Friginal López, J.; Andrés Martínez, DD.; Ruiz García, JC.; Martínez Raga, M. (2014). A survey of evaluation platforms for ad hoc routing protocols: a resilience perspective. Computer Networks. 75(A):395-413. doi:10.1016/j.comnet.2014.09.010.

A Survey of Evaluation Platforms for Ad Hoc Routing Protocols: a Resilience Perspective

Jesús Friginal[a], David de Andrés[b], Juan-Carlos Ruiz[b], Miquel Martínez[b]

jesus.friginal@laas.fr, {ddandres, jcruizg, mimarra2}@disca.upv.es

[a]*LAAS-CNRS, 7 Avenue du Colonel Roche. 31077 Toulouse Cedex, France*
[b]*STF-ITACA Universitat Politècnica de València, Campus de Vera s/n, 46022, Spain*

Abstract

bla bla bla

Keywords: Survey, resilience evaluation, fault and attack injection, ad hoc networks, experimental testbeds

---

1. Introduction

Despite being originally developed for military purposes, the unique properties of ad hoc networks make them very suitable for civilian and commercial purposes. These networks will open up new avenues across the full breadth of future information technologies. A diverse set of applications exploiting the ad hoc network communication paradigm in a number of different situations, are today a reality. Wireless Sensor Networks (WSN), which are typically used for monitoring and observing physical phenomena [1]. These networks have also been used for deploying communication and data networks in emergence situations, like in the hurricane Katrina [2] crisis, in 2006. Wireless Mesh Networks (WMN) are another interesting example of ad hoc network. They are capable of providing affordable Internet access to little or isolated populations far away from big city centres [3]. In the future, it is expected that many other application domains will benefit from ad hoc networks, e.g., Vehicular Ad hoc NETworks (VANET) [4] to enhance passengers comfort and safety, or aeronautical ad hoc networks [5] to increase the data rate of in-flight broadband Internet access.

Routing protocols are the backbone of ad hoc networks. Routing protocols create complete routes between every pair of nodes from the topology information they are able to perceive. Reactive ones, establish routes under demand (such as LAR1 [6], AODV [7], DSR [8], TORA [9] and SrcRR [10]), whereas proactive protocols periodically exchange routing information (such as WRP [11], OLSR [12], DSDV [13], Babel [14] and B.A.T.M.A.N [15]).

Regardless their type, there are different factors that limit the *capability of routing protocols to provide a persistent routing in spite of threats* (also known as resilience [16]). On one hand, the evolutions of the interacting nodes cannot all be anticipated and controlled a priori given the dynamic conditions of the environment, e.g. mobility, energy extenuation or interferences in the wireless communication channel may favour the occurrence of accidental faults [17]. On the other, the implicit trustworthiness relationship established between neighbour nodes may be exploited by malicious faults (also referred to as attacks). The occurrence of these accidental faults and attacks in the system may dramatically affect its expected behaviour. Such events may partition the network or induce a certain traffic overhead, thus causing retransmission, inefficient routing and impacting the quality of service provided to the upper system layers. So, designing ad hoc routing protocols with resilience in mind is essential for their successful exploitation [18]. In order to mitigate the effect of potential accidental faults and attacks on default versions of routing protocols, some authors have proposed se-

cure and fault-tolerant alternatives. Some examples are SEAD [19], SOLSR [20], SAODV [21] or CONFIDENT [22].

Traditionally, simulation has been the platform of choice to evaluate these secure and fault-tolerant protocols. However, there is growing awareness of the fact that current simulators make several simplifying assumptions to model many essential characteristics of real systems, thus limiting the credibility of their results [23]. The gap between simulated and experimental results may lead to differences between the behaviour of the simulated network and that of the real one. This gap becomes extremely important when addressing real systems that may be supported by these protocols, especially when human lives are at risk, large economic losses may derive from their malfunctioning or even when the reputation of service providers may be dramatically affected in non-critical domains. So, it is of prime importance to validate theoretical design and analysis of routing protocols and algorithms with sound experiments that are representative of their final deployment. Thus, the experimental resilience evaluation of ad hoc routing protocols plays an essential role to determine the confident use of ad hoc routing protocols.

Unfortunately, this evaluation typically involves the execution of real experiments that are inherently complex, normally time-consuming to set up and execute, and hard to repeat by other researchers. Basically, the difficulty to recreate and quantify the impact of the wide amount of faults and attacks threatening ad hoc routing protocols, is one of the reasons hindering their resilience evaluation.

Following previous axes, the goal of this article is to review, discuss, and classify the existing experimental platforms and testbeds for ad hoc networks to shed light on how these platforms address the challenges of experimental evaluation from a resilience perspective. In addition, we discuss their limitations and highlight future research issues.

To address this goal, the rest of the publication is structured as follows. Section 2 presents related work. Section 3 presents current platforms for the evaluation of such protocols. Section 4 studies the the measures typically used in their evaluation and the challenges they present. Section 5 analyses the experimental characteristics of current widely-used evaluation platforms in ad hoc networks. Section 6 identifies the lacks of such proposals when addressing the interpretation of obtained measures. Finally Section 7 presents the conclusions of the survey stating how the characteristics of resilience evaluation may contribute to enhance the quality features of the evaluation.

2. Related work

In the last years, some works have deeply analysed the evaluation techniques used in the domain of ad hoc networks. In [24], the authors classify the formal and non-formal techniques to evaluate ad hoc networks. This classification is used to discuss about common elements of existing protocol validation approaches. The goal is to show how to take advantage of such similarities, thus optimising the evaluation time. In [25], the authors present an analysis of the platforms to perform real-world experimentation. In particular, they report the technology used for the implementations as well as on key findings from real experiments. Similarly, in [26], the authors present an extensive survey covering real world and emulation testbeds in the domain of ad hoc networks. However, in these papers there is a lack of taxonomies to properly characterise evaluation platforms. The work presented in [27] proposes an interesting taxonomy to classify testbeds according to their heterogeneity, concurrency, scale, mobility, repeatability, user involvement and federation. However, it is finally used to characterise only five platforms. In general, all these works pose the need of more experimental platforms to allow for protocol validation and proof-of-concept implementations. However, they mainly focus on the performance perspective of the problem whereas the key resilience aspects of these platforms (posed in the introduction) are not analysed.

Alternatively, there is a wide variety of surveys that focus on secure and fault-tolerant routing protocols. For instance, [28] compares different security routing techniques to face threats based on various criteria. The work done in [29] and [30] performs a similar analysis but addressing the particular domain of WSNs. Additionally, in [31] and [32], the authors respectively analyse the concept of trust and security in state-of-the-art secure routing mechanisms for MANETs. However, these works do not explore the platforms considered to evaluate these secure and fault tolerant routing protocols, beyond the classic paradigm used: simulation, emulation or real-world experimentation. This is mainly due to the fact that the faults and attacks considered to exercise these routing protocols are typically implemented ad hoc for the purpose of each study. This choice is justified given the lack of open (and thus reusable) resilience evaluation platforms that enable the implementation of very particular threats. Without the aim of being exhaustive, Table 1 lists some of the wide amount of accidental faults and attacks used in the evaluation of secure and fault-tolerant routing protocols. Some of them are accidental while others are malicious. In any case, most of them have a transient nature given the dynamic features of ad hoc network deployments. Some of them, like variable signal interferences, or fading, are inherited from the traditional un-

steadiness and openness of the wireless communication medium while others, such as power consumption or accidental and malicious topological changes, are acquired given the unique properties of ad hoc networks. An in-depth analysis of the devices used in ad hoc networks shows that hardware gets more sensitive to manufacturing faults as the scale of components decreases. Likewise, increasing the level of sophistication of embedded software like operating systems or middleware leads to higher rates of design, programming, and configuration faults [17, 18].

Table 1: Accidental faults and attack used during the evaluation of secure routing protocols.

Network characteristic	**Main threats identified**
Resources limitations	Physical damage [33], peak in service demand [34], flooding attack [35], neighbours saturation [36], battery extenuation [37]
Wireless communication medium	Signal attenuation [38], multifading [39], ambient noise [40], jamming attack [41], exposed node [42], traffic analysis [43], cryptanalysis [44]
Mobility of nodes	Wrong nodes distribution [45], wrong routing protocol configuration [46], sequence number replay [35], replay attack, sybil attack [18], tampering attack [47], Doppler shift [4]
Absence of infrastructure	Sink hole, black hole [48], selective forwarding attack [49], jellyfish attack [50]

In conclusion, we can point out a lack of surveys that analyse how resilience is addressed by current evaluation platforms for ad hoc routing protocols. There exist many and varied challenges in the deployment of ad hoc routing protocols, but the need for frameworks to evaluate and justify their resilience is, without doubt, one of the most important. It is worth pointing out the difficulty to recreate and quantify the impact of the wide amount of accidental faults and attacks threatening ad hoc networks. However, their use in evaluation platforms is essential to determine whether the risks to which ad hoc networks are exposed are acceptable or not.

