Low-Cost Truthful Multicast in Selfish and Rational Wireless Ad Hoc Networks

Weizhao Wang⋆ Xiang-Yang Li⋆

Abstract— **It is conventionally assumed that all wireless devices will follow some prescribed routing protocols without any deviation. However, the scarce resources in wireless devices raise a concern about this assumption. Most often, the owners of wireless devices will try to manipulate the protocols for its own benefit, instead of faithfully following the protocols. Therefore, some new protocols intended for selfish and rational wireless devices need to be designed.**

In this paper, we specifically study the multicast in selfish and rational wireless ad hoc networks. By assuming that each wireless node has a private cost of forwarding data for other nodes, we give an efficient method to construct a multicast tree, namely VMST, whose cost is 5-approximation of the optimum multicast tree cost for homogeneous wireless networks modelled by unit disk graph. Based on VMST, we design a truthful payment scheme that pays minimum for any relay node among all truthful payment schemes based on VMST. We also conduct extensive experiments to study the practical performances of proposed protocol.

I. Introduction

Wireless networks have received significant attentions over past few years due to its potential applications in various situations such as battlefield, emergency relief and environmental monitoring, etc. Unlike wired networks and cellular networks, which have fixed infrastructures, wireless *ad hoc* networks enjoy a more flexible composition. A wireless ad hoc network is a collection of radio devices (transceivers) located in a geographic region. Each node is equipped with an omni-directional antenna and has limited transmission power. One of the distinctive features of wireless networks is that the signal sent out by each wireless device can be received by all nodes within its transmission range, i.e., it can use a broadcasting-like manner to distribute the message to all neighboring nodes. In this paper, we consider a wireless ad hoc network $G = (V, E)$ consisting of a set V of n nodes distributed in a two-dimensional plane, and an edge $uv \in E$ if u and v can receive signal from each other directly. When all nodes have the same maximum transmission range, by a proper scaling, the wireless networks are modelled by unit disk graphs in the literature: an edge uv exits iff $\|uv\| \leq 1$.

In multi-hop wireless networks, a communication session is established either through a single-hop radio transmission if the communication parties are close enough (within the transmission range of each other), or through relaying by intermediate devices . Many existing routing protocols for wireless ad hoc networks assume that each individual wireless node (possibly owned by individual selfish users) will follow prescribed routing protocols without deviation – except, perhaps, for the faulty or malicious ones However, some users may deviate from this, or even modify the behavior of routing protocols for self-interested reasons: a user may refuse to relay the messages for other nodes since it consumes its scarce energy and memory resources. Thus, a stimulation mechanism is required to encourage users to provide service to other nodes and follow the designed routing protocols.

Following the common assumption in the literature, we assume that each wireless device is *rational*: it will deviate from a protocol only if it improves its gain. We study how to design a multicast routing protocol such that every rational selfish wireless node will follow the protocol without any deviation. In our model, we assume that each wireless node v_k has a private cost c_k of forwarding data for any other node. Our protocol first requires every node declaring a minimum monetary value d_k it will charge for relaying a unit data. The protocol then finds a structure for multicast based on the report costs of all nodes and computes a payment p_k to compensate the cost for each node v_k. The profit of node v_k is then $p_k - c_k$ if it relays. Notice node v_k's declared cost d_k may be different from its actual cost c_k.

As to our knowledge, this is the *first* paper to study how to design a multicast protocol that is *truthful* in wireless ad hoc settings. Here truthful means that every node will get maximal non-negative utility when it declares its true cost. In addition, the cost of multicast tree (called VMST) used in our protocol is at most 5 times of the optimum when the original communication graph is modelled by unit disk graph. We further prove that our payment scheme based on VMST is the minimum for any relay node among all truthful payment schemes based on VMST. We also conduct extensive experiments to study the practical performances of our protocol compared to the most often used multicast routing protocol.

The rest of the paper is organized as follows. In Section II, we what is a truthful mechanism design. In Section III, we first present a method to construct a spanning tree whose total cost is within a constant factor of the optimum, and then present a truthful mechanism based on this multicast tree .Priori arts are reviewed in In Section II. We conclude our paper in Section VI by pointing out some possible future works.

II. Preliminaries

A. Network Model

In a wireless ad hoc Network, if a node sends a packet, then it will consume some energy and usually it is assumed that it wouldn't cost the receiving node any energy to receive the message. We consider a wireless ad hoc network consisting of a node set $V = \{v_1, v_2, \cdots, v_n\}$ distributed in a two dimensional plane. Each node v_i has a cost c_i to relay a unit data for other nodes. Here the unit data could be one packet, or the data sent in one communication session. In this paper, we assume that the cost c_i is a fixed constant known only to node v_i. When the node's cost are dynamic, we can show that our routing pro-

⋆Dept. of Computer Science, Illinois Institute of Technology, Chicago, IL. Email: wangwei4@iit.edu, xli@cs.iit.edu. The work of the second author is partially supported by NSF CCR-0311174.

tocols still work as long as it is static for the current communication session. We use v_1 to represent the access point (AP) of the wireless network to the wired network if it presents. A wireless network is then represented by a node weighted graph $G = (V, E, c)$, where E is the set of links uv such that u and v can receive the signal from each other, and $c = (c_1, c_2, \cdots, c_n)$ is the cost vector of all nodes. We assume that the graph G is node bi-connected.

In this paper, we devise a truthful routing protocol for multicast in a selfish and rational wireless network. In a multicast, we assume that there is a set of nodes $Q = \{q_1, q_2, q_3, \cdots, q_r\} \subset V$ forming a group. We assume that at some moment, a node $q_i \in Q$ wants to send data to all nodes in group Q, and all nodes in $Q \in q_i$ want the data. For simplicity, we assume that a set of of receivers $\{q_2, q_3, \cdots, q_r\}$ will get data from source node q_1. Since all nodes are in a same group, we assume that any receiver node q_j will relay the data for any other receiver nodes for free if it is chosen as relay node, i.e., $c_k = 0$ for $1 \leq k \leq r$. In this paper, all wireless nodes 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.

B. Truthful Mechanism Design

Traditionally, the following model is used to analyze scenarios in which the agents act according to their own self-interests. There are n agents, i.e., there are n selfish wireless devices. Each agent i, for $i \in \{1, \cdots, n\}$, has some private information t^i, called its *type*. In this paper, the type t^i is its cost to forward a unit data packet in a network environment. 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 $\mathcal{O}$ 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. In this paper, we only consider direct revelation mechanisms, i.e., the strategy of an agent is to report its type.

For each strategy vector $a = (a^1, \cdots, a^n)$, i.e., agent i plays strategy $a^i \in A^i$, the mechanism $= (\mathcal{O}, p)$ defines an *output* $\mathcal{O}(a)$ and a *payment* vector $p = (p^1, \cdots, p^n)$, where $p^i = p^i(a)$ is the money given to each participating agent i. The mechanism designer defines an output method $\mathcal{O}$ such that an objective function $g(o(a), t)$ is maximized under output method $\mathcal{O}$. For example, the objective function $g(o(a), t) = \sum_{i=1}^{n} v^i(t^i, o(a))$, maximizes the total valuation of all agents.

Agent i's *utility* (or called *profit* by some researchers) is $u^i = v^i(t^i, o) + p^i$. We assume that each agent is *rational*, i.e., agent i always tries to maximize its utility u^i. A mechanism satisfies the *individual compatibility* (IC), if each agent maximizes its utility by reporting its type t^i truthfully *regardless* of what other agents do. Clearly, an agent will not participate in a routing if its profit is negative. A mechanism satisfies the *individual rationality* (IR), (or called voluntary participation) if each agent gets non-negative profit by reporting its type t^i truthfully *regardless* of what other agents do. A mechanism is *truthful* (or called *strategyproof*) if it satisfies both IR and IC properties.

Arguably the most important positive result in mechanism design is what is usually called the family of generalized Vickrey-Clarke-Groves (VCG) mechanisms by Vickrey [1], Clarke [2], and Groves [3]. A VCG mechanism applies to mechanism design maximization problems where the objective function is utilitarian and the set of possible outputs is assumed to be finite. A maximization mechanism design problem is called *utilitarian* if its objective function is $g(o, t) = \sum_i v^i(t^i, o)$. Thus, a mechanism $M = (\mathcal{O}(t), p(t))$ belongs to the VCG family if (1) the output method $\mathcal{O}(t)$ maximizes the objective function $g(o, t) = \sum_i v^i(t^i, o)$, and (2) the payment to agent i is

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

where $h^i()$ is an arbitrary function of t^{-i} and different agent could have different function $h^i()$ as long as it is defined on t^{-i}. It is proved by Groves [3] that a VCG mechanism is truthful. Green and Laffont [4] proved that, under mild assumptions, VCG mechanisms are the only truthful implementations for utilitarian problems.

