A NOVEL CHARGING AND ACCOUNTING SCHEME IN MOBILE AD-HOC NETWORKS

Inna Kofman[1] and Nurul Huda[2]

[1]University of Duesseldorf, Department of Computer Science,
Duesseldorf, Germany
[2]Ted Rogers School of Information Technology Management,
Ryerson University, Toronto

ABSTRACT

Because of the lack of infrastructure in mobile ad hoc networks (MANETs), their proper functioning must rely on co-operations among mobile nodes. However, mobile nodes tend to save their own resources and may be reluctant to forward packets for other nodes. One approach to encourage co-operations among nodes is to reward nodes that forward data for others. Such an incentive-based scheme requires a charging and accounting framework to control and manage rewards and fines (collected from users committing infractions). In this paper, we propose a novel charging and accounting scheme for MANETs. We present a detailed description of the proposed scheme and demonstrate its effectiveness via formal proofs and simulation results [15]. We develop a theoretical game model that offers advice to network administrators about the allocation of resources for monitoring mobile nodes. The solution provides the optimal monitoring probability, which discourages nodes from cheating because the gain would be compensated by the penalty.

KEYWORDS

Mobile Ad-hoc Networks, Cooperation, Security, Game Theory, Inspection Game.

1. INTRODUCTION

Mobile ad hoc networks (MANETs) [5] are networks that consist of mobile nodes with limited transmission ranges. Several important and challenging application areas have been identified for mobile ad-hoc networks. In particular the application of MANETs to inter-vehicle communication, multi hop Internet access and disaster relief has received significant attention. In order to allow communication beyond the range, nodes have to forward data on behalf of other nodes. Since a node that forwards packets sent by others spends its own resources, such as battery power, it needs to have a reason to do so. One approach to encourage node cooperation is to use a charging and accounting scheme to pay the owner of the device an amount of money for each forwarded data packet [14]. This can be implemented by allowing the sender to insert a set of coins into a packet to be sent. Every node that forwards the packet is allowed to extract one coin for the job done. A forwarding node can collect coins and redeem for a reward later.

When monetary rewards are involved, dishonest nodes may cheat to gain more than they deserve. For example, a node may take two coins for forwarding a packet instead of the allowed amount of one coin. A sender may re-use coins that were already spent on a previously sent packet. These are deemed cheating or illegal actions.

To prevent cheating, existing schemes use strong cryptography [16] [25] [32] [10] [26] [12] [27] [20] [21] [19] [13] [22] (which is time and power consuming for energy-constrained mobile

DOI : 10.5121/ijnsa.2016.8203

devices), and/or tamper-proof devices [4] [17] [28] [26] [31] (which require modifications to existing devices and increased cost to device owners/buyers).

In this article, we propose a new approach to the charging and accounting problem in MANETs. In this approach, a MANET is monitored by a small number of dedicated mobile nodes - which are termed *police nodes* (PNs), and are trusted entities in the system.

The main purpose of PNs is to encourage regular nodes to act within the parameters of predetermined rules. Node behaviors are monitored by PNs, and misbehaving nodes are penalized for their illegal actions, possibly with monetary punishments. PNs periodically report all collected data about observed nodes to the network administrator (NA) for processing. The NA then determines punishments for nodes that violated charging and accounting rules, as well as rewarding nodes that forwarded data for other nodes. Comparable real world scenario: this scheme passes an envelope and a bag of money to public asking to pass the envelop and take $1 from the bag. Also, warns that there might be police around who might catch you if you are dishonest.

The deployment, management and maintenance of PNs impose extra costs to the network owner or administrator, which should be covered by the fines collected from cheating nodes. One of the objectives of our work presented in this paper is to advise the NA on how much resource (i.e. amount of PNs) to allocate to network monitoring so that the cost incurred does not exceed the amount of fines collected from misbehaving nodes.

We consider this problem as an inspection game [2] between PNs that represent the NA and regular nodes. Inspection games have found a wide practical use in different areas such as arms control, enforcement of environmental regulations, crime control, and economics (accountancy and auditing). In such games, there are two players called inspector and inspectee. The inspector's goal is to ensure that the inspectee obeys pre-determined rules. The inspectee may have a tendency to violate rules if that is beneficial to him. We propose a game model based on a Nash equilibrium that offers a strategy for allocating PNs, and demonstrate that this strategy will discourage nodes from cheating since they will not benefit from such illegal acts [15].

The remainder of this paper is organized as follows. Related work is presented in Section 2. In Section 3, we describe the proposed charging and accounting scheme in detail. In Section 4, we present the proposed game model and extensions to the model using more relaxed assumptions discussed in [15]. We conclude the paper and outline our future work in Section 5.

2. RELATED WORK

One way to stimulate collaboration in mobile ad-hoc networks is the use of reputation-based schemes (e.g. [3], [23]). The general idea of these schemes is as follows: an intermediate end-system forwards a packet only if the sender has a good reputation. The reputation of an end-system is improved by forwarding packets for other end-systems and may be decreased by selfish behavior. While reputation-based approaches encourage collaboration, they do not provide the opportunity to charge users for the service provided by other end-systems. In contrast, charging and accounting-based approaches stimulate the collaboration in ad hoc networks through the exchange of real or virtual currency.

Existing work on charging and accounting can be separated into five main categories: access point-based schemes, schemes using a tamper-proof (or a tamper resistant) device in each node,

schemes that use micropayment methods, schemes where nodes store small receipts locally then report them later to an accounting system and pricing models.

Access point-based schemes require that a fixed access point is integrated into the mobile ad-hoc network, acting as a central authority for charging and crediting [16] [25] [26]. Approaches in this category are based on the assumption that it is difficult to provide charging and accounting mechanisms in a completely distributed fashion. Furthermore the authors argue that central authorities such as ISPs must exist in order to provide Internet access. Access-point based approaches exploit this fact by relying on a central authority to aid in charging and accounting for ad-hoc networks. The central authority is expected to control access points deployed in the geographic area covered by the ad-hoc network. Through the access points the central authority provides means to authenticate and authorize mobile nodes in addition to the basic support for charging and accounting. Communication between the mobile nodes and the central authority is founded on multi-hop relaying performed by intermediate nodes. In existing approaches for access point based schemes charging is done on the basis of number of packets, packet size and the number of hops that a packet has to traverse.

