Dealing With Selfishness and Moral Hazard in Non-Cooperative Wireless Networks

Yanwei Wu, *Student Member, IEEE,* Shaojie Tang, Ping Xu and Xiang-Yang Li, *Senior Member, IEEE,*

***Abstract*— For non-cooperative networks in which each node is a selfish agent, certain incentives must be given to intermediate nodes to let them forward the data for others. What makes the scenario worse is that, in a multi-hop non-cooperative network, 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. Thus, in the absence of properly designed incentive schemes, rational and selfish intermediate nodes may choose to forward data packets at a very low priority or simply drop the packets, and they could put the blame on the unreliable channel. In this paper, assuming the receiver is a trusted authority, we propose several methods that discourage the hidden actions under hidden information in multi-hop non-cooperative networks with high probability. We design several algorithmic mechanisms for a number of routing scenarios such that each selfish agent will maximize its expected 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 payments by our mechanisms are only slightly larger than the *actual cost* incurred by all intermediate nodes.**

***Index Terms*— Non-cooperative networks, reliability, hidden action, hidden-information, selfish, truthful, mechanism.**

I. Introduction

Many networks are composed of terminals that are owned by incentive-driven individuals. We call them non-cooperative networks hereafter. 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, limitations 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. Therefore, if we assume that all users are selfish, incentives must be provided to encourage networking terminals' participation and thus to 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 $\mathbf{c}(i, j)$ if it is asked to forward a *unit* amount of data to a neighboring node j. Here we assume that the cost $\mathbf{c}(i, j)$ is a piece of *private* information known *only* by node i (some studies, *e.g.* [46], assumed that it is determined by both i and j if $\mathbf{c}(i, j)$ only depends on transmission power). Consider a unicast routing and forwarding protocol based on the least cost path (LCP): each terminal is asked to declare a cost for forwarding 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* in the literature since it removes speculation and counter-speculation among terminals.

An essential argument in [46] for showing 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 causes of failure to either the deliberate action of some intermediate node(s), or some external factors beyond the control of these intermediate nodes, such as network congestion, channel interference, or data corruption. We 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

Department of Computer Science, Illinois Institute of Technology. Emails: {ywu24,stang7,pxu3}@iit.edu, xli@cs.iit.edu. Part of the paper has been published at ACM MobiHoc 2008. The research of authors are partially supported by NSF CNS-0832120, National Natural Science Foundation of China under Grant No. 60828003, the Natural Science Foundation of Zhejiang Province under Grant No.Z1080979, National Basic Research Program of China (973 Program) under grant No. 2010CB328100, 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, Hong Kong RGC HKUST 6169/07, the RGC under Grant HKBU 2104/06E, and CERG under Grant PolyU-5232/07E.

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 concerning information asymmetry and incentives.

We will focus on designing cost-efficient strategyproof routing protocols under both hidden information and hidden action. We assume that all sessions have a common receiver, which is a trusted authority. We propose several efficient methods that can induce truthfulness and eliminate the hidden actions with high probability for various network models and routing requirements. In our protocols, each selfish agent will maximize its expected profit when it *truthfully* declares its cost and acts truthfully following its declared actions. Compared with some closely related results [15], [46], the main contributions of this paper are as follows.

In all our models, the source node $\mathbf{s}$ has a valuation $\nu(\mathbf{s}, \mathbf{d})$ for sending a unit amount of data from the source to the destination node $\mathbf{d}$. We consider two complementary cases: $\nu(\mathbf{s}, \mathbf{d})$ is infinite and $\nu(\mathbf{s}, \mathbf{d})$ is finite. A naive method is to ask the source declare $\nu(\mathbf{s}, \mathbf{d})$ and each other node declare its cost for forwarding a unit amount of data. The least cost path is used to connect $\mathbf{s}$ and $\mathbf{d}$ and the Vickrey-Clarke-Groves (VCG) mechanism is then used to compute a payment for every node on LCP. The source node $\mathbf{s}$ decides to perform routing only when the total VCG payment to all relay nodes is at most $\nu(\mathbf{s}, \mathbf{d})$. We show that when $\nu(\mathbf{s}, \mathbf{d})$ is infinite, the VCG mechanism is still strategyproof. We then show by example that, however, when $\nu(\mathbf{s}, \mathbf{d})$ is finite, this simple VCG mechanism will *not* induce truthfulness from agents. We then design strategyproof routing schemes for this scenario. In our mechanism, the payment to each node is still VCG payment, but the rule by the source node for deciding whether to conduct routing is carefully designed to ensure strategyproofness. Besides strategyproof schemes, we design a routing scheme using a Nash Equilibrium and show how to efficiently find a Nash Equilibrium that maximizes the payment.

Our routing scheme takes into account the unreliability of wireless links and each node will declare a quality of service (QoS) for forwarding the packets and the marginal cost for providing such QoS. Thus, a selfish agent could manipulate multiple parameters such as its service cost and QoS. The reliability QoS and cost information will be collected from individual agents to the trusted authority. The final routing should always ensure a reliable delivery of packets from the source to the destination. We study two reliable routing models: link layer reliability (where retransmissions start from the failed any link) and transport layer reliability (where retransmissions start from the source node). Strategyproof routing schemes are designed for almost all possible combinations of reliable routing models, one/multi-parameter models, and valuation models. We show that in our schemes, fulfilling the declared QoS is the optimal strategy for each agent. Our mechanisms can deal with both hidden-information and hidden-action and they can be easily integrated with existing routing protocols. Here we mainly focus on reliability as a QoS measurement. We leave it as a future work to design strategyproof mechanisms that can ensure other QoS measurements such as (soft or hard) deadline requirements.

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 payments by our mechanisms are only slightly larger than the *actual cost* incurred by all intermediate nodes even the costs of different nodes could vary significantly.

The rest of the paper is organized as follows. In Section II, we present our network model, and the problems to be studied. In Section III, we present our methods that can efficiently deal with the hidden actions and hidden information and ensure the reliable routing in the link layer. In Section IV, we present routing mechanisms for the reliable transport layer routing. We briefly discuss the protocols when senders could be selfish in Section V. We report our simulation results in Section VI, review the related work in Section VII, and conclude the paper in Section VIII.

II. Preliminaries, Network Model

A. Network Model

We assume that there is a set V of $n = |V|$ of devices (called nodes hereafter) and each node is assigned a unique ID $i \in [1, n]$. For simplicity, we assume that the nodes are static or can be viewed as static for a long-enough time period. 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 either the sender $\mathbf{s}$ or the receiver $\mathbf{d}$ of a communication, and the agents are the intermediate nodes capable of forwarding packets between the communication endpoints. We also assume that in the network, there is a trusted authority that will perform all payment computations and is in charge of payment management. For example, the gateway node(s) in a mesh network can serve as a trusted authority [12]. For simplicity, we assume that all communication sessions have the trusted authority as the destination node. Thus, the trusted receiver will be called *principal* hereafter. Notice that it has been proved in [39] that no mechanism is strategyproof and budget-balanced when the shortest path is used for routing. We will later discuss on the scenario when the principal could also be selfish.

The sender will get a value $\nu(\mathbf{s}, \mathbf{d})$ 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 $\nu(\mathbf{s}, \mathbf{d}) - \chi$. Obviously, the sender will conduct routing only if $\nu(\mathbf{s}, \mathbf{d}) - \chi \geq 0$. As in [12], before all nodes will participate in finding a routing path from the sender to the destination, the sender has to declare its willing payment ν. The sender may also have a minimum quality of service $\theta(\mathbf{s}, \mathbf{d})$ for the routing path. Here the QoS provided by a path is often defined as the minimum quality of service provided by all nodes on the path.

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 wireless 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 may be difficult to separate them in non-cooperative networks with selfish nodes.

In addition, the cost incurred by a node u may also depend on the quality of service which is provided in sending packets to v. We assume that *all agents* can provide a given set $\mathcal{Q} = \{\theta_1, \theta_2, \cdots, \theta_\kappa\}$ of κ forwarding services. For simplicity, we assume that the QoS θ_i is better than θ_j for $i > j$. We also assume that *dropping packets* is the worst QoS θ_1. For example, the services could be simply $\{$*always drop*, *always forward*$\}$. The quality of service provided by node u will affect 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*, which could be public information. For example, when the available services are $\{\theta_1 =$ *always drop*, $\theta_2 =$ *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. 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 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) **node cost model**, where the cost $\mathbf{c}_j(u,v)$ is dependent on u while is independent of the node v; (2) **link cost model**, where the cost $\mathbf{c}_j(u,v)$ is dependent on both nodes u and 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 private *type* t_i: (1) the cost $\mathbf{c}_j(v_i,w)$ to provide a forwarding service to every outgoing neighbor w with a QoS θ_j, (2) the corresponding service reliability $\alpha_j(v_i,w)$. We assume that the physical link reliability $\beta(u,v_i)$ is a public information.

