Cross-Layer Metrics for Reliable Routing in Wireless Mesh Networks*

Stefano Paris*, Cristina Nita-Rotaru†, Fabio Martignon‡ and Antonio Capone§

*LIPADE
Université Paris Descartes
stefano.paris@parisdescartes.fr

†Dep. of Computer Science
Purdue University
crisn@cs.purdue.edu

‡LRI
Université Paris-Sud
fabio.martignon@lri.fr

§DEI
Politecnico di Milano
capone@elet.polimi.it

***Abstract*—Wireless Mesh Networks (WMNs) have emerged as a flexible and low-cost network infrastructure, where heterogeneous mesh routers managed by different users collaborate to extend network coverage.**

This paper proposes a novel routing metric, EFW (Expected Forwarded Counter), and two further variants, to cope with the problem of selfish behavior (i.e., packet dropping) of mesh routers in a WMN. EFW combines, in a cross-layer fashion, routing-layer observations of forwarding behavior with MAC layer measurements of wireless link quality to select the most reliable and high-performance path.

We evaluate the proposed metrics both through simulations and real-life deployments on two different wireless testbeds, performing a comparative analysis with ODSBR (On-Demand Secure Byzantine Resilient Routing Protocol) and ETX (Expected Transmission Counter). The results show that our cross-layer metrics accurately capture the path reliability, and considerably increase the WMN performance, even when a high percentage of network nodes misbehave.

***Index Terms*—Wireless Mesh Networks, Selfish Nodes, Data Dropping, Routing Metrics, Experimental Testbed.**

I. Introduction

Wireless Mesh Networks (WMNs) have emerged as a technology for next generation wireless networking, fostering the development of new network paradigms such as wireless mesh community networks (WMCNs) [2]. Since many applications envisioned to run on WMCNs have high-throughput requirements, recent research [3], [4] has introduced several link layer metrics that capture the quality of wireless links to select the network paths with the highest delivery rates.

However, most of the proposed metrics have been designed assuming that each wireless mesh router participates honestly in the forwarding process. While this assumption may be valid in a network managed by a single network operator, it is not necessarily met in a network where the participants are managed by different entities that may benefit from not forwarding all the traffic. Specifically, in a WMCN, a selfish user that provides connectivity through his own mesh routers might try to greedily consume the available bandwidth in his favor to the detriment of others, by selectively dropping packets sent by other nodes [2]. Such selfish behavior can cause unfairness and severe performance degradation, since periodic dropping at relaying nodes decreases the throughput of closed loop connections established by other nodes, even when the fraction of dropped packets is small [5], [6].

*Preliminary results of this work have been presented in [1].

Previous work focused primarily on the detection of nodes that exhibit selfish behavior and on their exclusion from the network [7], [8], [9], [10], [11]. Two of these protocols rely on routing metrics that consider the selfish behavior of network nodes [7], [11] to increase the hop count of a network path proportionally to the number of selfish nodes that belong to that path. However, these metrics do not consider the wireless link quality, and thus fail to choose high-throughput paths between a source and a destination in the presence of selfish nodes which drop packets at the network layer.

In this paper we propose a cross-layer metric that selects the path with the highest packet delivery rate considering both the quality of wireless links and the reliability of network nodes. While many factors contribute to the former, like interference and received signal strength, the latter is mainly influenced by the selfishness of the users that control and manage the network devices. In an effort to understand how this issue impacts the performance of WMNs, our work makes the following contributions:

- We design Expected Forwarding Counter (EFW), a new reliability metric that combines information across the routing and MAC layers to cope with the problem of selfish behavior (i.e., packet dropping) of mesh routers in a WMN. Our metric combines direct observation of routing-layer forwarding behavior of neighbors with the MAC-layer quality of wireless links in order to select the most reliable and high-performance path.
- We propose two variants of EFW, Minimum Expected Forwarding Counter (MEFW) and Joint Expected Forwarding Counter (JEFW), which capture the worst and joint dropping behavior of the nodes that have established the wireless link, in order to reduce the complexity of the network topology representation and the signaling overhead.
- We show that our proposed metrics are loop-free when used to construct a hop-by-hop forwarding scheme based on the Dijkstra's algorithm. We also analyze the robustness of the three proposed metrics to selfish nodes trying to manipulate the metric computation and show that MEFW is the most robust to such attacks.
- We integrate the proposed metrics with OLSR [12], a well-known routing protocol for WMNs, and extend the IEEE 802.11 MAC layer through the implementation of a forwarding probability estimation technique that

evaluates the network node reliability in a distributed fashion. To this aim, we developed a customized version of *olsrd* [13] and *madwifi* wireless driver.
- We perform a detailed comparative evaluation of the proposed metrics with ETX and ODSBR using the NS2 simulator [14] and two real-life wireless testbeds. Numerical results show that EFW improves the network performance with respect to existing approaches more than 200% when several selfish mesh routers are placed inside the network. Moreover, the two refined optimizations, MEFW and JEFW, perform closely to EFW, thus representing an effective yet feasible solution for reliable routing in WMCNs.

The rest of the paper is structured as follows: Section II discusses related work. Section III presents the network and adversary models considered in our work. Section IV illustrates the proposed metrics as well as the monitoring mechanism used to evaluate the forwarding behavior of neighbor nodes. Section V analyzes the properties of our proposed metrics. Section VI provides a numerical evaluation of the proposed framework, while Section VII illustrates the results obtained by testing our solution on two real-life testbeds. Finally, concluding remarks describing the main findings of our work are illustrated in Section VIII.

II. Related Work

Several works presented in the recent research literature focus on reliable data transmission in wireless multi-hop networks with selfish participants. In particular, two different approaches have been proposed to address this problem based on *detection techniques* and *incentives* to enforce and obtain the collaboration among network nodes, respectively.

Detection-Based Techniques. Detection-based techniques comprise works like [7], [8], [9], [10], which focus on detecting the dropping actions and, if necessary, excluding the guilty nodes from the network.

ODSBR [7] leverages an active probing technique to detect unreliable links controlled by adversary nodes, and defines a route discovery mechanism to avoid network paths containing such links. A similar detection technique is exploited by Awerbuch et al. in [15] to define a source routing protocol that is also tolerant to arbitrary and adaptive denial of service attacks. Castor [8] is an opportunistic routing protocol that uses both flooding and unicast transmission techniques to deliver reliably the message to the destination. Sprout [10] is a routing protocol that probabilistically generates a multiplicity of link-disjoint paths to reach other network nodes and deliver the messages using the most reliable route. The secure message transmission (SMT) protocol proposed by Papadimitratos et al. in [9] exploits multiple node-disjoint paths to increase the end-to-end delivery rate using a message dispersion scheme that enables the destination to recover the information contained in data packets by increasing its redundancy. JANUS [16] provides a secure and reliable routing framework for hybrid cellular and Wi-Fi networks. In [17] the authors propose ARCS, an innovative routing protocol that enforces the cooperation among selfish nodes, limiting at the same time the damage caused by malicious devices. In [18], Zhang et al. provide a comprehensive analysis of the aforementioned acknowledgment-based detection techniques and compute theoretical bounds on the performance of the main variants of these systems.

Incentive-Based Techniques. Incentive-based approaches propose solutions in which the collaboration emerges as the best strategy for rational and selfish players. The routing task is modeled as a game, defining the utility perceived by a network node as a function of the cost incurred in packet relaying and the reward obtained from the devices interested in the node collaboration (whether source or destination nodes).

In [19], [20], Srinivasan et al. determine the optimal throughput that each node should receive under the assumption that forwarding actions are mainly driven by selfish interests, like battery lifetime, and propose the Generous Tit For Tat (GTFT) algorithm through which nodes converge to the optimal operating point in a distributed fashion. Similarly to the aforementioned works, in [21] the authors prove that in order to maximize the utility and be robust to cheating behavior, the optimal strategy for any node is to relay the same amount of packets forwarded by other nodes. Furthermore, the authors propose a new routing protocol that achieves Pareto optimality, cheat proofness, and absolute fairness. SPRITE [22] defines a rewarding mechanism which enforces forwarding as the best strategy. Anderegg et al. design Ad Hoc-VCG [23], a routing protocol based on the well-known Vickrey, Clarke, and Groves auction, to guarantee that each intermediate node is refunded at least the cost incurred to relay packets, and that it behaves according to the protocol specifications. The Commit scheme [24] further develops this approach to enforce the truthfulness property even when the source node behaves strategically.

The performance of the previous incentive-based schemes is analytically evaluated by Jaramillo et al. in [25]. The modeling and the analysis of their basic properties led to the design of DARWIN, a new protocol robust to imperfect measurements and collusion attacks. In [26] the truthful pricing mechanism proposed by Vickrey, Clarke, and Groves is used to solve a broad class of problems concerning the non-cooperative behavior of intermediate nodes. Zhong et al. in [27] exploit two solution concepts defined in game theory to consider also the collusion among network devices, showing that even if Group Strategy-proof Equilibrium cannot be satisfied at the routing level, their proposed solutions reach Strong Nash Equilibria[1] among network nodes.

Note that all incentive-based approaches capture neither transmission errors nor dropping behavior caused, for example, by external interference or temporary malfunctions, thus leading to the selection of unreliable network paths.

Other protocols that define a rewarding mechanism to foster node cooperation are proposed in [28], [29]. In [28] the authors propose a distributed algorithm based on the concept of reciprocity among nodes, where credit is represented by the amount of traffic directly or indirectly forwarded by other

[1] A solution defining the strategies played by the agents of a coalition is a *Strong Nash Equilibrium*, if it is robust to deviations of any component of the coalition.

network nodes. In [29], Buttyan et al. propose two forwarding approaches, the Packet Purse Model (PPM) and the Packet Trade Model (PTM), through which the intermediate nodes trade in packets.

Our proposed solution falls in the category of systems based on data dropping *detection* techniques. However, the routing metrics we propose are computed using a passive detection technique that does not introduce network overhead. Therefore, our system can be coupled with any of the *detection-based* solutions discussed above, in order to strengthen the network delivery reliability.

