Efficient Strategyproof Multicast in Selfish Networks

Xiang-Yang Li⋆ Weizhao Wang⋆

Abstract— **In this paper, we propose several strategyproof mechanisms for multicast for both node weighted graphs and link weighted graphs. A node in node weighted graphs or a link in link weighted graphs is called an agent. For a multicast with a source node and a set of receiver nodes, we assume that they will pay each agent to carry the traffic from source to receivers. The utility of an agent is its payment received, minus its cost if it is in the multicast tree. We assume that the cost of each agent is private and each agent can manipulate its weight to maximize its utility. A payment scheme is strategyproof if every agent maximizes its utility when it reports its cost truthfully. In this paper, we proposed several strategyproof payment schemes based on various structures. We prove that each of our payment schemes is optimum for the corresponding structure used.**

I. Introduction

Consider a network $G = (V, E)$ consisting of a node set $V = \{v_1, v_2, \cdots, v_n\}$ and a set of links $E = \{e_1, e_2, \cdots, e_m\}$. For node weighted graph, we assume that each node v_i has a fixed cost c_i to relay a unit data for other nodes, which is only known to v_i. Then the network is represented by a node weighted graph $G = (V, E, c)$, where $c = (c_1, c_2, \cdots, c_n)$ is the cost vector of all nodes. For link weighted graph, we assume that each link e_i has a fixed cost c_i to carry a unit transit traffic, and the cost is only known to e_i. Then the network is represented by a link weighted graph $G = (V, E, c)$, where $c = (c_1, c_2, \cdots, c_m)$ is the cost vector of all links.

In this paper, we study the truthful mechanism design for multicast. We assume that there is a set of of receivers $Q = \{q_1, q_2, q_3, \cdots, q_k\} \subset V$, and $Q \backslash \{q_i\}$ could get data from q_i, for any $1 \leq i \leq k$. For simplicity of notations, we assume that $q_i = v_i$, for $1 \leq i \leq k$. Assume that each node is willing to pay other nodes or links to carry the transit traffic incurred by it. In this paper, all nodes (or links) are assumed to be *rational*, i.e., they respond to well-defined incentives and will deviate from the protocol only if it improves its gain. We want to design payment schemes for multicast such that every node (or link) has to report its true cost to maximize its profit. For node weighted graphs, we propose a payment scheme based on the spider structure whose total cost is within $2 \ln k$ times of the optimum. We prove that our payment scheme is truthful: satisfying both the incentive compatibility (IC) property and the individual rationality (IR) property.

Organization: The rest of the paper is organized as follows. In Section II, we review some definitions and priori arts on truthful mechanism design for multicast. In Section III, we present the first strategyproof mechanism for Steiner tree problem (or called multicast). The output of our mechanism (a tree) has cost within a constant factor of the optimum, and the payment is minimum among any truthful mechanism having this output.

II. Preliminaries and Priori Art

Preliminaries: A standard economic model for analyzing scenarios in which the agents act according to their own self-interest is as follows. There are n agents. Each agent i, for $i \in \{1, \cdots, n\}$, has some private information t^i, called its *type*. Then the set of n agents define a type vector $t = (t^1, t^2, \cdots, t^n)$, which is called the *profile*. There is an output specification that maps each type vector t to a set of allowed outputs. Agent i's preferences are given by a valuation function v^i that assigns a real number $v^i(t^i, o)$ to each possible output o. Here, we assume that the valuation of an agent does not depend on other agents' types. Everything in the scenario is public knowledge except the type t^i, which is a private information to agent i. Each agent i has a set of strategies A^i that the agent can choose from.

For each strategy vector $a = (a^1, \cdots, a^n)$, i.e., agent i plays strategy $a^i \in A^i$, the mechanism computes an *output* $o = o(a)$ and a *payment* vector $p = (p^1, \cdots, p^n)$, where $p^i = p^i(a)$. Here the payment p^i is the money given to each participating agent i if all agents playing under strategy vector a. Agent i's *utility* is $u^i = v^i(t^i, o) + p^i$. By assumption of rationality, agent i always tries to maximize its utility u^i. A mechanism is *strategyproof* (or called truthful) if the types are part of the strategy space A^i and each agent maximizes its utility by reporting its type t^i as input *regardless* of what other agents do. The *efficiency* of a mechanism is $\sum_i v^i(o, t^i)$.

Let a^{-i} denote the vector of strategies of all other agents except i, i.e., $a^{-i} = (a^1, a^2, \cdots, a^{i-1}, a^{i+1}, \cdots, a^n)$. Let $a|^i b = (a^1, a^2, \cdots, a^{i-1}, b, a^{i+1}, \cdots, a^n)$, i.e., each agent $j \neq i$ uses strategy a^j and the agent i uses strategy b. The following are some natural constraints which any truthful mechanism must satisfy:

1. **Incentive Compatibility (IC)**: Each agent i maximizes its utility if it reveals t^i, i.e., $v^i(t^i, o(a|^i t^i)) + p^i(a|^i t^i) \geq v^i(t^i, o(a|^i a^i)) + p^i(a|^i a^i)$.
2. **Individual Rationality (IR)**: Every agent i must have non-negative utility if it reveals t^i.
3. **Polynomial Time Computability (PC)**: All computation is done in polynomial time.

Arguably the most important positive result in mechanism design is what is usually called the generalized Vickrey-Clarke-Groves (VCG) mechanism by Vickrey [18], Clarke [4], and Groves [10]. A direct revelation mechanism $m = (o(t), p(t))$ belongs to the VCG family if (1) the output $o(t)$ computed based on the type vector t maximizes the objective function $g(o, t) = \sum_i v^i(t^i, o)$, and (2) the

⋆Dept. of Computer Science, Illinois Institute of Technology, Chicago, IL. Email: xli@cs.iit.edu,wangwei4@iit.edu.

payment to agent i is

$$p^i(t) = \sum_{j \neq i} v^j(t^j, o(t)) + h^i(t^{-i}).$$

Here $h^i()$ is an arbitrary function of t^{-i}. It is proved by Groves [10] that a VCG mechanism is truthful. Green and Laffont [8] proved that, under mild assumptions, VCG mechanisms are the only truthful implementations for problems with objective function $g(o,t) = \sum_i v^i(t^i, o)$.

Notice, for VCG mechanism, the output o is required to maximize the objective function $g(o,t)$. This makes the mechanism computationally intractable in many cases. Replacing the optimal output with non-optimal approximation usually leads to untruthful mechanisms. In their seminal paper on algorithmic mechanism design, Nisan and Ronen [15] added computational efficiency to the set of concerns that must be addressed in designing truthful mechanisms.

