COPS: Cooperative Statistical Misbehavior Mitigation in Network-Coding-aided Wireless Networks

Angelos Antonopoulos, *Senior Member, IEEE,* and Christos Verikoukis, *Senior Member, IEEE*

***Abstract*—The altruistic user behavior and the isolation of malicious users are fundamental requirements for the proper operation of any cooperative network. However, the widespread use of new communication techniques that improve the cooperative performance (e.g., network coding) hinders the application of traditional schemes on malicious users detection, which are mainly based on packet overhearing. In this paper, we introduce a cooperative nonparametric statistical framework, namely COPS, for the mitigation of user misbehavior in network coding scenarios. Given that the behavior of adversaries cannot be characterized by certain probability distributions, the proposed scheme exploits two well-known nonparametric statistical methods, i.e., Kruskal-Wallis analysis and Conover-Iman multiple comparisons, for the detection and identification, respectively, of malicious users in the network. It is worth noting that COPS framework does not require monitoring of the wireless channel and additional overhead, as its operation is based on the processing of the existing control packets. We assess the performance of the proposed scheme in various scenarios, showing that COPS is able to effectively handle attacks in the network, even when malicious users adopt a smart probabilistic misbehavior.**

***Index Terms*—Misbehavior detection, packet forwarding, cyber-physical systems, malicious, selfish, non-parametric statistics.**

I. Introduction

The introduction of 4G wireless communications has triggered the appearance of new technologies, such as Cyber-Physical Systems (CPS) and Device-to-Device communications (D2D), that allow the direct communication among devices and imply network densification. Moving towards 5G, the network densification is expected to be even higher with billions of devices (either sensors or smart terminals) deployed and used worldwide [1]. As a result of this network evolution and paradigm shift in communications, cooperative communication is now an intrinsic network mechanism and not merely an alternative way of transmission, while techniques that exploit and enhance the advantages of cooperation are moving strongly to the forefront. One of the techniques that have been widely developed during the last decade is network coding [2], [3], which enables the intermediate nodes in a cooperative network to mix different packets, thus offering reliable communication and improved network performance.

Regardless of the particular adopted technology and techniques, the basis for every functional cooperative network lies in the altruistic user behavior and the packet forwarding by the intermediary nodes [4]–[7]. However, this collaboration implies for the relays valuable energy and capacity resources that could be instead exploited for the transmissions of their own data. This problem is further aggravated in wireless networks, where the radio and energy resources are usually scarce, thus increasing the probability of a non-cooperative selfish attitude. In addition, the broadcast nature of the wireless medium makes wireless networks more vulnerable in security attacks by malicious adversaries.

A. Motivation and Related Work

In the aforementioned context, the protection of the network against selfish and malicious users has become of vital importance and the proposed solutions can be classified in two basic categories [8]: *proactive* and *reactive* security mechanisms. The works that fall in the former category [9]–[12] aim at constructing a network of trust, either by building reputation tables or providing the nodes with incentives to cooperate. On the other hand, the reactive mechanisms [13]–[20] operate in real-time, as the nodes overhear the neighbor transmissions to monitor the network activity or employ special acknowledgements (control packets) for the end-to-end communication.

The characteristics of the reactive mechanisms (i.e., simplicity, scalability, applicability) constitute them ideal candidates for security provision in current networks, which are highly unsettled and volatile. As a result, the scientific research community was urged towards the design of new security solutions that guarantee the proper network operation by enabling the online tracing of malicious users. In their pioneer work [13], Marti et al. proposed a basic detection technique, namely watchdog, that identifies the misbehaving nodes by enabling the wireless nodes to operate in promiscuous mode and monitor their neighbors in order to determine whether the latter have forwarded the data to the final destination. Another interesting reactive approach has been proposed by Liu et al. in [14], where

A. Antonopoulos and C. Verikoukis are with the Telecommunications Technological Centre of Catalonia (CTTC/CERCA), Castelldefels, Spain (e-mail: aantonopoulos@cttc.es; cveri@cttc.es).

This work has been supported by the research projects IoSense (692480), SEMIoTICS (780315), CellFive (TEC2014-60130-P), 5GSTEPFWD (722429) and CONNECT (737434).

the authors introduce an acknowledgement-based scheme (2ACK) to mitigate the impact of misbehaving nodes. The operation of 2ACK is based on special control packets that are employed to acknowledge the correct two-hop communication. In [15], a distributed sequential misbehavior detection technique based on Sequential Probability Ratio Test (SPRT) is developed, where the nodes overhear all control packets to make statistical decisions. However, these works, as well as subsequent similar efforts that were based on the same ideas [16]–[20], suffer evident disadvantages, such as increased power consumption and additional packet overhead.

The introduction of network coding is an additional deterrent for the application of the above mentioned techniques in current large-scale cooperative networks [21], as the network coded packets are no longer tractable by conventional or artificial intelligence-based schemes [22]. Moreover, the simple packet acknowledgements have been replaced by cumulative reports [23] that contain information about the senders of the received data at a given node. To overcome these limitations, Kim et al. [24] introduced algebraic watchdog, a network coding oriented monitoring scheme that allows the detection of routing misbehavior in a probabilistic manner by overhearing the neighboring transmissions and exploiting the algebraic properties of network coding. Despite its novel insights in network coding scenarios, algebraic watchdog faces common problems of monitoring techniques, as its operation is exclusively based on the promiscuous packet overhearing, while it requires nodes of advanced functionality, which are capable of processing complex computational algebraic functions.

B. Contribution

In this paper, motivated by the impact of malicious users in cooperative communications, we introduce a novel reactive statistical technique for mitigating misbehavior issues in packet forwarding in mobile networks. Given that the behavior of adversary users cannot be characterized by certain probability distributions, the proposed method exploits well-known nonparametric statistical approaches to detect and identify the malicious users in the network. Unlike most existing approaches (i.e., [13]–[20]), our solution is compatible with network coding scenarios, while it does not imply any additional computational or power overhead for the nodes (unlike [24]), since its operation relies on the analysis of existing control packets. Our contribution can be summarized in the following points:

1) We introduce an integrated **co**operative non**p**arametric **s**tatistical framework, namely COPS, for the mitigation of user misbehavior in network coding scenarios. The proposed method analyzes the existing control packets in the network without requiring any additional overhead, any *a priori* knowledge for the type of the malicious users, nor any monitoring of the wireless channel. More specifically, COPS consists of two interactive modules: i) a detection method, based on the Kruskal-Wallis statistical test [25], which examines whether all the control packets belong to a single population (i.e., honest users) and ii) a post hoc identification technique, based on the Conover-Iman multiple comparison [26], which is triggered by the detection of abnormal activity in the network and identifies the malicious users in the network.
2) We assess the performance of the proposed scheme in various scenarios and use cases, showing that COPS is able to effectively confront network attacks, even when malicious users adopt a smart probabilistic misbehavior.

