Multi-Path Routing and Forwarding in Non-Cooperative Wireless Networks

Xueyuan Su, Gang Peng, and Sammy Chan, *Member, IEEE*

Abstract—Multi-path routing and forwarding in non-cooperative networks is extremely challenging due to the co-existence of both *rational* and *Byzantine* nodes. They both might deviate from the protocol; however, their intentions and behaviors are totally different. Rational nodes aim to maximize their utilities, while Byzantine nodes purposefully deviate from the protocol to disrupt the normal operation of a network. Most work in the literature treat both kinds of misbehavior without distinction and thus lead to ineffective solutions. This paper presents a hybrid design that seamlessly integrates mechanisms for different misbehavior in a unified framework. The *GSP auction* provides incentives for rational nodes to cooperate and results in truth-telling Nash equilibria. With the possible inclusion of Byzantine nodes in the least cost paths selected by GSP, the *FORBID mechanism* builds a decentralized reputation system such that malicious behavior is effectively detected. This in turn triggers the GSP auction to update the least cost paths so as to exclude the malicious nodes from being selected for communication. It is proved that the unified protocol is *cooperation-optimal*. Experiments have been conducted to further investigate the performance of the proposed protocol and the impact of various parameters.

Index Terms—Distributed networks, non-cooperative networks, routing and forwarding, mechanism design and analysis

1 Introduction

Non-cooperative wireless networks have been received much attention in recent years, due to the fast advancement of mobile communication and P2P computing. Such a network can be formed by numerous heterogeneous personal mobile devices, such as laptops, tablets, smart phones, and so on. Because these devices are owned by different individuals, their behavior might deviate from the norm due to various reasons.

Early designs of networking protocols often assume that a network consists of *altruistic* nodes that always follow the protocols, and thus mainly focus on handling exceptions due to other communication issues, such as transmission delay and packet loss. These designs tend to be less effective in non-cooperative networks, where there is no guarantee that *selfish* and *malicious* nodes obey the rules. A selfish player is also called *rational* in game theory literature [16], whose intention is to maximize its own utility. A malicious player is referred to be *Byzantine* in distributed computing literature [12], who deliberately deviates from the protocols to disrupt the normal operation of a network. Protocol design becomes quite challenging with the involvement of both rational and Byzantine behavior in the same network.

In this work, we focus on multi-path routing and forwarding in non-cooperative wireless networks, where nodes rely on each other to forward packets to the destination. Sending packets via multiple paths provide benefits such as route resilience, interference avoidance, and load/energy balancing. In the literature, two major approaches are developed for handling routing misbehavior, either incentivizing nodes to cooperate [1], [4], [5], [18], [19], [28], [29], or punishing nodes that refuse to collaborate [2], [14], [15], [17], [22]. Both approaches essentially treat selfish and malicious behaviors non-discriminatingly. However, we understand that neither approach is adequate to effectively deal with both kinds of misbehavior. No incentive-based approach could encourage Byzantine nodes to cooperate, as they are not interested in their utility. On the other hand, to force rational nodes to cooperate via some punishment-based approaches is not reasonable in many cases. For example, if a rational node has already been overloaded or its battery level has dropped below a critical level, forcing it to cooperate certainly degrades the service quality and brings in potential loss. Therefore, we separate the two kinds of misbehavior and treat them accordingly.

We have developed several approaches in previous work to deal with misbehavior individually in each category. We have adapted the GSP auction mechanism from Internet advertising [6], [23] to incentivize rational nodes to cooperate [18], [19], assuming that there are no Byzantine nodes in the network. Usually a mechanism requires payment from the traffic sender which exceeds the total cost incurred at relay nodes, such that they have incentive to participate. The difference between the payment and total cost is referred to as *overpayment*. On the other hand, it is natural that the sender wants to be serviced at low cost and hence small overpayment. It has also been proved in [18] that the GSP mechanism guarantees lower overpayment than the popular VCG mechanism [11], [21]. We have proposed the FORBID

- *X. Su is with Oracle Corporation, Redwood Shores, CA 94065 USA. E-mail: xueyuan.su@oracle.com.*
- *G. Peng and S. Chan are with the Department of Electronic Engineering, City University of Hong Kong, Kowloon Tong, Hong Kong, China. E-mail: gpeng2@alumni.cityu.edu.hk; eeschan@cityu.edu.hk.*

Manuscript received 4 Apr. 2013; revised 15 July 2013; accepted 28 July 2013. Date of publication 14 Aug. 2013; date of current version 17 Sept. 2014. Recommended for acceptance by R. Baldoni.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TPDS.2013.200

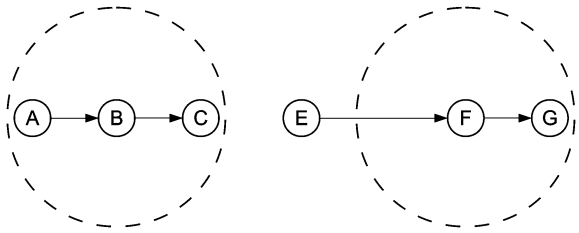


Fig. 1. Unreliable overhearing. Dashed circle represent the transmission range of centering nodes.

mechanism [20] based on a decentralized reputation system to detect and isolate Byzantine behavior, assuming that there are no rational nodes or they have been incentivized to cooperate by other schemes. We have separately shown the effectiveness of the two schemes to their target category of misbehavior, with the assumption the other category of misbehavior does not exist.

In this paper, we make one step further towards appropriate treatment for both categories of misbehavior in a unified framework. The main contributions are:

1. We eliminate the fundamental assumption on the existence of either kind of misbehavior in previous work.
2. We present a hybrid design that seamlessly incorporates GSP and FORBID in a unified framework. With the possible inclusion of Byzantine nodes in the least cost paths selected by GSP, FORBID detects malicious behavior in the forwarding stage, and in turn, triggers GSP to update the least cost paths to exclude the Byzantine nodes from future involvement.
3. We rigourously prove in two steps that the proposed routing protocol is cooperation-optimal.
4. We complement the theoretical analysis with extensive experimental evaluations and demonstrate how rational and Byzantine behaviors are distinctly and effectively handled by the unified protocol.

The rest of this paper is organized as follows. Section 2 formalizes the problem. Section 3 presents the mechanism design. Section 4 provides a rigorous theoretical analysis of the proposed protocol. Section 5 evaluates the performance through extensive experiments. Section 6 reviews related work, and Section 7 concludes the work.

2 Problem Formalization

2.1 The Network Model

The network is a directed weighted graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathcal{W}\}$, where $\mathcal{V}$ is the set of nodes, $\mathcal{E}$ is the set of directed edges representing wireless links, and $\mathcal{W}$ is the set of edge weights representing the actual cost on each individual transmission link. The existence of a directed edge (i, j) between node i and node j depends on the transmission power. Assume each node i is associated with a set $\mathcal{P}_i$ of discrete transmission power levels. For any $i, j \in \mathcal{V}$, there is a minimum power level P_{ij} at which node i could transmit packets to node j. If $P_{ij} \leq \max(\mathcal{P}_i)$, then j is reachable from i and thus $(i, j) \in \mathcal{E}$.

