Collaborative Watchdog to Improve the Detection Speed of Black Holes in MANETs

Manuel D. Serrat-Olmos, Enrique Hernández-Orallo, Juan-Carlos Cano, Carlos T. Calafate, Pietro Manzoni*

***Abstract*—Watchdogs are a well-known mechanism to detect threats and attacks from misbehaved and selfish nodes in computer networks. In infrastructureless networks, such as MANETs, attack detection and reaction is a key issue to the whole network. Watchdog systems overhear traffic and perform analysis using data collected to decide the grade of misbehaviour that their neighbour nodes present, so accuracy and detection speed play a key role in achieving the right level of network security and performance. The problem behind the use of watchdogs is that they can cause a relatively high level of false positives and false negatives, so increasing their accuracy is also basic to successfully implement this technology in real MANET environments. This paper proposes a collaborative approach for detecting black holes and selfish nodes in MANETs, using a set of watchdogs which collaborate to enhance their individual and collective performance. The paper shows that using this collaborative watchdog approach the detection time of misbehaved nodes is reduced and the overall accuracy increased.**

***Index Terms*—MANET; black holes; watchdog; collaborative**

I. Introduction

A Mobile Ad Hoc Network, usually known as MANET, consists in a set of wireless mobile nodes that function as a network in the absence of any kind of centralized administration and networking infrastructure. These networks rely on cooperation from their nodes to correctly work, that is, every network node generates and sends its own packets and forwards packets in behalf of other nodes. These nodes could be classified [1] as well-behaved nodes, if they cooperate with the MANET forwarding activities to achieve the community goals, or as misbehaved nodes, if they act against those global goals. In this case, nodes are further classified into three classes: faulty nodes, if they do not cooperate due to a hardware or software malfunction; selfish nodes, if they drop all the packets whose destination node are not themselves, but they use other nodes to send their own packets; and malicious nodes, when they try to disturb the normal network behaviour for their own profit.

When a MANET is deployed, we have to assume that there could be a percentage of misbehaved nodes. The types of misbehaved nodes, their number, and their positions and movement patterns are key issues which deeply impact the mobile ad hoc network performance [2]. Additionally, network performance could be drastically reduced if nothing is done to cope with these threats. To this end, an effective protection against misbehaved nodes will be mandatory to preserve the correct functionality of the MANET [3].

In a MANET there are basically two kinds of packet flows: data packets and route maintenance packets. However, not all misbehaved nodes have the same impact on network performance, due to the type of packet flows they affect. A really malicious node could damage the network, spoofing routes, flooding the wireless channel, or carrying out a man-in-the-middle attack. These are classical attacks that every network could suffer, and solutions have been devised for them.

It is clear that some classical attacks can be easily carried out in MANETs because of the nature of the wireless communications channel. We are interested in those potential attacks which are specific to MANETs or whose effects are significantly worse in this kind of networks.

All types of misbehaved nodes –faulty, selfish and malicious– have a common behaviour: they do not participate in forwarding activities, thus being characterized as black holes. A *black hole attack* is a type of attack in which a node intends to disrupt the communication with its neighborhood by attracting all traffic flows in the network and then dropping all packets received without forwarding them to their final destination [4]. To avoid or significantly reduce this type of attack in MANETs, several proposed approaches are based on monitoring the traffic heard by every node to detect misbehaved nodes, and then taking the appropriate actions to avoid the negative effects of that misbehaviour [5].

