Towards Trustworthy Simulation of Wireless MAC/PHY Layers: A Comparison Framework

Laura Marie Feeney
Swedish Institute of Computer Science
lmfeeney@sics.se

ABSTRACT

Simulation is an essential tool for evaluating the performance of wireless protocols and applications because of its cost effectiveness and its high level of control over the simulated system. As the research community envisions increasingly complex applications in areas like mobile social networks and wireless sensor networks, the need for trustworthy simulation of core communication protocols increases.

This paper presents a comparative evaluation of several IEEE 802.11 and IEEE 802.15.4 simulators. The paper defines a set of simple experiments that are intended to highlight handling of backoff and contention.

A wide variation in results is observed. Although this is partially explained by differences in wireless communication models, the experiments also reinforce existing concerns about the reliability of simulation results. A long term goal is that this methodology will encourage the adoption of common test and evaluation scenarios. This would lead to better understanding of simulation behavior and help to improve the quality of – and confidence in – simulation results.

Categories and Subject Descriptors

I.6 [**Computing Methodologies**]: Simulation and Modeling

Keywords

simulation, wireless, MAC/PHY, OMNeT++

1. INTRODUCTION

Testbed experiments are often used in evaluating wireless protocols and applications and are generally viewed as the gold standard for performance evaluation. However, testbed results reflect the irregularities of a specific radio environment and the effects of external interferences, as well as complex cross-layer interactions in the communicating devices. Although testbeds provide "ground truth", this lack of control can make it extremely difficult to reason about observed results.

By contrast, simulation provides complete visibility and control of the environment in which the simulated system operates. This allows the user to explore a wide range of scenarios, to create structured experiments, and to reason carefully about system behavior. But it is not possible to take advantage of these capabilities, unless the underlying simulations are accurate and well understood.

This is especially important in areas like mobile social networks and wireless sensor networks because their low power device-to-device communication architecture results in very complex interactions between wireless devices. Moreover, these networks are very application-oriented and their design and performance can depend heavily on external processes like human interaction or natural phenomena. As a result, there is increasing participation from research communities whose expertise lies outside wireless communication. A researcher studying information dissemination in mobile social networks or distributed data processing algorithms for monitoring air quality may not have – and should not need – expertise in low-level details of wireless simulation.

There is therefore a growing demand for tools that provide trustworthy components for simulation of core wireless communication protocols. However, there is currently no consensus of high confidence in wireless simulation results, due to concerns about software quality and validation. The community also lacks a high-level systematic understanding of the behavior of various wireless simulation environments. Attempts to validate simulators with testbed experiments have generally proven somewhat unsatisfying, largely due to limitations in parameterizing simulations to reflect real radio hardware and environments.

This paper presents a comparative evaluation of several IEEE 802.11 and IEEE 802.15.4 simulators. The paper defines a set of simple, well-designed experiments that highlight handling of backoff and contention in the presence of hidden terminals. The issue of parameterization is carefully addressed: Comparable scenarios are defined and normalized in terms of nominal transmit range, which encapsulates the variation in wireless communication models.

The results demonstrate significant variation in behavior. Some of these reflect legitimate differences between simulators, while others reveal issues that need further investigation. It is hoped that the availability of simple, well-defined, and revealing test and evaluation scenarios will improve quality and confidence in simulation results. Specifically, this paper:

MSWiM'12, October 21–25, 2012, Paphos, Cyprus.

- Characterizes the structure and parameterization of wireless network simulations, with particular attention to testbed vs simulation experiments.

- Proposes a small set of well-designed scenario-based experiments that demonstrate the behavior of key components of the simulation. Nominal transmit range is used as the basis of comparable simulations.

- Compares five OMNeT++-based simulators supporting IEEE 802.11 and IEEE 802.15.4 wireless standards and observes some significant variation. (Note that the scenarios themselves are not OMNeT++ specific.)

- Concludes that diversity of simulators is generally positive, not least because it enables this kind of cross-comparison. The long-term goal is that simulation quality is improved by the wider adoption of common test and evaluation scenarios, leading to increased confidence in simulation results.

2. RELATED WORK

This section briefly presents the rather sparse body of existing work comparing simulation results and proposing common environments. The discussion focuses on work comparing simulation and testbed results and its implications for comparing simulation results directly. These ideas are important for understanding the problem of creating sound comparison experiments in both areas.

Comparing simulations There is a large survey literature (e.g. [12]) comparing the features and usability of various wireless simulation environments. But there have been very few large-scale comparisons of the results obtained from different simulators

In 2002, paper [4] reported significantly different results when simulating a simple IEEE 802.11 broadcast flooding scenario in ns-2, OPNET and Glomosim. An unpublished technical report [6] from 2008 compares ns-2 (versions 2.26, 2.29 and 2.31) and OMNeT++'s inet and mobility-fw. Like our work, this work emphasizes simple hidden terminal scenarios. However, it attempts to create comparable scenarios by adjusting radio sensitivity parameters, which seems potentially problematic because these are fixed hardware properties. The results suggest that inet is more permissive than mobility-fw in hidden terminal scenarios. This is the opposite of the result seen in our study of more recent software.

Comparisons with testbeds It is often suggested that simulations should be validated by experimental measurements. Unfortunately, there have been few satisfying results comparing simulation and testbed results.

It is hard to parameterize simulation models for a specific communication environment and hardware. Visible effects, such as packet reception rate or link quality indication reflect the combined effect of many channel and hardware parameters. When a single simulation parameter, such as path-loss, is tuned to match testbed results, there is risk of over-fitting.

It is also sometimes possible to find a parameterization that gives a fairly close match in high-level statistics such as throughput or packet reception ratio (PRR), while detailed statistics such as latency or jitter remain highly inconsistent. This inconsistency makes it difficult to use results to reason about low-level internal behavior. Moreover, even the best work does not seem to provide robust conclusions: Does a parameterization obtained in one environment give similar results when tested in another, similar environment?

One way to try minimize the minimize the complexity of the wireless environment and the associated parameterization problems is to use a very restricted testbed scenario. This approach is sometimes helpful, but in other cases it reduces the usefulness of the results, without significantly reducing complexity.

Comparisons that use specialized hardware to essentially eliminate all wireless effects and focus on the MAC protocol implementation often show very good results. For example, paper [3] compares a four node 802.11g network with an OMNeT++ simulator (based on inet + cocorada). The nodes were in an anechoic chamber and separated by only 0.5-1.5m, giving near perfect connectivity. The simulation results suggested that the IEEE 802.11g protocol simulation itself was quite accurate. Testbed and simulation results for throughput were nearly identical and in 99% of cases, even inter-packet times were in agreement. Deviation in the other 1% of measurements suggest some occasional phenomena were not accounted for. Obviously, this approach has limited relevance for simulating wireless communication, but it is very valuable for investigating the accuracy of protocol models and inconsistencies in vendor implementations.