B. Games and Mechanism Design

There are two different types of games, namely, *simultaneous games*, and *sequential games*. In this paper, we focus on designing mechanisms for routing in multi-hop wireless networks. A mechanism $M = (\mathcal{O}, \mathcal{P})$ is composed of two parts: an output function $\mathcal{O}$ that maps a declared type vector $\tau = (\tau_1, \cdots, \tau_n)$ by all nodes to a path o connecting the sender and the receiver and a *payment* function $\mathcal{P}$ that decides the monetary payment $\mathcal{P}_i(\tau)$ for every agent i. Here τ_i may be different from the actual type t_i for node i. Each node i has a valuation function $\nu_i(o, \mathbf{t})$ that expresses its preference over different outcomes. Node i's *utility* (also called *profit*) is $\mu_i(\mathcal{O}(\tau), \mathbf{t}) = \nu_i(\mathcal{O}(\tau), \mathbf{t}) + \mathcal{P}_i(\tau)$, given the declared vector type τ. A node i is said to be *rational* if it always chooses its strategy τ_i that maximizes its utility μ_i. A strategy by a node i is a *dominant strategy* of i if it maximizes its utility no matter what other nodes do.

Let $\tau_{-i} = (\tau_1, \cdots, \tau_{i-1}, \tau_{i+1}, \cdots, \tau_n)$, *i.e.*, the strategies of all other agents except i. When every other agent $j \neq i$ chooses type τ_j and agent i chooses t_i, we denote the action profile as $\tau|^i t_i = (\tau_1, \tau_2, \cdots, \tau_{i-1}, t_i, \tau_{i+1}, \cdots, \tau_n)$. A mechanism is *strategyproof* if for every agent i, revealing its true type t_i is a dominant strategy. For games studied in this paper, the valuation of an agent is actually a random variable: it depends on how the forwarding game is materialized. For example, when an intermediate node u has to ensure that the data packet reaches its next hop neighbor on a routing path, the actual number of transmissions is a variable with *geometric distribution*. In certain mechanisms, even the payment to an agent could be a variable depending on the actual outcome of the game. It is thus often not interesting to require that the mechanism is strategyproof for *every possible* value of the valuation materialized. In this paper, we are only interested in mechanisms $M = (\mathcal{O}, \mathcal{P})$ that satisfy the following:

1) **Incentive Compatibility with Expectation (ICE)**: Revealing the true type is a dominant strategy, *i.e.*, $\forall i$, $\forall \tau$,
$$\mathscr{E}\left(\nu_i(\mathcal{O}(\tau|^i t_i), \mathbf{t}) + \mathcal{P}_i(\tau|^i t_i)\right) \geq \mathscr{E}\left(\nu_i(\mathcal{O}(\tau), \mathbf{t}) + \mathcal{P}_i(\tau)\right).$$
2) **Individual Rationality with Expectation (IRE)**: Each agent has a non-negative *expected* utility, *i.e.*, $\forall i$, $\forall \tau$,
$$\mathscr{E}\left(\nu_i(\mathcal{O}(\tau|^i t_i), \mathbf{t}) + \mathcal{P}_i(\tau|^i t_i)\right) \geq 0.$$

Here $\mathscr{E}(X)$ is the expected value of a variable X. A mechanism is called *strategyproof with expectation* (or simply strategyproof) if it satisfies ICE and IRE. In other words, we will not consider any betting preferences of agent with regard to uncertain outcomes. Or equivalently, the behavior of every agent can be predicted based on expected value of its utility.

Arguably the generalized Vickrey-Clarke-Groves (VCG) mechanism [9], [17], [37] is the most positive result in mechanism design. Besides the dominant strategy design, there are several other concepts for mechanism design, *e.g.*, Nash equilibrium. A vector of action profile $\mathbf{a} = (a_1, a_2, \cdots, a_n)$, where agent i plays action a_i, 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.

For sequential games, several refined equilibriums have been proposed in the literature, *e.g.*, *subgame perfect equilibrium* (**SPE**). 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.* [31]) as follows. First, one determines the optimal strategy of the player who makes the last move of the game. Then, the optimal action of the one next to last player is determined by assuming the last player's action is given. The process continues until all player's actions have been determined.

C. 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 participating nodes will have the following components.

1. Collect Information: The sender announces its required QoS and a total willing payment ν for sending a unit amount of data to $\mathbf{d}$. The principal (*e.g.*, the receiver as a trusted authority) then asks every node u in the network to declare what QoS θ_r it will provide for forwarding, the cost $\overline{\mathbf{c}}(u,v)$ for forwarding a unit amount of data to a neighbor v by this QoS θ_r, the corresponding service reliability $\overline{\alpha}_r(u,v)$, and the observed link reliability $\overline{\beta}(w,u)$ for every neighbor w.

Here we assume that every agent will encrypt its declaration and then digitally sign the encrypted declaration. We assume that the encrypted declaration of any intermediate node will be forwarded to the principal as being done in [46].

2. Select Path: This is called the *routing subgame* in [46]. Based on the information collected from all nodes, the trusted principal then finds a path $v_{i_1}v_{i_2}\cdots v_{i_h}$ for routing with a certain minimum QoS, *i.e.*, every node on the path can provide at least this minimum QoS θ_r. Here $\mathbf{s}=v_{i_1}$, $\mathbf{d}=v_{i_h}$. Principal also computes a certain incentive given to each intermediate node on the path to compensate the cost incurred for forwarding data.

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 $v_{i_1}v_{i_2}\cdots v_{i_h}$ is used for routing where $\mathbf{s}=v_{i_1}$, $\mathbf{d}=v_{i_h}$ and direct links $v_{i_j}v_{i_{j+1}}$, $1\le j\le h-1$, 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 $\overline{\mathbf{c}}(v_{i_j},v_{i_{j+1}})$). 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 *dropping packets* as one of the possible quality of services. 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 [46]. In [46], Zhong *et al.* essentially assume that there are two quality of services: $\{$*drop*,*forward*$\}$.

4. Materialize Payment: Depending on the outcome of the forwarding subgame, the principal would materialize the incentive computed during the routing subgame, *e.g.*, transfer the monetary value to intermediate nodes.

In this paper, we will study a number of routing protocols using the above general framework. Specifically, the following two routing scenarios will be studied.

1. Reliable link layer: 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.

2. Reliable transport layer: if the transmission from a node v_{i_j} to node $v_{i_{j+1}}$ is not successful, node v_{i_j} will discard the data and then the source node $\mathbf{s}$ will start the retransmission after receiving the time-out signal. This is called the *TCP model*. Assume that the sender has certain implemented mechanism to know if the data arrives at the target node.

D. Least Cost Path Construction

First, given the information of all agents, we need to find a path to connect the source node $\mathbf{s}$ and the target node $\mathbf{d}$. The reliable minimum cost unicast routing problem is to find a path connecting $\mathbf{s}$ and $\mathbf{d}$ such that the *expected* total cost spent by all nodes (including the retransmissions) is minimized, given the cost $\mathbf{c}(u,v)=\mathbf{c}_j(u,v)$ for each link (u,v), and the probability $\delta(u,v)=\beta(u,v)\cdot\alpha_j(u,v)$ that the transmission over the link is successful. Here we assume that node u will provide forwarding to node v with quality of service θ_j. The construction of the least cost path has been studied recently in [3] when reliable link layer is implemented and in [10] when reliable transport layer is implemented. Given a simple path $\mathbf{P}=v_{i_1}v_{i_2}\cdots v_{i_h}$ connecting $\mathbf{s}$ and $\mathbf{d}$, where $\mathbf{s}=v_{i_1}$, $\mathbf{d}=v_{i_h}$, we briefly show how to compute the *expected* cost, denoted as $\mathscr{E}(\mathbf{P})$, of this path under both link-layer and transport-layer reliability models

When a link-layer reliability is implemented, obviously, the *expected* cost of path $\mathbf{P}$ is $\mathscr{E}(\mathbf{P})=\sum_{j=1}^{h-1}\frac{1}{\delta(v_{i_j},v_{i_{j+1}})}\cdot\mathbf{c}(v_{i_j},v_{i_{j+1}})$. Here $\frac{1}{\delta(v_{i_j},v_{i_{j+1}})}$ is the *expected* number of total transmissions over link $(v_{i_j},v_{i_{j+1}})$ including the initial transmission and all retransmissions.

When a transport-layer reliability is implemented, let $\mathbf{P}|_{i_j}$ be the subpath of $\mathbf{P}$ from node $\mathbf{s}=v_{i_1}$ to node v_{i_j}. The *expected* cost of path $\mathbf{P}$ under transport-layer reliability model is then

$$\mathscr{E}(\mathbf{P})=\frac{\mathscr{E}(\mathbf{P}|_{i_{h-1}})+\mathbf{c}(v_{i_{h-1}},v_{i_h})}{\delta(v_{i_{h-1}},v_{i_h})}=\sum_{j=2}^{h}\frac{\mathbf{c}(v_{i_{j-1}},v_{i_j})}{\prod_{t=j}^{h}\delta(v_{i_{t-1}},v_{i_t})}$$

The LCP path in transport-layer reliability model thus satisfies that the subpath in $\mathbf{LCP}(\mathbf{s},\mathbf{d})$ from $\mathbf{s}=v_{i_1}$ to any node v_{i_j} is also the LCP path $\mathbf{LCP}(\mathbf{s},v_{i_j})$.

Thus, the LCP can always be found in time $O(m+n\log n)$ under both the reliable link-layer model or the reliable transport-layer model for a network of n nodes and m links.