An important observation here is that the output function of a VCG mechanism is required to maximize the objective function. This makes the mechanism computationally intractable in many cases, such as the multicast problem studied in this paper. Notice that replacing the optimal algorithm with non-optimal approximation usually leads to untruthful mechanisms. To made the mechanism polynomial time computable, we have to add computational efficiency to the set of concerns that must be addressed [5].

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.

III. MULTICAST

In this section, we propose a truthful multicast routing protocol for wireless ad hoc networks such that each selfish and rational node will follow the protocol out of its own self-interest.

A. Problem Statement

We consider a wireless ad hoc network consisting of a node set $V = \{v_1, v_2, \cdots, v_n\}$. Every node v_i has a fixed transmission range and thus transmit a unit data needs a fixed cost c_i. Usually we need to communicate among a group of nodes $Q = \{q_1, q_2, \cdots, q_r\} \subset V$ 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 $1 \leq i \leq r$. 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 broadcasting tree T spanning all nodes in Q such that, whenever an internal node $v \in T$ receiving a new message, it relays the message to all its neighbors (except the neighboring node from which the message came). Remember that such relaying can be done by a single message in wireless networks. The summation of cost of every node in T is called the weight of the tree T, denoted as $\omega(T)$. Remember that for wireless broadcasting, a leave node in T does not incur any cost here and it must be a receiver (otherwise, we can shrink the tree by removing the non-receiver leaf node). To save the energy consumption, we want to find a tree T_{opt} whose weight is the minimum among all trees used for multicast. It is well-known (see e.g., [8]) that it is NP-hard to find the optimal solution when given an arbitrary wireless ad hoc network modelled by a node weighted graph G. Finding the minimum cost multicast tree is at least as hard as to approximate the set cover problem. Guha and Khuller [8] showed that it can be approximated within $O(\ln k)$, where k is the number of receivers. Thus, we have to rely on some heuristics to approximate the optimum multicast tree T_{opt}. In this paper, given an arbitrary wireless network modelled by a node weighted unit disk graph G, we present an efficient method to construct a spanning tree T whose total cost $\omega(T)$ is at most 5 times the cost of the minimum $\omega(T_{opt})$.

B. Multicast Tree Construction

Our method constructing a cost efficient spanning tree for multicast routing works as follows. First, we calculate the pairwise shortest path $\mathsf{LCP}(q_i, q_j, G)$ between any two nodes $q_i, q_j \in Q$ for a network modelled by a node weighted graph $G = (V, E, c)$, where the node cost vector is c. We then construct a complete edge weighted graph $K(G, Q, w)$ using Q as its vertices, where edge q_iq_j corresponds to $\mathsf{LCP}(q_i, q_j, G)$, and its weight $w(q_iq_j)$ is the cost of $\mathsf{LCP}(q_i, q_j, G)$, i.e., $w(q_iq_j) = \|\mathsf{LCP}(q_i, q_j, G)\|$. For our later convenience, here the total weight of the least cost path $\mathsf{LCP}(q_i, q_j, G)$ does not include the cost of two end-points q_i and q_j. For convenience of our analysis, we also assume that no two edges in $G = (V, E, c)$ have the same length, and there are no two paths in $G = (V, E, c)$ have the same length. Dropping this assumption doesn't change the result of our analysis.

Algorithm 1: **Virtual MST Algorithm**

1. First, we construct the virtual weighted complete graph $K(G, Q)$ on the original network $G = (V, E, c)$.
2. Construct the minimum spanning tree (MST) on $K(G, Q)$. The resulting MST is denoted as $VMST(G)$.
3. For each edge q_iq_j selected in $VMST(G)$, we find the corresponding least cost path $\mathsf{LCP}(q_i, q_j, G)$ in G. We mark every internal node v_k on the path $\mathsf{LCP}(q_i, q_j, G)$ as *relay node*.
4. In graph G, build a spanning tree using all nodes marked with *relay node* and all receiver nodes Q, and denote the final spanning tree on G as $SVMST(G)$.

Notice a node is in tree $SVMST(G)$ if and only if it is on some virtual edges in $VMST(G)$, thus we consider the structure $VMST(G)$ instead of $SVMST(G)$.

Theorem 1: $VMST(G)$ is a 5-approximation of the optimal solution in terms total cost if the wireless network is modelled by unit disk graph.

PROOF. Assume that the optimal solution is a tree called T_{opt}. Let $V(T_{opt})$ be the set of nodes used in the tree T_{opt}. Clearly, $\omega(T_{opt}) = \sum_{v_i \in V(T_{opt})} c_i$. Similarly, for any spanning tree T of $K(G, Q)$, we define $\omega(T) = \sum_{e \in T} w(e)$. In order to prove the theorem, we prove a stronger result: $5 \cdot \omega(T_{opt}) \geq \omega(VMST(G))$.

First, for all nodes in T_{opt}, when disregarding the node weight, there is a spanning tree T'_{opt} on $V(T_{opt})$ with node degree at most 5 since the wireless network is modelled by a unit disk graph. This is due to a well-known fact that there is an Euclidean minimum spanning tree with the maximum node degree at most 5 for any set of two-dimensional points. Note here we do not need construct such spanning tree with maximum degree at most 5 explicitly. Obviously, $\omega(T_{opt}) = \omega(T'_{opt})$. Thus, tree T'_{opt} is also an optimal solution. with maximal node degree degree at most 5.

For spanning tree T'_{opt}, we root it at an arbitrary node and duplicate every link in T'_{opt} (the resulting structure is called DT'_{opt}. Clearly, every node in DT'_{opt} has even degree now. Thus, we can find an Euler circuit, denoted by $EC(DT'_{opt})$, that visits every vertex of DT'_{opt} and uses every edge of DT'_{opt} exactly once, which is equivalent to say that every edge in $T'_{opt}(G)$ is used exactly twice. Consequently, we know that every node v_k in $V(T_{opt})$ is used exactly $deg_{T'_{opt}}(v_k)$ times. Here $deg_G(v)$ denotes the degree of a node v in a graph G. Thus, the total weight of the Euler circuit is at most 5 times of the weight $\omega(T'_{opt})$, i.e.,

$$\omega(EC(DT'_{opt})) \leq 5 \cdot \omega(T'_{opt}).$$

Notice that here if a node v_k appears multiple times in $EC(DT'_{opt})$, its weight is also counted multiple times in $\omega(EC(DT'_{opt}))$.

If we walk along $EC(DT'_{opt})$, we visit all receivers, and length of any subpath between receivers q_i and q_j is no smaller than $|\mathsf{LCP}(q_i, q_j, G)|$. Thus, the cost of $EC(DT'_{opt})$ is at least $\omega(VMST(G))$ since $VMST(G)$ is the minimum spanning tree spanning all receivers and the cost of the edge q_iq_j in $VMST(G)$ corresponds the path with the least cost $\|\mathsf{LCP}(q_i, q_j, G)\|$. In other words,

$$\omega(EC(DT'_{opt})) \geq \omega(VMST(G)).$$

Consequently, we have

$$\omega(VMST(G)) \leq \omega(EC(DT'_{opt})) \leq 5 \cdot \omega(T'_{opt}).$$

This finishes the proof. □

Notice that the assumption that the receiver nodes will relay the transit traffic for other receiver nodes for free is crucial in the above proof. If this is not the case, then Theorem 1 does not hold anymore. Let us assume that the receiver node does charge for relay. Remember that we will not count the cost of all leaf nodes (which must be receivers) when we count the cost of a multicast tree T_{opt}. First, we cannot guarantee any relation between the cost of T_{opt} and the tree T'_{opt} with bounded degree 5. Secondly, when we transform a tree T'_{opt} to an Euler circuit, we cannot say that the weight of an virtual edge q_iq_j in $EC(DT'_{opt})$ is larger than its weight in $VMST(G)$ anymore. It is because we only count the cost of internal node of T'_{opt} when compute the cost of q_iq_j in $EC(DT'_{opt})$, but on the other hand, we have to count the cost of two end nodes q_i and q_j when compute the cost of q_iq_j in $K(Q, G)$. Figure III-B illustrates an example that $VMST(G)$ does not give constant approximation when receiver nodes charge for relaying transit traffic. In the example, node v_n has cost $c_n = M + \epsilon$. There are r receivers q_1, q_2,

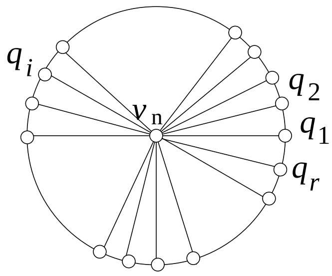


Fig. 1. VMST does not approximate minimum cost multicast tree when receiver nodes charge for relay.