Other experiments use fully connected networks with short distances and and unobstructed radio environments (e.g. sports fields). This eliminates the effects of hidden terminals and unstable links. It also reduces the relevance of the results, because these are essential factors in the performance of wireless networks and (equivalently) a large proportion of simulation code is not exercised in such a test.

Paper [1] compares ns-2 and two OMNeT++ simulators, Castalia and the MAC simulator (an older framework of which MiXiM is a descendant). The experiment used a fully connected network of 12 Tmote-Sky (IEEE 802.15.4) nodes, deployed outdoors in a 5m area minimizing reflections and interference. Unlike [10], the channel parameters (path-loss exponent = 2.4 and $\sigma = 4$) are defined *a priori* (radio hardware parameters were not discussed). The authors claim that for ns-2, the shadowing component has no effect on results, while for Castalia log-normal shadowing significantly underestimates the network performance. Overall, the simplest communication models for both ns-2 and Castalia give results within 10-30% of testbed results.

Paper [10] compares ns-2 (ns-2.29) simulation of IEEE 802.11 with a multi-hop indoor network of 16 nodes. The simulation was parameterized by simulating the network for various values of path-loss and selecting the value that gave the best overlap between the set of connected node pairs in the simulated and testbed topologies. It is not clear whether the (rather large) chosen path-loss value of 5.7 reflects the actual path-loss or whether it is acting more like an arbitrary tuning parameter. For video traffic, good agreement was seen for packet delivery ratios, but latencies varied by much as -68% and +125%, depending on hopcount.

Paper [14] compares ns-2 (ns-2.34) and Qualnet to results obtained in indoor and outdoor testbeds using a variety of IEEE 802.11 mesh network hardware. There is little discussion of parameterization, but neither simulator was able to model the behavior of even simple UDP traffic, especially with respect to environmental effects. Later experiments suggested ns-2 path-loss parameters of 2.4 (indoor) and 2.5 (outdoor), but these values were not used in further com-

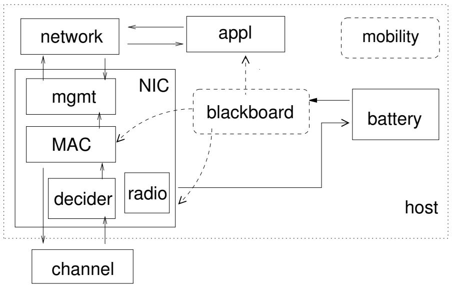


Figure 1: Example of a simplified wireless host.

simulator	version	IEEE 802.11	IEEE 802.15.4	cites
inet	20110225	b, AP	no	133
MiXiM	2.1 (pre-release)	b, DCF	csma	218
inetmanet	151cd4	a/b/g/e, AP	PAN only	80
Castalia	3.2	no	PAN only	200
mobility-fw	2.0p3 (extended)	b, DCF	csma	137

Table 1: The simulators used in this paper are all available via www.omnetpp.org. The citation counts are *very rough* estimates based on Google Scholar.

parison. The authors also note significant effects related to bitrate adaptation, dual antennas and flow interference.

Another way to avoid parameterization issues is to use testbed-specific empirical data. Paper [9] uses a detailed Tossim simulation of a 22-node multi-hop sensor network (mica2-like nodes). Each link in the simulated network was parameterized using signal strength measurements for the corresponding link in the testbed. When using an SNR-based communication model and empirical signal strength measurements, there was very good agreement in packet delivery ratio for several sensor network protocols. But more detailed metrics, such as latency, showed less agreement.

Similarly, paper [5] compares the OMNeT++ MAC simulator with a fully connected four-node network of Tmote-Sky nodes (IEEE 802.15.4). Each link was parameterized with frame reception probabilities based on testbed measurements. This technique gave good agreement for throughput, but the testbed used for parameterization and for testing seem to have been identical, making the results unclear.

Shared evaluation scenarios One contribution of this paper is the idea that well-designed common test and evaluation scenarios will allow developers to detect problems and may increase the user community's confidence in simulation results. The idea of creating a common basis for comparing simulation environments is also explored in [16]. However, this work focuses on configuration and mobility modeling and does not present comparative simulation results.

3. SIMULATORS AND TOOLS

OMNeT++[15] is a public source, highly modular, general purpose discrete event simulation environment. An OMNeT++ simulation is a defined as a network (directed graph) of modules that exchange messages and event notifications. By itself, OMNeT++ does not provide any domain-specific functionality: It "merely" provides the discrete event scheduling and message passing engine, the network and message description languages, and support for configuring and managing simulations, gathering and analyzing statistics, and visualization.

The community has developed OMNeT++-based simulators for a wide range of application areas. OMNeT++'s modular structure often makes it feasible to integrate different simulators, making it potentially very well-suited for the kind of multidisciplinary wireless network research described earlier. The modular structure of the MiXiM wireless host is shown in Figure 1.

The five OMNeT++-based wireless network simulators used in this work are described below. Although the OMNeT++ architecture naturally leads to some similarity of structure, they are largely independent in their design and implementation. This combination of independent simulators and a common management framework is ideal for our experiments. However, the approach and scenarios described here are not specific to OMNeT++.

The simulators used in this experiment are inet[13], an inet fork called inetmanet[13], the mobility-framework [7] and its successor MiXiM [11, 17], and Castalia [2], described in more detail below and in Table 1. The simulators are all open source and freely available for (at least) academic use and have been widely cited in the literature.

Inet is one of the largest and most important OMNeT++ network simulation frameworks and its development team includes some OMNeT++ core developers. Inet mainly provides support for (wired) Internet protocols, but it also includes IEEE 802.11b in both ad hoc and AP modes.

Inetmanet is an independent fork of inet that focuses on wireless networks and MANET protocols and includes models ported from a wide variety of sources. It includes support for IEEE 802.15.4 PANs (but not unslotted CSMA), as well as IEEE 802.11a/e/g and IEEE 802.16. Its IEEE 802.11b implementation shares code with inet, so only inet results are reported.

MiXiM is the successor to several earlier OMNeT++ based simulators, including mobility-fw (below). It focuses on radio signal modeling, supporting not only IEEE 802.11b (ad hoc mode only) and IEEE 802.14.5 (unslotted CSMA only), but also UWB and multi-channel models.