Priori Arts: Routing has been part of the algorithmic mechanism-design from the beginning. Nisan and Ronen [15] provided a polynomial-time strategyproof mechanism for unicast in a centralized computational model for link weighed graph where each link is an agent. Their mechanism is essentially a VCG scheme. Feigenbaum *et. al* [6] then addressed the truthful low cost routing for node weighted graph: each node k incurs a transit cost c_k for each transit packet. For any two nodes i and j of the network, $T_{i,j}$ is the intensity of the traffic (number of packets) originating from i and destined for j. Their scheme again is essentially the VCG mechanism. They gave a distributed method such that each node i can compute the payment to any relay node k. The engineering approaches presented in [2], [13], [3], [1] pay each relay node a *nuglet* and the source is charged of h nuglets if there are h relay nodes.

For multicast flow, Feigenbaum *et. al* [7] considered a set of users Q resided at the set of nodes V. Additionally, for each node $v_i \in V$, they assumed a *fixed* path from the source to it, determined by the multicast routing infrastructure. Then for every subset R of receivers, the delivery tree $T(R)$ is merely the union of the fixed paths from the source to the nodes containing receivers in R. They also assumed that each link e has a publicly known cost $c(e)$; each receiver q_i has a privately known valuation w_i for the reception of the data from the source. Based on reported w_i' from each q_i, the source node then decides a subset R of receives to get the data and a charge p^i to each receiver q_i, and pays the links on $T(R)$. The welfare u^i of a node q_i is $x_i w_i - p^i$, where x_i is indicator function of whether $q_i \in R$. The scheme is said to be *efficient* if it maximizes $\sum_{q_i \in Q} u^i$. The scheme is said to be *budget balanced* if the revenue raised from the receivers covers the cost of transmission links *exactly*. It is a classical result [9] that a strategyproof mechanism satisfying NPT (nonpositive transfer), VP (Voluntary Participation), CS (Consumer Sovereignty) cannot be both efficient and budget balanced. Feigenbaum *et. al* [7] studied how to share the link costs among receivers. Feigenbaum *et. al* [5] showed that there is no strategyproof multicast cost-sharing mechanism satisfying NPT, VP, and CS that is both approximately efficient and approximately budget-balanced. The benefit of fixing the multicast tree is relatively simple to implement and it avoids difficulty caused by the NP-hardness of finding optimal multicast tree.

Given a node weighted graph G and k terminals from G, the node weighted Steiner tree problem is to find a tree spanning all terminals with minimum cost. It is well-known [14] that the node weighted Steiner tree cannot be approximated less than $\ln k$, assuming that $NP \not\subseteq DTIME[n^{O(\log \log n)}]$. Several methods [14], [11] have been proposed to approximate the node weighted Steiner tree within $O(\ln k)$.

III. Strategyproof Multicast in Link Weighted Graph

A. Problem Statement

Consider any link weighted network $G = (V, E, c)$, where $E = \{e_1, e_2, \cdots, e_m\}$ are the set of links, and c_i is the weight of the link e_i. The multicast problem is to find a tree that spanning a given set of nodes $Q = \{q_1, q_2, \cdots, q_k\} \subset V$. For the simplicity of notations, we assume that $q_i = v_i$, for $1 \leq i \leq k$. We assume that the source node, say q_1, (and/or all receivers) will pay each relay node to carry the traffic from the source to receivers. Thus, each link is required to report its cost. The utility of a link is its payment received, minus its cost if it is selected in the multicast tree. We assume that the cost of each link is private and each link can manipulate its reported cost to maximize its utility. A payment scheme is strategyproof if every link maximizes its utility when it reports its cost truthfully. In this section, we propose several strategyproof payment schemes based on various structures. We prove that each of our payment schemes is optimum for the corresponding structure used.

Notice that a link may declare a cost other than its actual cost. Let $d = (d_1, d_2, \cdots, d_m)$ be the declared costs, where m is the number of links in G. Since, for strategyproof mechanisms, every link e_i maximizes its profit when $d_i = c_i$, we will simply use c_k to denote the reported cost of node v_k if it is clear from the context. If we change the cost of a link $e_i \in E$ to c_i', we denote the new graph as $G' = (V, E, c|^i c_i')$, or simply $c|^i c_i'$. If we remove one link e_i from G, we denote the resulting graph as $G \backslash e_i$, or simply $c|^i \infty$. Given a tree T spanning all receivers, the valuation of a link e_i is $-e_i$ is it is in T; otherwise its valuation is 0.

In order for every node $q_i \in Q$ to broadcast the message to the other receiving nodes in Q, we first should construct a multicast tree T spanning all nodes in Q. The summation of the cost of every link in a graph $H \subseteq G$ is called the weight of H, denoted as $\omega(H)$. Here, a leaf node in T must be a receiver and it does not incur any cost. It is well-known (see e.g., [17], [16]) that it is NP-hard to find the minimum cost multicast tree when given an arbitrary link weighted graph G. Takahashi and Matsuyama [17] first gave a polynomial time algorithm that approximates the minimum cost Steiner tree with approximation ration 2. Then a series of results have been developed to improve the approximation ratio. The current best result is due to

Robins and Zelikovsky [16], in which the authors presented a polynomial time method with approximation ratio $1 + \frac{\ln 3}{2}$.

B. VCG Strategyproof Mechanism

Obviously, we can design a strategyproof mechanism using VCG directly as follows: We first construct the minimum cost Steiner tree, denoted by $MCST(G)$, spanning all receivers Q; the payment to a node $v_i \in MCST(G)$ is

$$p^i = \omega(MCST(G \backslash v_i)) - \omega(MCST(G)) + c_i. \qquad (1)$$

Clearly, this scheme is truthful since $MCST(G)$ maximizes the valuation of all agents. It is well-known that VCG mechanism is the most *efficient*. However, on the other hand, since it is computational intractable to find the minimum cost tree $MCST(G)$ spanning all receivers for an arbitrary node weighted graph G, it will be computational expensive to implement VCG payment scheme on $MCST(G)$. In the following sections, we propose several strategyproof mechanisms that can be implemented in polynomial time. The payment schemes used in the following subsections are different from traditional VCG. This is in stark contrast with the almost universal use of VCG scheme for devising strategyproof mechanisms.

C. Strategyproof Mechanism Based on LCPS

C.1 Constructing LCPS

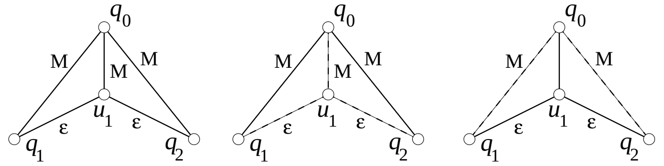
Assume that the source node is q_1. For each receiver q_i, we compute the least cost path, denoted by $\mathsf{P}(q_1, q_i, d)$, from the source to q_i under the reported cost profile d. The union of all least cost paths from the source to receivers is called *least cost path star*, denoted by LCPS. Clearly, we can construct LCPS in time $O(n \log n + m)$.