Furthermore, even though works like [30] show that ETX does not lead to select the best end-to-end paths in congested wireless networks due to the inaccurate estimation of link quality, the technique we illustrate in this paper, which combines cross-layer measurements, does not focus on link quality and provides an effective yet general solution to detect unreliable network paths due to the presence of selfish nodes. Moreover, any technique able to improve the link quality estimation can be combined with our approach to increase the overall routing performance.

Finally, we observe that unlike path-based approaches like those proposed in [31], [32], our metrics can be integrated in several routing and forwarding schemes as we show in Section V-A, thus representing a viable solution for realistic Wireless Mesh Community Networks, which may adopt a wide set of routing protocols.

III. System Model and Assumptions

This section presents the communication and threat models considered in our architecture, as well as the definitions and assumptions we adopt in the design of our metrics.

A. Network Model

This paper considers a wireless mesh community network composed of two different types of devices: *mesh routers* that form the infrastructure of the WMCN and are maintained by different community users, and *customer devices* that are only interested in the services provided by the WMCN (e.g., Internet access).

Since the network architecture we consider has a hierarchical structure (wireless mesh routers are in fact dedicated nodes which are deployed to offer backhaul services), we assume the existence of a subset of community participants that are liable for all management tasks.

We assume that all mesh routers communicate with each other using the wireless medium; in particular, they use the IEEE 802.11 MAC protocol to coordinate access to the channel. All mesh routers are equipped with at least one omni-directional antenna for backbone communications.

Community users, which are the mesh routers owners, can connect directly to the backbone network with their wireless devices, whereas the customers can only access the WMCN services through mesh routers.

B. Security and Adversary Models

We assume that there exists a public key infrastructure managed by a trusted Certification Authority (CA). For each new mesh router that a community user wants to add to the WMCN, the CA generates a unique public/private key pair and issues a certificate that binds the identity of the mesh router to its public key. The cryptographic keys can be used to implement message authentication schemes similar to those proposed in [33], [34], in order to prevent message forgery and replay attacks. Furthermore, to avoid packet manipulation attacks against data traffic, which may seriously affect the performance of closed-loop connections like TCP, we require that a secure end-to-end tunnel (like IPSec) is established between any two devices that communicate with each other through an application layer protocol.

Note that these attacks are orthogonal to the problem considered in this paper; therefore, the above solutions can be easily integrated with our proposal to increase the overall network security.

We assume that the community users, owners of mesh routers, are rewarded based on the amount of traffic forwarded by their devices. However, since they generate their own traffic, which competes with that of other users and customers, they can exhibit selfish behavior by selectively dropping packets crossing their mesh routers while privileging their own traffic. The rewarding mechanism applied by the community network is out of the scope of this paper, and can be obtained applying, for example, mechanisms like those proposed in [35], [36].

Note, however, that two different rewarding policies can be considered. The first policy does not distinguish, in the packets forwarded by mesh routers, between traffic originated by router owners and traffic of other users and customers, due for instance to higher layer encryption mechanisms. The second policy rewards mesh router owners only for forwarded packets originated by other users and customers. In this case, the user can apply discarding policies only to a subset of his mesh routers in order to damage traffic that competes with his own for some network resources (e.g., the gateway to an ISP).

Finally, we underline that attacks against the routing control plane, in which nodes simply ignore some of the procedures defined by the routing protocol, represent an orthogonal issue to the problem investigated in this paper, and can be addressed using detection schemes like those proposed in [11], [37].

IV. Cross-Layer Routing Metrics Design

This section presents our proposed metrics, the Expected Forwarding Counter (EFW), and two alternative refinements, the Minimum Expected Forwarding Counter (MEFW) and the Joint Expected Forwarding Counter (JEFW).

We first review ETX and illustrate the reasons which motivate the adoption of our proposals. We then show how to combine data-link and network layer measurements to strengthen the overall routing reliability, and describe the mechanisms designed to estimate the dropping probability of nearby nodes.

A. Expected Forwarding Counter Metric (EFW)

Several routing metrics have been proposed in recent years to select the path with the highest delivery rate in wireless multi hop networks. The essence of all these metrics lies in the selection of reliable network paths, avoiding lossy wireless links prone to transmission errors. However, in the presence of selfish mesh routers that drop packets sent by other network nodes, these metrics fail to select the network path with the highest delivery rate, and thus with the highest end-to-end throughput.

Routing metrics for wireless multi hop networks like ETX adopt a probabilistic model to represent the transmission reliability of a wireless link. Specifically, ETX measures the expected number of transmissions, including retransmissions, needed to correctly send a unicast packet over a wireless link. In order to compute ETX, it is necessary to estimate the packet loss probability in both directions, since in wireless networks based on the IEEE 802.11 protocol the destination must acknowledge each received data frame. Let (i, j) be a wireless link established between nodes i and j; p_{ij} and p_{ji} denote the packet loss probability of the wireless link (i, j) in forward and reverse directions, respectively [2]. The probability of a successful transmission on the wireless link (i, j) can therefore be computed as $p_{s,ij} = (1 - p_{ij}) \cdot (1 - p_{ji})$.

Then, the expected number of transmissions necessary to deliver the data packet, considering both its transmission and the successive acknowledgment as required by the IEEE 802.11 protocol, can be evaluated according to expression (1):

$$ETX = \frac{1}{p_{s,ij}} = \frac{1}{(1 - p_{ij}) \cdot (1 - p_{ji})}. \quad (1)$$

Despite the purpose of selecting the most reliable paths, ETX does not model accurately the delivery rate of a network link, since it does not consider the forwarding behavior of the nodes that have established such link. In particular, ETX and its derived metrics, like for example [32], [38], do not take into account that a selfish node might discard the packet after its correct reception, if it benefits from not forwarding it.

Note that the utilization of estimation techniques like EAR (Efficient and Accurate link-quality monitoR) [39], which use the data traffic to monitor the quality of wireless links, permits to detect even those nodes that intentionally ignore the transmission of acknowledgment frames. Therefore, the best strategy for a rational, selfish node is to drop data packets sent by other nodes at the *network* layer instead of the *data-link* layer, after the reception of the data frame and the successive transmission of the acknowledgment. In fact, if the selfish node drops the data packets at the *data-link* layer, without transmitting the acknowledgement after the reception of the corresponding data frame, the sending node would increase the packet loss probability in the reverse direction, $p_{r,ij}$, and thus this selfish action would be considered in the ETX metric by lowering the data-link layer reliability.

To address the problem caused by the dropping behavior of selfish participants, we combine the link quality measured by the ETX routing metric with the forwarding reliability of a relaying node j by improving the probabilistic model on which ETX is based. Let $p_{d,ij}$ be the *dropping* probability at the network layer of node j; then $p_{f,ij} = (1 - p_{d,ij})$ represents its *forwarding* probability. Since a network node can drop selectively the traffic sent by its neighbors, the dropping probability of any node j is identified both by the sending node i and the relaying node j. As a consequence, the probability that a packet sent through a node j will be successfully forwarded can be computed as $p_{fwd,ij} = p_{s,ij} \cdot (1 - p_{d,ij})$.

Then, the expected number of transmissions necessary to have the packet successfully forwarded, EFW, can be measured according to the following equation:

$$EFW_{ij} = \frac{1}{p_{fwd,ij}} = \frac{1}{(1 - p_{ij}) \cdot (1 - p_{ji})} \cdot \frac{1}{(1 - p_{d,ij})}. \quad (2)$$

The first part of equation (2), which coincides with the ETX metric, considers the quality of the physical and MAC layers, whereas our contribution takes into account the *network* layer reliability. Therefore, EFW represents a cross-layer metric that models both the physical conditions of the wireless medium and the selfishness of the node with which the link is established.

In addition to detecting the misbehaving nodes, the representation of the link reliability provided by the EFW metric permits to use the network paths with the highest delivery performance, without pruning the alternative routes that contain selfish nodes.

B. Minimum and Joint Expected Forwarding Counter Metrics (MEFW and JEFW)

The EFW metric requires the network topology to be represented with a directed graph, since the forwarding probabilities of two neighbor nodes i and j may differ (i.e., $p_{fwd,ij} \neq p_{fwd,ji}$), due to their different dropping probabilities (i.e., $p_{d,ij} \neq p_{d,ji}$). More specifically, since $p_{fwd,ij} \neq p_{fwd,ji}$, the communication link that these two nodes can establish has to be represented using two different arcs: (i, j) and (j, i), whose weights are equal to EFW_{ij} and EFW_{ji}, respectively. However, this representation increases the memory required to store the network topology and can lead to selecting different forward and reverse paths for the packets of closed loop connections, like TCP, thus hampering performance.

To address this limitation, we design the Minimum Expected Forwarding Counter (MEFW), a close approximation of the EFW metric that considers only the worst dropping behavior, yet allowing a simpler representation of the network topology using only one arc. Specifically, for each link (i, j) that a node i can establish with each neighbor j, we consider the maximum among the dropping probabilities of the two end nodes of the communication link, according to equation (3):

$$MEFW_{ij} = MEFW_{ji} = \\ = \frac{1}{(1 - p_{ij}) \cdot (1 - p_{ji})} \cdot \frac{1}{(1 - \max\{p_{d,ij}, p_{d,ji}\})}. \quad (3)$$

Even though the MEFW metric simplifies considerably the topology representation, capturing the worst link value measured by the EFW, it requires the exchange of the forwarding probabilities related to the nodes that have established the

[2] $(1 - p_{ij})$ and $(1 - p_{ji})$ are called *link qualities* in forward and reverse direction, respectively.

communication link, in addition to the forward and reverse loss probabilities, resulting in a higher signaling overhead.

