

Hidden Information and Actions in Multi-Hop Wireless Ad Hoc Networks

Xiang-Yang Li*, YanWei Wu†, Ping Xu‡, GuiHai Chen‡, Mo Li§

ABSTRACT

For multi-hop ad hoc networks formed by individually owned nodes, the endpoints can only observe whether or not the end-to-end transaction was successful or not, but not the individual actions of intermediate nodes. Consequently, in the absence of properly designed incentive schemes, rational (*i.e.*, selfish) intermediate nodes may choose to forward data packets at a very low priority or simply drop the packets at all, and it could put the blame on the unreliable wireless channel. Using a principal-agent model, we propose several efficient methods that can eliminate the hidden actions under hidden information in multi-hop wireless networks with high probability. We design several algorithmic mechanisms for a number of routing scenarios such that each selfish agent will maximize its utility (*i.e.*, profit) when it *truthfully* declares its *type* (*i.e.*, cost and its actions) and it truthfully follows its declared actions. Our simulations show that the payment by our mechanisms is only slightly larger than the *actual cost* incurred by all intermediate nodes.

Categories and Subject Descriptors

C.2.1 [Network Architecture and Design]: Wireless communication; C.2.2 [Network Protocols]: Routing Protocols; G.2.2 [Graph Theory]: Network problems, Graph algorithms

*The author is with State Key Laboratory of Novel Software Technology, NanJing University, NanJing, P.R.China, also with Department of Computer Science, Illinois Institute of Technology, Chicago, IL, USA. Partial of the work was done when the author visited Microsoft Research Asia. The research of the author was partially supported by National Basic Research Program of China (973 Program) under grant No. 2006CB30300, the National High Technology Research and Development Program of China (863 Program) under grant No. 2007AA01Z180, the RGC under Grant HKBU 2104/06E and CERG under Grant PolyU-5232/07E. Email: xli@cs.iit.edu.

†Department of Computer Science, Illinois Institute of Technology, Chicago, IL, USA. Email: {ywu24,pxu3}@iit.edu

‡State Key Laboratory of Novel Software Technology, NanJing University, NanJing, P.R.China. Guihai Chen's work is partly supported by China NSF grants (60573131,60673154,60721002) and Jiangsu High-Tech Research Project of China (BG2007039). Email: gchen@nju.edu.cn

§Department of Computer Science & Engineering, Hong Kong University of Science and Technology. Email: limo@cse.ust.hk

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MobiHoc'08, May 26–30, 2008, Hong Kong SAR, China.
Copyright 2008 ACM 978-1-60558-083-9/08/05 ...\$5.00.

General Terms

Algorithms, Design, Economics, Theory

Keywords

Ad hoc networks, hidden action, hidden-information, selfish, truthful mechanism.

1. INTRODUCTION

In multi-hop wireless ad hoc networks, it is commonly assumed that, each terminal offers its local resources to forward the data for other terminals to serve the common good, and benefits from resources offered by other terminals to route its packets in return. However, the limitation of energy supply, memory and computing resources of these individual devices raise concerns on this traditional assumption. A networking device owned by an individual user may prefer not participating in the routing to save its energy and resources. Thus, if we assume that all users are selfish, incentives must be provided to encourage their participation and thus maintain the robustness and availability of networking systems.

The question turns to how the incentives are designed. Assume that each intermediate wireless node i will incur a marginal cost $c(i, j)$ if it is asked to forward a *unit* amount of data to a neighboring node j . Here we assume that the cost $c(i, j)$ is a *private* information known *only* by node i (some studies, *e.g.* [26], assumed that it is determined by both i and j if $c(i, j)$ only depends on transmission power). This private information is called its *type*. Consider a unicast routing and forwarding protocol based on the least cost path (LCP): each terminal is asked to declare a cost to forward a unit amount of data for other terminals, and the least cost path connecting the source and the target terminal is then selected. A very naive incentive is to pay each terminal its declared cost. However, the individual terminal may declare an arbitrarily high cost for forwarding to increase its payment. Then the “least cost path” selected based on the *falsely* declared cost information may be different from the actual least cost path computed based on the *truthful* cost information of all individual nodes. We call this as the **hidden information game**: the information needed to find the best output is hidden from the decision-maker. Here, we would like to design a payment scheme such that every terminal will report its cost *truthfully* and always forward others' traffic out of its own interest to maximize its profit. This payment scheme is called *strategyproof* since it removes speculation and counter-speculation among terminals.

In multi-hop networks with selfish nodes, two fundamental issues *routing* and *packet forwarding* must be addressed to build a complete system [26]. A number of mechanisms has been designed in the literature (*e.g.*, [6, 22, 24]) for soliciting the *truthful*

cost declaration from selfish agents so that a certain optimal routing structure could be built to connect the source node and destination node(s). On the other hand, several results (e.g., [9, 11, 19, 25]) concentrated on ensuring that intermediate nodes will indeed forward the data packets. Unfortunately, Zhong and Li, *et al.*, [26] elegantly showed that no dominant strategy solution exists in which every node always forwards others' packets.

The essential argument in [26] shows that no dominant strategy exists for forwarding is that the action of each intermediate node is hidden from others. When a node u dropped certain packets, it is difficult (if not impossible) for other nodes to distinguish whether this node u intentionally dropped the packets or the packets were lost simply due to the unreliability of channels (e.g., noise, interference from other nodes) although node u did send the packets. We call this kind of game as **hidden action game** where the endpoints can only observe whether or not the packet has reached the destination but cannot attribute failure to a specific node on the path. Notice that, in multi-hop wireless ad hoc networks, even if some monitoring mechanisms are in place to allow the sender or the receiver to pinpoint the location of the failure, they may still be unable to attribute the cause of failure to either the deliberate action of the intermediate node, or some external factors beyond the control of the intermediate node, such as network congestion, channel interference, or data corruption. Observe that we implicitly assume that the link failures are independent among different links. We want to design protocols that can eliminate the hidden action without using additional monitoring scheme. Notice that when the failures are not totally independent, it may be possible for some agents (a wireless node in wireless ad hoc networks) to tell whether a failure is due to natural hazard, or due to intentionally dropping data by a node. Our protocols will remove the intentional drop using a simple payment scheme without relying on any monitoring scheme. The problem of hidden action is also known as *moral hazard*, which has long been of interest in the economics literature.

In the paper, we will focus on designing cost-efficient routing protocols under both hidden information and hidden action. Using the principal-agent model, we propose several efficient methods that can induce truthfulness and eliminate the hidden actions in non-cooperative multi-hop networks with high probability, *i.e.*, each selfish agent will maximize its profit when it *truthfully* declares its cost and act truthfully following its declared actions. Compared with the closely related results [8, 26], the main contributions of this paper are as follows. (1) We assume the principal has a finite valuation to get the data from the source to the target, or has an infinite valuation instead. We show that, when principal has a finite valuation, a simple extension of previous methods will *not* induce truthfulness from agents. We then design strategyproof routing schemes for this scenario. (2) Besides strategyproof schemes, we design efficient routing schemes with efficient Nash Equilibrium and show how to find the maximize Nash Equilibrium efficiently. (3) We take into account the Quality of Service when designing routing schemes and show that following the declared QoS is the optimal strategy for each agent under our schemes. Moreover, we integrate the hidden-action into the QoS. (4) We design the first strategyproof multi-path routing by taking account the link capacity and interference. Since it is NP-hard to find the multi-path routing with minimum cost while flows are schedulable, we relax the problem and design a strategyproof multi-path routing scheme where the allocated flows are guaranteed to be schedulable and the routing cost is within a constant factor of the optimum. (5) We assume that a selfish agent could manipulate multiple parameters such as its service cost and service reliability. Traditionally, an agent can only manipulate only one parameter. (6) We focus on link reliability

model. Strategyproof routing schemes are designed for almost all possible combinations of reliability models, one/multi-parameter models, and valuation models. Our mechanisms can deal with both hidden-information and hidden-action and they can be easily integrated with existing routing protocols. We also conduct extensive simulations to study the overhead of our proposed mechanisms compared with the ideal situation when all intermediate nodes are cooperative. Our simulations show that the payment by our mechanisms is only slightly larger than the *actual cost* incurred by all intermediate nodes.

The rest of the paper is organized as follows. In Section 2, we present our network model, and the problems to be studied. In Section 3, we present our methods that can efficiently deal with the hidden actions with known information, and in Section 4 we focus on scenario with hidden information and actions. Section 5 studies multi-path routing. We report our simulation results in Section 6 and we review the related works in Section 7. We conclude our paper in Section 8 with discussion of future researches.