C.2 VCG Mechanism on LCPS Is Not Strategyproof

Intuitively, we would use the VCG payment scheme in conjunction with the LCPS tree structure as follows

$$p^i = \omega(LCPS(G \backslash v_i)) - \omega(LCPS(G)) + c_i.$$

We will show by example that the above payment scheme is not strategyproof. In other words, if we simply apply VCG scheme on LCPS, a link may have incentives to lie about its cost. Figure 1 illustrates such an example where link q_0u_1 can lie its cost to improve its utility. The payment to link q_0u_1 is 0 and its utility is also 0 if it reports its cost truthfully. The total payment to link q_0u_1 when q_0u_1 reported a cost $\underline{c_3} = M - 2\epsilon$ is $\omega(LCPS(c|^3\infty)) - \omega(LCPS(c|^3\underline{c_3})) + \underline{c_3} = 2M-(M-2\epsilon+2\epsilon)+M-2\epsilon = 2M-2\epsilon$ and the utility of link q_0u_1 becomes $u^3(c|^3\underline{c_3}) = 2M-2\epsilon-(M+\epsilon) = M-3\epsilon$, which is larger than $u^3(\underline{c}) = 0$, when $0 < \epsilon < M/3$.



(a) Graph G (b) LCPS (c) LCPS with lie

Fig. 1. Here the cost of links are $c(q_0q_1) = c(q_0q_2) = c(q_0u_1) = M$, and $c(q_1u_1) = c(q_2u_1) = \epsilon$.

C.3 Strategyproof Mechanism on LCPS

For each receiver q_j, we compute the least cost path from the source (say q_1) to q_i, and compute a payment $p_i^j(d)$ to every link e_j on the least cost path using the scheme for unicast: $p_i^j(c) = x_i^j d_j + \|\mathsf{P}(q_1, q_i, d|^j\infty)\| - \|\mathsf{P}(q_1, q_i, d)\|$. Here x_i^j denotes whether link e_j is used in least cost path from q_1 to q_i. The total payment to a link e_j is then

$$p^j(d) = \max_{q_i \in Q} p_i^j(d) \qquad (2)$$

Theorem 1: Payment based on LCPS is truthful and it is minimum among all truthful payments based on LCPS.
Proof. Clearly, when link e_j reports its cost truthfully, it has non-negative utility, i.e., the payment scheme satisfies the IR property. In addition, it also satisfies the IC property, since every link e_j cannot lie about its cost to increase its payment $p_i^j(c)$ based on receiver q_i. Thus, it cannot improve $\max_{q_i \in Q} p_i^j(c)$ by lying.

We then show that the above payment scheme pays the minimum among all strategyproof mechanism using LCPS as output. Assume that there is another payment scheme $\tilde{p}$ that pays less for a link e_j in a network G. Let $\delta = p^j - \tilde{p}^j$, then $\delta > 0$. Without loss of generality, assume that $p^j(c) = p_i^j(c)$. Thus, link e_j is on the least cost path $\mathsf{P}(q_1, q_i, c)$ and $\|\mathsf{P}_{-e_j}(q_1, q_i, c)\| - \mathsf{P}(q_1, q_i, c)\| = p^j - c_j = u^j(c)$. Then consider another graph G' with cost profile $c' = c|^j(c_j + u^j(c) - \frac{\delta}{2})$. Obviously, link e_j is still on the least cost path to q_i in G' since $\|\mathsf{P}_{-e_j}(q_1, q_i, c')\| = \|\mathsf{P}_{-e_j}(q_1, q_i, c)\|$ and $\|\mathsf{P}_{e_j}(q_1, q_i, c')\| = \|\mathsf{P}_{e_j}(q_1, q_i, c)\| + u^j(c) - \frac{\delta}{2}$. Notice that $\|\mathsf{P}_{e_j}(q_1, q_i, c)\| = \|\mathsf{P}(q_1, q_i, c)\|$. Thus, $e_j \in LCPS(G')$. From the IC property, we know that the payment to link e_j in graph G must be the same as in graph G (the payment to a node v_j is independent of its cost as long as the valuation of e_j does not change). Notice that $u^j(c) = p^j - c_j = \delta + \tilde{p}^j - c_j$. The utility of link e_j under payment scheme $\tilde{p}$ becomes $\tilde{p}^j - (c_j + u^j(c) - \frac{\delta}{2}) = -\frac{\delta}{2} < 0$. In other words, in graph G', even link e_j reports its true cost, link e_j gets negative utility under payment scheme $\tilde{p}$. This finishes our proof. □

Notice that the payment based on $p^k(c) = \min_{q_i \in Q} p_i^k(c)$ is not truthful since a link may lie its cost upward so it can discard some low payment from some receiver. In addition, the payment $p^k(c) = \sum_{q_i \in Q} p_i^k(c)$ is *not* truthful either.

D. Strategyproof Mechanism Based on Approximation of Minimum Cost Steiner Tree

D.1 Approximation Minimum Cost Steiner Tree

In the following we review the method by Takahashi and Matsuyama [17] that approximates the minimum cost Steiner tree with approximation ration 2. Remember that a series of results have been developed to improve the approximation ratio. The current best approximation ratio

$1+\frac{\ln 3}{2}$ is due to Robins and Zelikovsky [16]. The method by Takahashi and Matsuyama [17] works as follows. First, we find the receiver, say q_i, that is closest to the source, i.e., the least cost path $\mathsf{P}(q_i, q_1, c)$ has the least cost among all receivers. We then connect this receiver to the source using the least cost path between them. We contract this selected path to one virtual vertex, which will be used as virtual source node for next round, and remove some edges if necessary. If there is any other receivers remaining, we repeat the above step. We call such operation one *round*: finding the receiver that is closest to the virtual source node, finding the shortest path connecting this receiver and the virtual source, contracting this path to a new virtual source node. Let P_r be the path found in round r, and t_r be the receivers it connects with the virtual source node. Given k receivers, the method will terminate in k rounds. Hereafter, let $ST(G)$ be the final tree constructed using the method by Takahashi and Matsuyama. They [17] proved that $\omega(ST(G)) \leq 2\omega(MCST(G))$.

D.2 VCG Mechanism on Approximating Minimum Cost Steiner Tree Is Not Strategyproof

Given a tree $ST(G)$ approximating the minimum cost Steiner tree, a natural payment would be to pay each node based on VCG scheme, i.e., the payment to a node $v_i \in ST(G)$ is

$$p^i = \omega(ST(G \backslash v_i)) - \omega(ST(G)) + c_i.$$

We will show by example that this payment scheme does *not* satisfy IR property, i.e., it is possible that some node has negative utility. Figure 2 illustrates such an example

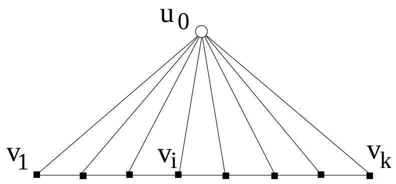


Fig. 2. Here nodes v_i, $1 \leq i \leq k$ are receivers; the cost of each link u_0v_i is $1+\epsilon$, where ϵ is a sufficiently small positive real number. The cost of each link v_iv_{i+1} is 2.

with node v_1 being the source node. It is not difficulty to show that, in the first round, link v_1v_2 is selected to connect terminals v_1 and v_2 with cost 2; in round r, we will select link v_rv_{r+1} to connect to v_r with cost 2. Thus, the tree $ST(G)$ will be just the path $v_1v_2\cdots v_k$, whose cost is $\sum_{i=1}^{k-1} c(v_iv_{i+1}) = 2(k-1)$.

