D. Turnow, The DARPA packet radio network protocols, in: Proc. of the IEEE 75, January 1987, pp. 21–32.

[37] G. Judd, P. Steenkiste, Repeatable and realistic wireless experimentation through physical emulation, SIGCOMM Comput. Commun. Rev. (CCR) 34 (1) (2004) 63–68.

[38] J.T. Kaba, D.R. Raichle, Testbed on a desktop: strategies and techniques to support multi-hop MANET routing protocol development, in: Proc. of the 2nd ACM Int. Symp. on Mobile ad hoc networking and computing (MobiHoc), 2001, pp. 164–172.

[39] R.E. Kahn, S.A. Gronemeyer, J. Burchfiel, R.C. Kunzelman, Advances in packet radio technology, in: Proc. of the IEEE 66, November 1978, pp. 1468–1496.

[40] V. Kawadia, Y. Zhang, B. Gupta, System services for ad-hoc routing: Architecture, implementation and experiences, in: Proc. of the First Int. Conf. on Mobile systems, applications, and services (MobiSys), May 2003.

[41] Q. Ke, D. Maltz, D.B. Johnson, Emulation of multi-hop wireless ad hoc networks, in: Proc. of 7th Int. Workshop on Mobile Multimedia Communications (MoMuC), 2000.

[42] E. Kohler, R. Morris, B. Chen, J. Jannotti, The Click Modular Router, ACM Trans. Comput. Syst. 18 (3) (2000) 263–297.

[43] K. Konolige, C. Ortiz, R. Vincent, A. Agno, Michael, Eriksen, B. Limketkai, M. Lewis, L. Briesemeister, E. Ruspini, D. Fox, J. Ko, B. Stewart, L. Guibas, DARPA software for distributed robotics, Technical report, DARPA, December 2002.

[44] D. Kotz, C. Newport, R.S. Gray, J. Liu, Y. Yuan, C. Elliott, Experimental evaluation of wireless simulation assumptions, in: Proc. of the 7th ACM Int. Symp. on Modeling, analysis and simulation of wireless and mobile systems (MSWiM), October 2004, pp. 78–82.

[45] A. Laouiti, C. Adjih, Mesures des performances du protocole OLSR, IEEE SETIT, March 2003. (in French).

[46] B. Leiner, D. Nielson, F. Tobagi, (Eds.), in: Proc. of the IEEE Special Issue, packet radio networks 75 (January) (1987).

[47] P. Levis, N. Lee, M. Welsh, D. Culler, TOSSIM: accurate and scalable simulation of entire tinyOS applications, in: Proc. of the 1st Int. Conf. on Embedded networked sensor systems (SenSys), 2003, pp. 126–137.

[48] T. Lin, S. Midkiff, J.S. Park, A dynamic topology switch for the emulation of wireless mobile ad hoc networks, in: Proc. of IEEE Wireless Local Network (WLN), jointed with LCN 2002, (2002).

[49] J. Liu, Y. Yuan, D.M. Nicol, R.S. Gray, C.C. Newport, D. Kotz, L.F. Perrone, Simulation validation using direct execution of wireless ad-hoc routing protocols, in: Proc. of the Workshop on Parallel and Distributed Simulation (PADS), May 2004, pp. 7–16.

[50] J. Liu, Y. Yuan, D.M. Nicol, R.S. Gray, C.C. Newport, D. Kotz, L.F. Perrone, Validation of wireless models using direct-execution simulation of ad-hoc routing protocols, Simulation: Transactions of The Society for Modeling and Simulation International, January 2005.

[51] D. Lundberg, Ad hoc protocol evaluation and experiences of real world ad hoc networking, Master's thesis, University Uppsala, 2002.

[52] H. Lundgren, D. Lundberg, J. Nielsen, E. Nordström, C. Tschudin, A large-scale testbed for reproducible Ad Hoc protocol evaluations, in: Proc. of the IEEE Wireless Communications and Networking Conference (WCNC), March 2002, pp. 337–343.

[53] H. Lundgren, E. Nordström, C. Tschudin, Coping with communication gray zones in IEEE 802.11b based ad hoc networks, in: Proc. of the fifth ACM Int. workshop on Wireless mobile multimedia (WOWMOM), September 2002, pp. 49–55.