Projects such as DBench [51], introduce the principles of comparative evaluation of systems, basically identifying three main stages: (i) the experiments configuration, where the system measures are specified; (ii) the experiments execution, which presents the experimental properties that evaluation platforms should satisfy (including the capacity to inject faults and attacks in the system); and (iii) the analysis of results, that introduces the keys of measurements processing.

In our prior work [52], we briefly identified the three dimensions introduced

in the DBench project in the the domain of ad hoc networks. In the present paper we ambition at studying how current evaluation approaches for ad hoc routing protocols address these three basic aspects. In consequence, next Section will introduce the most well-known evaluation platforms for ad hoc routing protocols in the last decade, which will be the target of our survey.

3. Evaluation platforms and routing protocols

This section briefly introduces most of current evaluation platforms in the domain of ad hoc networks. Let us present them according to the three major strategies to address the evaluation of ad hoc routing protocols: use of models, prototypes or emulation.

3.1. Evaluation platforms

Model-based approaches use formal techniques like classic process algebras [53], timed automata [54], model checking [33] or the use of Stochastic Activity Networks (SAN), an extension of Stochastic Petri Nets [55] to recreate real-world characteristics. These approaches are finally animated through discrete-event simulation platforms [56]. Simulation platforms are in general a good option to check design proposals and discard them before they become too costly to modify in next stages of their lifecycle. Most well-known approaches today are Network Simulator (NS) 2 [57] and 3 [58], Opnet [59] and Glomosim [60].

Conversely to simulation, real-world prototyping, that refers to the execution of experiments in real scenarios, provides most representative results to evaluate ad hoc networks. Real-world prototyping is a good option to evaluate mature developments in their expected environment. In the bibliography it is possible to find different open-source prototype-based approaches enabling the use of real devices and applications, such as Roofnet [61], Floornet [62], the Ad Hoc Protocol Evaluation Testbed (APE) [63], the Reconfigurable Mobile Multi-hop Wireless Network Testbed (MINT) [64], Indriya [65], or the Testbed for Wireless Indoor Experiments (TWIST) [66].

Emulation is a hybrid solution halfway simulation and prototype-based evaluation platforms. Emulation is the most recommendable option to quickly evaluate developments (being mature or not) in different scenarios. It considers typical elements from real deployments like the use of real devices and applications, and simulates others that are difficult to recreate and repeat, like mobility. Some of the most representative platforms are the Open-Access Research Testbed for Next-Generation Wireless Networks (ORBIT) [67], the Carnegie Mellon University

Wireless Emulator (CMUWE) [68], Castadiva [69], Mobiemu [70], Emulab [71] and the Resilience Evaluation Framework for Ad Hoc Networks (REFRAHN) [72].

3.2. Evaluation of routing protocols

On one hand, some of these evaluation platforms, basically those based on simulation, are built in such a way that any routing protocol model they may evaluate must be compatible with the platform. This involves that routing protocol models must be implemented in the language used by the platform, which typically differs from one to another. The difficulty to find routing protocol models guaranteeing an acceptable quality for a given simulator is a limitation that leads simulators developers to include a minimum set of implementations in their platform by default. Table 2 present default routing protocols implemented for each simulator.

Table 2: Protocol-dependent evaluation platforms.

Evaluation platform	**Routing protocols considered by default**
Opnet Opnet [59]	AODV, DSR, TORA
NS2 [57]	AODV, DSR, OLSR, DSDV
NS3 [58]	AODV, OLSR, DSDV
Glomosim [60]	AODV, DSR, DSDV LAR1, WRP

On the other hand, prototype- and emulation-based platforms are normally routing-protocol-independent. This characteristic eases the evaluation of a given routing protocol implementation in different platforms. Consequently, conversely to model-based platforms, they do not require to include any set of routing protocols by default for users. Table 3 provides a list of well-known protocols, as well as the works where they have been evaluated by targeted platforms.

4. Experiments configuration

According to [51], the selection of measures is one of the key aspects of the experiment configuration. Regarding their typology, experimental measures can be classified according to their capacity to provide a confident insight of the performance, resources consumption and resilience of the systems under evaluation.

Measures can be generic, if applicable to any ad hoc network or routing protocol (such as throughput); or concrete, if they just make sense in a limited subset

Table 3: Protocol-independent evaluation platforms.

Evaluation platform	**Routing protocols evaluated**
CMUWE	DSR [73]
ORBIT	AODV [74], OLSR [74]
Castadiva	AODV [75], OLSR [75]
Emulab	DSMR [76], DSRP [76], OLSR [77], BMX6 [77]
Mobiemu	AODV [78, 79], OLSR [78]
REFRAHN	AODV [72], OLSR [72], SOLSR [72], B.A.T.M.A.N [72], Babel [72]
NetEye	TMA [80]
TutorNet	CTP [81], MultihopLQI [81]
Senslab	RPL [82]
Wisebed	Digimesh [83]
Roofnet	SrcRR [84]
Floornet	OLSR [62]
APE	AODV [63], OLSR [63]
MINT	AODV [64]
Indriya	ORW [85], CTP [85], TMA [80]
TWIST	ORW [85], CTP [85]
Citysense	OLSR [86]

(such as the Expected Transmission Count, also referred to as ETX [87], a measure that can be only considered in routing protocols whose routes are computed using the notion of link quality). Despite this last type enables to acquire a more detailed observation of the system, its particular nature makes their consideration inadequate for the comparison of heterogeneous evaluation platforms that pursue different purposes. Thus, to ease such comparison, this paper focuses on generic measures.

4.1. Performance measures

Performance measures are in general widely-used in the domain of ad hoc networks since they easily represent the behaviour of their functional aspects. In particular, considered evaluation platforms typically rely on quality-of-service measures to characterise the behaviour of the system such as *Throughput*, *Packet delivery ratio*, *Packet loss*, *Delay*, *Routing overhead* and *Link flapping*.

- Probably, *Throughput* is one of the most considered measures in the bibliography of ad hoc networks. It estimates the average amount of information correctly delivered (typically in Mbits and Kbits) per time unit (normally a second). Throughput, also referred to goodput when just computing applicative (non-routing) traffic, is usually computed between the source and

destination nodes of a communication route, or as a general measure to estimate the overall behaviour of a network. This measure is interpreted the higher the better. Experimental platforms such as Opnet [59], NS2/3 [57, 58], Glomosim [60], CMUWE [68], ORBIT [67], Castadiva [69], Emulab [71], MobiEmu [70], Roofnet [61], Floornet [62], MINT [64], Indriya [85], TWIST [85] and Citysense [86] employ it.

- The *Packet delivery ratio* (PDR) is similar to throughput but considering the percentage of packets correctly delivered with respect to the total sent. Evaluation platforms such as Opnet, NS2/3, Glomosim, CMUWE, REFRAHN [72], Senslab [88], Wisebed [89], APE [63], Indriya, NetEye and TutorNet provide this measure.

- *Packet loss* is the percentage of packets not delivered from the total sent. As one can deduce, this measure complements the delivery ratio. From a performance viewpoint, this measure is the lower the better. Opnet, NS2/3, Glomosim, ORBIT, REFRAHN, Wisebed, Indriya and TWIST are the evaluation platforms providing packet loss by default.

- *Delay* is the average time required by a packet to get from source to destination. Obviously, the longer it is, the worse for the network behaviour. Together with *throughput*, *delay* is one of the most used measure in the domain of ad hoc networks. Given its popularity, is it widely considered by Opnet, NS2/3, Glomosim, CMUWE, ORBIT, Castadiva, Emulab, MobiEmu, REFRAHN, Wisebed, Roofnet, MINT, Indriya and TWIST.

4.2. Resources consumption measures

In the domain of ad hoc networks, the consideration of resources consumption measures typically concerns *routing overhead* and *energy consumption*.

- *Routing overhead* represents the rate of packets exchanged between the nodes to keep the routing protocol operative. Since a high value may overload the wireless communication channel, thus reducing its capacity to transport applicative packets, best routing protocols are typically those providing lower levels of routing overhead. Opnet, NS2/3, Glomosim, Senslab and Wisebed are examples of evaluation platforms providing this measure.

- *Energy consumption* is a measure that computes the energetic waste of routing protocols. It is specially useful in environments where the battery is a

limited resource, although it can also be considered as a measure of added-value in green computing to make aware users about a responsible consumption. This measure is provided by evaluation platforms such as Opnet, NS2/3, Glomosim, Senslab, Wisebed and REFRAHN.

4.3. Resilience measures

Research in the field of resilience evaluation [16], highlights the importance of characterising the ability of ad hoc networks to detect, diagnose, repair, and restore the normal behaviour of the system in presence of faults and changes. Thus, in case of threats, the routing protocol with the lowest and more accurate detection and diagnosis time could be able to react faster and, consequently, limit the effect of threats in the system. Likewise, low repair and restoration times are interesting to reduce the system downtime. Some of the measures that characterise the resilience of ad hoc networks are *Link flapping*, *Number of eliminated links*, *Link availability*, *Link integrity*, *Threat exposure* and *Perturbation effectiveness*. Considering them may complement traditional evaluation, for example, by assisting evaluators to select the more robust ad hoc routing protocol for a given system.