$\cdots$, q_r on the unit circle centered at node v_n, each with cost M. Clearly, the VMST tree will be formed by links $q_i q_{i+1}$ for $1 \le i \le r-1$. Thus, the relay nodes chosen by VMST will be q_i, $1 \le i \le r-1$, and the total cost of all chosen relay nodes is $(r-1) \cdot M$. On the other hand, the minimum cost multicast tree will be formed by links $v_n q_i$, for $1 \le i \le r$. The total cost of the optimum tree is $M + \epsilon$.

C. Payment Scheme

In the previous subsection, we show that the virtual minimum spanning tree is a 5 approximation for the minimum cost multicast tree for homogeneous wireless networks modelled by unit disk graphs. Remember a truthful multicast routing protocol is composed of two parts: a spanning tree used for multicast and the payment paid to each relay node. We use the spanning tree $VMST$ for multicast and what we remain to solve is then how each node on $VMST$ will be paid to compensate its cost.

In a truthful multicast routing protocol, every node is required to report his relay cost, notice a node's declared cost d_k may be different from its actual cost c_k. We use cost vector $d = \{d_1, d_2, \cdots, d_n\}$ to denote all nodes' declared cost. For simplicity, we will use cost vector d to represent the graph $G = (V, E, d)$ if there is no confusion. Thus, $\mathsf{LCP}(s, t, G)$ can be simplified as $\mathsf{LCP}(s, t, d)$ and $VMST(G)$ is simply denoted as $VMST(d)$ when $G = (V, E, d)$. If we change the cost of a node $v_k \in V$ to d'_k, we denote the new graph as $d|^k d'_k$. If we remove one vertex v_k from G, we denote the resulting graph as $d|^k \infty$.

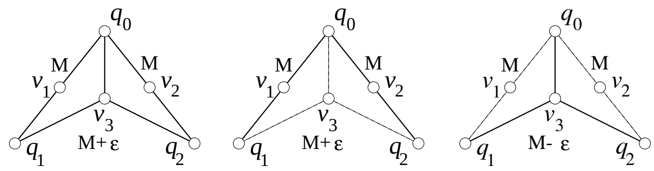
C.1 VCG mechanism Is Not Truthful

VCG mechanisms have been used to design strategy-proof protocols to problems such as unicast [9], [10], single minded auctions [11]. Thus, using VCG mechanism is a nature way to design a payment scheme for multicast. The payment to a node v_k selected in $VMST$ based on VCG mechanisms is as follows

$$p^k_{VCG} = \omega(VMST(d|^k \infty)) - \omega(VMST(d)) + d_k.$$

In other words, the payment to a relay node v_k equals its declared cost plus the difference between the VMST constructed without this node v_k and the VMST constructed using v_k.

Unfortunately, if we compensate relay nodes based on the payment computed using VCG mechanisms, a wireless node may have incentives to lie about its cost to improve its profit, or will refuse to relay the packets since its profit may be negative. Figure 2 illustrates such an example where node v_3 can lie its cost to improve its utility when output is VMST structure. The payment to node v_3 is 0 and its utility is also 0 if it reports its cost truthfully since it will not be selected. Let us see what happens if node v_3 lies its cost to $\underline{c_3} = M - \epsilon$. It is easy to see that the total payment to node v_3 when v_3 reported a cost $\underline{c_3} = M - \epsilon$ is $\omega(VMST(c|^3\infty)) - \omega(VMST(c|^3\underline{c_3})) + \underline{c_3} = 2M - (M - \epsilon) + M - \epsilon = 2M$ and the utility of node v_3 becomes $u^3(c|^3\underline{c_3}) = 2M - (M + \epsilon) = M - \epsilon$, which is larger than $u^3(c) = 0$. This example prevents us from using VCG mechanism to compute payments for relay nodes. Next, we will study how to design a payment scheme p, such that the multicast routing based on VMST and the payment scheme p is truthful.



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

Fig. 2. A node could lie to improve its utility when payment is computed based on VMG mechanisms. Here the cost of nodes are $c_1 = c_2 = M$ and $c_3 = M + \epsilon$.

C.2 Truthful Payment Based on VMST

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 $LE(p, q, d)$ to denote $LE(p, q, VMST(d))$.

Based on the structure $VMST(d)$, we then design a truthful mechanism for calculating the payment to relay nodes on $VMST(d)$ as follows.

Algorithm 2: Truthful payment using $VMST(G)$

1. Each node $v_k \in V$ is required to report a cost, say d_k.
2. For every node $v_k \in V \setminus Q$ in G, first calculate $VMST(d)$ and $VMST(d|^k\infty)$ according to the nodes' declared costs vector d.
3. Find $E_k(d_k)$ which is the set of edges $q_i q_j$ such that $v_k \in \mathsf{LCP}(q_i, q_j, d)$ and $q_i q_j \in VMST(d)$ when node v_k declares a cost d_k.
4. For any edge $e = q_i q_j \in E_k(d_k)$ and any node $v_k \in \mathsf{LCP}(q_i, q_j, d)$, we define the payment to node v_k based on the virtual link $q_i q_j$ as

$$p^k_{ij}(d) = \|LE(q_i, q_j, d|^k\infty)\| - \|\mathsf{LCP}(q_i, q_j, d)\| + d_k. \quad (1)$$

Here $\|\Pi\|$ denotes the total cost of a path Π. The final payment to node v_k based on $VMST(d)$ is

$$p^k(d) = \max_{q_i q_j \in E_k(d_k)} p^k_{ij}(d). \quad (2)$$

We next show that our payment scheme is truthful, i.e., node v_k cannot lie about its cost to improve its non-negative profit. Throughout our proofs, we fix the cost d_{-k} of all nodes other than k.

Assume that node v_k is used in the least cost path $\mathsf{LCP}(q_i, q_j, d)$ and $e = q_iq_j \in VMST(d)$. We first show that node v_k cannot manipulate its declared cost $d_k \neq c_k$ to improve its payment based on the virtual edge q_iq_j from $K(G, R)$.

Lemma 1: Assume that node $v_k \in \mathsf{LCP}(q_i, q_j, d)$ with cost d_k and $e = q_iq_j \in VMST(d)$,then the payment $p_{ij}^k(d)$ does not depend on v_k's declared cost d_k.

PROOF. From the payment definition, when node v_k declares a cost d_k and edge q_iq_j is still in $VMST(d)$, its payment $p_{ij}^k(d)$ based on edge q_iq_j is

$$\|LE(q_i, q_j, d|^k\infty)\| - \|\mathsf{LCP}(q_i, q_j, d)\| + d_k.$$

Notice that the first part $LE(q_i, q_j, d|^k\infty)$ is the longest edge of the unique path from q_i to q_j on tree $VMST(d|^k\infty)$. Clearly, the spanning tree $VMST(d|^k\infty)$ does not depend on d_k. Thus, $LE(q_i, q_j, d|^k\infty)$ is independent of d_k.

Now consider the least cost path $\mathsf{LCP}(q_i, q_j, d)$. From the assumption we know that $v_k \in \mathsf{LCP}(q_i, q_j, d)$, thus, the path $\mathsf{LCP}(q_i, q_j, d)$ remains the same regardless of v_k's declared cost d_k as long as $v_k \in \mathsf{LCP}(q_i, q_j, d)$. Thus, the summation of all nodes' cost on $\mathsf{LCP}(q_i, q_j, d)$ except node v_k is a fixed cost, which equals to $\|\mathsf{LCP}(q_i, q_j, d|^k0)\| = \|\mathsf{LCP}(q_i, q_j, d)\| - d_k$. In other word, the second part $-\|\mathsf{LCP}(q_i, q_j, d)\| + d_k$ is also independent of d_k. This finishes the proof. □

Based on the above lemma, we can rewrite the payment to node v_k based on an edge q_iq_j as

$$p_{ij}^k(d) = \|LE(q_i, q_j, d|^k\infty)\| - \|\mathsf{LCP}(q_i, q_j, d|^k0)\|,$$

when node $v_k \in \mathsf{LCP}(q_i, q_j, d)$ and $q_iq_j \in VMST(d)$.