The direction on edges indicate that wireless links in our model are *asymmetric*. Symmetry on wireless links is a fundamental assumption in many previous studies. For example, the famous Watchdog [14] and many other mechanisms based on passive overhearing [2], [15], [17] work well under this assumption. As shown in Fig. 1, when an intermediate node B is forwarding packets from the upstream node A to the downstream next hop C, A could monitor its behavior by overhearing the ongoing transmission. However, this passive overhearing technique fails easily with asymmetric links. When node F is forwarding packets from E to the downstream node G, E is out of F's transmission range and thus may impose false punishment on F.

2.2 The Packet Delivery Model

Packet delivery from source node s to destination node d is divided into two stages: *routing* and *forwarding*.

Routing includes route discovery and selection. In the route discovery process, each intermediate node bids with the reported cost of the outgoing links, which are not necessarily the same as the true cost. After obtaining all the link information and constructing the network graph, the routing protocol computes the set of possible paths as the *least cost path* (LCP) candidates. Let $\mathcal{C}_i$ be the per-packet cost of the path LCP_i. The paths are ordered in such a way that $i < j$ if $\mathcal{C}_i < \mathcal{C}_j$. Among the LCP candidates, the first m paths are selected for packet forwarding. A fraction of data traffic f_i will be forwarded through LCP_i. The per-packet payment is calculated according to the routing decision and the bids placed by intermediate nodes. The bid of each intermediate node is kept confidential by encryption and can only be exposed to s and d. Once the route discovery process is finished, nodes cannot change their bids. Therefore, the auction is a simultaneous-move, one-shot strategic game.

The forwarding stage completes data packet delivery after the paths have been constructed by the routing stage. Each node i is assigned an inherent value $\gamma_i \in [0, 1]$, indicating the probability that a data packet will not be successfully forwarded by i if a path including i is chosen for packet forwarding. γ_i is private for each i. We set a threshold T. If $\gamma_i \leq T$, i is regarded as non-Byzantine and the possible packet loss is caused by interference or link failures. If $\gamma_i > T$, i is regarded as Byzantine and malicious behavior contributes most to packet loss. We accept that non-Byzantine nodes could have non-zero packet dropping probability, such that broader network scenarios are included in our model.

2.3 Assumptions

In our model, we assume s and d always follow the protocol and the player set of this multi-path routing and forwarding game are the *intermediate* nodes $\mathcal{V} - \{s, d\}$. Each intermediate node incurs a *per-packet cost* of forwarding traffic, and this cost is private to itself.

For the sake of simplicity, we have two more assumptions. First, there is no collusion or sybil attack among players. Note that a sybil node can easily form a colluding group of virtual nodes, and thus it violates our no collusion assumption. More advanced techniques from game theory and distributed computing are needed to deal with collusion. Such further development is beyond the scope of this work. Second, each rational node has at least one non-Byzantine node in its transmission range, and Byzantine nodes do not totally

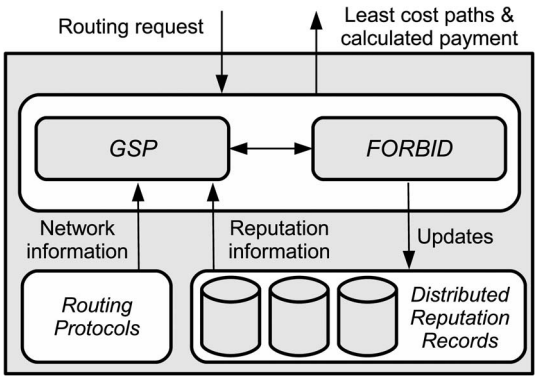


Fig. 2. General framework of the protocol.

partition the source node from the destination node. If a rational node is totally isolated by Byzantine nodes, there is no way for it to communicate correctly with other non-malicious nodes. Similarly, if an s-d cut of the graph that consists of links affected by Byzantine nodes can be formed, then it is impossible to find any path for packet delivery between s and d when all such links are forced down.

3 The Mechanism Design

3.1 Design Objectives

Node i's per-packet payoff or utility u_i is given by $u_i = p_i - c_i$, where c_i is node i's cost and p_i is the received payment. Each rational node aims to maximize its utility.

The design objective of the entire system is not utility maximization. First, it is important to ensure that the packet delivery is still functional despite the presence of selfish and malicious behavior. This requires effective stimulation for selfish nodes, and quick detection and isolation of malicious behavior. Second, it is also desirable to minimize the total transmission cost by properly charging and allocating traffic among the selected paths.

3.2 The General Framework

As illustrated in Fig. 2, the general framework includes two main components, *GSP* and *FORBID*. They work iteratively together to guarantee the normal operation of routing and forwarding. GSP represents generalized second price auction. It is built on top of existing routing protocols to handle route discovery and maintenance. Rational nodes are stimulated by GSP to cooperate on forwarding packets. FORBID stands for flocking oriented Bayesian inference with detections. It maintains and updates reputation information of the nodes in a decentralized manner. With the possible inclusion of Byzantine nodes in the least cost paths selected by GSP, FORBID detects malicious nodes in the forwarding stage. Once malicious behavior is detected by FORBID, GSP is triggered to update the routing paths to exclude the malicious node from being selected for packet forwarding.

3.3 GSP: Stimulating Cooperation for Rational Nodes

The GSP auction mechanism for incentivizing cooperation among rational nodes was first proposed in our early work [18], [19]. We briefly describe its design here.

3.3.1 Route Discovery

The proposed mechanism is built on top of existing routing architecture with minor modifications. We use the dynamic source routing (DSR) to demonstrate how this works. The basic route discovery and route maintenance are handled by DSR by default. Source s broadcasts route request (RREQ) messages to discover available *node-disjoint* paths to d. Intermediate nodes insert routing information into such RREQ messages and forward them to the next hop. Destination d collects routing information stored in RREQ messages and returns it to s via route reply (RREP) messages.

There are several modifications that we made to the standard routing protocol. First, instead of using the hop-count metric, GSP requests each node to bid with its transmission cost and include the bid in the RREQ messages. Such bids are sealed by cryptographic method and can only be opened at the destination d. Second, as will be stated later, a node might decide to drop all the RREQ messages that contain nodes flagged as malicious by the FORBID component. Third, instead of sending the RREQ message along the least cost path, RREP messages are returned to the source s via all available paths.

3.3.2 Auction Design

The GSP mechanism in Internet advertising [6], [23] requires each player in the auction pays the opportunity cost that its presence introduces to the player who obtains the next position. In the routing game [18], [19], we designed GSP such that each node receives payment based on the bids of nodes that form the next candidate path. Suppose node i is on LCP_k. Because of its presence, path LCP_k obtains the k-th position and is allocated with f_k fraction of traffic. If it were not present and thus LCP_k could not be formed, then path LCP_{k+1} would have obtained the k-th position and received f_k of traffic. Let $c_{i,k}$, $\mathcal{C}_k$, and $\mathcal{C}_{k+1}$ be the per-packet cost of node i, LCP_k, and LCP_{k+1}, respectively. Then the per-packet payment for node i is calculated as the difference in total cost between the two cases normalized by the traffic amount,

$$p_{i,k} = \frac{\mathcal{C}_{k+1} f_k - (\mathcal{C}_k - c_{i,k}) f_k}{f_k} = (\mathcal{C}_{k+1} - \mathcal{C}_k) + c_{i,k}.$$

Then the per-packet utility for node i is

$$u_i = p_{i,k} - c_{i,k} = \mathcal{C}_{k+1} - \mathcal{C}_k.$$

3.3.3 Policy Enhancement

There are several basic policies used by GSP.