When link $e_1 = v_1v_2$ is not used, it is easy to see that the final tree $ST(G \backslash e_1)$ will only use node u_0 to connect all receivers with total cost $k(1+\epsilon)$. Thus, the utility of link $e_1 = v_1v_2$ is $\omega(ST(G \backslash e_1)) - \omega(ST(G)) = k(1+\epsilon) - 2(k-1) = k\epsilon - k + 2$, which is negative when $\epsilon < \frac{k-2}{k}$. Thus, the payment to link v_1v_2 does not satisfy the incentive rationality property.

D.3 Strategyproof Mechanism on Approximation of Minimum Cost Steiner Tree

We now describe in this subsection a strategyproof mechanism (without using VCG) based on the tree $ST(G)$. Instead of paying the node based on the final structure, we will pay the node based on each round. Let $w_r(d)$ be the cost of the path P_r selected in the rth round if the cost profile is d.

Before we present out payment scheme, we first study the conditions that a link e_i will be selected in some round. Assume that the cost of all links (except link e_i) are fixed. First, if the link e_i is selected in round 1 when it reports a cost d_i, then clearly it will be selected in round 1 when it reports a cost $d_i' < d_i$. Thus, there is a cut off value, denoted by c_i^1, such that when link e_i reports a cost $d_i' < c_i^1$, e_i is always selected in round 1; when link e_i reports a cost $d_i' > c_i^1$, e_i is always *not* selected in round 1; when link e_i reports a cost $d_i' = c_i^1$, it is unknown whether e_i will be selected in round 1 (this depends on the way how ties are broken among different paths). Here c_i^1 could be 0 or ∞. This further implies that if link e_1 is selected in round 2, it must report a cost $d_i' \geq c_i^1$. Similarly, we know that there is also a cut off value $c_i^2 \geq c_i^1$ such that when link e_i reports a cost $c_i^1 < d_i' < c_i^2$, e_i is always selected in round 2. We thus can conclude that there are a sequence of real non-negative numbers

$$0 = c_i^0 \leq c_i^1 \leq c_i^2 \leq \cdots \leq c_i^r \leq \cdots \leq c_i^{k-1} \leq c_i^k = \infty$$

such that when link e_i reports a cost $c_i^{r-1} < d_i' < c_i^r$, e_i is guaranteed to be selected in round $1 \leq r \leq k$. Here a link selected in round k means that this link is not in $ST(G)$.

We then study whether the existence of a link e_i contributes to the shortest path found in round r. When $c_i^{r-1} = c_i^r$, it means that link e_i is not guaranteed to be selected in round r (the only case it being selected is e_i reports cost c_i^r and the shortest path using e_i wins through tie-breaking). When $c_i^{r-1} < c_i^r$, link e_i can always let itself be selected in round r by declaring a cost d_i with $c_i^{r-1} < d_i < c_i^r$. In other words, the existence of link e_i contributes to the found shortest path P_r in round r. Then we pay link e_i for this contribution in round r as

$$p_r^i = c_i^{r-1} + w_r(c|^i\infty) - w_r(c|^i c_i^{r-1}).$$

If link e_i is selected in the structure $ST(G)$, the payment to the link e_i is

$$p^i = \max_r p_r^i = \max_r \left(c_i^{r-1} + w_r(c|^i\infty) - w_r(c|^i c_i^{r-1})\right). \quad (3)$$

If link e_i is not selected in the structure $ST(G)$, the payment to e_i is then 0.

Theorem 2: The payment function based on Equation (3) satisfies the IR and IC properties.

Proof. We first show that the payment scheme satisfies the individual rationality property. If link e_i is selected in round r when it reports its true cost c_i. We then know that $c_i^{r-1} \leq c_i \leq c_i^r$. Note that $p_r^i = c_i^{r-1} + w_r(c|^i\infty) - w_r(c|^i c_i^{r-1})$. It is easy to see that the shortest path $\mathsf{P}_r(c)$

selected in round r under cost profile c is the same as the shortest path $\mathsf{P}_r(c|^i c_i^{r-1})$ selected in round r under cost profile $c|^i c_i^{r-1}$. Thus, we have $w_r(c|^i c_i^{r-1}) = w_r(c) - c_i + c_i^{r-1}$. It implies that the utility of link e_i is $p_r^i - c_i = w_r(c|^i\infty) - w_r(c)$, which is obviously non-negative. When link e_i is not selected in any round, then its payment is 0 and its utility is also 0.

We then show that the payment scheme satisfies the incentive compatibility property. We first consider the case that link e_i is selected in structure $ST(G)$ if it reports its cost truthfully. Clearly, when it lies down its cost, it will still be selected, thus, its valuation does not change. Clearly, the payment p^i to link e_i does not change also. Then its utility remains the same. When link e_i lies up its cost, if link e_i is not selected anymore, link e_i will have 0 utility instead of non-negative utility if it reported truthfully. If link e_i is still selected, the utility remains the same.

We then consider the case that link e_i is not selected when it reports its true cost. Clearly, link e_i can only improve its utility by lying down its cost, say d_i, so it is selected in some round. Notice that in any round r, if the link e_i report a cost larger than c_i^r, link e_i is not selected. It means that path $\mathsf{P}_r(c|^i\infty)$ has the less cost then any path containing link e_i from some receiver to the virtual source node. Thus, $w_r(c|^i\infty) \le w_r(c|^i c_i^r) = w_r(c|^i c_i^{r-1}) + c_i^r - c_i^{r-1}$. Then the payment p^i to a link e_i in $ST(G)$ is at most $\max_{1\le r\le k-1}\left(c_i^r - c_i^{r-1}\right)$, which is no more than c_i^{k-1}. Since link e_i is not selected in $ST(G)$ when its cost is c_i, it means $c_i \ge c_i^{k-1}$. Then, lying down its cost to $d_i \le c_i^{k-1}$, link e_i has utility $p^i(c|_i^d) - c_i \le c_i^{k-1} - c_i \le 0$. Thus, link e_i cannot improve its utility by lying down its cost. □

Theorem 3: The payment function based on Equation 3 is optimum among all truthful payments based on structure $ST(G)$.

Proof. We prove this by contradiction. Assume that there is a truthful payment scheme $\tilde{p}$ that pays less to link e_i for a cost profile c when the communication graph is G.