Given two receivers q_i and q_j and another node v_k, we divide all the paths connecting q_i and q_j in G (denoted by $\Pi(q_i, q_j)$) into two categories: the paths with node v_k (denoted by $\Pi_{v_k}(q_i, q_j)$) and the paths without node v_k (denoted by $\Pi_{-v_k}(q_i, q_j)$). The least cost path in $\Pi_{v_k}(q_i, q_j)$ is denoted as $\mathsf{LCP}_{v_k}(q_i, q_j, d)$, and the least cost path in $\Pi_{-v_k}(q_i, q_j)$ is denoted as $\mathsf{LCP}_{-v_k}(q_i, q_j, d)$. Clearly, the path $\mathsf{LCP}_{-v_k}(q_i, q_j, d)$ is independent of the declared cost d_k of node v_k. Notice $\mathsf{LCP}_{v_k}(q_i, q_j, d)$ and $\mathsf{LCP}_{-v_k}(q_i, q_j, d)$ doesn't depend on node v_k's declared cost d_k. For simplicity, we denote the total cost of nodes on the least cost path $\mathsf{LCP}_{v_k}(q_i, q_j, d)$, other than node v_k, as c_{ij}^k.

In our proof of the the truthfulness, we consider two cases: (1) whether the node has the incentive to lie its cost upward; (2) whether the node has the incentive to lie its cost downward. In order to simplify and clarify our proofs, we use the following notations.

If a node v_k lies its cost upward, we denote the new cost as $\overline{c_k}$, and the VMST calculated from $\overline{c_k}$ as $VMST(d|^k\overline{c_k})$. Similarly, if node v_k lies its cost downward, we denote the new cost as $\underline{c_k}$, and the VMST calculated from $\underline{c_k}$ as $VMST(d|^k\underline{c_k})$.

We first consider the case when the node v_k declares a cost $\underline{c_k}$. In this case, we have the following lemma.

Lemma 2: $E_k(c_k) \subseteq E_k(\underline{c_k})$.

PROOF. Consider any edge q_iq_j from $E_k(c_k)$. We show that this edge q_iq_j is still kept in $E_k(\underline{c_k})$.

We first show that v_k is still in the least cost path connecting q_i and q_j when v_k declares a cost $\underline{c_k}$. Since $v_k \in \mathsf{LCP}(q_i, q_j, d|^kc_k)$, we have $\|\mathsf{LCP}_{-v_k}(q_i, q_j, d|^kc_k)\| > \|\mathsf{LCP}_{v_k}(q_i, q_j, d|^kc_k)\| > |\mathsf{LCP}_{v_k}(q_i, q_j, d|^k\underline{c_k})|$. Remember that $|\mathsf{LCP}_{-v_k}(q_i, q_j, d|^k\underline{c_k})| = |\mathsf{LCP}_{-v_k}(q_i, q_j, d|^kc)| > |\mathsf{LCP}_{v_k}(q_i, q_j, d|^k\underline{c_k})|$, so we got $v_k \in \mathsf{LCP}(q_i, q_j, d|^k\underline{c_k})$.

We then show that $q_iq_j \in VMST(d|^k\underline{c_k})$. Here we consider the node partition $\{Q_i, Q_j\}$ introduced by removing link q_iq_j from $VMST(d)$, where $q_i \in Q_i$ and $q_j \in Q_j$. Remember that the least cost path corresponding to virtual edge q_iq_j contains node v_k and keeps the same, so the weight of virtual edge q_iq_j decreased by $c_k - \underline{c_k}$. When v_k changes its cost from c_k to $\underline{c_k}$, all virtual edges in $K(Q, G)$ decreases at most $c_k - \underline{c_k}$. Thus, q_iq_j is still the bridge over $\{Q_i, Q_j\}$. From the Observation 1, we have $q_iq_j \in VMST(d|^k\underline{c_k})$. This finishes the proof. □

Observation 1: If $\{V_1, V_2\}$ is a partition of vertices in graph $G = (V, E)$ and v_sv_t is the bridge over V_1 and V_2 with minimum length, then $v_sv_t \in MST$.

Similar to Lemma 2, we have the following lemma.

Lemma 3: $E_k(\overline{c_k}) \subseteq E_k(c_k)$.

In order to prove the truthfulness (IC and IR property) of this mechanism, we first give some related definitions. Consider any spanning tree T of graph $K(G, Q)$. Removing any edge $q_iq_j \in T$ will partition the tree T into two trees. All nodes of the two trees form two disjoint vertex sets $Q_i(T)$ and $Q_j(T)$ such that $q_i \in Q_i(T)$ and $q_j \in Q_j(T)$. An edge q_sq_t satisfying the following property is called bridge over $Q_i(T)$ and $Q_j(T)$: $q_s \in Q_i(T)$ and $q_t \in Q_j(T)$ or $q_s \in Q_i(T)$ and $q_t \in Q_j(T)$.

Definition 1: Considering the graph $K(G, Q)$ ($G = (V, E, d)$) and a node partition $\{Q_i, Q_j\}$ of Q, we define the follows:

1. All bridges q_sq_t over node partition Q_i, Q_j of graph $K(G, Q)$ satisfying $v_k \notin \mathsf{LCP}(q_s, q_t, d)$ forms a set $B^{-v_k}(Q_i, Q_j)$, and the one with the minimum length is denoted as $BM^{-v_k}(Q_i, Q_j, d)$ when the nodes' cost vector is d.
2. All bridges q_sq_t over node partition Q_i, Q_j satisfying $v_k \in \mathsf{LCP}(q_s, q_t, d)$ form a set $B^{v_k}(Q_i, Q_j)$, among them the bridge with the minimum length is denoted as $BM^{v_k}(Q_i, Q_j, d)$ when the nodes' cost vector is d.
3. All bridges q_sq_t over node partition Q_i, Q_j form a set $B(Q_i, Q_j)$, among them the bridge with the minimum length is denoted as $BM(Q_i, Q_j, d)$ when the nodes' cost vector is d.

Some observations regarding the bridges are listed as follows (proofs are omitted due to space limit or its simplicity). For a disjoint node partition $\{Q_i, Q_j\}$, we have

1. $BM(Q_i, Q_j, d) = \min(BM^{v_k}(Q_i, Q_j, d), BM^{-v_k}(Q_i, Q_j, d))$.
2. The paths $BM^{v_k}(Q_i, Q_j, d)$ and $BM^{-v_k}(Q_i, Q_j, d)$ in the graph $G = (V, E, d)$ are independent of v_k's declared cost d_k. In other words, node v_k cannot change these two paths in G by merely changing its declared cost d_k.

We are now ready to prove that the payment scheme described in Algorithm 2 satisfies the IR and IC property.

Theorem 2: Our payment scheme satisfies IR property, i.e., for any node v_k

$$u^k(d|^kc_k) \geq 0.$$

PROOF. First of all, if node v_k is not chosen as relay node, then its payment $p^k(d|^k c_k)$ is clearly 0 and its valuation is also 0. Thus, its utility $u^k(d|^k c_k)$ is 0.

When node v_k is chosen as a relay node, we show that its payment is non-negative by showing $p^k_{ij}(d|^k c_k) \geq c_k$ for any edge $q_i q_j \in VMST(d|^k c_k)$ and $v_k \in \mathsf{LCP}(q_i, q_j, d|^k c_k)$. Let $\{Q_i, Q_j\}$ be the node partition introduced by removing link $q_i q_j$ from $VMST(d|^k c_k)$. Consider the unique path connecting q_i and q_j in the spanning tree $VMST(d|^k \infty)$. Clearly, there is at least one edge, say $q_I q_J$, that crosses Q_i and Q_j in this unique path. Here $q_I \in Q_i$ and $q_J \in Q_j$. Clearly, $\|\mathsf{LCP}_{-v_k}(q_I, q_J, d|^k c_k)\| > \|\mathsf{LCP}(q_i, q_j, d|^k c_k)\|$ since edge $q_i q_j$ has the minimum weight among all bridge edges over $\{Q_i, Q_j\}$ when the node cost vector is $d|^k c_k$. By definition, $\|LE(q_i, q_j, d|^k \infty)\| \geq \|\mathsf{LCP}_{-v_k}(q_I, q_J, d|^k c_k)\|$. The theorem then follows from $p^k_{ij}(d|^k c_k) = \|LE(q_i, q_j, d|^k \infty)\| - \|\mathsf{LCP}(q_i, q_j, d|^k c_k)\| + c_k > c_k$. This finishes the proof. □

We then prove that no node can lie about its cost to improve its utility.

Theorem 3: Our payment scheme satisfies the incentive compatibility (IC).

PROOF. We prove the theorem by showing that a node will neither lie up its cost, nor lie down its cost. We consider them case by case as follows.

Case 1: node v_k lies up its cost to $\overline{c_k}$. We prove that v_k doesn't have any incentive to lie upward. If $VMST(d|^k \overline{c_k}) = VMST(d|^k c_k)$, then node v_k gains nothing since the payment to node v_k is independent of its declared cost in this situation (from Lemma 1). If $VMST(d|^k \overline{c_k}) \neq VMST(d|^k c_k)$, from Lemma 3, we know that $E_k(\overline{c_k}) \subseteq E_k(c_k)$. In addition, from Lemma 1, we have $p^k_{ij}(d|^k c_k) = p^k_{ij}(d|^k \overline{c_k})$ for any edge $q_i q_j \in E_k(\overline{c_k})$. This means that v_k can't increase its payment by lying upward.