The main problem that arises at this point is how to detect these black holes avoiding as much as possible wrong diagnostics, like false positives or false negatives. A false positive appears when the selected technique identifies a well-behaved node as a misbehaved node. A false negative appears when the technique can not detect a misbehaved node, so the network believes that it is a normal node, with its potentially disruptive effects. So, accuracy and detection speed are critical issues when designing an approach for black holes detection in MANETs. Another problem is what to do when a black hole is detected. Basically, there are two approaches in the literature: isolation and incentivation. Isolation methods are intended to keep the misbehaved nodes outside the network, excluding them from any ongoing communication. Incentivation methods try to convince the selfish nodes to change their behaviour, being collaborative instead of selfish. Isolation protects the working network, although it could lead to network partitioning. Incentivation tries to improve the MANET communication capabilities by increasing the number of collaborative nodes

*Departamento de Informática de Sistemas y Computadores. Universidad Politécnica de Valencia. mdserrat@upvnet.upv.es, (ehernandez,jucano,calafate,pmanzoni)@disca.upv.es

and the general collaboration level. Isolation is the only suitable method for all classes of black holes. Incentivation is useful only for selfish nodes.

In this work we propose a collaborative watchdog which integrates techniques from reputation systems and bayesian filtering, and makes extensive use of the collaborative nature of MANETs. This watchdog must be considered as an Intrusion Detection Systems (IDS), which is a software piece that collects and analyzes network traffic to detect a set of attacks. In this context, intrusion detection systems aim at monitoring the activity of the nodes in the network in order to detect misbehaviour [4]. Usually, this kind of software products are built using two building blocks: a Detection (or sensor) module, like watchdogs, and a Response module.

The rest of this paper is organized as follows. Section II summarizes the related work on the isolation and incentivation issues in MANETs. Section III presents the concept of bayesian watchdog, which is a basic technique to detect black holes in MANETs. Section IV presents an enhanced proposal for a collaborative watchdog designed to perform that task. Section V evaluates its benefits and, finally, Section VI provides some concluding remarks.

II. Related work

Several solutions have been proposed for detecting and isolating or incentivating misbehaved nodes in MANETs. Marti et al. [6] proposed a Watchdog and a Pathrater over DSR protocol to detect non-forwarding nodes, maintaining a rating for every node and selecting routes with the highest average node rating. The response module of this technique only relieve misbehaved nodes from forwarding packets, but they continue getting their traffic forwarded across the network. Buchegger and Le Boudec [7] proposed the CONFIDANT protocol over DSR, which combines a watchdog, a reputation system, Bayesian filters and information obtained from a node and its neighbours to accurately detect misbehaved nodes. The system's response is to isolate those nodes from the network, punishing them indefinitely.

Others approaches do not use reputation systems, in favor of incentivation. Buttyan and Hubaux [8], [9] presented a method using a virtual currency called *nuglet*. Every node has a credit counter which will be increased when the node forwards packets, and decreased when a node sends his own packets. When a node has no nuglets, it can't send its packets, so it is a motivation for nodes to forward packets for the network benefit. Zhong et al. [10] proposed SPRITE, a credit-based system to incentivate participation of selfish nodes in MANET communication. It's based on a Central Clearance System, which charges or gives credit to nodes when they send or forward a message. So, if a node wants to send a message, it must have sufficient credit to do it. That credit is earned by forwarding messages for other nodes. The response module of this method is integrated into the incentivation method, so that if a node doesn't forward other nodes' messages, it wont have credit to send its own messages.

Many of these approaches use the concept of reputation to improve the detection of black holes, just as reputation is used in human relations. If a node group says that other node is malicious, it is quite probable that this is true. So, it seems a good idea to integrate reputation systems in the mechanism to detect misbehaved nodes. Therefore, watchdog cooperation will probably increase accuracy and detection speed.

III. Bayesian Watchdog

As we stated earlier, to detect misbehaved nodes, network monitoring is needed. Every node must be aware of its neighbours' behaviour, and watchdogs are a popular component for Intrusion Detection System dedicated to this task. The main problem is that watchdogs are characterized by a significative amount of false positives [4], basically due to mobility and signal noise. Previous works from our group [11] have evaluated a bayesian watchdog over Ad-hoc On-demand Distance Vector (AODV) routing in MANETs. This bayesian watchdog results from the aggregation of a bayesian filter with a standard watchdog implementation.