Approaches belonging to the second category require that each end-system is equipped with a tamperproof (or at least a tamper resistant) device [4] [17] [28] [31]. The tamper resistant device typically contains the critical functionality that is required to manage the accounting within each node. As a consequence the approaches presented in these papers do have the advantage that they do not require a central authority to perform the accounting. However, some still require a central authority for other purposes and the question whether devices can indeed be made totally tamper proof at an acceptable cost is still controversial.

Work in the third category uses micropayment methods [12] [27] [10] [18] [20] [19]. Micropayment systems allow small payments during data transmission, in such a way providing a security for every participant. For payments usually a chain of hash values is used created by one-way hash functions.

The fourth category includes schemes where nodes store receipts of their actions locally and then report them to a central accounting system [19] [32] [13] [22]. An approach in this category has some fragments similar to previous categories. Like in the first category it is assumed a presence of a central authority. However, the information that is used by the central authority to decide which node will be charged and which one will be rewarded is provided to the central authority by small reports (receipts) that are sent by nodes themselves. Similar to the approaches from the second category, the schemes in the third category commonly use virtual currencies. However, it is not required to equip each node with the temper proof/resistant devices.

The additional fifth category, effect of price, considers different pricing for channel models and could serve as a supplement for approaches from previous categories [11] [30].

In this paper, we propose a new approach different from the above five categories. It is similar to ensuring that laws are followed like in a modern society: with a presence of a cooperation incentive or/and enforcement mechanism in the system, individual nodes are expected to follow the rules, e.g. to pay for services they have used. A simple model is presented similar to one introduced in [17], but without using a tamper-proof device in each node. Besides, the model described in [17] suffers from two serious drawbacks. First, the average number of nuglets per node does not remain constant since credit messages from neighbours may get lost and since the distance to the destination is estimated. This is dangerous since it may either lead to inflation or starvation. Even if this problem was solved, individual nodes may still starve if they are not fortunate enough to be used as an intermediate node for forwarding packets. The authors also

mention that it is possible for a node that contains two security modules to bounce a message back and forth to earn nuglets without actually transmitting data. The impact of this attack is not yet fully understood. Second, the establishment of a security association may cause significant overhead if the neighbourhood relationship changes frequently.

Our model does not require or depend on any additional hardware. Also there is no necessity for centralized services, like for example, the authentication, authorization and accounting (AAA) architecture. The most important characteristics of the approach are that it is still light-weight and flexible. It is light-weight since we do not need to guarantee that all mobile nodes obey the rules which would require additional overhead and resources. It is flexible because it can be applied for a single network or combinations of networks which belong to different authorities.

3. THE PROPOSED CHARGING AND ACCOUNTING SCHEME FOR MANETS

3.1. Overview of the Proposed Charging and Accounting Scheme

We propose a simple model that motivates nodes in a MANET to cooperate. Our proposed charging and accounting scheme is based on the reward approach used in [17], but we do not use any hardware-based tamper-proof devices to prevent cheating. Instead, we deploy PNs to discourage illegal behaviours. The general idea is that the sender of a packet purchases several coins and load them into the packet. Every intermediate node that forwards the packet to the next hop towards the destination will extract one coin from the packet as the payment for the forwarding. Any ad hoc routing protocol, like AODV [34] or DSR [33], can be used to discover a route to the destination. Forwarding nodes collect coins and send them to the network administrator (NA) to redeem rewards.

A coin c has following structure: $c = (S, n_c)$, where S is the identifier of the purchaser, and n_c is a coin sequence number. A coin is a unique unit, since it contains unique identifiers (S, n_c). Also, a Message Authentication Code (MAC) is appended to each coin to provide the authenticity and integrity of the virtual money. The MAC is a cryptographic checksum created by the NA using the coin and the secret key shared between the NA and the purchaser. When the purchaser receives the coin, it computes the MAC the same way the NA did. If the computed MAC matches the MAC appended to the coin, then the purchaser is assured that the coin was created by NA and was correctly received. In similar manner the NA verifies the validity of submitted coins.

Each coin can be used only once by the purchaser to pay for a packet transmission. Once a coin is spent (*used coin*), it is no longer valid and cannot be reused.In order to pay for the delivery of a packet to the destination, the originator of the packet must purchase a set of coins and load them into the packet. When a sender S wants to transmit a message m to destination D, it estimates the number of hops h that is enough to reach D. The value of h can be chosen as a maximal number of hops is needed for the desired transmission [8], or assessed analytically [29]. S forms a packet that contains the following data:

$(m, S, D, c_1, c_2, c_3, ..., c_h, n_{seq}, N_1)$
m - message
S - source ID
D - destination ID
$(c_1, c_2, c_3, ..., c_h)$ - set of coins, where h = the number of the estimated hops
n_{seq} - sequence number (to prevent replay attacks)
N_1 - the first forwarding node on the routing path

The sender then creates a digital signature on the packet with its private key, using a signature algorithm such as an elliptic curve, in order to ensure integrity and authentication of the node. Indeed, when the signature is verified the next node is assured that the packet was not modified as well as the previous node is the node that sent the packet. In addition to integrity and authentication, the signature provides non-repudiation that can demonstrate the evidence to a PN. Both the packet and the digital signature are sent to the first forwarding node (N_1):

(m, S, D, c_1, c_2, c_3, ..., c_h, n_{seq}, N_1)Sig_S
where Sig_S is the digital signature on the packet, as computed by S.