To avoid the transmission of the forwarding probability within routing messages, we further refine the EFW metric, proposing the Joint Expected Forwarding Counter (JEFW), where both forwarding probabilities are multiplied to take into account the cumulative effect of the selfish behavior, according to equation (4). Indeed, the link quality transmitted within routing messages can be replaced with the product of the link quality $(1-p_{ij})$ and the forwarding rate of the corresponding neighbor node $(1-p_{d,ij})$. This improvement maintains the signaling overhead equivalent to that required by the ETX routing metric.

$$JEFW_{ij} = JEFW_{ji} = \\ = \frac{1}{(1-p_{ij})\cdot(1-p_{ji})}\cdot\frac{1}{(1-p_{d,ij})\cdot(1-p_{d,ji})}. \quad (4)$$

C. Forwarding Probability Estimation

The routing metrics we proposed in the previous sections require the estimation of the dropping probability (or equivalently, the forwarding probability) of relaying nodes. The estimation mechanism represents a core component of the entire architecture, since the selection of the most reliable network paths directly depends on its accuracy. Furthermore, the mechanism should incur a low communication overhead to represent a viable alternative to other solutions. To this end, we designed a passive mechanism operating at the MAC layer that evaluates the forwarding behavior of the neighbor nodes using only the local observations gathered from the analysis of the transmissions over the wireless channel. Note that even though the proposed mechanism operates at the MAC layer, it captures the routing-layer forwarding behavior, thus representing a cross layer solution for the estimation of the network layer reliability of network nodes.

Our approach relies on the broadcast nature of the wireless channel, which enables a network node to overhear the transmissions of any device within its radio range, and on the ARQ (Automatic Repeat reQuest) mechanism defined by the IEEE 802.11 standard, which requires the transmission of an acknowledgment to notify the successful reception of each data frame. In order to overhear the packet transmissions of its neighbors, we assume that the wireless interface of each network node is in monitoring mode [40]. We underline that this approach can be applied both with omni-directional and directional antennas as long as the radio interface of the overhearing node is able to decode the captured frames.

Each node i maintains for each neighbor j the number of successfully received packets, that is, the number of frames for which it has received an acknowledgement from the neighbor j, c_{ij}^{ack}, and the number of forwarded packets with the same source address of the acknowledged packets, c_{ij}^{fwd}. The ratio between these two values represents the estimated forwarding probability of the neighbor node, $\hat{p}_{f,ij} = (1 - p_{d,ij}) = {}^{c_{ij}^{fwd}}/_{c_{ij}^{ack}}$.

We emphasize that the monitoring mechanism on node i performs simple and not simultaneous (due to the 802.11 MAC) tasks on the outgoing and incoming links: on the *outgoing* link it must monitor only the IP packets encapsulated in the corresponding 802.11 data frames that have been correctly received by the nearby device j (the next hop on the routing path), and update accordingly the c_{ij}^{ack} counter. On the *incoming* link, which corresponds to the outgoing link of the nearby router j, node i needs to monitor only the transmission of an 802.11 frame containing the same IP packet in order to increment also the c_{ij}^{fwd} counter.

The packets used to update the counters c_{ij}^{ack} and c_{ij}^{fwd} are processed using two auxiliary lists, L_{ij}^{ack} and L_{ij}, which store the packets that have been acknowledged and those for which an acknowledgement has not been overheard yet, respectively. Specifically, when node i overhears the transmission of a packet towards node j (not destined to j), it stores the packet in the list L_{ij} and sets a timer that determines the maximum validity time of such entry for the estimate. The timer is tuned to take into account processing and transmission delays. Upon the reception of the corresponding acknowledgment frame from j, node i increments c_{ij}^{ack} and moves the packet in the list L_{ij}^{ack}, which contains all packets that have already been acknowledged.

If node i overhears the retransmission of the packet stored in the list L_{ij}^{ack} before the timer expires, then it increments the counter c_{ij}^{fwd} and removes the packet from such list; otherwise the packet is removed from L_{ij}^{ack} without increasing the corresponding counter.

We illustrate the forwarding probability evaluation performed by a mesh router by referring to the example network scenario shown in Figure 1, where solid and dotted lines represent the transmission of packets and acknowledgments, respectively. When mesh router $N1$ receives from $N2$ the acknowledgment for a previously sent packet, $N1$ monitors the wireless channel until it hears the transmission of the same packet performed by $N2$ (towards $N3$, see Figure 1(a)). If such transmission does not occur before the timer expires, $N1$ will conclude that $N2$ has not forwarded its packet and will increment only the counter of the number of acknowledged packets, c_{12}^{ack}; otherwise it will increment also the number of forwarded packets, c_{12}^{fwd}. We underline that the counter c_{12}^{ack} is not increased by $N1$ until the acknowledgment of the previously transmitted packet is received from $N2$, and thus the corresponding packet is not considered in the estimate if the acknowledgment is lost.

To increase the opportunity to detect the forwarding behavior of nearby mesh routers, the monitoring node considers all the packets originated by nodes inside its transmission range. As shown in Figure 1(b), $N1$ considers also the packets transmitted by $N4$. If $N1$ does not hear the retransmission of the acknowledged packet sent by $N4$ before the timeout expires, it will conclude that $N2$ has dropped it; in this case, $N1$ will update only the number of packets acknowledged by $N2$, c_{12}^{ack}. Note that the described monitoring technique may underestimate the neighbor forwarding probability, since traditional medium access protocols, such as the IEEE 802.11 CSMA/CA, guarantee the absence of collisions only at the receiver side, while the nodes that are overhearing the transmission can still be involved in collisions, due for example to

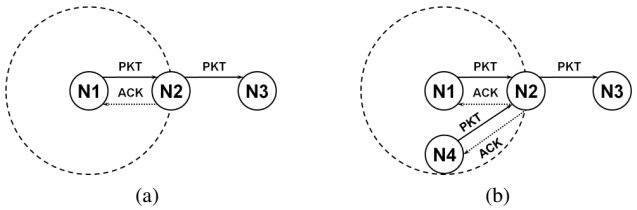


Fig. 1: Example of forwarding probability estimation performed by node $N1$.

the Hidden Terminal problem [41]. Even if these collisions do not affect the correct reception of a packet, they may prevent the correct estimation of the forwarding probability, since the monitoring node may not decode the packet. However, we have experimentally found that in all the considered network scenarios the estimation error incurred by our monitoring mechanism is always lower than 10%, even when the capacity of the wireless channel is saturated (see Section VII-A for more details).

The proposed solution is highly modular, since the scheme used to update the information about the neighbor forwarding rate is implemented only by the monitoring mechanism, while the routing algorithm performs the computation of the three cross-layer metrics expressed in Equations (2), (3) and (4), in addition to the routing messages generation and processing routines. Therefore, the proposed scheme can be seamlessly integrated also in the IEEE 802.11s standard [42] to strengthen its security framework by extending the *Airtime Link Metric*.

V. Analysis of Proposed Metrics

In this section, we first analyze the routing properties of our proposed metrics, showing that they are all loop-free when used by hop-by-hop forwarding schemes. We then study the resilience of our metrics against lying attacks, proving that MEFW is the most robust metric against such attacks.

A. Loop-free Routing

The overall cost $C(P)$ of a network path P composed of n wireless links is equal to the sum of the weights assigned to the links belonging to that path, $c(i;j)$. For the EFW metric, for example, $C(P)$ has the following expression:

$$C(P) = \sum_{(i;j)\in P} c(i;j) = \sum_{(i;j)\in P} EFW_{ij} = \\ = \sum_{(i;j)\in P} \frac{1}{(1-p_{ij})\cdot(1-p_{ji})} \cdot \frac{1}{(1-p_{d,ij})}. \quad (5)$$

Given the linearity property of the function used to compute the path cost, we observe that the three proposed metrics are loop free, since they satisfy the isotonicity property, illustrated in [43]. As a consequence, any hop-by-hop forwarding scheme based on the Dijkstra's algorithm results in a loop free routing protocol when our enhanced metrics are applied. We can therefore formulate the following theorem.

Theorem V.1. *Any hop-by-hop forwarding scheme based on the Dijkstra's algorithm results in a loop free routing protocol when our proposed metrics are used.*

We formally prove Theorem V.1 considering only the EFW metric, since a similar analysis can be easily applied both to MEFW and JEFW.

PROOF: To prove Theorem V.1, we must show that the EFW metric satisfies the isotonicity property as illustrated in [43]. Therefore, we must demonstrate that the order of two paths P_1 and P_2 is not affected by placing either before or after a common path P_3, i.e., $C(P_1) \leq C(P_2) \Rightarrow C(P_3 \oplus P_1) \leq C(P_3 \oplus P_2)$ or $C(P_1) \leq C(P_2) \Rightarrow C(P_1 \oplus P_3) \leq C(P_2 \oplus P_3)$, where $\oplus$ represents the concatenation operator.

The proof follows directly from the definition of the link metric and the path cost, consisting of the sum of all link weights of the path. The addition of a new link does not affect the weight assigned to previous or successive links, since the EFW metric captures the link quality and the neighbor's forwarding probability. From the above consideration, the addition of a new link increments in equal measure the two paths cost, since $C(P \oplus (h;k)) = \sum_{(i;j)\in P} \frac{1}{p_{fwd,ij}} + \frac{1}{p_{fwd,hk}}$. Therefore, the following relation holds: $C(P_1) \leq C(P_2) \Rightarrow C(P_1 \oplus (h;k)) \leq C(P_2 \oplus (h;k))$, since the same quantity has been added to both hand sides of the inequality. Similar considerations can be inferred for proving that $C(P_1) \leq C(P_2) \Rightarrow C((h;k) \oplus P_1) \leq C((h;k) \oplus P_2)$ holds. ■

Note that Theorem V.1 is valid only when the routing protocol does not introduce any additional loop. Specifically, our metrics do not prevent the creation of transitory routing loops, caused for example by the distributed computation of the best paths using outdated network topology information.