$\mathcal{P}^{(1)}$: *The number of selected LCP candidates m is always less than the total number of available LCP candidates.*

$\mathcal{P}^{(2)}$: *The fraction of data traffic forwarded through each selected LCP follows that $\sum_{i=1}^{m} f_i = 1$ and $f_1 > f_2 > \cdots > f_m > 0$.* Traffic fractions sum to one. The positivity of traffic fraction guarantees no degeneration on the number of traffic paths. Moreover, allocating more traffic to less expensive path helps minimizing the total cost.

$\mathcal{P}^{(3)}$: *For any $1 \le p < q \le m$, we have $[\mathcal{C}_{p+1} - \mathcal{C}_p]\ f_p > [\mathcal{C}_{q+1} - \mathcal{C}_p]\ f_q > [\mathcal{C}_{q+1} - \mathcal{C}_q] f_q$.* We refer policy $\mathcal{P}^{(3)}$ as the *traffic*

allocation condition. The first inequality $[\mathcal{C}_{p+1} - \mathcal{C}_p]\ f_p > [\mathcal{C}_{q+1} - \mathcal{C}_q]\ f_q$ indicates that a player on a path with less total cost tends to have a better utility than one on a path with more total cost. The second inequality $[\mathcal{C}_{p+1} - \mathcal{C}_p]\ f_p > [\mathcal{C}_{q+1} - \mathcal{C}_p]\ f_q$ states that by overbidding to go from a path with less total cost to one with higher total cost, a player cannot increase its utility.

3.4 FORBID: Detecting and Isolating Byzantine Nodes

The FORBID mechanism for quick detection and isolation of Byzantine nodes was introduced in our previous work [20]. We describe several key design considerations here and refer interested readers to [20] for more details.

3.4.1 Packet Forwarding and Payment Realization

The source s chains several data packets together and sends them as a block with its digital signature. After it receives the confirmation from the destination d and passes the validation check, s sends out the next block.

Each intermediate node j checks the validation upon receiving a data packet from its upstream node i. If it is valid, j returns a MAC layer ACK to i with some cryptographic information that cannot be forged. It then forwards the packet to the next hop node k and keeps the ACK from k as a receipt. This receipt is used to get deserved payment in case the backward confirmation from d cannot be delivered, and to help the reputation system narrow down the suspects of Byzantine nodes. If j cannot get ACK from k, it suspects that k is Byzantine and makes corresponding notes in its reputation record. Node j is also required to retransmit the packet with its highest transmission power level such that at least one neighbor node can overhear the retransmission and act as a witness during the detection process.

Upon successfully receiving each block, d sends a backward confirmation to s. If such confirmation cannot be delivered to s in some extreme cases, the intermediate nodes need to rely on the receipt of MAC layer ACKs discussed above for getting proper payment.

3.4.2 Detection: Identifying Byzantine Behaviors

Data packets are forwarded through each LCP in proportional to its allocated percentage. If packets cannot get through a chosen path LCP_k, s reschedules its allocation to other chosen ones and recalculates payments before going through the expensive rerouting procedure.

At the same time, s initiates a detection process that requires each intermediate node along LCP_k to report its receipt of ACKs from the next hop. If a node can neither report the receipt of ACKs nor have any neighbor node as a witness, its reputation is degraded.

3.4.3 Bayesian Inference: Updating Internal Reputation

Each node i internally maintains a reputation list, which keeps the reputation record for other nodes in the network. Let α_{ij} and β_{ij} be the number of occurrences of evidence available at node i that node j deviates from and follows the protocol, respectively. Let random variable $X_{i,j}$ represent i's belief that how likely j is Byzantine. Based on α_{ij} and β_{ij}, node i computes $\mathbf{E}[X_{i,j}]$ as the reputation value for j, or i's evaluation of γ_j. Each node maintains its own reputation beliefs to make decisions.

Initially, this belief is neutral, i.e., node i regards node j as a Byzantine node with probability 50 percent. This belief is continuously updated by node i when new evidences become available. When such a belief exceeds the threshold T, it discards all the routing information that includes j. Node i will not ask j to forward packets until the belief becomes below the threshold. We rely on Bayesian inference for such internal updates. The beta distribution with parameters (α, β) is used as the prior distribution. The expectation is computed as

$$\mathbf{E}[X_{i,j}] = \frac{\alpha_{ij}}{\alpha_{ij} + \beta_{ij}}.$$

3.4.4 Flocking: Disseminating Reputation Information

FORBID carefully takes advantage of information dissemination to shorten the misbehavior detection time. Each node periodically sends reputation information to its neighbors. From a neighboring node j, node i receives the reputation information $(\alpha_{jk}, \beta_{jk})$ for all $k \neq i, j$. Such information is cryptographically signed by j. After validation check, node i updates its reputation record with a weighted flocking algorithm. Similar to page rank algorithms [13], this algorithm assigns weights to information from neighbors according to their trust levels.

Let $v_t(i)$ be a value stored at i at time t, such as α_{ij} or β_{ij}. The update rule is

$$v_{t+1}(i) = w_t(i,i)v_t(i) + \sum_{j:(i,j)\in G_t} w_t(i,j)v_t(j),$$

where $w_t(i,j)$ is the weight that node i assigns to node j and G_t is the connectivity graph at time t. Since the internal value obtained by detection and observation has the highest priority, $w_t(i,i)$ is set to a large value. The weights assigned to the neighbors are determined by their *trust levels* computed by node i. Assume j is a neighbor of i. Let $\ell(i,j)$ denote its trust level computed by node i. We compute $\ell(i,j)$ as

$$\ell(i,j) = 1 - \mathbf{E}[X_{i,j}].$$

Then the weight that i assigns to j is calculated as

$$w(i,j) = [1 - w(i,i)] \cdot \frac{\ell(i,j)}{\sum_{k:(i,k)\in G_t} \ell(i,k)}.$$

3.5 Discussion on Computational Overhead

Compared to DSR, the major additional computational overhead of our proposed routing protocol is the encryption/decryption of the link cost based on public key cryptography. Efficient algorithms, such as elliptic curve cryptography (ECC), can be used in the implementation of our protocol to reduce this additional computational load. It has been demonstrated that even using a low-computation-power MicaZ mote, the signature and verification times of 160-bit ECC are not more than 2 seconds [10]. As the non-cooperative networks considered in this paper are formed by mobile devices with much higher computational capabilities, such as laptops, tablets, smart phones, much less computation time due to signature and verification would be incurred. Therefore, the computational overhead of our proposed routing protocol would be comparable to that of DSR.

4 Cooperation-Optimality Analysis

In this section, we analyze the proposed protocol and show it is cooperation-optimal. A routing protocol is *routing-optimal*, if Byzantine nodes cannot disrupt the routing stage, and in each of the equilibria it is optimal for each rational node to follow the protocol, which maximizes the prospective utility after forwarding stage. Similarly, a routing protocol is *forwarding-optimal*, if Byzantine nodes cannot disrupt the forwarding stage, and in each of the equilibria it is optimal for each rational node to follow the protocol, which maximizes the prospective utility. A routing protocol is *cooperation-optimal*, if it is both routing-optimal and forwarding-optimal. The main theorem is formally stated as follows.

Theorem 4.1. *The proposed protocol that unifies the GSP and FORBID mechanisms is cooperation-optimal.*

We prove this theorem in two steps. Theorem 4.4 shows the proposed routing protocol is routing-optimal, and Theorem 4.7 shows it is also forwarding-optimal. Therefore, Theorem 4.1 immediately follows.