An intermediate node i verifies the packet signature, checks whether the sequence number is greater than this one in the previous packet (i.e., verifies that the packet is not replayed) and confirms the availability of preloaded coins (i.e., verifies that the packet is not empty of coins). If one of these checks fails, then the packet is dropped. Otherwise, the intermediate node takes one coin (from the set of available coins) and replaces the identity of the next node on the route by N_{i+1}. Then, it computes a digital signature on the packet using its private key, applies it to the packet, and forwards it to the next hop N_{i+1}:
(m, S, D, c_1, c_2, c_3, ..., c_{h-1}, n_{seq}, N_{1+1})Sig_{Ni}

Intermediate nodes submit collected coins to the NA when they have a fast connection to exchange for rewards.

To prevent cheating (e.g., a node extracting more than one coin from the packet for a one-hop forwarding), a number of PNs are distributed throughout the network randomly, which observe nodes' behaviours with respect to charging and accounting, document their behaviours and report the information to the NA periodically (when they have a fast connection). A PNs can be mobile or static. For their distribution the existing urban infrastructure could be used that evenly covers an area. For example, PN devices could be installed on buses, police cars, gas stations, traffic lights, buildings, exhibition halls, or university campuses. The NA uses the information reported by the PNs to reward cooperating nodes as well as identify cheating nodes and impose fines. The collected fines will be used to pay for the cost of deploying and managing PNs.

It should be noted that the proposed scheme does not aim at monitoring each and every node in the network with 100% accuracy. Instead, the objective is to provide an ability to detect any kind of illegal activity with *a certain probability*. The theoretical game model proposed in this paper aims at providing the NA with a strategy for allocating PNs that will discourage nodes from cheating since they will not benefit from such actions.

3.2. Overhead Analysis

The computation overhead is related to essential extra data and preloaded coins. We assume that the size of a node identifier is 8 bytes, and that the sequence number is 2 bytes. According to the coin structure, it contains a purchaser identifier and a sequence number, so that the size of each coin is 10 bytes plus MAC. For the MAC we chose for example SHA-2 [9] with digest size 32 bytes, because it is a commonly used, secure and with sensible performance algorithm. Thus, the coin size is 10 + 32 = 42 bytes.

The number of coins preloaded into the packet may impose overhead in the case of long-distance transmissions. Since each intermediate node will extract a coin from the packet, the number of coins stored in a packet decreases with each hop.

The coins overhead through the packet transmission route of h hops is: $42(h + (h - 1) + (h - 2) + ... + 2 + 1) = 42(1 + h)h/2$ bytes, and $32(1 + h)h/2$ bytes if only digests are preloaded.

The average extra data overhead is the average length of the fields in the packet header (listed in subsection 3.1): the source ID, the destination ID, coins, the sequence number, the identifier of the next hop node and the appended digital signature 42 bytes long (that obtained from [6]). That is: $8 + 8 + (32(1 + h)h/2)/h + 2 + 8 + 42 = 16h + 84$ bytes, which is acceptable for a not extremely large h.

In addition, it is possible to reduce the overhead significantly if coins are represented by a hash chain [24]. Since this aspect is not the key contribution of the current paper, further investigation is left for future work.

It should be noted that since the proposed scheme assumes the presence of the routing path in a packet, more data overhead might be imposed. In case of DSR routing protocol there is no additional overhead, because the routing path is included into the packet header. In case of AOVD, a sender records the routing path along with the coins. In this case the overhead increases by $8h$ since a node ID is 8 bytes long. Thus, the average extra data overhead is the overhead calculated above ($16h + 84$) plus the routing path length ($8h$): $(16h + 84) + 8h = 32h + 84$, where $h \leq 5$ [8] to ensure a reasonable packet delivery ratio.

3.3. Examples of Common Attacks

Let C_S denote a set of valid coins, i.e., coins that have been purchased by node S and have not yet been used. Correspondingly, NOT(C_S) is a set of invalid coins:

NOT(C_S) = {C_{forge}, C_{Sused}}, where {C_{forge}, C_{Sused}} is the set of coins that were not purchased by S, or "garbage", and C_{Sused} is the set of used coins (that were purchased by S and were preloaded, at one time, into a packet).

- Double use of coins. A node S puts into the generated by the node packet $c_i \notin C_S$ ($0 < i \leq h$). After a PN reports collected data, the NA can discover this after it verifies whether all coins preloaded into the packet are valid (i.e., coins were purchased by the node and were not yet used).
- Double use of coins (another example). A node S puts into the generated by the node packet $c_i \in C_{S'}$ ($0 < i \leq h$), where coin c_i was purchased by another node S'. After a PN reports collected data, the NA can discover it after it verifies whether all coins preloaded into the packet were purchased by the node and not used coins.

- Double charging. An intermediate node takes more than one coin from the packet $c_i, c_j \in C_S$ ($0 < i, j \leq h, i \neq j$), where S is the source of the message. A PN can notice this when checking whether the set of incoming coins is identical to the outgoing set (except for one coin that was taken by the node). Continuous monitoring is not required, since the scenario can be reconstructed by the NA using complementary data reported by more than one PN.

- Double coin submission. An intermediate node F_x takes one coin c_i (as is expected) and just copies another coin c_j of the remaining coins in the packet. F_x then submits both coins $c_i, c_j \in C_S$ ($0 < i, j \leq h, i \neq j$), where S is the original sender of the message. Another intermediate node F_y forwards the packet and extract coin c_j (legally). A PN observes that coin c_j is extracted legally and sends the info to the NA. In this case, the NA can infer that F_x is a cheater.

- Packet drop. A PN can recognize when a packet has not been forwarded by an intermediate node, in accordance with the routing time. When the timer of a packet expires and the outgoing packet has not been matched, the observed packet is considered a dropped packet [7]. Packets might be dropped, for example, due to mobility or bad channel, buffer overflow, and, as a result, the node could be considered a cheater. The solution to this ”false accusation” problem is discussed in [15].