The standard watchdog simply overhears the packets transmitted and received by its neighbours, counting the packets that should be retransmitted, and computing a trust level for every neighbour as the ratio of "packets retransmitted" to "packets that should have been retransmitted". If a node retransmits all the packets that it should had retransmitted, it has a trust level of 1. If a node has a trust level lower than the configured tolerance threshold, that node is marked as malicious.

The role of the bayesian filter in the watchdog is to probabilistically estimate a system's state from noisy observations [11]. The mathematical foundation of the bayesian filter is the following: at time t, the state is estimated by a random variable ϑ, which is unknown, and this uncertainty is modeled by assuming that ϑ itself is drawn according to a distribution that is updated as new observations become available. It is commonly called *belief* or $Bel_t(\vartheta)$. To illustrate this, let's assume that there is a sequence of time-indexed observations $z_1, z_2, ..., z_n, ..., z_t$. The $Bel_i(\vartheta)$ is then defined by the posterior density over the random variable ϑ conditioned on all sensor data available at time t:

$$Bel_t(\vartheta) = p(\vartheta|z_1, z_2, ..., z_n, ..., z_t) = Beta(\alpha_t, \beta_t, \vartheta) \quad (1)$$

In this approach, the random variable ϑ belongs to the interval [0,1]. Bayesian filtering relies on the Beta distribution, which is suitable to estimate the belief in this interval, as shown in expression 1; α and β represent the state of the system, and they are updated according to the following equations:

$$\alpha_{t+1} = \alpha_t + z_t \quad (2)$$

$$\beta_{t+1} = \beta_t + z_t \quad (3)$$

The Beta function only requires two parameters that are continuously updated as observations are made or reported. In this approach, the observation z_t represents the information from the local watchdog obtained in time interval $[t, t + \Delta t]$ about the percentage of non-forwarded packets. The bayesian watchdog uses three parameters: the first two parameters are α and β, which are handled over to the Beta function to obtain an estimation of the node's maliciousness. Thus, we can say that α and β are the numeric representation of a node's reputation. The third parameter is γ, which represents the devaluation that old observations must suffer to adapt the watchdog's behaviour to a continuously changing scenario without penalizing certain nodes forever. It is a mechanism to reintegrate nodes into the MANET if they change their behaviour to a more cooperative one.

As a result of their work, Hortelano et al. [6] found that, compared to the standard one, the bayesian watchdog reached a 20% accuracy gain, and it presents a faster detection on 95% of times.

IV. Collaborative Bayesian Watchdog

Based on the bayesian watchdog presented in Section III, we have implemented a collaborative bayesian watchdog based on a message-passing mechanism in every individual watchdog that allows publishing both self and neighbour reputations. Every node running our watchdog collects the reputation information to obtain the values of α' and β' for every neighbour. The underlying idea of our approach is that if a bayesian watchdog works well for detecting black holes, a group of collaborating neighbouring bayesian watchdogs would be able to perform faster and more accurate detections.

Similarly to the bayesian watchdog, the collaborative bayesian watchdog overhears the network to collect information about the packets that its neighbours send and receive. Additionally, it obtains the α and β values for its whole neighbourhood. These values are exactly the same that those obtained by the bayesian watchdog with the same observations; we call them 'first hand information' or 'direct reputations'. Periodically, the watchdog shares these data with its neighbours, and we call them 'second hand information' or 'indirect reputation'. In our implementation, indirect reputations are modulated using a parameter δ. Whenever requiered, every node running the collaborative bayesian watchdog calculates, using expressions (4) and (5), the values of α' and β', which in this case are passed to the beta function to obtain an estimation of the maliciousness of a node.