Mobility-fw is the oldest and generally fastest and simplest of the simulators and is based on an earlier version of OMNeT++. The simulators above all have at least a little bit of mobility-fw in their ancestry, so it is included, despite its age. It includes support for IEEE 802.11b and a separate package, ucsma, is a generic unslotted CSMA model that can be configured to model IEEE 802.15.4.

Castalia has the least in common with the other simulators. It is intended for simulating sensor networks, especially body area networks, and emphasizes detailed modeling of the wireless channel and radio, including specialized proprietary noise mapping functionality (not studied here). Castalia provides support for IEEE 802.15.4 PANs (but not unslotted CSMA) and IEEE 802.15.6 BANs, as well as some wireless sensor network MAC's.

ApplTestTool Implementing comparisons in this paper were made much easier by the OMNeT++ ApplTestTool[8] test framework. A core module of the ApplTestTool is the EnsembleApplLayer, which is a configurable traffic generator and statistics collector. Since this module is an application layer, it is relatively easy to port to each of the OMNeT++ simulation frameworks. For this work, Ensem-

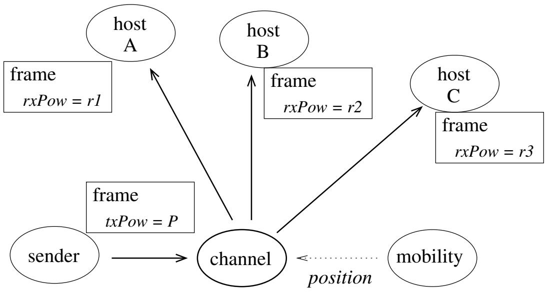


Figure 2: Example of channel propagation.

bleApplLayer was ported to run directly above the NIC/MAC modules in each simulator. This prevented the simulators' very different addressing and ARP implementations from affecting the experiment results.

4. SIMULATING WIRELESS NETWORKS

This section outlines in general terms how wireless networks are modeled, with a focus on OMNeT++-based simulators and their parameterization. For our purposes, this functionality can be divided into three parts: The first models the transmission and propagation of RF signals and the second models their detection and decoding. The third models sending and receiving of data and control frames according to MAC and higher layer management protocols.

Transmission For each outgoing transmission, the propagation model determines the resulting RF energy at each receiver. This computation depends on the antenna (ignored here, since all of the simulators assume omni-directional antennas with unit gain) and channel models.

Figure 2 shows the basic structure of most OMNeT++-based wireless simulators: When a host has determined that it is allowed to access the channel, a host (sub-)module sends a message (i.e. an event) to the global channel control module. This "airframe" message includes the transmit power, which the channel control uses to compute the received signal strength at each of the other hosts.

Transmit power is an example of a parameter that is usually controlled externally by higher layer protocol or application. For both IEEE 802.11 and IEEE 802.15.4, limits on transmit power are defined by the standard and available settings are given in various hardware specifications. Since there is no natural default transmit power, experiments are explicitly configured with consistent values.

Propagation All of the simulators in this paper use analytic, rather than empirical, channel models. These are based on the usual well-known models. Nevertheless, both the range of models and their instantiation vary considerably between simulators. Mobility-fw provides only a simple path-loss model, while MiXiM uses complex data structures supporting signal mapping in time, frequency, and space, though this functionality is not explored fully here. MiXiM, Castalia and inet all provide a log-normal shadowing model, but with non-trivial differences in implementation. This variation complicates the problem of creating "comparable" simulation configurations.

Exponential path loss, with signal strength proportional to $d^{-\alpha}$, is the most significant effect. Because this parameter is common to all models, it is explicitly configured. (MiXiM's breakpoint path-loss model, with two path-loss parameters, is not studied here. But its existence highlights the issue of creating parameter-identical comparisons.)

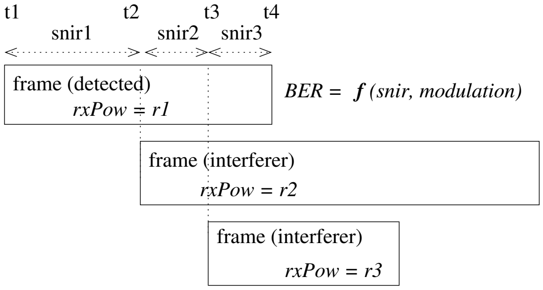


Figure 3: Example of frame reception.

Where possible, this is combined with normal variation with mean μ and standard deviation σ. In general, the default values for these parameters vary between simulators, as there is no agreement on a "typical" environment, so these are also explicitly matched among models that use them. Further parameters are defined as needed, generally following each simulator's defaults. MiXiM's log-normal model takes a time scale parameter, while Castalia's takes a parameter for bi-directional variance.

The channel module sends a copy of the "airframe" message to the appropriate sub-module of each host, annotated with the received signal strength. A host's frame reception model therefore has information about the signal strength of all ongoing transmissions.

All of the simulators do filtering to avoid the cost of computing interferences that are extremely unlikely to affect detection or frame reception. Filtering is intended to have negligible impact on simulation accuracy and is meaningful only in large multi-hop networks, so the default parameters for each simulator's filtering mechanism are used.

Position information is usually managed by the mobility module, but details of mobility modeling and interaction with the channel control module vary considerably between simulators. Only static networks are considered here.

Detection and frame reception The PHY model must determine whether (or which) frame arriving at the receiver is successfully decoded. The model must also determine whether the channel is observed to be clear or busy; this is needed for MAC layers that perform clear channel assessment (CCA). An example of a model in which signal strength is constant, but interference varies over time is shown in Figure 3.

Differences in detection and frame reception models have a large impact on simulation results and the simulators' reception models are even more varied than their transmission and propagation models. This includes not only detection and capture models, but also the detail with which SNIR is modeled over the duration of a frame and the computation that determines whether a frame is correctly received.

The simplest model, mobility-fw, uses a simple SNIR threshold. The first arriving frame is accepted if the minimum SNIR over the duration of the packet is greater than some threshold. Inet uses a similar model. For this model, key parameters are the radio sensitivity and this threshold. As these values are hardware specific, each simulator's default values are used. For IEEE 802.14.5, inetmanet uses a sensitivity parameter of -85dBm (the minimum allowed by the standard) and mobility-fw uses -95dBm (typical of commercial hardware). The threshold value for inetmanet is 4dB, while the mobility-fw uses 5dB (the standard describes 5-6dB as an achievable value for low cost hardware).

Castalia and MiXiM use more complex calculations that model the bit (or symbol) error rate as function of the SNIR and the frame modulation. The two vary somewhat in their level of detail modeling of time-variation of interference and modulation over the duration of the frame, but the underlying computations are well defined. CCA is determined by radio sensitivity, and MiXiM's model also applies start of frame detection (SFD). (The inetmanet reception model can include a BER calculation, but this does not seem to be fully implemented.)