[54] D. Maltz, J. Broch, D. Johnson, Quantitative lessons from a full-scale multi-hop wireless ad hoc network testbed, in: Proc. of the IEEE Wireless Communications and Networking Conference (WCNC), September 2000.

[55] D.A. Maltz, J. Broch, D.B. Johnson, Experiences designing and building a multi-hop wireless ad hoc network testbed, Technical Report CMU-CS-99-116, School of Computer Science, Carnegie Mellon University, 1999.

[56] M. Matthes, H. Biehl, M. Lauer, O. Drobnik, MASSIVE: An Emulation Environment for Mobile Ad-Hoc Networks, in: Proc. of the Second Annual Conference on Wireless On-demand Network Systems and Services (WONS), 2005, pp. 54–59.

[57] Melbourne wireless. Available from: <http://www.melbournewireless.org.au>.

[58] M. Möske, Real-World Evaluation of a Vehicular Ad Hoc Network using Position-Based Routing. Master's thesis, Department of Computer Science, University of Mannheim, 2003.

[59] M. Möske, H. Füßler, H. Hartenstein, W. Franz, Performance Measurements of a Vehicular Ad Hoc Network, in: Proc. of the IEEE Vehicular Technology Conference (VTC), May 2004.

[60] J. Nagler, F. Kargl, S. Schlott, M. Weber, Ein Framework für MANET Routing Protokolle. In 1. deutscher Workshop über Mobile Ad-Hoc Netzwerke (WMAN), 2002, pp. 153–165 (in German).

[61] M. Neufeld, A. Jain, D. Grunwald, Nsclick: Bridging Network Simulation and Deployment, in: Proc. of the 5th ACM Int. Symp. on Modeling, analysis and simulation of wireless and mobile systems (MSWiM), September 2002, pp. 74–81.

[62] B.D. Noble, M. Satyanarayanan, G.T. Nguyen, R.H. Katz, Trace-based mobile network emulation, in: Proc. of the 1997 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM), 1997, pp. 51–61.

[63] The ns-2 network simulator. Available from: <http://www.isi.edu/nsnam/ns/>.

[64] NSF workshop on network research testbeds, October 2002. Available from: <http://www-net.cs.umass.edu/testbed_workshop/>.

[65] The ORBIT testbed. Available from: <http://www.orbit-lab.org>.

[66] E. Osipov, C. Tschudin, A Path Density Protocol for MANETs, in: Proc. of the IEEE ICPS Workshop on Multi-hop Ad hoc Networks: from theory to reality (REALMAN), July 2005, pp. 69–76.

[67] C.E. Perkins, Ad Hoc Networking, Addison Wesley Professional, 2001.

[68] K. Ramachandran, K. Almeroth, E. Belding-Royer, A novel framework for the management of large-scale wireless