The rest of the paper is organized as follows. The problem statement is introduced in Section II. The system model and the impact of malicious activity are presented in Section III. The proposed COPS framework along with the detailed description of its two basic modules is introduced in Section IV. The performance evaluation is provided in Section V, while Section VII concludes this paper.

II. Problem Statement

In this section, we intend to clarify the problem under study and shed some light on the employed concepts. The origin of the problem lies in the lack of explicit acknowledgements in network coding scenarios, since, in such scenarios, the nodes receive linear combinations of the original packets through different relays. Hence, the simple packet acknowledgements have been substituted by bulk acknowledgements that verify the reception of many packets, while they include the portion of information that the node has received by each relay.

For instance, let us consider the case where the destination node is located in the range of two relay nodes that transmit network coded packets. The typical situation in this particular example (e.g., similar channel conditions and fair medium access control) is that, at the end of the communication, each relay should have transmitted approximately half (i.e., 50%) of the total packets. Hence, the destination can issue a bulk acknowledge that declares the portion of information that has received by each relay and, consequently, this control packet can be used as a report to characterize the relay's behavior. However, the employment of these new acknowledgements raises an important issue, as the reliability of these reports is questionable. In particular, two types of fake reporting have been identified [11]: *i) under-reporting*, the acknowledged portion of information is lower than the actually received, and *ii) over-reporting*, where the acknowledged portion of information is higher than the actually received.

Fig. 1 provides an example of a cooperative network with one source node (S) that disseminates information at four destination (D) nodes through five relay (R) nodes, while the destinations periodically issue reports to characterize the corresponding relay nodes. In such scenarios, different types of misbehavior can be observed, as also discussed above. More specifically, the relays can

be either *cooperative/good* or *selfish/bad* (depending on whether they participate in the packet relaying or not), while the destination nodes can be *honest* or *malicious*. Honest nodes (depicted in green color) issue always *truthful* reports, while malicious nodes (depicted in orange color) may issue *fake* reports about the relay's behavior and can be categorized as types U and O, depending on whether they under- or over-report the received packets, respectively. For example, as D_2 (type U) receives packets by both R_1 and R_2, it may under-report R_1 to damage the network. As a result, D_2 will still receive the total information by R_2, but the false characterization of R_1 as selfish will significantly affect node D_1. Regarding the malicious users of type O, we see that node D_3 might be indifferent for the selfish behavior of some relays, as it receives packets through multiple links. Consequently, by over-reporting R_5 (either to cause network damage or because they cooperate), node D_4 will be excluded from the network.

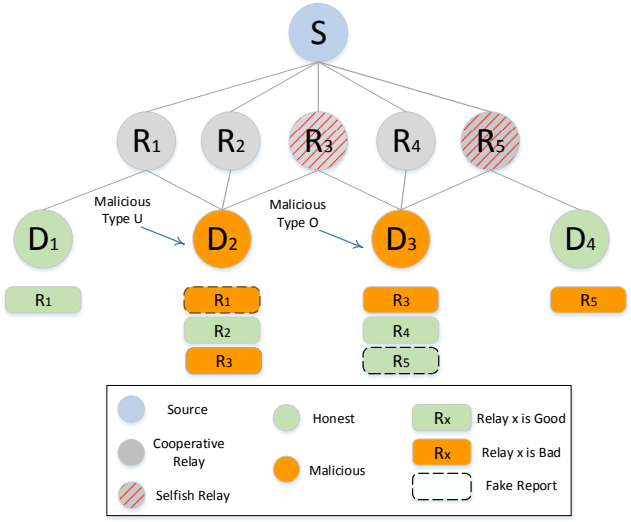


Fig. 1: Network Example

The aforementioned example highlighted the different types of users that can be encountered in a wireless network and provided some indications about the potential malicious activities. Apparently, in real large-scale networks with a plethora of devices the problems are further escalated. Therefore, given the shortcomings of the traditional misbehavior mitigation approaches and the existence of coded information, the proposal of new effective methods for the identification of malicious users in the network is now compulsory. To that end, the filtering of the transmitted reports seems an appealing solution, as the identification of the fake reports would assist towards the identification of the malicious and selfish nodes in the network.

III. System Model and Impact of Malicious Activity

In this section, we describe the system model and we study the impact of malicious users on the application of security mechanisms in the network. The notation employed throughout the paper is summarized in Table I.

TABLE I: Notation

Symbol	Definition
$\mathcal{C}$	Set of sink nodes
U_h	Set of honest users
M_U	Set of under-reporting users
M_O	Set of over-reporting users
c_m	Report of node m
T_r	Time interval between reports
$B(t)$	Behavior of source node in time t
p_e	False report probability due to channel error
p_f	False report probability due to misbehavior
$d^{CE}(t)$	Central decision for source node
δ_m	XOR between report c_m and central decision $d^{CE}(t)$
L	Number of feedback periods
D_m	Expected value of δ_m over L
f_x	Probability mass function (pmf) of x
H_0/H_1	Null/Alternative Hypothesis
Q	Number of D_m samples
r_m	Rank of sample
R_m	Total sum of individual ranks
N	Total number of samples

A. System Model

The architecture of a real network can be decomposed into similar parts that follow the same pattern. Hence, without loss of generality, we assume a network (Fig. 2), where node A transmits network coded information to a set of M nodes $\mathcal{C} = \{C_m : 0 < m \leq M\}$. In periodic time intervals, the receivers issue a report $c_m(t) \in \{0, 1\}$ that characterizes the behavior of node A, i.e., $c_m(t) = 1$ and $c_m(t) = 0$ denote that A is cooperative or selfish, respectively. Therefore, taking into account that the time between two consecutive reports is T_r, the report generation constitute a discrete random process, denoted by $\{c_m(t) : t = t_0 + nT_r, n \in \mathbb{Z}\}$, where t_0 is the generation moment of the first report.

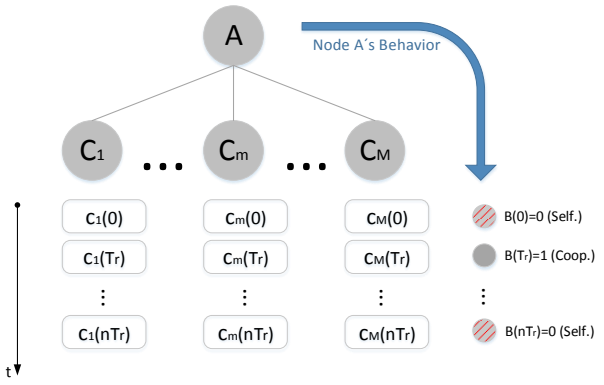


Fig. 2: System Model

Regarding the behavior of node A, we introduce the binary variable $B(t) \in \{0, 1\}$, which is either 1 or 0 in case that node A is cooperative or selfish, respectively. Assuming the existence of only honest users, under ideal

channel conditions, it should hold that $c_m(t) = B(t), \forall t$. However, false reports could be also generated due to unpredictable factors (e.g., channel conditions) or due to the existence of malicious users. In our work, potential channel errors are considered through a probability of false reports (p_e), while the probability of deliberate fake reports (p_f) is also taken into account.