B. Metrics Robustness

In general, routing algorithms are vulnerable to attacks where nodes lie about other nodes or attempt to modify information originated by other nodes, with the goal of controlling the path selection. However, since in OLSR the link metrics are flooded in the network to reach nodes farther away than 2 hops, and since we assume that messages are protected against modification or injection by using authentication mechanisms, a misbehaving node can advertise false topology information only about its direct neighbors. For sake of clarity, we refer to this attack as *neighbor metric attack*. Unlike other scenarios, in our application model nodes in the network while attempting to lie about their neighbors, are still interested in maximizing their utility. Below, we describe the function that models the user's utility and analyze the robustness of each metric to *neighbor metric attack*.

The function representing the utility perceived by community user o for the device k can be formulated according to equation (6), which states that when node k lies on the path connecting S and D, the community user o is rewarded for relaying the data traffic flowing from S to D, and thus o perceives a positive utility. On the contrary, if the selected network path does not contain the device, the community user's utility is null.

$$u_k^o(P_{S,D}) = \begin{cases} 1 & \text{if } k \in P_{S,D} \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

In the utility function (6), $P_{S,D}$ represents the network path with the smallest cost connecting nodes S and D, i.e.,

$P_{S,D} = \underset{\substack{P\in\mathcal{P}\\ n_0=S, n_f=D}}{\arg\min} \sum_{(i;j)\in P} MEFW_{(i;j)}$, where $\mathcal{P}$ is the set of all network paths, while n_0 and n_f represent the first and last nodes of the path P, respectively.

Expected Forwarding Counter (EFW). The routing algorithm represents the network topology using a directed graph, when EFW is used as link metric. Therefore, a selfish node cannot restore the network path on which it lies as best alternative, if its dropping behavior is detected by the previous node on the path. However, the selfish node can easily affect the selection of the reverse path of the connection by conducting a *neighbor metric attack*.

Joint Expected Forwarding Counter (JEFW). JEFW is the most vulnerable among the three proposed metrics to *neighbor metric attacks*, since the forwarding rates of both the adjacent nodes are combined to compute the link cost. The problem can be partially addressed by replacing the link quality and the forwarding rate of the neighbor transmitted within the routing message with the product of the two values. In addition to reducing the signaling overhead, this improvement increases the robustness against lying attacks, since a selfish node cannot distinguish between the two factors that contribute to the whole link cost.

Minimum Expected Forwarding Counter (MEFW). The MEFW metric is the most robust of the three metrics against *neighbor metric attacks*. Given the aforementioned definitions, we now demonstrate the following theorem, showing that a rational selfish node cannot increase its utility when conducting the attack.

Theorem V.2. *Given that the utility of selfish network nodes is modeled according to Equation (6), then the MEFW routing metric is robust against* neighbor metric attacks.

PROOF: Let us assume that node $m \in P_{S,D}$ starts dropping packets, and the neighbor node $i \in P_{S,D}$ correctly decreases the advertised forwarding probability, $p_{f,im} = (1 - p_{d,im})$. Furthermore, we assume that after decreasing $p_{f,im}$, this value is the minimum between the two forwarding probabilities of the link connecting nodes m and i, i.e, $p_{f,im} = \min\{p_{f,im}, p_{f,mi}\}$. Therefore, the cost of the link connecting the two nodes is $MEFW_{im} = \frac{1}{(1-p_{im})\cdot(1-p_{mi})} \cdot \frac{1}{p_{f,im}} = MEFW_{mi}$.

We first prove that node m cannot increase its utility announcing a worse or better forwarding probability for node i, when node m still lies on the path on which the data connection is routed ($m \in P_{S,D}$). In fact, if m increases the advertised forwarding probability, i.e., $\tilde{p}_{f,mi} > p_{f,mi}$, then $\min\{p_{f,im}, \tilde{p}_{f,mi}\} = p_{f,im}$; hence $u^o_m(\tilde{P}_{S,D}) = u^o_m(P_{S,D})$, where $\tilde{P}_{S,D}$ represents the network path selected by the routing algorithm when m advertises $\tilde{p}_{f,mi}$. On the contrary, if node m decreases the advertised forwarding probability to a lower value than $p_{f,mi}$, i.e., $\tilde{p}_{f,im} < p_{f,mi} < p_{f,im}$, then we have the following two cases:

- $m \in \tilde{P}_{S,D}$: in this case, the deflating attack has not modified the route, thus $u^o_m(\tilde{P}_{S,D}) = u^o_m(P_{S,D})$;
- $m \notin \tilde{P}_{S,D}$: in this case, the routing decision has been affected by the decreasing action (since there is a network path with a lower cost), thus $u^o_m(\tilde{P}_{S,D}) < u^o_m(P_{S,D})$.

Indeed, from the above considerations, we can conclude that $u^o_m(\tilde{P}_{S,D}) \leq u^o_m(P_{S,D})$. Since the selfish node m cannot increase its utility in any way, disseminating the true forwarding rate represents the best strategy for the device m.

Similar considerations can be inferred when $m \notin P_{S,D}$, as well. Therefore, we can conclude that *true-telling* is the strategy that maximizes the community user's utility for device m. ■

The strategic behavior of the selfish node is illustrated in the example network scenario depicted in Figure 2, where dashed lines represent wireless links connecting two neighbor nodes. Let us assume that a data connection established between nodes S and D is routed on the network path $P^1_{S,D} = \{(S;1),(1;2),(2;D)\}$. We further assume that $m = 2$ is the selfish node, whose dropping behavior is detected by node 1, and $\min\{p_{f,12}, p_{f,21}\} = p_{f,12}$.

The utility perceived by a generic community user o for the dropping device $m = 2$ cannot be increased by disseminating false information about the forwarding rate of node 1, since increasing $p_{f,21}$ does not influence the cost of the path $c(P^1_{S,D})$. On the contrary, decreasing the value of $p_{f,21}$ can only increase the path cost, resulting in the selection of the alternative path $P^2_{S,D} = \{(S;3),(3;4),(4;D)\}$ due to its lower cost $c(P^2_{S,D}) < c(P^1_{S,D})$.

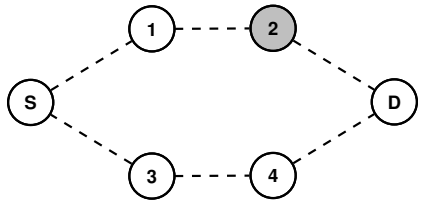


Fig. 2: Example of harmful lying behavior. Theorem V.2 demonstrates that selfish node 2 cannot increase its utility either by inflating or by deflating the forwarding rate advertised for node 1, when MEFW is used as link cost.

VI. Simulation Results

This section discusses the numerical results obtained evaluating the proposed routing metrics with the NS2 simulator [14]. We first describe the experimental methodology employed in our simulations, then we illustrate the performance improvements obtained using the proposed metrics.

A. Experimental Methodology

Nodes Configuration. All nodes employ the IEEE 802.11a MAC protocol and use the same wireless channel. We use as MAC and physical layers the implementation proposed in [44], since it models both layers more accurately than the basic version provided by NS2, including the cumulative SINR computation, the preamble and PLCP header processing, and a more realistic frame body capture. Regarding the routing protocol, we use OLSR opportunely modified to compute the shortest network paths according to our proposed metrics, and ODSBR (On-Demand Secure Byzantine Resilient Routing), a representative protocol which is based on active probing techniques to detect selfish nodes.

Network Topologies. In our simulations, we consider typical WMCN topologies composed of 49 mesh routers placed over a $1000\ m \times 1000\ m$ area. The maximum channel capacity is 6 Mbit/s, while the transmission range is set to $90\ m$, as suggested in [44]. We compare the proposed metrics (EFW, MEFW, and JEFW) to the standard ETX metric and the ODSBR protocol considering the two following network topologies:

- *Grid Topology*: the mesh routers form a square grid topology.
- *Highly Dense Random Topology*: the nodes are randomly placed over the square area, forming a connected network. The minimum degree of all network nodes is fixed to 7.

Attack Scenarios. We consider the two following scenarios:

- *No Attack*: there are no adversaries in the network. This scenario represents the ideal case and provides an upper bound on network performance for our scheme.
- *Data Dropping Attack*: in this scenario, the adversary nodes vary their packet drop rate in the 0% to 100% range.

Adversary Nodes Placement. To provide a more complete comparison, we also evaluate two different placements of the adversary nodes. Specifically, we consider the following configurations:

- *Anywhere Placement*: any network node can be selected as selfish node.
- *Central Placement*: only nodes placed in the middle of the network topology can be selected to act selfishly.

Data Traffic Pattern. In the *Grid* topology, we establish 7 data connections between each node on the first column and the corresponding destination node at the right end of the same row (the number of connections is therefore equal to the 7 rows in the grid). This data traffic pattern permits to evaluate the proposed routing metrics in congested networks, and to which extent the intra-flow and inter-flow interference impairs the accuracy of the monitoring technique.

In the *Random* topology, the source and destination nodes of the data connections are randomly selected among all network nodes. For a fair comparison of the two topologies, we set up the same number of connections in both network topologies. However, due to the random selection of the source and destination nodes of the data connections, only the *Central* placement attack is evaluated in the *Random* topology.

We evaluate the network performance using CBR and FTP traffic transmitted over UDP and TCP connections, respectively. In particular, CBR sources generate data packets of 1000 bytes with a rate equal to 100 kbit/s, whereas FTP traffic is transmitted over TCP connections using the Selective Acknowledgment (SACK) mechanism.

Performance Metrics. We consider as performance metrics the *Average Packet Delivery Rate* (PDR) achieved by the 7 UDP connections and the network fairness measured using the *Jain's Fairness Index*, defined according to equations (7) and (8), respectively. In these equations x_i and y_i represent the PDR and the average throughput (in kbit/s) of the i^{th} connection, whereas n represents the number of connections established in the network.

$$\text{Average PDR} \triangleq \frac{1}{n} \cdot \sum_{i=1}^{n} x_i \tag{7}$$

$$\text{Jain's Fairness Index} \triangleq \frac{(\sum_{i=1}^{n} y_i)^2}{n \cdot \sum_{i=1}^{n} {y_i}^2} \tag{8}$$

The value of both metrics lies between 0 and 1. As for the Jain's Fairness Index, the higher the value, the greater the network fairness among the n connections. Specifically, when the Jain's index is equal to 1, all connections experience the same throughput, whereas a value equal to k/n indicates that only k out of n connections receive an equal share of the network bandwidth.