- The *Link flapping* is a measure to determine the number of times a communication route changes the links. Although the dynamic alternation of links can be understood as a mechanism to react against changes, a high link flapping can be also counter-productive due to its impact on the convergence time of links. So, the higher it is, the worse for the network behaviour. APE, Senslab, Indriya and TutorNet are the only approach providing it by default.

- The *Connectivity* is computed as the ratio between broken and established links along the experimentation time. Despite it could be also considered a measure of performance, its usefulness to estimate the volatility of neighbours with respect to a node led us to finally classify it as a resilience measure. Approaches such as APE and Roofnet consider this measure.

- The *Route availability* is a measure that determines the percentage of time a communication route was ready to be used with respect to the total time. The higher the better for the network. REFRAHN is the only evaluation platform that provides this measure by default.

- The *Packet integrity* computes the average percentage of packets received at destination nodes whose content was not illegitimately altered by malicious

nodes with respect to the total time of experimentation. The higher this value, the better. REFRAHN is the only evaluation platform that provides this measure by default.

- The *Threat exposure* determines the average percentage of time the communication route was exposed to any perturbation with respect to the total time of experimentation. The lower the better. REFRAHN is the only evaluation platform that provides this measure by default.

- The *Perturbation effectiveness* represents the average percentage of the threat exposure time when the perturbation succeeded on impacting the network behaviour. The lower the better from a resilience viewpoint. REFRAHN is the only evaluation platform that provides this measure by default.

Table 4: Summary of measures considered by current evaluation platforms.

Measure		*Opnet* [59]	*NS2* [57, 58]	*Glomosim* [90]	*CMUWE* [68]	*ORBIT* [67]	*Castadiva* [69]	*Emulab* [71]	*MobiEmu* [70]	*REFRAHN* [72]	*NetEye* [91]	*TutorNet* [92]	*Senslab* [88]	*Wiselab* [89]	*Roofnet* [61]	*Floormet* [62]	*APE* [63]	*MINT* [64]	*Indriya* [65]	*TWIST* [66]	*Citysense* [66]
Performance	Throughput	✓	✓	✓	✓	✓	✓	✓	✓						✓	✓		✓	✓	✓	✓
	Packet delivery ratio	✓	✓	✓	✓					✓	✓	✓	✓	✓			✓		✓		
	Packet loss	✓	✓	✓		✓				✓				✓					✓	✓	
	Delay	✓	✓	✓	✓	✓	✓	✓	✓	✓				✓	✓			✓	✓	✓	
Resources consumption	Routing overhead	✓	✓	✓									✓	✓							
	Energy consumption	✓	✓	✓						✓			✓	✓							
Resilience	Link flapping											✓	✓		✓		✓		✓		
	Connectivity																✓				
	Route availability									✓											
	Route integrity									✓											
	Threat exposure									✓											
	Pertur. effectiveness									✓											

4.4. Measures analysis

Table 4 summarises the measures previously listed in this Section. Results show how the behaviour of ad hoc routing protocols is massively characterised through performance measures reporting the throughput, delay, routing overhead or the packet delivery ratio exhibited by the network. Indeed, almost 90% of the evaluation platforms considered in this study provide throughput and delay as a reference measure, which is an example of how to date, the evaluation of ad hoc routing protocols has been generally limited to functional aspects.

Additionally, it is worth noting that non-functional aspects surprisingly remains in the background. With respect to resources consumption, few measures are normally considered. In particular, only those evaluation platforms that are able to recreate networks with resources limitations, such as wireless sensor ones, consider them. From a resilience viewpoint, it is striking the limited presence of measures to quantify other interesting aspects related to the ability of the system to coexist and tolerate the presence of threats in the system. Although there are studies that evaluate the impact of threats in the system through the degradation of mainstream performance measures (process typically referred as performability evaluation), it is also necessary the definition and use of specific resilience measures to understand and explain the behaviour of routing protocols. This lack is one of Achilles' heels in the evaluation of ad hoc networks. Concretely, REFRAHN seems being the approach providing the widest variety of measures, offering 66% of the total of measures provided in general by all the evaluation platforms. We claim that more platforms are necessary to cover the gap between the ad hoc networks we are able to design and implement, and the confident service they are able to deliver.

5. Experimental procedure

Once measures studied, it is necessary to determine the experimental procedure to obtain them. Although many authors in the domain of ad hoc networks present measures that appear generally correct and sensible, sometimes the approach followed to obtain them varies from a paper to another [93]. This fact is due to the lack of widespread rules and practices to conduct their assessment following reference or standardised criteria, which hinders the comparison of measures. Consequently, while the focus is set of the results they provide, the quality of the evaluation platforms is seldom discussed. Most surveys in the field of evaluation platforms of ad hoc routing protocols limit their contribution to classify existing proposals according to their typology (i.e., if they rely on simulation, real-world experimentation or emulation). But beyond this fact, the implications of such design decisions are not rigorously analysed or quantified.

5.1. Desirable properties of resilience evaluation platforms for ad hoc routing protocols

The present section defends the interest of identifying and quantifying a set of basic properties as a criterion to assess the resilience of ad hoc routing protocols. To cope with this goal, this paper proposes to consider the properties used to

characterise dependability benchmarks [94]. Dependability benchmarks represent an agreement between the industry and/or the user community, gathering aspects such as the way and conditions under which measurements are obtained. This fact necessarily involves the properties they must satisfy are well-defined and widely-accepted. Such properties gather operational aspects (such as controllability and observability, which are essential attributes of control theory [95]), metrological aspects (such as intrusiveness, repeatability and accuracy, typically applied to measurement systems [96], and other maintainability aspects (such scalability, portability, configurability, traditionally used in performance benchmarks like TPC [97]). Furthermore, as a novelty, we introduce the notion of injectability to determine the capability of platforms to introduce changes and faults in the system under evaluation.

Hence, these properties, as explained in Table 5), can be useful to characterise the level of adequacy of current evaluation platforms from a more objective viewpoint. In concordance, Table 7 qualitatively characterises previous properties according to three intuitive levels: high, medium and low.

Once such three-level criteria established, characterising current evaluation platforms results an easier task. Next subsection studies the evaluation platforms introduced in this survey according to such a classification.

5.2. Critical characterisation of evaluation platforms

Given the diversity and heterogeneity of considered evaluation platforms, it is very difficult to establish clear criteria to compare existing tools, which hinders the selection of the appropriate approach. Hereafter, evaluation platforms are classified according to the levels of satisfaction of the properties under study.

5.2.1. Model-based approaches

More than 80% of works about evaluation in the domain of ad hoc networks is based on simulation [23]. This is because experiments are highly controllable, repeatable, scalable and, since no real device is considered, they present a low level of intrusiveness. Indeed, simulation platforms such as NS2 [57], NS3 [58], Opnet [59] or Glomosim [60] can scale the network up to thousands of nodes considering well-known mobility patters such as Random Way Point (RWP) or Manhattan. However, while the open-source nature of platforms such as NS2 and NS3 or Glomosim is highly configurable, other proprietary solutions such as Opnet lack of support to integrate additional configuration options (such as a wider variety routing protocols by default). With regard to their portability, all these

Table 5: Desired properties in an evaluation platform for ad hoc routing protocols.

Property	**Description**
Controllability	Ability of the evaluation platform to move the internal state of a system from any initial state to any other final state in a finite time interval [98]. The higher the better.
Observability	Capability of the evaluation platform to infer the behaviour of the system under evaluation through measurements [98]. The higher the better.
Intrusiveness	Degree of perturbation (either spatial or temporal) introduced in the evaluation platform by monitoring probes, workload/faultload generators and control components [96]. The lower the better.
Repeatability	Property of the evaluation platform to provide statistically equivalent measures through the same experimental procedure in the same location and operating conditions [96]. The higher the better.
Accuracy	Degree of closeness of measurements of a quantity to that quantity's actual (true) value [96]. The higher the better.
Scalability	Property related to the easiness to increment (or decrement) the dimensions of the experiment or the size of the evaluation platform (e.g., number of nodes, data flows, etc) [99]. The higher the better.
Portability	Capability to deploy and use the same evaluation platform (including both hardware and software components) to evaluate different systems [99]. The higher the better.
Configurability	Capability of the evaluation platform to configure different types of experiments through a wide variety of parameters and options which do not restrict its use (type of nodes, data flows, mobility patters, etc) [100]. The higher the better.
Computational complexity	Degree of difficulty of the evaluation platform to address or solve complex experimentation using its computational resources [99]. The lower the better.
Injectability	Capability of the evaluation platform to animate the behaviour of the system through a fault-load and a change-load. The higher the better.

simulators are typically available for a wide set of architectures and operating systems, which makes them highly portable.