We assume that all generated reports at time t are collected by a central entity/controller that is able to monitor the network (e.g., network administrator). According to the majority of these reports, which is a common case of the "k-out-of-n" decision rule [27], the central entity makes a decision $d^{CE}(t)$ about node A. This can be mathematically expressed as

$$d^{CE}(t) = \begin{cases} 1 & \text{, if } \sum_{m=1}^{M} c_m(t) \geq M/2 \\ 0 & \text{, otherwise} \end{cases}. \quad (1)$$

B. Impact of Malicious Activity

As there is no indication for the behavior $B(t)$ of a given node, the decision $d^{CE}(t)$ is based exclusively on the received reports. Therefore, the reception and analysis of fake reports could alter the correct decision, causing serious malfunctions in the network. In order to capture the potential discrepancies between the individual reports and the central decision, let us define as $\delta_m(t) = c_m(t) \bigoplus d^{CE}(t)$, $\delta_m(t) \in \{0, 1\}$ the binary variable that compares a particular report with the final decision. Apparently, $\delta_m(t) = 0$ when the report is identical with the decision, while $\delta_m(t) = 1$ when the report is different from the decision. However, the characterization of the nodes cannot be based only on the instantaneous reports, mainly due to the dynamic nature of the wireless medium, as it is possible that some transmissions are not received owing to channel errors (in this case the destination may falsely characterize the corresponding relay as non-cooperative). As a result, to increase the robustness of the decision, it is essential to study the discrepancies of the reports over time. To that end, instead of employing a specific $\delta_m(t)$ value, we focus on the expected value of δ_m in L consecutive feedback periods, defined as

$$D_m = \mathbb{E}[\delta_m] = \frac{1}{L} \sum_{n=0}^{L-1} \delta_m(t - n \cdot T_r). \quad (2)$$

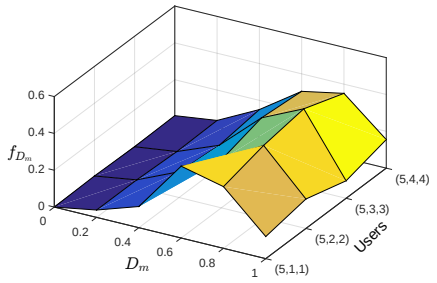
In order to study and identify the different behaviors in the network, we need to focus on the properties of D_m. Since δ_m is a binary variable, D_m can take $L+1$ discrete values $d_i = \frac{i}{L}$, where $i \in \mathbb{Z}$ and $i \in [0, L]$. Therefore, the range of D_m can be defined as $\mathbb{D} = \{d_0, d_1, ..., d_L\}$, with probability mass function (pmf)

$$f_{D_m}(d_i) = Pr(D_m = d_i), d_i \in \mathbb{D} \quad (3)$$

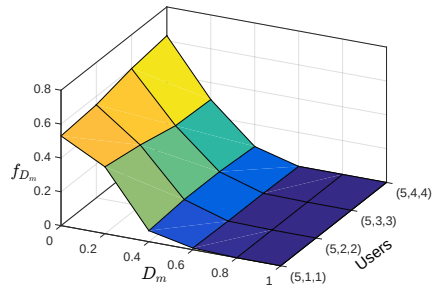
and

$$f_{D_m}(d_i) = 0, d_i \notin \mathbb{D}. \quad (4)$$

The graphical illustration of f_{D_m} is essential for the comprehensive demonstration of the user discrepancies. To that end, Fig. 3 provides an example of f_{D_m} in case of a malicious user of type U (Fig. 3(a)) and type O (Fig. 3(b)), respectively. In this specific example, we assume that the relay node is not cooperative (i.e., $B(t) = 0$) and does not forward the incoming packets. In addition, we assume a total number of 1000 reports, while the number of consecutive feedback periods for our study is $L = 5$. The probability that a malicious user transmits a fake report is $p_f = 0.7$ and the probability of wrong reports due to channel errors or other imponderable factors is $p_e = 0.1$. Finally, we consider four different combinations of $\{|U_h|, |M_U|, |M_O|\}$ users in the network, that is {5,1,1}, {5,2,2}, {5,3,3}, and {5,4,4}.



(a) Type U



(b) Type O

Fig. 3: Probability mass function (f_{D_m}) for different types of malicious users

The plots presented in Fig. 3 highlight the important differences that may be observed in the reports of different users and, therefore, indicate that the comparison of the f_{D_m} would be an effective way for malicious user detection. More specifically, the central entity has access to M D_m random variables, generated by one or more types of users, where each type is characterized by a different behavior and probability of communicating fake reports. In the ideal case, where the random variables would be normally distributed, the application of parametric statistical methods (e.g., analysis of variance or ANOVA) would efficiently identify the different types of users in the network. However, in our problem, the normality assumption is too strict (as it can be seen in Fig. 3), something that complicates the malicious user detection in the network by employing classical parametric statistical methods and stresses the need for non-parametric approaches.

IV. Cooperative Nonparametric Statistical Detection and Identification of Malicious Users

In this section, we introduce a nonparametric statistical scheme, namely COPS, for the mitigation of malicious activity in network-coding-enabled wireless networks. The proposed scheme, which is based on the transmitted control packets, is executed in two phases, as it is also depicted in Fig. 4:

i) In the detection phase (Phase I), the central entity collects the reports for a given node and performs a statistical analysis based on the Kruskal-Wallis nonparametric method.
ii) In the identification phase (Phase II), which is executed only when malicious users have been detected during the first phase, the central entity conducts multiple comparisons to the sample reports based on the Conover-Iman posthoc method[1].

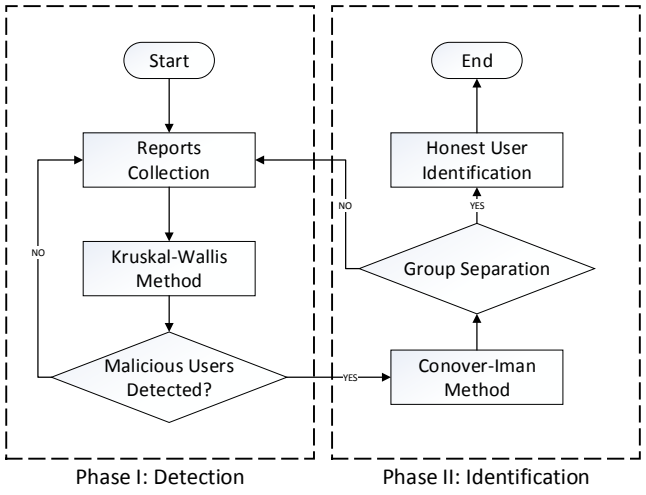


Fig. 4: Flowchart of COPS framework

In the following sections, we analytically present the two algorithms and the statistical analysis that takes place in each module.