A successfully decoded frame is passed up to the MAC layer and possibly higher layers. Alternatively, the MAC may be notified that the channel has become busy or idle.

This discussion again highlights the complexity surrounding "consistent" parameterization of different simulators: Different models take different parameters or use different semantics for similar parameters. Even models that take the same parameters may define different, but entirely plausible defaults (e.g. standard defined limits vs typical hardware).

MAC protocols Simulating communication over a wireless channel is a challenging problem in finding suitable abstractions for representing complex physical phenomena. By contrast, modeling the handing of data frames in the MAC layer is more straightforward.

The simulation is intended to mirror the operation of well-defined protocols like IEEE 802.15.4 or IEEE 802.11, which are specified by standards documents. Default values for parameters such as header lengths and backoff and retransmission limits are defined in standards documents.

It is reasonable to expect simulators' default configuration should match these defaults. Among the simulators in this work, only inet uses non-default MAC parameters (a smaller initial contention window and large retry limit for IEEE 802.11). These values were not modified for this study.

The main differences between simulators arise from the detail with which NIC internal operations such as processing and radio turnaround are modeled. These timing values parameters are generally defined in the standard or in hardware specifications. Each simulator's defaults are assumed to represent typical hardware and are not modified.

Because protocols are well defined, accurate protocol simulation is largely dependent on good software engineering. However there may still be issues where the specification is ambiguous (e.g. interframe space after failed channel access in IEEE 802.15.4). Clarity is also important for behavior that is defined as implementation dependent (e.g. IEEE 802.11 bitrate adaptation). Similar variation is observed – for similar reasons – among hardware from different vendors.

5. NOMINAL TRANSMIT RANGE

This section describes the radio characterization experiments that are used to determine the nominal transmit range for each configuration. This transmit range is used to create scenarios that are normalized across simulators.

	MiXiM	**mobility fw**	**inet-manet**	**Castalia**
shadowing	$\mu = 0, \sigma = 0$	no	no	$\mu = 0$, $\sigma = 4$ bi-$\sigma = 1.0$
sensitivity	-100dBm	-95 dBm	-85 dBm	-95dBm
threshold	n/a	5dB	4dB	n/a
BER model	yes	no	no	yes
PAN	no	no	yes	yes

Table 2: IEEE 802.15.4 simulation parameters.

	MiXiM	**inet**	**mobility fw**
shadowing	$\mu = 0, \sigma = 10$ t = 1s	$\mu = 0, \sigma = 10$	no
sensitivity	-119 dBm	-85 dBm	-119 dBm
threshold	0.1259 (-9dB)	4dB	-9dB
cwMin	31	7 (data)	31
cwMax	1023	1023	1023
retry (short)	4	7	4
retry (long)	7	7	7

Table 3: IEEE 802.11b simulation parameters.

By now it should be clear that it is not possible to create identical parameterizations for these simulators, due to their very different models of wireless propagation and frame reception. Nominal transmit range is intended to encapsulate the combined effects of the various propagation and frame reception models and parameters. Topologies that are defined in terms of transmit range have similar effective node densities, despite large differences in absolute range. However, this cannot eliminate the inherent differences in propagation and reception models and the topologies may have different interference and other characteristics.

The radio characterization experiment measures the packet reception rate (PRR) sending 1000 frames from a sender to a receiver at each of several distances. The nominal transmit range is defined as the largest distance that gives at least 92% PRR in the absence of interfering transmissions. For IEEE 802.11, the payload is 1024-bytes and for IEEE 802.15.4, it is 100-bytes.

The IEEE 802.11 results are shown in Table 4. The transmit ranges for MiXiM and mobility-fw are in agreement, which is unsurprising given their shared history. The values are large compared to realistic networks, however. The range for both inet and inet with log-normal shadowing configurations are smaller, reflecting the higher SNIR threshold required to successfully receive the frame. Including log-normal shadowing with $\sigma = 10$ increases the nominal range. This is due to the gradual decrease in the PRR, which first drops below 99% at distances similar to the much sharper transition in the simple path loss model.

The IEEE 802.15.4 results are shown in Table 5. Castalia and inetmanet simulations use IEEE 802.15.4 PAN in direct transfer mode, while MiXiM and mobility-fw use unslotted CSMA. However, PAN operation do not significantly affect the radio characterization: The test transmissions are simply slightly delayed to allow association to complete. The transmit ranges, especially for MiXiM and the mobility-fw, seem rather large, but there does not seem to be any clear determining factor.

6. SCENARIO EXPERIMENTS

In this section, we present results from three scenarios. The very stylized sync-star scenario is used to examine contention and backoff behavior and is the core of comparison

	MiXiM ± 2.5m	**MiXiM w/ shadowing** μ=0, σ=10 ± 2.5m	**mobility fw** ± 2.5m	**inet** ± 1m	**inet w/ shadowing** μ=0, σ=10 ± 1m
2 mW					
α=3.0	287.5	418.9	287.5	40.0	62.0
4.0	67.5	92.0	72.5	16.0	22.0
10 mW					
α=3.0	492.5	716.0	492.5	67.0	105.0
4.0	102.5	138.0	102.5	24.0	33.0
50 mW					
α=3.0	837.5	1224.0	847.5	117.0	179.0
4.0	157.5	207.0	157.5	36.0	50.0

Table 4: Nominal transmit ranges IEEE 802.11b @ 2Mbps: 92% frame reception rate with 1024 byte payload.

	MiXiM ± 1m	**mobility-fw** ± .5m	**inet-manet** ± .5m	**Castalia** ± .5m
-25dBm				
α = 2.0	81.5	31.5	9.5	2.5
2.5	33.0	15.5	6.5	2.5
3.0	18.0	9.5	4.5	1.5
-15dBm				
α = 2.0	255.0	99.5	31.5	8.5
2.5	85.0	39.5	15.5	5.5
3.0	41.0	21.5	9.5	4.5
-10dBm				
α = 2.0	453.0	176.5	55.5	15.5
2.5	135.0	62.5	25.5	9.5
3.0	60.0	31.5	14.5	6.5
-5dB m				
α = 2.0	805.0	314.5	99.5	28.5
2.5	212.0	99.5	39.5	14.5
3.0	87.0	46.5	21.5	9.5
0dBm				
α = 2.0	> 1K	559.5	176.6	49.5
2.5	260.0	157.5	62.5	22.5
3.0	103.0	67.5	31.5	13.5

Table 5: Nominal transmit ranges IEEE 802.15.4: 92% frame reception rate over 100 byte payload.

framework. The slightly more realistic WSN-like scenario represents traffic to a central gateway and the AP-like scenario represents traffic to and from an access point.