III. Link Layer Reliable Routing

A. Known Information and Hidden Action

We first consider the case when all information about the selfish nodes are already known by the principals. Let $\theta_r=\theta(\mathbf{s},\mathbf{d})$ be the minimum quality of service required by the sender. Thus, the only actions taken by a selfish node are (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 forwarding subgame) is then to ensure that every selfish agent fulfill its declared forwarding service. Algorithm 1 presents our routing scheme. In Algorithm 1, the parameter $\eta \geq 0$ is a control parameter used to ensure that each relay node will get a positive profit with high probability when it relays enough packets. For example, we can set $\eta = 1/2$. If $\eta = 0$, the expected profit of all nodes is 0.

Algorithm 1 Link Layer Reliable Routing with Known Info

Input: graph G, QoS θ_r required by sender and its valuation $\nu(\mathbf{s}, \mathbf{d})$ for a unit amount of data, and a fixed constant 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 $\mathscr{E}(\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d}))$ be its expected cost.

2: If $(1 + \eta) \cdot \mathscr{E}(\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})) \leq \nu(\mathbf{s}, \mathbf{d})$, the sender 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 sender 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

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

where v is the next-hop node of u. This payment will be materialized only if the forwarding subgame is finished.

3: **Forwarding subgame:** When an intermediate node v_{i_j} $(2 \leq j \leq h-1)$ received a data packet by the sender, 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 reduced its QoS, every node on the path will *not* receive any payment.

Theorem 1: If a node u forwards N units amount of data truthfully, the probability that node u runs deficit is at most $\frac{1-\delta}{1-\delta+N\cdot\eta\cdot\delta}$ where $\delta = \alpha_r(u, v) \cdot \beta(u, v) \leq 1$.
See proof [25].

The above 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 $\frac{1-\delta}{1-\delta+N\cdot\eta\cdot\delta}$. If we require that the probability that an intermediate node loses money is at most ϵ, then forwarding $N = \lceil \frac{(1-\delta)(1-\epsilon)}{\eta\delta\epsilon} \rceil$ units of data is sufficient. Keep in mind that the *actual* cost of an intermediate node by forwarding data correctly is a *random variable* with geometric distribution, which is memoryless.

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.
See proof [25].

Lemma 3: When an intermediate node does not keep forwarding the packet until it is received by its next-hop neighbor, its expected profit will be smaller.

Proof: Assume that for kth unit of data, node u tried U_k transmissions (including the initial transmission and all retransmissions). Here node u could have selfishly stopped the transmission after U_k transmissions although the packet has not been received correctly. Assume that the cost for one transmission is 1 unit. Then the total cost for N units of data is $\sum_{i=1}^{N} U_i$. Let X_k be a random variable denoting whether the kth data packet is received correctly or not by using U_k transmissions. Thus, the payment by the sender to this relay node for a successful reception is $X_k \cdot (1 + \eta)/\delta$. Clearly $\mathbf{Pr}\,(X_k = 1) = 1-(1-\delta)^{U_k}$. Thus, the expected total payment by this sequence of strategies is $\sum_{k=1}^{N} \mathscr{E}(X_k) \cdot (1 + \eta)/\delta = \frac{1+\eta}{\delta} \cdot \sum_{k=1}^{N} \left(1 - (1 - \delta)^{U_k}\right) \leq \frac{1+\eta}{\delta} \cdot \sum_{i=1}^{N} (U_i \delta) = (1 + \eta) \cdot \sum_{i=1}^{N} U_i$. Thus, the expected profit by the intermediate node u is at most $(1+\eta) \cdot \sum_{i=1}^{N} U_i - \sum_{i=1}^{N} U_i = \eta \cdot \sum_{i=1}^{N} U_i$. Let us see what the profit node u would have if it used strategies $U_k' \geq U_k$ such that the kth data is received correctly. Clearly, node u will *not* change the cost of successful transmissions where $X_k = 1$. Let m be the number of unsuccessful transmissions where $X_i = 0$ for $1 \leq i \leq N$. Node u will keep retransmitting till these packets are received correctly. Let U_i' be the actual number of retransmissions incurred by node u for the ith packet. Then $\sum_{i=1}^{N} U_i' - U_i$ is the *additional cost* by node u and $m \cdot \frac{1+\eta}{\delta}$ is the additional payment node u will get from these additional m successful transmissions. Notice that the *expected* additional cost for getting successful m transmissions is still $m \cdot \frac{1}{\delta}$ since the geometry distribution is memory-less. Thus, a node will improve the *expected* profit by $m \cdot \frac{1+\eta}{\delta} - m \cdot \frac{1+\eta}{\delta} = m \cdot \frac{\eta}{\delta}$, if it keeps retransmitting unsuccessful packets till they are received correctly. ■

B. Strategyproof Routing with Infinite Valuation

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, and the valuation ν of the sender is infinite. Naturally, we will use the VCG mechanism to induce the truthfulness from all relay agents; and the sender 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. We will show that this mechanism is *not* strategyproof if the sender has a finite valuation ν.

We first study whether the routing scheme described in Algorithm 2 is strategyproof for relay agents when the sender

Algorithm 2 Naive Link Layer Reliable Routing

1: **Routing subgame:** First the principal asks the sender to declare its willing payment $\nu(\mathbf{s}, \mathbf{d})$. Then it asks every node u, to declare its cost (denoted by $\bar{\mathbf{c}}_r(u, v)$) 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 $\overline{\alpha}_r(u, v)$. For all links (u, v), we define its *weight* $\omega(u, v)$ as $\bar{\mathbf{c}}_r(u, v)/(\overline{\alpha}_r(u, v) \cdot \beta(u, v))$. Let G' be the resulted graph and $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})$ be the least cost path in G' to connect $\mathbf{s}$ and $\mathbf{d}$. Let $\mathbf{P}_{G' \setminus u}(\mathbf{s}, \mathbf{d})$ be the least cost path in 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).

2: 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 $\mathcal{P}^{\mathrm{VCG}}(u)$ is always 0. For each node u selected on this path, its VCG payment $\mathcal{P}^{\mathrm{VCG}}(u)$ is

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

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.

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

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

the sender 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 sender will not initiate the routing, and the output $\mathbf{o}$ is $\emptyset$ and the final payment to every node is 0.

4: **Forwarding subgame:** It is similar to the forwarding subgame of Algorithm 1. The principal materializes the VCG payment to every intermediate agent v_{i_j} only if the target node $\mathbf{d}$ received the data correctly with the given QoS.

has an infinite valuation $\nu(\mathbf{s}, \mathbf{d})$, *i.e.*, the sender would pay whatever it costs to get the data transmitted.

Theorem 4: Assume that $\nu(\mathbf{s}, \mathbf{d})$ is infinity. Truth-telling is a *dominant strategy* and forwarding truthfully is a SPE (*subgame perfect equilibrium*) for all intermediate relay nodes when VCG mechanism is used for routing with link layer reliability.

Proof: To prove the above statement, we need to show that no node can involve in the following misbehavior:

1) declaring the cost truthfully, but providing lower QoS forwarding service, or
2) declaring the cost falsely, although providing the correct (or better) QoS forwarding service, or
3) declaring the cost falsely, and providing a lower QoS forwarding service.

The property that the truth-telling is a dominant strategy (*i.e.*, misbehavior (2) cannot happen) directly follows from the strategyproofness of the VCG mechanism since our final path will minimize the total expected cost of all agents, *i.e.*, maximizing the total valuations of all agents. Similar to the proof of Theorem 2, we can prove that forwarding truthfully (*i.e.*, misbehavior (1) and (3) cannot happen) is a SPE. ■

When the sender has an infinite valuation $\nu(\mathbf{s}, \mathbf{d})$, Algorithm 2 is budget-balanced, *i.e.*, the money charged from the sender is same as the total money paid to all intermediate relay agents.

C. Strategyproof Routing with Finite Valuation

This subsection is devoted to designing a strategyproof routing scheme when the sender has a finite valuation.

Notice that routing scheme described in Algorithm 2 fails to induce the truthfulness when the valuation $\nu(\mathbf{s}, \mathbf{d})$ is finite (see example [25]), 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 5: If the sender performs routing only when a condition $\mathscr{C}$ is satisfied, to induce truthfulness from agents, the condition $\mathscr{C}$ should not depend on the information of agents who are selected by the strategyproof mechanism for 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 sender. 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 $\mathscr{C}$ when the sender should perform the routing. Obviously, the condition $\mathscr{C}$ should not use any information of agents on path $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})$.

Lemma 6: If a condition $\mathscr{C}$, by which the sender 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 induces the truthful information declaration from all nodes.

See proof [25].

In next, we discuss in detail how our routing mechanism is designed based on whether we have link cost model or node cost model.

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 (*i.e.*, no power adjustment is used), 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 total VCG payment to all intermediate nodes.

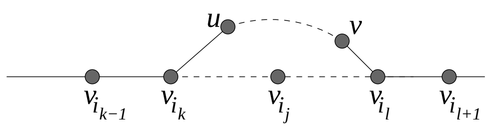


Fig. 1. Example of a bridge for node v_{i_j} on the LCP $v_{i_1} v_{i_2} \cdots v_{i_j} \cdots v_{i_h}$.