A. Kruskal-Wallis Detection

As we have seen in Sec. III-B, in our case, the trend and the behavior of the users cannot be modeled according to well-known statistical distributions. For this reason, we are motivated to adopt the Kruskal-Wallis nonparametric statistical method to detect the existence of malicious users in the network. The Kruskal-Wallis test is based on the ranking of the samples and is able to detect whether different sets of samples belong to the same population, i.e., they have been generated by the same probability distribution.

The analysis of the median is the main characteristic of the nonparametric statistical methods, as there is no normality assumption in the distribution of the samples. In particular, median is defined as the numeric value that separates the higher from the lower half of a sample set, a population, or a probability distribution, thus being more robust to vulnerabilities (e.g., outliers) compared to the mean value. This robustness is the key point that makes the analysis of the median and the application of nonparametric statistics appropriate for our proposed scenario. Defining as η_m the median of D_m, the null (H_0) and the alternative (H_1) hypothesis of the statistical Kruskal-Wallis test can be expressed as

$$\begin{cases} H_0 : \eta_1 = \eta_2 = \ldots = \eta_M \\ H_1 : \eta_{m_i} \neq \eta_{m_j} \text{ for at least one } m_i \neq m_i. \end{cases} \tag{5}$$

In this context, the acceptance of the null hypothesis implies that the differences between the medians of the different samples are not significant and, hence, all M random variables belong to the same population. On the other hand, the rejection of the null hypothesis suggests that the M random variables have been generated by different populations, something that verifies the existence of malicious users in the network.

The maintenance of a database with these samples is fundamental for the application of the proposed method. More specifically, at a given time instant t, the central controller makes a decision about node A according to the majority of the collected reports by the M nodes. Subsequently, this decision is compared with the individual reports to generate M δ_m binary variables and, after L received reports by each node, one single sample for the variable D_m is calculated. However, the application of statistical methods requires several samples, hereafter denoted by Q, while the nonparametric statistics are based on the ranks of the samples, rather than their actual values. Hence, given a set of samples $\mathcal{S} = \{D_m(t - qT_D) : 1 \leq m \leq M, 0 \leq q \leq Q-1\}$, the rank of each sample $D_m(t - qT_D)$ is denoted by $r_m(t - qT_D)$ and is equal to the number of observations in the set $\mathcal{S}$ that are smaller or equal to $D_m(t - qT_D)$. Accordingly, the test statistic can be written as [25]

$$H = \frac{12}{N(N+1)} \sum_{m=1}^{M} \frac{R_m^2}{Q} - 3(N+1), \tag{6}$$

where N is the total number of samples (i.e., $N = MQ$) and R_m is the total sum of the individual ranks for node m, equal to

$$R_m = \sum_{q=0}^{Q} r_m(t - qT_D). \tag{7}$$

Moreover, since, in practice, the behavior of H can be approximated by the chi-square distribution with $M-1$ degrees of freedom, the null hypothesis is rejected if

$$H \geq \chi^2_{\alpha, M-1}, \tag{8}$$

where α corresponds to the significance level. An indicative table with the values of $\chi^2_{\alpha,M-1}$ for different degrees of

[1]For a detailed explanation of nonparametric statistical methods, we refer the interested reader to [28].

freedom and significance levels is provided in [29, Table 1].

It is worth noting that sometimes the observations are tied (i.e., they have the same value). In our particular case, the Boolean nature of the reports leads to quantized average values and, consequently, strong ties among the samples. As a result, the test statistic should be replaced by

$$H = \frac{1}{\sigma^2}\left[\sum_{m=1}^{M}\frac{R_m^2}{Q} - \frac{N(N+1)^2}{4}\right], \tag{9}$$

where σ^2 denotes the variance of the ranks, calculated as

$$\sigma^2 = \frac{1}{N-1}\left[\sum_{q=}^{Q-1}\sum_{m=1}^{M} r_m(t-qT_D)^2 - \frac{N(N+1)^2}{4}\right]. \tag{10}$$

In case that the number of ties is moderate, the difference between the two tests becomes negligible. However, the complex type of the test is more complete and can be used in any case. The algorithmic process of the Kruskal-Wallis detection is summarized in Algorithm 1.

Algorithm 1 Kruskal-Wallis Detection

Input: Q, L, chi-squared distribution, α
Output: Malicious user detection

repeat ▷ Repeat the process until the collection of Q samples
  for i=1 to L **do**
    for all m **do**
      Collect report c_m ▷ Collect reports from all nodes about the behavior of the source
      Make decision d^{CE} ▷ Make decision for the source based on the majority of the reports
      Compute δ_m ▷ Compare the decision with the individual reports
    end for
    Calculate D_m ▷ Calculate the expected value of δ_m from L samples
  end for
until Samples=Q
Rank Table $D[m][Q]$ ▷ Substitute the D_m values with their respective ranks
for all m **do**
  Compute R_m ▷ Estimate the sum of the ranks for every node
end for
Calculate H ▷ Calculate H according to Eq. (9)
if $H \geq \chi^2_{\alpha,M-1}$ **then**
  KWD=1 ▷ Malicious users are detected in the network
else
  KWD=0 ▷ No malicious users in the network
end if

B. Conover-Iman Identification

The rejection of the null hypothesis by the Kruskal-Wallis method indicates the existence of malicious users in the network, but it is not sufficient for their recognition. More specifically, a post hoc or *a posteriori* analysis is required to identify the malicious users in the network. In our case, given that the normality assumption does not hold, the most appropriate method to distinguish the different populations is the Conover-Iman test, which is based on multiple comparisons between pairs of samples. In COPS, all users are sorted in an ascending order according to their sum rank (Eq. (7)) and, for every two consecutive users, it is tested whether

$$\frac{|R_{m_i} - R_{m_j}|}{\lambda} > t_{N-M,1-\alpha/2}, \tag{11}$$

where

$$\lambda = \sqrt{2Q\sigma^2\frac{N-1-H}{N-M}} \tag{12}$$

and $t_{N-M,1-\alpha/2}$ is a quantile from the Student's t-distribution on $N-M$ degrees of freedom and a level of significance $1-\alpha/2$.

If Eq. (11) holds, the two users are considered to belong to different populations and, every time a new discrepancy is identified, a new set is created as the users are already ordered. The algorithmic process of the Conover-Iman identification is summarized in Algorithm 2.

Algorithm 2 Conover-Iman Identification

Input: Q, N, M, $List[R_m]$, Student-t distribution, α
Output: Malicious user identification

Sort $List[R_m]$ in ascending order ▷ The list with the sums of the ranks
for i=1 to M **do** ▷ Start the pairwise comparison
  if $\frac{|R_{m_i} - R_{m_{i+1}}|}{\lambda} > t_{N-M,1-\alpha/2}$ **then**
    Different Set ▷ The two nodes belong to different populations
  else
    Same Set ▷ The two nodes belong to the same population
  end if
end for

C. Computational Complexity

The proposed algorithm includes two phases: i) the Kruskal-Wallis detection and ii) the Conover-Iman identification. Regarding the first phase, defining as n the total number of collected samples/reports, the executed functions and the corresponding complexities are as follows:

- Scan the table to make the decision $\rightarrow \mathcal{O}(n)$
- Compare the decision with each report $\rightarrow \mathcal{O}(n)$
- Sort the table to rank each sample $\rightarrow \mathcal{O}(n\log n)$.