Unfortunately, the execution of network models may require a heavy computation, which involves a high computational complexity. Furthermore, in any case, most of the results obtained from simulation are based on assumptions which typically simplify in excess the behaviour of the system and limits the accuracy of results, potentially leading researchers to biased or wrong conclusions. For example, simulations have been criticised for not using realistic mobility models [101] or because of assuming unrealistic wireless medium characteristics [102]. From a viewpoint of resilience, it is worth noting the low level of injectability presented in default implementations of current simulation platforms. However, this characteristic is shared with the rest of approaches, as we will see in next subsections. Probably, due to this reason, observability is typically bounded by the type and amount of measurements that system probes are able to collect, which in practice is limited to performance measures. To some extent, this point justifies the lack of

Table 6: Qualitative levels for the characterisation of evaluation platforms.

Property	High	Medium	Low
Controllability	Given all initial times and all initial states, the evaluation platform state always evolves towards the expected state in the proper time. The platform implements mechanisms to correct all the potential deviations in the control of experiments.	Given all initial times and all initial states, the evaluation platform state evolves towards a final state that, despite slightly deviated from the expected at some finite time, can be accepted as correct. Implemented mechanisms to correct potential deviations in the control of experiments may not be enough to exactly conduct the system towards the desired state.	Given all initial times and all initial states, the evaluation platform evolves towards a transient states which does not guarantee reaching a final state at some finite time. The platform does not implement mechanisms to correct some potential deviations in the control of experiments, or they are ineffective.
Observability	The evaluation platform is instrumented with probes in such a way it is easy for the user to take any type of measurements from any layer or component of the system under evaluation.	The probes deployed by the evaluation platform limit the type or amount of measurements the user may take from the system under evaluation.	The probes deployed by the evaluation platform limit the type and amount of measurements the user may take from the system under evaluation.
Intrusiveness	The measurements obtained from the platform are spatially and temporally influenced by internal components (e.g., the monitoring probes, workload/faultload generators and control components) or external interferences (e.g., the wireless channel).	The measurements obtained from the platform are either spatially or temporally influenced by internal components (e.g., the monitoring probes, workload/faultload generators and control components) or external interferences (e.g., the wireless channel).	The measurements obtained from the platform are not influenced by internal components (e.g., the monitoring probes, workload/faultload generators and control components) or external interferences (e.g., the wireless channel), or their effect is negligible.
Repeatability	The evaluation platform is able to repeat the same events in the same order, experiment after experiment, and the measurements observed from the system under evaluation are exactly the same.	The evaluation platform is able to repeat the same events in the same order, experiment after experiment, but the system under evaluation is subjected to sources of uncertainty that avoid the measurements observed are exactly the same.	Neither the evaluation platform is able to repeat the same events in the same order, experiment after experiment, nor the measurements observed from the system under evaluation are exactly the same, which hardens the comparability of results during their analysis.
Accuracy	The experiments deployed by the evaluation platform are representative of the behaviour of the system under evaluation in the real world. All the elements of the system under evaluation are real.	The evaluation platform makes some assumptions or simplifications in the experimentation. Some of the elements of the system under evaluation may not be real (e.g., the mobility of nodes or the type of nodes).	The evaluation platform implements a model of the system under evaluation. As any element of the system under evaluation is real, obtained measurements may radically vary from the expected depending on the goodness of the model.
Scalability	The evaluation platform enables the increment of the number of resources or assets involved in the experimentation (e.g., the number of nodes, the number of data flows, etc) very easily and it is completely affordable.	The evaluation platform enables the increment of the number of resources or assets involved in the experimentation (e.g., the number of nodes, the number of data flows, etc) with a moderate effort and/or a relatively affordable cost.	The evaluation platform does not enable the increment of the number of resources or assets involved in the experimentation (e.g., the number of nodes, the number of data flows, etc), or it is very costly in terms of time or money.
Portability	The evaluation platform is designed using a clear specification and/or methodology, and implemented using software and hardware COTS as long as possible, which eases that other evaluators may replicate the platform in a different location.	The evaluation platform is either designed using a clear specification and/or methodology, or it is implemented through custom software and hardware components, which hardens that other evaluators may replicate the platform in a different location.	The evaluation platform is neither designed using a clear specification and/or methodology, nor it is implemented using software and hardware COTS.
Configurability	The evaluation platform can be easily configured to enable the execution of different types of experiments through a wide variety of parameters and options which do not restrict its use (e.g., type of nodes, mobility patters, routing protocol, etc). Furthermore, it is open to easily admit the introduction of new types of parameters.	The evaluation platform can be configured to enable the execution of different types of experiments through a wide variety of parameters and options which do not restrict its use (e.g., type of nodes, mobility patters, routing protocol, etc), or it is open to admit the introduction of new types of parameters.	The evaluation platform cannot be configured to enable the execution of different types of experiments, and it cannot admit the introduction of new types of parameters.
Computational complexity	The execution duration of the experiments carried out in the evaluation platform strongly depends on the complexity of experiments, and the complexity of experiments may require a considerable processing power and memory to execute an experiment.	The execution duration of the experiments carried out in the evaluation platform strongly depends on the complexity of experiments, and the complexity of experiments may require a considerable processing power or memory to execute an experiment.	The execution duration of the experiments carried out by the evaluation platform is independent from the complexity of experiments. One second of experimentation is equivalent to one second of real time.
Injectability	The evaluation platform is able to introduce a wide variability of accidental and malicious faults in the system and dynamically change the evolution of nodes mobility.	The evaluation platform is only able to either introduce a limited amount of faults in the system or dynamically change the evolution of nodes mobility.	The evaluation platform does not support neither the introduction of any fault in the system nor the dynamic change of nodes mobility.

resilience measures previously observed in Section 4.

5.2.2. Prototype-based approaches

Generally, most of prototype-based approaches are custom-built for specific projects addressing performance aspects, and are consequently non-reusable for other experiments and purposes. Furthermore, as real devices are used, these platforms are more difficult to manage and control. However, their accuracy is high as far as the experiments executed are representative of the real world. This fact also reduces the system complexity because the system does not require the use of models. External (unpredictable) sources of signal interference may affect the intrusiveness and controllability of this kind of platforms. This means that effective

throughput is much more lower than the nominal, given the lack of control over the external interferences.

The scalability of these platforms varies from tens to hundreds of nodes. In the first case, we can consider evaluation platforms such as Roofnet [61], Floornet [62] or Citysens [86]. Two features make these testbeds particularly interesting: their realism and domain specificity provided by a permanent outdoor installation in an urban environment and the realization of the control and management based solely on wireless links. In the second case, indoor testbeds such as Indriya [65] and TWIST [66] are able to deploy up to 120 and 204 usb-based motes respectively. All of them are projects deployed for conducting experiments to understand the nature of large-scale wireless networks. So they are able to provide more repeatable experiments given their static nature. However, they lack of flexibility for the deployment of different topologies. Alternatively, some others like APE [63] provide experiments with mobility, but these experiments are difficult to configure and repeat because node movement must be reproduced manually or through self-moving devices (such as programmable robots, in the case of [64]). Furthermore, there exist some limitations regarding the type of mobility of nodes or their speed, that bound their configurability.

From an injectability viewpoint, most of these approaches present a low level. Exceptionally, TWIST and MINT present a medium level. TWIST enables the injection of accidental faults given its capability to support on-the-fly configuration changes. Thus, TWIST is able to emulate the death of sensor nodes due to energy depletion (fault injection) or the addition of new nodes to the network, while assuring complete repetitiveness of the experiment across different software solutions. MINT is able to orchestrate the recreation of accidental faults such as packet dropping. However, the effect of such perturbations is just appreciated through performance measures. Since all the analysed platforms are instrumented with probes to observe the performance of the network, their observability is similar to the exhibited by simulators.

5.2.3. Emulation-based approaches

Generally, the easiness to control emulators enhances the repeatability of the evaluation platform. However, the simplification of certain elements such as mobility, unavoidably reduces the accuracy of the experiments with respect to the values obtained through real-world experiments. Regarding the system complexity, the time required to execute experiments considering mobility models is independent from the complexity of the model used. Normally the model is offline precomputed before the experimentation, thus not affecting the experimentation

time.

Typically, emulators can be subdivided into physical and MAC layer emulators. In physical layer ones, emulators mangle the radio signal emitted by the wireless network interface cards of the nodes to mimic the effects that radio waves would experience in a real-world setup. One possibility to do this is to attenuate the emitted signal as shown in [103] using Radio Frequence (RF) attenuators. ORBIT [67] and NetEye [91] are indoors radio grid of fixed nodes that works in that way. ORBIT deploys 400 nodes while NetEye installs up to 130 motes. Since, each node is actually static the mobility of each node in is emulated through a separate mobility server. Unfortunately, the topology generation is limited to the grid mobility model, which reduces the platform configurability. Similarly, CMUWE [68] and TutorNet [81] are indoors evaluation platform defined to be shared by multiple users. They support real devices (up to tens of routers in the case of CMUWE and hundreds of them in the case of TutorNet), applications, and MAC and physical layers on a network-wide scale while maintaining experimental control and repeatability. However, the fact of sharing the same resources of the platform (i.e., the same control network or the same communication channel) with other users concurrently, involves assuming certain degree of intrusiveness in the experiment results.