2. NETWORK MODEL AND ROUTING PROTOCOL DEFINITION

2.1 Network Model

We assume that there is a set V of $n = |V|$ of devices (called nodes hereafter) distributed in a region. For simplicity, for the majority results presented in this paper, we assume that the nodes are static or can be viewed as static (its movement will not change the network structure) for a long-enough time period, and each node is assigned a unique ID $i \in [1, n]$. The multi-hop network is modeled by a directed communication graph $G = (V, E)$, where E is the set of $m = |E|$ directed links. We always assume that the nodes are selfish agents: they will take the actions to maximize their own benefits. We consider a principal-agent model, where the principal is a pair of communication endpoints (*i.e.*, (s, d)) who wish to communicate over a multi-hop network, and the agents are the intermediate nodes capable of forwarding packets between the endpoints. In this paper, we will not distinguish the difference of principals here. The principal will get a value $v(s, d)$ if a unit amount of data reaches the destination. If it has to pay χ amount of monetary value to intermediate relay agents, then its pure profit (*i.e.*, utility) will be $v(s, d) - \chi$. Obviously, the principal will conduct routing only if $v(s, d) - \chi \geq 0$. The principal may also have a minimum QoS $\theta(s, d)$ for the routing path. Here the QoS provided by a path is often defined as the minimum QoS provided by all nodes on the path. Notice that a node may be principal node for one routing session and be forwarding node in other routing sessions. We assume that all nodes could lie about their information when they are queried.

We assume that when an individual node u is asked to forward *once* a unit amount of data to a neighboring node v , node u will incur a certain amount of marginal cost. Notice that in wireless networks even node u has sent signal to node v , node v may still not be able to decode the signal correctly due to noise and interference. We use $0 \leq \beta(u, v) \leq 1$ to denote the *reliability* of the link (u, v) . In other words, node v will receive the data correctly with probability $\beta(u, v)$ after node u sends data to node v . In wireless networks, whether node v received the data packets correctly or not is only known to node v itself. When the information $\beta(u, v)$ is only observable by node v , we call it the *private information* of node v , *i.e.*, it is part of *type* of node v . Node v can choose to declare this private information correctly or wrongfully, depending on which will maximize its own utility. In practice, the physical layer unreliability is often caused by the Gaussian noise, which is predictable

by the system or observable by many nodes if certain monitoring mechanism is implemented, thus, $\beta(u, v)$ is viewed as *public information* in this scenario. In this paper, most of our studies will view this link reliability as public information. Notice that this link reliability is different from the service reliability on this link, although in practice it is difficult to separate them in non-cooperative networks with selfish nodes. We also point out here that, in practice, the cost and reliability are not fixed value; instead they depend on a number of factors such as interference. Here we will use the *estimated* expected value as the value of these variables.

In addition, the cost incurred by a node u may also depend on the QoS which is provided in sending packets to v . We assume that *all agents* can provide a given set $\mathcal{Q} = \{\theta_1, \theta_2, \dots, \theta_\kappa\}$ of κ quality of forwarding services. For simplicity, we may assume that the QoS θ_i is better than θ_j for $i > j$. We also assume that *drop packets always* is the worst QoS θ_1 . For example, the services could be simply $\{\text{always drop}, \text{always forward}\}$. The QoS provided by node u will effect the probability that node v receives the data packets from u correctly. We use $\alpha_j(u, v)$ to denote the probability when node v will receive the data from u correctly when node u provides forwarding service θ_j to node v , assuming that the physical link has a perfect reliability for the moment. We call $\alpha_j(u, v)$ as *service reliability*. The service reliability could be public information. For example, when the available services are $\{\theta_1 = \text{always drop}, \theta_2 = \text{always forward}\}$, we have $\alpha_1(u, v) = 0$ and $\alpha_2(u, v) = 1$. The service reliability could also be *private information* which is controlled by the sending node u . This often happens when there are several QoSs in \mathcal{Q} . For convenience, we denote $\delta(u, v) = \alpha_j(u, v) \cdot \beta(u, v)$ as the value of the corresponding service from node u to node v .

For each forwarding service θ_j provided by a node u to forward a unit amount of data to node v , node u will incur a certain *service cost* $\mathbf{c}_j(u, v)$. There are two possible cases here: (1) the cost $\mathbf{c}_j(u, v)$ is independent of the neighboring node v ; (2) the cost $\mathbf{c}_j(u, v)$ is dependent of the neighboring node v . For simplicity, we always use the general $\mathbf{c}_j(u, v)$, which could be same regardless of the node v .

In summary, depending on applications and whether certain monitoring schemes are implemented, a selfish agent v_i may have all (or part) of the following as its *type* t_i : (1) the cost $\mathbf{c}_j(v_i, w)$ to provide a forwarding service to every outgoing neighboring node w with quality of service θ_j , (2) the corresponding service reliability $\alpha_j(v_i, w)$. We often assume that the physical link reliability $\beta(u, v_i)$ is a public information. The action of an agent in routing subgame is to declare its cost for providing forwarding with certain implicitly agreed quality of service θ_r (corresponding to its declared cost). The action θ' in the forwarding subgame is to provide forwarding with certain quality of service, which may be different from θ_r . A routing mechanism is called *strategyproof* if every agent will maximize its utility when it declares its cost truthfully and fulfills its forwarding service with QoS θ_r .

2.2 Problem Specification

Assume that we want to implement a routing protocol that will route data from a source node \mathbf{s} to a target node \mathbf{d} . The routing protocol with selfish intermediate nodes will have the following components.

1. Collect Information: The principal (could be sender \mathbf{s} , or receiver \mathbf{d} or both) first asks every node u to declare what QoS θ_r it will provide for forwarding, the cost $\bar{\mathbf{c}}(u, v)$ for forwarding a unit amount of data to a neighboring node v by this QoS θ_r , the corresponding expected service reliability $\bar{\alpha}_r(u, v)$, and the observed physical reliability $\bar{\beta}(w, u)$ for all possible incoming neighbors w .

2. Select Path: This is called the *routing subgame* in [26]. Based on the declared information collected from all nodes, the principal then has to find a path to connect the source and the target, which has a certain minimum QoS, *i.e.*, every node on the path can provide at least this minimum QoS θ_r . Principal also computes a certain incentive given to each intermediate node on the path to compensate the cost incurred by forwarding.

3. Forward Packets: The principal then asks the nodes on the chosen path to forward the data if certain conditions (which are protocol dependent) are met. Assume that a simple path $v_{i_1} v_{i_2} \dots v_{i_h}$ is used for routing where $\mathbf{s} = v_{i_1}$, $\mathbf{d} = v_{i_h}$ and direct links $v_{i_j} v_{i_{j+1}}$, $1 \leq j \leq h - 1$, belong to the network G . Then node v_{i_j} is *asked* to forward the data packets to node $v_{i_{j+1}}$ (implicitly under QoS θ , where θ corresponding to the QoS θ_r that node v_{i_j} would provide when node v_{i_j} declared its cost $\bar{\mathbf{c}}(v_{i_j}, v_{i_{j+1}})$). A key observation is that an intermediate node v_{i_j} may choose to forward the data using some QoS $\theta' \in \mathcal{Q}$ other than its initial intention θ_r . Notice that we treat *drop packets always* as one of the possible quality of services here. How to ensure that each intermediate node will forward the data packets using its initial intention is called the *forwarding subgame*, which generalizes the forwarding subgame defined in [26]. In [26], Zhong *et al.* essentially assume that there are two quality of services: $\{\text{drop}, \text{forward}\}$.

4. Materialize Payment: Depending on the outcome of the forwarding game, the principal would materialize the incentive computed during the routing subgame, *e.g.*, transfer the monetary value to intermediate nodes if a monetary payment scheme is used to induce the cooperation from these nodes.

In this paper, we will focus on UDP routing using the above general framework: if the transmission from a node v_{i_j} to node $v_{i_{j+1}}$ is not successful, node v_{i_j} is required to resend the data till node $v_{i_{j+1}}$ successfully receives the data. The goal of each intermediate node is to maximize its profit. We want to design a *truthful* routing mechanism and payment scheme such that each intermediate node will maximize its *expected* profit if it truthfully declared its type and followed the routing. A scheme is truthful if it satisfies:

1. *Incentive Compatibility with Expectation (ICE)*: Revealing true type is a dominant strategy;
2. *Individual Rationality with Expectation (IRE)*: Each agent has a non-negative *expected* utility;
3. *Polynomial Time Computability (PC)*: routing path and payment are computed in polynomial time.

Besides dominant strategy design, a number of implementation concepts have been proposed, *e.g.*, Nash design. For forwarding subgame, we want the truthful forwarding to be a subgame perfect equilibrium. A vector of action profile is a *Nash Equilibrium* (resp. ϵ -Nash Equilibrium) if no agent i can improve its own utility (reps. by more than ϵ) by unilaterally changing its action a_i to some other action a'_i when the actions of all other agents a_{-i} are fixed. Subgame perfect equilibrium (SPE) is a sequence of actions by players such that players' strategies constitute a Nash equilibrium in every subgame of the original game. It can be found using *backward induction* (see *e.g.* [17]).

3. KNOWN INFORMATION AND HIDDEN ACTION

We first assume that all information about the selfish nodes are already known by the principals. Let $\theta_r = \theta(\mathbf{s}, \mathbf{d})$ be the minimum QoS required by principals. Thus, the only action taken by a selfish node is (1) declaring whether it can provide such service, and (2) providing what kind of forwarding service actually. The main goal of the routing scheme (composed of routing subgame and for-

Algorithm 1 Link Layer Rel-Routing with Known Information

Input: graph G , QoS θ_r required by principal and its valuation $\nu(\mathbf{s}, \mathbf{d})$ for a unit amount of data, and a fixed parameter $\eta \geq 0$.