In many cases, there is considerable variation between simulation results and some speculation is made as to the sources of these differences. But it should be emphasized that the goal of this work is *not* to perform five-way cross-simulator debugging, but rather to demonstrate the value of the comparison scenarios to highlight behavior worth further investigation.

6.1 Sync-star

As the name suggests, this is a star topology, with nodes sending synchronized, periodic transmissions to the central receiver. This scenario is intended to highlight contention and backoff behavior by ensuring that all senders are equidistant from the receiver and simultaneously contending for the channel. The period is such that the contentions are essentially independent experiments; each frame has either succeeded or exhausted its retry opportunities in each period.

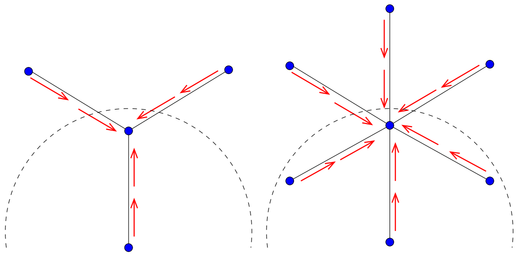


Figure 4: Hidden terminals in the sync-star scenario. The dotted line shows the range. The communication model determines which nodes are able to detect each other's transmissions.

Varying the radius and density of the star controls the existence of hidden terminals (Fig. 4). In these experiments, the radius is either 0.2 × nominal transmit range, giving a fully connected network with high signal strengths, or 0.9 × nominal transmit range, so that nodes on opposite sides of the star are hidden terminals.

The experiment measures the total number of packets received, the packet reception rate and the latency, for networks with varying numbers of senders. The latency is the per-packet application latency. Because this scenario explicitly avoids queuing, the latency reflects channel access time for each frame, plus any processing delays modeled by the simulator. (Note that the 95% CI is for the mean latency across multiple simulation runs, not the variation in latency within a simulation.)

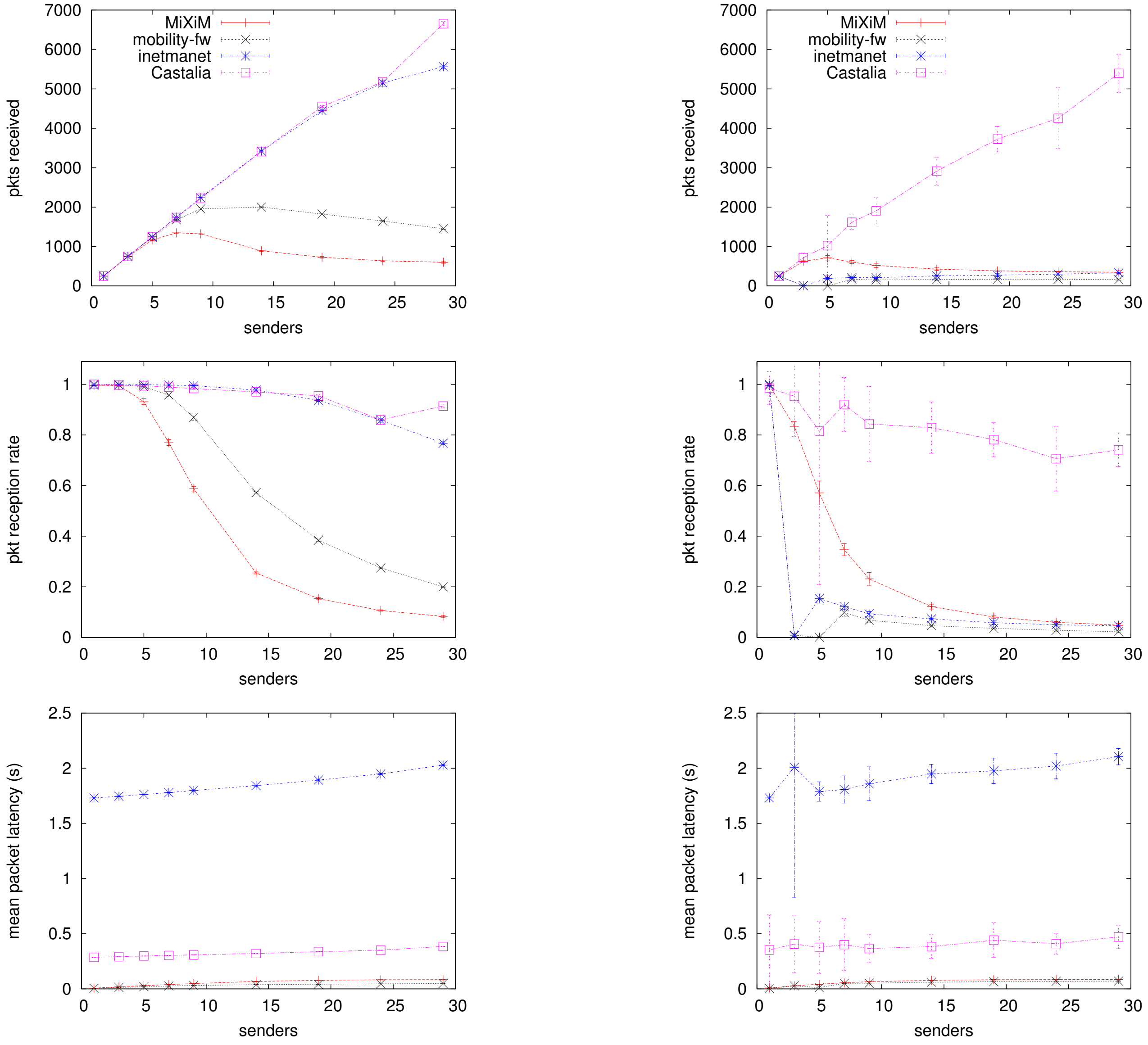
IEEE 802.15.4 The IEEE 802.15.4 results are shown in Figure 5. Each sender generates 250 packets with 100 bytes payload and the experiment is repeated up to 10 times.

In the fully connected network, mobility-fw and MiXiM show peak throughput with around seven to nine senders. Mobility-fw, which has a simple SNIR threshold packet reception model, gives a higher packet delivery and lower latency than MiXiM and supports more senders. The reason for this is unclear, since all frames are received by all nodes with high signal strength, the effect of differences in interference and reception models should be small.