Assume the LCP 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. In node cost model, the (internal) cost $\|B\|_I$ of a bridge B is defined as the total cost of all internal nodes of B. In link cost model, $\|B\|_I$ is the cost of all links in B. Given a path $\mathbf{P}$ and two nodes v_{i_a}, v_{i_b} on $\mathbf{P}$, let $\mathbf{B}_{\min}^{i_a,i_b}$ be the bridge with the minimum cost, starting at node v_{i_a} and ending at node v_{i_b}. Observe that $\mathbf{P}_{G' \backslash \mathbf{LCP}}(\mathbf{s}, \mathbf{d})$ is a bridge of any internal node v_{i_j}. Clearly, the VCG payment to any 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 cost $\|\mathbf{B}_{\min}(v_{i_j})\|_I = \min_{i_a < i_j < i_b} \mathbf{B}_{\min}^{i_a,i_b}$. Thus, under both cost models, we have

Lemma 7: $\forall v_{i_j} \in \mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})$, $\mathcal{P}^{\text{VCG}}(v_{i_j}) \le \|\mathbf{B}_{\min}(v_{i_j})\|_I$.

Notice that $\|\mathbf{B}_{\min}(v_{i_j})\|_I$ is not affected by any node on the LCP $v_{i_1} v_{i_2} \cdots v_{i_j} \cdots v_{i_h}$. See Figure 1. Moreover, the bridge $\mathbf{B}_{\min}(v_{i_j})$ can be found in time $O(n^3)$ using all pairs shortest path in $G' \backslash v_{i_j}$. Thus, based on Lemma 6, the following mechanism is strategyproof when the sender has a finite valuation ν, and each node has a cost (independent of out-going neighbors) for forwarding.

1) It uses the least cost path for routing.
2) The sender decides to conduct routing only when
$$\sum_{j=2}^{h-1} \|\mathbf{B}_{\min}(v_{i_j})\|_I \le \nu(\mathbf{s}, \mathbf{d}). \tag{2}$$
3) If the routing is performed, the payment to each relay node is computed using VCG payment.

Then this criterion will induce the truthfulness from all agents. Algorithm 3 summarizes our method.

For some networks, it is possible that there may have *no bridge* at all for some intermediate node. We show that this happens if and only if the original network is **not** 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, and vice versa. It is a not difficult to show that the VCG payment exists if and only if 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 at all and have to rely on some other mechanism such as the Nash Equilibrium which will be discussed later. Then we have the following lemma

Lemma 8: For nodal cost, Algorithm 3 works correctly as long as the network is 2-connected. VCG payment is a finite value whenever the network is 2-connected.

2) Routing Mechanism with Link Cost: If a relay node has different costs for relaying to different nodes, we need a different criterion for deciding whether to conduct the routing. Again take Figure 1 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 link (v_{i_k}, u) contributes to the value of $\mathbf{B}_{\min}(v_{i_j})$. We then study how to design criterion when the sender should perform routing in this case. We define the bridge similarly as previous case. Let $\|\mathbf{B}\|$ denote the total cost of all links on a bridge $\mathbf{B}$. Let $\mathbf{B}_{\min}(v_{i_j})$ be the bridge for v_{i_j} with the minimum cost. Then, we have $\mathcal{P}^{\text{VCG}}(v_{i_j}) \le \|\mathbf{B}_{\min}(v_{i_j})\|$. Notice that if we still use criterion (inequality (2)), *i.e.*, the sender performs routing if $\sum_{j=2}^{h-1} \|\mathbf{B}_{\min}(v_{i_j})\| \le \nu(\mathbf{s}, \mathbf{d})$, an internal node may lie to improve its utility as follows. Assume that sender 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 sender will perform routing and all nodes on **LCP** will improve its utility.

Thus, we get a condition as follows. Instead of considering bridges that could start with any internal node on the LCP, we will only consider the bridge that starts with the source node $\mathbf{s}$. For an intermediate node v_{i_j}, a simple path $\mathbf{B}^{\mathbf{s}}$ 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 $\mathbf{B}^{\mathbf{s}}_{\min}(v_{i_j})$. Then, the sender decides to perform routing only when
$$\sum_{j=2}^{h-1} \|\mathbf{B}^{\mathbf{s}}_{\min}(v_{i_j})\| \le \nu(\mathbf{s}, \mathbf{d}). \tag{3}$$
is satisfied, and the payment to each intermediate node is just its VCG payment.

In summary, we have the following strategyproof routing scheme (Algorithm 3) for both the case of node cost and the case of link cost when the valuation of the sender is finite.

Algorithm 3 Strategyproof Link Layer Reliable Routing

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

1: **Routing subgame:** Same as Algorithm 2, the principal first asks the sender to declare its willing payment $\nu(\mathbf{s}, \mathbf{d})$ and then it collects the declared cost $\bar{\mathbf{c}}_r(u, v)$ by u. It constructs the graph G' and finds 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 each node u on the LCP, the principal computes its VCG payment $\mathcal{P}^{\text{VCG}}(u)$ as
$$\mathcal{P}^{\text{VCG}}(u) = \|\mathbf{P}_{G' \backslash u}(\mathbf{s}, \mathbf{d})\| - \|\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})\| + \omega(u, v)$$
where v is the next-hop node of u on the path $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})$.

3: The sender decides to conduct the routing based on the following conditions:
- **Link Cost Case:** When the private of information (service cost and service reliability) of any agent is *dependent* on its out-going neighbor, the sender starts routing if $\sum_{j=2}^{h-1} \|\mathbf{B}^{\mathbf{s}}_{\min}(v_{i_j})\| \le \nu(\mathbf{s}, \mathbf{d})$.
- **Nodal Cost Case:** When the private of information of every agent is *independent* on its out-going neighbor, the sender starts routing if $\sum_{j=2}^{h-1} \|\mathbf{B}_{\min}(v_{i_j})\|_I \le \nu(\mathbf{s}, \mathbf{d})$.

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

4: **Forwarding subgame:** It is similar to the forwarding subgame of Algorithm 1. The principal materializes the VCG payment $\mathcal{P}^{\text{VCG}}(v_{i_j})$ to every intermediate agent v_{i_j} when the target node $\mathbf{d}$ received the data correctly.

Based on Lemma 6, we have

Theorem 9: Algorithm 3 is strategyproof for routing subgame and truthfully forwarding by every intermediate node is a subgame perform equilibrium.

It is easy to show that the conditions, used in Algorithm 3 by the sender to decide when to perform routing, are tight.

Recall that in the previous node-cost case, the 2-connectivity is a necessary and sufficient condition for Algorithm 3 to work correctly. When a node v has different costs for sending data to different neighbors, it is easy to show that the sufficient and necessary condition for having a finite VCG payment in this case is that, for every node v, the graph H_{-v} (by removing v and all its incident links from G') is still connected. In other words, 2-connectivity is still necessary and sufficient condition for having a finite VCG payment. Unfortunately, we need a much stronger condition to ensure that there is a bridge starting with the source node s. We can construct a network example, such that VCG payment is finite, but we do not have $\mathbf{B}^{\mathbf{s}}_{\min}(v_{i_j})$ for some node: it has 5 nodes v_1, v_2, v_3, v_4, and v_5 and 7 links, $v_i v_{i+1}$ ($i = 1, 2, 3, 4$) and $v_i v_{i+2}$ ($i = 1, 2, 3$). Clearly, when source is v_1 and destination is v_5, we do not have bridge $\mathbf{B}^{v_1}_{\min}(v_3)$. By definition, the sufficient and necessary condition for having a bridge (starting with the source $\mathbf{s}$) for every node v_{i_j} on the LCP path from the source $\mathbf{s}$ to the destination node $\mathbf{d}$ is that $G' \setminus \cup_{k=1}^{j} v_{i_k}$ has a path from v_{i_0} to some v_{i_b} with $k < b \le t$, where $v_{i_0} v_{i_1} v_{i_2} \cdots v_{i_t}$ is the LCP path connecting nodes $\mathbf{s} = v_{i_0}$ and $\mathbf{d} = v_{i_t}$. Thus, we have

Lemma 10: For link cost, VCG payment is a finite value whenever the network is 2-connected, but 2-connectivity is not sufficient for having a bridge $\mathbf{B}^{\mathbf{s}}_{\min}(v_{i_j})$ for every v_{i_j} on the LCP.

D. ϵ-Nash Equilibrium based Routing Subgame

So far, the routing schemes presented in this section will induce the truthfulness from agents. Possible disadvantages of ensuring the truthfulness are that (1) the sender could lose some opportunities to send data from the source to the target, and (2) the payment by the sender 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 follows.

Let $\tau = (\tau_1, \tau_2, \cdots, \tau_n)$ 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 [29] 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 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 11: For any declaration vector τ that is a Nash equilibrium, the optimal path $\mathcal{O}(\tau)$ is the same as the real optimal path $\mathcal{O}(\mathbf{t})$ when all agents are truthful.

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 $\overline{\mathbf{c}}_r(u, v)$ and the service reliability $\overline{\alpha}_r(u, v)$ for each of its out-going neighbor v. Define graph $G' = (V, E)$ where the weight of a link (u, v) is $\overline{\mathbf{c}}_r(u, v)/(\overline{\alpha}_r(u, v) \cdot \beta(u, v))$. 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.

