Ad hoc-VCG: A Truthful and Cost-Efficient Routing Protocol for Mobile Ad hoc Networks with Selfish Agents

Luzi Anderegg
Institute of Theoretical Computer Science
ETH Zurich
anderegg@inf.ethz.ch

Stephan Eidenbenz*
Basic and Applied Simulation Science (CCS-5)
Los Alamos National Laboratory
eidenben@lanl.gov

ABSTRACT

We introduce a game-theoretic setting for routing in a mobile ad hoc network that consists of greedy, selfish agents who accept payments for forwarding data for other agents if the payments cover their individual costs incurred by forwarding data. In this setting, we propose Ad hoc-VCG, a reactive routing protocol that achieves the design objectives of truthfulness (i.e., it is in the agents' best interest to reveal their true costs for forwarding data) and cost-efficiency (i.e., it guarantees that routing is done along the most cost-efficient path) in a game-theoretic sense by paying to the intermediate nodes a premium over their actual costs for forwarding data packets. We show that the total overpayment (i.e., the sum of all premiums paid) is relatively small by giving a theoretical upper bound and by providing experimental evidence. Our routing protocol implements a variation of the well-known mechanism by Vickrey, Clarke, and Groves in a mobile network setting. Finally, we analyze a very natural routing protocol that is an adaptation of the Packet Purse Model [8] with auctions in our setting and show that, unfortunately, it does not achieve cost-efficiency or truthfulness

Categories and Subject Descriptors

C.2.2 [**Computer Systems Organization**]: Computer-Communication Networks—*Network Protocols*

General Terms

Algorithms, Performance, Economics

Keywords

VCG mechanism, routing, selfish agents, mechanism design, ad hoc networks, energy efficiency, game theory

*Los Alamos National Laboratory Publication No. LA-UR:03-1404

MobiCom'03, September 14–19, 2003, San Diego, California, USA.
Copyright 2003 ACM 1-58113-753-2/03/0009 ...$5.00.

1. INTRODUCTION

Routing in mobile ad hoc networks has been the subject of intense research efforts over the past few years; these efforts have resulted in numerous proposals for routing protocols (see [33] for a survey). Most routing protocols assume that all the devices that make up the ad hoc network are cooperative, in particular they are willing to act as intermediate nodes in a routing path by forwarding data for other network nodes. The cooperativeness assumption may be reasonable in some settings, but willingness to cooperate can certainly not be assumed in a general ad hoc setting since forwarding data for other network nodes can drain the battery of a node without this node ever being the source or the destination of the data that it forwards. If the network nodes are not owned by a single entity, but are profit-oriented independent agents, they are indeed selfish. In this paper, we propose a method of coping with this selfishness in a mathematically rigorous way while also achieving the globally desirable goal of energy-efficiency. In order to motivate and illustrate our main objective of designing a routing protocol that induces selfish nodes to participate in routing along the globally most cost-efficient path, we describe a game-theoretic model of routing in which we then introduce Ad hoc-VCG, a routing protocol that achieves cost-efficiency and truthfulness.

Energy Efficiency

The traditional approach of dealing with the problem of battery drainage has been to minimize energy consumption from a global point-of-view of the network (but not considering individual network nodes). Indeed, *energy-efficiency* is a key objective in many routing protocols (see [20] for a survey and [10, 12] for more recent work). An energy-efficient routing protocol ensures that a packet from a source node to a destination gets routed along the most energy-efficient path possible via intermediate nodes. The total energy of a routing path is the sum on the emission energy levels used at the source and at each intermediate node. We ignore other types of energy consumption such as listening to signals as they tend to be magnitudes smaller than the emission energy which grows with an exponent of one to six in the distance from one intermediate node to the next. Ideally, an intermediate node uses an emission energy level that will allow its emitted signal to just barely reach the next node on a given route. Modern technology (such as wireless cards) can alter the power level for transmitting a message up to a maximum power $P_{\max}$ and can consequently vary the transmission range. A node receiving a message can determine the signal strength at which it receives this message; if the node additionally knows the energy level or signal strength with which the message was sent off, it can estimate the minimal power required for a message between these two nodes to some precision. Thus, if an emitting node adds its signal emission strength to the header of the packet, the receiving node knows the minimal power required to communicate with the sending node and it can forward this information to the sending node (see [10, 12] for details)[1].

Selfishness and Payments

Energy-efficiency is a must for routing protocols in ad hoc networks. However, energy-efficiency is only desirable from a global point of view, but not from the point of view of an individual and selfish node: if a network node gets chosen as an intermediate node with the duty of forwarding packets very often, the knowledge that it is on the most energy-efficient route is all but comforting since the forwarding actions drain its battery; the reasonable thing to do for this node is to play dead as soon as it realizes that its battery level keeps decreasing, thus simply refusing to forward messages. This non-cooperative behavior is a very basic problem in any ad hoc network in which the nodes are owned by different profit-maximizing entities. It might even be the root cause why the deployment of ad hoc networks has not been as progressive as projected a few years back. The ad hoc network community has recognized this issue and several protocols that stimulate cooperation among nodes have been proposed (see [4] for a survey). These protocols are either based on repudiation models (such as [21, 5, 6, 23, 24]), where nodes are punished for non-cooperation, or based on monetary incentives (such as [7, 9, 3, 35, 19]), where nodes are awarded payments for forwarding messages. In payment-based models, the question arises how much a node should be paid for forwarding messages. An obvious answer would be the cost it incurs when forwarding the message. The cost of forwarding messages could be defined and determined in various ways taking into account factors such as cost of energy used to forward messages, cost of energy when recharging the battery, current battery level, as well as other factors. We propose to model this cost through a parameter: the cost-of-energy parameter c_i of dollars per watt is individual for each node in the network. A node covers its true costs for forwarding a unit-size packet requiring an emission signal strength of P^{emit} watts, if it receives a payment of $c_i \cdot P^{emit}$ dollars for forwarding. On the other hand, if a node does not get a payment that is sufficient to cover its costs, it will simply refuse to forward. Thus, in our model, a node cannot be forced to participate (which is different in repudiation-based protocols). Participation is always voluntary in this sense. The cost-of-energy parameter can be a complex function of several factors (mentioned above) and also of time. For example, a node might have a cost-of-energy that depends inversely on some power of its current battery level. Given the complexity of these functions, we do not pretend to know them, but rather let a node declare the parameter to the other nodes. In terms of implementation, we could imagine that a device user can set the function for parameter c_i according to her preferences.