$$\underset{j \in N_i}{\forall} \; \underset{k \in N_j}{\forall} \quad \alpha(i)'_j = \frac{\alpha(i)_j + \delta \cdot mean(\alpha(i)^k{}_j)}{2} \tag{4}$$

$$\underset{j \in N_i}{\forall} \; \underset{k \in N_j}{\forall} \quad \beta(i)'_j = \frac{\beta(i)_j + \delta \cdot mean(\beta(i)^k{}_j)}{2} \tag{5}$$

where

- *i* is the node which is performing detection
- N_i is the neighbourhood of node *i*
- $\alpha(i)_j$ is the value of α calculated for every neighbour *j* of i, obtained from direct observations at *i*
- $\beta(i)_j$ is the value of β calculated for every neighbour *j* of i, obtained from direct observations at *i*
- $\alpha(i)^k{}_j$ is the value of α calculated for every neighbour *j* of i, obtained from observations of every neighbour *k of j*
- $\beta(i)^k{}_j$ is the value of β calculated for every neighbour *j* of i, obtained from observations of every neighbours *k of j*
- δ represents the level of trust or the relative importance that a neighbour's observed reputations have for node i

When indirect reputations arrive at a node from one of its neighbours, it only processes those reputations for its own neighbours, because reputations about nodes that are not in its neighbourhood are useless. Once the reputations have been obtained, and the adequate analysis have been done, the detection only needs a predefined tolerance threshold to identify if a node is misbehaved or not.

Figure 1 shows the main components of the collaborative bayesian watchdog. First, each individual watchdog overhears the network to make direct observations of its neighbours, thereby detecting black holes as the bayesian watchdog does. Periodically, it receives reputation information from its neighbours and evaluates their behaviour taking into account this second hand information and its direct observations.

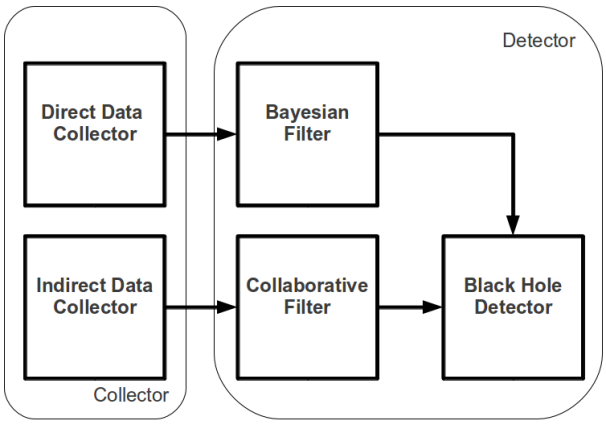

Figure 1. Main components of the collaborative bayesian watchdog

The algorithm of the detector module is outlined in Algorithm 1. Basically, the BayesianDetection function performs analysis over direct observations, obtaining the values of α and β. The relationship between α and β exceeds a predefined tolerance level, the watchdog identifies that node as malicious. Thes values of α and β are also used in the CollaborativeDetection function, according to expressions (4) and (5). This function operates in a similar way that the BayesianDetection function, although it uses second hand information and other parameters to perform its task.

Let's see an example to clarify our proposal. Figure 2 shows an example of a MANET, where dashed lines represent neigh-

Algorithm 1 Black Hole Detector processing algorithm

```
Every observation_time Do
  For all Node_j which is a neighbour
    If ( BayesianDetection() or CollaborativeDetection() )
    Then Node_j is malicious
    EndIf
  EndFor
EndEvery

Function BayesianDetection()
  Obtain observations
  Compute α and β
  If relationship between α and β exceeds tolerance
  Then return true
  Else return false
  EndIf
EndFunction

Function CollaborativeDetection()
  Obtain neighbourhood reputations
  Compute α' and β'
  If relationship between α' and β' exceeds tolerance
  Then return true
  Else return false
  EndIf
EndFunction
```

bour relationships. Table I shows the second hand information received by node A from its neighbours[1].