The opposite is true in the hidden terminal case, where MiXiM's more sensitive radio model presumably allows senders to detect competing transmissions more accurately. For both MiXiM and Castalia, their probabilistic reception models also seem to lead to more variation between simulation runs.

Unsurprisingly, the beacon-enabled PAN's modeled by inetmanet and Castalia support significantly more traffic than either of the unslotted CSMA networks, albeit with high latency (unreasonably so in the case of inetmanet). In the hidden terminal case, the Castalia PAN continues to have the expected high throughput, but inetmanet PAN performs worse than the CSMA networks. This behavior almost certainly reflects some significant issue in the inetmanet implementation. (Inetmanet's IEEE 802.15.4 simulation is documented as preliminary work.)

Another interesting feature is the throughput-PRR anomaly in the hidden terminal case: When three senders are distributed on opposite sides of the star, they cannot detect



Fully connected (radius = 0.2 × nominal range).

Hidden terminals (radius = 0.9 × nominal range).

Figure 5: Sync-star: Star topology with synchronized senders (IEEE 802.15.4). Path loss $\alpha = 2.5$; shadowing μ=0, σ=4; txPower = -10dBm. The error bars give the 95% CI over 10 simulation runs.

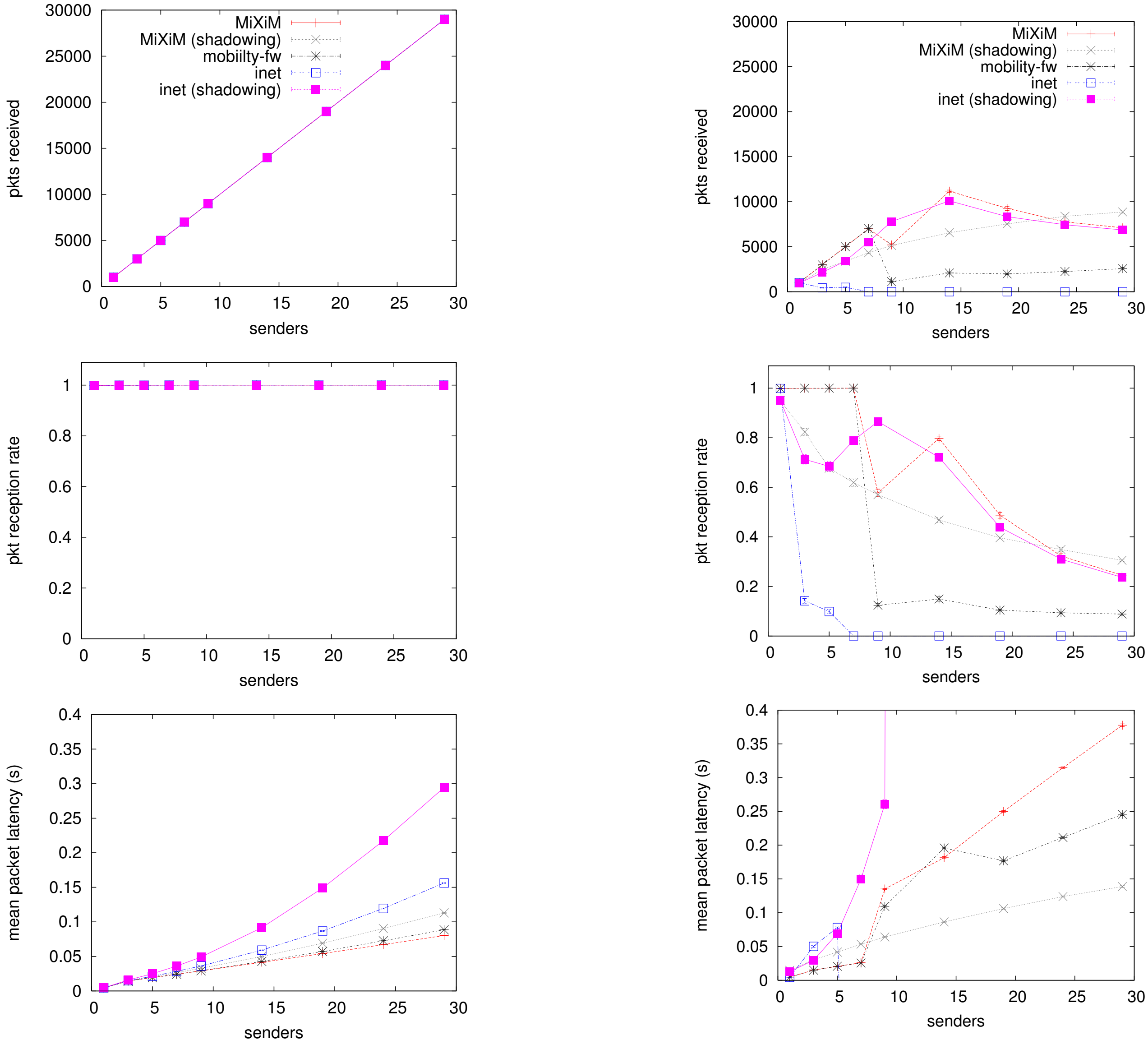
each other's transmissions, so they consistently, incorrectly assess the channel as clear, resulting in collisions at each transmission attempt. As the number of nodes grows, node become closer to their left- and right-hand neighbors and are again able to defer to some contending transmissions (Fig. 4). The effect is seen in both mobility-fw and inetmanet, though more strongly in the former, where it persists even with five senders. Even Castalia shows a small effect; there is considerably more variability in the PRR and latency for small numbers of senders. By contrast, MiXiM's sensitive radio model seems to allow it detect to competing senders, even in sparse networks.

IEEE 802.11 The IEEE 802.11 experiment is shown in Figure 6. Each sender generates 100 packets with 1024-byte payload, transmitting one packet every 10s.

For the fully connected case, all the simulators agree on nearly 100% PRR in all cases, though both inet variants show more rapidly increasing latencies than either MiXiM or mobility-fw. For both MiXiM and inet, the variant that includes shadowing shows larger latencies than its simple-path loss counterpart.

The hidden terminal case is more interesting. Mobility-fw shows threshold behavior, with nearly perfect PRR up to seven senders, and about 10% PRR thereafter. Inet with simple path loss has significant loss with three senders and fails to deliver any packets with more than five senders, despite having a similar frame reception model. This result almost certainly signifies some issue in inet, given its much more plausible behavior under log normal shadowing.

The other three simulators have more generally consis-



Fully connected (radius = 0.2 × nominal range).

Hidden terminals (radius = 0.9 × nominal range).