Obviously e_i is selected in $ST(G)$ since our scheme pays 0 to links not selected, and we cannot pay negative to these links from the individual rationality property.. Assume that $\tilde{p}^i = p^i - \delta$ for some $\delta > 0$. Let r be the round such that $p_r^i = p^i$ in our payment scheme. In other words, we have $p^i = c_i^{r-1} + w_r(c|^i\infty) - w_r(c|^i c_i^{r-1})$. We then consider a graph G' in which the actual cost of link e_i is $d_i = p^i - \frac{\delta}{2}$. Obviously, $d_i = \tilde{p}^i + \delta > 0$, and

$$d_i = c_i^{r-1} + w_r(c|^i\infty) - w_r(c|^i c_i^{r-1}) - \frac{\delta}{2}. \qquad (4)$$

We first show that link e_i is also selected in the tree $ST(G')$. Assume that link e_i is not selected before we start round r. Then, in round r, among all paths using e_i connecting some receivers to the virtual source, path $\mathsf{P}_r(c|^i c_i^{r-1})$ has the least cost. Notice that the definition of the sequence c_i^r, $1 \le r \le k$, ensures that e_i does appear at round r when its cost is c_i^{r-1}) (if it did not appear in previous round). Thus, the cost of path $\Pi = \mathsf{P}_r(c|^i c_i^{r-1})$ (which containing e_i) is no more than the cost of path $\mathsf{P}_r(c|^i\infty)$. Let's consider the path $\Pi = \mathsf{P}_r(c|^i c_i^{r-1})$. If replacing the cost of link e_i with d_i, its total cost becomes $w_r(c|^i c_i^{r-1}) - c_i^{r-1} + d_i$, which is equal to $w_r(c|^i\infty) - \frac{\delta}{2}$ from Equation (4). In addition, the total cost of any path containing e_i will have the same amount of increment of cost, i.e., $d_i - c_i^{r-1}$. Thus, the path Π is still the least cost path to connect some receivers to the virtual source node. Consequently, link e_i will be selected in round r when the cost profile is $c|^i d_i$.

Since the valuation of link e_i is the same when it has cost c_i or cost d_i, the payment $\tilde{p}^i$ to link e_i should be the same (otherwise, link e_i could lie its cost depending which case pays higher). Thus, the payment $\tilde{p}^i(c|^i d_i) = \tilde{p}^i(c) = p^i(c) - \delta$. Then the utility of link e_i, when the actual cost profile is $c|^i d_i$, is $\tilde{p}^i(c|^i d_i) - d_i = p^i(c) - \delta - (p^i - \frac{\delta}{2}) = -\frac{\delta}{2}$, which is negative. This clearly violates the individual rationality property. □

IV. Strategyproof Multicast in Node Weighted Graph

A. Problem Statement

Consider any node weighted network $G = (V, E, c)$, where c_i is the weight of node v_i. Usually we need to communicate among a group of nodes $Q = \{q_1, q_2, \cdots, q_k\} \subset V$ instead of a pair of nodes, which is known as multicast problem. For the simplicity of notations, we assume that $q_i = v_i$, for $1 \le i \le k$. We assume that the source node, say q_1, (and/or all receivers) will pay each relay node to carry the traffic from the source to receivers. Thus each node is required to report its cost. The utility of a node is its payment received, minus its cost if it is in the multicast tree. We assume that the cost of each node is private and each node can manipulate its reported cost to maximize its utility. A payment scheme is strategyproof if every node maximizes its utility when it reports its cost truthfully. In this section, we propose several strategyproof payment schemes based on various structures. We prove that each of our payment schemes is optimum for the corresponding structure used.

Notice that the nodes may declare cost other than c. Let $d = (d_1, d_2, \cdots, d_n)$ be the declared costs. Since, for strategyproof mechanisms, every node v_i maximizes its profit when $d_i = c_i$, we will simply use c_k to denote the reported cost of node v_k if it is clear from the context. If we change the cost of a node $v_i \in V$ to c_i', we denote the new graph as $G' = (V, E, c|^i c_i')$, or simply $c|^i c_i'$. If we remove one vertex v_i and all its incident links from G, we denote the resulting graph as $G\backslash v_i$, or simply $c|^i\infty$.

In order for every node $q_i \in Q$ to broadcast the message to the other receiving nodes in Q, we first should construct a multicast tree T spanning all nodes in Q. The summation of cost of every node in T is called the weight of the tree T, denoted as $\omega(T)$. Here, a leaf node in T must be a receiver and it does not incur any cost. It is well-known (see e.g., [12]) that it is NP-hard to find the minimum cost multicast tree when given an arbitrary node weighted graph G, and it is at least as hard to approximate as the set cover problem.

Guha and Khuller [12] showed that it can be approximated within $O(\ln k)$, where k is the number of receivers.

B. VCG Strategyproof Mechanism

Obviously, we can design a strategyproof mechanism using VCG directly as follows: We first construct the minimum cost Steiner tree, denoted by $MCST(G)$, spanning all receivers Q; the payment to a node $v_i \in MCST(G)$ is

$$p^i = \omega(MCST(G \backslash v_i)) - \omega(MCST(G)) + c_i. \qquad (5)$$

Clearly, this scheme is truthful since $MCST(G)$ maximizes the valuation of all agents. It is well-known that VCG mechanism is the most *efficient*. However, on the other hand, since it is computational intractable to find the minimum cost tree $MCST(G)$ for an arbitrary node weighted graph G, it will be computational expensive to implement VCG payment scheme on $MCST(G)$. In the following sections, we propose several strategyproof mechanisms that can be implemented in polynomial time. The payment schemes used in the following subsections are different from traditional VCG. This is in stark contrast with the almost universal use of VCG scheme for devising strategyproof mechanisms.

C. Strategyproof Mechanism Based on LCPS

C.1 Constructing LCPS

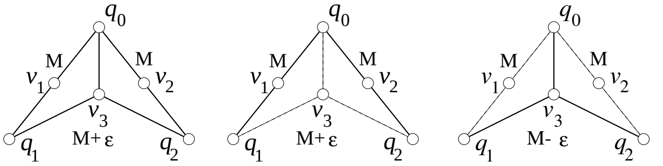
Assume that the source node is q_1. For each receiver q_i, we compute the least cost path, denoted by $\mathsf{P}(q_1, q_i, d)$, from the source to q_i under the reported cost d. The union of all least cost paths from source to receivers is called *least cost path star*, denoted by LCPS. Clearly, we can construct LCPS in time $O(n \log n + m)$.

C.2 VCG Mechanism on LCPS Is Not Strategyproof

Intuitively, we would use the VCG payment scheme in conjunction with the LCPS tree structure as follows

$$p^i = \omega(LCPS(G \backslash v_i)) - \omega(LCPS(G)) + c_i.$$

We will show by example that the above payment scheme is not strategyproof. In other words, if we simply apply VCG scheme on LCPS, a node may have incentives to lie about its cost. Figure 3 illustrates such an example where node v_3 can lie its cost to improve its utility. The payment to node



(a) Graph G (b) LCPS (c) LCPS with lie

Fig. 3. Here the cost of nodes are $c_1 = c_2 = M$ and $c_3 = M + \epsilon$.

v_3 is 0 and its utility is also 0 if it reports its cost truthfully. The total payment to node v_3 when v_3 reported a cost $\underline{c_3} = M - \epsilon$ is $\omega(LCPS(c|^3\infty)) - \omega(LCPS(c|^3\underline{c_3})) + \underline{c_3} = 2M - (M - \epsilon) + M - \epsilon = 2M$ and the utility of node v_3 becomes $u^3(c|^3\underline{c_3}) = 2M - (M + \epsilon) = M - \epsilon$, which is larger than $u^3(c) = 0$.