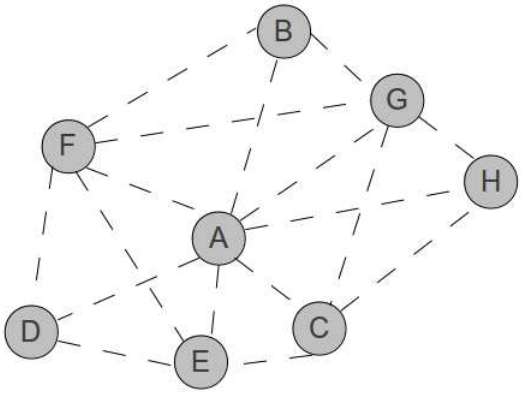

Figure 2. Example of a MANET

Node A combines data from Table I with the direct reputations obtained by itself, and uses a δ value of 1. These operations are executed in every node running the collaborative bayesian watchdog with its own received and produced data, but in this example we show in Table II only the values obtained at node A.

[1]Information received about node A is discarded by the node, and it is not shown here

Table I
SECOND HAND INFORMATION RECEIVED IN NODE A

Neighbour	Reputations received ($\{\alpha(A)^k{}_j, \beta(A)^k{}_j\}$)
B	F: {5,1}, G:{11,1}
C	E:{1,4}, G:{18,1}, H:{1,1}
D	E:{1,2}, F:{7,1}
E	C:{34,1}, D:{1,6}, F:{15,1}
F	B:{1,1}, D:{1,4}, E:{1,3}, G:{13,1}
G	B:{1,2}, C:{52,1}, F:{27,1}, H:{1,6}
H	C:{21,2}, G:{2,13}

Table II
VALUES OF COLLABORATIVE REPUTATIONS CALCULATED AT NODE A OF THE EXAMPLE

Neighbour	Reputations		$\{\alpha(A)'_j, \beta(A)'_j\}$	Detected as Black Hole?	
	Direct	Indirect		Bayesian	Collaborative
B	{1, 2}	{1, 1.5}	{1, 1.75}	No	No
C	{43, 1}	{57, 1}	{50, 1}	No	Yes
D	{1, 4}	{1, 5}	{1, 4.5}	No	No
E	{1, 1}	{1, 3}	{1, 2}	No	No
F	{1, 4}	{14, 1}	{7.5, 2.5}	No	No
G	{3, 1}	{14, 1}	{8.5, 1}	No	No
H	{68, 1}	{44,1}	{56, 1}	Yes	Yes

Having introduced both the mathematical model and the algorithms designed, we now set the objectives we are trying to achieve with this collaborative bayesian watchdog. In this case, we want to:

1) increase the detection speed, reducing the time needed to detect a black hole
2) reduce the production of false positives
3) reduce the production of false negatives

V. EVALUATION

We have implemented our collaborative bayesian watchdog as a Network Simulator 2 (ns-2) extension to the AODV routing protocol. We evaluate the impact that our approach has over the accuracy and the detection speed. We compare the results from the collaborative bayesian watchdog with those obtained using the non-collaborative versions, both bayesian and standard. Table III shows the characteristics of the scenarios we have selected for our performance evaluation.

Table III
SIMULATION PARAMETERS

Parameter	Value
Nodes	50
Area	1000 x 1000 m.
Wireless interface and bandwith	802.11 at 54 Mbps
Antenna	Onmidirectional
Node speed	5, 10, 15 and 20 m/s.
% of black holes	10%
δ	0.8
γ	0.85
Fading	1
Neighbour time	1s.
Observation time	0.2s.
UDP Unicast traffic	Three flows
UDP Broadcast traffic	every 5s.
Simulation time	352 s.
Scenarios	20

Some of these parameters, like area, number of nodes or speed, are needed by ns-2 to execute the simulation. Others, like δ, γ, or *Observation time*, are needed by our code to perform its functionality. For each test, we averaged the results of 20 independent simulations. To obtain normalized results, we simultaneously executed a simulation of the standard watchdog, the bayesian watchdog, and the collaborative bayesian watchdog with the same scenarios and parameters.