Figure 6: Sync-star: star topology with synchronized senders (IEEE 802.11b). Path loss $\alpha = 3.5$; shadowing μ=1, $\sigma = 10$; txPower = 50mW. The (very small) error bars give the 95% CI over 10 simulation runs.

tent behavior. MiXiM follows mobility-fw in having near 100% PRR up to seven senders, but its PRR then declines more gradually, though the anomaly at nine senders remains mysterious as it does not seem to be related to the topological anomaly described above. When log-normal shadowing is added, MiXiM's results are completely smooth. Inet with log-normal shadowing seems to show a trace of the anomaly seen in the IEEE 802.15.4 data. At higher loads, all three have similar throughput, but inet's latency becomes extremely high. This seems to be associated with a very high number of duplicate packets (i.e. lost ACKs) and is not seen in the MiXiM data.

6.2 WSN-like traffic

The second scenario, called "WSN-like", is a simple representation of single-hop sensor network, with randomly placed senders generating data and forwarding it to a sink. Compared to sync-star, this scenario introduces differences in received signal strength and more complex contention timing scenarios. Only results for IEEE 802.15.4 are reported, further results are available online.

A designated node (sink) is placed "near" the center of the rectangular field. Specifically, its x,y coordinates are uniformly randomly distributed within ±20% of the center. The other nodes are uniformly and randomly distributed over the field, which is dimensioned such that all nodes are within nominal transmit range of the sink. Consequently, the network is mostly connected: the field is a square whose edge is approximately 1.2 × nominal range. In these experiments, senders generate frames with exponentially dis-

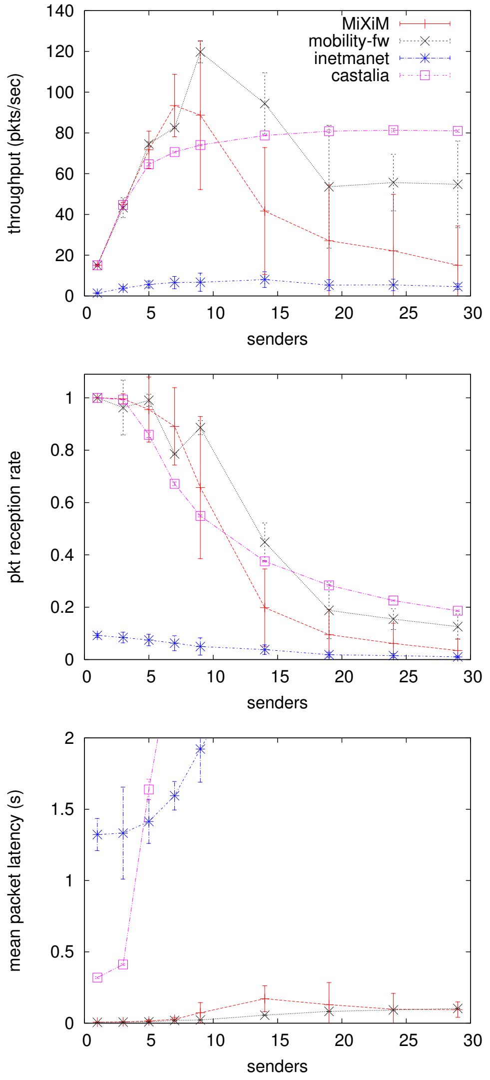


Figure 7: IEEE 802.15.4 WSN-like scenario: random topology, exponentially distributed uplink traffic. Path loss α=3.0; shadowing μ=0, σ=4; txPower = -5dBm. Error bars show 95% CI over 10 runs.

tributed arrivals at rate $\lambda = 15$ and transmit them to the sink. Each experiment is 300 s long and is repeated up to 10 times. The results are shown in Figure 7. The metrics are as in the sync-star experiment, though here latency most definitely reflects queuing as well as channel access delay.

MiXiM and mobility-fw both show a fairly plausible peak application throughout of about 80-95 kbps on the 250kbps channel. As in the fully connected sync-star case, mobility-fw has slightly higher throughput. MiXiM's higher variation between simulations may reflect increased sensitivity to topology: With mobility-fw's simple SNIR threshold radio model, nodes are likely to be either be hidden terminals, or not. MiXiM's radio model gives probabilistic losses in intermediate cases.

For the Castalia PAN network, the peak throughput is slightly lower than in the unslotted CSMA case, but the PAN scheduling allows the network to maintain that throughput at high load. This is very much the expected behavior.

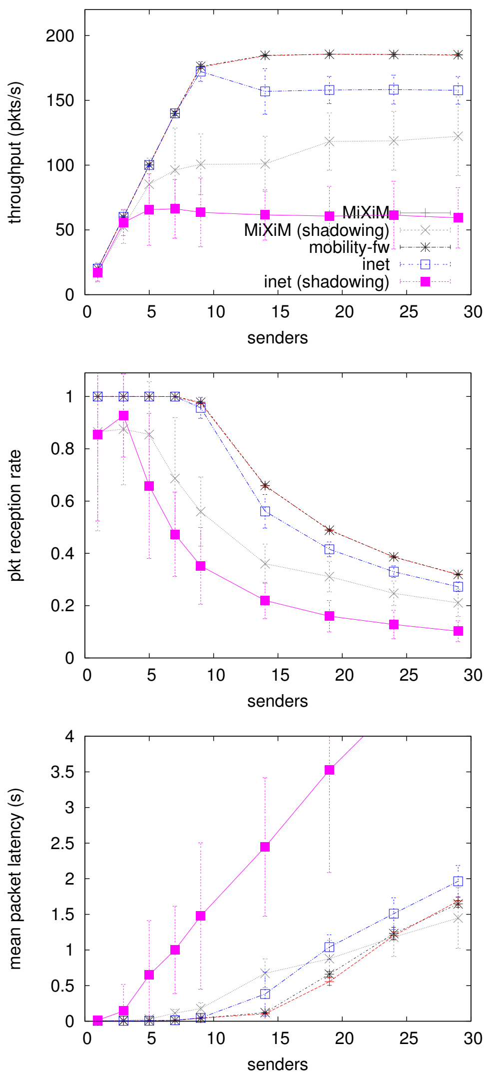


Figure 8: IEEE 802.11 AP-like scenario: random topology, exponentially distributed uplink and downlink traffic. Path loss $\alpha = 3.5$; shadowing μ=0, $\sigma = 10$; txPower = 50mW. Error bars show 95% CI over 10 runs.

The outlier is inetmanet, which shows poor performance in all cases, contrasting with its plausible (though very high latency) throughput in the sync-star scenario. Again, this model is documented as preliminary and the result clearly reflects this (some simulations did not run to completion).