- 1: **Routing Subgame:** First the principal asks every node u , whether it can provide a forwarding service θ_r on link (u, v) for each one of its out-going neighbors v . We remove the node w and all the incident links (w, v) from G where node w replied that it cannot provide such forwarding service θ_r required by the principal. For each remaining link (u, v) , we define its weight as $\mathbf{c}_r(u, v) / (\alpha_r(u, v) \cdot \beta(u, v))$. Let G' be the resulted graph. Let $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d}) = v_{i_1} v_{i_2} \cdots v_{i_h}$ (where $\mathbf{s} = v_{i_1}$, $\mathbf{d} = v_{i_h}$) be the least cost path from G' to connect \mathbf{s} and \mathbf{d} and $\mathcal{E}(\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d}))$ be its expected cost. Notice that, since the link reliability is a variable, the actual cost of a path is also a variable; and $\mathcal{E}(\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d}))$ is the expected cost of a path, which can be computed easily if known reliability of links (or nodes). If $(1 + \eta) \cdot \mathcal{E}(\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})) \leq \nu(\mathbf{s}, \mathbf{d})$, the principal will decide to conduct the routing, and the output \mathbf{o} of the routing subgame is $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})$. Otherwise, the principal will not initiate the routing, and the output \mathbf{o} is \emptyset . For each node not on the final path, its payment is always 0. For each node u selected on the output \mathbf{o} , its payment is

$$\frac{\mathbf{c}_r(u, v)}{\alpha_r(u, v) \cdot \beta(u, v)} (1 + \eta)$$

where v is the next-hop node of u . This payment will not be materialized until the packet forwarding subgame is finished.

- 2: **Forwarding subgame:** When an intermediate node v_{i_j} ($2 \leq j \leq h - 1$) received a data packet by the principal, it will forward the packet to the next-hop node $v_{i_{j+1}}$ using QoS θ_r , or using some other forwarding QoS. We will prove that, to maximize its expected benefit, the intermediate node will keep forwarding the data till it is correctly received by next-hop node.

The principal materializes the payment to every intermediate agent v_{i_j} only if the target node \mathbf{d} received the data correctly with the given QoS. In other words, if any node on the path lowers its forwarding service QoS, every node on the path will *not* receive any payment.

warding subgame) is then to ensure that every selfish agent fulfill its declared forwarding service. Our routing scheme is designed as Algorithm 1.

THEOREM 1. *If all intermediate nodes forward N units amount of data truthfully, the probability that a forwarding node u suffers loss is at most $\frac{1-\delta}{1-\delta+N\cdot\eta\cdot\delta}$ where $\delta = \alpha_r(u, v) \cdot \beta(u, v) \leq 1$. The probability of any node on a path of h -hops suffers loss is no more than $\frac{h(1-\delta)}{1-\delta+N\cdot\eta\cdot\delta}$.*

PROOF. Assume that there are N units of data transmitted from \mathbf{s} to \mathbf{d} . For a specific intermediate node u , let U_k be the variable denoting the cost for transmitting the k th unit of data to its next-hop neighbor v , and $\delta = \alpha_r(u, v) \cdot \beta(u, v)$. Then $\mathcal{E}(U_k) = \mathbf{c}_r(u, v) / \delta$ and the standard deviation is $\sigma(U_k) = \mathbf{c}_r(u, v) \sqrt{(1-\delta) / \delta}$. Consequently, the central limit theorem implies that $U = \frac{\sum_{k=1}^N U_k}{N \cdot \mathbf{c}_r(u, v)}$ is a random variable with mean $\frac{1}{\delta}$ and standard deviation $\sigma(U) = \sigma(U_k) / (\sqrt{N} \cdot \mathbf{c}_r(u, v)) = \sqrt{\frac{1-\delta}{N \cdot \delta}}$. Then we have

$$\begin{aligned} \Pr\left(\sum_{k=1}^N U_k \geq (1 + \eta)N \cdot \frac{\mathbf{c}_r(u, v)}{\delta}\right) &= \Pr\left(U - \frac{1}{\delta} \geq \eta \cdot \frac{1}{\delta}\right) \\ &\leq \frac{\sigma(U)^2}{\sigma(U)^2 + (\frac{\eta}{\delta})^2} = \frac{1 - \delta}{1 - \delta + N \cdot \eta \cdot \delta} \end{aligned}$$

This is from one-sided Chebyshev Inequality: For a random variable X with mean 0, we have $\Pr(X > a) \leq \frac{\sigma^2}{\sigma^2 + a^2}$. The above analysis shows that if all intermediate nodes indeed forward the data truthfully, the probability that any node will lose money is

small if there is enough data transmitted between \mathbf{s} and \mathbf{d} . Given N units of data to be transferred, the probability that an intermediate node will lose money is at most $\frac{1-\delta}{1-\delta+N\cdot\eta\cdot\delta}$. \square

Proof is omitted due to space limit. If an intermediate node required that the probability to lose money is at most ϵ , then transmitting $N = \lceil \frac{(1-\delta)(1-\epsilon)}{\eta\delta\epsilon} \rceil$ units of data is enough.

THEOREM 2. *For every agent selected on the path $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})$, forwarding data using the agreed QoS θ_r is a SPE.*

PROOF. Notice that the subgame perfect equilibrium of a sequential game can be computed using backward induction [17]. Assume that the last intermediate node $v_{i_{h-1}}$ got the data from its predecessor $v_{i_{h-2}}$. Let's then study what is its best action. If it decides to provide a forwarding service with QoS which is worse than θ_r , the principal will *not* pay it at all. Thus, it gets 0 utility always. If it forwards the data using QoS θ_r , it gets an *expected* utility $\eta \frac{\mathbf{c}_r(v_{i_{h-1}}, v_{i_h})}{\alpha_r(v_{i_{h-1}}, v_{i_h}) \cdot \beta(v_{i_{h-1}}, v_{i_h})}$, since the expected cost by this agent to provide this forwarding service is $\frac{\mathbf{c}_r(v_{i_{h-1}}, v_{i_h})}{\alpha_r(v_{i_{h-1}}, v_{i_h}) \cdot \beta(v_{i_{h-1}}, v_{i_h})}$. In other words, forwarding packets satisfies IRE. Notice that Theorem 1 also implies that forwarding packets satisfies IR with high probability for $\eta > 0$ and sufficiently large N . Similarly, by induction, every intermediate node will forward the data with QoS θ_r . Thus, forwarding truthfully without hidden action is a subgame perfect equilibrium for all intermediate nodes. \square

When an intermediate node does not keep forwarding the packet until it is received by its next-hop neighbor, we can show that its expected profit will be smaller (proof omitted here due to space limit).

4. HIDDEN INFORMATION AND HIDDEN ACTION

We then study how to design routing protocols when a selfish agent has certain private information, *e.g.*, its service cost, and/or the service reliability. Naturally, we will use the VCG(Vickrey-Clarke-Groves) mechanism to induce the truthfulness from all relay agents; and the principal will conduct the routing only if the total VCG payment to all agents is at most its valuation. In other words, intuitively, we will have the following routing scheme described in Algorithm 2.

4.1 Strategyproof Routing with Infinite Valuation

We first study whether the routing scheme described in Algorithm 2 is strategyproof for relay agents when the principal has an infinite valuation $\nu(\mathbf{s}, \mathbf{d})$, *i.e.*, the principal would pay whatever it costs to get the data transmitted.

THEOREM 3. *Assume that $\nu(\mathbf{s}, \mathbf{d})$ is infinity. Truth-telling is a dominant strategy and forwarding truthfully is a SPE when VCG mechanism is used for routing with link layer reliability.*

The proof is omitted here due to space limit. When the principal has an infinite valuation $\nu(\mathbf{s}, \mathbf{d})$, Algorithm 2 is budget-balanced, *i.e.*, the money charged from the principal is same as the total money paid to all intermediate relay agents. Unfortunately, when the principal has a finite valuation and also is selfish in declaring its valuation, it was proved in [23] that *no* mechanism is strategyproof for all intermediate agents and principals, while it is still budget-balanced. It was also proved that the total charge from the principal could be as small as only $\frac{1}{n}$ of the total payment to relay agents

Algorithm 2 Simple Link Layer Reliable Routing

1: **Routing subgame:** First the principal asks every node u , to declare its cost (denoted by $\bar{c}_r(u, v)$) to provide a forwarding service θ_r on link (u, v) for each one of its out-going neighbors v , and the corresponding service reliability $\bar{\alpha}_r(u, v)$. For all links (u, v) , we define its *weight* $\omega(u, v)$ as $\bar{c}_r(u, v)/(\bar{\alpha}_r(u, v) \cdot \beta(u, v))$. Let G' be the resulted graph and $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})$ be the least cost path from G' to connect \mathbf{s} and \mathbf{d} . Let $\mathbf{P}_{G' \setminus u}(\mathbf{s}, \mathbf{d})$ be the least cost path from G' to connect \mathbf{s} and \mathbf{d} without using the node u . For any path \mathbf{P} , let $\|\mathbf{P}\|$ denote the total weight of all intermediate nodes on the path (*i.e.*, excluding the source). The payment is computed using the VCG mechanism. For each node u not on the path $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})$, its payment $\mathcal{P}^{VCG}(u)$ is always 0. For each node u selected on this path, its VCG payment $\mathcal{P}^{VCG}(u)$ is

$$\|\mathbf{P}_{G' \setminus u}(\mathbf{s}, \mathbf{d})\| - \|\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})\| + \omega(u, v)$$

where v is the next-hop node of u on the path $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})$. Notice that the VCG payment will not be materialized until the packet forwarding subgame is finished.