On the other hand, in case of detection during the first phase of the algorithm, there are two distinct functions on the second phase:

- Sort the table in ascending order $\rightarrow \mathcal{O}(n\log n)$
- Pairwise comparisons in the table $\rightarrow \mathcal{O}(n)$.

Consequently, as the complexity is determined by the most important term, we may claim that the complexity of the proposed algorithm is $\mathcal{O}(n \log n)$.

V. Performance Assessment

We have developed an event-driven C++ simulator that executes the rules of the proposed COPS framework for malicious user detection and identification in network-coding-based wireless networks. Extensive Monte-Carlo simulations have been carried out to evaluate the effectiveness of the proposed mechanism in different scenarios. In this section, we present the simulation setup along with the experimental results.

A. Simulation Scenario

We consider a scenario as illustrated in Fig. 2, where node A transmits network coded packets to a set of M nodes. In periodic time intervals, each receiver generates a report c_m that characterizes the behavior of node A (i.e., either selfish or cooperative). According to the majority of these reports, a central entity i) makes a decision about node's A behavior, ii) creates the database (the table $D[m][Q]$ in Algorithm 1) and, ii) computes the value of D_m every L samples. Upon the collection of Q D_m samples, the central entity applies the Kruskal-Wallis method to detect any malicious activity in the network. In case that malicious users have been detected in the network, the second module of the scheme is triggered and pairwise multiple comparisons, based on the Conover-Iman procedure, are conducted in the set of the collected samples to identify the different populations. Regarding the reports c_m, we assume that honest users send false reports with a probability p_e due to channel errors, while malicious users provide deliberately false reports with a probability p_f. The simulation parameters are summarized in Table II.

TABLE II: Simulation Parameters

Parameter	Value	Parameter	Value
M	[4-12]	p_e	0.1
L	[0-30]	p_f	[0.15-0.4]
Q	[0-30]	α	[0.01-0.2]

B. Simulation Results

Fig. 5 presents the detection probability vs. L and Q, which represent the interval between the generations of the D_m and the total number of D_m, respectively. In this particular scenario, we assume that there are four users in the range of node A ($M = 4$), where three of them are honest and one is malicious, providing fake reports for the behavior of A with a probability $p_f = 0.25$. In addition, the significance level $\alpha = 0.05$, which is a relatively tight value. The plot in Fig. 5 clearly demonstrates that the probability of malicious user detection depends on the product of L and Q, rather than on their individual values.

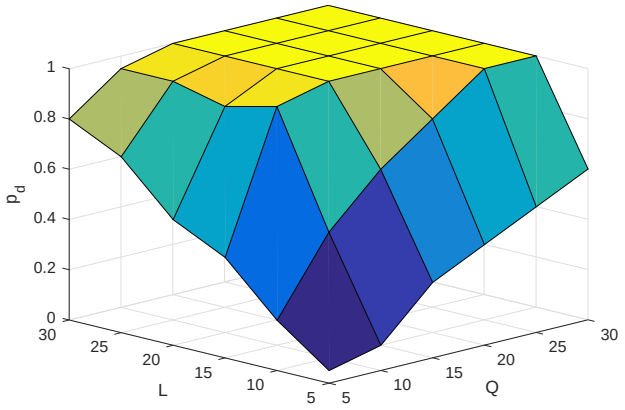


Fig. 5: Detection Probability ($|U_h| = 3, |M_U| = 1, p_f = 0.25$)

In particular, when $LQ > 200$ the malicious user is always detected with probability equal to 1.

Fig. 6 presents again the detection probability for a similar scenario. However, in this case, we assume that the malicious user is more aggressive, providing deliberately fake reports with a higher probability, i.e., $p_f = 0.4$. As we can observe, the probability of detecting the malicious user in the network converges much faster to 1, indicating that a lower number of samples is required for the detection of a malicious user in the network.

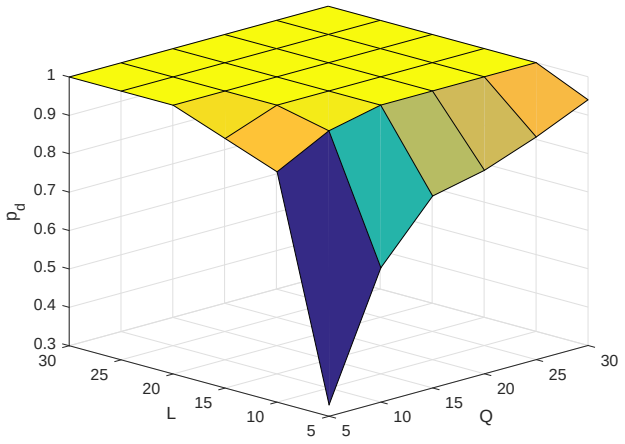
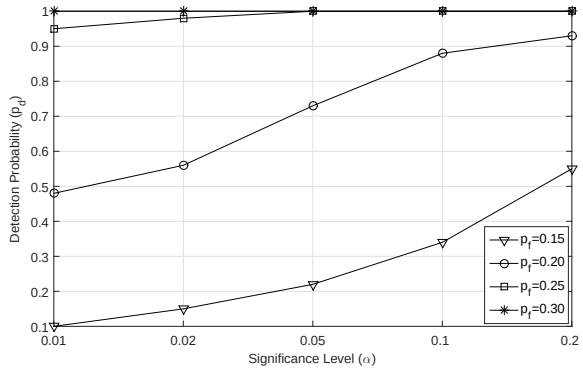
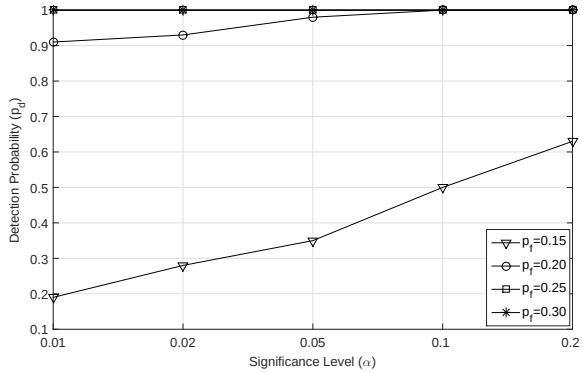


Fig. 6: Detection Probability ($|U_h| = 3, |M_U| = 1, p_f = 0.4$)