4.1 Routing Optimality

We show that GSP results in truth-telling Nash equilibria, and the protocol is routing-optimal.

Theorem 4.2. *During the routing stage, under the traffic allocation condition $\mathcal{P}^{(3)}$, there are Nash equilibria among rational nodes where they honestly bid with their true costs to maximize their prospective utilities.*

Proof. Let $a_k^{(r)}$ be the action of node k bidding its true cost of link ℓ. Let $\overline{a}_k^{(r)} \neq a_k^{(r)}$ be a different action. Let $a_{-k}^{(r)}$ be the action profile of all other rational nodes behaving honestly in this stage. For the prospective utility, we show that $u_k\ (\overline{a}_k^{(r)}, a_{-k}^{(r)}) \leq u_k\ (a_k^{(r)}, a_{-k}^{(r)})$ in all cases.

- Case (1): With $a_k^{(r)}$, Link ℓ was on LCP_j, $j > m$, and node k exaggerates the cost of ℓ.
- Case (2): With $a_k^{(r)}$, Link ℓ was on LCP_j, $j > m$, and node k understates the cost of ℓ.
- Case (3): With $a_k^{(r)}$, Link ℓ was on LCP_i, $1 < i \leq m$, and node k exaggerates the cost of ℓ.
- Case (4): With $a_k^{(r)}$, Link ℓ was on LCP_i, $1 < i \leq m$, and node k understates the cost of ℓ.

In case (1), exaggerating the cost could not increase the chance of LCP_j being selected, i.e., $u_k\ (\overline{a}_k^{(r)}, a_{-k}^{(r)}) = u_k\ (a_k^{(r)}, a_{-k}^{(r)}) = 0$.

In case (2), understating the cost could increase the chance of LCP_j being selected. Recall that $\mathcal{C}_j$ as the cost of LCP_j if node k bids the true cost. After understating the cost of link ℓ, if LCP_j is still not selected, then node k will not make a profit and $u_k(a_k^{(r)}, a_{-k}^{(r)}) = u_k(\overline{a}_k^{(r)}, a_{-k}^{(r)}) = 0$. If LCP_j is selected after understating the cost of ℓ, it becomes one of the selected LCP candidates LCP_i, $1 \leq i \leq m$. Note for true cost $\mathcal{C}_j > \mathcal{C}_{i+1}$. Therefore $u_k\ (\overline{a}_k^{(r)}, a_{-k}^{(r)}) = (\mathcal{C}_{i+1} - \mathcal{C}_j) \cdot f_i < 0 = u_k\ (a_k^{(r)}, a_{-k}^{(r)})$.

In case (3), there are three possibilities. First, with $\overline{a}_k^{(r)}$, if link ℓ becomes not selected, then $u_k(\overline{a}_k^{(r)}, a_{-k}^{(r)}) = 0 < u_k(a_k^{(r)}, a_{-k}^{(r)})$. Second, with $\overline{a}_k^{(r)}$, if link ℓ is moved from LCP_i to LCP_j, $i < j \leq m$, according to policy $\mathcal{P}^{(3)}$, we have $u_k(\overline{a}_k^{(r)}, a_{-k}^{(r)}) = (\mathcal{C}_{j+1} - \mathcal{C}_i) f_j < (\mathcal{C}_{i+1} - \mathcal{C}_i) f_i = u_k\ (a_k^{(r)}, a_{-k}^{(r)})$. Third, with $\overline{a}_k^{(r)}$, if link ℓ is still on LCP_i, the nodes on LCP_{i-1} do not have incentive to switch the position down (by overbidding) because their utility is better as is, $(\mathcal{C}_i - \mathcal{C}_{i-1}) f_{i-1} > (\mathcal{C}_{i+1} - \mathcal{C}_{i-1}) f_i$ due to policy $\mathcal{P}^{(3)}$. This observation coincides with the locally envy-free property [6]. Hence $u_k\ (\overline{a}_k^{(r)}, a_{-k}^{(r)}) = u_k\ (a_k^{(r)}, a_{-k}^{(r)})$.

In case (4), there are two possibilities. First, with $\overline{a}_k^{(r)}$, if link ℓ is still on LCP_i, then $u_k\ (\overline{a}_k^{(r)}, a_{-k}^{(r)}) = u_k\ (a_k^{(r)}, a_{-k}^{(r)})$. Second, with $\overline{a}_k^{(r)}$, link ℓ is moved from LCP_i to LCP_j, $1 \leq j < i$. Note for true cost $\mathcal{C}_i > \mathcal{C}_{j+1}$. Therefore $u_k\ (\overline{a}_k^{(r)}, a_{-k}^{(r)}) = (\mathcal{C}_{j+1} - \mathcal{C}_i) f_j < 0 < u_k\ (a_k^{(r)}, a_{-k}^{(r)})$.

Therefore we conclude that node k can only maximize its utility by bidding the true cost of link ℓ. □

Theorem 4.2 is important in the sense that it demonstrates that by policy $\mathcal{P}^{(3)}$, our mechanism results in a set of Nash equilibria where each rational node reveals the true cost. This is an important distinction from the original GSP used in Internet advertising [6], [23].

Lemma 4.3. *Byzantine nodes cannot disrupt the routing stage of the protocol.*

Proof. A Byzantine node has the following possible actions during the routing stage.

- Case (1): It does not forward the RREQ/RREP.
- Case (2): It reports false link costs.
- Case (3): It acts honestly during the routing stage (but potentially does harm in forwarding stage).

By the network model assumption that Byzantine nodes do not partition the source and destination nodes, the RREQ and RREP messages are guaranteed to be delivered successfully in case (1). For the other two cases, although a Byzantine node might be included into the paths for the forwarding stage (and FORBID deals with them later), it does not stop the paths from being computed by the protocol. Therefore, in none of the three cases the routing stage is disrupted by Byzantine nodes. □

Theorem 4.4. *The proposed protocol that unifies the GSP and FORBID mechanisms is routing-optimal.*

Proof. Lemma 4.3 states that the routing stage is not disrupted by Byzantine nodes. Theorem 4.2 states that GSP creates Nash equilibria where rational nodes truthfully reveal their costs. This action maximizes their prospective utilities. Therefore, it is optimal for the rational nodes to follow the protocol during the routing stage, and thus the protocol is routing-optimal. □

4.2 Forwarding Optimality

Lemma 4.5. *Byzantine nodes cannot disrupt the forwarding stage of the protocol.*

Proof. In the forwarding stage, a Byzantine node has the following possible actions.

- Case (1): It receives packets and returns ACK, but does not forward the packets to the next hop.

- Case (2): It pretends not having received any packets from the upstream node—no ACK returned, no data packets forwarded.
- Case (3): It behaves correctly in forwarding the data packets and returning ACK, but does not forward the backward confirmation from the destination d.