Let $\mathcal{P}(\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d}))$ be the total computed VCG payment to all intermediate nodes on the path $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})$. If

$$\mathcal{P}(\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})) \leq \nu(\mathbf{s}, \mathbf{d}), \quad (1)$$

the principal will decide to conduct the routing, and the output \mathbf{o} of the routing subgame is $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})$. Otherwise, the principal will not initiate the routing, and the output \mathbf{o} is \emptyset and the final payment to every node is 0.

2: **Forwarding subgame:** It is same as the forwarding subgame of Algorithm 1.

when we require that the charge from the principal is at most the total payment to relay agents. Notice that in this paper, we apply the principal model, which means that the sender/receiver are **not** selfish. Thus, it may still be possible to design a budget-balanced and strategyproof mechanism.

4.2 Strategyproof Routing with Finite Valuation

This subsection is devoted to design a strategyproof routing scheme when the principal has finite valuation. Notice that the above routing

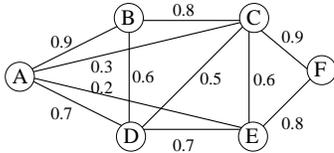


Figure 1: Example of 6 nodes network with link successful delivery probabilities shown along the edges of the graph.

ing scheme (Algorithm 2) is *not* strategyproof anymore if $\nu(\mathbf{s}, \mathbf{d})$ is not infinite. Take Figure 1 for example, where the truthful cost of every node is 1. Assume that A is the source, F is the target and $\nu(\mathbf{s}, \mathbf{d}) = 3.6$. When every node declared its cost and reliability truthfully, then $ABCF$ is the path with the minimum expected cost. The path $ADEF$ is the best path without node B , and it is also the best path without node C . The payment to B is $\frac{1}{0.7} + \frac{1}{0.7} + \frac{1}{0.8} - (\frac{1}{0.9} + \frac{1}{0.8} + \frac{1}{0.9}) + \frac{1}{0.8} \simeq 1.89$. Similarly, the payment to node C is about 1.75. Then no routing will be performed at all since the valuation of the principal is less than the total payment to all intermediate nodes. Let us see what happens if node B declared its cost little bit higher to 1.2. It is not difficult to see that $ABCF$ is still the shortest path and $ADEF$ is still the best path without B , or C . The payment to node B remains

the same as $\simeq 1.89$, but the payment to node C now is decreased to $\simeq 1.5$. Then the total payment by the principal is reduced to $\simeq 3.39 < 3.6$. Thus, the routing is performed and both node B and node C has a positive profit.

Notice that routing scheme described in Algorithm 2 fails to induce the truthfulness when the valuation $\nu(\mathbf{s}, \mathbf{d})$ is finite, which is not because of the VCG mechanism but because of the criterion (see inequality (1)) to decide when to conduct the routing. It is not difficult to observe the following.

OBSERVATION 4. *If the principal performs routing only when a condition \mathcal{C} is satisfied, to induce truthfulness from agents, the condition \mathcal{C} should not depend on the information of agents who are selected by the truthful routing subgame.*

The intuition behind this is that, although any agent i cannot change its payment under a strategyproof payment mechanism by manipulating its declared information, it can reduce the payment to other agents on the output and thus reduce the total payment required from the principal. By careful manipulation, an agent can change the outcome of the game: it makes a positive profit by performing the routing which will not be performed under the truthful declaration. Thus, we need to design other criterion \mathcal{C} when the principal should perform the routing. Obviously, the condition \mathcal{C} should not use any information of agents on path $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})$.

LEMMA 5. *If a condition \mathcal{C} , by which the principal decides whether to perform routing using the path **LCP** (replacing the condition (1) of Algorithm 2), does not depend on the information of any node on **LCP**, routing scheme described by Algorithm 2 is strategyproof.*

PROOF. First of all, for any agent that is not selected on the least cost path **LCP** = $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})$, it cannot lie to improve its profit: even it is selected after false declaration, its new VCG payment (based on the false cost vector) is smaller than its expected cost because VCG payment mechanism is strategyproof. We then show that any node u on the least cost path will not lie either. Notice that any such node cannot change its VCG payment as long as it is selected to the least cost path. Additionally, since the condition \mathcal{C} used by the principal is not affected by the information of any node on **LCP**, a node u cannot have any impact on the decision of the principal (whether it will perform routing). \square

In next, we discuss how our routing mechanism is designed whether the forwarding cost is nodal cost or link cost.

1. Routing Mechanism with Node Cost

When the information (service cost and the service reliability) of a node u is independent of the out-going neighbor v , we next show how to design a criterion that induces the truthfulness of agents. Our approach is to find a new bound that is independent of nodes on **LCP** and is at least the VCG payment to the node for each intermediate node u . Assume the least cost path is $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d}) = v_{i_1} v_{i_2} \cdots v_{i_j} \cdots v_{i_h}$. For an intermediate node v_{i_j} , a simple path \mathbf{B} that starts at some node v_{i_a} and ends at some node v_{i_b} is called a *bridge* of v_{i_j} if (1) $a < j$, (2) $b > j$, and (3) \mathbf{B} and $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})$ have no other common node. Observe that $\mathbf{P}_{G' \setminus \mathbf{LCP}}(\mathbf{s}, \mathbf{d})$ is a bridge of any internal node v_{i_j} . Obviously, the VCG payment to node v_{i_j} is no more than the total cost of all internal nodes of *any* bridge of v_{i_j} . Among the bridges for node v_{i_j} , we use $\mathbf{B}_{\min}(v_{i_j})$ to denote the bridge with the minimum total cost of internal nodes, which is denoted as $\|\mathbf{B}_{\min}(v_{i_j})\|_I$. Thus, $\mathcal{P}^{VCG}(v_{i_j}) \leq \|\mathbf{B}_{\min}(v_{i_j})\|_I$.

For some networks, it is possible that there may have *no bridge* at all for some intermediate node. We show that this happens iff the original network is **not** 2-connected. When every intermediate

node v has a bridge, then the removal of this node will *not* disconnect the source and the target, *i.e.*, the network is 2-connected. When network is 2-connected, then the removal of any node will not disconnect the network, which can clearly serve as a bridge. Recall that it is well-known that the VCG payment exists iff the network is 2-connected (*i.e.*, there is no monopoly node in the network). Thus, when the network is not 2-connected, we cannot apply VCG mechanism and have to rely on some other mechanism (*e.g.*, Nash Equilibrium) which will be discussed later.

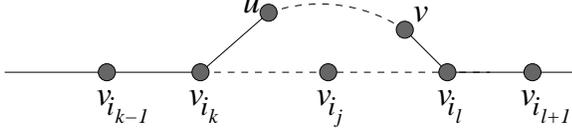


Figure 2: Example of different bridges of node v_{i_j} .

Notice that $\mathbf{B}_{\min}(v_{i_j})$ is not affected by any node on the least cost path $v_{i_1}v_{i_2}\cdots v_{i_j}\cdots v_{i_h}$ since the value of $\mathbf{B}_{\min}(v_{i_j})$ is the sum of the cost from node u to node v . See Figure 2. Moreover, it can be found in polynomial-time using all pairs shortest path in $G' \setminus v_{i_j}$. Thus, based on Lemma 5, the following decision criterion will induce the truthfulness from all agents: the principal conducts the routing only when

$$\sum_{j=2}^{h-1} \|\mathbf{B}_{\min}(v_{i_j})\|_I \leq \nu(\mathbf{s}, \mathbf{d}). \quad (2)$$

2. Routing Mechanism with Link Cost

If the information of some agents depends on its out-going neighbors, *i.e.*, the costs are on links, the situation will be totally deferent. Again take Figure 2 for example, node v_{i_k} now can affect $\mathbf{B}_{\min}(v_{i_j})$ by manipulating the cost of the link (v_{i_k}, u) since it contributes part value of $\mathbf{B}_{\min}(v_{i_j})$. We then study how to design criterion when the principal should perform routing when the information of some agents may depend on its out-going neighbors, *i.e.*, the costs are on links. We define the bridge similarly as previous. Let $\|\mathbf{B}\|$ denote the total cost of all links on the bridge \mathbf{B} . Let $\mathbf{B}_{\min}(v_{i_j})$ be the bridge for v_{i_j} with the minimum cost. Then, we have $\mathcal{P}^{VCG}(v_{i_j}) \leq \|\mathbf{B}_{\min}(v_{i_j})\|$. Notice that if we still use criterion (inequality (2)), *i.e.*, the principal performs routing if $\sum_{j=2}^{h-1} \|\mathbf{B}_{\min}(v_{i_j})\| \leq \nu(\mathbf{s}, \mathbf{d})$, an internal node may lie to improve its utility as follows. Assume that principal decides not to perform routing since the inequality (2) is barely violated. Assume that the bridge $\mathbf{B}_{\min}(v_{i_j})$ starts with some internal node v_{i_a} with $1 < a < j$. Assume that $v_{i_a}u$ is a link on $\mathbf{B}_{\min}(v_{i_j})$. Node v_{i_a} can lie down its service cost on link $v_{i_a}u$ such that the LCP does not change. After such manipulation, $\sum_{j=2}^{h-1} \|\mathbf{B}_{\min}(v_{i_j})\|$ could be reduced to be less than $\nu(\mathbf{s}, \mathbf{d})$, which implies that the principal will perform routing and all nodes on LCP will improve its utility.