3.4. Algorithm for Node Type Definition and Data Collection by Police Nodes (PNs)

Assume that a PN is in the transmission range of a node n_A and can observe incoming and outgoing fields of the packet header of the node. Based on this information, as well as on the routing time, a PN can recognize what kind of node n_A is, with respect to fields of the packet header: a sender, a forwarding node, a destination node, an uninvolved node, or an unknown node. A PN determines whether the observed node is likely to be a forwarding node (*FN*), a destination node (*DN*) or an uninvolved node (*UN*), according to the incoming packet, and how a node can be recognized as a sender node (*SN*), a forwarding node (*FN*) or an unknown node (as a sequence of illegal/wrong actions or insufficient information obtained by the PN), according to the outgoing (and/or incoming) packet.

Table 1. Types of records used by PN during monitoring.

Record	Description
Rec1: (Node Id, Node type, Packet header)	the node is an uninvolved node (*UN*)
Rec2: (Node Id, Node type, Packet header)	The node is a destination node (*DN*)
Rec3: (Node Id, Node type, Packet header)	the node is a sender node (*SN*)
Rec4: (Node Id, Node type, Packet header)	the node's type is unknown
Rec5: Rec5.1 (Node Id, Node type, Packet header) + Rec5.2 (Node Id, Node type, Packet header, Coin)	the node is a forwarding node (*FN*) and its incoming packet (Rec5.1) and outgoing packet (Rec5.2) correspond the packet forwarding rules
Rec6: (Node Id, Node type, Packet header)	the node's type is unknown - its outgoing packet has not been matched
Rec7: (Node Id, Node type, Packet header)	the node's type is unknown - its incoming packet has not been matched

The PN nodes use seven types of records, which are summarized in Table 1. When a PN observes that n_A accepts an incoming packet, it checks that the packet is addressed to the node. If not (i.e. $N_i \neq n_A$), the node is a *UN* and the record (Rec1: (n_A, *UN*, Packet header)) is saved. Otherwise, it checks whether the node is a *DN*, i.e., whether *D* in the packet contains the identifier of n_A. If so (i.e. $D = n_A$), the observed node is a *DN*, and the record (Rec2: (n_A, *DN*, Packet header)) is saved. Otherwise, it is a *FN*. This means that the packet was delivered to the node, probably for forwarding. In this case, for the observed incoming packet, a corresponding outgoing packet is expected. Both incoming packet and outgoing packet are needed in order to verify whether forwarding is being performed correctly by n_A. For that reason, every incoming packet, which n_A is recognized as a *FN*, is inserted by a PN into a queue *Q* and a timer is set for routing time t_r. When the timer of a packet expires and the outgoing packet has not been matched, the PN

removes the packet from the queue Q and a corresponding record (Rec6: (n_A, unknown, Packet header)) is saved.

When a PN observes an outgoing packet, it checks whether or not the source field S in the packet header contains n_A. If it does (i.e. $S = n_A$), then PN checks the correctness of the next hop selection toward the destination using the routing path in the packet header (n_S, n_1, n_2, ..., n_D). If so (i.e. $N_1 = n_1$), the PN determines the node's type as a sender and saves the record (Rec3: (n_A, SN, Packet header)). If the next hop is unproved (i.e. $N_1 \neq n_1$ - the next hop is not the "best" next hop), then the node's conduct is unknown to the PN and a record is saved as Rec4: (n_A, unknown, PacketHeader).

In the case S does not contain the identifier of n_A (i.e. $S \neq n_A$), it checks whether the queue Q is empty, since it is probably an outgoing packet that is part of a forwarding process. If the queue Q is empty, then such behavior is not specified by any of the node types in the system. This situation is unknown to the PN, because it may possess only partial information (due to the nodes' mobility) or a node's action may not have been provided. Therefore, since this situation must be reported to the NA for consideration, the PN saves a corresponding record (Rec7: (n_A, unknown, Packet header)). If Q is not empty, the PN matches the packet to one of the packets in the queue Q that fulfils the requirements of the packet forwarding rules. In our model, incoming and outgoing packets must be identical, except in the following cases:

- One coin is taken by the observed forwarding node as a reward. A PN should check to see that the set of coins in the incoming packet is the same as the set of coins in the outgoing packet, apart from the one missing coin.

- The next hop node is the "best" next hop towards the destination.

If matching is successful, the PN removes the record with the incoming packet from Q and saves corresponding records for incoming and outgoing packets (Rec5.1: (n_A, FN, Packet header), Rec5.2: (n_A, FN, Packet header, c_i)). Otherwise, the observation is unknown to the PN, and it is saved into (Rec7). Because of the mobility of ad-hoc nodes, it is possible that incoming and outgoing packets are observed by different PNs. After PNs report collected data to the NA, it can verify whether the forwarding process was performed correctly.

4. THE PROPOSED GAME MODEL FOR NODE MONITORING

Our proposed mathematical model is based on the Passenger Ticket Control (PTC) model [1] originally proposed for the Munich Transport and Fares Tariff association (MVV). The purpose of the PTC model is to suggest how to find the optimum frequency of control through which the MVV could monitor passengers in a cost-effective manner. We focus on PTC inspection game model not only because it proofed itself, but because the PTC problem is very similar in its structure and faces similar challenges to the problem of charging and accounting in MANETs.
Indeed, both problems belong to the same category of rule enforcement in modern society problems and both require operational models to resolve them. Since network monitoring performed partially, the inspectee (a passenger and a mobile node, respectively) might want to embrace the opportunity and not to comply with the rules. The inspector (a public transportation inspector and a PN, respectively) verifies the inspectee and tries to reveal its illegal behavior with a maximal possible probability. The total control is not the primary goal that the models strive to achieve (i.e. to control every individual in the system), but an effective and productive deterring effect to prevent disorders. In this section, we adapt the PTC model to develop a game model for the charging and accounting scheme.