In cases (1) and (2), the Byzantine node will be directly identified during the detection process since it could neither report the receipt of ACK from next hop nor have neighbor nodes as witnesses. As designed in FORBID, the innocent upstream node in case (2) retransmits the packet at the highest power level so as to have a witness for justification in the detection process. In case (3), the Byzantine node tries to prevent the upstream nodes from getting their deserved payment. Two methods in FORBID is used to defend against this kind of attack. First, the backward confirmation could go back to s from other available paths. Second, each intermediate node still gets the deserved payment by submitting the receipt of ACKs. Therefore, in none of the three cases the forwarding stage is disrupted by Byzantine nodes. □

Lemma 4.6. *During the forwarding stage, there is a Nash equilibrium among rational nodes where they follow the protocol to maximize their prospective utilities.*

Proof. Let $a_i^{(f)}$ be the action of rational node i following the protocol and $\overline{a}_i^{(f)} \neq a_i^{(f)}$ be the other action. Let $a_{-i}^{(f)}$ be the action profile of all other rational nodes following the protocol. For the prospective utility, we show that $u_i\ (\overline{a}_i^{(f)}, a_{-i}^{(f)}) < u_i\ (a_i^{(f)}, a_{-i}^{(f)})$. Assume that node i is on LCP_j. If it does not forward the packets, then its utility is $u_i\ (\overline{a}_i^{(f)}) = 0$. Now consider the case when node i forwards the packets. Byzantine nodes along the path could try to prevent it from getting payment by dropping packets, ACKs or backward confirmation from the destination d. Under such circumstances, node i is guaranteed to get payment by either submitting the receipt of ACK from next hop, or retransmitting at the highest level and having another node in the neighborhood as a witness. The second approach is possible by the assumption that each rational node has at least one non-Byzantine node in its transmission range. In this case, its per-packet utility is $\mathcal{C}_{j+1} - \mathcal{C}_j > 0$ and thus $u_i\ (a_i^{(f)}) > 0$. Therefore, $u_i\ (\overline{a}_i^{(f)}) < u_i\ (a_i^{(f)})$ and node i would not unilaterally deviate from action a_i^f. This applies to any rational node in the network. Thus, there exists a Nash equilibrium in which each rational node follows the protocol. The resulting utility of each is also optimal. □

Theorem 4.7. *The proposed protocol that unifies the GSP and FORBID mechanisms is forwarding-optimal.*

Proof. Immediately follows Lemmas 4.5 and 4.6. □

5 Experimental Evaluations

We evaluate the proposed protocol via extensive simulations. Consider a network consisting of non-colluding nodes. Nodes do not move once their locations are fixed. We have updated the previously developed event-driven simulator in [19], such that wireless links that form the underlying network topology are determined based on nodes' geographical location and their transmission ranges. Each source node splits traffic among multiple paths according to the protocol. Without loss of generality, packets are generated at each source node at the rate of one packet per time unit. The one-hop transmission latency of a single packet is also one time unit.

Unless stated otherwise, the default parameters are as follows. The network size is $|\mathcal{V}| = 100$. Byzantine nodes constitute 30 percent of the total population, and the rest are rational. Link costs are drawn from [1, 5] independently and uniformly at random. During each simulation run, a random permutation of $\mathcal{V}$ is generated and nodes follow this order to sequentially start their transmission sessions. When it comes to its turn to transmit, a source node s randomly picks a destination node d and generates 100 packets for the transmission session. The session for the next node in the permutation begins immediately after the current one finishes. At the beginning of each run, a fraction of random nodes are chosen to be Byzantine. When acting as an intermediate node, a Byzantine node pretends to work well during the routing stage, but potentially misbehaves during the forwarding stage. We generate two kinds of malicious behaviors: (1) during the forwarding stage, a Byzantine node drops each data packet with some probability generated from a normal distribution $N(0.7, 0.05)$ and truncated to be in [0, 1]; (2) during each step of reputation dissemination, a Byzantine node randomly selects another node and sends false accusation against it with probability 0.5. The flocking algorithm adopts $w_t(i, i) = 0.998$ for all $i \in \mathcal{V}$ and all t in time. The threshold is set to be $T = 0.51$.

The experiments are conducted in a series of scenarios. Section 5.1 mainly targets at rational nodes and evaluates the effect of network topology. Section 5.2 shows the effectiveness of FORBID on detecting and isolating misbehavior. Section 5.3 demonstrate how rational and Byzantine nodes are distinguished by the proposed protocol. For each scenario, the results are obtained by averaging over five independent experiments. Additional results can be found in the supplementary material which is available in the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPDS.2013.200.

5.1 Effect of Network Topology

The effect of network topology is studied. Assume nodes follow the protocol. The metrics are the *credit balance*, which is the payment received from other nodes minus the payment made to others, and the *utility*, which is the payment received from others minus the cost involved in forwarding packets. A set of representative nodes are chosen based on *degrees* and *costs* of their links.

Fig. 3a shows the credit balances of this set of nodes. It is observed that, on one extreme, nodes with high degree and low-cost links are more likely to be on a path selected for transmission. As a result, such nodes have more opportunities to forward packets and thus earn the most positive credit balance. On the other extreme, nodes sparsely connected with high-cost links are less likely to be chosen

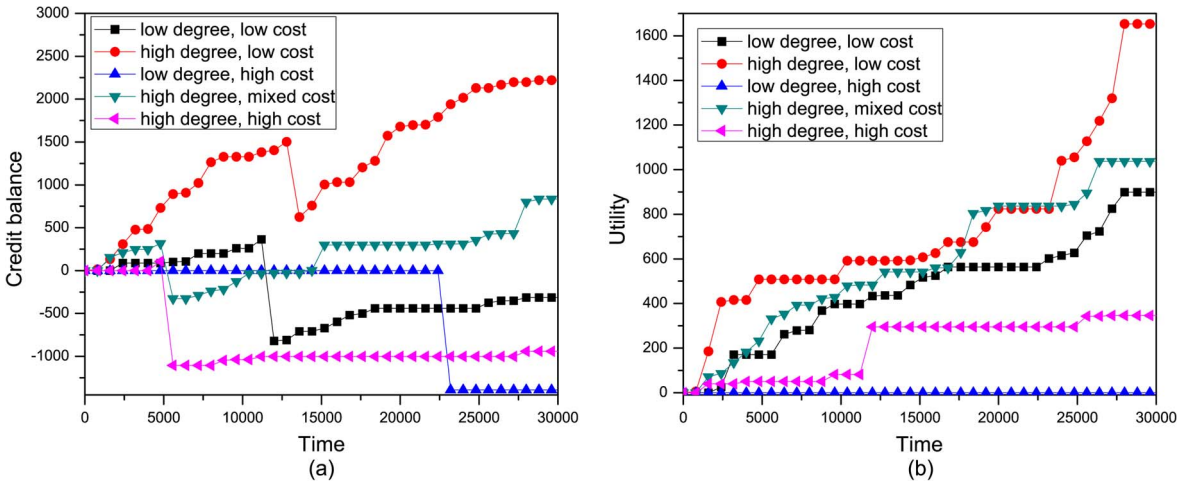


Fig. 3. Impact of network topology on (a) credit balance, (b) utility.

on the forwarding path. Therefore, they exhibit the most negative credit balance after making payment to others. For other nodes, as expected, the curves representing their credit balance lie in between the two extremes. The results on utilities is shown in Fig. 3b. Note that all utility values are positive, and thus the protocol provides incentives for cooperation. Another observation is that nodes of good connectivity and low cost are favored and thus earn better utility.

5.2 Detection and Isolation of Byzantine Behavior

The effectiveness of FORBID on detecting and isolating Byzantine nodes is accessed. We compare FORBID with three other schemes: *Defenseless* scheme employs no mechanism against Byzantine behaviors; *Watchdog-1* enhances Watchdog with an information exchange mechanism by linear opinion pool [3]; *Watchdog-2* further eliminates the false overhearing issue caused by link asymmetry. The performance metric is the *packet loss rate*, which is defined as the ratio between the number of lost packets and the total number of packets sent by all source nodes. Lower packet loss rate indicates quicker detection and isolation of malicious behaviors.