Based on the above discussion, it is not difficult to get a condition as follows. Instead of considering bridges that could start with any internal node, we will only consider the bridge that starts with the source node \mathbf{s} . For an intermediate node v_{i_j} , a simple path \mathbf{B}^s that starts at the source node \mathbf{s} and ends at some node v_{i_b} is called a *source-bridge* of v_{i_j} if (1) $b > j$, and (2) \mathbf{B} and $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})$ have no other common node. Among the source-bridges for v_{i_j} , we find the one with the minimum cost, denoted as $\mathbf{B}_{\min}^s(v_{i_j})$. Then, the principal decides to perform routing only when

$$\sum_{j=2}^{h-1} \|\mathbf{B}_{\min}^s(v_{i_j})\| \leq \nu(\mathbf{s}, \mathbf{d}). \quad (3)$$

is satisfied, and the payment to each intermediate node is just its VCG payment. In summary, we have the following truthful routing scheme (Algorithm 3) when the principal has finite valuation.

Algorithm 3 Link Layer Reliable Routing with Finite Valuation

Input: graph G , QoS θ_r required, valuation $\nu(\mathbf{s}, \mathbf{d})$.

- 1: **Routing subgame:** Same as Algorithm 2, the principal first collects the declared cost $\bar{c}_r(u, v)$ by u , and then build the graph G' and the least cost path $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})$, denoted as $v_{i_1}v_{i_2}\cdots v_{i_j}\cdots v_{i_h}$. For each node u on the LCP, the principal computes its VCG payment $\mathcal{P}^{VCG}(u)$ as

$$\|\mathbf{P}_{G' \setminus u}(\mathbf{s}, \mathbf{d})\| - \|\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})\| + \omega(u, v)$$

where v is the next-hop node of u on the path $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})$.

The principal decides to conduct the routing based on the following conditions:

- When the private of information (service cost and service reliability) of any agent *depends* on its out-going neighbor, the principal starts routing if $\sum_{j=2}^{h-1} \|\mathbf{B}_{\min}^s(v_{i_j})\| \leq \nu(\mathbf{s}, \mathbf{d})$.
- When the private of information of every agent is *independent* of its out-going neighbor, the principal starts routing if $\sum_{j=2}^{h-1} \|\mathbf{B}_{\min}(v_{i_j})\|_I \leq \nu(\mathbf{s}, \mathbf{d})$.

If the above conditions are not met, the principal will not initiate the routing, and the final payment to every node is 0.

- 2: **Forwarding subgame:** The forwarding is same as that for the forwarding subgame of Algorithm 1.
-

Based on Lemma 5, we have

THEOREM 6. *Algorithm 3 inducing truthful declaration and forwarding from all intermediate nodes is a subgame perfect equilibrium. Furthermore, the conditions used by the principal to decide when to perform routing is tight.*

Discussions: Observe that in our truthful routing schemes, we showed that *simply applying VCG mechanism directly does not induce truthfulness* when requestors have finite valuation for sending packets. Notice that truthfulness of our truthful routing scheme does not come without cost: we often will pay more than the actual cost to achieve truthfulness; the payment computation is also more complicated than the traditional shortest path routing. On the other hand, paying more is necessary to achieve truthfulness due to individual rationality requirement. Efficient payment computation method has been proposed in [22], which can compute the payment to *all intermediate nodes* in time $O(m + n \log n)$. Next we present a method that is easier to implement using Nash design.

4.3 ϵ -Nash Equilibrium Routing

So far, the routing schemes presented in this subsection will induce the truthfulness from agents. Possible disadvantages of ensuring the truthfulness are that (1) the principal could lose some opportunities to send data from the source to the target, and (2) the payment by the principal could be larger than that by some other mechanism even the routing is performed. We then present a simple routing scheme which will choose the same routing path as VCG mechanism, and pay less than the VCG mechanism, in the expense of losing the truthful declaration from agents. The routing scheme works as Algorithm 4.

Let $\tau = (\tau_1, \tau_2, \dots, \tau_n)$ be a vector of the declared information by all nodes, where τ_i is all the service cost and service reliability information declared by node v_i . Let $\mathcal{O}(\tau)$ be the optimal path found under the declaration τ . Since the path selection will involve tie-breaker, it was shown in [15] that there is *no* Nash Equilibrium in the unicast routing subgame. They propose to use ϵ -Nash Equilibrium, in which any agent cannot improve its utility by more than

Algorithm 4 ϵ -Nash Equilibrium Based Link Layer Routing

Input: graph G , QoS θ_r required, valuation $\nu(\mathbf{s}, \mathbf{d})$.

- 1: **Routing subgame:** First the principal asks every node u its service cost $\bar{\mathbf{c}}_r(u, v)$ and the service reliability $\bar{\alpha}_r(u, v)$ for each of its outgoing neighbor v . Define graph $G' = (V, E)$ where the weight of a link (u, v) is $\bar{\mathbf{c}}_r(u, v) / (\bar{\alpha}_r(u, v) \cdot \beta(u, v))$. Find the least cost path $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})$ connecting \mathbf{s} and \mathbf{d} that can support the given QoS θ_r . If $\mathcal{L}(\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})) \leq \nu(\mathbf{s}, \mathbf{d})$, the principal will decide to conduct the routing. Otherwise, the principal will not initiate the routing. For each node not on the final path, its payment is always 0. For each node u selected, its payment is

$$\frac{\bar{\mathbf{c}}_r(u, v)}{\bar{\alpha}_r(u, v) \cdot \beta(u, v)}$$

where v is the next-hop node of u on the LCP path.

- 2: **Forwarding subgame:** It is same as that of Algorithm 1.
-

ϵ if it unilaterally changes its declaration. Here ϵ is an input parameter (with arbitrarily small value) controlled by the game. For simplicity, we still call ϵ -Nash Equilibrium as Nash Equilibrium in this paper. Recall that \mathbf{t} is the vector of truthful declaration.

LEMMA 7. For any Nash Equilibrium declaration τ the optimal path $\mathcal{O}(\tau)$ is the same as the optimal path $\mathcal{O}(\mathbf{t})$.

The proof is similar to the proof in [15] and thus is omitted here. Lemma 7 immediately implies that the social efficiency of the outcome under a Nash equilibrium is maximized, which is one of the key property of VCG mechanism. We then categorize the set of Nash equilibriums that can be formed. We only consider the Nash equilibrium where any agent that is not selected on the final least cost path will declare truthfully. Notice that any such agent cannot lie to improve its utility. Without loss of generality, we assume that $v_{i_1} v_{i_2} \cdots v_{i_h}$ is the optimal path under truthful declaration. Let $x_{i_j} = \frac{\bar{\mathbf{c}}_r(v_{i_j}, v_{i_{j+1}})}{\bar{\alpha}_r(v_{i_j}, v_{i_{j+1}}) \cdot \beta(v_{i_j}, v_{i_{j+1}})}$, which is the *expected* cost of node v_{i_j} viewed by the principal under the Nash equilibrium declaration τ , and is the payment node v_{i_j} will receive under our new routing scheme. Let $\mathbf{B}_{\min}^{i_a, i_b}$ be the bridge with the minimum cost and starting at node v_{i_a} and ending at node v_{i_b} . Based on Lemma 7, the Nash equilibrium declaration must also result in the same optimal path $v_{i_1} v_{i_2} \cdots v_{i_h}$, i.e., for any minimum cost bridge $\mathbf{B}_{\min}^{i_a, i_b}$,

$$\sum_{k=a}^{b-1} x_{i_k} \leq \|\mathbf{B}_{\min}^{i_a, i_b}\|, 1 \leq a < b \leq h, b \geq a + 2 \quad (4)$$

The declaration of any node v_{i_k} must satisfy that

$$x_{i_j} \geq \frac{\mathbf{c}_r(v_{i_j}, v_{i_{j+1}})}{\alpha_r(v_{i_j}, v_{i_{j+1}}) \cdot \beta(v_{i_j}, v_{i_{j+1}})}. \quad (5)$$

These conditions (inequalities (4) and (5)) will define a higher dimensional polytope. Any node on the boundary of the polytope (including the truthful declaration of nodes not in the LCP) is a Nash equilibrium declaration. Let $\chi = \sum_{k=1}^h x_{i_k}$ be the sum of the costs of all nodes on the shortest path under some Nash Equilibrium. Obviously, χ varies under different Nash Equilibrium, and we are interested in the worst case performance of our routing mechanism, i.e., we want to find the Nash Equilibria such that χ is maximized. For notational simplicity, we term the Nash Equilibrium that maximizes the total costs χ as the *maximum Nash Equilibria*. Note that $\max \sum_{k=1}^h x_{i_k}$ subject to constraint sets (4) and (5) can be solved in polynomial time using linear programming, and the solution to the linear programming is on the boundary of the polytope. However, solving linear programming with up

to $O(n)$ variables and $O(n^2)$ constraints could be quite costly. Following we summarize a simple method (Algorithm 5) that can elegantly compute one maximum Nash Equilibrium that also has some practical meanings. The proof of the correctness of this method is omitted here due to space limit.