We also evaluate the strength of the attacks described above on 7 long-lived TCP connections using as metric the Goodput Decrease Ratio (GDR), defined in [7] as:

$$\text{GDR} \triangleq \frac{z_n - z_a}{z_n}, \tag{9}$$

where z_a and z_n represent the *Average Goodput* when the network is under attack and not under attack, respectively. Therefore, the higher is the GDR, the lower is the resilience of the network against the attack.

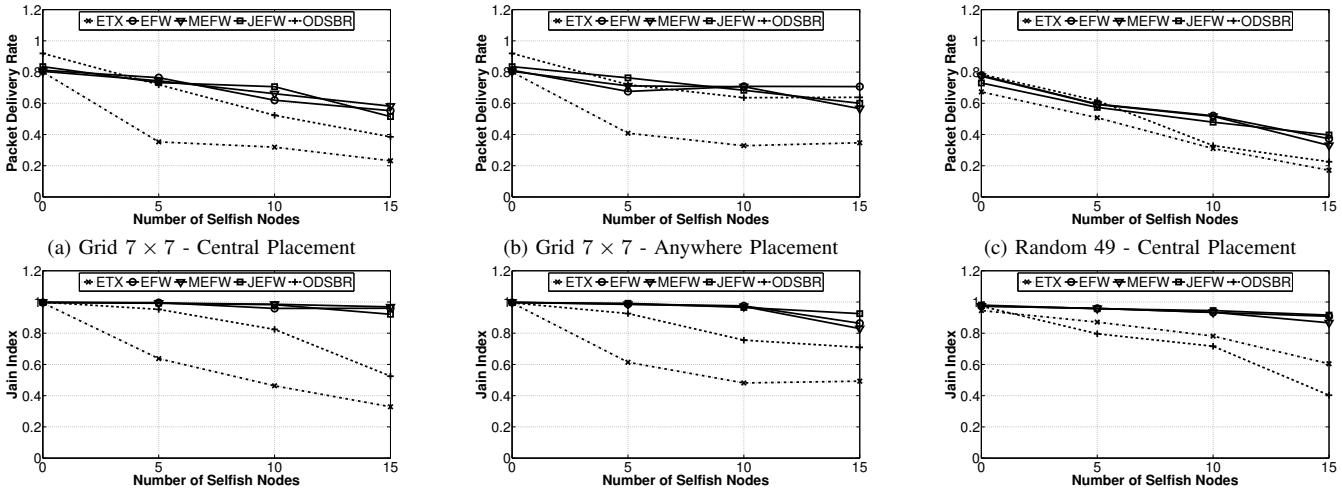
For each scenario we performed 10 independent measurements, computing very narrow 95% confidence intervals. We underline that in the worst case scenario the size of the confidence interval was approximately equal to 10% of the measured mean value. The simulation time on which we evaluated the performance was equal to 300 seconds.

B. Performance Analysis with Connection-less Traffic

Effect of Adversary Size. We first evaluate the effect of the number of adversary nodes on the network performance using the three proposed metrics, in terms of packet delivery rate and fairness of the established CBR connections. We vary the percentage of adversary nodes in the 10% to 30% range. The mesh routers selected as adversaries drop all the traffic sent by other nodes.

Figures 3(a) and 3(b) show the average PDR as a function of the number of adversary nodes in the *Grid* topology considering the *Central* and *Anywhere* placements, respectively.

As expected, the three proposed metrics increase the resilience against the considered attack, since the delivery rate experienced by all CBR connections is enhanced with respect to the baseline approach (ETX metric). In particular, the PDR using the ETX metric decreases quickly in the presence of adversary nodes. In the *Central* placement, which represents the worst case scenario, 15 adversary nodes (30% of the overall number of network nodes) cause an average PDR drop of 70%, considerably greater than the delivery degradation experienced using our proposed metrics, whose PDR reduction is less than 35%. This reflects both the inability of ETX to model the dropping behavior of the relaying nodes and the inherently uniform structure of the *Grid* topology, where even a low number of dropping mesh routers placed in sensitive positions can partition the network and cause severe throughput degradations.



(a) Grid 7 × 7 - Central Placement (b) Grid 7 × 7 - Anywhere Placement (c) Random 49 - Central Placement

(d) Grid 7 × 7 - Central Placement (e) Grid 7 × 7 - Anywhere Placement (f) Random 49 - Central Placement

Fig. 3: **Effect of adversary size on UDP connections**. Average PDR and Jain's Fairness Index measured in the *Grid* and *Random* network topologies as a function of the number of adversary nodes.

Figure 3(c) illustrates the results measured in the *Random* topology. The PDR obtained using the ETX metric decreases almost linearly, since in this case the network presents a higher connectivity that, in turn, increases the number of available paths and thus the survivability to the attack. However, the higher proximity of the network nodes reduces the spatial reuse and increases the network interference, since all nodes periodically broadcast their topology information.

To provide a more in-depth comparison, we also measured the Jain's Fairness Index, which provides an indication of the variance of the delivery rate, and thus the throughput, of the CBR connections. The corresponding results measured in the *Grid* topology considering the *Central* and *Anywhere* placements are illustrated in Figures 3(d) and 3(e), respectively, whereas Figure 3(f) shows the performance in the *Random* network.

As shown in the Figures, the fairness in the *Grid* topologies keeps decreasing as long as the number of adversary mesh routers increases. However, as discussed above, the lower vulnerability of the *Random* network reduces also the network unfairness.

It can be further observed from Figures 3(a)-3(f) that our metrics allocate the available network bandwidth much more fairly than ODSBR. This result shows that our solutions permit to improve the overall network robustness against selfish nodes, since an adversary could exploit the ODSBR's unfairness to target the dropping attack against a specific data connection, in order to increase the network bandwidth for its own data traffic.

All previous figures highlight that the proposed metrics improve the network fairness, reducing the convenience of the dropping attack as a means to greedily consume the available network bandwidth. Specifically, even in the presence of a high number of adversary nodes, the routing algorithm coupled with our metrics is able to restore the network fairness among all data connections.

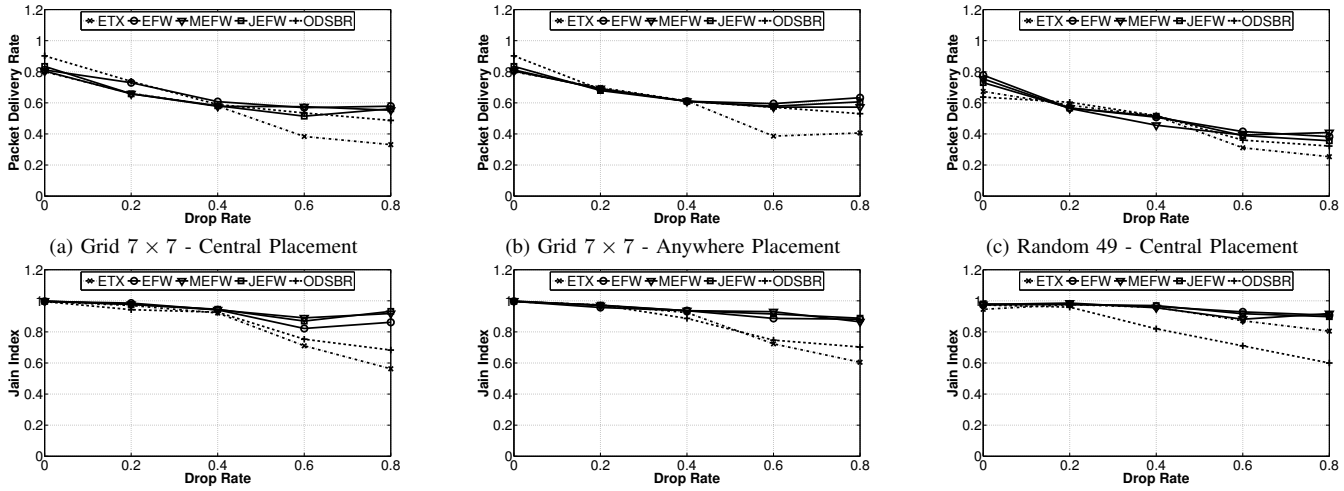
Finally, we observe that the utilization of our metrics within reactive routing protocols (like ODSBR), which are characterized by a lower signaling overhead than proactive protocols, would increase the achievable PDR. The evaluation of our metrics using other routing protocols is left as future work.

Effect of Drop Rate. The second set of simulated scenarios, whose results are illustrated in Figure 4, aims to evaluate the effectiveness of the three proposed metrics when the nodes selected to act selfishly drop only *some* traffic that should be forwarded. In the following simulations, the number of adversary mesh routers is fixed and equal to 30% of the total number of network nodes (i.e., 15 nodes are selected randomly as adversaries), while their drop rates vary from 0% to 80%.

It can be observed that in all these experimental scenarios, the three proposed metrics outperform the baseline metric (ETX) only when the drop rate is higher than 40%. This is due to the cross-layer nature of these metrics, which model both the data-link and the network layer reliabilities in the computation of the cost assigned to each network link. In fact, in a congested network, where the high channel contention causes a degradation of the link reliability, the routing decision is mainly driven by the cost that models the quality of the wireless link.

However, as the dropping attack becomes more severe, the PDR obtained using the ETX metric keeps decreasing, whereas our proposed metrics improve significantly the performance. Specifically, the performance degradation caused by the presence of adversary nodes is 60% and 30% using ETX and our metrics, respectively.

As for the network fairness, Figures 4(d), 4(e), and 4(f) illustrate the Jain's Fairness Index in the *Grid* and *Random* topologies. It can be observed that these results confirm the trends obtained under the attack described above. Specifically, ODSBR is unable to restore the fairness among data connections, thus increasing the vulnerability of the network against data dropping attacks specifically targeted against a subset of connections.