2: If $\mathscr{E}(\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})) \le \nu(\mathbf{s}, \mathbf{d})$, the sender will decide to conduct the routing. Otherwise, the sender 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

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

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

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

Proof: It is equivalent to prove that any node selected in $\mathcal{O}(\mathbf{t})$ (called legitimate) is still selected in $\mathcal{O}(\tau)$. Notice if a legitimate node is not selected, it will reduce its declaration to its true type such that it will be selected. The proof is similar to the proof in [29] and thus is omitted here. ■

Lemma 11 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 a 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 $x_{i_j} = \frac{\overline{\mathbf{c}}_r(v_{i_j}, v_{i_{j+1}})}{\overline{\alpha}_r(v_{i_j}, v_{i_{j+1}}) \cdot \beta(v_{i_j}, v_{i_{j+1}})}$. This is the *expected* cost of node v_{i_j} viewed by the principal under the Nash equilibrium declaration τ, and also is the payment that node v_{i_j} will receive under our new routing scheme. Based on Lemma 11, 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 $\mathbf{B}^{i_a, i_b}_{\min}$,

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

The declaration must satisfy the IR property, *i.e.*, $\forall v_{i_j}$,

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

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 Equilibriums, 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 constraints 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. This shows that maximum Nash Equilibrium can be solved by the linear programming. However, solving linear programming with up to $O(n)$ variables and $O(n^2)$ constraints could be quite costly. We present a simple method (Algorithm 5) that can elegantly compute a maximum Nash Equilibrium that also has some practical meanings. The proof of its correctness is omitted here due to space limit. We also leave it as future work to design protocols that can reduce the number of possible Nash Equilibria.

Algorithm 5 Sequential Algorithm to Find a Maximum Nash

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

Output: A maximum Nash Equilibrium.

Build the graph G and compute the least cost path $\mathbf{P}_G(\mathbf{s}, \mathbf{d})$, denoted as $v_{i_1} v_{i_2} \cdots v_{i_j} \cdots v_{i_h}$.

for $k = 1$ to h **do**

 Compute the VCG payment $\mathcal{P}^{\text{VCG}}(v_{i_k})$ for node v_{i_k} using updated cost for previous links as

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

 Adjust the cost for link $(v_{i_k}, v_{i_{k+1}})$ as $\mathbf{c}_r(v_{i_k}, v_{i_{k+1}}) \leftarrow \mathcal{P}^{\text{VCG}}(v_{i_k}) \alpha_r(v_{i_j}, v_{i_{j+1}}) \beta(v_{i_j}, v_{i_{j+1}})$. This new cost is used for compute the payment for subsequent links.

Return $x_{i_k} = \frac{\mathbf{c}_r(v_{i_k}, i_{k+1})}{\alpha_r(v_{i_j}, v_{i_{j+1}}) \beta(v_{i_j}, v_{i_{j+1}})}$ for $1 \le i \le h$. Here x_{i_k} will be the cost declared by node v_{i_k} and it is also the payment to node v_{i_k}.

IV. Transport Reliable Routing

Here we present several routing schemes to ensure the transport layer reliability. ~~Recall that in link layer, retransmission starts from the node which loses the packet, while in transport layer, TCP layer, retransmission starts from the starting node. Thus, we often will have different paths that optimize the total relay costs by intermediate relay nodes in link-layer reliable routing, and in transport-layer reliable routing.~~ We will show that the widely celebrated VCG mechanism does not induce the truth-telling action from all selfish agents when an agent can manipulate its declared cost and also the reliability of the incident links. We prove that the truth-telling action is still dominant strategy when an agent can only declare its service cost.

A. Known Information and Hidden Action

Here we assume that the principal knows all information about the selfish nodes and also the minimum quality of service $\theta_r = \theta(\mathbf{s}, \mathbf{d})$ required. The main goal of the routing scheme is then to ensure that every selfish agent fulfills its declared forwarding service. Our routing scheme is similar to Algorithm 1. The difference is how the path with the minimum expected cost under reliable transport model is found and how much each node should be paid. Dong *et al.* [10] presented a polynomial-time method to find such optimal path, which cannot be found using traditional shortest path algorithm. Let $\mathbf{P}_{G'}^T(\mathbf{s}, \mathbf{d})$ be the path with the minimum cost under the TCP model and $\mathscr{E}^T(\mathbf{P})$ be the expected cost of a path $\mathbf{P}$ under TCP model. Notice that the expected cost of a path $\mathbf{P} = v_{i_1} v_{i_2} \cdots v_{i_h}$ under the TCP model is $\mathscr{E}^T(\mathbf{P}) = \sum_{j=2}^{h} \frac{\mathbf{c}(v_{i_{j-1}}, v_{i_j})}{\prod_{t=j}^{h} \delta(v_{i_{t-1}}, v_{i_t})}$.

The routing mechanism works as follows, with a fixed constant $\eta > 0$ as a control parameter:

1) If $(1+\eta) \cdot \mathscr{E}^T(\mathbf{P}_{G'}^T(\mathbf{s}, \mathbf{d})) \le \nu(\mathbf{s}, \mathbf{d})$, the sender will decide to conduct the routing. Here $\eta > 0$ is a fixed constant.
2) The payment to a node not on the path is 0. For each node $v_{i_{j-1}} \in \mathbf{P}_{G'}^T(\mathbf{s}, \mathbf{d})$, its payment is $\frac{\mathbf{c}(v_{i_{j-1}}, v_{i_j})}{\prod_{t=j}^{h} \delta(v_{i_{t-1}}, v_{i_t})}(1+\eta)$.

Notice that the *expected* profit of a node $v_{i_{j-1}}$ that transmits N units amount of data is $\eta N \frac{\mathbf{c}(v_{i_{j-1}}, v_{i_j})}{\prod_{t=j}^{h} \delta(v_{i_{t-1}}, v_{i_t})}$. Similar to Theorem 1 and Theorem 2, we have the following theorems.

Theorem 12: If all intermediate nodes forward N units amount of data truthfully, the probability that a node v_{i_j} runs deficit is no more than $\frac{\sigma^2}{\sigma^2 + N \cdot \eta \cdot \mu}$, where σ, μ are the standard deviation and mean of the random variable X, denoting the total number of transmissions by node v_{i_j} to let the receiver get the data correctly. Here η is a fixed constant.

Theorem 13: For all agents selected on the path $\mathbf{P}_{G'}(\mathbf{s}, \mathbf{d})$, forwarding data using the agreed QoS θ_r is a SPE.

B. Hidden Cost 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. We will focus on the *one-parameter* game where *the service cost is private* while all other information is public.

Since only the cost $\mathbf{c}(u, v)$ is private, the objective function $\mathscr{E}^T(\mathbf{P}) = \sum_{j=2}^{h} \frac{\mathbf{c}(v_{i_{j-1}}, v_{i_j})}{\prod_{t=j}^{h} \delta(v_{i_{t-1}}, v_{i_t})}$ that we want to minimize by finding a path $\mathbf{P}$ is utilitarian. Notice that the valuation $\nu_{i_j}(\mathbf{LCP}, \mathbf{t}) = \frac{\bar{\mathbf{c}}(v_{i_j}, v_{i_{j+1}})}{\prod_{t=j}^{h} \delta(v_{i_t}, v_{i_{t+1}})}$ of a node v_{i_j} on $\mathbf{LCP}$ not only depends on its own cost but also depends on the reliability $\delta(v_{i_t}, v_{i_{t+1}})$ of all links $v_{i_t} v_{i_{t+1}}$ for $t \ge j$. This is a sharp contrast to the majority previous studies on designing strategyproof mechanisms for multi-hop networks, where the valuation of an agent only depends on its own type and whether it is selected or not. The VCG mechanism can still be applied to design a strategyproof mechanism, in which the payment for node v_{i_j} is

$$\mathcal{P}^{\text{VCG}}(v_{i_j}) = \mathscr{E}^T(\mathbf{P}_{G' \setminus v_{i_j}}^T(\mathbf{s}, \mathbf{d})) - \mathscr{E}^T(\mathbf{P}_{G'}^T(\mathbf{s}, \mathbf{d})) + \frac{\bar{\mathbf{c}}(v_{i_{j-1}}, v_{i_j})}{\prod_{t=j}^{h} \delta(v_{i_{t-1}}, v_{i_t})} \quad (6)$$

Recall that here $\bar{\mathbf{c}}(v_{i_{j-1}}, v_{i_j})$ is the marginal cost declared by node $v_{i_{j-1}}$ for forwarding a unit amount of data to node v_{i_j}. This could be uniform for all neighbors.

Algorithm 6 presents our routing scheme. In our method, the sender decides whether to perform routing using some criterion to ensure the strategyproofness. Clearly, the criterion should not be dependent on any information controlled by any node on the LCP. Recall that we assume that a node can only declare its marginal (node or link) cost for forwarding. We choose the a special bridge $\mathbf{B}^{\mathbf{s},\mathbf{d}}_{\min}$ for each relay node v_{i_j} to ensure that the cost of this bridge is at least the VCG payment to this node v_{i_j} under the transport-layer reliability model. We leave it as a future work to design a possibly tight criterion by using some special bridges.

Algorithm 6 Strategyproof Transport Layer Reliable Routing

1: **Routing subgame:** First the sender announces the willing payment $\nu(\mathbf{s},\mathbf{d})$ for sending a unit amount of data. The principal then collects data from all nodes in the network as previous algorithms. Let G' be the resulted graph and $\mathbf{P}^T_{G'}(\mathbf{s},\mathbf{d}) = v_{i_1}v_{i_2}\cdots v_{i_h}$ be the least cost path under the TCP model, where $\mathbf{s} = v_{i_1}$, $\mathbf{d} = v_{i_h}$. Let $\mathbf{P}^T_{G'\backslash u}(\mathbf{s},\mathbf{d})$ be the least cost path without using a node u. Path $\mathbf{P}^T_{G'}(\mathbf{s},\mathbf{d})$ is used for routing.