C.3 Strategyproof Mechanism on LCPS

For each receiver q_j, we compute the least cost path from the source (say q_1) to q_i, and compute a payment $p_i^j(d)$ to every node v_k on the least cost path using the scheme for unicast: $p_i^j(c) = x_i^j d_j + \|\mathsf{P}(q_1, q_i, d|^j\infty)\| - \|\mathsf{P}(q_1, q_i, d)\|$. Here x_i^j denotes whether node v_j is used in least cost path from q_1 to q_i. The total payment to a node v_j is then

$$p^j(d) = \max_{q_i \in Q} p_i^j(d) \qquad (6)$$

Theorem 4: Payment based on LCPS is truthful and it is minimum among all truthful payments based on LCPS.
Proof. Clearly, when node v_j reports its cost truthfully, it has non-negative utility, i.e., the payment scheme satisfies the IR property. In addition, it also satisfies the IC property, since every node v_j cannot lie about its cost to increase its payment $p_i^j(c)$ based on receiver q_i. Thus, it cannot improve $\max_{q_i \in Q} p_i^j(c)$ by lying.

We then show that the above payment scheme pays the minimum among all strategyproof mechanism using LCPS as output. Assume that there is another payment scheme $\tilde{p}$ that pays less for a node v_j in a network G. Let $\delta = p^j - \tilde{p}^j$, then $\delta > 0$. Without loss of generality, assume that $p^j(c) = p_i^j(c)$. Thus, v_j is on the least cost path $\mathsf{P}(q_1, q_i, c)$ and $\|\mathsf{P}_{-v_j}(q_1, q_i, c)\| - \mathsf{P}(q_1, q_i, c)\| = p^j - c_j = u^j(c)$. Then consider another graph G' with cost profile $c' = c|^j(c_j + u^j(c) - \frac{\delta}{2})$. Obviously, v_j is still on the least cost path to q_i in G' since $\|\mathsf{P}_{-v_j}(q_1, q_i, c')\| = \|\mathsf{P}_{-v_j}(q_1, q_i, c)\|$ and $\|\mathsf{P}_{v_j}(q_1, q_i, c')\| = \|\mathsf{P}(q_1, q_i, c)\| + u^j(c) - \frac{\delta}{2}$. Thus, $v_j \in LCPS(G')$. From the IC property, we know that the payment to node v_j in graph G must be the same as in graph G (the payment to a node v_j is independent of its cost as long as the valuation of v_j does not change). Notice that $u^j(c) = p^j - c_j = \delta + \tilde{p}^j - c_j$. The utility of node v_j under payment scheme $\tilde{p}$ becomes $\tilde{p}^j - (c_j + u^j(c) - \frac{\delta}{2}) = -\frac{\delta}{2} < 0$. In other words, in graph G', even node v_j reports its true cost, node v_j gets negative utility under payment scheme $\tilde{p}$. This finishes our proof. □

Notice that the payment based on $p^k(c) = \min_{q_i \in Q} p_i^k(c)$ is not truthful since a node may lie its cost upward so it can discard some low payment from some receiver. In addition, the payment $p^k(c) = \sum_{q_i \in Q} p_i^k(c)$ is *not* truthful either.

D. Strategyproof Mechanism Based on RMST

D.1 Constructing RMST

Our method constructing a cost efficient spanning tree works as follows. First, we calculate the pairwise shortest path $\mathsf{P}(q_i, q_j, G)$ between any two receiver nodes $q_i, q_j \in Q$. Then construct a complete edge weighted graph $K(G, Q, w)$ using Q as its vertices, where edge $q_i q_j$ corresponds to $\mathsf{P}(q_i, q_j, G)$, and its weight $w(q_i q_j)$ is the cost of $\mathsf{P}(q_i, q_j, G)$, i.e., $w(q_i q_j) = \|\mathsf{P}(q_i, q_j, G)\|$. Remember that the total weight of a least cost path $\mathsf{P}(q_i, q_j, G)$ does not include the cost of two end-points q_i and q_j. We then construct the minimum spanning tree (MST) on $K(G, Q, w)$ and denote it as $RMST(G)$. The $RMST(G)$

is calculated based on the reported cost profile. If we construct RMST on a graph G removed of node v_k, we denote the resulting MST as $RMST(G\backslash v_k)$. Obviously, $RMST(G\backslash v_k) = RMST(G|^k\infty)$.

D.2 VCG Mechanism on RMST Is Not Strategyproof

Figure 3 also illustrates such an example where node v_3 can lie its cost to improve its utility when output is RMST structure. The payment to node v_3 is 0 and its utility is also 0 if it reports its cost truthfully. The total payment to node v_3 when v_3 reported a cost $\underline{c_3} = M-\epsilon$ is $\omega(RMST(c|^3\infty)) - \omega(RMST(c|^3\underline{c_3})) + \underline{c_3} = 2M - (M-\epsilon) + M - \epsilon = 2M$ and the utility of node v_3 becomes $u^3(c|^3\underline{c_3}) = 2M - (M+\epsilon) = M - \epsilon$, which is larger than $u^3(c) = 0$.

D.3 Strategyproof Mechanism on RMST

Given a spanning tree T, and a pair of nodes p and q on T, clearly there is a unique path connecting them on T. We denote such path as $\Pi_T(p,q)$, and the edge with the maximum length on this path as $LE(p,q,T)$. For simplicity, we use $LE(p,q,c)$ (or $LE(p,q,G)$) to denote $LE(p,q,RMST(G))$ and use $LE(p,q,c|^k d_k)$ to denote $LE(p,q,RMST(G|^k d_k))$.

Based on the structure $RMST(G)$, we then design a truthful mechanism for calculating the payment to relay nodes on $RMST(G)$ as follows. For every node $v_k \in V \setminus Q$ in G, first calculate $RMST(G)$ and $RMST(G|^k\infty)$ according to the nodes' declared costs vector c. For any edge $e = q_iq_j \in RMST(G)$ and any node $v_k \in \mathsf{P}(q_i,q_j,G)$, we define the payment to node v_k based on the virtual link q_iq_j as

$$p_{ij}^k(c) = \|LE(q_i,q_j,c|^k\infty)\| - \|\mathsf{P}(q_i,q_j,c)\| + c_k. \qquad (7)$$

Here $\|\Pi\|$ denotes the total cost of a path Π. If a node v_k is not on $\mathsf{P}(q_i,q_j,G)$, then the payment $p_{ij}^k(c)$ to node v_k based on the virtual link q_iq_j is 0. If the path $\mathsf{P}(q_i,q_j,G)$ is not used in $RMST(G)$, then the payment to any node on path $\mathsf{P}(q_i,q_j,G)$ based on edge q_iq_j is also 0. The final payment to node v_k based on $RMST(G)$ is

$$p^k(c) = \max_{q_iq_j \in RMST(G)} p_{ij}^k(c).$$

In citeWL03-STOC, we present an efficient method to compute the payment in time $\min\{O(k^2 n\log n + k^2 m), O(n^2\log n + mn)\}$. The following result was also proved in [19].