Since in MAC layer emulators all the nodes are in the same radio range, they filter the packets that should receive each node according to the desired topology: if a node is emulated to be within radio range of another node, a filtering approach allows the exchange of packets between them, if the nodes are out of each others range, such packets are dropped. Thus, each node applies its own rules to determine its visibility with the rest of nodes of the network. Some examples of this approach are Castadiva [69], Mobiemu [70] and Emulab [71]. While the first thee platforms are characterised by the exclusive use of resources, the last one is a shared testbed. By dynamically adding and removing filter rules, the emulator can also create scenarios with node movement. This characteristic enables this type of evaluation platforms to be quite configurable. REFRAHN [72], is an alternative solution characterised by the exclusive use of resources that enables the recreation of both accidental an malicious faults to assess their impact in ad hoc routing protocols. Conversely to the rest of emulators, the capability to inject more than ten different sources of threat encompassing generic (protocol independent) perturbations such as *signal attenuation*, *ambient noise*, *battery extenuation*, *traffic peak*; and protocol-dependent ones that must be instantiated according to the particular nature of each routing protocol such as *sink hole attack*, *replay attack*, *tampering attack*, *selective forwarding attack*, *jellyfish attack*, *flooding attack*, *neighbour sat-*

uration and *sequence number replay*, (protocol dependent), provides REFRAHN a high injectability. This platform is instrumented with probes to take measurements of performance, resilience and energy consumption to provide a high observability of the system.

In general, the scalability and portability of emulators is limited given the level of ambient noise generated by the high concentration of devices in a reduced same area. To face this problem, initiatives such as Senslab [88] and Wisebed [89] provide large federated platforms composed of multiple testbeds. Senslab deploys 1024 motes and it is distributed among 4 interconnected locations in France, 2 of which offer access to mobile nodes implemented by robots. Similarly, Wisebed considers in- and outdoor 750 motes in 9 different locations in Europe and mobility is supported using 40 mobile robots.

5.3. Analysis of evaluation strategies

Model-based evaluation platforms provide a high degree of controllability, repeatability, configurability and scalability to study different types of networks in a repeatable way. However, its most important disadvantage is the limited accuracy of results. Most of the results obtained from these works, are based on assumptions which typically simplify in excess the behaviour of the system, which might lead researchers to biased or wrong conclusions. Conversely, the highest degree of applicability and therefore transferability of results, is given in the case of prototype or real deployments. However, experiments are typically non-repeatable given the difficulty of recreating of actual deployments and the unsteadiness of the wireless communication medium. Furthermore, the cost from the viewpoint of the use of real hardware and the manpower required, limits the scalability of the approach. Emulation is a hybrid approach comprised of both real and virtual parts representing a trade-off between prototype-based and model-based proposals. The advantage of emulation environments over real world experiments is the possibility of scaling to larger scenarios with a minor effort. However, the degree of realism in emulation strongly depends on which components or actions of the system are real or virtualised (devices, mobility, network interfaces, etc).

It is worth mentioning that neither simulation, nor real-world prototyping nor emulation present, by design, a significant advantage neither from the viewpoint of observability, nor the capability of introducing threats in the system (defined as injectability). Instead, it depends of the particular evaluation platform to deploy such capacities. From all the evaluation surveyed, REFRAHN is that providing with the highest levels of observability and injectability. This can be explained

because, as seen in Section 4, REFRAHN proposes the evaluator a set of measures a wide variety of resilience measures. In consequence, REFRAHN needs to animate the system with the presence of threats, and instrument different types of probes to observe its behaviour from different perspectives. We claim that more practical evaluation platforms like REFRAHN are required to address the execution of experiments from a resilience viewpoint. Table 7 summarises the different evaluation alternatives referred in this section.

Table 7: Comparative review of evaluation platforms.

	Platform	Controllability	Observability	Intrusiveness	Repeatability	Accuracy	Scalability	Portability	Configurability	Comp. complexity	Injectability
Model	*Opnet* [59]	High	Medium	Low	High	Low	High	High	Medium	High	Low
	NS-2/3 [57, 58]	High	Medium	Low	High	Low	High	High	High	High	Low
	Glomosim [60]	High	Medium	Low	High	Low	High	High	High	High	Low
Emulation	*CMUWE* [68]	Medium	Medium	Medium	Medium	Medium	Medium	Low	Low	Low	Low
	ORBIT [67]	Medium	Medium	Medium	Medium	Medium	High	Low	Low	Low	Low
	Castadiva [69]	Medium	Medium	Low	Medium	Medium	Medium	High	High	Low	Low
	Emulab [71]	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Low	Low	Low
	MobiEmu [70]	Medium	Medium	Low	Medium	Medium	Medium	High	High	Low	Low
	REFRAHN [72]	Medium	High	Low	Medium	Medium	Medium	High	High	Low	High
	NetEye [91]	Medium	High	Low	Medium	Medium	High	High	High	Low	High
	TutorNet [92]	Medium	High	Medium	Medium	Medium	High	High	High	Low	High
	Senslab [88]	Low	Medium	Low	Low	High	High	Medium	Low	Low	Medium
	Wisebed [89]	Medium	High	Medium	Medium	Medium	High	High	High	Low	High
Real-world	*Roofnet* [61]	Medium	Medium	Low	Medium	High	Medium	Medium	Medium	Low	Low
	Floornet [62]	Medium	Medium	Low	Medium	High	Medium	Medium	Medium	Low	Low
	APE [63]	Low	Medium	Low	Low	High	Medium	Medium	Low	Low	Low
	MINT [64]	Low	Medium	Low	Low	High	Medium	Medium	Low	Low	Medium
	Indriya [65]	Low	Medium	Low	Low	High	Medium	Medium	Low	Low	Low
	TWIST [66]	Low	Medium	Low	Low	High	High	Medium	Low	Low	Medium
	Citysense [86]	Low	Medium	Low	Low	High	High	Medium	Low	Low	Medium

6. Analysis and interpretation of results

The analysis of results is crucial as the conclusions of the resilience evaluation will rely on this stage. Once measures processed, resilience evaluators face a crucial problem that strongly influences the analysis of results. To be useful, the measures extracted must be correctly interpreted following systematic criteria. The context where the ad hoc network is deployed is a key factor that should be taken into account, as measures of the same network in different contexts could be interpreted in many different ways. Despite the importance of this point, most of the

evaluation platforms considered in this survey still analyse measures generically without applying a proper interpretation about where the network is deployed in.

As we have seen in the previous section, the controllability and repeatability of experiments is essential in resilience evaluation. However, and without taking importance away from this point, controllability and repeatability also affects other stages of the evaluation process, such as the analysis of results. The reader should understand that resilience evaluation introduces the need of performing a more complex analysis of target systems, considering their behaviour in the presence of faults and attacks, and characterising such behaviour with a larger set of measures, including dependability and security specific ones. This evidence becomes a challenge when considering the evaluation of distributed systems formed by a large and heterogeneous set of sub-systems and components. The challenge is there not only for the amount of measures to consider, but also for their variety of origin and typology. To date, the analysis of evaluation results in ad hoc networks has been an aspect strongly relying on the human factor. Evaluators subjectively interpret measures following key criteria that are usually omitted in the reports. In consequence, repeating the same analysis of measures and obtaining the same conclusions, even when results are the same, becomes a complex task.

The underlying problem of controllability and repeatability in the analysis of measures in the domain of ad hoc networks reveals that most proposals limit their purpose to the delivery of measures. By the time being, the fact of considering a representative set of measures has been traditionally enough to justify their selection [23]. However, the approach to quantitatively analyse such measures is usually focused only on the numerical output, whereas little or no attention is devoted to their interpretation. Without contextualising their meaning throughout factors such as the environment, the type of system targeted, or the evaluation performer, same results may have different interpretations depending on the evaluation consumer's subjectivity. From a practical viewpoint, the comparison among different systems becomes quite hard, if not meaningless. Even if the effort is performed, the analysis and interpretation of results remains an error-prone process requiring a very deep expertise in the domain of ad hoc networks and routing protocols, increasing the *uncertainty* of evaluation analyses, and thus affecting the credibility of the conclusions obtained. This ambiguous interpretation of concepts is commonly known as *semantic heterogeneity* [104].