2: For each node $u \notin \mathbf{P}^T_{G'}(\mathbf{s},\mathbf{d})$, its payment $\mathcal{P}^{\mathrm{VCG}}(u) = 0$. For each internal node v_{i_j} $1 < j < h$ on this path, its payment $\mathcal{P}^{\mathrm{VCG}}(v_{i_j})$ is

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

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

$$(h-1)\|\mathbf{B}^{\mathbf{s},\mathbf{d}}_{\min}\| \le \nu(\mathbf{s},\mathbf{d}) \quad (7)$$

4: **Forwarding subgame:** Same as Algorithm 1.

The following theorem can be proved by directly using the VCG mechanism, thus, the proof is omitted here.

Theorem 14: Truth-telling is a *dominant strategy* and forwarding truthfully is a SPE when our algorithms for routing with transport layer reliability are used.

When both the cost and the reliability are private information, one may expect that Algorithm 6 is still a strategyproof routing scheme. Unfortunately, it is *not* true. The reason is now the valuation of a node over an outcome not only depends on its own service cost, but also depends on the service reliability of other agents. This violates the requirement of VCG mechanism. In this case, the least cost path output is monotonic: when an agent on the LCP reduces its cost (in either node or link cost model), or increases its link reliabilities, the agent is still on the LCP using new vector of cost and reliabilities. Then a mechanism can always be designed using LCP as an output [24], [38].

V. Selfish Principals

Unfortunately, when the sender has a finite valuation and could be selfish, the following result was proved in [39].

Lemma 15: [39] When the sender has a finite valuation and is selfish in declaring its valuation, *no* mechanism using the shortest path routing is strategyproof for all intermediate agents and the sender, while it is still budget-balanced.

Here a mechanism is budget-balanced if the money paid to all relay nodes is same as the money charged to the sender. It was proved in [39] that, in a strategyproof mechanism, the total charge from the sender could be as small as only $\frac{1}{n}$ of the total payment to relay agents when we require that the charge from the sender is *at most* the total payment to relay agents. Thus, there is a tradeoff of budget-balance and strategyproof when the sender and the relay nodes are all selfish. In the rest of the paper we assume that the sender and the principal are truthful. Thus, it is possible to design a budget-balanced and strategyproof mechanism.

In [12], Eidenbenz *et al.* proposed a novel scheme COMMIT that uses the cost of the *global replacement path* as the measurement whether a routing will take place. It has been pointed out in [12] that the protocol COMMIT is also not budget-balanced: they assume that the receiver will subsidize all the deficit by the sender. It is not difficult to show that the protocols presented here can also be extended to strategyproof protocols that deal with selfish principals when budget-imbalance is allowed. Let $\nu(\mathbf{s},\mathbf{d})$ be the maximum willing payment by the principals. Let $m = \nu'(\mathbf{s},\mathbf{d})$ be the declared willing payment by the source node, which could be different from $\nu(\mathbf{s},\mathbf{d})$. Let $P_{-\mathrm{LCP}}$ be the "least cost path" (depending on the routing requirement) connecting the source and destination nodes without using any nodes and links of the least cost path LCP. Path $P_{-\mathrm{LCP}}$ is also called *global replacement path* in [12]. Let $\mathscr{E}(P_{-\mathrm{LCP}})$ be the cost of the path $P_{-\mathrm{LCP}}$. We then present the following routing and charging protocol:

1) We perform the routing when $m \ge \mathscr{E}(P_{-\mathrm{LCP}})$ and the price paid by the sender is $\mathscr{E}(P_{-\mathrm{LCP}})$.
2) If the routing is performed, each node on the least cost path LCP will be paid based on protocols discussed in previous sections.

Lemma 16: This scheme is strategyproof for both sender and the relay nodes.

Proof: When the routing is performed, the profit of the principal is $\nu(\mathbf{s},\mathbf{d}) - \mathscr{E}(P_{-\mathrm{LCP}})$. Since $\mathscr{E}(P_{-\mathrm{LCP}})$ is independent of the actions by the principal, the principal cannot gain anything by reporting false valuation $m' \neq \nu(\mathbf{s},\mathbf{d})$. When $\nu(\mathbf{s},\mathbf{d}) < \mathscr{E}(P_{-\mathrm{LCP}})$, reporting a larger m will make the principal to potentially lose money.

Secondly, the value $\mathscr{E}(P_{-\mathrm{LCP}})$ is independent of the nodes on the shortest path, thus, the node on LCP cannot report a false value to improve its profit also. When $\mathscr{E}(P_{-\mathrm{LCP}}) > m$, to potentially make profit, a node on the shortest path must report some false cost such that the LCP changes and thus the path $P_{-\mathrm{LCP}}$ changes. To do so, it must report a higher cost since reporting a lower cost does not change LCP. This implies that the cost of the global replacement path $P_{-\mathrm{LCP}}$ will not be reduced at all. Thus, the condition $\mathscr{E}(P_{-\mathrm{LCP}}) > m$ still holds and thus the routing will not be performed.

When $\mathscr{E}(P_{-\mathrm{LCP}}) \le m$, the routing is performed. In this case, the payment to a relay node will be based on our protocols and we have already proved that they cannot report false value to improve its profit. Thus, in both cases, relay nodes cannot misreport to improve its profit. ■

Similarly, we can adapt our protocols (described in Algorithm 3 and Algorithm 6) for reliable link-layer routing and reliable transport layer routing to deal with potentially selfish senders as follows. The principal will let the sender to perform the routing when

1) $\sum_{j=2}^{h-1} \|\mathbf{B}^{\mathbf{s}}_{\min}(v_{i_j})\| \leq \nu(\mathbf{s}, \mathbf{d})$ and the sender will be charged $\sum_{j=2}^{h-1} \|\mathbf{B}^{\mathbf{s}}_{\min}(v_{i_j})$ if the link-layer reliability is required.
2) $(h-1)\|\mathbf{B}^{\mathbf{s},\mathbf{d}}_{\min}\| \leq \nu(\mathbf{s}, \mathbf{d})$ and the sender will be charged $(h-1)\|\mathbf{B}^{\mathbf{s},\mathbf{d}}_{\min}\|$ if transport layer reliability is required.

It is not difficult to prove that

Lemma 17: These modified schemes are strategyproof for both sender and the relay nodes.

However, these protocols are not budget-balanced. Recall that it has been proved in [39] that no strategyproof protocols exist that uses the least cost path for routing and is also budget-balanced.

VI. 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. Clearly, $\varrho^{\mathcal{A}}(\mathbf{P}) \geq 1$ for any payment scheme $\mathcal{A}$ developed in this paper, *e.g.*, Nash payment, VCG payment. In the worst case, the ratio $\varrho(\mathbf{P})$ could be as large as $O(n)$ when VCG payment is used for a network of n nodes [2]. In [2], the authors proposed the ratio by comparing the total payment $\mathcal{P}(\mathbf{P})$ with the cost of the new path obtained from the graph $G\backslash\mathbf{P}$, which represents the graph G without $\mathbf{P}$.

In our simulations, 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 the transmission range is r_n. Each pair of nodes within distance r_n will form a link with a probability p. In order to ensure that the network is bi-connected with high probability, it is proved in [4], [42] that the probability p and r_n should satisfy $n\pi p r_n^2 \geq \log n + c(n)$ for some $c(n) > 0$. We first set the probability $p = c \cdot \frac{\log n}{n}$, where c is a constant. We randomly pick $m = p \cdot n^2/2.0$ pairs of nodes within distance r_n and construct a link between them. To ensure that the graph is 2-connected, we check, for each node, whether its removal will disconnect the network; if it does, we add a new edge using nodes from two disconnected components. For all results reported here, the network is always 2-connected. We randomly assign the cost and the reliability to nodes and links. The cost is uniformly drawn from all integers in $[1, 40]$ and the reliability of a link is uniformly drawn from $0.01 \cdot k$ with integer k 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. The results reported here are the average of 50 independent runs, where we do routing for a fixed pair of nodes in every run.

A. Effect of Network Size

We first study the effect of the network size, for $n \in [70, 90]$. In this simulation, we fix the parameter $p = 0.1054337$, which results in 2-connected random networks, w.h.p., for results reported here. We measure the performances of our protocols based on the following metrics: the ratio of the Nash payment over the actual cost, the ratio of the VCG payment over the actual cost, and the ratio of the total bridge cost over the actual cost. Recall that in all our protocols, the bridge cost is *not* the cost paid by the sender. The total payment by the sender is still the total VCG payment. The bridge cost is only used by the sender to decide whether to perform the routing: routing is performed if the valuation ν is at least the total bridge cost.

We first study the cost variations when the network size changes under different reliability models (link layer reliability or transport layer reliability). We find that costs decrease when the network size increases. The following relations always hold: actual cost is less than the Nash payment, Nash payment is less than the VCG payment, and VCG payment is less than the bridge cost. Figure 2 and 3 plot the costs and 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 (which is only for link layer reliability). 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 total cost of all relaying agents, the VCG payment is about 1.5 to 2 times of the actual cost, while the bridge costs are about 2 to 2.5 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 2 to 2.5 times of the actual total 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 sender has a finite valuation: it will not be able to perform routing even its valuation is enough to cover the VCG payment (but smaller than the total bridge cost). In other words, Nash payment may be better off if performing routing is more important than inducing truthfulness from all agents.