As long as the average gain attained by illegal action is not greater than the average loss incurred from the penalty, there is no rational reason for the node to cheat. It is obvious that the decision of a rational individual node whether or not to cheat will depend on the potential average gain and the potential risk and loss of being caught. When, using game theory, we can know how many PNs to deploy based on the cost, the gain accrued from cheating will not exceed the loss incurred from the penalty. The proposed game model is based on the assumptions discussed in [15] that also accommodates realistic assumptions like finite punishments and imperfect monitoring.
In particular, since we assume that all nodes behave rationally, it follows that nodes will not cheat if they run any risk of being caught. The key question, then, that needs to be examined is: can illegal behaviour be detected with at least a minimal positive probability? We answer this question in following subsections.

4.1. The Proposed Game Model

Like the PTC problem, we consider the monitoring problem of the charging and accounting scheme described in Section 3 to be a two-player inspection game in which a PN (representing the NA) plays the role of an inspector and a regular node is an inspectee. The PN (the first player) monitors node behaviours in the network whereas the regular node (the second player) may or may not cheat.

Monitoring of nodes in our system has a character similar to the inspection of passengers in the PTC problem. During monitoring at a certain location, a PN may observe a number of nodes which are in its reception range, similar to a number of passengers in a public transportation vehicle that are being inspected. In contrast to the PTC, a PN cannot be recognized by a mobile node. Also, the incidence, density and travel of mobile nodes vary differently than those of passengers in the public transportation system at a different time of day.

A solution to the problem we consider is a game solution using a Nash equilibrium, as in the PTC problem. Let (see Table 2 that summarizes all notations):

1. f: denotes the average expenditure of a node when it acts legally. Similarly to a passenger fare in PTC, it is a value to pay for the service, i.e. a number of coins that should be preloaded into the packet to pay intermediate nodes for the packet transmission.

2. g: the nodes average gain from illegal actions, i.e. a number of coins that were illegally saved or obtained by the mobile node.

3. b: denotes the penalty for a misbehaving node - a number of coins analogous to fine in PTC model.

4. e: the cost of monitoring per node in coins (including a deployment cost, i.e. expenditures for PNs setting, maintenance/repair, taking the readings from monitors if necessary, etc.)[1] ($e < b$).
Then the game can be presented in the normal form as shown in Figure 1, where (p, 1 - p) is the mixed strategy of the first player (the probability assigned to monitoring/no monitoring), and (q, 1 - q) is the mixed strategy of the second player (the probability assigned to legal/illegal behaviours).

[1] Additional costs/gains are not taken into account, since they do not affect players directly.

Table 2. Notations for nodes control model.

Notation	Description
f	average expenditure of a node when it acts legally
g	node's average gain from illegal actions
b	penalty for a misbehaving node
e	cost of monitoring per node
p	probability the system is monitored
q	probability the mobile node behaves legally

According to the game model, the expected payments E_1 and E_2 of the PN and the regular node, respectively, are as follows:

$$E_1(p, q) = (f - e)pq + (b - e - g)p(1 - q) + f(1 - p)q - g(1 - p)(1 - q)$$
$$E_2(p, q) = -fpq + (g - b)p(1 - q) - f(1 - p)q + g(1 - p)(1 - q)$$

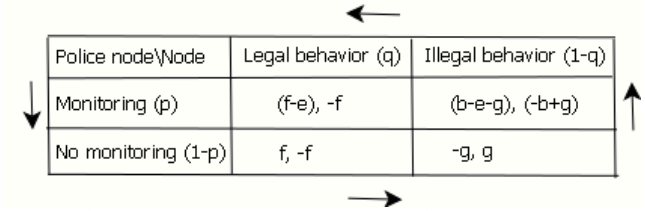

Figure 1. The nodes control game in Ad Hoc Networks

Like the PTC problem, the game has no pure strategy equilibrium because of the cyclical preferences of the players. The pair of mixed strategy equilibrium (p^*, q^*) is comprised of the two players' strategies, where p^* is the inspector's "best response" (optimal monitoring probability) to the passenger's choice of q^* (optimal probability of behaving legally), and q^* is the passenger's "best response" to the inspector's choice of p^*. Based on the fact that two players the PN and the mobile node must be indifferent between all pure strategies and there is no other better pure strategy, the resulting mixed strategy Nash equilibrium can be calculated as following. First, we consider the equilibrium strategy for the PN. The expected payoff for the mobile node from each strategy:

Legal behavior: $(f - e)p + f(1 - p)$
Illegal behavior: $(b - e - g)p + (-g)(1 - p)$

then, setting these payoffs equal to each other, because the PN must be indifferent between his strategies in equilibrium, receive:

$$p^* = (f + g)/b \qquad (1)$$

Next, we calculate the equilibrium strategy for the mobile node, using the payoffs for the PN:
Monitoring: $(f - e)q + (b - e - g)(1 - q)$
No monitoring: $fq + (-g)(1 - q)$

setting equal the expected payoffs, because the mobile node must be indifferent in his strategies, receive:

$$q^* = 1 - e/b \qquad (2)$$

The obtained optimal control probability p^* makes the node indifferent about his two possible action choices, based on the same explanation given in the PTC model in [1]. In fact, equation (1)

holds when probability p^* is chosen for monitoring. Indeed a node which behaves legally pays $-f$, on the one hand, and pays $-bp^* + g$ when it behaves illegally, on the other.

As previously mentioned in the PTC model, the expenditure outlay for control is compensated by the penalty collected when a node chooses q^*. In fact, the difference between the expenditure for monitoring per node (ep) and the gain from the penalty ($bp(1 - q)$) is zero for any p only if the node chooses q^*:

$$ep - bp(1 - q) = p(e - b(1 - q^*)) = 0.$$

For example, let's consider an ideal mobile ad-hoc network with no illegal activities (though that does not exist in reality), $q^* = 1$, and hence no collected fine. In the computed Nash equilibrium the strategy of PNs is the best response to the strategy of mobile nodes, and the strategy of mobile nodes is the best response to the strategy of PNs if f and b are fixed. Thus, for the mobile nodes' strategy not to cheat, the best response of PNs would be not to monitor (with no expenditures for the PNs deployment). In other words this example is a special case of mixed strategies, i.e. a pure strategy. If $p \in \{0, 1\}$ is chosen to be 0 and $q \in \{0, 1\}$ is chosen to be 1, then the PN always plays "No monitor" and the mobile node always plays "Legal behavior".