Algorithm 5 Sequential Algorithm to find one Maximum Nash

Input: Declared cost $\mathbf{c}_r(u, v)$ for each node u .

Output: A maximum Nash Equilibrium.

- 1: Build the graph G and compute the least cost path $\mathbf{P}_G(\mathbf{s}, \mathbf{d})$, denoted as $v_{i_1} v_{i_2} \cdots v_{i_j} \cdots v_{i_h}$.
- 2: **for** $k = 1$ to h **do**
- 3: Compute the VCG payment $\mathcal{P}^{VCG}(v_{i_k})$ for node v_{i_k} using updated cost for previous links as

$$\|\mathbf{P}_{G \setminus u}(\mathbf{s}, \mathbf{d})\| - \|\mathbf{P}_G(\mathbf{s}, \mathbf{d})\| + \omega(v_{i_k}, v_{i_{k+1}}).$$

- 4: Adjust the cost for link $(v_{i_k}, v_{i_{k+1}})$ as $\mathbf{c}_r(v_{i_k}, v_{i_{k+1}}) \leftarrow \mathcal{P}^{VCG}(v_{i_k}) \alpha_r(v_{i_j}, v_{i_{j+1}}) \beta(v_{i_j}, v_{i_{j+1}})$. This new cost is used for compute the payment for subsequent links.
 - 5: Return $x_{i_k} = \frac{\mathbf{c}_r(v_{i_k}, v_{i_{k+1}})}{\alpha_r(v_{i_j}, v_{i_{j+1}}) \beta(v_{i_j}, v_{i_{j+1}})}$ for $1 \leq i \leq h$.
-

4.4 Reliable Transport Layer Routing

We then study how to design reliable transport layer protocols when a selfish agent has certain private information, e.g., its service cost and/or the service reliability.

1. Routing Mechanism with Node Cost: When the information (service cost and the service reliability) of a node u is independent of the out-going neighbor v , we next show how to design a criterion that induces the truthfulness of agents. Our approach is to find a new bound that is independent of nodes on **LCP** and is at least the VCG payment to the node for each intermediate node u . Assume the least cost path is $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d}) = v_{i_1} v_{i_2} \cdots v_{i_j} \cdots v_{i_h}$. For an intermediate node v_{i_j} , a simple path \mathbf{B} that starts at some node v_{i_a} and ends at some node v_{i_b} is called a *bridge* of v_{i_j} if (1) $a < j$, (2) $b > j$, and (3) \mathbf{B} and $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})$ have no other common node. Observe that $\mathbf{P}_{G' \setminus \text{LCP}}(\mathbf{s}, \mathbf{d})$ is a bridge of any internal node v_{i_j} . Obviously, the VCG payment to node v_{i_j} is no more than the total cost of all internal nodes of *any* bridge of v_{i_j} . Among the bridges for node v_{i_j} , we use $\mathbf{B}_{\min}(v_{i_j})$ to denote the bridge with the minimum total cost of internal nodes, which is denoted as $\|\mathbf{B}_{\min}(v_{i_j})\|_I$. Thus, $\mathcal{P}^{VCG}(v_{i_j}) \leq \|\mathbf{B}_{\min}(v_{i_j})\|_I$.

2. Routing Mechanism with Link Cost: If the information of some agents depends on its out-going neighbors, i.e., the costs are on links, the situation will be totally deferent. Take Figure 2 for example, node v_{i_k} now can affect $\mathbf{B}_{\min}(v_{i_j})$ by manipulating the cost of the link (v_{i_k}, u) since it contributes part value of $\mathbf{B}_{\min}(v_{i_j})$. We then study how to design criterion when the principal should perform routing when the information of some agents may depend on its out-going neighbors, i.e., the costs are on links. We define the bridge similarly as previous. Let $\|\mathbf{B}\|$ denote the total cost of all links on the bridge \mathbf{B} . Let $\mathbf{B}_{\min}(v_{i_j})$ be the bridge for v_{i_j} with the minimum cost. Instead of considering bridges that could start with any internal node, we will only consider the bridge that starts with the source node \mathbf{s} . For an intermediate node v_{i_j} , a simple path \mathbf{B}^s that starts at the source node \mathbf{s} and ends at some node v_{i_b} is called a *source-bridge* of v_{i_j} if (1) $b > j$, and (2) \mathbf{B} and $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})$ have no other common node. Among the source-bridges for v_{i_j} , we find the one with the minimum cost, denoted as $\mathbf{B}_{\min}^s(v_{i_j})$.

Algorithm 6 presents the routing scheme when the principal has an infinite valuation.

Algorithm 6 Simple Transport Layer Reliable Routing

1: **Routing subgame:** First the principal asks every node u , to declare its cost (denoted by $\bar{c}_r(u, v)$) to provide a forwarding service θ_r on link (u, v) for each one of its out-going neighbors v . The principal also collects the corresponding service reliability $\bar{\alpha}_r(u, v)$ if it is a private information of a node. For all links (u, v) , we define its link reliability as $\delta(u, v) = \bar{\alpha}_r(u, v) \cdot \beta(u, v)$. Let G' be the resulted graph and $\mathbf{P}_{G'}^T(\mathbf{s}, \mathbf{d}) = v_{i_1}v_{i_2} \cdots v_{i_h}$ be the least cost path under the TCP model, where $\mathbf{s} = v_{i_1}$, $\mathbf{d} = v_{i_h}$. Let $\mathbf{P}_{G' \setminus u}^T(\mathbf{s}, \mathbf{d})$ be the least cost path without using a node u . Path $\mathbf{P}_{G'}^T(\mathbf{s}, \mathbf{d})$ is used for routing. For each node u not on the routing path $\mathbf{P}_{G'}^T(\mathbf{s}, \mathbf{d})$, its payment $\mathcal{P}^{VCG}(u) = 0$ always. For each internal node v_{i_j} $1 < j < h$ on this path, its payment $\mathcal{P}^{VCG}(v_{i_j})$ is

$$\mathcal{E}^T(\mathbf{P}_{G' \setminus v_{i_j}}^T(\mathbf{s}, \mathbf{d})) - \mathcal{E}^T(\mathbf{P}_{G'}^T(\mathbf{s}, \mathbf{d})) + \frac{\bar{c}(v_{i_{j-1}}, v_{i_j})}{\prod_{t=j}^h \delta(v_{i_{t-1}}, v_{i_t})}.$$

If the principal has an infinite valuation $\nu(\mathbf{s}, \mathbf{d})$, the principal will always perform the routing. Otherwise, the principal decide to perform the routing only if

$$(h-1) \|\mathbf{P}_{\min}^{\mathbf{s}, \mathbf{d}}\| \leq \nu(\mathbf{s}, \mathbf{d}) \quad (6)$$

2: **Forwarding subgame:** When an intermediate node v_{i_j} ($2 \leq j \leq h-1$) received a data packet by the principal, it can forward the packet to the next-hop node $v_{i_{j+1}}$ using QoS θ_r , or using some other forwarding QoS.

The principal materializes the payment to every intermediate agent v_{i_j} only if the target node \mathbf{d} received the data correctly with the given QoS. In other words, if any node on the path lowers its forwarding service QoS, every node on the path will *not* receive any payment.

THEOREM 8. Assume that $\nu(\mathbf{s}, \mathbf{d})$ is infinity. Truth-telling is a dominant strategy and forwarding truthfully is a SPE when our algorithm for routing with transport layer reliability is used.

The proof of this theorem is omitted due to space limit.