The significance level α is a crucial parameter for the execution of the Kruskal-Wallis method, since it characterizes the sensitivity of the proposed algorithm to the differences among the samples. To that end, Fig. 7 presents the detection probability vs. different significance levels in two scenarios with seven honest users (i.e., $|U_h| = 7$), different number of malicious users in the network and various cases of malicious user activity (i.e., $p_f = 0.15$, $p_f = 0.20$, $p_f = 0.25$ and $p_f = 0.30$). In the first case



(a) $|M_U| = 2$



(b) $|M_U| = 5$

Fig. 7: Detection Probability vs. Significance Level α ($|U_h| = 7, Q = 15, L = 10$)

(Fig. 7(a)) with two malicious users, we observe that low values of significance level imply low detection probability, especially when the malicious users are not particularly aggressive (i.e., low p_f). However, the malicious users can be detected with higher probability, as they become more aggressive (i.e., higher p_f). More specifically, the proposed algorithm is able to detect any malicious activity with probability equal to 1 when the malicious users adopt a $p_f = 0.3$, which practically means that malicious activity is detected even if (nearly) only one out of three reports is fake. Regarding the case of five malicious users (Fig. 7(b)), we can see that the detection probability is much higher, even when the malicious users are not so aggressive (i.e., $p_f = 0.20$). As a result, the number of malicious users in the network has a direct influence in their detection, a fact that can be also conceived intuitively.

In Fig. 8, we study how the detection probability is affected by the number of malicious users. More specifically, we assume a network with three honest users (i.e., $|U_h| = 3$) and different number (i.e., one to nine) of malicious users. In this plot, we assume that $\alpha = 0.05$, while various cases for the misbehavior probability p_f are considered. Again, the L and Q values have been selected equal to 10 and 15, respectively. Apparently, the number of malicious users does not significantly affect the detection probability when p_f is relatively low (e.g., $p_f = 0.15$), since, in this case, the probability of deliberately transmitting false reports is very close to the probability of transmitting false reports due to channel errors (i.e., $p_e = 0.10$), thus hindering the distinction between malicious and honest users and honest users. On the other hand, as the probability of misbehavior grows, the detection probability significantly increases with the probability of misbehavior, rendering the detection even of only one malicious user possible, in case of $p_f = 0.3$.

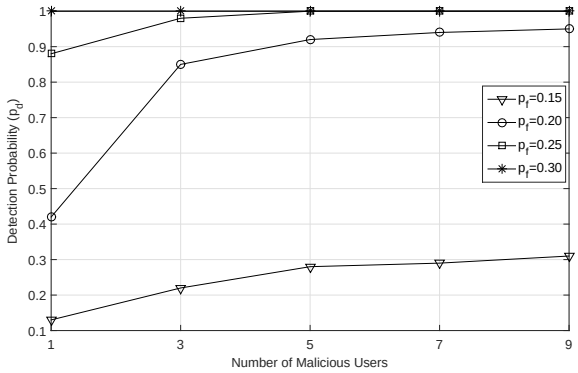


Fig. 8: Detection Probability vs Number of Malicious Users ($\alpha = 0.05, Q = 15, L = 10$)

Apparently, by adopting a high significance level (e.g., $\alpha = 0.2$), the central entity would be always able to detect the malicious users in the network. However, high values of α cause an increased probability of false detection, i.e., high probability of characterizing honest users as malicious. To that end, we study a scenario with ten honest users (i.e., $|U_h| = 10$) and no malicious user in the network. Fig. 9 presents the false detection probability for different values of significance level and Q samples. As we can see, the detection probability increases up to $p_d = 0.18$ for $\alpha = 0.2$ and $Q = 5$. As Q increases, the probability of falsely detecting honest users as malicious decreases, but it does not annihilate for high significance levels. The most important conclusion from this figure is that the significance level and the number of samples are strongly related, as both affect the false detection probability. More specifically, the significance level should be selected according to the available number of samples. For small numbers of Q, the significance level should be quite low in order to guarantee zero false detection probability, while a big number of samples enables the selection of a more flexible significance level (e.g., for $Q = 20$, the false detection probability is almost zero, even if $\alpha = 0.05$).

In Fig. 10 we evaluate the Conover-Iman identification in scenarios where the Kruskal-Wallis method detects the malicious users in the network with probability equal to one. More specifically, we assume eight honest users (i.e., $|U_h| = 8$) in the network with $p_e = 0$ and we study two different cases with one and four malicious users. As we may see, in case that the malicious users adopt an relatively aggressive strategy with $p_f = 0.3$, COPS

TABLE III: Comparison with existing approaches

Scheme	Node overhead (Computational)	Control overhead (extra ACKs)	Power overhead (Monitoring)	Network Coding support
Watchdog [13]	✗	✗	✓	✗
2ACK [14]	✗	✓	✗	✗
SPRT-based [15]	✗	✗	✗	✗
SCAN [16]	✗	✗	✓	✗
EEACK [17]	✓	✓	✗	✗
CWS [18]	✗	✗	✓	✗
AMD [19]	✗	✗	✓	✗
LDK [20]	✓	✗	✓	✗
Algebraic Watchdog [24]	✓	✗	✓	✓
COPS (our work)	✗	✗	✗	✓

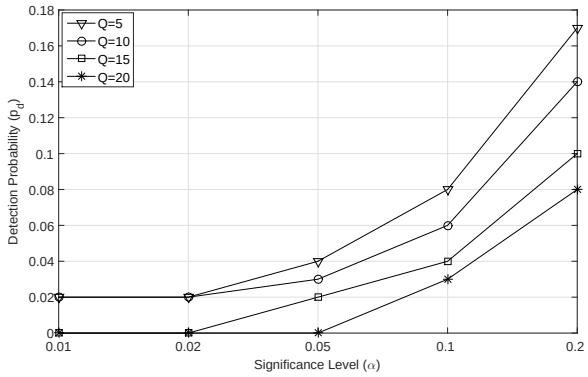


Fig. 9: False detection Probability vs Significance Level α ($|U_h| = 10, L = 10$)

is able to identify all of them with high confidence (i.e., $\alpha = 0.05$). On the other hand, when the malicious users have a mild behavior with one out of five fake reports, their identification is much more complicated in low significance levels, especially as their number increases, since their discrepancies with the honest users become negligible. Nonetheless, in all cases, COPS is able to identify the malicious users using a high significance level (i.e., $\alpha = 0.2$), although a higher misdetection risk would be implied. This tradeoff is determined by the desired security level of the network and its study would be of great interest.