The equilibrium expected payoffs E_1^* and E_2^* of the PN and the regular node respectively, given the mixed strategy pair (p^*, q^*), are:

$$E_1^*(p^*, q^*) = f(1 - e/b) - eg/b,$$
$$E_2^*(p^*, q^*) = -f.$$

The expected payoff of the regular node given its mixed strategy $(1 - q^*) > 0$ remains the same. Thus we can draw the same conclusion as in the case of the PTC model: the payoff of a legally behaving node is the same as that of a node that behaves illegally and whose illegally achieved gain is negated by the imposed penalty.

4.2. Example

The Nash equilibrium can be interpreted as a steady state. It means that (p^*, q^*) corresponds to a steady state when a randomly chosen player, from the population that he belongs to, plays the same way and does not want to change his behaviour (according to the player's previous experience). Though a player's choices may vary with a certain probability, the way choices are done remains constant during playing the game. It results the same outcome of every play which is the Nash equilibrium. In following we present an example for how the steady state can be found in the PNs and mobile nodes game, in particular, how to advise to the NA on how much PNs to allocate to network monitoring.

Inspector/Passenger	Legal behavior (q)	Illegal behavior (1-q)
Control (p)	0, -3	6, -9
No control (1-p)	3, -3	-1, 1

Figure 2. Outcomes of the nodes control game

Let's assume that f - the average payment of a node when it acts legally - is 3 coins. Since in ad-hoc networks communications are usually up to 5 hops [8], in our example we assume an average hop number, i.e. 3 hops, and hence, 3 coins is the cost to the node for 3 hop distance connection. Also, we assume that g is 1 coin, e.g. every node (in average) uses 1 coin twice. As the

punishment should be as large as possible (see [15]), we chose b to be 10 coins, i.e. the double value of the cost for 5 hops transmission. Assume e equals 3 coins - the cost of monitoring per node that includes all earlier mentioned expenditures. This amount would be exactly covered by the node's legal payment of value f. The outcomes of the game is summarized in Figure 2. Let's p be the probability that the PN chooses to monitor, and $1 - p$ the probability the PN chooses not to monitor. Respectively, let's q be the probability that the mobile node chooses to act legally, and $1 - q$ the probability the mobile node chooses to act illegally. According to equation (1):

$p^* = (f + g)/b = (3 + 1)/10 = 0.4$
$f_{legal}(p) = (f - e)p + f(1 - p) = (3 - 3)p + 3(1 - p) = 3(1 - p)$
and from behaving illegally is:
$f_{illegal}(p) = (b - e - g)p + (-g)(1 - p) = (10 - 3 - 1)p + (-1)(1 - p) = 7p - 1$

If the PN monitor with probability $p^* = 0.4$ and does not monitor with probability $1 - p^* = 0.6$, then the mobile node's payoffs from the legal behaviour and the illegal behaviour are the same and equal to:
Legal behavior: 3 x 0.6 = 1.8 coin
Illegal behavior: 7 x 0.4 - 1 = 1.8 coin

In Figure 3 we illustrate expected payoffs of the mobile node player $f_{legal}(p)$ and $f_{illegal}(p)$ - as a functions of probability p chosen by the PN to monitor the system.

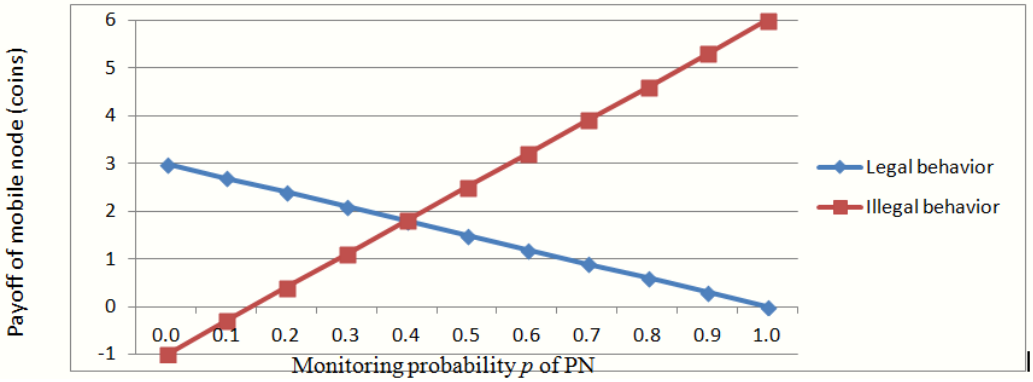

Figure 3. Expected payoff of mobile node for legal and illegal behavior against PN monitoring probability p.

In Figure 3 we can see two linear expected payoff functions intersect when $p = 0.4$. It means that when the PN chooses probability 0.4 for monitoring, the mobile node becomes indifferent between legal and illegal acting.

Now, let's see how the opponent of the mobile node, i.e. the PN, will behave. According to equation (2), the optimal choice of q is:

$$q^* = 1 - e/b = 1 - 3/10 = 0.7$$

For the mobile node's choice of q, the PN's expected payoff from monitor acting is (see Section 4.1):
$f_{monitor}(q) = (f - e)q + (b - e - g)(1 - q) = (3 - 3)q + (10 - 3 - 1)(1 - q) = 6(1 - q)$
and from no monitoring is:
$f_{noMonitor}(q) = fq + (-g)(1 - q) = 3q + (-1)(1 - q) = 4q - 1$

If the mobile node acts legally with probability $q^* = 0.7$ and acts illegally with probability $1 - q^* = 0.3$, then the PN's payoffs from the monitoring and not monitoring are the same and equal to:
Monitor: 6 x 0.3 = 1.8 coin
No monitor: 4 x 0.7 - 1 = 1.8 coin

In Figure 4 we illustrate expected payoffs of the PN player $f_{monitor}$ and $f_{noMonitor}$ - as a functions of probability q chosen by the mobile node to behave legally.