Case 2: node v_k lies down its cost to $\underline{c_k}$. We further divide this case into two subcases: whether node v_k is originally selected as relay node or not.

Subcase 2.1: node v_k is *not* originally selected as relay node. Obviously, node v_k can possibly improve its utility when $VMST(d|^k \underline{c_k}) \neq VMST(d|^k c_k)$ and v_k is on some edge of $VMST(d|^k \underline{c_k})$ after v_k lies its cost downward. Assume that $v_k \in \mathsf{LCP}(q_i, q_j, d|^k \underline{c_k})$ and $e = q_i q_j \in VMST(d|^k \underline{c_k})$. The payment $p^k_{ij}(d|^k \underline{c_k})$ to node v_k based on $q_i q_j$ is

$$\begin{aligned} & \|LE(q_i, q_j, d|^k \infty)\| - \|\mathsf{LCP}(q_i, q_j, d|^k \underline{c_k})\| + \underline{c_k} \\ = \; & \|LE(q_i, q_j, d|^k \infty)\| - \|\mathsf{LCP}_{v_k}(q_i, q_j, d|^k c_k)\| + c_k \end{aligned}$$

We then prove that $p^k_{ij}(d|^k \underline{c_k}) \leq c_k$.

From the assumption that $v_k \notin VMST(G)$, we have $VMST(d|^k c_k) = VMST(d \backslash v_k)$. Remember that $LE(q_i, q_j, d|^k \infty)$ is the longest edge (say $q^k_I q^k_J$) of the unique path connecting q_i and q_j in $VMST(d|^k \infty)$. Thus, $LE(q_i, q_j, d|^k \infty)$ is also in $VMST(d|^k c_k)$. We will then prove that $\|LE(q_i, q_j, d|^k \infty)\| \leq \|\mathsf{LCP}(q_i, q_j, d|^k c_k)\|$ by contradiction. Assume $\|LE(q_i, q_j, d|^k \infty)\| \geq \|\mathsf{LCP}(q_i, q_j, d|^k c_k)\|$. Then we can replace $q^k_I q^k_J$ with $q_i q_j$ in $VMST(d|^k \infty)$ and get a tree with smaller weight, which is a contradiction. Thus,

$$\begin{aligned} \|LE(q_i, q_j, d|^k \infty)\| &\leq \|\mathsf{LCP}(q_i, q_j, d|^k c_k)\| \\ &= \|\mathsf{LCP}_{v_k}(q_i, q_j, d|^k c_k)\|. \end{aligned}$$

Applying this to our payment scheme, the payment $p^k_{ij}(d|^k \underline{c_k})$ to node v_k is $\|LE(q_i, q_j, d|^k \infty)\| - \|\mathsf{LCP}_{v_k}(q_i, q_j, d|^k c_k)\| + c_k \leq c_k$. This finishes the proof for this subcase.

Subcase 2.2: node v_k is originally selected as relay node. We prove that v_k doesn't have any incentive to lie downward. From Lemma 2, we know that $E_k(c_k) \subseteq E_k(\underline{c_k})$. Thus, we only need focus our attention on these edges in $E_k(\underline{c_k}) - E_k(c_k)$. Consider any such edge $e = q_i q_j \in E_k(\underline{c_k}) - E_k(c_k)$. Let $q^k_I q^k_J$ be the edge with the largest weight among all edges on the unique path connecting q_i and q_j in $VMST(d|^k \infty)$. In the spanning tree $VMST(d|^k \infty)$, if we remove the edge $q^k_I q^k_J$, we have a vertex partition $\{Q^k_I, Q^k_J\}$, where $q_i \in Q^k_I$ and $q_j \in Q^k_J$.

In the graph $K(G, Q)$, we consider the bridge edge $BM(Q^k_I, Q^k_J, c)$ whose weight is minimum when the nodes cost vector is c. There are two cases here: 1) $v_k \notin BM(Q^k_I, Q^k_J, d|^k c_k)$ or 2) $v_k \in BM(Q^k_I, Q^k_J, d|^k c_k)$. We discuss them individually.

The first situation is $v_k \notin BM(Q^k_I, Q^k_J, d|^k c_k)$ which implies $BM^{-v_k}(Q^k_I, Q^k_J, d|^k c_k) = BM(Q^k_I, Q^k_J, d|^k c_k)$. Notice edge $q^k_I q^k_J$ has the minimum weight among all bridge edges over $\{Q^k_I, Q^k_J\}$ when graph $d|^k \infty$ is considered. From assumption $v_k \notin BM(Q^k_I, Q^k_J, d|^k c_k)$, we know that $\|\mathsf{LCP}(q^k_I, q^k_J, d|^k \infty)\|$ is still minimum among all bridge edges over $\{Q^k_I, Q^k_J\}$ when graph G is considered. In other words, $BM(Q^k_I, Q^k_J, d|^k c_k) = \|\mathsf{LCP}(q^k_I, q^k_J, d|^k \infty)\|$. Since $q_i q_j$ is also a bridge edge over $\{Q^k_I, Q^k_J\}$, we have

$$\begin{aligned} \|LE(q_i, q_j, d|^k \infty)\| &= \|\mathsf{LCP}(q^k_I, q^k_J, d|^k \infty)\| \\ &\leq \|\mathsf{LCP}(q_i, q_j, d|^k c_k)\| \end{aligned}$$

Consequently,

$$p^k_{ij}(d|^k c_k) = \|LE(q_i, q_j, c|^k \infty)\| - \|\mathsf{LCP}(q_i, q_j, d|^k c_k)\| + c_k \leq c_k,$$

which implies that v_k will not benefit from lying its cost downward.

The second situation is that $v_k \in BM(Q^k_I, Q^k_J, d|^k c_k)$. From the assumption that $q_i q_j \notin VMST(d|^k c_k)$, we know edge $q_i q_j$ cannot be $BM(Q^k_I, Q^k_J, d|^k c_k)$. Thus, there exists an edge $q_s q_t \neq q_i q_j$ such that $v_k \in \mathsf{LCP}(q_s, q_t, d|^k c_k) = BM(Q^k_I, Q^k_J, d|^k c_k)$, which is guaranteed to be in $VMST(d|^k c_k)$. Obviously, $\|\mathsf{LCP}(q_i, q_j, d|^k c_k)\| \geq \|\mathsf{LCP}(q_s, q_t, d|^k c_k)\|$. Notice that $q_s q_t$ is also a bridge edge over Q^k_I and Q^k_J. Thus, $q^k_I q^k_J$ is on the path from q_s to q_t on graph $VMST(d|^k \infty)$, which implies that $\|\mathsf{LCP}(q^k_I, q^k_J, d|^k \infty)\| = \|LE(q_i, q_j, d|^k \infty)\| \leq \|LE(q_s, q_t, d|^k \infty)\|$. Using Lemma 2, we have $\mathsf{LCP}(q_s, q_t, d|^k \underline{c_k}) \in VMST(G|^k \underline{c_k})$. Thus,

$$\begin{aligned} & p^k_{ij}(d|^k \underline{c_k}) \\ = \; & \|LE(q_i, q_j, d|^k \infty)\| - \|\mathsf{LCP}(q_i, q_j, d|^k \underline{c_k})\| + \underline{c_k} \\ = \; & \|LE(q_i, q_j, d|^k \infty)\| - \|\mathsf{LCP}_{v_k}(q_i, q_j, d|^k c_k)\| + c_k \\ \leq \; & \|LE(q_s, q_t, d|^k \infty)\| - \|\mathsf{LCP}_{v_k}(q_i, q_j, d|^k c_k)\| + c_k \\ \leq \; & \|LE(q_s, q_t, d|^k \infty)\| - \|\mathsf{LCP}(q_s, q_t, d|^k c_k)\| + c_k \\ = \; & p^k_{st}(d|^k c_k) \end{aligned}$$

This inequality concludes that even if v_k lies its cost downward to introduce some new edges in $VMST(d|^k \underline{c_k})$ that contain v_k, the payment based on these newly introduced edges is not larger than the payment on some edges already contained in $VMST(d|^k c_k)$. This finishes the proof. □

D. Mechanism Optimality

We have already proved that our payment scheme is truthful. In this section, we will prove that it is optimal, i.e., the payment to any individual relay node is minimum among all truthful mechanisms based on VMST structure. Before we prove this, we prove the following lemma regarding all truthful payment schemes based on VMST.