(a) Grid 7 × 7 - Central Placement (b) Grid 7 × 7 - Anywhere Placement (c) Random 49 - Central Placement

(d) Grid 7 × 7 - Central Placement (e) Grid 7 × 7 - Anywhere Placement (f) Random 49 - Central Placement

Fig. 4: **Effect of drop rate on UDP connections**. Average PDR and Jain's Fairness Index measured in the *Grid* and *Random* network topologies as a function of the drop rate (the number of adversary nodes is fixed and equal to 30%).

In addition to confirming the validity of the proposed approaches, Figures 3 and 4 show that in congested networks, installing a relatively high number of adversary nodes that drop less than 40% of the data traffic represents a better strategy for selfish community users than installing a low number of adversary nodes that drop all the data traffic. In the presence of adversary nodes with high dropping rates, the proposed metrics restore the network fairness, distributing the damage among all data connections, thus reducing the effectiveness of the attack.

C. Performance Analysis with Connection-Oriented Traffic

In this section, we evaluate the effect of the dropping behavior on the performance of closed-loop, TCP connections. For the sake of brevity, we only show the performance achieved by the FTP connections varying the percentage of adversary nodes in the 0% to 30% range.

Figures 5(a) and 5(b) show the GDR as a function of the number of adversary nodes in the *Grid* topology considering the *Central* and *Anywhere* placements, whereas Figure 5(c) illustrates the Jain's Fairness Index measured in the same topology.

As expected, the dropping attack highly affects the performance of TCP connections, since periodic packet losses lead to a decrease of the congestion window, which in turn reduces the goodput and the fairness of closed-loop connections [5]. However, as illustrated in these Figures, our solution permits to enhance the resilience against dropping attacks on connection oriented connections, increasing the achievable delivery rate. Indeed, the GDR increases more sharply using the baseline ETX metric or the ODSBR protocol than using our metrics.

We underline that congestion control algorithms implemented by TCP reduce the opportunity to detect the forwarding behavior of intermediate nodes, since they decrease the transmission rate when the connection experiences severe losses. To mitigate this problem, we can increase the validity time of the routes computed by OLSR.

VII. Experimental Study

In this section, we illustrate the experimental results obtained evaluating our solution on two real-life wireless mesh networks, one deployed within the Computer Science Department building at Purdue University and the other by the ORBIT consortium [45], respectively.

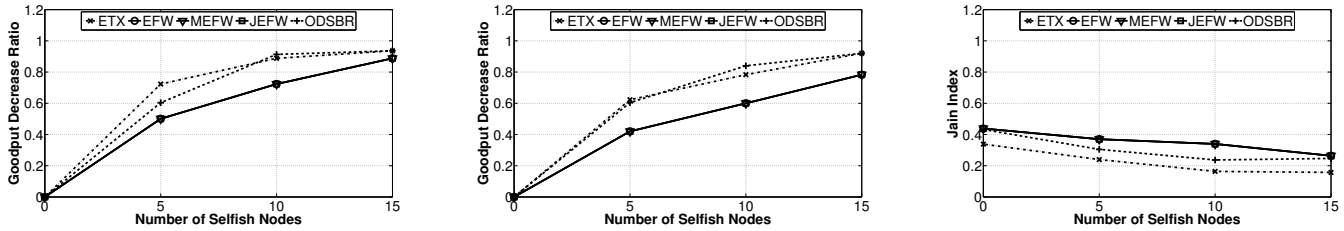
We first describe the implementation of the monitoring mechanism used by the *olsrd* routing process, then we illustrate the results we obtained on the two testbeds.

A. Prototype Implementation

We implemented two alternative versions of the monitoring mechanism: the first version runs as a *user-space* process exploiting the *libpcap* library to overhear the traffic transmitted over the wireless channel, whereas the second solution extends the *madwifi* driver with the monitoring capabilities described in Section IV-C.

The user-space version of the monitoring technique requires that the wireless interface works in monitoring mode, in order to overhear both the data and acknowledgment frames. Since some chipsets do not allow the transmission of any frame when the wireless card is in monitoring mode, an additional interface might be required to establish the mesh backbone links among mesh routers. We underline, however, that the use of an additional interface depends only on the chipset capabilities of the wireless card; therefore it does not represent a limitation of our architecture.

In order to reduce the computational overhead caused by monitoring the traffic transmitted over the wireless channel and to overcome the chipset limitation described above, we have also developed a driver version of the monitoring mechanism that requires only a slightly higher processing time than that caused by the computation in ad-hoc/mesh mode, due to the functionalities implemented to estimate the forwarding probability. To this end, we have modified both the receiving and transmission code of the madwifi driver adding the necessary



(a) Grid 7×7 - Central Placement (b) Grid 7×7 - Anywhere Placement (c) Grid 7×7 - Central Placement

Fig. 5: **Effect of adversary size on TCP connections**. GDR and Jain's Fairness Index measured in the *Grid* network topology as a function of the number of adversary nodes.

control structures and functions to estimate the forwarding rate of every neighbor node.

We first consider the two network scenarios already illustrated in Figure 1, on page 6, to measure the estimation error of our implementations. In the first network scenario, depicted in Figure 1(a), a CBR connection is established between nodes $N1$ and $N3$, whereas in the second scenario (Figure 1(b)) an additional CBR connection is set up between nodes $N4$ and $N3$. Solid arrows illustrate the flow of the CBR packets encapsulated in data frames, whereas dashed arrows show the acknowledgment frames captured by node $N1$ during the experiment. In both experiments, we have measured the relative estimation error of node $N2$'s forwarding rate incurred by the two alternative implementations of the monitoring mechanism installed on node $N1$, which is illustrated in Table I as a function of the overall traffic originated by nodes $N1$ and $N4$. It can be observed that both the implementations reach a high detection accuracy, even for traffic loads that saturate the network capacity (the maximum aggregated throughput was equal to 1.32 Mb/s). In such extreme condition, the error incurred by both monitoring alternatives is lower than 10%. Moreover, we underline that the estimation scheme can benefit

TABLE I: **Detection Error**. Forwarding rate estimation error performed by the two alternative implementations of the monitoring mechanism in the network topologies illustrated in Figure 1.

One CBR Source (Fig. 1(a))				
CBR Traffic (Mb/s)	**0.1**	**0.5**	**1.0**	**1.5**
User	0%	1%	2%	5%
Kernel	0%	1%	3%	6%

Two CBR Sources (Fig. 1(b))				
CBR Traffic (Mb/s)	**0.1**	**0.5**	**1.0**	**1.5**
User	0%	2%	5%	10%
Kernel	0%	2%	2%	8%

from the utilization of the techniques designed for reducing the effect of the Hidden Terminal Problem, like for example the RTS/CTS handshake, since collisions, which are the main causes affecting the estimation accuracy, are considerably decreased.

In our experiments, we used a fixed transmission rate in the network. Note, however, that rate adaptation mechanisms can impair the detection accuracy of the monitoring mechanism only when the sending rate used by the neighbor to forward packets is strictly higher than its incoming receiving rate. Since the PLCP header is always transmitted at the lowest data rate, a network node can correctly detect the start of a data transmission and stop the monitoring of its neighbor, thus reducing the effect of those observations which would lead to a wrong estimation of the neighbor's forwarding behavior.

Finally, the proposed metrics were developed as a loadable plug-in of *olsrd* [13]. Since simulation results show that the approximated metrics provide similar performance to EFW, we implement only JEFW and MEFW, which represent the solutions with the lowest overhead and with the highest robustness, respectively. For a fair comparison with the NS2 version of OLSR, we disabled the FishEye algorithm implemented by olsrd [13] in order to force the dissemination of Topology Control messages into the entire network.

B. Performance Evaluation on CS Testbed

The first set of experiments that we performed on the testbed deployed at the Computer Science Department of Purdue University (CS testbed) aimed at evaluating the effectiveness of the monitoring mechanism and the proposed metrics against the data dropping attack, i.e., when the nodes selected to act selfishly drop the traffic they should forward.

Testbed Setup. The testbed is composed of 7 nodes statically placed on the second floor of the Computer Science Department, in offices and laboratories, as shown in Figure 6(a). The presence of floor-to-ceiling walls and solid wood doors introduces a significant attenuation of the wireless signal. Therefore, the underlay network topology does not require the setting of filtering rules to enforce a multi-hop communication among the mesh routers. The wireless network that enables the multi-hop communication among the 7 devices that constitute the CS testbed is based on the ad hoc network paradigm.

The nodes are general purpose PC (Dell Optiplex GX620) based on the i386 hardware architecture equipped with two PCI Wifi cards based on the Atheros chipset (specifically, both a Cisco Aironet and an ORiNOCO adapter). Each node runs Linux 2.6.22 as Operating System and the 0.9.4 version of the madwifi wireless driver.

Experimental Methodology. We consider as performance metrics the *Packet Delivery Rate* (PDR) achieved by a CBR connection established between the two farthest mesh routers, namely 1 and 7 of the topology illustrated in Figure 6(a). The transmission rate was fixed to 50 kbit/s, enough to saturate the available bandwidth of the end-to-end path composed of 4 wireless links. The CBR traffic was generated using the *iperf* application.

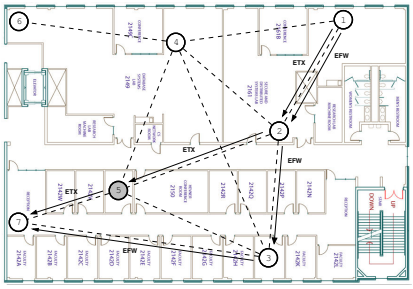
Attack Scenarios. Similarly to the simulation scenarios, we consider the *No Attack* and the *Data Dropping Attack*. In the

experiments we vary the percentage of traffic that an adversary node drops (i.e., its drop rate) from 0% to 80%.