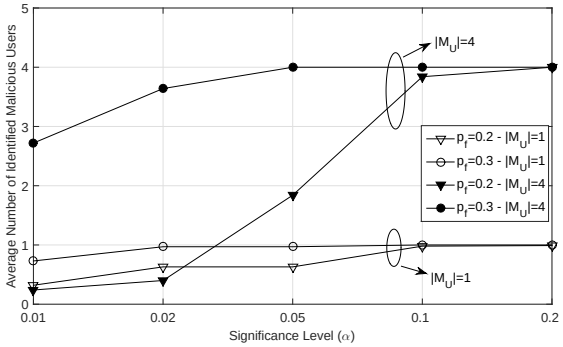


Fig. 10: Average Number of Identified Malicious Users

Finally, in Fig. 11, we compare the network energy efficiency in case that COPS and Algebraic Watchdog, respectively, is applied in the network. More specifically, we assume the existence of monitoring nodes to study a scenario where both schemes are able to successfully detect the node misbehavior. In Algebraic Watchdog, the monitoring nodes overhear the transmissions of the source, while in COPS the monitoring nodes could be in idle or even in sleep mode. The power levels for the various modes are in line with [30]. Apparently, as the number of monitoring nodes increases, the network energy efficiency of the Algebraic Watchdog decreases, as more nodes have to overhear the transmissions. On the other hand, the network energy efficiency is up to 56% higher in COPS when the monitoring nodes are in idle mode and the improvement can reach 78% if the nodes switch to sleep mode. It is also worth noting that, when the monitoring nodes are able to be in sleep mode, the network energy efficiency remains almost constant, something very important for future applications, where the network lifetime will be of crucial importance.

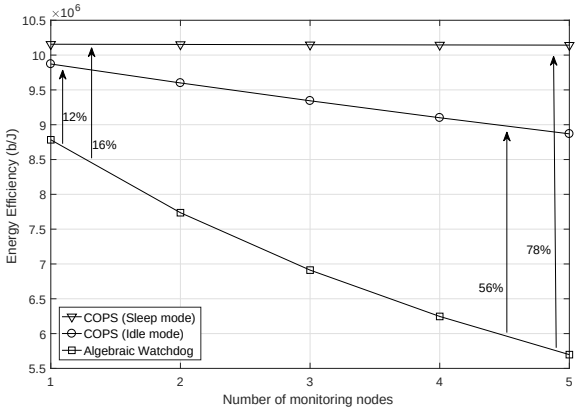


Fig. 11: Energy Efficiency ($|U_h| = 2, |M_O| = 1, p_f = 0.3$)

In a nutshell, we have provided a thorough and extensive evaluation of the proposed COPS scheme in various scenarios, aiming at assessing the impact of different parameters (i.e., L, Q, U_h, M_U/M_O, p_f, p_e and α) on its performance. The most important conclusions can be summarized as follows:

- The detection probability is highly dependent on the product LQ, which should be optimized in order to

guarantee an efficient detection of malicious users, keeping the total time for their detection in low levels.
- The number of malicious users (M_U or M_O) in the network, the probability of misbehavior (p_f) and the significance level (α) are very critical factors for the detection of malicious activity in the network, as detection probability increases when either of these factors increases.
- The particular values of deliberate malicious activity (p_f) and false transmissions due to channel conditions (p_e) determine the detection probability, since malicious users are more easily tracked when these values are different. However, the proposed scheme has been shown to operate efficiently even when the aggressiveness of the malicious users is not high (i.e., $p_f = 0.3$) and close enough with the false estimation due to channel errors (i.e., $p_e = 0.1$).
- The appropriate selection of α is of fundamental importance, as high values of α cause high false detection probability (i.e., honest users can be characterized as malicious), affecting the smooth network operation.
- The proposed algorithm effectively detects and identifies the malicious users in the network regardless of their percentage in the network (e.g., even when M_U exceeds U_h), as its operation is based on the detection of populations with different behavior in the network.

VI. Challenges and Open Issues

In this section, we present some challenges and open issues related to our proposed scheme. However, before this discussion, we would also like to provide a clear view on the benefits that COPS offer compared to the existing schemes. To that end, Table III summarizes an overview of the main characteristics of the state of the art approaches. Apparently, COPS is the only scheme that offers network coding support without any additional overhead. Nevertheless, there are still important challenges that pave the way for future research, as follows.

A. Protocol optimization

Let us recall that the performance of COPS depends both on the number (L) of feedback periods for the calculation of D_m and the total number (Q) of D_m samples. The estimation of the optimal values of L and Q would be of crucial importance in realistic systems, where the timely detection of malicious users is required. Although a high number of L and Q increases the probability for accurate detection, it also increases the detection delay. Moreover, as these values highly depend on the particular scenario, their adaptability in real scenarios where the rest variables (e.g., misbehavior probability or channel conditions) could change, would be another interesting research line.

B. Level of centralization

The operation of COPS framework is based on the existence of a central controller (e.g., base station) that is able to receive the ACK packets from the nodes and makes all the processing and the decisions. However, different levels of centralization could be also considered in order to bring our proposal closer to distributed solutions. The placement of the controller could be also subject to optimization and the possible trade offs could be also studied.

C. Testbed implementation

The implementation of COPS in an experimental testbed would be the first step towards its application in real scenarios. More specifically, under ideal channel conditions, it is expected that our solution would have identical performance as in the conducted simulations. However, as the performance of COPS is highly dependent on channel errors (since a node could be falsely characterized as non-cooperative), its deployment in real networks under realistic channel conditions would reveal additional aspects for its performance.

VII. Concluding Remarks

In this paper, we introduced COPS, a novel cooperative nonparametric statistical framework for the mitigation of user misbehavior in network-coding-aided scenarios. The proposed scheme can be easily applied in wireless networks, as its operation is based on the processing of existing control packets and does not require any additional overhead nor channel monitoring. More specifically, COPS detects and identifies the malicious users by incorporating two well-known statistical methods: i) the Kruskal-Wallis test, which examines whether all the control packets have been generated by a single population (i.e., honest users), and ii) the Conover-Iman multiple pairwise comparisons, which identify and separate the different groups of users (i.e., honest and malicious) in the network. Extensive simulation results have shown that COPS detects and identifies efficiently the malicious users in the network, even when they adopt a flexible probabilistic misbehavior (e.g., transmitting only one out of three fake reports).

References

[1] C. Bockelmann, N. Pratas, H. Nikopour, K. Au, T. Svensson, C. Stefanovic, P. Popovski, and A. Dekorsy, "Massive machine-type communications in 5G: Physical and MAC-layer solutions," *IEEE Communications Magazine*, vol. 54, no. 9, pp. 59–65, September 2016.

[2] R. Ahlswede, N. Cai, S.-Y. Li, and R. Yeung, "Network information flow," *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1204–1216, Jul 2000.

[3] J. Zhu, "Exploiting opportunistic network coding for improving wireless reliability against co-channel interference," *IEEE Transactions on Industrial Informatics*, vol. PP, no. 99, pp. 1–1, 2015.

[4] G. Theodorakopoulos and J. Baras, "Game theoretic modeling of malicious users in collaborative networks," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 7, pp. 1317–1327, September 2008.

[5] N. Marchenko, T. Andre, G. Brandner, W. Masood, and C. Bettstetter, "An experimental study of selective cooperative relaying in industrial wireless sensor networks," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 3, pp. 1806–1816, Aug 2014.