Lemma 4: If a mechanism based on VMST with payment function $\tilde{p}$ is truthful, then for every internal node v_k, if $v_k \in VMST(d)$ and all other nodes do not change their declared costs, the payment function $\tilde{p}^k(d)$ should be independent of d_k.
PROOF. We prove it by contradiction. Suppose that there exists a truthful payment scheme such that $\tilde{p}^k(d)$ depends on d_k. There must exist two valid declared costs a_1 and a_2 such that $a_1 \neq a_2$ and $\tilde{p}^k(d|^k a_1) \neq \tilde{p}^k(d|^k a_2)$. Without loss of generality we assume that $\tilde{p}^k(d|^k a_1) > \tilde{p}^k(d|^k a_2)$. Now consider a node v_k with actual cost $c_k = a_2$. Obviously, it can lie its cost as a_2 to increase his utility, which violates the incentive compatibility (IC) property. This finishes the proof. □

Lemma 5: If we have $v_k \in VMST(d)$ for graph $G = (V, E, d)$, then as long as $d'_k < p^k(d)$, $v_k \in VMST(d')$ where $d' = d|^k d'_k$.
PROOF. We prove by contradiction by assuming that $v_k \notin VMST(d')$, which implies $VMST(d') = VMST(d|^k\infty)$. Since $v_k \in VMST(d)$, we have $E_k(d) \neq$. Thus, we can assume there exists some edge $q_i q_j$ such that $p^k(d) = p^k_{ij}(d')$, i.e., its payment is computed based on edge $q_i q_j$ in $VMST(d)$. Let $q_I q_J$ be the longest edge $LE(q_i, q_j, d|^k\infty)$. Let $\{Q_i, Q_j\}$ be the vertex partition introduced by removing edge $q_I q_J$ from the tree $VMST(d|^k\infty)$, where $q_i \in Q_i$ and $q_j \in Q_j$. From the assumption $d'_k < p^k(d)$, we rewrite $d'_k = p^k(d) - \delta$ where $\delta > 0$. The payment to node v_k in $VMST(d)$ is $p^k(d) = \|\mathsf{LCP}(q_I, q_J, d|^k\infty)\| - c^{v_k}_{ij}$, where $c^{v_k}_{ij} = \mathsf{LCP}_{v_k}(q_i, q_j, d|^k 0)$. Thus, $c^{v_k}_{ij} = \|\mathsf{LCP}(q_I, q_J, d|^k\infty)\| - p^k(d)$. When v_k's cost becomes d'_k, the length of the original path $\mathsf{LCP}(q_i, q_j, d)$ in G becomes $c^{v_k}_{ij} + d'_k = \|\mathsf{LCP}(q_I, q_J, d|^k\infty)\| - p^k(d) + d'_k = \|\mathsf{LCP}(q_I, q_J, d|^k\infty)\| - \delta$. In other words, $\|\mathsf{LCP}_{v_k}(q_i, q_j, d')\| = \|\mathsf{LCP}(q_I, q_J, d|^k\infty)\| - \delta$. Thus,

$$\begin{aligned} \|\mathsf{LCP}(q_i, q_j, d')\| &\leq \|\mathsf{LCP}_{v_k}(q_i, q_j, d')\| \\ &< \|\mathsf{LCP}(q_I, q_J, d|^k\infty)\|. \end{aligned}$$

Now consider the spanning tree $VMST(d')$. Remember we assume that $v_k \notin VMST(d')$, i.e., $VMST(d') = VMST(d|^k\infty)$. Thus, among the bridge edges over Q_i, Q_j, edge $q_I q_J$ has the least cost when graph is $d|^k\infty$. However, this is a contradiction to we just proved: $\|\mathsf{LCP}(q_i, q_j, d')\| < \|\mathsf{LCP}(q_I, q_J, d|^k\infty)\|$. This finishes the proof. □

We then show that our payment scheme is optimal among all truthful mechanisms using VMST.

Theorem 4: For structure $VMST$, the payment based on (2) to any node v_k is minimum among all truthful mechanisms based on VMST.
PROOF. We prove it by contradiction. Assume that there is another truthful payment scheme, say $\mathcal{A}$, based on VMST, whose payment is smaller than our payment for a node v_k on a graph $G = (V, E, d)$. Assume that the payment calculated by $\mathcal{A}$ for node v_k is $\tilde{p}^k(d) = p^k(d) - \delta$, where $p^k(d)$ is the payment calculated by our algorithm and $\delta > 0$.

Now consider another graph $G' = (V, E, d')$ where $d' = d|^k d'_k$ and $d'_k = p^k(d) - \frac{\delta}{2}$. From Lemma 5, we know that v_k is still in $VMST(d')$. Using Lemma 4, we know that the payment for node v_k using algorithm $\mathcal{A}$ should be $p^k(d) - \delta$, which is independent of node v_k's declared cost. Here, node v_k's utility is $p^k(d) - \delta - (p^k(d) - \frac{\delta}{2}) = -\frac{\delta}{2} < 0$. Thus, node v_k has a negative utility under payment scheme $\mathcal{A}$ for graph G', which violates the incentive compatibility (IC). This finishes the proof. □

E. Fast Payment Computing

We continue to discuss how to compute the payment to every relay node efficiently. Assume that the original communication graph G has n vertices and m edges.

One method of computing the payment works as follows. First we construct the complete graph from the original graph $G = (V, E, d)$: for every node $q_i \in Q$, we construct the shortest path tree rooted at q_i, which will take time $O(n \log n + m)$. Notice that $\|Q\| = r$. Thus, we need $O(rn \log n + rm)$ time to construct the complete weighted graph $K(G, Q)$. Secondly, we construct the spanning tree $VMST(d)$ on $K(G, Q)$, which takes time $O(r \log r + r^2) = O(r^2)$. Thus, the overall time complexity to construct $VMST(G)$ is $O(r^2 + rn \log n + rm) = O(rn \log n + rm)$.

Next, we study how to find the payment for a single node $v_k \in VMST(d)$ efficiently. In order to calculate the payment for node v_k, we should construct the tree $VMST(d|^k\infty)$, which will take time $O(rn \log n + rm)$. If $v_k \in \mathsf{LCP}(q_i, q_j, d) \in VMST(d)$, then we need to calculate the payment $p^k_{ij}(d)$. Finding the longest edge $LE(q_i, q_j, d|^k\infty)$ will take time $O(r)$. In the worst case, node v_k may appear on $O(r)$ edges of $VMST(G)$. Thus, we can calculate the payment for the single node v_k in time $O(r^2) + O(rn \log n + km) = O(rn \log n + rm)$. In the worst case, there could be $O(n)$ nodes on $VMST(d)$, so we calculate the payment for all relay nodes in tree $VMST(d)$ in time $O(rn^2 \log n + rmn)$. It is natural to ask whether we can compute it more efficiently?

Our improvement is to use the fast payment for unicast as a subroutine. For a pair of nodes $q_i q_j$, we calculate the path $\mathsf{LCP}(q_i, q_j, d|^k\infty)$ for every node $v_k \in \mathsf{LCP}(q_i, q_j, d)$, which can be done in time $O(n \log n + m)$ [10], [12]. It will take $O(r^2 n \log n + r^2 m)$ to find the complete graph $K(d|^k\infty, Q)$ for every node v_k. Finding the MST on each such complete graph will take time $O(r^2)$. Thus, we can construct VMSTs for all these n complete graphs in time $O(r^2 n)$. Based on these

n VMSTs, it will take $O(r^2)$ to calculate the payment for one node. Thus, in the worst case, it will also take $O(r^2 n)$ to calculate the payment to every relay node. Overall, the time complexity of this approach is $O(r^2 n \log n + r^2 m) + O(r^2 n) + O(r^2 n) = O(r^2 n \log n + r^2 m)$. When $r = o(\sqrt{n})$, this approach outperforms the naive approach with time complexity $O(n^2 \log n + mn)$. When r is a constant, the time complexity of the above approach becomes $O(n \log n + m)$, which is optimum.

F. Truthful Payments based on Other Structures

Although we proved that our payment scheme is optimal among all truthful payment schemes based on VMST, there are many other structures for multicast.

One example of multicast structures is the least cost path star (LCPS). For each receiver q_i, we compute the least cost path from the source to q_i, the union of all paths to all receivers is called the least cost path star. We can show that the payment based on VCG mechanisms using LCPS as output is not truthful. Details are omitted here due to space limit. We instead define a truthful payment as follows. First, we compute a payment $p_i^k(d)$ to every node v_k on the least cost path using the scheme for unicast [10], [12]. The total payment to a node v_k is $p^k(d) = \max_{q_i \in Q} p_i^k(d)$. This payment scheme is truthful since node v_k cannot lie about its cost to improve any $p_i^k(d)$.