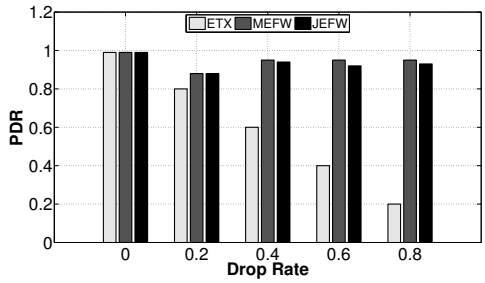
For each scenario we performed 10 independent measurements, as in the simulated evaluation. The total time of a CBR connection on which we evaluated the performance was equal to 600 seconds.

Results. Figure 6(b) illustrates the PDR measured in the network scenario described above as a function of the drop rate of node 5 (the only adversary node in the CS testbed). It can be observed that the proposed metrics (MEFW, JEFW) outperform the baseline metric (ETX) for any value of drop rate employed by the adversary node 5.

The results confirm the effectiveness of the proposed metrics to model the expected number of transmissions necessary to have the packet successfully forwarded. More specifically, as the drop rate increases, the ETX metric, unlike the proposed metrics, fails to model the actual reliability of the network links, which includes both the link quality and the relaying node selfishness. As the drop rate increases, the PDR obtained using the ETX metric keeps decreasing, whereas our proposed metrics permit to select the most reliable route among the alternative paths, and thus, improve significantly the network performance. In fact, as depicted in Figure 6(b), the performance gain obtained using our solutions ranges from 10% to 80%.



(a) Network Topology



(b) Packet Delivery Rate

Fig. 6: **CS Testbed**. Network topology and Average PDR as a function of the drop rate of node 5.

C. Performance Evaluation on ORBIT Testbed

We further performed experiments on the ORBIT testbed aim to evaluate the effectiveness and the scalability of the proposed solution.

Testbed Setup. The ORBIT testbed is an open access indoor radio grid testbed for controlled experimentation consisting of 400 wireless nodes equipped with IEEE 802.11a/g wireless cards laid out in a 20×20 grid with 1 meter spacing between nodes. Each node is connected via multiple high-speed Ethernet links that permit to remotely control the testbed and transfer applications or data.

Due to wireless card requirements and the high interference generated by the proximity of the wireless nodes, the network scenario employed in our experiments was composed of 40 nodes placed to form a grid topology 5×8, as illustrated in Figure 7.

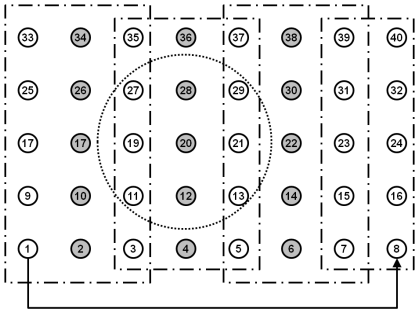


Fig. 7: **ORBIT Topology**. Network topology used for the experiments performed on the ORBIT testbed. The gray circles represent the nodes that can be selected to act selfishly.

Since all nodes of the ORBIT testbed are in the same radio range, we forced the grid topology both by using orthogonal channels and filtering rules. Specifically, we split the group composed of 40 devices in 4 subsets, each composed of at most 15 nodes using orthogonal channels (i.e., we split the entire grid into smaller 5×3 grids). The second interface of the nodes that belong to the first and last column of each subset was configured to ensure the complete connectivity of the network. The 4 subgroups of nodes obtained using orthogonal channels are identified by the 4 dashed boxes. The overlapped boxes identify the nodes that connect two adjacent groups using two radio interfaces, thus acting as bridges. We select as selfish nodes only the mesh routers with one active wireless interface in order to evaluate also the monitoring mechanism and thus have a complete picture of the effectiveness of the proposed solution. In Figure 7 the dotted circle represents the set of nodes whose routing messages are not filtered by a sample node (node 20) and that can establish a symmetric communication link with this node.

Experimental Methodology. Similarly to the grid scenario presented in Section VI, we measured the *Packet Delivery Rate* (PDR) achieved by 5 CBR connections established between the nodes on the two sides of the grid topology illustrated in Figure 7. The transmission rate and the packet size of each CBR connection were fixed to 50 kbit/s and 1470 bytes, respectively. The CBR traffic was generated using the traffic generator *iperf*.

We consider as attack scenarios the *No Attack* and the *Data Dropping Attack*, as illustrated in Section VII-B.

We evaluate the effectiveness of the proposed metrics against the data dropping attack varying both the number of selfish nodes and their drop rates. Specifically, we select randomly 4, 8, and 12 nodes (equivalent to 10%, 20% and 30% of the overall number of network nodes) placed in the central area of the grid to act selfishly.

Results. Figure 8 shows the average PDR measured as a function of the drop rate considering the attack and placement scenarios presented above. The results confirm the effectiveness of the proposed metrics to model the expected number of transmissions necessary to have the packet successfully forwarded.

In a real scenario, the performance degradation caused by a selfish node is more severe than in the simulated scenario, due to the lower network congestion. It can be observed that even a small fraction of adversary nodes with a relatively low drop rate can drastically reduce the end-to-end throughput. For example, when OLSR uses the ETX metric and 10% of nodes drop 20% of the traffic that they should forward, the PDR decreases by 24%. This performance is halved when the drop rate increases from 20% to 40%. Furthermore, as the number of adversary nodes increases, the impact on the PDR becomes even more evident. As Figure 8(c) illustrates, when 30% of network nodes are selfish, they can seriously affect the network performance and cause unfairness among data connections. In this case, the PDR quickly decreases to less than 10% of the performance obtained using our proposed metrics. On the contrary, the monitoring mechanism coupled with the proposed routing metrics select the most reliable network paths resulting in no evident performance degradation even considering severe attack scenarios.

VIII. Conclusion

Routing metrics proposed in recent years for wireless multi-hop networks fail to select the network paths with the highest delivery rate in the presence of intermediate nodes whose forwarding behavior is driven by selfish interests. To overcome this problem, we propose a cross-layer routing metric, EFW, and two alternative refinements, MEFW and JEFW, to select the most reliable path by considering both the quality of wireless links and the forwarding behavior of network nodes. We evaluate the effectiveness and the scalability of the proposed metrics through simulations and real testbed measurements performed in typical network scenarios. Our results show that the proposed solutions increase considerably both the network throughput and fairness with respect to the baseline approach that takes into account only the successful transmission rate of a wireless link.

We can therefore conclude that the proposed metric and its refinements represent an effective solution for achieving highly resilient routing and thus high delivery rates in WMCNs.

Acknowledgment

This work was funded by the Italian PRIN 2009 project GATECOM and by the European Commission through the FP7 project FLAVIA.

References

[1] S. Paris, C. Nita-Rotaru, F. Martignon, and A. Capone. EFW: A Cross-Layer Metric for Reliable Routing in Wireless Mesh Networks with Selfish Participants. *IEEE INFOCOM*, April 2011.

[2] N. Nandiraju, D. Nandiraju, L. Santhanam, B. He, J. Wang, and D.P. Agrawal. Wireless Mesh Networks: Current Challenges and Future Directions of Web-In-The-Sky. *IEEE Wireless Communications*, 14(4):79–89, 2007.

[3] D.S.J. De Couto, D. Aguayo, J. Bicket, and R. Morris. A High-Throughput Path Metric for Multi-Hop Wireless Routing. *Wireless Networks*, 11(4):419–434, 2005.

[4] S. Roy, D. Koutsonikolas, S. Das, and Y.C. Hu. High-Throughput Multicast Routing Metrics in Wireless Mesh Networks. *Ad Hoc Networks*, 6(6):878–899, 2008.

[5] I. Aad, J.-P. Hubaux, and E.W. Knightly. Impact of denial of service attacks on ad hoc networks. *IEEE/ACM Transactions on Networking*, 16(4):791–802, August 2008.

[6] I. Aad, J.P. Hubaux, and E.W. Knightly. Denial of Service Resilience in Ad hoc Networks. *ACM MobiCom*, pages 202–215, 2004.

[7] B. Awerbuch, R. Curtmola, D. Holmer, C. Nita-Rotaru, and H. Rubens. ODSBR: An On-Demand Secure Byzantine Resilient Routing Protocol for Wireless Ad Hoc Networks. *ACM Transactions on Information and System Security (TISSEC)*, 10(4):1–35, 2008.

[8] W. Galuba, P. Papadimitratos, M. Poturalski, K. Aberer, Z. Despotovic, and W. Kellerer. Castor: Scalable Secure Routing for Ad Hoc Networks. *IEEE INFOCOM*, pages 1–9, 2009.

[9] P. Papadimitratos and Z.J. Haas. Secure Message Transmission in Mobile Ad Hoc Networks. *Ad Hoc Networks*, 1(1):193–209, 2003.

[10] J. Eriksson, M. Faloutsos, S.V. Krishnamurthy, and C. MIT. Routing Amid Colluding Attackers. *IEEE ICNP*, pages 184–193, 2007.

[11] F. Oliviero and S.P. Romano. A Reputation-Based Metric for Secure Routing in Wireless Mesh Networks. *IEEE GLOBECOM*, 2008.

[12] T. Clausen and P. Jacquet. Optimized Link State Routing Protocol (OLSR) RFC 3626. *http://www.ietf.org/rfc/rfc3626.txt*, 2003.

[13] olsrd: Ad hoc wireless mesh routing daemon. *Available at URL: http://www.olsr.org/*.

[14] S. McCanne, S. Floyd, and K. Fall. Vint project U.C. Berkeley, ns-2 network simulator. *URL: http://www.isi.edu/nsnam/ns/*.

[15] B. Awerbuch, D. Holmer, H. Rubens, and R. Kleinberg. Provably Competitive Adaptive Routing. *IEEE INFOCOM*, pages 631–641, 2005.

[16] B. Carbunar, I. Ioannidis, and C. Nita-Rotaru. JANUS: A Framework for Scalable and Secure Routing in Hybrid Wireless Networks. *IEEE Transactions on Dependable and Secure Computing*, pages 295–308, 2008.