We first vary the *percentage of Byzantine nodes*. The result is as shown in Fig. 4a. Even with a small percent of Byzantine nodes, say 20 percent, Defenseless suffers from significant packet loss rate of over 53 percent. When passive overhearing is accurate, Watchdog is able to effectively mitigate packet loss rate down to 10 percent. When overhearing is less reliable, the packet loss rate increases to about 16 percent. Under the same setting, FORBID reduces the packet loss rate to 3 percent. In addition, FORBID always exhibits obvious advantage over the other schemes when the percentage of Byzantine nodes increases up to 90 percent.

Next we change the network size. The result is shown in Fig. 4b. It is observed that all schemes suffer from the increase of network size. This is mainly because larger networks contain more Byzantine nodes and asymmetric links. They also introduce higher latency to operations like information exchange, and thus delay the detection of malicious behaviors. Although its performance is slightly degraded by the increase of network size, FORBID generally maintains a fairly stable curve.

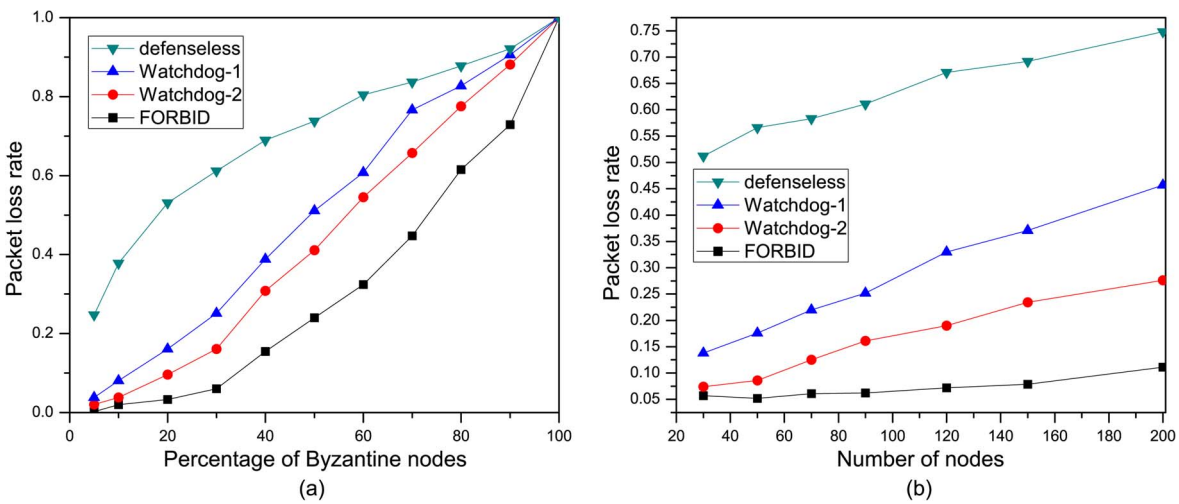


Fig. 4. Packet loss rate with different network parameters: (a) percentage of Byzantine nodes, (b) network size.

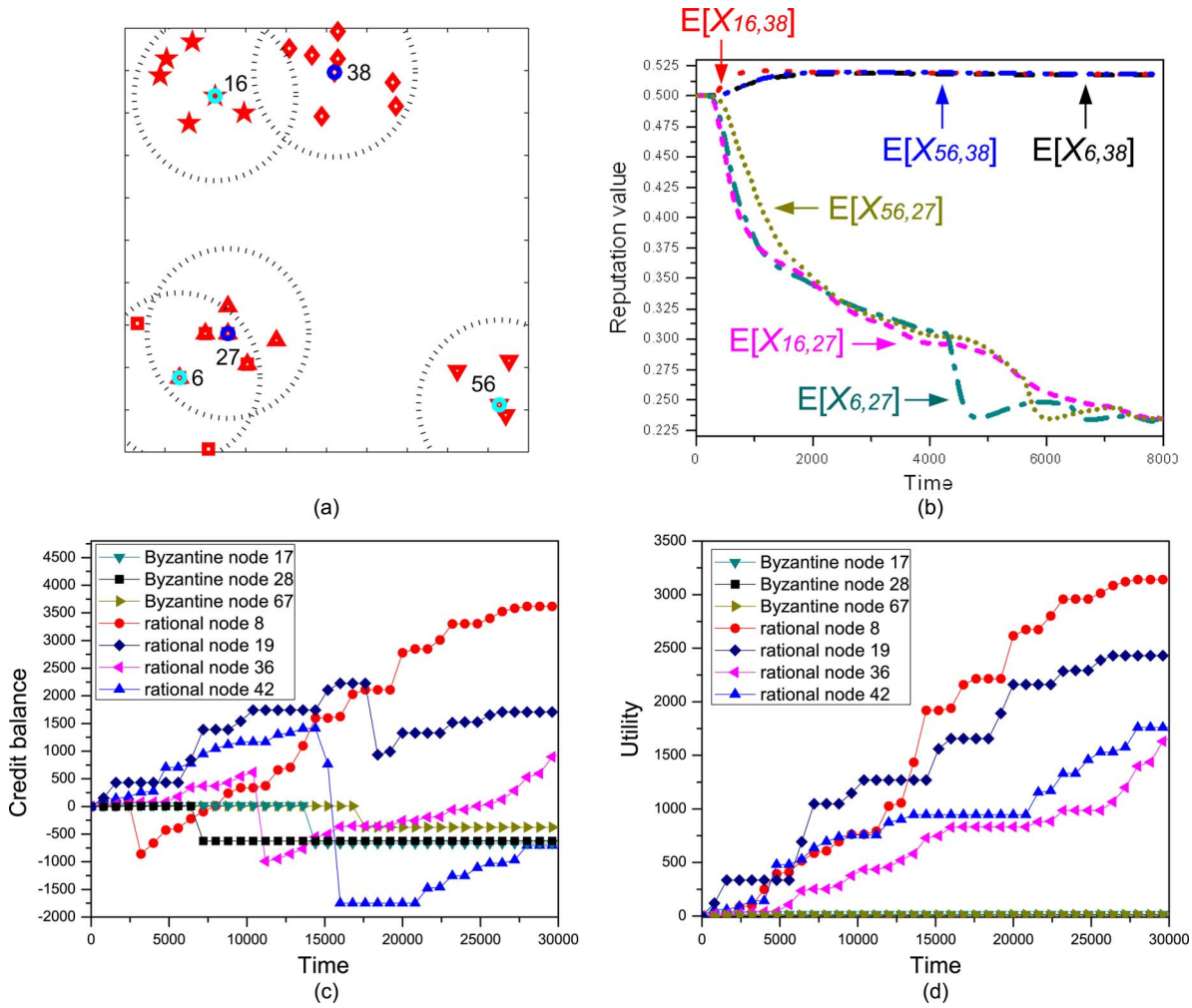


Fig. 5. Byzantine vs. rational nodes on: (a) location, (b) reputation evolution, (c) credit balance, (d) utility.

5.3 Rational Nodes vs. Byzantine Nodes

How the protocol distinctly treats rational and Byzantine behaviors is investigated here. We randomly pick a rational node 27 and a Byzantine node 38 as the target nodes, and keep track of their reputation values on the reference nodes 6, 16, and 56 placed on disparate network locations. Their locations are depicted in Fig. 5a. The dashed circles indicate the transmission ranges of the selected nodes placed at the center. The locations of nodes within their transmission ranges are also plotted. Reference node 16 is near Byzantine node 38.