6.3 AP-like traffic

The third scenario, called "AP-like", is similar to the WSN-like scenario. The topology is defined the same way, but with both uplink and downlink traffic to the designated node (AP). This traffic is roughly similar to an IEEE 802.11 AP although it does not reflect the flow control or timing associated with TCP or multimedia traffic. IEEE 802.11 results are presented, further results are available online.

Each sender generates 1024-byte packets, with exponentially distributed arrivals with rate $\lambda = 10$ and transmits them to the AP. The AP also generates traffic, with intensity $\lambda_{AP} = \lambda \times numSenders$, and the destination of each frame being randomly chosen from among the senders. Each experiment is 300 s long and is repeated up to 10 times. For this scenario, IEEE 802.11 RTS/CTS is used, otherwise parameters are as in the sync-star experiment.

The results are shown in Figure 8. Three simulators show similar throughput and packet reception rate. The AP-like scenario is a mostly connected network, so it is not surprising that it largely reflects the good agreement between simulators seen in the sync-star case. The network capacity is a quite plausible 1.4Mbps for the mobility-fw and MiXiM, inet is slightly lower. The main differences between simulators are small variations in latency, which may reflect differences in queuing as much as channel access.

The MiXiM and inet models that include log-normal shadowing both show much lower throughput than their simple path-loss counterparts. This result is especially confusing given that both of these simulators showed relatively higher throughput at higher loads in the hidden terminal sync-star scenario. (It is also noted that some MiXiM simulations with log-normal shadowing did not run to completion.)

7. CONCLUSION

The research community envisions increasingly complex applications in mobile social networks and wireless sensor networks. There is therefore a growing need for trustworthy simulation components for core protocols elements such as IEEE 802.11 and IEEE 802.15.4.

This work contributes to that goal by introducing a scenario framework for comparing results obtained in different simulation environments. Because it is not possible to create equivalent parameterizations of simulators with very different propagation and frame reception models, the framework defines nominal transmit range as an (imperfect) metric that encapsulates channel parameters and frame reception model.

Normalized test and evaluation scenarios are then defined in terms of nominal transmit range. Experimental results for five different OMNeT++-based simulations of IEEE 802.11 and IEEE 802.15.4 are presented. The differences that they uncovered demonstrate their effectiveness in comparing simulation results and highlighting behavior worth further investigation, even when the cause of the anomaly remained unclear. This approach allows us to leverage the diversity of models and implementations to identify problems, while still respecting valid differences between simulators.

Clearly, a result showing good agreement among simulators (or even the identification of a single outlier) would have been more reassuring. However, a long-term goal is that the framework described here (which is not specific to OMNeT++) will lead to the adoption of common test and evaluation scenarios and increase the quality of and confidence in simulation results. Most of the code and further output are available at `http://www.sics.se/nets`.

8. REFERENCES

[1] L. Bergamini, C. Crociani, A. Vitaletti, and M. Nati. Validation of WSN simulators through a comparison with a real testbed. In *Proc. of the 7th ACM Workshop on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks*, PE-WASUN, 2010.

[2] A. Boulis. Castalia: Revealing pitfalls in designing distributed algorithms in WSN. In *Proc. of the 5th Int'l Conf. on Embedded Networked Sensor Systems*, SenSys, 2007.

[3] M. Bredel and M. Bergner. On the accuracy of IEEE 802.11g wireless LAN simulations using OMNeT++. In *Proc. of the 2nd Int'l Conf. on Simulation Tools and Techniques*, Simutools, 2009.

[4] D. Cavin, Y. Sasson, and A. Schiper. On the accuracy of MANET simulators. In *Proc. of the 2nd ACM Int'l Workshop on Principles of Mobile Computing*, POMC, 2002.

[5] U. M. Colesanti, C. Crociani, and A. Vitaletti. On the accuracy of OMNeT++ in the wireless sensor networks domain: simulation vs. testbed. In *Proc. of the 4th ACM Workshop on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks*, PE-WASUN, 2007.

[6] P. Di, Y. Houri, K. Kutzner, and T. Fuhrmann. Towards comparable network simulations. Interner Bericht 2008-9, Dept. of Computer Science, Universität Karlsruhe (TH), Aug 2008.

[7] W. Drytkiewicz, S. Sroka, V. Handziski, A. Köpke, and H. Karl. A mobility framework for OMNeT++. TU-Berlin Tech. Report, 2003.

[8] L. M. Feeney. Poster abstract: Two simple tools for testing wireless communication modules in OMNeT++. In *Proc. of the 4th Int'l Workshop on OMNeT++*, 2011.

[9] G. P. Halkes and K. G. Langendoen. Experimental evaluation of simulation abstractions for wireless sensor network mac protocols. *EURASIP J. Wirel. Commun. Netw.*, 2010:24:1–24:2, 2010.

[10] S. Ivanov, A. Herms, and G. Lukas. Experimental validation of the ns-2 wireless model using simulation, emulation, and real network. In *Proc. of the 4th Workshop on Mobile Ad-Hoc Networks*, WMAN, 2007.

[11] A. Köpke, M. Swigulski, K. Wessel, D. Willkomm, P. K. Haneveld, T. Parker, O. Visser, H. S. Lichte, and S. Valentin. Simulating wireless and mobile networks in OMNeT++: The MiXiM vision. In *Proc. of the 1st Int'l Workshop on OMNeT++*, 2008.

[12] J. Lessmann, P. Janacik, L. Lachev, and D. Orfanus. Comparative study of wireless network simulators. In *Proc. of the 7th Int'l Conf. on Networking*, ICN, 2008.

[13] The OMNeT++ homepage. `http://www.omnetpp.org`.

[14] K. Tan, D. Wu, A. Chan, and P. Mohapatra. Comparing simulation tools and experimental testbeds for wireless mesh networks. *Pervasive and Mobile Computing*, 7(4):434–448, 2011.

[15] A. Varga. OMNeT++. In K. Wehrle, M. Günes, and J. Gross, editors, *Modeling and Tools for Network Simulation*. Springer-Verlag, 2010.

[16] T. Voigt, J. Eriksson, F. Österlind, R. Sauter, N. Aschenbruck, P. J. Marrón, V. Reynolds, L. Shu, O. Visser, A. Koubaa, and A. Köpke. Towards comparable simulations of cooperating objects and wireless sensor networks. In *Proc. of the 4th Int'l Conf. on Performance Evaluation Methodologies and Tools*, VALUETOOLS, 2009.

[17] K. Wessel, M. Swigulski, A. Köpke, and D. Willkomm. Mixim – The physical layer: An architecture overview. In *Proc. of the 2nd Int'l Workshop on OMNeT++*, 2009.