B. Effect of Network Density

We then study other effects by fixing the network size ($n = 70$ 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 and 5 show different costs when the network density changes from the minimum p_0 needed for connectivity to $p_0 + 0.09$ with step size 0.01. 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. We also find that the ratio of the VCG payment over the actual cost and the ratio of the Nash payment over the actual cost do seem to decrease when the transport layer reliability is implemented and each node has uniform cost to all its outgoing links.

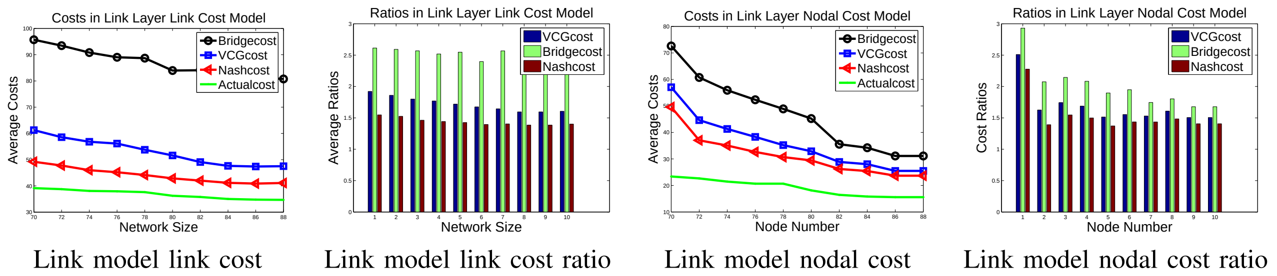


Fig. 2. Different costs and ratios when network size changes under link model.

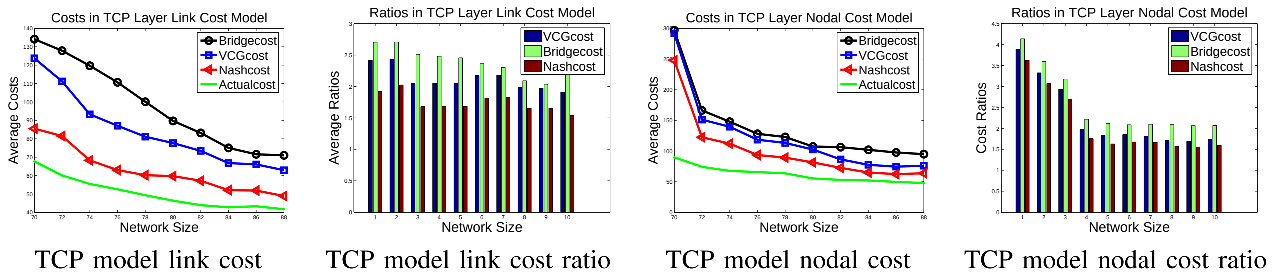


Fig. 3. Different costs and ratios when network size changes under TCP model.

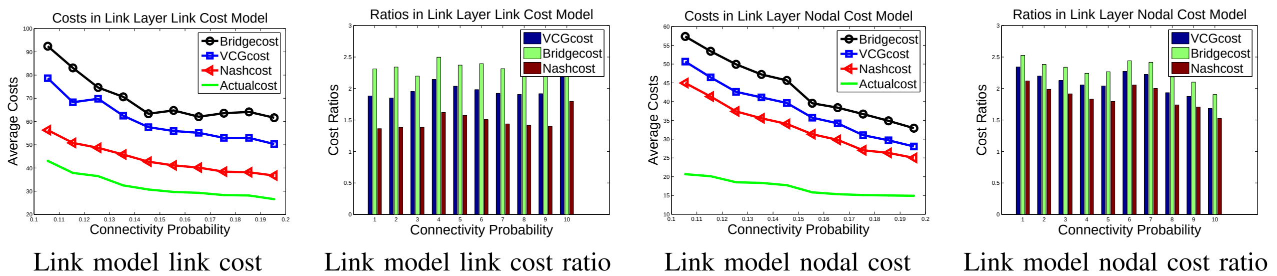


Fig. 4. Different costs and ratios when networking density changes under link model.

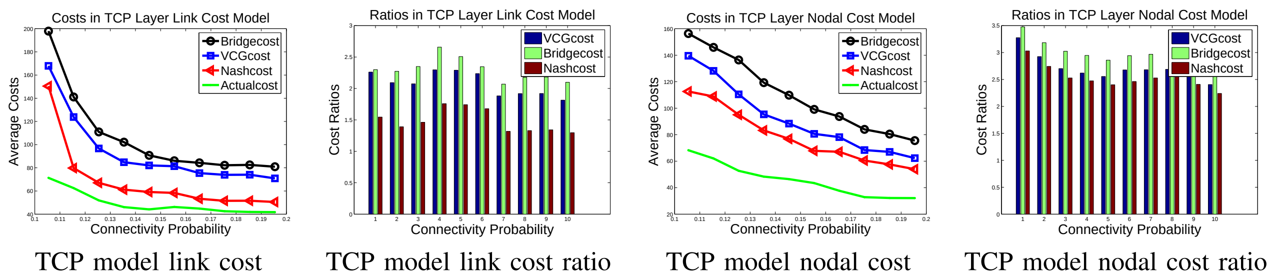


Fig. 5. Different costs and ratios when networking density changes under TCP model.

VII. Related Work

In any multi-hop routing scheme, cooperation by the intermediate nodes are essential for the successful delivery of traffic. In the past years, several new routing protocols [1], [13], [14], [28], [32], [40], [41] have been proposed to deal with possible selfishness of these intermediate nodes. Generally, there are two approaches to deal with the potential selfish nodes: (1) *mechanism design* or (2) *reputation*.

Routing has been an important part of the algorithmic mechanism-design from the very beginning when the costs of agents are *hidden information*. Nisan and Ronen [30] provided a polynomial-time strategyproof mechanism using VCG for optimal unicast route selection in a centralized computational model. Feigenbaum *et. al* [13] then addressed the truthful low cost routing in a different network model. Their mechanism again is the VCG mechanism. Optimal methods are presented in [18] to compute payments to all links and in [40] to compute the payments to all individual nodes. Anderegg and Eidenbenz [1] 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.* [41] 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.* [32] studied the selfish routing in Internet-like environments. In [5], [6], [19], a secure mechanism to stimulate nodes to cooperate is presented. Chen *et al.* in [8] provided iPass, an auction based scheme, to encourage forwarding behavior in mobile ad hoc networks. Combining with topology control, Eidenbenz *et al.* in [11], [12] provided a truthful protocol COMMIT based on VCG payment scheme. Zhong *et al.* [46] 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.

Achieving cooperation among selfish terminals in network was previously addressed by several authors using mainly reputation based scheme. In [43], Yu *et al.* provided a mechanism to bound the malicious attack. In [44], Yu *et al.* provided a set of mechanism, in which nodes kept a route traffic observer and built friendship with other nodes to monitor the route and detect cheating nodes. Jaramillo *et al.* in [20] presented DARWIN to quickly restore the cooperation after falsely detecting a node as selfish. In [35], [36], several methods are presented such that nodes' actions will form Nash Equilibrium and the energy efficiency is achieved at the equilibrium. In [28], 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.* [14], [15] studied the hidden action in multi-hop routing. They studied the mechanism design for UDP model (given the fixed routing path) with known uniform cost and reliability for agents, with known uniform reliability but private cost for agents. In [21]–[23], Kannan *et al.* developed a game-theoretic metric to measure the contribution of individual node and the qualitative performance of different routing mechanisms. In [27], Liu *et al.* presented a polynomial-time algorithm to find a Nash equilibrium path.

VIII. Conclusion

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 an agent can manipulate multiple parameters (*e.g.*, its costs and the reliability). The second challenge is to eliminate hidden action under hidden information. Notice that [14], [15] mainly deal with hidden action after the routing path is already selected.

Notice that in certain ad hoc routing protocols, routing paths may be used just once or a few times in a mobile environment. Such a high discount rate for future interactions enables a lazy node to drop traffic irrelevant to itself and allows malicious one to selectively forward traffic or propagate false information. This makes the detection of hidden action much difficult. A possible approach to overcome free-riding is to let members form small groups. New members must first transact with other members of the neighborhood to establish legitimacy. Once trust has been established inside the clustering of nodes, outside transactions can occur through established channels between groups. This could establish a credible observation threat to forestall hidden-action. The backbone based routing may thus become favorite since routing tables rarely change significantly and repeated interactions occur frequently. So any persistent failures will be quickly noted. We thus need to study the impact of possible hidden-actions on various backbone based routings, *e.g.*, [7], [26], [34].

References

[1] Anderegg, L., and Eidenbenz, S. Ad hoc-vcg: a truthful and cost-efficient routing protocol for mobile ad hoc networks with selfish agents. In *ACM MobiCom* (2003), pp. 245–259.

[2] Archer, A., and L. Tardos. Frugal path mechanisms. In *ACM SODA* (2002), pp. 991–998.

[3] Banerjee, S., and Misra, A. Minimum energy paths for reliable communication in multi-hop wireless networks. In *ACM MobiHoc* (2002), pp. 146–156.