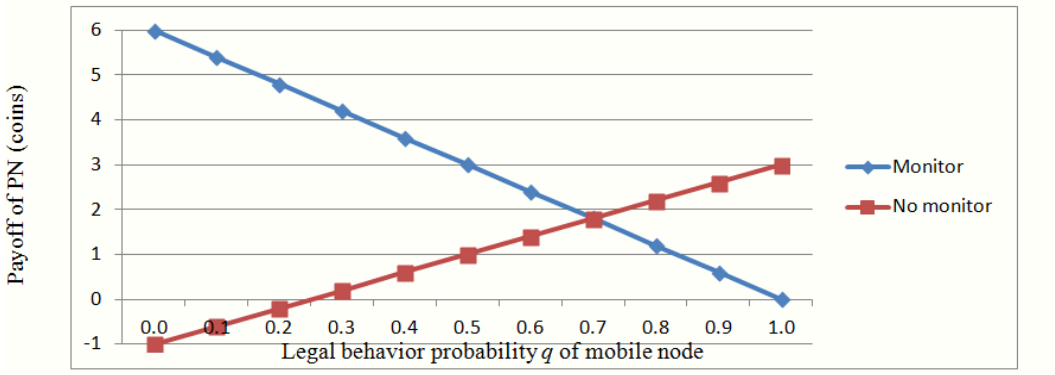

Figure 4. Payoff of PN for monitor and no monitor actions against mobile node legal behavior probability q.

In Figure 4 we can see two linear expected payoff functions intersect when $q = 0.7$. It means that when the mobile node chooses probability 0.7 for acting legally, the PN becomes indifferent between to monitor and not monitor.

The average payoffs when players play their equilibrium strategies (see Section 4.1), for the PN and the mobile node respectively, are:

$E_1^*(p^*, q^*) = f(1 - e/b) - eg/b = 3(1 - 3/10) - 3 \times 1/10 = 1.8$ coin
$E_2^*(p^*, q^*) = -f = -3$ coins

In the example we showed that there is a pair of actions (0.4, 0.7) that is compatible with a steady state, i.e. the state when the PN and the mobile node act always the same way during playing the game. It can be noticed that the monitoring probability 0.4 advised by the model to the NA is relatively high. The reason is that the fraud level, where every individual in average acts illegally in the network, is too high. This situation is an extremum, because in any rational society such a huge crime ratio is unsustainable. However, the proposed model provides a relevant advice how to cope even with the such an unrealistic scenario.

In this example, the model gives an advice to the NA to monitor the system with probability 0.4, which means that 40% of the mobile nodes in the network PNs should be allocated. Such a high level of monitoring makes sense, since the cheat in the example is unusually high, and it is up to the NA whether to reduce PNs resources by increasing the punishment. If the NA would double the punishment value, i.e. $b = 20$ coins, then the required number of PNs would be dropped to 20%, which is a quite reasonable amount allowing to achieve the same goals.

5. CONCLUSIONS

We propose a novel charging and accounting scheme to control and manage rewards and fines in an incentive-based system. The proposed scheme employs police nodes to monitor the network

and report illegal behaviors to the network administrator. We demonstrated the effectiveness of PNs' operations via formal proofs and simulation results [15]. We also propose a theoretical game model that offers advice to the network administrator about the allocation of resources for node monitoring. The solution provides the optimal monitoring probability, which discourages nodes from cheating because the gain would be compensated by the penalty.

In our future work, we will investigate methods to optimally distribute PNs in a network. In addition, non-monetary punishment schemes will be studied, as well as a combination of both kinds of penalties. We will implement the full algorithm of the proposed charging and accounting scheme and evaluate its performance under various network conditions and different routing algorithms, and will investigate further potential attacks. We will also measure the network performance in terms of packet delivery ratio, throughput, and end-to-end delay in the presence of the charging and accounting algorithm to evaluate the overheads of the algorithm when being deployed in a real network.

ACKNOWLEDGEMENTS

We would like to thank Prof. Martin Mauve for help and advice. We are grateful to Prof. Shmuel Zamir for his general comments on the game theory segments in this paper. Also, we would like to thank to Dr. Ariel Frank and Dr. Shifra Hochberg for editorial assistance.

REFERENCES

[1] R. Avenhaus. Applications of inspection games. Mathematical Modelling and Analysis, 9(3):179192,2004.

[2] R. Avenhaus, B. von Stengel, and S. Zamir. Inspection games. In R.J. Aumann and S. Hart (Eds.), Handbook of Game Theory, Volume 3, North-Holland, Amsterdam, 1947 - 1987, 2000.

[3] S. Buchegger and J.-Y. L. Boudec. Performance analysis of the CONFIDANT protocol: Cooperation of nodes -fairness in dynamic ad-hoc networks. In In Proceedings of IEEE/ACM Workshop on Mobile Ad Hoc Networking and Computing (MobiHOC), June 2002.

[4] L. Butty´an and J.-P. Hubaux. Enforcing service availability in mobile ad-hoc WANs. In MobiHoc 2000, pages 87–96, 2000.

[5] I. Chlamtac, M. Conti, and J. J. Liu. Mobile ad hoc networking: Imperatives and challenges. IEEE Networks, 1(1), 2003.

[6] W. Dai. Crypto++ 5.2.1 Benchmarks. http://www.eskimo.com/_weidai/benchmarks.html, 2004.

[7] K. Edemacu, M. Euku, and R. Ssekibuule. Packet drop attack detection techniques in wireless ad hoc networks: a review. International Journal of Network Security & Its Applications (IJNSA), 6(5), September 2014.

[8] P.Gupta and P. R. Kumar. The Capacity of Wireless Networks. In IEEE Transactions on Information Theory, vol. IT-46, no. 2, pages 388–404, March 2000.