Notice that the payment based on $p^k(d) = \min_{q_i \in Q} p_i^k(d)$ 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 $p^k(d) = \sum_{q_i \in Q} p_i^k(d)$ is *not* truthful neither.

Although the above payment based on the union of least cost paths is truthful, the structure LCPS could have cost $\Theta(r)$ times the cost of VMST. Figure III-F illustrates such an example. In the example, node s is the source and q_i, $1 \leq i \leq r$

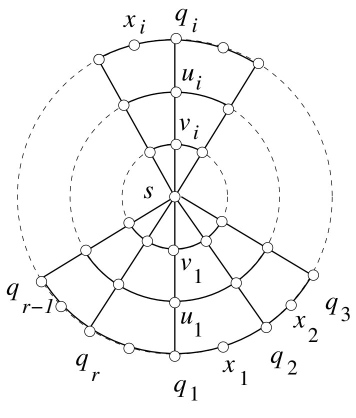


Fig. 3. LCPS could have cost $\Theta(r)$ times of the cost of VMST.

are receivers. Node u_i, v_i and x_i, $1 \leq i \leq r$ are relay candidates. Node u_i and v_i, $1 \leq i \leq r$, each has cost 1, and each w_i has a sufficiently small cost $\epsilon > 0$. Clearly, $\mathsf{LCP}(s, q_i, G) = s v_i u_i q_i$ and it has cost 2. Thus, the total cost of LCPS is $2r$. On the other hand, VMST is path $s v_1 u_1 q_1$ followed by path $q_1 x_1 q_2 x_2 \cdots q_{r-1} x_{r-1} q_r$, and its cost is $2 + (r-1) \cdot \epsilon$. Consequently, in this case, LCPS has cost $\Theta(r)$ times of the cost of VMST. Notice that for any graph G, we will show that LCPS has cost at most r times of the cost of VMST. For any node q_i, clearly, $\|\mathsf{LCP}(s, q_i, G)\| \leq \omega(VMST)$. Thus, $\omega(LCPS) = \sum_{i=1}^{r} \|\mathsf{LCP}(s, q_i, G)\| \leq r \cdot \omega(VMST)$.

Let $\mathcal{P}_{\mathcal{A}}(c)$ be the total payment to all relay nodes under a payment scheme $\mathcal{A}$. Although our payment scheme is based on a structure VMST whose total cost is within 5 times of the minimum cost spanning tree for UDG, we cannot guarantee any relations between the total payments $\mathcal{P}_{VMST}(c)$, $\mathcal{P}_{LCP}(c)$, and $\mathcal{P}_{VCG}(c)$.

IV. Experimental Results

Remember no matter the underline structures is VMST or LCPS, the payment is always greater or equals the actual cost. For a structure H, let $c(H)$ be its cost and $p_s(H)$ be the payment of scheme s based on this structure. We define the overpayment ratio of the payment scheme s based on structure H as

$$OR_s(H) = \frac{p_s(H)}{c(H)}. \quad (3)$$

When it is clear from the context, we often simplify the notation as $OR(H)$.

In [**?**], Archer and Tardos presented a simple example to show that the overpayment for unicast could be as large as $\Theta(n)$. With a little modification of their example, it is not difficult to show that the overpayment ratio for We conducted extensive simulations to study the overpayment ratio of various schemes proposed in this paper.

In our experiments, we compare the performance of structure LCPS and VMST according to three different metrics: actual cost, total payment and overpayment ratio. Figure 4 shows the LCPS and VMST structure when the original graph is a unit disk graph (UDG). Here, the grey nodes are receivers.

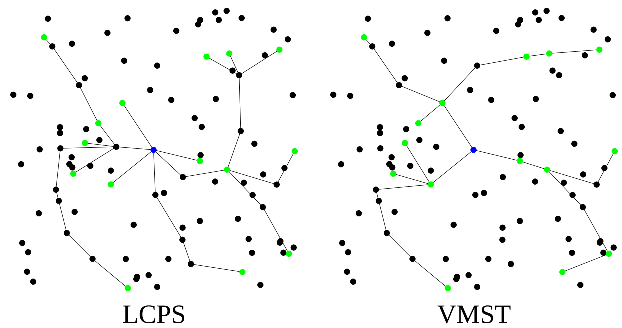


Fig. 4. LCPS and VMST structure

In our experiment, we randomly generate n terminals uniformly in a $2000ft \times 2000ft$ region. The transmission range $range$ of each terminal is set to $300ft$. The cost c_i of a terminal v_i is $c_1 + c_2 * 300^{\kappa}$, where c_1 takes value from 300 to 500, c_2 takes value from 10 to 50. The ranges of c_1 and c_2 we used here reflects the actual power cost in one second of a node to send data at $2Mbps$ rate.

A. Fixed receiver number and varies total node number

In our first experiment, we vary the number of terminals in this region from 150 to 480, and fix the number of sender to 1 and receivers to 20. For a specific number of terminals, we generate 500 different networks, and compare the performance of different structures according to six different metrics: average cost (AC), maximum cost (MC), average payment (AP) and maximum payment (MP), average overpayment ratio (AOR) and maximum overpayment ratio (MOR).

As shown in figure 5, both network cost, payment and overpayment ratio decreases when the number of nodes increases. It is also clear in the figure that our proposed multicast structure VMST is better than the commonly used LCPS structure for all six performance metrics. But we should point out that more computational power is needed to carry out the payment for VMST than for LCPS.

B. Fixed total node number and varied receiver number

In this experiment, we vary the number of receivers in this region from 5 to 45, and fix the number of sender to 1 and total node number to 350. When the number of receivers increases, it is very natural to expect the network cost and total payment will increase. Thus, we define two new metrics that is meaningful to measure the performance: one is Average Cost Per receiver (ACP) which is the network cost divided by the number of receivers and another is Average Payment Per receiver (APP). Notice in this paper, we didn't discuss how to share the payment among all receivers, but these two metrics reflects how much each receivers need to contribute in some extent. For a specific number of terminals, we generate 500 different networks, and also compare the performance of different structures according to six different metrics: AC, ACP, AP, APP, AOR and MOR.

The second part in figure 6 shows that when the number of the receivers increases, the network cost and payment divided by the number of receivers is decreased. It is just what we expected because in wireless ad hoc networks, very node use a broadcast manner to distribute the packet. One interesting observation in the first part of figure 6 is that the network cost and payment doesn't increase when the number of receiver increases. Instead, it display a bimodal manner such that when the number of receivers is greater than some threshold, the total network cost and payment will decrease. We guess this is because of our assumption that all receivers will relay for free. Considering when all nodes are receivers, the network cost and payment will become 0. Another thing deserve attention is in the third part of figure 6, both MOR and AOR for LCPS increase when the number of receivers increase. This interesting phenomena needs future study.

V. Priori Arts

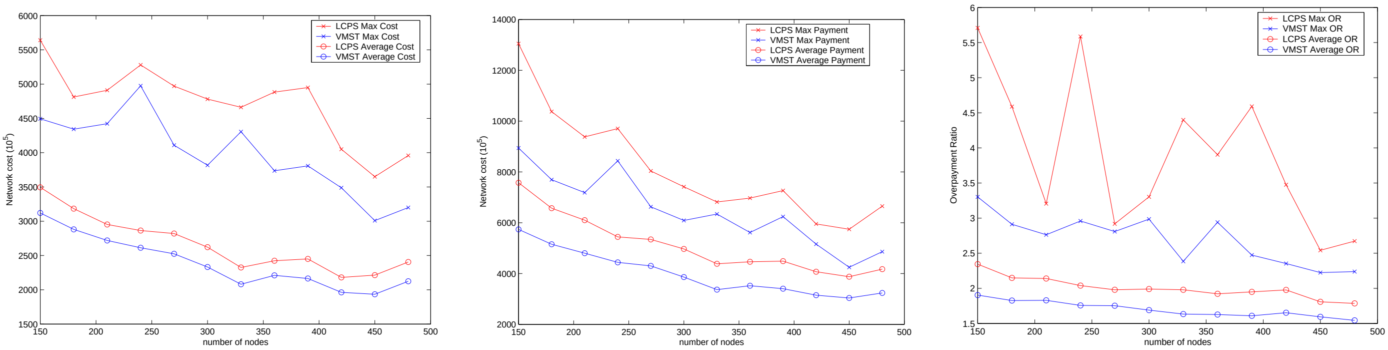
Routing has been part of the algorithmic mechanism-design from the beginning. Nisan and Ronen [5] provided a polynomial-time strategyproof mechanism for optimal unicast route selection in a centralized computational model. In their formulation, the network is modelled as an abstract graph $G = (V, E)$. Each edge e of the graph is an agent and has a private type t^e, which represents the cost of sending a message along this edge. They used the least cost path between two nodes x and y to routing the packet. Their payment scheme is a VCG mechanism. The payment to agent e is 0 if e is not on $\mathsf{LCP}(x, y, G)$, and the payment is $D_{G-\{e\}}(x, y) - D_G(x, y)$ if e is on $\mathsf{LCP}(x, y, G)$. Here $D_{G-\{e\}}(x, y)$ is the cost of the LCP through G when edge e is not presented and $D_G(x, y)$ is the cost of $\mathsf{LCP}(x, y, G)$ through G. In addition, the result in [5] can be easily extended to deal with all-to-all traffics instead of the fixed source and destination node.