5. MULTI-PATH ROUTING

To the best of our knowledge, all previous routing protocols with selfish agents assume single path routing, including our study in previous sections. In certain situations, the bandwidth of any single path may be smaller than the required bandwidth \mathbf{F} by the principal. Multi-path routing is thus needed. Assume that for each directed link $e = (u, v)$, its capacity is $\mathbf{B}(e)$ when there is no interference and no contention. Then the multi-path routing is to find $\mathbf{x}(e)$ for each directed link e such that link e will carry $\mathbf{x}(e)\mathbf{B}(e)$ total traffic from the principal. The total cost incurred by all relay agents is $\sum_e \mathbf{c}(e)\mathbf{x}(e)\mathbf{B}(e)/\delta(e)$ when the link reliability is implemented. The multi-path routing must satisfy several constraints: (1) Flow Conservation, (2) Minimum Demand, (3) Capacity Constraint, (4) Interference Constraint. The multi-path routing is to solve the following:

$$\min \sum_e \frac{\mathbf{c}(e)\mathbf{B}(e)}{\delta(e)} \mathbf{x}(e)$$

$$s.t. \begin{cases} \sum_u \mathbf{x}(u, v)\mathbf{B}(u, v) = \sum_w \mathbf{x}(v, w)\mathbf{B}(v, w) & \forall v \\ \sum_v \mathbf{x}(s, v)\mathbf{B}(s, v) \geq \mathbf{F} & \\ 0 \leq \mathbf{x}(e)/\delta(e) \leq 1 & \forall e \\ \frac{\mathbf{x}(e)}{\delta(e)} + \sum_{e' \in I(e)} \frac{\mathbf{x}(e')}{\delta(e')} \leq 1 & \forall e \end{cases}$$

We then design a strategyproof routing as follows. Let $\mathcal{O}(\bar{\mathbf{c}}, \bar{\alpha}) = (\mathbf{x}(e_1), \mathbf{x}(e_2), \dots, \mathbf{x}(e_m))$ be the solution found when the declared

cost vector is $\bar{\mathbf{c}}$, and the declared service reliability is $\bar{\alpha}$. A node v_k is *selected* if at least one of its out-going links has a positive $\mathbf{x}(e)$ in $\mathcal{O}(\bar{\mathbf{c}}, \bar{\alpha})$. Let $\mathcal{O}(\bar{\mathbf{c}}_{-k}, \bar{\alpha}_{-k}) = (\mathbf{x}_{-k}(e_1), \mathbf{x}_{-k}(e_2), \dots, \mathbf{x}_{-k}(e_m))$ be the solution found when a node v_k is removed from the network. In our mechanism, the payment to any non-selected node is always 0. The payment to a selected node v_k is

$$\sum_e \frac{\mathbf{c}(e)\mathbf{B}(e)}{\delta(e)} \mathbf{x}_{-k}(e) - \sum_e \frac{\mathbf{c}(e)\mathbf{B}(e)}{\delta(e)} \mathbf{x}(e) + \sum_{e: e=(v_k, u)} \frac{\mathbf{c}(e)\mathbf{B}(e)}{\delta(e)} \mathbf{x}(e).$$

THEOREM 9. The above payment scheme is strategyproof, the allocated flows guarantee to be schedulable, and the routing cost is within a constant factor of optimum while the flow is at least $\Theta(F)$.

PROOF. The proof of strategyproof is similar to VCG and thus omitted here due to space limit. Let \mathbf{x}^* be the optimum schedulable flow allocation and \mathbf{x}^{LP} be the schedulable flow allocation found by our linear programming. Condition $\frac{\mathbf{x}(e)}{\delta(e)} + \sum_{e' \in I(e)} \frac{\mathbf{x}(e')}{\delta(e')} \leq 1$ ensures that the solution \mathbf{x}^{LP} is schedulable, see [1, 12, 13]. Let $cost(\mathbf{y})$ be the total cost under a flow allocation \mathbf{y} . From [1, 12, 13], all schedulable flows must satisfy that $\frac{\mathbf{x}(e)}{\delta(e)} + \sum_{e' \in I(e)} \frac{\mathbf{x}(e')}{\delta(e')} \leq \mathbf{C}$ for a constant $\mathbf{C} > 1$ depending on interference model. Thus, $\frac{\mathbf{x}^*}{\mathbf{C}}$ satisfies all the constraints of our linear programming, except that the achieved flow is at least F/\mathbf{C} . It implies that $cost(\mathbf{x}^{LP}) \leq cost(\frac{\mathbf{x}^*}{\mathbf{C}}) = \frac{cost(\mathbf{x}^*)}{\mathbf{C}}$, i.e., the cost of our multi-path routing is within a constant factor of optimum. \square

6. PERFORMANCE STUDY

We conduct extensive simulations to study the performance of our routing schemes for hidden information and hidden actions. Given a routing path \mathbf{P} , we define the *overpayment ratio* (OR) of \mathbf{P} as $\varrho^{\mathcal{A}}(\mathbf{P}) = \frac{\mathcal{P}^{\mathcal{A}}(\mathbf{P})}{c(\mathbf{P})}$, where $\mathcal{P}^{\mathcal{A}}(\mathbf{P})$ is the total payment to all agents on the path by a scheme \mathcal{A} , and $c(\mathbf{P})$ is the total cost of all agents on the path.

In our experiment, we generate random networks with n nodes, where n is a parameter. To reflect the wireless network property, we first randomly generate n nodes placed in a unit area, and then assign each wireless node a transmission range r_n . We then produce the unit disk graph where two nodes are connected only if their Euclidean distance is at most r_n . Then a network $G(n, r_n, p)$ is generated by randomly selecting any edge from the unit disk graph with probability p . For all results reported here, we typically choose p and r_n such that the network is 2-connected with high probability. We then randomly assign the cost and the reliability to nodes and links. The cost of each node is then uniformly drawn from all integers in $[1, 50]$ and the reliability of a link is uniformly drawn from $0.01 \cdot k$ with integer k distributed in $[1, 100]$. By choosing different parameters, we study which aspects of the network affect the cost of the selected paths and the overpayment ratio.

6.1 Effect of Network Size

In this simulation, we fix the parameter p and r_n to $npr_n^2 = \log n/2$, which means that on average the node degree will be around $\frac{\pi}{2} \log n$. We first study the cost variations when the network size changes and different cost models (nodal cost or link cost). We find that there is no fixed relation between the network size and the cost. The following relations always hold: actual cost \leq Nash payment \leq VCG payment \leq the bridge cost. Figure 3 plots the cost ratios for different models when the network size changes. We also specifically study the Nash payment over actual cost, the VCG payment over actual cost, and the bridge cost over actual cost. We observe that the ratios do not have a pattern depending on the

network size. In our simulations, we find that the Nash cost is about 1.5 times of the actual cost of all relaying agents, the VCG payment is about 2 times of the actual cost, while the bridge costs about 3 times of the actual cost of the routing path. The simulation results show that the price for achieving truthful declaration from all relay agents is *small*: the payment needed by truthful VCG mechanism is only about $4/3$ times of the payment by Nash equilibrium. Notice that the scheme by Nash Equilibrium has its own disadvantages: it is more expensive to implement this scheme since it requires multiple iterations to converge, and the output is also not stable since there may have multiple Nash equilibriums. The fact that the bridge cost is about 3 times of the actual cost and about $3/2$ times of the VCG payment to all relay agents implies that some additional price is required to induce the truthfulness from all relay agents when the principal has only a finite valuation: it will not be able to perform routing even its valuation is enough to cover the VCG payment. In other words, Nash payment may be better off if performing routing is more important than inducing truthfulness from all agents.

6.2 Effect of Network Density

We then study other effects by fixing the network size (100 in the results reported here). We specifically study the effect of the network density by changing the average number of links in a random network. Figure 4 shows different costs when the network density changes from the minimum p_0 needed for connectivity to $p_0 + 0.3$. We can observe that the costs decrease when the network density increases. This is because the path length will decrease and the competition for the shortest path also will increase when the network density increases. On the other hand, the ratios do not change much when link layer reliability is implemented.

7. RELATED WORK

In the past five years, several new routing protocols [2,6,7,14,18,22,24] have been proposed to deal with possible selfishness of intermediate nodes. Generally, there are two approaches to deal with the potential selfish nodes: (1) *mechanism design* based on game theory: give selfish nodes certain incentives (*e.g.*, virtual or actual payment or differentiated services) such that they will behave as desired out of their own interests; (2) *reputation* based approach: detect these misbehaving nodes and then route around them by collectively isolating these misbehaving nodes. Routing has been an important part of the algorithmic mechanism-design from the very beginning when the cost of agents are *hidden information*. Nisan and Ronen [16] provided a polynomial-time strategyproof mechanism using VCG for optimal unicast route selection in a centralized computational model. Feigenbaum *et. al* [6] then addressed the truthful low cost routing in a different network model. They assume that each node k incurs a transit cost c_k for each transit packet it carries. Their mechanism again is the VCG mechanism. Optimal methods are presented in [10] to compute payments to all links and in [22] to compute the payments to all individual nodes. Anderegg and Eidenbenz [2] recently proposed a similar routing protocol for wireless ad hoc networks based on VCG mechanism again. They assumed that each link has a cost and each node is a selfish agent. Wang *et al.* [24] proposed several strategyproof mechanisms for multicast such that every selfish node will maximize its profit in the multicast structure if it declares its privately known cost truthfully. Qiu *et al.* [18] studied the selfish routing in Internet-like environments. Zhong *et al.* [26] showed that there does not exist a dominant strategy solution in the forwarding subgame and present *Corsac*, a cooperation-optimal protocol consisting of a routing protocol and a forwarding protocol. Their routing protocol integrates VCG with a novel cryptographic techniques (1) to address the chal-

lenges in wireless ad hoc networks that a link's cost is determined by sender and receiver together; and(2) to design a forwarding protocol to enforce the routing decision such that fulfilling the routing decision is the *optimal* action of each intermediate node.

Achieving cooperation among selfish terminals in network was previously addressed by several authors using mainly reputation based scheme. In [4,5,11], a secure mechanism to stimulate nodes to cooperate is presented. Each terminal maintains a counter, called *nugget counter*, in a tamper resistant hardware module, which is decreased when the terminal originates a packet and increased when the terminal forwards a packet. In [20,21], several methods are presented such that nodes' actions will form Nash Equilibrium and the energy efficiency is achieved at the equilibrium. In [14], nodes, which agree to relay traffic but do not, are termed as misbehaving. Their protocol avoids routing through these misbehaving nodes based on *Watchdog* and *Pathrater*.