network testbeds, in: Proc. of the 1st Workshop on Wireless Network Measurements (WINMee), April 2005.
[69] K. Ramachandran, E. Belding-Royer, K. Almeroth, DAMON: A distributed architecture for monitoring multi-hop mobile networks, in: Proc. of IEEE Int. Conf. on Sensor and Ad hoc Communications and Networks (SECON), October 2004.
[70] R. Ramanathan, R. Hain, An ad hoc wireless testbed for scalable, adaptive QoS support, in: Proc. of the IEEE Wireless Communications and Networking Conference (WCNC), September 2000, pp. 998–1002.
[71] R. Ramanathan, J. Redi, C. Santivanez, D. Wiggins, S. Polit, Ad hoc networking with directional antennas: A complete system solution, in: Proc. of the IEEE Journal on Selected Areas In Communications, March 2005.
[72] D. Raychaudhuri, I. Seskar, M. Ott, S. Ganu, K. Ramachandran, H. Kremo, R. Siracusa, H. Liu, M. Singh, Overview of the ORBIT Radio Grid Testbed for Evaluation of Next-Generation Wireless Network Protocols, in: Proc. of the IEEE Wireless Communications and Networking Conference (WCNC), March 2005.
[73] J. Robinson, K. Papagiannaki, C. Diot, X. Guo, L. Krishnamurthy, Experimenting with a multi-radio mesh networking testbed, in: Proc. of the 1st Workshop on Wireless Network Measurements (WINMee), April 2005.
[74] MIT Roofnet. Avaialble from: <http://www.pdos.lcs.mit.edu/roofnet/>.
[75] E.M. Royer, C.E. Perkins, An Implementation Study of the AODV Routing Protocol, in: Proc. of the IEEE Wireless Communications and Networking Conference (WCNC), September 2000.
[76] A.K. Saha, K. To, S. PalChaudhuri, S. Du, D.B. Johnson, Physical implementation and evaluation of ad hoc network protocols using unmodified simulation models, in: Proc. of the ACM SIGCOMM Asia Workshop, April 2005.
[77] S. Sanghani, T.X. Brown, S. Bhandare, S. Doshi, EWANT: The emulated wireless ad hoc network testbed, in: Proc. of the IEEE Wireless Communications and Networking Conference (WCNC), March 2003.
[78] M. Singh, M. Ott, I. Seskar, P. Kamat, ORBIT Measurements Framework and Library (OML): Motivations, Design, Implementation, and Features, in: Proc. of the First Int. IEEE/Create-Net Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities (Tridentcom), February 2005.
[79] M. Takai, J. Martin, R. Bagrodia, Effects of wireless physical layer modeling in mobile ad hoc networks, in: Proc. of the 2nd ACM Int. Symp. on Mobile ad hoc networking and computing (MobiHoc), October 2001, pp. 87–94.
[80] C.-K. Toh, R. Chen, M. Delwar, D. Allen, Experimenting with an ad hoc wireless network on campus: insights and experiences, SIGMETRICS Perform. Eval. Rev. 28 (3) (2000) 21–29.
[81] N.H. Vaidya, J. Bernhard, V. Veeravalli, P.R. Kumar, R. Iyer, Illinois wireless wind tunnel: A testbed for experimental evaluation of wireless networks, Technical report, Wireless Networking Group, University of Illinois at Urbana–Champaign, August 2004.
[82] R. Vincent, Re: Centibots: Communication among robots, personal communication, June 2004.
[83] B. White, J. Lepreau, S. Guruprasad, Lowering the barrier to wireless and mobile experimentation, SIGCOMM Comput. Commun. Rev. (CCR) 33 (1) (2003) 47–52.
[84] The WHYNET project. Available from: <http://pcl.cs.ucla.edu/projects/whynet/>.
[85] Wikipedia, mesh networks. Available from: <http://en.wikipedia.org/wiki/Wireless_community_network>.
[86] The Wireless Leiden Foundation. Available from: <http://wirelessleiden.nl>.
[87] M. Yarvis, W. Conner, L. Krishnamurthy, J. Chhabra, B. Elliott, A. Mainwaring, Real-world experiences with an interactive ad hoc sensor network, in: Proc. of the Int. Workshop on Ad Hoc Networking (IWAHN), August 2002, pp. 143–151.
[88] Y. Zhang, W. Li, An integrated environment for testing mobile ad-hoc networks, in: Proc. of the 3rd ACM Int. Symp. on Mobile ad hoc networking and computing (MobiHoc), 2002, pp. 104–111.
[89] J. Zhao, R. Govindan, Understanding packet delivery performance in dense wireless sensor networks, in: Proc. of the 1st Int. Conf. on Embedded networked sensor systems (SenSys), 2003, pp. 1–13.

Wolfgang Kiess studied computer science at the University of Mannheim from 1997 to 2003. In the academic year 2000/2001 he stayed at the University of Nice Sophia Antipolis, France. He received the M.S. degree in March 2003, the subject of his diploma thesis was "Hierarchical Location Service for mobile ad-hoc networks". Currently, he is a Ph.d. student at the University of Düsseldorf.

Martin Mauve received the M.S. and Ph.D. degrees in computer science from the University of Mannheim, Germany, in 1997 and 2000 respectively.

From 2000 to 2003, he was an Assistant Professor at the University of Mannheim. In 2003, he joined the Heinrich-Heine-University, Düsseldorf, Germany, as a Full Professor and Head of the research group for computer networks and communication systems. His research interests include distributed multimedia systems, multimedia transport protocols, mobile ad-hoc networks and inter-vehicle communication.