Feigenbaum *et. al* [9] then addressed the truthful low cost routing in a different network model. They assumed that each node k incurs a transit cost c_k for each transit packet it carries. For any two nodes i and j of the network, $T_{i,j}$ is the total traffic (number of packets) from i to j. Their payment scheme again is essentially the VCG mechanism. They also gave a distributed method such that each node i can compute a number $p_{ij}^k > 0$, which is the payment to node k for carrying the transit traffic from node i to node j if node k is on $\mathsf{LCP}(i, j)$. The algorithm converges to a stable state after d' rounds, where d' is the maximum of diameters of graph G removing a node k, over all k. Since the mechanism is truthful, any node cannot lie its cost to improve its profit in their distributed algorithm. However, as they pointed [9], it is unclear how to prevent these selfish nodes from running a different algorithms in computing a payment that is more favorable to themselves since we have to rely on these nodes to run the distributed algorithm, although we know that the nodes will input their true values.

For multicasting flow, Feigenbaum *et. al* [17] assumed that there is a multicast infrastructure, given any set of receivers $Q \subset V$, connects the source node to the receivers. Additionally, for each user $q_i \in Q$, 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 receivers R. They also assumed that there is a link cost associated with each communication link in the network and the link cost is *known* to everyone. For each receiver q_i, there is a valuation w_i that this node values the reception of the data from the source. This information w_i is only known to q_i. Node q_i will report a number w_i', which is the amount of money he/she is willing to pay to receive the data. The source node then select a subset $R \subset Q$ of receivers to maximize the difference $\sum_{i \in R} w_i' - C(R)$, where $C(R)$ is the cost of the multicast tree $T(R)$ to send data to all nodes in R. The approach of fixing the multicast tree is relatively simple to implement but could not model the greedy nature of all wireless nodes in the network since it requires that the link costs of the tree are known priori to every node.

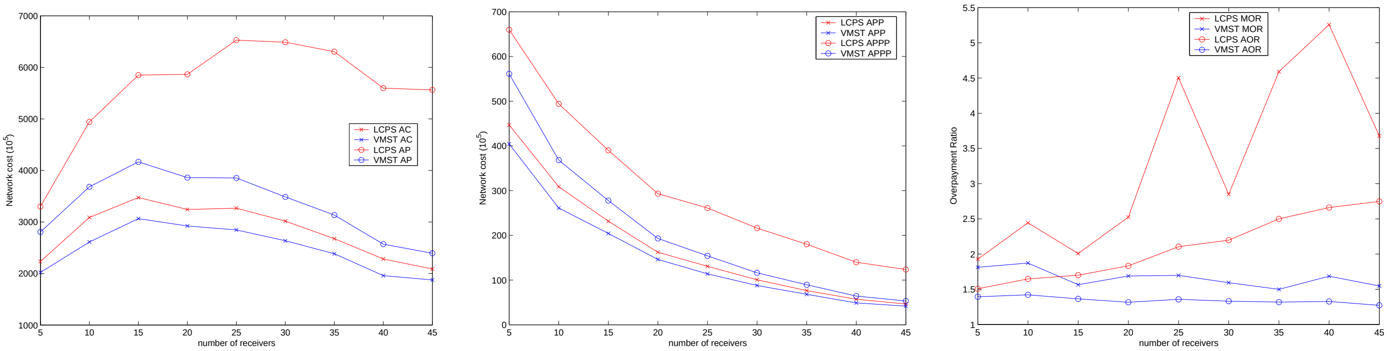
VI. Conclusion

In this paper, we studied how to design a multicast routing protocol for selfish and rational wireless ad hoc networks, in which each wireless node will relay the data packets for other nodes when it receives a payment to compensate its cost. We proposed the first truthful mechanism that is based on a multicast structure whose total cost is within 5 times of the optimum when the wireless networks are modelled by unit disk graphs. We also gave efficient method to compute the payment for all relay nodes on the constructed multicast tree. We proved that each node will follow the protocol and will maximize its profit when it declares its true cost. Our payment scheme also works when the network is modelled by a general graph, but we cannot prove that the total cost of the routing structure is within a constant factor of the optimum. It remains an open problem to design an efficient truthful mechanism that can be computed in



AC / MC for LCPS and VMST AP / MP for LCPS and VMST AOR / MOR for LCPS and VMST

Fig. 5. Results when the number of nodes in the networks varies from 150 to 320 for LCPS and VMST. Here, we fix the transmission range to $300ft$.



AC / AP for LCPS and VMST ACP / APP for LCPS and VMST AOR / MOR for LCPS and VMST

Fig. 6. Results when the number of receivers in the networks varies from 5 to 45 for LCPS and VMST. Here, we fix the number of nodes to 350.

polynomial time when the network is modelled by an arbitrary node weighted graph. Here a protocol is efficient if the total cost of the output structure is within a constant factor of the best possible among any polynomial time computable outputs.

REFERENCES

[1] W. Vickrey, "Counterspeculation, auctions and competitive sealed tenders," *Journal of Finance*, pp. 8–37, 1961.

[2] E. H. Clarke, "Multipart pricing of public goods," *Public Choice*, pp. 17–33, 1971.

[3] T. Groves, "Incentives in teams," *Econometrica*, pp. 617–631, 1973.

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

[5] N. Nisan and A. Ronen, "Algorithmic mechanism design," in *ACM Symposium on Theory of Computing*, 1999.

[6] D.S. Johnson, M. Minko, and S. Phillips, "The prize collecting steiner tree problem: Theory and practice," in *Proc. 11th Symposium on Discrete Algorithms*, 2000, pp. 760–769.

[7] Michel X. Goemans and David P. Williamson, "A general approximation technique for constrained forest problems," *SIAM Journal on Computing*, vol. 24, no. 2, pp. 296–317, 1995.

[8] Sudipto Guha and Samir Khuller, "Improved methods for approximating node weighted steiner trees and connected dominating sets," in *European Symposium on Algorithms*, 1996, pp. 179–193.

[9] J. Feigenbaum, C. Papadimitriou, R. Sami, and S. Shenker, "A BGP-based mechanism for lowest-cost routing," in *Proceedings of the 2002 ACM Symposium on Principles of Distributed Computing.*, 2002.

[10] WeiZhao Wang and Xiang-Yang Li, "Truthful low-cost unicast in selfish wireless network," in *WMAN workshop of IPDPS*, 2004, To appear.

[11] Daniel J. Lehmann, Liaden Ita O'Callaghan, and Yoav Shoham, "Truth revelation in approximately efficient combinatorial auctions," in *ACM Conference on Electronic Commerce*, 1999, pp. 96–102.

[12] Weizhao Wang and Xiang-Yang Li, "Low-cost routing in selfish and rational wireless ad hoc networks," 2004, Submitted for IEEE Transactions on Mobile Computing.

[13] L. Buttyan and J. Hubaux, "Stimulating cooperation in self-organizing mobile ad hoc networks," *ACM/Kluwer Mobile Networks and Applications(MONET)*, vol. 8, no. 5, October 2003.

[14] M. Jakobsson, J.-P. Hubaux, and L. Buttyán, "A Micro-Payment Scheme Encouraging Collaboration in Multi-Hop Cellular Networks," in *Proceedings of Seventh International Financial Cryptography Conference (FC)*, 2003.

[15] L. Buttyan and J. P. Hubaux, "Enforcing service availability in mobile ad-hoc wans," in *Proc. of IEEE/ACM Workshop on Mobile Ad Hoc Networking and Computing (MobiHOC)*, 2000.

[16] L. Blazevic, L. Buttyan, S. Capkun, S. Giordano, J. P. Hubaux, and J. Y. Le Boudec, "Self-organization in mobile ad-hoc networks: the approach of terminodes," *IEEE Communications Magazine*, vol. 39, no. 6, June 2001.

[17] J. Feigenbaum, C. H. Papadimitriou, and S. Shenker, "Sharing the cost of multicast transmissions," *Journal of Computer and System Sciences*, vol. 63, no. 1, pp. 21–41, 2001.