A. Detection speed

Accuracy is a key issue when detecting black holes, but speed is also important. A watchdog that detects 100% of black holes but requires 10 minutes is a useless watchdog. So, it is crucial for accuracy and speed to be well balanced. In that sense, watchdog enhancements will target both speed and accuracy issues.

The collaborative bayesian watchdog performed well in terms of speed. Table IV shows that, on average, 7% of the times our approach detected black holes before the bayesian watchdog, with the same traffic pattern. The rest of the cases, it detects the malicious nodes at the same time. When a node B enters[2] node A's neighbourhood, our approach allows node A to identify node B as a black hole with only a reputations sharing phase with its common neighbours. This means that even if node B does not send or receive any data or routing packet when enters node A's neighbourhood, if it has been previously detected as black hole, node A will quickly mark it as a black hole too.

Table IV
PERCENTAGE OF DETECTIONS WHERE THE COLLABORATIVE BAYESIAN WATCHDOG DETECTS THE BLACK HOLES BEFORE THE BAYESIAN WATCHDOG DOES IT

Node Speed (m/s.)	Percentage of earlier detections
5	1.04%
10	11.88%
15	9.66%
20	5.72%

In dense networks with traffic load equally balanced between malicious and well-behaved nodes, both watchdog versions will perform nearly equally, despite of the smaller number of packets that the collaborative bayesian watchdog needs to perform detections. This is because the interval between packets is very short. Nevertheless, in networks with low traffic load and with black holes that transmit a very small amount of packets, the difference of performance between the two approaches could be more significative in terms of time. A single packet would make the difference between detecting or not a black hole, and the collaborative bayesian watchdog obtains better results in those cases.

Additionally, we can say that the collaborative bayesian watchdog obtains the best results at node speed of 10 m/s. In fact, when node move at 10 m/s and 20 m/s our approach behaves nearly 12% and 6% better respectively. These results lead to the conclusion that the collaborative bayesian watchdog becomes a suitable implementation for Vehicle Area Networks, or VANETs, a type of MANET formed by vehicles in movement which share data when they cross with another car, or communicate with a fixed network infrastructure.

B. Accuracy

Figure 3 shows that the accuracy in detecting false positives and false negatives is also slightly better than with the non-collaborative bayesian watchdog, which comes from the decreased level of false negatives. The fact is that a small amount of black holes, that are not detected with the bayesian watchdog, are now detected by the collaborative bayesian watchdog. In fact, our approach is able to detect cases where a black hole enters and exits from the range of a watchdog quickly. As shown in Figure 3, although there is not a big difference between them, the collaborative bayesian watchdog performs better in terms of accuracy than the bayesian watchdog, despite of the node speed[3]. With respect to the standard watchdog, our approach clearly surpasses it in terms of detection accuracy.

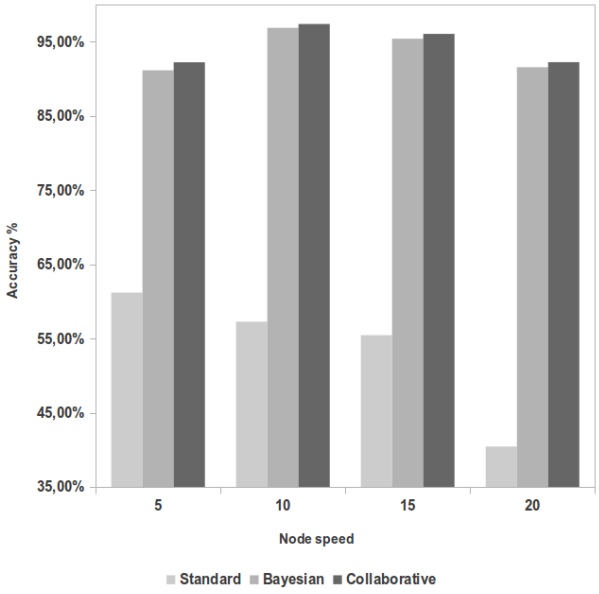

Figure 3. Accuracy comparison of the different watchdog versions

VI. CONCLUSIONS AND FUTURE WORK

In this paper we showed that a bayesian watchdog performs better than a standard watchdog, reducing the amount of false positives. We have also demonstrated that analyzing second-hand information using a collaborative bayesian watchdog will also help at boosting its performance by decreasing the amount of false negatives and speeding up the detection process. As