This challenge could be addressed through a process of *semantic reconciliation* [104]. Such process involves covering the existing gap between the explicit result of the evaluation, that is, the conclusions distilled from the analysis of measures, and the implicit real intention of evaluators, which concerns the interpre-

tation procedure to obtain such conclusions. This fact increases the sensitivity of analyses, potentially revealing surprising insights about the system under evaluation. This approach is specially useful when there is no obvious optimal (or unanimous) solution due to the large number of criteria that need to be taken into account, or when decisions often require the fulfilment of conflicting objectives (e.g., design or choice of ad hoc routing protocols maximising their dependability or performance). It has also the potential for improving the work of system evaluators by leading them to unequivocal and more objective conclusions. Unfortunately, to date, the *semantic reconciliation* remains an rarely issue in the domain of evaluation platforms for ad hoc networks. In particular, REFRAHN is the only approach from the list of evaluation platforms from Table 7 stating the mandatoriness of addressing this challenge. REFRAHN orchestrates a multi-criteria analysis methodology to ease the multiple interpretations that the measures issued from performance, resources consumption and resilience evaluation may have depending on the criteria followed by evaluators. The goal of this methodology is to make explicit the subjective interpretation rules that evaluators typically apply implicitly when determining to what extent measures satisfy evaluation requirements. Doing this in a systematic and repeatable way is essential when different evaluators need to make a fair comparison of their results, so the methodology relies on a mathematical formalism. Second, defining our methodology in such a way it may satisfy the conflicting positions between (i) those evaluation consumers that prefer having all the possible measures as field data for enabling deep result analysis and promote data sharing among community members [105] (e.g., people from academia), and (ii) those adopting a more pragmatical viewpoint that ask for an small set of meaningful and representative scores to characterise, rank and compare evaluated systems [106] (e.g., people from industry). To cope with this goal REFRAHN relies on the notion of quality model, adopted from ISO/IEC 25000 standards [107], to formulate not only rigorous but also usable and flexible interpretation rules.

Despite the efforts done in REFRAHN, there are still open questions requiring further research in the analysis of evaluation measures such as (i) how to systematically aggregate such measures to capture in a single or small set of scores the information required to characterise the overall system quality, and (ii) how to ensure the consistency of interpretations issued from the use of such scores with respect to the conclusions obtained from the direct analysis of evaluation measures. The choice for a representation of measures has important consequences in terms of expressiveness. Simplistic approaches may skew in excess the representation of the model, whereas representations with a high expressiveness can add

unnecessary complexity to the model or can be cumbersome in its use for decision making. Therefore it is important to find an equilibrium between the possibility of representing as much situations as possible but at the same time maintaining a good degree of usability.

7. Conclusion

Resilience is an essential non-functional aspect when studying the effect of faults and changes in ad hoc networks. However, it is to note that current market demands a reduction in the cost of production and the time to commercialisation of solutions and services. Under these conditions it is very difficult to guarantee an acceptable level of resilience. This problem becomes specially meaningful in those application contexts where the incorrect behaviour of the network may imply a severe economic or human loss. Even in less critic environments, resilience may have a decisive impact on the reputation of the service provider, thus conditioning the level of penetration of the product within the market. This fact has increased the need for designing new and efficient techniques and tools to evaluate not only the functional aspects of ad hoc network systems, but also non-functional ones.

This paper has explored existing gaps in the practical evaluation of ad hoc routing protocols to determine which are the impairments limiting a better understanding about the resilience of ad hoc networks. Thus, the lack of resilience measures, the difficulty in recreating the presence of threats, and the absence of approaches to guide the interpretation of issuing results is a challenging task in practice that, to date, limits the achievement of this goal. The deliberated introduction of threats in the system, as well as the consideration of resilience measures to estimate their impact on ad hoc networks and the systematisation of user-friendly techniques to interpret them in a repeatable way could result useful to evaluate the ability of routing protocols to keep on providing the routing service despite the activation of faults and the presence of changes. To date, the interest is not only in proposing new fault tolerance mechanisms to face such impairments, but in providing methodologies and tools to introduce these aspects within the evaluation of ad hoc networks. There exist many and varied challenges in the deployment of ad hoc routing protocols, but the need for frameworks to evaluate and justify their resilience is, without doubt, one of the most important. By the time being, most initiatives in the domain of ad hoc networks, like CeNSE [108], WiSeNts [109] or GENI [110] focus their goal in the deployment of ad hoc networks formed by massively interconnected devices measuring and processing

data in real-time. However, such efforts will remain questionable in practice while suitable techniques to guarantee acceptable levels of resilience in their implementations remain unavailable. The resilience evaluation of ad hoc routing protocols opens new opportunities to study how the dynamic features of ad hoc networks may affect their dependability. This aspect is very important to estimate how good (or bad) a given routing protocol or fault tolerance complement adapts to changes in the environment. Likewise, resilience evaluation could be very useful to assess the risk of subjecting (new or existing) fault tolerance mechanisms to the presence of known threats they were designed (or not) against.

Ad hoc networks are now in a stage where more practical aspects need to be investigated. Among them, resilience evaluation is essential to drive a stronger market penetration in the context of medium- and large-scale wireless networks and to enable the use of new applications and services in existing networks, thus stimulating market competition, as well as business models and realistic use cases to make this technology appealing for public institutions and private companies.

Acknowledgements

This work is partially supported by the Spanish project ARENES (TIN2012-38308-C02-01), the ANR French project AMORES (ANR-11-INSE-010), and the Intel Doctoral Student Honour Programme 2012.

References

[1] H. Bai, et al., Wireless sensor network for aircraft health monitoring, in: Proceedings of the First International Conference on Broadband Networks (BROADNETS), pp. 748–750.

[2] U.S. Executive Office of the President, The Federal Response to Hurricane Katrina: Lessons Learned, 2006.

[3] I. F. Akyildiz and others, Wireless mesh networks: a survey, IEEE Radio Communications 43 (2005) S23–S30.

[4] M. Asefi, J. Mark, X. Shen, A mobility-aware and quality-driven retransmission limit adaptation scheme for video streaming over vanets, in: IEEE Global Telecommunications Conference (GLOBECOM), pp. 1 –5.

[5] E. Sakhaee, A. Jamalipour, N. Kato, Aeronautical ad hoc networks, in: Wireless Communications and Networking Conference, 2006. WCNC 2006. IEEE, volume 1, pp. 246 –251.

[6] Y.-B. Ko, N. H. Vaidya, Location-aided routing (lar) in mobile ad hoc networks, Wirel. Netw. 6 (2000) 307–321.

[7] C. Perkings, Ad hoc On-Demand Distance Vector(AODV) Routing, RFC 3561 (2003).

[8] D. Johnson, et al., Dynamic Source Routing protocol(DSR), RFC 4728 (2007).

[9] TORA, Temporally-Ordered Routing Algorithm (TORA), [Online]. Available: http://tools.ietf.org/html/draft-ietf-manet-tora-spec-04, 2001.

[10] SrcRR, SrcRR: A High-Throughput Routing Protocol for 802.11 Mesh Networks, [Online]. Available: http://pdos.csail.mit.edu/rtm/srcrr-draft.pdf, 2004.

[11] S. Murthy, J. J. Garcia-Luna-Aceves, An efficient routing protocol for wireless networks, Mob. Netw. Appl. 1 (1996) 183–197.

[12] T. Clausen and P. Jacquet, Optimized Link State Routing Protocol(OLSR), RFC 3626 (2003).

[13] C. E. Perkins, P. Bhagwat, Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers, in: Proceedings of the conference on Communications architectures, protocols and applications, SIGCOMM '94, ACM, New York, NY, USA, 1994, pp. 234–244.

[14] J. Chroboczek, BABEL, [Online]. Available: http://www.pps.jussieu.fr/ jch/software/babel/, 2010.

[15] D. Johnson, N. Ntlatlapa, C. Aichele, A simple pragmatic approach to mesh routing using batman, in: 2nd IFIP International Symposium on Wireless Communications and Information Technology in Developing Countries (CSIR), pp. 1–10.

[16] J.-C. Laprie, Resilience for the scalability of dependability, in: Proceedings of the Fourth IEEE International Symposium on Network Computing and Applications, NCA '05, pp. 5–6.

[17] L. Moreira, Ft-cowisenets: A fault tolerance framework for wireless sensor networks, International Conference on Sensor Technologies and Applications (2007) 289–294.

[18] B. Wu, et al., A survey on attacks and countermeasures in mobile ad hoc networks, in: Wireless/Mobile networks security, Springer-Verlag, 2006.

[19] Y.-C. Hu, D. B. Johnson, A. Perrig, Sead: Secure efficient distance vector routing for mobile wireless ad hoc networks, Ad Hoc Networks 1 (2003) 175–192.

[20] C. Adjih, T. Clausen, P. Jacquet, A. Laouiti, P. Muhlethaler, D. Raffo, Securing the olsr protocol, in: Proceedings of Med-Hoc-Net, pp. 25–27.

[21] M. G. Zapata, Secure ad hoc on-demand distance vector routing, ACM SIGMOBILE Mobile Computing and Communications Review 6 (2002) 106–107.