[9] P. Hawkes, M. Paddon, and G. Rose. On Corrective Patterns for the SHA-2 Family. Cryptology ePrint Archive, Report 2004/207, 2004.

[10] J. Herrera-Joancomarti and H. Rifa. A Forwarding Spurring Protocol for Multihop Ad Hoc Networks (FURIES). Lecture Notes in Computer Science, 4712, pages 281–293, 2007.

[11] O. Ileri. M.S. Nov. 2003, Thesis: Pricing for Enabling Forwarding for Self-Configuring Ad Hoc Networks (currently pursuing Ph.D. at Rutgers).

[12] M. Jakobsson, J.-P. Hubaux, and L. Butty´an. A Micro-Payment Scheme Encouraging Collaboration in Multi-Hop Cellular Networks. In Proceedings of International Financial Cryptography Conference. Gosier, Guadeloupe, January, 2003.

[13] H. Janzadeh, S. K. Fayazbakhsh, M. Dehghan, and M. S. Fallah. A secure creditbased cooperation stimulating mechanism for MANETs using hash chains. In Future Generation Computer Systems, vol. 25, issue 8, pages 926–934, September 2009.

[14] I. Kofman and M. Mauve. Light-Weight Charging and Accounting in Mobile Ad-Hoc-Networks. ACM SIGMOBILE MobiCom 2005 Poster Session, Cologne, Germany,September 2005.

[15] I. Kofman, U. T. Nguyen, and H. L. Nguyen. A Node Control Model for the Charging and Accounting Problem in MANETs. International Conference on Mobile, Ubiquitous, and Intelligent Computing (MUSIC 2012), Vancouver, Canada, June 26-28, 2012.
[16] B. Lamparter, K. Paul, and D. Westhoff. Charging support for ad hoc stub networks. Elsevier Journal of Computer Communications, 26(13):1504–1514, August, 2003.
[17] L.Butty´an and J.-P. Hubaux. Stimulating Cooperation in Self-Organizing Mobile Ad Hoc Networks. In ACM Mobile Networks & Applications, 8(5), 2003.
[18] M. Mahmoud and X. Shen. Stimulating Cooperation in Multi-hop Wireless Networks Using Cheating Detection System. In Proc. IEEE INFOCOM'10, pages 14–19, San Diego, California, USA,March 2010.
[19] M. Mahmoud and X. Shen. RISE: Receipt-Free Cooperation Incentive Scheme for Multihop Wireless Networks. In Proc. IEEE ICC'11, Kyoto, Japan, June 5 - 9 2011.
[20] M. Mahmoud and X. Shen. DSC: Cooperation Incentive Mechanism for Multi-Hop Cellular Networks. Proc. of IEEE ICC09, Dresden, Germany, June 14–18, 2009.
[21] M. E. Mahmoud and X. Shen. Secure Cooperation Incentive Scheme with Limited Use of Public Key Cryptography for Multi-Hop Wireless Network. In GLOBECOM' 2010, pages 1–5, 2010.
[22] M. E. Mahmoud and X. Shen. FESCIM: Fair, Efficient, and Secure Cooperation Incentive Mechanism for Multi-hop Cellular Networks. In IEEE Trans. on Mobile Computing, Vol. 11, No. 5, pages 753–766, March 2012.
[23] P. Michiardi and R. Molva. Core: A COllaborative REputation Mechanism To Enforce Node Cooperation In Mobile Ad Hoc Networks. In In Proceedings of The 6th IFIP Communications and Multimedia Security Conference, Portorosz, Slovenia, September 2002.
[24] K. Q. Nguyen, Y. Mu, and V. Varadharajan. Digital coins based on hash chain. Proceedings of The 20th National Information Systems Security Conference, Baltimore, USA, pp. 72-79, 1997.
[25] B. Salem, L. Butty´an, J.-P. Hubaux, and M. Jakobsson. A Charging and Rewarding Scheme for Packet Forwarding in Multi-hop Cellular Networks. In Proceedings of ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc), Annapolis, MD, USA, 2003.
[26] N. B. Salem, L. Butty´an, J.-P. Hubaux, and M. Jakobsson. Node cooperation in hybrid ad hoc networks. IEEE Transactions on Mobile Computing, 5(4):365–376, 2006.
[27] H. Tewari and D. O ´ Mahony. Multiparty micropayments for ad-hoc networks. In IEEE Wireless Communications and Networking Conference, WCNC03, New Orleans, Louisiana, USA, 2003.
[28] A. Weyland and T. Braun. Cooperation and Accounting Strategy for Multi-hop Cellular Networks. In Proceedings of IEEE Workshop on Local and Metropolitan Area Networks (LANMAN 2004), pages 193–198, Mill Valley, CA, USA, 2004.
[29] O. Younes and N. Thomas. Analysis of the Expected Number of Hops in Mobile Ad Hoc Networks with Random Waypoint Mobility. In Electronic Notes in Theoretical Computer Science, 275, pages 143–158, 2011.
[30] E. M. Y. Yufang Xi. Pricing, Competition, and Routing for Selfish and Strategic Nodes in Multi-Hop Relay Networks. In INFOCOM, pages 1463–1471, 2008.
[31] Y. Zhang, W. Lou, W. Liu, and Y. Fang. A secure incentive protocol for mobile ad hoc networks. Wireless Networks (WINET), 13, issue 5, October, 2007.
[32] S. Zhong, J. Chen, and Y. R. Yang. Sprite: A Simple, Cheat-Proof, Credit-Based System for Mobile Ad-Hoc Networks. In Proceedings of IEEE INFOCOM, San Francisco, CA, USA, March-April 2003.
[33] M. Mauve, J. Widmer, and H. Hartenstein. A Survey on Position-Based Routing in Mobile Ad Hoc Networks. IEEE Network, 1(6), Dec, 2001, pp. 30–39.
[34] C.Perkins and E. Royer. Ad-hoc on-demand distance vector routing. In In Proc. of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, pages 90–100, 1999.