The reputation values of the two target nodes at the three reference nodes are shown in Fig. 5b. For example, $\mathbf{E}[X_{6,38}]$ describes the belief of node 6 on how likely node 38 is Byzantine. Initially, both nodes 38 and 27 are treated neutrally. Over time, the rational node 27 gains better reputation (small value) among other nodes. The reason is that by forwarding packets for other nodes, it gains better reputation and in turn gains more chance of being chosen by the routing protocol due to its good reputation. On the other hand, the reputation values for the Byzantine node 38 quickly increases over time. It is successfully identified by all three reference nodes when its reputation value stabilizes above the threshold. As expected, node 16 is the first one to identify node 38 as Byzantine due to the shortest distance between the two.

Fig. 5c shows the credit balance. Rational nodes participate in the routing and forwarding games, their credit balances vary according to how much they have paid to other nodes and how much they have received from other nodes. On the other hand, the credit balances of Byzantine nodes are non-increasing in the long run, due to the fact that they are effectively isolated by the protocol. Similarly, Fig. 5d confirms that only cooperating nodes have their utilities increased over time.

5.4 Impact of System Parameters on Performance

We have also carried out some experiments to study the impact of $w_t(i,i)$, T and the percentage of Byzantine nodes on the performance of our protocol. Due to space limit, such results are reported in the supplementary material available online.

6 Related Work

Distributed algorithmic mechanism design is a recent branch of algorithmic mechanism design into the distributed computing area [8], where routing and forwarding in

non-cooperative networks is an important problem of interest. A mechanism design for lowest-cost unicast routing in Internet that is built on top of BGP routing protocol was proposed by Feigenbaum *et al.* [7]. Feldman *et al.* demonstrated how an appropriate mechanism design could overcome certain hidden action problems in distributed multi-hop networks [9]. This design builds up contracts directly between endpoints and each intermediate router, as well as recursively between each intermediate router and its next hop. Such examples have demonstrated the power of combining economics concepts and cryptographic techniques with distributed routing protocols.

One common approach for handling routing misbehavior is to incentivize nodes for cooperation. Buttyan *et al.* proposed to use a per-hop payment carried in each packet called nuglets, to serve as incentives for packet forwarding. Following that, the authors proposed another form of incentives called counters to complement the design of nuglets [5]. Both schemes are limited by the requirement that a special secure hardware device is deployed at each node, and thus cannot be easily extended to more general networks. Zhong *et al.* proposed a credit-based system that does not require tamper-proof hardware at each node for credit maintenance [28]. Anderegg *et al.* proposed ad hoc-VCG auction to calculate proper payment for packet forwarding [1]. Combining VCG with a cryptographic technique, Zhong *et al.* proposed an incentive-compatible solution that corresponds to a relaxation of a dominant-action solution [29]. The VCG mechanism was also used for multi-path routing [25], [26]. With the strengths such as strategy-proofness and ex-post efficiency, VCG suffers from the overpayment problem [11], [21]. Wang *et al.* proposed the OURS [24] protocol for unicast routing systems. Instead of relying on a variant of VCG mechanism, OURS is built based on the concept of Nash equilibria.

Another popular approach to deal with routing misbehavior is to establish punishment against non-cooperation. Marti *et al.* proposed a reputation system consisting of Watchdog and Pathrater, to identify and avoid Byzantine nodes in [14]. Watchdog relies on passive overhearing to detect denied packet forwarding, and then Pathrater rates every path based such detections. Limitations include unreliable overhearing due to asymmetric links and the fact that Byzantine nodes are avoided rather than isolated. Buchegger *et al.* proposed CONFIDANT that updates reputation record via experience, observation, and report from friends [2]. A path manager is used by CONFIDANT for path selection based on available reputation record. In a followup work [3], the authors proposed a Bayesian approach to improve the representation and update of reputation records. Limitations are that how to identify and select friends is not clear, and that the proposed protocols may fail in the case with false accusation. Michiardi *et al.* proposed CORE to complement Watchdog with a sophisticated reputation mechanism [15]. This mechanism distinguishes subjective reputation by Watchdog observation, from indirect reputation reported by others, and functional reputation specified by task-specific behavior. Similar to the mechanism in [14], CORE suffers from Watchdog's reliability issue due to link asymmetry. This method also tends to overestimate the selfish behavior and treat packet loss due to temporary link failure or interference as intentional forwarding denial. Rodriguez *et al.* proposed some refinements to improve the reliability of Watchdog [17]. Instead of being punished immediately, a node caught misbehaving is put into a warning mode so that it has a chance to correct its behavior. If such misbehavior is later identified to be caused by link failure, the node's reputation is not degraded. Yang *et al.* designed a Dirichlet reputation system to distinguish good, selfish and malicious nodes [27]. Reputation record is updated based on a node's direct observations and reports on first-hand observations from immediate neighbors. A potential issue is that besides malicious nodes, the path selection scheme may also avoid rational nodes, which is contrary to the belief that better performance can be obtained if rational nodes are incentivized to cooperate.

7 Conclusion

Due to the co-existence of both rational and Byzantine misbehaviros, designing multi-path routing and forwarding protocols for non-cooperative networks is pretty interesting and challenging. In this paper, we have presented a unified framework that includes two schemes from our previous work to distinctly handle the two categories of misbehavior. The GSP auction mechanism enhanced with traffic allocation polices incentivizes rational nodes to cooperate. The FORBID mechanism actively detects malicious violations from the norm and isolates such activities from the packet forwarding paths.

We have formally proved that the interaction between GSP and FORBID in the unified framework leads to a cooperation-optimal routing protocol. Rational nodes bid with their true costs and maximize their utilities by participating in packet forwarding, while Byzantine nodes cannot disrupt the normal operation of the network. We complement our theoretical analysis with extensive experiments for better understanding the behavior of the protocol. The simulation results have confirmed that rational nodes gain better utility and credit balance by following the protocol. We have also observed that Byzantine nodes are quickly detected and isolated from the forwarding paths, leading to other network performance improvement such as lower packet loss rate.

Acknowledgment

This work was completed while X. Su was with the Department of Computer Science, Yale University.

References

[1] L. Anderegg and S. Eidenbenz, ''Ad Hoc-VCG: A Truthful and Cost-Efficient Routing Protocol for Mobile Ad Hoc Networks with Selfish Agents,'' in *Proc. MobiCom*, 2003, pp. 245-259.

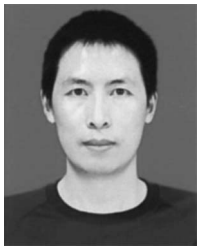
[2] S. Buchegger and J. Le Boudec, ''Performance Analysis of the CONFIDANT Protocol,'' in *Proc. MobiHoc*, 2002, pp. 226-236.

[3] S. Buchegger and J. Le Boudec, ''The Effect of Rumor Spreading in Reputation Systems for Mobile Ad-Hoc Networks,'' in *Proc. Workshop Model. Optim. Mobile, Ad Hoc Wireless Netw.*, 2003, pp. 131-140.

[4] L. Buttyán and J. Hubaux, ''Enforcing Service Availability in Mobile Ad-Hoc WANs,'' in *Proc. ACM MobiHoc*, 2000, pp. 87-96.