[2]In this context, entering a node's neighbourhood means that this node is in communication range and it announces its presence, for example, with a standard HELLO message

[3]The standard watchdog has a poor performance, as stated in [6] and as shown in Figure 3. Further comparisons with that version of the watchdog will not be done.

a result, in the scenarios we tested our approach improves detection speed of black holes, and slightly increases the accuracy of that detection process. These conclusions evidence that, compared to previous versions, our watchdog technique fits not only generic MANET environments, but also VANET environments.

As future work, we will implement this mechanism in a hardware testbed (Castadiva), working on the fine tuning of the collaborative bayesian watchdog.

ACKNOWLEDGMENTS

This work was partially supported by the *Ministerio de Ciencia e Innovación*, Spain, under Grant TIN2011-27543-C03-01.

REFERENCES

[1] C.-K. Toh, D. Kim, S. Oh, and H. Yoo, "The controversy of selfish nodes in ad hoc networks," in *Proceedings of the Twelveth international conference on Advanced communication technology (ICACT'10)*, 2010.

[2] T. Sundarajan and A. Shammugam, "Modeling the behavior of selfish forwarding nodes to stimulate cooperation in manet," *International Journal of Network Security and its Applications (IJNSA)*, vol. 2, April 2010.

[3] F. Kargl, A. Klenk, S. Schlot, and M. Webber, "Advanced detection of selfish or malicious nodes in ad hoc networks," in *Proceedings of the First European Conference on Security in Ad-Hoc and Sensor Networks (ESAS 2004)*, 2004.

[4] J. Hortelano, J.-C. Cano, C.-T. Calafate, and P. Manzoni, "Watchdog intrusion detection systems: Are they feasible in manets?," in *XXI Jornadas de Paralelismo (CEDI'2010)*, 2010.

[5] L. Xu, Z. Lon, and A. Ye, "Analysis and countermeasures of selfish node problem in mobile ad hoc network," in *Proceedings of the Tenth International Conference on Computer Supported Cooperative Work in Design (CSCWD '06)*, 2006.

[6] S. Marti, T. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *Proceedings of the Sixth International Conference on Mobile Computing and Networking (MobiCom'00)*, 2000.

[7] S. Buchegger and J.-Y. Le Boudec, "Self-policing mobile ad hoc networks by reputation systems," *IEEE Communications Magazine*, July 2005.

[8] L. Buttyan and J.-P. Hubaux, "Enforcing service availability in mobile ad-hoc wans," in *IEEE/ACM Workshop on Mobile Ad Hoc Networking and Computing (MobiHOC'2000)*, 2000.

[9] L. Buttyan and J.-P. Hubaux, "Stimulating cooperation in self-organizing mobile ad hoc networks," *Mobile Networks and Applications*, vol. 8, October 2003.

[10] S. Zhong, J. Chen, and Y. Yang, "Sprite: A simple, cheat-proof, credit-based system for mobile ad-hoc networks," in *Proceedings of the Twenty-second Annual Joint Conference of the IEEE Computer And Communications Societies (INFOCOM'03)*, 2003.

[11] J. Hortelano, C.-T. Calafate, J.-C. Cano, M. de Leoni, P. Manzoni, and M. Mecella, "Black-hole attacks in p2p mobile networks discovered through bayesian filters," in *Proceedings of OTM Workshops'2010*, pp. 543–552, 2010.