[22] S. Buchegger, J.-Y. Le Boudec, Performance analysis of the confidant protocol, in: Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing, ACM, pp. 226–236.

[23] S. Kurkowski, T. Camp, M. Colagrosso, Manet simulation studies: the incredibles, SIGMOBILE Mob. Comput. Commun. Rev. 9 (2005) 50–61.

[24] A. C. Viana, S. Maag, F. Zaidi, One step forward: Linking wireless self-organizing network validation techniques with formal testing approaches, ACM Comput. Surv. 43 (2011) 7:1–7:36.

[25] W. Kiess, M. Mauve, A survey on real-world implementations of mobile ad-hoc networks, Ad Hoc Netw. 5 (2007) 324–339.

[26] M. Kropff, T. Krop, M. Hollick, P. S. Mogre, R. Steinmetz, A survey on realworld and emulation testbeds for mobile ad hoc networks, in: TRIDENTCOM.

[27] A. Gluhak, S. Krco, M. Nati, D. Pfisterer, N. Mitton, T. Razafindralambo, A survey on facilities for experimental internet of things research, IEEE Communications Magazine 49 (2011) 58–67.

[28] Q. Gu, Secure routing protocols, in: Encyclopedia of Cryptography and Security, Springer, 2011, pp. 1130–1134.

[29] S. Sharma, S. K. Jena, A survey on secure hierarchical routing protocols in wireless sensor networks, in: Proceedings of the 2011 International Conference on Communication, Computing & Security, ICCCS '11, ACM, New York, NY, USA, 2011, pp. 146–151.

[30] C. Karlof, D. Wagner, Secure routing in wireless sensor networks: Attacks and countermeasures, Ad hoc networks 1 (2003) 293–315.

[31] Poonam, K. Garg, M. Misra, Trust based security in manet routing protocols: A survey, in: Proceedings of the 1st Amrita ACM-W Celebration on Women in Computing in India, A2CWiC '10, ACM, New York, NY, USA, 2010, pp. 47:1–47:7.

[32] D. Djenouri, L. Khelladi, N. Badache, A survey of security issues in mobile ad hoc networks, IEEE communications surveys 7 (2005) 2–28.

[33] S. Maag, F. Zaidi, Testing methodology for an ad hoc routing protocol, in: Proceedings of the ACM international workshop on Performance monitoring, measurement, and evaluation of heterogeneous wireless and wired networks, PM2HW2N '06, pp. 48–55.

[34] R. Prasad, A. Mihovska, New Horizons in Mobile and Wireless Communications: Ad hoc networks and PANs, Mobile Communications, Artech House, 2009.

[35] D. Raffo, Security Schemes for the OLSR Protocol for Ad Hoc Networks, Ph.D. thesis, Universite Paris 6 and INRIA Rocquencourt, 2013.

[36] A. Bustos, J. Friginal, D. de Andrés, J.-C. Ruiz, An aspect-oriented approach to face neighbour saturation issues in proactive ad hoc routing protocols: olsrd as a case study, in: Proceedings of the 1st European Workshop on AppRoaches to MObiquiTous Resilience, ACM, p. 3.

[37] I. Kang, R. Poovendran, Maximizing static network lifetime of wireless broadcast ad hoc networks, in: IEEE International Conference on Communications (ICC), volume 3, pp. 2256 – 2261 vol.3.

[38] J. seung Yeom, N. Wisitpongphan, S. Panichpapiboon, O. Tonguz, A testbed emulator for cross-layer studies in mobile ad hoc wireless networks, in: 3rd International Conference on Testbeds and Research Infrastructure

for the Development of Networks and Communities (TridentCom), pp. 1 –10.

[39] A. Al Hanbali, E. Altman, N. Philippe, A survey of tcp over ad hoc networks, IEEE Communications Surveys and Tutorials 1 (2005) 22–36.

[40] A. Sheth, C. Doerr, D. Grunwald, R. Han, D. Sicker, Mojo: a distributed physical layer anomaly detection system for 802.11 wlans, in: MobiSys '06: Proceedings of the 4th international conference on Mobile systems, applications and services, pp. 191–204.

[41] J. Ben-Othman, A. Hamieh, Defending method against jamming attack in wireless ad hoc networks, in: IEEE 34th Conference on Local Computer Networks (LCN), pp. 758 –762.

[42] Y. Zhou, et al., Balancing the hidden and exposed node problems with power control in csma/cabased wireless networks, in: Proceedings of IEEE Wireless Communications and Networking Conference (WCNC), pp. 683–688.

[43] X. Fu, On traffic analysis attacks and countermeasures, Ph.D. thesis, College Station, TX, USA, 2005. AAI3246468.

[44] C.-T. Li, Y.-P. Chu, Cryptanalysis of threshold password authentication against guessing attacks in ad hoc networks., I. J. Network Security (2009) 166–168.

[45] J. Hoydis, M. Petrova, P. Mahonen, Effects of topology on local throughput-capacity of ad hoc networks, in: IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), pp. 1 –5.

[46] C. Gomez, et al., Improving Performance of a Real Ad Hoc Network by Tuning OLSR Parameters, in: Symposium on Computers and Communications, USA, pp. 16–21.

[47] H. L. Nguyen, U. T. Nguyen, A study of different types of attacks on multicast in mobile ad hoc networks, Ad Hoc Netw. 6 (2008) 32–46.

[48] J.-C. Ruiz, et al., Black Hole Attack Injection in Ad hoc Networks, in: Supplemental Volume of IEEE Dependable Systems and Networks, USA, pp. G34–G35.

[49] M. N. Lima, A. L. dos Santos, G. Pujolle, A survey of survivability in mobile ad hoc networks, IEEE Communications Surveys and Tutorials 11 (2009).

[50] I. Aad, Impact of denial of service attacks on ad hoc networks, IEEE/ACM Trans. Netw. 16 (2008) 791–802.

[51] DBench project consortium, online: http://spiderman-2.laas.fr/DBench/Deliverables/ETIE1.pdf, 2013.

[52] M. MartÃnez, J. Friginal, D. AndrÃ©s, J.-C. Ruiz, Open challenges in the resilience evaluation of ad hoc networks, in: M. Vieira, J. Cunha (Eds.), Dependable Computing, volume 7869 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2013, pp. 194–197.

[53] A. Fehnker, L. Van Hoesel, A. Mader, Modelling and verification of the lmac protocol for wireless sensor networks, in: Proceedings of the 6th international conference on Integrated formal methods, IFM'07, pp. 253–272.

[54] S. Tschirner, L. Xuedong, W. Yi, Model-based validation of qos properties of biomedical sensor networks, in: Proceedings of the 8th ACM international conference on Embedded software, EMSOFT '08, pp. 69–78.

[55] M. Cinque, D. Cotroneo, C. D. Martinio, S. Russo, Modeling and assessing the dependability ofwireless sensor networks, in: IEEE Symposium on Reliable Distributed Systems, SRDS '07, IEEE Computer Society, Los Alamitos, CA, USA, 2007, pp. 33–44.

[56] N. Purohit, R. Sinha, K. Maurya, Simulation study of black hole and jellyfish attack on manet using ns3, in: 2011 Nirma University International Conference on Engineering (NUiCONE), IEEE, pp. 1–5.

[57] Network Simulator 2, Network Simulator 2, [Online]. Available: http://www.isi.edu/nsnam/ns/, 2014.

[58] Network Simulator 3, Network Simulator 3, [Online]. Available: http://www.nsnam.org/, 2014.

[59] Opnet, OPNET Simulator, [Online]. Available: http://www.opnet.com, 2013.

[60] GloMoSim, GloMoSim, [Online]. Available: http://pcl.cs.ucla.edu/projects/glo-mosim/, 2013.

[61] Roofnet testbed, Roofnet testbed, [Online]. Available: http://pdos.csail.mit.edu/roofnet, 2013.

[62] Floornet testbed, Floornet: A Wireless Multihop Testbed, [Online]. Available: http://http://floornet.org/, 2013.

[63] E. Nordstrom, et al., A testbed and methodology for experimental evaluation of wireless mobile ad hoc networks, in: Testbeds and Research Infrastructures for the Development of Networks and Communities, Italy, pp. 100–109.

[64] MINT testbed, An Autonomic Reconfigurable Miniaturized Mobile Wireless Experimentation Testbed , [Online]. Available: http://www.ecsl.cs.sunysb.edu/mint/, 2013.

[65] M. Doddavenkatappa, M. C. Chan, A. L. Ananda, Indriya: A low-cost, 3d wireless sensor network testbed, in: Testbeds and Research Infrastructure. Development of Networks and Communities, Springer, 2012, pp. 302–316.

[66] V. Handziski, A. Köpke, A. Willig, A. Wolisz, Twist: a scalable and reconfigurable testbed for wireless indoor experiments with sensor networks, in: Proceedings of the 2nd international workshop on Multi-hop ad hoc networks: from theory to reality, ACM, pp. 63–70.