Feldman *et al.* [7,8] studied the hidden action in multi-hop routing. They studied the mechanism design for UDP model (given the routing path) with known uniform cost and reliability for agents, with known uniform reliability but private cost for agents. For the first problem, using a principal-agent model, they showed how the hidden-action problem can be overcome through appropriate design of contracts, in both the direct and recursive cases. For the second problem, they applied VCG mechanism.

8. CONCLUSION

In this paper we studied both the hidden information and the hidden action phenomena in multihop non-cooperative wireless networks. We designed several protocols such that the routing principal can deal with hidden information and eliminate the negative effect of the hidden action by selfish terminals. One of our protocols can induce the truthful declarations from all relay agents such that a social efficient routing path can be formed; furthermore our carefully designed payment scheme ensures that forwarding the data packets using the correct QoS is a subgame perfect equilibrium (SPE) for each intermediate relay agent. We also design routing schemes whose induced Nash equilibriums are good. To make it possible to perform more routings, we relaxed the dominant strategy requirement for relay agents to a Nash equilibrium.

This paper is just the start of studying both hidden information and hidden action. There are still a number of challenges left unsolved. The first challenge is to design a proper routing scheme when the actual cost and link reliability depending on the set of links that will be active. Notice that TDMA link scheduling can make the link cost and reliability more measurable and stable, but TDMA link scheduling itself needs all nodes collaborate with each other, which contradicts our assumption that all nodes are selfish and non-cooperative. Another challenge is to deal with possible collusion and shill bidding in ad hoc networks.

9. REFERENCES

- [1] ALICHERY, M., BHATIA, R., AND LI, L. E. Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks. In *MobiCom* (2005), pp. 58–72.
- [2] ANDEREGG, L., AND EIDENBENZ, S. Ad hoc-vcg: a truthful and cost-efficient routing protocol for mobile ad hoc networks with selfish agents. In *MobiCom* (2003), pp. 245–259.
- [3] ARCHER, A., AND L. TARDOS. Frugal path mechanisms. In *SODA* (2002), pp. 991–998.
- [4] BUTTYAN, L., AND HUBAUX, J. Stimulating cooperation in self-organizing mobile ad hoc networks. *MONET* 5, 8 (2003).
- [5] BUTTYAN, L., AND HUBAUX, J. P. Enforcing service availability in mobile ad-hoc wans. In *MobiCom* (2000), pp. 87–96.

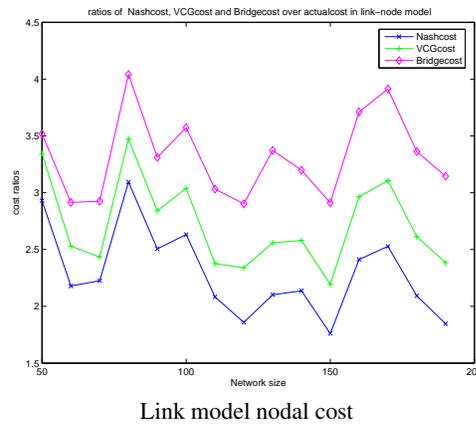
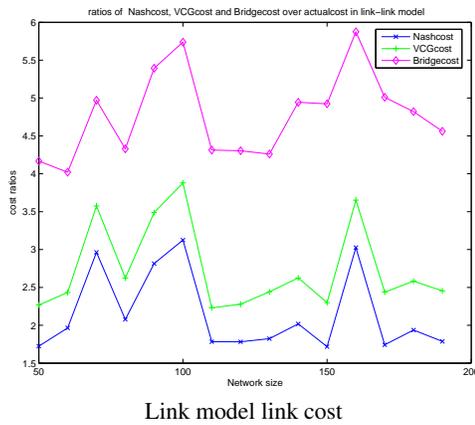


Figure 3: Different cost ratios when networking size changes.

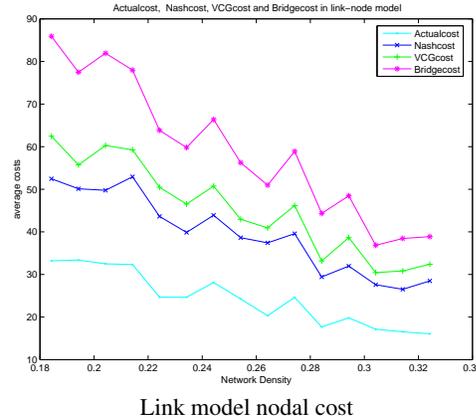
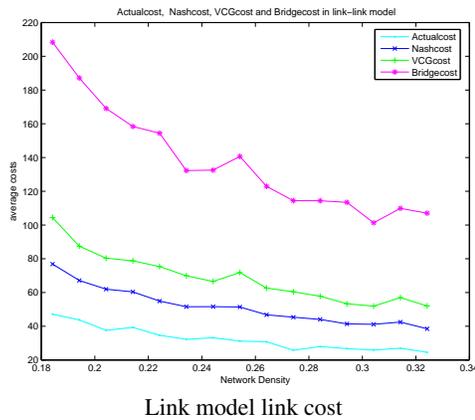


Figure 4: Different costs when networking density changes.

- [6] FEIGENBAUM, J., PAPADIMITRIOU, C., SAMI, R., AND SHENKER, S. A BGP-based mechanism for lowest-cost routing. In *ACM PODC* (2002), pp. 173–182.
- [7] FELDMAN, M., AND CHUANG, J. Hidden-action in multi-hop routing. In *Second Workshop on the Economics of P2P Systems, in conjunction with ACM Sigcomm* (2004).
- [8] FELDMAN, M., CHUANG, J., STOICA, I., AND SHENKER, S. Hidden-action in multi-hop routing. In *ACM E-Commerce Conference* (2005).
- [9] FELEGYHAZI, M., BUTTYAN, L., AND HUBAUX, J. Equilibrium analysis of packet forwarding strategies in wireless ad hoc networks – the static case. In *Personal Wireless Communications* (2003).
- [10] HERSHBERGER, J., AND SURI, S. Vickrey pricing in network routing: Fast payment computation. In *IEEE FOCS* (2001), pp. 252–259.
- [11] JAKOBSSON, M., HUBAUX, J.-P., AND BUTTYAN, L. A micro-payment scheme encouraging collaboration in multi-hop cellular networks. In *Proceedings of Financial Cryptography* (2003).
- [12] KODIALAM, M., AND NANDAGOPAL, T. Characterizing achievable rates in multi-hop wireless networks: the joint routing and scheduling problem. In *MobiCom* (2003), pp. 42–54.
- [13] KUMAR, V. S. A., MARATHE, M. V., PARTHASARATHY, S., AND SRINIVASAN, A. Algorithmic aspects of capacity in wireless networks. *SIGMETRICS Perform. Eval. Rev.* 33, 1 (2005), 133–144.
- [14] MARTI, S., GIULI, T. J., LAI, K., AND BAKER, M. Mitigating routing misbehavior in mobile ad hoc networks. In *ACM MobiCom* (2000).
- [15] N. IMMORLICA, D. KARGER, E. N., AND SAMI, R. First-price path auctions. In *ACM Conf. Electronic Commerce* (2005).
- [16] NISAN, N., AND RONEN, A. Algorithmic mechanism design. In *ACM STOC* (1999), pp. 129–140.
- [17] OSBORNE, M. J., AND RUBINSTEIN, A. *A course in game theory*. The MIT Press, 2002.
- [18] QIU, L., YANG, Y. R., ZHANG, Y., AND SHENKER, S. On selfish routing in internet-like environments. In *ACM SIGCOMM* (2003), pp. 151–162.
- [19] SALEM, N. B., BUTTYAN, L., HUBAUX, J.-P., AND KAKOBSSON, M. A charging and rewarding scheme for packet forwarding in multi-hop cellular networks. In *ACM MobiHoc* (2003).
- [20] SRINIVASAN, V., NUGGEHALI, P., CHIASSERINI, C. F., AND RAO, R. R. Energy efficiency of ad hoc wireless networks with selfish users. In *European Wireless Conf.* (2002).
- [21] SRINIVASAN, V., NUGGEHALI, P., CHIASSERINI, C. F., AND RAO, R. R. Cooperation in wireless ad hoc wireless networks. In *IEEE Infocom* (2003).
- [22] WANG, W., AND LI, X.-Y. Low-cost routing in selfish and rational wireless ad hoc networks. *IEEE Trans. on Mobile Computing* (2005).
- [23] WANG, W., AND LI, X.-Y. Nash equilibrium, dominant strategies in routing. In *Workshop for Int. and Network Economics* (2005).
- [24] WANG, W., LI, X.-Y., AND WANG, Y. Truthful multicast in selfish wireless networks. In *ACM MobiCom* (2004).
- [25] ZHONG, S., CHEN, J., AND YANG, Y. Sprite: A simple, cheat-proof, credit-based system for mobile ad hoc networks. In *IEEE INFOCOM* (2003).
- [26] ZHONG, S., LI, E. L., LIU, Y., AND YANG, Y. R. On designing incentive-compatible routing and forwarding protocols in wireless ad-hoc networks. In *ACM Mobicom* (2005).