[6] C. Alocious, H. Xiao, and B. Christianson, "Resilient misbehaviour detection MAC protocol (MD-MAC) for distributed wireless networks," in *2016 IEEE Wireless Communications and Networking Conference*, April 2016, pp. 1–6.

[7] L. Zhang, G. Ding, Q. Wu, Y. Zou, Z. Han, and J. Wang, "Byzantine attack and defense in cognitive radio networks: A survey," *IEEE Communications Surveys Tutorials*, vol. 17, no. 3, pp. 1342–1363, thirdquarter 2015.

[8] S. Djahel, F. Nait-abdesselam, and Z. Zhang, "Mitigating packet dropping problem in mobile Ad Hoc networks: Proposals and challenges," *IEEE Communications Surveys Tutorials*, vol. 13, no. 4, pp. 658–672, Fourth 2011.

[9] O. Ileri, S.-C. Mau, and N. Mandayam, "Pricing for enabling forwarding in self-configuring Ad Hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 1, pp. 151–162, Jan 2005.

[10] D. Zhang, R. Shinkuma, and N. B. Mandayam, "Bandwidth exchange: An energy conserving incentive mechanism for cooperation," *IEEE Transactions on Wireless Communications*, vol. 9, no. 6, pp. 2055–2065, June 2010.

[11] T. Chen and S. Zhong, "An enforceable scheme for packet forwarding cooperation in network-coding wireless networks with opportunistic routing," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 9, pp. 4476–4491, Nov 2014.

[12] T. Ning, Y. Liu, Z. Yang, and H. Wu, "Incentive mechanisms for data dissemination in autonomous mobile social networks," *IEEE Transactions on Mobile Computing*, vol. PP, no. 99, pp. 1–1, 2017.

[13] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile Ad Hoc networks," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '00. New York, NY, USA: ACM, 2000, pp. 255–265.

[14] K. Liu, J. Deng, P. Varshney, and K. Balakrishnan, "An acknowledgment-based approach for the detection of routing misbehavior in MANETs," *IEEE Transactions on Mobile Computing*, vol. 6, no. 5, pp. 536–550, May 2007.

[15] S. Dehnie, S. Tomasin, and R. Ghanadan, "Sequential detection of misbehaving nodes in cooperative networks with HARQ," in *MILCOM 2009 - 2009 IEEE Military Communications Conference*, Oct 2009, pp. 1–6.

[16] H. Yang, J. Shu, X. Meng, and S. Lu, "SCAN: Self-organized network-layer security in mobile Ad Hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 261–273, Feb 2006.

[17] E. Shakshuki, N. Kang, and T. Sheltami, "EAACK - A secure intrusion-detection system for MANETs," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 3, pp. 1089–1098, March 2013.

[18] J. Dias, J. Rodrigues, F. Xia, and C. Mavromoustakis, "A cooperative watchdog system to detect misbehavior nodes in vehicular delay-tolerant networks," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 12, pp. 7929–7937, Dec 2015.

[19] Y. Zhang, L. Lazos, and W. Kozma, "AMD: Audit-based misbehavior detection in wireless Ad Hoc networks," *IEEE Transactions on Mobile Computing*, vol. 15, no. 8, pp. 1893–1907, Aug 2016.

[20] N. Marchang, R. Datta, and S. K. Das, "A novel approach for efficient usage of intrusion detection system in mobile ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 2, pp. 1684–1695, Feb 2017.

[21] J. Dong, R. Curtmola, and C. Nita-Rotaru, "Secure network coding for wireless mesh networks: Threats, challenges, and directions," *Computer Communications*, vol. 32, no. 17, pp. 1790 – 1801, 2009.

[22] N. Alrajeh, S. Khan, J. Lloret, and J. Loo, "Artificial neural network based detection of energy exhaustion attacks in wireless sensor networks capable of energy harvesting," *Adhoc and Sensor Wireless Networks*, vol. 22, no. 1-2, pp. 109–133, 2014.

[23] D. Koutsonikolas, C.-C. Wang, and Y. Hu, "Efficient network-coding-based opportunistic routing through cumulative coded acknowledgments," *IEEE/ACM Transactions on Networking*, vol. 19, no. 5, pp. 1368–1381, Oct 2011.

[24] M. Kim, M. Medard, and J. Barros, "Algebraic watchdog: Mitigating misbehavior in wireless network coding," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 10, pp. 1916–1925, December 2011.

[25] W. H. Kruskal and W. A. Wallis, "Use of ranks in one-criterion variance analysis," *Journal of the American Statistical Association*, vol. 47, no. 260, pp. pp. 583–621, 1952.

[26] W. J. Conover and R. L. Iman, "On multiple-comparisons procedures," Los Alamos Scientific Laboratory, Tech. Rep. LA-7677-MS, 1979.

[27] Q. Zhang, P. Varshney, and R. Wesel, "Optimal bi-level quantization of i.i.d. sensor observations for binary hypothesis testing," *IEEE Transactions on Information Theory*, vol. 48, no. 7, pp. 2105–2111, Jul 2002.

[28] J. Gibbons and S. Chakraborti, *Nonparametric statistical inference*, 5th ed. Chapman and Hall/CRC, 7 2010.

[29] C. Walck, "Handbook on statistical distributions for experimentalists," University of Stockholm, Stockholm, Sweden, Tech. Rep. SUF-PFY/96-01, 2007.

[30] E. Ibarra, A. Antonopoulos, E. Kartsakli, J. J. P. C. Rodrigues, and C. Verikoukis, "QoS-aware Energy Management in Body Sensor Nodes powered by Human Energy Harvesting," *IEEE Sensors Journal*, vol. 16, no. 2, pp. 542–549, Jan 2016.

Angelos Antonopoulos received the Ph.D. degree from the Technical University of Catalonia (UPC) in 2012. He is currently a Researcher with CTTC/CERCA. He has authored over 80 peer-reviewed publications on various topics, including Internet of Things, energy efficient network planning, 5G wireless networks, cooperative communications and network economics. He has been nominated as Exemplary Reviewer for the IEEE Communications Letters, while he has received the best paper award in IEEE GLOBECOM 2014, the best demo award in IEEE CAMAD 2014, the 1st prize in the IEEE ComSoc Student Competition (as a Mentor) and the EURACON best student paper award in EuCNC 2016.

Christos Verikoukis received his Ph.D. degree from UPC in 2000. He is currently a Senior Researcher with CTTC/CERCA and an Adjunct Professor with UB. He has authored over 115 journal papers and over 180 conference papers. He has coauthored over three books, 14 chapters, and two patents. He has participated over 30 competitive projects and has served as the principal investigator of national projects. He has supervised 15 Ph.D. students and five postdoctoral researchers. He is currently the Chair of the IEEE ComSoc CSIM TC. He received a best paper award in the IEEE ICC 2011, the IEEE GLOBECOM 2014 and 2015, EUCNC 2016 and the EURASIP 2013 best paper award of the Journal on Advances in Signal Processing.