[4] Bollobás, B. *Random Graphs*. Cambridge University Press, 2001.

[5] Buttyan, L., and Hubaux, J. Stimulating cooperation in self-organizing mobile ad hoc networks. *ACM/Kluwer Mobile Networks and Applications 5*, 8 (October 2003).

[6] Buttyan, L., and Hubaux, J. P. Enforcing service availability in mobile ad-hoc wans. In *ACM MobiHoc* (2000), pp. 87–96.

[7] Chen, B., Jamieson, K., Balakrishnan, H., and Morris, R. SPAN: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. In *Mobile Computing and Networking* (2001), pp. 85–96.

[8] Chen, K., and Nahrstedt, K. ipass: an incentive compatible auction scheme to enable packet forwarding service in manet. *IEEE ICDCS* (2004), 534–542.

[9] Clarke, E. H. Multipart pricing of public goods. *Public Choice* (1971), 17–33.

[10] Dong, Q., Banerjee, S., Adler, M., and Misra, A. Minimum energy reliable paths using unreliable wireless links. In *ACM MobiHoc* (2005).

[11] Eidenbenz, S., Resta, G., and Santi, P. Commit: a sender-centric truthful and energy-efficient routing protocol for ad hoc networks with selfish nodes. *IEEE IPDPS* (April 2005).

[12] Eidenbenz, S., Resta, G., and Santi, P. The commit protocol for truthful and cost-efficient routing in ad hoc networks with selfish nodes. *Mobile Computing, IEEE Transactions on 7*, 1 (2008), 19–33.

[13] Feigenbaum, J., Papadimitriou, C., Sami, R., and Shenker, S. A BGP-based mechanism for lowest-cost routing. In *ACM PODC* (2002), pp. 173–182.

[14] Feldman, M., and Chuang, J. Hidden-action in multi-hop routing. In *Second Workshop on the Economics of Peer-to-Peer Systems, in conjunction with ACM Sigcomm* (2004).

[15] Feldman, M., Chuang, J., Stoica, I., and Shenker, S. Hidden-action in multi-hop routing. In *ACM E-Commerce Conference* (2005).

[16] 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 (PWC)* (2003).

[17] Groves, T. Incentives in teams. *Econometrica* (1973), 617–631.

[18] Hershberger, J., and Suri, S. Vickrey pricing in network routing: Fast payment computation. In *IEEE FOCS* (2001), pp. 252–259.

[19] 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).

[20] Jaramillo, J. J., and Srikant, R. Darwin: distributed and adaptive reputation mechanism for wireless ad-hoc networks. In *ACM MobiCom* (2007), ACM, pp. 87–98.

[21] Kannan, R., and Iyengar, S. Game-theoretic models for reliable path-length and energy-constrained routing with data aggregation in wireless sensor networks. *Selected Areas in Communications, IEEE Journal on 22*, 6 (Aug. 2004), 1141–1150.

[22] Kannan, R., Sarangi, S., and Iyengar, S. S. Sensor-centric energy-constrained reliable query routing for wireless sensor networks. *J. Parallel Distrib. Comput. 64*, 7 (2004), 839–852.

[23] Kannan, R., Sarangi, S., Iyengar, S. S., and Ray, L. Sensor-centric quality of routing in sensor networks. In *IEEE INFOCOM* (2003).

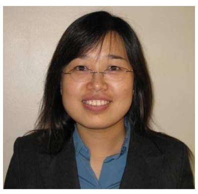
[24] Kao, Ming-Yang, Li, Xiang-Yang, and Wang, WeiZhao Towards Truthful Mechanisms for Binary Demand Games: A General Framework, In *ACM Conference on Electronic Commerce*, 2005.

[25] Li, X., Wu, Y., Xu, P., Chen, G., and Li, M. Hidden information and actions in multi-hop wireless ad hoc networks. In *ACM MobiHoc* (2008).

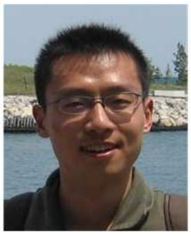
[26] Liang, B., and Haas, Z. J. Virtual backbone generation and maintenance in ad hoc network mobility management. In *IEEE INFOCOM* (2000), vol. 3, pp. 1293–1302.

[27] Liu, H., and Krishnamachari, B. A price-based reliable routing game in wireless networks. In *GameNets '06: Proceeding from the 2006 workshop on Game theory for communications and networks* (2006), p. 7.

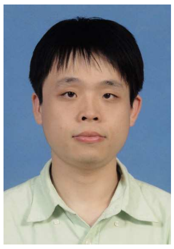
[28] Marti, S., Giuli, T. J., Lai, K., and Baker, M. Mitigating routing misbe-havior in mobile ad hoc networks. In *ACM MobiCom* (2000).
[29] N. Immorlica, D. Karger, E. N., and Sami, R. First-price path auctions. In *Proceedings of the 6th ACM Conference on Electronic Commerce* (2005).
[30] Nisan, N., and Ronen, A. Algorithmic mechanism design. In *ACM STOC99* (1999), pp. 129–140.
[31] Osborne, M. J., and Rubinstein, A. *A course in game theory*. The MIT Press, 2002.
[32] Qiu, L., Yang, Y. R., Zhang, Y., and Shenker, S. On selfish routing in internet-like environments. In *ACM SIGCOMM* (2003), pp. 151–162.
[33] 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).
[34] Sinha, P., Sivakumar, R., and Bharghavan, V. Cedar: Core extraction distributed ad hoc routing algorithm. *IEEE Journal on Selected Areas in Communications 17*, 8 (August 1999), 1454 –1465.
[35] Srinivasan, V., Nuggehalli, P., Chiasserini, C. F., and Rao, R. R. Energy efficiency of ad hoc wireless networks with selfish users. In *European Wireless Conference 2002 (EW2002)* (2002).
[36] Srinivasan, V., Nuggehalli, P., Chiasserini, C. F., and Rao, R. R. Cooperation in wireless ad hoc wireless networks. In *IEEE INFOCOM* (2003).
[37] Vickrey, W. Counterspeculation, auctions and competitive sealed tenders. *Journal of Finance* (1961), 8–37.
[38] Wang, W., Li, X.-Y., Wang Y., and Sun Z. Designing Multicast Protocols for Non-Cooperative Networks, *IEEE JSAC*, January, 2008.
[39] Wang, W., Eidenbez, S., Wang, Y. and Li, X.-Y. OURS- Optimal Unicast Routing Systems in Non-Cooperative Wireless Networks In *ACM MobiCom*, 2006.
[40] Wang, W., and Li, X.-Y. Low-cost routing in selfish and rational wireless ad hoc networks. *IEEE Transactions on Mobile Computing 5*, 5 (May 2006), 596–607.
[41] Wang, W., Li, X.-Y., and Wang, Y. Truthful mutlicast in selfish wireless networks. In *ACM MobiCom* (2004).
[42] Yi, C.-W. *Probabilistic Aspects of Wireless Ad Hoc Networks*. PhD thesis, Illinois Institute of Technology, 2005.
[43] Yu, W., and Liu, K. Attack-resistant cooperation stimulation in autonomous ad hoc networks. *Selected Areas in Communications, IEEE Journal on 23*, 12 (Dec. 2005), 2260–2271.
[44] Yu, W., Sun, Y., and Liu, K. Hadof: defense against routing disruptions in mobile ad hoc networks. *IEEE INFOCOM* (2005), 1252–1261 vol. 2.
[45] Zhong, S., Chen, J., and Yang, Y. Sprite: A simple, cheat-proof, credit-based system for mobile ad hoc networks. In *IEEE INFOCOM* (2003).
[46] 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).



Yanwei Wu is a Computer Science PhD student at Illinois Institute of Technology. She received B.Eng and M.E. from TianJin University, P.R.China, 1998 and 2003 respectively. Her research interest spans in algorithm design and analysis in wireless networks, including capacity/throughput optimization, energy efficiency, game theoretical study and security study. She also researched on agent based modeling as a Research Aide in Argonne National lab in 2007.



Shaojie Tang has been a PhD student of Computer Science Department at the Illinois Institute of Technology since 2006. He received BS degree in Radio Engineering from Southeast University, China, in 2006. His current research interests include algorithm design and analysis for wireless ad hoc network and online social network.



Ping Xu is a Computer Science PhD student at Illinois Institute of Technology. He received B.Eng and M.E. from ShangHai JiaoTong University, P.R.China, 2003 and 2006 respectively. His research interests span wireless networks, game theoretical study of networks, optimization in mesh network, and security in wireless network. He also worked on cognitive radio networks as a Summer Intern at Microsoft Research Asia in 2008.

Dr. Xiang-Yang Li has been an Associate Professor since 2006 and Assistant Professor of Computer Science at the Illinois Institute of Technology from 2000 to 2006. He was a visiting professor of Microsoft Research Asia from May 2007 to August 2008. He hold MS (2000) and PhD (2001) degree at Computer Science from University of Illinois at Urbana-Champaign. He received B.Eng. at Computer Science and Bachelor degree at Business Management from Tsinghua University, P.R. China in 1995. His research interests span wireless ad hoc and sensor networks, non-cooperative computing, computational geometry, and algorithms. He was a guest editor of special issues for ACM Mobile Networks and Applications, IEEE Journal on Selected Areas in Communications.