[67] ORBIT project, The ORBIT radio grid emulator (ORBIT), [Online]. Available: http://www.orbit-lab.org/, 2013.

[68] CMUWE, The Carnegie Mellon University Wireless Emulator, [Online]. Available: http://www.cs.cmu.edu/ emulator/, 2013.

[69] J. Hortelano, et al., Castadiva: A Test-Bed Architecture for Mobile AD HOC Networks, in: IEEE Symposium on Personal, Indoor and Mobile Radio Communications, pp. 1–5.

[70] MobiEmu, Mobility Emulator (MobiEmu), [Online]. Available: http://mobiemu.sourceforge.net/, 2013.

[71] Emulab, Emulab - Network Emulation Testbed, [Online]. Available: http://www.emulab.net/, 2013.

[72] J. Friginal, An Experimental Methodology to Evaluate the Resilience of Ad Hoc Routing Protocols, Editorial Universitat Politecnica de Valencia, 2013.

[73] D. A. Maltz, J. Broch, D. B. Johnson, Lessons from a full-scale multihop wireless ad hoc network testbed, Personal Communications, IEEE 8 (2001) 8–15.

[74] D. Rastogi, S. Ganu, Y. Zhang, W. Trappe, C. Graff, A comparative study of aodv and olsr on the orbit testbed, in: IEEE Military Communications Conference (MILCOM) 2007., IEEE, pp. 1–7.

[75] J. Hortelano, J.-C. Cano, C. T. Calafate, P. Manzoni, Evaluating the performance of real time videoconferencing in ad hoc networks through emulation, in: 22nd Workshop on Principles of Advanced and Distributed Simulation (PADS'08)., IEEE, pp. 119–126.

[76] X. Huang, S. Ganapathy, T. Wolf, A scalable distributed routing protocol for networks with data-path services., in: ICNP, pp. 318–327.

[77] G. Daneels, Analysis of the BMX6 routing protocol, Ph.D. thesis, University of Antwerp, 2013.

[78] A. Medina Santos, et al., Comparativa de los protocolos aodv y olsr con un emulador de redes ad-hoc (2006).

[79] B. Seshasayee, K. Schwan, Mobile service overlays: Reconfigurable middleware for manets, in: Proceedings of the 1st international workshop on Decentralized resource sharing in mobile computing and networking, ACM, pp. 30–35.

[80] X. Liu, H. Zhang, Q. Xiang, X. Che, X. Ju, Taming uncertainties in real-time routing for wireless networked sensing and control, IEEE Transactions on Smart Grid 4 (2013) 288–301.

[81] O. Gnawali, L. Guibas, P. Levis, A case for evaluating sensor network protocols concurrently, in: Proceedings of the fifth ACM international workshop on Wireless network testbeds, experimental evaluation and characterization, ACM, pp. 47–54.

[82] K. Heurtefeux, H. Menouar, Experimental evaluation of a routing protocol for wireless sensor networks: Rpl under study, in: Wireless and Mobile Networking Conference (WMNC), 2013 6th Joint IFIP, IEEE, pp. 1–4.

[83] H. Schaffers, A. Sallstrom, M. Pallot, J. M. Hernández-Muñoz, R. Santoro, B. Trousse, Integrating living labs with future internet experimental platforms for co-creating services within smart cities, in: 2011 17th International Conference on Concurrent Enterprising (ICE), IEEE, pp. 1–11.

[84] D. Aguayo, J. Bicket, S. Biswas, D. S. De Couto, R. Morris, Mit roofnet implementation (2003).

[85] O. Landsiedel, E. Ghadimi, S. Duquennoy, M. Johansson, Low power, low delay: opportunistic routing meets duty cycling, in: Proceedings of the 11th international conference on Information Processing in Sensor Networks, ACM, pp. 185–196.

[86] R. N. Murty, G. Mainland, I. Rose, A. R. Chowdhury, A. Gosain, J. Bers, M. Welsh, Citysense: An urban-scale wireless sensor network and testbed, in: Technologies for Homeland Security, 2008 IEEE Conference on, IEEE, pp. 583–588.

[87] X. Ni, K.-c. Lan, R. Malaney, On the performance of expected transmission count (etx) for wireless mesh networks, in: Proceedings of the 3rd International Conference on Performance Evaluation Methodologies and Tools (ValueTools), pp. 77:1–77:10.

[88] C. B. Des Rosiers, G. Chelius, E. Fleury, A. Fraboulet, A. Gallais, N. Mitton, T. Noël, et al., Senslab very large scale open wireless sensor network testbed, in: Proc. 7th International ICST Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities (TridentCOM).

[89] I. Chatzigiannakis, S. Fischer, C. Koninis, G. Mylonas, D. Pfisterer, Wisebed: an open large-scale wireless sensor network testbed, in: Sensor Applications, Experimentation, and Logistics, Springer, 2010, pp. 68–87.

[90] A. K. Pandey, H. Fujinoki, Study of manet routing protocols by glomosim simulator, International Journal of Network Management 15 (2005) 393–410.

[91] X. Ju, H. Zhang, D. Sakamuri, Neteye: a user-centered wireless sensor network testbed for high-fidelity, robust experimentation, International Journal of Communication Systems 25 (2012) 1213–1229.

[92] TutorNet, Tutornet: A Tiered Wireless Sensor Network Testbed, [Online]. Available: http://enl.usc.edu/projects/tutornet, 2010.

[93] A. Ceccarelli, A. Bondavalli, D. Iovino, Trustworthy evaluation of a safe driver machine interface through software-implemented fault injection, in: Proceedings of the 2009 15th IEEE Pacific Rim International Symposium on Dependable Computing, PRDC '09, IEEE Computer Society, Washington, DC, USA, 2009, pp. 234–241.

[94] DBench, Dependability Benchmarking, IST Programme, European Commission, IST 2000-25425, [Online]. Available: http://www.laas.fr/DBench, 2013.

[95] E. Gilbert, Controllability and observability in multivariable control systems, SIAM Journal of Control 1 (963) 128 –151.

[96] Guides in metrology, Guide to the Expression of Uncertainty in Measurement (GUM) and International Vocabulary of Metrology (VIM), [Online]. Available: http://www.bipm.org/utils/common/documents/jcgm/JCGM_200_2008.pdf, 2013.

[97] TPC standard, Transaction Processing Performance Council, [Online]. Available: http://www.tcp.org/, 2013.

[98] K. Ogata, Modern Control Engineering, Fifth Edition, Prentice-Hall electrical engineering series. Instrumentation and controls series, Prentice Hall, 2010.

[99] Y. Chen, F. Raab, R. H. Katz, From TPC-C to Big Data Benchmarks: A Functional Workload Model, Technical Report UCB/EECS-2012-174, EECS Department, University of California, Berkeley, 2012.

[100] F. Zyulkyarov, A. Cristal, S. Cvijic, E. Ayguade, M. Valero, O. Unsal, T. Harris, Wormbench: a configurable workload for evaluating transactional memory systems, in: Proceedings of the 9th workshop on MEmory performance: DEaling with Applications, systems and architecture, MEDEA '08, ACM, New York, NY, USA, 2008, pp. 61–68.

[101] J. Yoon, M. Liu, B. Noble, Random waypoint considered harmful, in: INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies, volume 2, pp. 1312 – 1321 vol.2.

[102] H. Lundgren, E. Nordstrom, C. Tschudin, The gray zone problem in ieee 802.11b based ad hoc networks, SIGMOBILE Mob. Comput. Commun. Rev. 6 (2002) 104–105.

[103] P. Johnson, E. Nourbakhsh, T. R. Burchfield, J. Dix, R. Prakash, S. Venkatesan, N. Mittal, Assert: Advanced wireless environment research testbed, in: SenSys '09: Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems, pp. 297–298.

[104] A. Anaby-Tavor, A. Gal, A. Trombetta, Evaluating matching algorithms: the monotonicity principle, in: IIWeb, pp. 47–52.

[105] K. Kanoun, Y. Crouzet, A. Kalakech, A.-E. Rugina, P. Rumeau, Benchmarking the dependability of windows and linux using postmark/spl trade/ workloads, in: 16th IEEE International Symposium on Software Reliability Engineering (ISSRE), pp. 10 –20.

[106] European New Car Assessment Programme (EuroNCAP), online: www.euroncap.com/, 2013.

[107] International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC), ISO/IEC 25000. Software Engineering - Software product Quality Requirements and Evaluation (SQuaRE) - Guide to SQuaRE, Geneve ISO, 2010.

[108] Central Nervous System for the Earth (CeNSE), online: http://www.hpl.hp.com/news/2009/oct-dec/cense.html, 2013.

[109] Cooperating Embedded Systems for Exploration and Control featuring Wireless Sensor Networks (WiSeNts), online: http://www.embedded-wisents.org, 2013.

[110] GENI project, The Global Environment for Network Innovations (GENI), [Online]. Available: http:www.geni.net/, 2013.