Theorem 5: The payment scheme is truthful and this payment scheme pays each node minimum among all truthful mechanisms using RMST.

E. Strategyproof Mechanism Based on Spider

E.1 Constructing Spider

However, the structures LCPs and RMST could have cost $\Theta(k)$ times the optimum cost, i.e., it is not efficient. In the following we review the method by Klein and Ravi [14] (improved by Guha and Khuller [11] later) that constructs a structure that can approximate the optimum cost within $2\ln k$. Remember that the best known lower bound on the approximation ratio is $\ln k$, assuming that $NP \not\subseteq DTIME[n^{O(\log\log n)}]$. Their method is based on a structure called *spider*. A spider is defined as a tree having at most one node of degree more than two. Such a node (if one exists) is called the *center* of the spider. The cost of a spider is the sum of the weights of the nodes in the spider. The ratio of a spider is the ratio of its cost to the number of its terminals (or receivers in this paper). Contracting a spider is the operation of contracting all the nodes of the spider to form one virtual vertex. If we contract a spider S in a graph G, making the contracted vertex into a terminal, and the cost of the new virtual terminal is 0. The method by Klein and Ravi [14] repeatedly contracts the spider with minimum ratio. At the first round, all receivers are terminals with cost 0. Their method finds the spider that connects some terminals with the minimum ratio and then contracts it. We call such operation one *round*: finding the minimum ratio spider and then contracting it. Let S_r be the spider found in round r, and t_r be the number of terminals it connects. Here each virtual terminal is counted as 1 although it often contains many receivers. Then $\omega_r = \omega(S_r)/t_r$ is the ratio of that spider. Given k receivers, the method will terminate in less than k rounds. Hereafter, let $ST(G)$ be the final tree constructed using the spiders.

E.2 VCG Mechanism on Spider Is Not Strategyproof

Again, we may want to pay nodes based on VCG scheme, i.e., the payment to a node $v_i \in ST(G)$ is

$$p^i = \omega(ST(G\backslash v_i)) - \omega(ST(G)) + c_i.$$

We show by example that the payment scheme does *not* satisfy IR property: it is possible that some node has negative utility. Figure 4 illustrates such an example. It is not

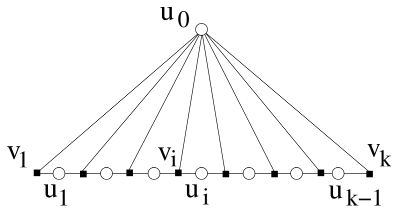


Fig. 4. Here nodes v_i, $1 \le i \le k$ are receivers; the cost of node u_0 is 1. the cost of each node u_i is $\frac{2}{k+1-i} - \epsilon$, where ϵ is a sufficiently small positive number.

difficulty to show that, in the first round, node u_1 is selected to connect terminals v_1 and v_2 with cost ratio $\frac{1}{k} - \frac{\epsilon}{2}$ (while all other spiders have cost ratio at least $\frac{1}{k}$). Then nodes u_1, v_1 and v_2 form a virtual terminal. At the beginning of round r, we have a virtual terminal, denoted by V_r formed by nodes u_i, $1 \le i \le r-1$, and receivers v_i, $1 \le i \le r$; all other receivers v_i, $r < i \le k$ are the remaining terminals. It is easy to show that we will select node u_r at round r to connect V_r and v_{r+1} with cost ratio $\frac{1}{k+1-r} - \frac{\epsilon}{2}$. Thus, the total cost of the tree $ST(G)$ is $\sum_{i=1}^{k-1}(\frac{2}{k+1-i} - \epsilon) = 2H(k) - 2 - (k-1)\epsilon$.

When node u_1 is not used, it is easy to see that the final tree $ST(G\backslash u_1)$ will only use node u_0 to connect all receivers with cost ratio $\frac{1}{k}$ when $\frac{1}{k-1} - \frac{\epsilon}{2} > \frac{1}{k}$. Notice that this condition can be trivially satisfied by letting $\epsilon = \frac{1}{k^2}$. Thus, the utility of node u_1 is $p^1 - c(u_1) = \omega(ST(G\backslash u_1)) - \omega(ST(G)) = -2H(k)+3+(k-1)\epsilon$, which is negative when $k \geq 8$, and $\epsilon = 1/k^2$.

E.3 Strategyproof Mechanism on Spiders

We now describe in this subsection a strategyproof mechanism (without using VCG) based on the tree constructed using spiders. Instead of paying the node based on the final structure, we will pay the node based on each round. Let $\omega_r(d)$ be the cost of the spider S_r selected in the rth round if the cost profile is d. Let $t_r(d)$ be the number of the terminals connected by the spider S_r if the cost profile is d. Then $\frac{\omega_r(d)}{t_r(d)}$ is the ratio of the spider S_r. Let $S_r^i(d)$ be the spider in round r containing node v_i and has the minimum ratio among all spiders containing node v_i. Let $\omega_r^i(d)$ be the cost of the spider $S_r^i(d)$. Let $t_r^i(d)$ be the number of the terminals connected by the spider $S_r^i(d)$. Let $S_r^{-i}(d)$ be the spider in round r without node v_i and has the minimum ratio among all spiders without node v_i. Let $\omega_r^{-i}(d)$ be the cost of the spider $S_r^{-i}(d)$. Let $t_r^{-i}(d)$ be the number of the terminals connected by the spider $S_r^{-i}(d)$.

We first show that, in a round r, if node v_i is selected to the spider structure under cost profile c, then node v_i is still selected in the spider structure under cost profile $c|^i d_i$. Here the spider under cost profile $c|^i d_i$ could be different from the spider under cost profile c in round r. Obviously, we have $\frac{\omega_r^i(c)}{t_r^i(c)} < \frac{\omega_r^{-i}(c)}{t_r^{-i}(c)}$. Then, for spider $S_r^i(c)$, if we replace the cost c_i of node v_i by $d_i < c_i$, the cost of $S_r^i(c)$ becomes $\omega_r^i(c) - c_i + d_i < \omega_r^i(c)$. Thus, the spider $S_r^i(c)$ still has less ratio than the spider $S_r^{-i}(c)$, which is the same as $S_r^{-i}(c|^i d_i)$. Our claim then holds since $S_r^i(c|^i d_i)$ has ratio at most the ratio of $S_r^i(c)$. It implies that we have a sequence of real numbers

$$0 = B_i^0 \leq B_i^1 \leq \cdots \leq B_i^{r-1} \leq B_i^r \leq \cdots \leq B_i^{k-1} \leq \infty.$$

such that if the cost c_i of node v_i satisfies $B_i^{r-1} < c_i < B_i r$, then node v_i is guaranteed to be selected in round r.