[17] W. Yu and K.J.R. Liu. Attack-Resistant Cooperation Stimulation in Autonomous Ad hoc Networks. *IEEE Journal on Selected Areas in Communications*, 23(12):2260–2271, 2005.

[18] X. Zhang, A. Jain, and A. Perrig. Packet-dropping Adversary Identification for Data Plane Security. *ACM CoNEXT*, pages 1–12, 2008.

[19] V. Srinivasan, P. Nuggehalli, C.F. Chiasserini, and R.R. Rao. Cooperation in Wireless Ad hoc Networks. *IEEE INFOCOM*, 2003.

[20] V. Srinivasan, P. Nuggehalli, C.F. Chiasserini, and R.R. Rao. An Analytical Approach to the Study of Cooperation in Wireless Ad hoc Networks. *IEEE Transactions on Wireless Communications*, 4(2):722–733, 2005.

[21] W. Yu and K.J.R. Liu. Game Theoretic Analysis of Cooperation Stimulation and Security in Autonomous Mobile Ad hoc Networks. *IEEE Transactions on Mobile Computing*, 6(5):507–521, 2007.

[22] S. Zhong, J. Chen, and Y.R. Yang. Sprite: A Simple, Cheat-Proof, Credit-Based System for Mobile Ad-Hoc Networks. *IEEE INFOCOM*, pages 1987–1997, 2003.

[23] L. Anderegg and S. Eidenbenz. Ad hoc-VCG: A Truthful and Cost-Efficient Routing Protocol for Mobile Ad hoc Networks with Selfish Agents. *ACM MobiCom*, pages 245–259, 2003.

[24] S. Eidenbenz, G. Resta, and P. Santi. The COMMIT Protocol for Truthful and Cost-Efficient Routing in Ad Hoc Networks with Selfish Nodes. *IEEE Transactions on Mobile Computing*, 7(1):19–33, 2008.

[25] J.J. Jaramillo and R. Srikant. DARWIN: Distributed and Adaptive Reputation Mechanism for Wireless Ad hoc Networks. *ACM MobiCom*, pages 87–98, 2007.

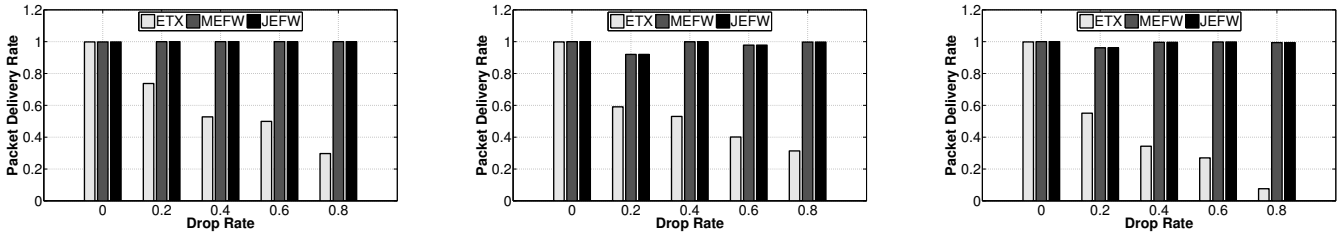
[26] Y. Wu, S. Tang, P. Xu, and X.Y. Li. Dealing With Selfishness and Moral Hazard in Non-Cooperative Wireless Networks. *IEEE Transactions on Mobile Computing*, 9(3):420–434, 2009.

[27] S. Zhong and F. Wu. On Designing Collusion-Resistant Routing Schemes for Non-Cooperative Wireless Ad hoc Networks. *ACM MobiCom*, pages 278–289, 2007.

[28] E. C. Efstathiou, P. A. Frangoudis, and G. C. Polyzos. Stimulating Participation in Wireless Community Networks. *IEEE INFOCOM*, pages 1–13, April 2006.

[29] L. Buttyan and J.P. Hubaux. Enforcing Service Availability in Mobile Ad Hoc WANs. *ACM MobiCom*, pages 87–96, 2000.

[30] D. Johnson and G. Hancke. Comparison of Two Routing Metrics in OLSR on a Grid based Mesh Network. *Ad Hoc Networks*, 7(2):374–387, 2009.



(a) PDR with 10% Selfish Nodes (b) PDR with 20% Selfish Nodes (c) PDR with 30% Selfish Nodes

Fig. 8: **ORBIT Testbed**. Average PDR measured in the grid scenario illustrated in Figure 7 as a function of the number of selfish nodes and the drop rate.

[31] G. Jakllari, S. Eidenbenz, N. Hengartner, S.V. Krishnamurthy, and M. Faloutsos. Link Positions Matter: A Noncommutative Routing Metric for Wireless Mesh Network. *IEEE INFOCOM*, pages 744–752, 2008.

[32] R. Draves, J. Padhye, and B. Zill. Routing in Multi-radio, Multi-hop Wireless Mesh Networks. *ACM MobiCom*, pages 114–128, 2004.

[33] S. Capkun, L. Buttyán, and J.P. Hubaux. Self-organized public-key management for mobile ad hoc networks. *IEEE Transactions on Mobile Computing*, 2(1):52–64, 2003.

[34] Y.C. Hu, A. Perrig, and D.B. Johnson. Ariadne: A secure on-demand routing protocol for ad hoc networks. *Wireless Networks*, 11(1-2):21–38, 2005.

[35] X. Ai, V. Srinivasan, and C.K. Tham. Wi-Sh: A Simple, Robust Credit Based Wi-Fi Community Network. *IEEE INFOCOM*, 2009.

[36] Y. Zhang and Y. Fang. ARSA: an attack-resilient security architecture for multihop wireless mesh networks. *IEEE Journal on Selected Areas in Communications*, 24(10):1916–1928, 2006.

[37] J. Eriksson, S.V. Krishnamurthy, and M. Faloutsos. Truelink: a Practical Countermeasure to the Wormhole Attack in Wireless Networks. *IEEE ICNP*, pages 75–84, 2006.

[38] M.E.M. Campista, P.M. Esposito, I.M. Moraes, L.H.M. Costa, O.C.M. Duarte, D.G. Passos, C.V.N. de Albuquerque, D.C.M. Saade, and M.G. Rubinstein. Routing Metrics and Protocols for Wireless Mesh Networks. *IEEE Network*, 22(1):6–12, 2008.

[39] K.H. Kim and K.G. Shin. On Accurate Measurement of Link Quality in Multi-hop Wireless Mesh Networks. *ACM MobiCom*, pages 38–49, 2006.

[40] S. Marti, T.J. Giuli, K. Lai, and M. Baker. Mitigating Routing Misbehavior in Mobile Ad Hoc Networks. *ACM MobiCom*, pages 255–265, 2000.

[41] E. A. Tobagi and L. Kleinrock. Packet Switching in Radio Channels: Part II – The Hidden Terminal Problem in Carrier Sense Multiple Access Modes and the Busy-tone Solution. *IEEE Transactions on Communications*, 23(12):1417–1433, 1975.

[42] J. Camp and E. Knightly. The IEEE 802.11s Extended Service Set Mesh Networking Standard. *IEEE Communications Magazine*, 46(8):120–126, 2008.

[43] Y. Yang and J. Wang. Design Guidelines for Routing Metrics in Multihop Wireless Networks. *IEEE INFOCOM*, pages 1615–1623, 2008.

[44] Q. Chen, F. Schmidt-Eisenlohr, D. Jiang, M. Torrent-Moreno, L. Delgrossi, and H. Hartenstein. Overhaul of IEEE 802.11 Modeling and Simulation in ns-2. *ACM MSWiM*, pages 159–168, 2007.

[45] D. Raychaudhuri, I. Seskar, M. Ott, S. Ganu, K. Ramachandran, H. Kremo, R. Siracusa, H. Liu, and M. Singh. Overview of the ORBIT Radio Grid Testbed for Evaluation of Next-Generation Wireless Network Protocols. *IEEE WCNC*, 2005.

Stefano Paris is an Assistant Professor at LIPADE (Laboratoire d'Informatique Paris Descartes), the department of Computer Science at Paris Descartes University. He received his M.S. degree in Computer Engineering from University of Bergamo in 2007, and the Ph.D. in Information Engineering from Politecnico di Milano in 2011. His main research interests include topics related to security and reliability in wireless networks. He is a member of the IEEE Communication Society.

Cristina Nita-Rotaru is an Associate Professor in the department of Computer Science at Purdue University. She leads the Dependable and Secure Distributed Systems Laboratory. She received BS and MS degrees from Politechnica University of Bucharest, Romania, in 1995 and 1996, and a PhD degree in Computer Science from The Johns Hopkins University in 2003. She served on the technical program committee of over 40 conference in networking, distributed systems, and security. She received the NSF CAREER award. Her research interests include security and fault-tolerance for distributed systems, and networks. She is a member of the ACM and IEEE Computer Society.

Fabio Martignon received the M.S. and the Ph.D. degrees in telecommunication engineering from the Politecnico di Milano in October 2001 and May 2005, respectively. He has been associate professor at University of Bergamo, and he is now full professor in the LRI (Laboratory for Computer Science) at Paris Sud University. His current research activities include multihop wireless networks, cognitive radio networks, network planning and game theory applications to networking problems.

Antonio Capone is Full Professor at the Information and Communication Technology Department (Dipartimento di Elettronica e Informazione) of Politecnico di Milano, where he is the director of the Advanced Network Technologies Laboratory (ANTLab). Dr. Capone is co-founder and CTO of MobiMESH (www.mobimesh.eu), a spin-off company of Politecnico di Milano. His expertise is on networking and his main research activities include protocol design (MAC and routing) and performance evaluation of wireless access and multi-hop networks, traffic management and quality of service issues in IP networks, and network planning and optimization. He received the M.S. and Ph.D. degrees in electrical engineering from the Politecnico di Milano in 1994 and 1998, respectively. He currently serves as editor of ACM/IEEE Trans. on Networking, Wireless Communications and Mobile Computing (Wiley), Computer Networks (Elsevier), and Computer Communications (Elsevier). He is a Senior Member of the IEEE.