[5] L. Buttyán and J. Hubaux, ''Stimulating Cooperation in Self-Organizing Mobile Ad Hoc Networks,'' *Mobile Netw. Appl.*, vol. 8, no. 5, pp. 579-592, Oct. 2003.
[6] B. Edelman, M. Ostrovsky, M. Schwarz, T.D. Fudenberg, L. Kaplow, R. Lee, P. Milgrom, M. Niederle, and A. Pakes, ''Internet Advertising and the Generalized Second Price Auction: Selling Billions of Dollars Worth of Keywords,'' *Amer. Econ. Rev.*, vol. 97, no. 1, pp. 242-259, Mar. 2005.
[7] J. Feigenbaum, C. Papadimitriou, R. Sami, and S. Shenker, ''A BGP-Based Mechanism for Lowest-Cost Routing,'' *Distrib. Comput.*, vol. 18, no. 1, pp. 61-72, July 2005.
[8] J. Feigenbaum and S. Shenker, ''Distributed Algorithmic Mechanism Design: Recent Results and Future Directions,'' in *Proc. 6th Int'l Workshop Discr. Algorithms Methods Mobile Comput. Commun.*, 2002, pp. 1-13.
[9] M. Feldman, J. Chuang, I. Stoica, and S. Shenker, ''Hidden-Action in Multi-Hop Routing,'' in *Proc. 6th ACM Conf. Electron. Commerce*, 2005, pp. 117-126.
[10] D. He, C. Chen, S. Chan, J. Bu, and P. Zhang, ''Secure and Lightweight Network Admission and Transmission Protocol for Body Sensor Networks,'' *IEEE J. Biomed. Health Inf.*, vol. 17, no. 3, pp. 664-674, May 2013.
[11] D. Karger and E. Nikolova, ''On the Expected VCG Overpayment in Large Networks,'' in *Proc. 45th IEEE Conf. Decision Control*, Dec. 2006, pp. 2831-2836.
[12] L. Lamport, R. Shostak, and M. Pease, ''The Byzantine Generals Problem,'' *ACM Trans. Programm. Lang. Syst.*, vol. 4, no. 3, pp. 382-401, July 1982.
[13] Y. Li, ''Toward a Qualitative Search Engine,'' *IEEE Internet Comput.*, vol. 2, no. 4, pp. 24-29, July/Aug. 1998.
[14] S. Marti, T. Giuli, K. Lai, and M. Baker, ''Mitigating Routing Misbehavior in Mobile Ad Hoc Networks,'' in *Proc. ACM MobiCom*, 2000, pp. 255-265.
[15] P. Michiardi and R. Molva, ''Core: A Collaborative Reputation Mechanism to Enforce Node Cooperation in Mobile Ad Hoc Networks,'' in *Proc. Adv. Commun. Multimedia Security, IFIP TC6/TC11 6th Joint Working Conf. Commun. Multimedia Security*, 2002, pp. 107-121.
[16] M. Osborne and A. Rubinstein, *A Course in Game Theory*. Cambridge, MA, USA: MIT Press, 1994.
[17] A. Rodriguez-Mayol and J. Gozalvez, ''Improving Selfishness Detection in Reputation Protocols for Cooperative Mobile Ad-Hoc Networks,'' in *Proc. IEEE PIMRC*, 2010, pp. 26-29.
[18] X. Su, S. Chan, and G. Peng, ''Auction in Multi-Path Multi-Hop Routing,'' *IEEE Commun. Lett.*, vol. 13, no. 2, pp. 154-156, Feb. 2009.
[19] X. Su, S. Chan, and G. Peng, ''Generalized Second Price Auction in Multi-Path Routing with Selfish Nodes,'' in *Proc. IEEE GLOBECOM*, 2009, pp. 3413-3418.
[20] X. Su, G. Peng, and S. Chan, ''FORBID: Cope with Byzantine Behaviors in Wireless Multi-Path Routing and Forwarding,'' in *Proc. IEEE GLOBECOM*, 2011, pp. 1-6.
[21] K. Talwar, ''The Price of Truth: Frugality in Truthful Mechanisms,'' in *Proc. 20th Annu. Symp. Theor. Aspects Comput. Sci.*, 2003, pp. 608-619.
[22] S. Tomasin, ''Consensus-Based Detection of Malicious Nodes in Cooperative Wireless Networks,'' *IEEE Commun. Lett.*, vol. 15, no. 4, pp. 404-406, Apr. 2011.
[23] H.R. Varian, ''Position Auctions,'' *Int'l J. Ind. Org.*, vol. 25, no. 6, pp. 1163-1178, Dec. 2007.
[24] W. Wang, S. Eidenbenz, Y. Wang, and X.-Y. Li, ''OURS: Optimal Unicast Routing Systems in Non-Cooperative Wireless Networks,'' in *Proc. MobiCom*, 2006, pp. 402-413.
[25] Y. Wang, V.C. Giruka, and M. Singhal, ''Truthful Multipath Routing for Ad Hoc Networks With Selfish Nodes,'' *J. Parallel Distrib. Comput.*, vol. 68, no. 6, pp. 778-789, June 2008.
[26] F. Wu, S. Zhong, and J. Liu, ''Cost-Effective Traffic Assignment for Multipath Routing in Selfish Networks,'' in *Proc. IEEE GLOBECOM*, 2007, pp. 453-457.
[27] L. Yang, A. Cemerlic, and X. Cui, ''A Dirichlet Reputation System in Reliable Routing of Wireless Ad Hoc Network,'' *Security Commun. Netw.*, vol. 3, no. 2/3, pp. 250-260, Mar.-June 2010.
[28] S. Zhong, J. Chen, and Y. Yang, ''Sprite: A Simple, Cheat-Proof, Credit-Based System for Mobile Ad-Hoc Networks,'' in *Proc. IEEE INFOCOM*, 2003, pp. 1987-1997.
[29] S. Zhong, L. Li, Y. Liu, and Y. Yang, ''On Designing Incentive-Compatible Routing and Forwarding Protocols in Wireless Ad-Hoc Networks: An Integrated Approach Using Game Theoretic and Cryptographic Techniques,'' *Wireless Netw.*, vol. 13, no. 6, pp. 799-816, Dec. 2007.

Xueyuan Su received the BE degree in communication and information engineering from the University of Electronic Science and Technology of China, in 2005, and the MPhil degree in electronic engineering from the City University of Hong Kong, in 2007, and the PhD degree in computer science from Yale University, in 2013. Since September 2013, he has been with Oracle Corporation, where he is currently a Senior Member of Technical Staff. His research interests include big data, parallel and distributed computing, and algorithm design and analysis.



Gang Peng received the BSc and MEng degrees from Huazhong Normal University, Wuhan, China, in 1998 and 2006, respectively, and the PhD degree from the City University of Hong Kong, Hong Kong, in 2012. His research interests include wireless ad hoc network, reputation system and complex networks.

Sammy Chan received the BE and MEngSc degrees in electrical engineering from the University of Melbourne, Australia, in 1988 and 1990, respectively, and the PhD degree in communication engineering from the Royal Melbourne Institute of Technology, Australia, in 1995. From 1989 to 1994, he was with Telecom Australia Research Laboratories, first as a research engineer, and between 1992 and 1994 as a senior research engineer and project leader. Since December 1994, he has been with the Department of Electronic Engineering, City University of Hong Kong, where he is currently an Associate Professor. He is a member of the IEEE.