Assume that, when node v_i has cost $c_i \in (B_i^{r-1}, B_i^r)$, the spider $S_r^i(c)$ selected has t terminals. When the cost of v_i is $d_i < c_i$, it is easy to show that the spider $S_r^i(c|^i d_i)$ could not have more terminals than $S_r^i(c)$, i.e., $t_r^i(c|^i d_i) \leq t_r^i(c)$ when $d_i < c_i$.

Note that, in round r, the number of total terminals is at most $k-r+1$. We now further refine the range $[B_i^{r-1}, B_i^r]$ using numbers

$$B_i^{r-1} = B_i^{r,1} \leq B_i^{r,2} \leq \cdots \leq B_i^{r,k+1-r} = B_i^r$$

such that, when the cost of node v_i is in range $(B_i^{r,t-1}, B_i^{r,t})$, the spider $S_r(c)$ has exactly t terminals, i.e., $t_r^i(c) = t$. Notice that spider $S_r(c)$ always contains node v_i in this case, i.e., $S_r^i(c)$ is same as $S_r(c)$.

We then define the payment to node v_i as follows. If node v_i is selected in some round, when its cost is c_i, then its payment is

$$p^i = \max_{1 \leq r \leq k-1,\ 2 \leq t \leq k-1-r} \max p_{r,t}^i, \tag{8}$$

where

$$p_{r,t}^i = \frac{t}{t_r^{-i}(c)} \omega_r^{-i}(c) - \omega_r^i(c|^i B_i^{r,t-1}) + B_i^{r,t-1}.$$

When node v_i is not selected in any round, then the payment to node v_i is 0.

Theorem 6: The payment function based on Equation (8) satisfies the IR and IC properties.

Proof. First of all, we show that the utility of each node v_i is non-negative. When node v_i is not selected, its utility is clearly 0. When node v_i is selected in round r, and assume that the selected spider has t terminals, i.e., $c_i \in [B_i^{r,t-1}, B_i^{r,t}]$. Note that $p_{r,t}^i = \frac{t}{t_r^{-i}(c)} \omega_r^{-i}(c) - \omega_r^i(c|^i B_i^{r,t-1}) + B_i^{r,t-1}$. Thus,

$$\begin{aligned} & p_{r,t}^i - c_i \\ = \ & \frac{t}{t_r^{-i}(c)} \omega_r^{-i}(c) - \omega_r^i(c|^i B_i^{r,t-1}) + B_i^{r,t-1} - c_i \\ = \ & t \cdot \left(\frac{\omega_r^{-i}(c)}{t_r^{-i}(c)} - \frac{\omega_r^i(c|^i B_i^{r,t-1}) - B_i^{r,t-1} + c_i}{t} \right) \\ \geq \ & 0 \end{aligned}$$

We then prove that no node can increase its utility by lying its cost. It is easy to show that, when node v_i is selected originally with cost c_i, node v_i cannot improve its utility by lying its cost. We only have to show that, when node v_i is not selected originally, node v_i cannot lie down its cost so it is selected to improve its utility. Since node v_i is not selected with cost c_i, we have, for any r,

$$\frac{\omega_r^{-i}(c)}{t_r^{-i}(c)} < \frac{\omega_r^i(c)}{t_r^i(c)}.$$

We now show that

$$p_{r,t}^i \leq c_i.$$

It is equivalent to prove that, for any feasible t and any feasible round r,

$$\frac{t}{t_r^{-i}(c)} \omega_r^{-i}(c) - \omega_r^i(c|^i B_i^{r,t-1}) + B_i^{r,t-1} - c_i \leq 0.$$

Assume that this is not true, i.e.,

$$\frac{\omega_r^{-i}(c)}{t_r^{-i}(c)} > \frac{\omega_r^i(c|^i B_i^{r,t-1}) - B_i^{r,t-1} + c_i}{t}.$$

This means that the spider $S_r^i(c|^i B_i^{r,t-1})$ with t terminals when node v_i has cost c_i has smaller ratio than spider $S_r^{-i}(c)$. Consequently, node v_i is guaranteed to be selected in round r if it is not selected before round r when its cost is c_i (the spider could be $S_r^i(c|^i B_i^{r,t-1})$ or some other spider containing node v_i. This is a contradiction to the fact the v_i is not selected in any round when its cost is c_i. This finishes the proof. □

Theorem 7: The payment function based on Equation 8 is optimum among all payments based on Spider structure.

Proof is omitted here due to space limit.

References

[1] Blazevic, L., Buttyan, L., Capkun, S., Giordano, S., Hubaux, J. P., and Boudec, J. Y. L. Self-organization in mobile ad-hoc networks: the approach of terminodes. *IEEE Communications Magazine 39*, 6 (June 2001).

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

[3] Buttyan, L., and Hubaux, J. P. Enforcing service availability in mobile ad-hoc wans. In *Proceedings of the 1st ACM international symposium on Mobile ad hoc networking & computing* (2000), pp. 87–96.

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

[5] Feigenbaum, J., Krishnamurthy, A., Sami, R., and Shenker, S. Hardness results for multicast cost sharing (extended abstract).

[6] Feigenbaum, J., Papadimitriou, C., Sami, R., and Shenker, S. A BGP-based mechanism for lowest-cost routing. In *Proceedings of the 2002 ACM Symposium on Principles of Distributed Computing.* (2002), pp. 173–182.

[7] Feigenbaum, J., Papadimitriou, C. H., and Shenker, S. Sharing the cost of multicast transmissions. *Journal of Computer and System Sciences 63*, 1 (2001), 21–41.

[8] Green, J., and Laffont, J. J. Characterization of satisfactory mechanisms for the revelation of preferences for public goods. *Econometrica* (1977), 427–438.

[9] Green, J., and Laffont, J.-J. North Holland, Amsterdam, 1979.

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

[11] Guha, S., and Khuller, S. Improved methods for approximating node weighted steiner trees and connected dominating sets. In *European Symposium on Algorithms* (1996), pp. 179–193.

[12] Guha, S., and Khuller, S. Improved methods for approximating node weighted steiner trees and connected dominating sets. In *Foundations of Software Technology and Theoretical Computer Science* (1998), pp. 54–65.

[13] 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).

[14] Klein, R. N., and Ravi, R. A nearly best-possible approximation algorithm for node weighted steiner trees. *Journal of Algorithms 19*, 1 (1995), 104–114.

[15] Nisan, N., and Ronen, A. Algorithmic mechanism design. In *Proc. 31st Annual Symposium on Theory of Computing (STOC99)* (1999), pp. 129–140.

[16] Robins, G., and Zelikovsky, A. Improved steiner tree approximation in graphs. In *Symposium on Discrete Algorithms* (2000), pp. 770–779.

[17] Takahashi, H., and Matsuyama, A. An approximate solution for the steiner problem in graphs. *Math. Japonica 24*, 6 (1980), 573–577.

[18] Vickrey, W. Counterspeculation, auctions and competitive sealed tenders. *Journal of Finance* (1961), 8–37.

[19] Wang, W., and Li, X.-Y. Truthful mutlicast in selfish ad hoc networks, 2003. Submitted for publication.