[1]Similar techniques for power control are widely used in cellular networks [27].

The notion of paying virtual money to intermediate nodes has been proposed in [8], where the authors introduce a currency called NUGLETS.[2] If intermediate nodes receive payments for forwarding messages, some entity will have to pay this money. The source node that wants its message to reach a destination is an obvious candidate to provide at least part of the payment, however, conceptually we could also think of an entity outside the network that funds the payments. Ideally, any routing protocol with a payment scheme should draw all the money from nodes in the network. We call a protocol with this property budget-balanced.

Attractive Cheating

Ad hoc-VCG, our routing protocol for ad hoc networks, consists of two distinct phases that roughly work as follows: In a first phase – the *route discovery* phase –, the underlying weighted graph is computed with vertices representing network nodes and weighted directed edges representing the payments an emitting node has to receive if it transmits a packet along this edge in order to cover its costs. The nodes determine the emission energy levels to reach their neighbors by first sending a packet with high emission energy indicating their emission signal strength in a packet header and receiving a packet back from their neighbors which contains the signal strength at which they received the packet. The nodes also communicate their cost-of-energy c_i to their neighbors. The destination node collects all edge weights and then computes the shortest path in this graph from source to destination, which corresponds to the most energy-efficient path. In a second phase – the *data transmission* phase –, packets are forwarded along the shortest path route and payments are made to the intermediate nodes.

However, there is a big caveat in the *route discovery* phase: the nodes have to indicate the signal strength at which they emit and they also need to forward information regarding their neighbors' received signal strengths. This opens the door to cheating: it may not be in a node's best interest to reveal the emission signal strength correctly. In fact, making exaggerated claims with respect to the emission signal strength will result in a higher payment that the node receives if it is on the shortest or most energy-efficient path. Even more obvious: a node that exaggerates its cost-of-energy c_i will receive a larger payment if it is chosen to be on the shortest path. Similarly, a node may profit from sending false information regarding received signal strength. Of course, we could make the assumption that no node has any information with respect to the current network structure, but that assumption is unrealistic if several communication sessions take place: a node might have partial knowledge of the underlying weighted graph and make a bet based on its knowledge. In reality, the amounts of money that we are dealing with are certainly small for a single transmission; however, cheating is very attractive over a long time horizon, when small amounts add up to considerable sums. Our approach to dealing with this problem is to make the payments attractive enough such that the nodes will not try to cheat. Our goal is to design a protocol that causes all nodes to act truthfully, i.e., to reveal their true costs.

[2]While we will use money as well to make payments, we will do so in a way that is different from what the authors of [8] propose and therefore we will stick to conventional dollars.

Mechanism Design

In conclusion, our objective is to design a protocol that routes along the most cost-efficient path and that is truthful. The setting outlined above is very well suited for analysis by means of game theory, more specifically by mechanism design. The goal of a mechanism design problem is to define a game (i.e., its rules and payoff functions) in such a way that the outcome of the game played by independent agents according to the rules set by the mechanism designer will be the desired outcome, which is called the social optimum. In other words, the game should be designed in such a way that choosing a strategy that results in the social optimum is a dominant strategy for each player, where dominant means that no player has an incentive to unilaterally deviate from the strategy. Generally, any game will result in all players playing dominant strategies and the resulting state is called a dominant-strategy equilibrium. The goal of a mechanism designer is to define rules such that the social optimum is a dominant-strategy equilibrium. See Chapter 23 in [22] for an introduction to economic mechanism design and the seminal paper of Nisan and Ronen [26] for an introduction to algorithmic mechanism design.

In our case, the independent agents correspond to network nodes. The desired outcome or social optimum is achieved if routing is done along the most cost-efficient path. In mechanism design, a game has a set of possible outcomes, which correspond to different routing paths in our situation. Every agent assigns a certain utility to each outcome and it wants to maximize this utility: in our routing game the utility for an agent corresponds to the payment received minus the cost incurred for forwarding the message in the *data transmission* phase. In the terminology of mechanism design every agent has private information called its type. The key idea of the concept of a type is that it is private and only known to the node. The type of a node is the cost of forwarding and is composed of the unit cost-of-energy c_i for that individual node times the emission energy P^{emit} required for forwarding a packet; since the node could falsely declare either factor, the product $c_i \cdot P^{emit}$ is the type of the node. As outlined above, however, this cost is known not only to the node but also to its neighbors who measure the received signal strength and who even forward this information to other nodes. The fact that an agent does not know its own type is a substantial deviation from a classical mechanism design model.

Our Contribution

We propose Ad hoc-VCG, a routing protocol that is guaranteed to find the most cost-efficient path and to be truthful. Unfortunately, our protocol is not budget-balanced, in the sense that the intermediate nodes who are on the routing path receive premiums over their actual costs; however, we will show that the resulting total overpayment is bounded by a factor $2^{\alpha+1}\frac{c_{\max}}{c_{\min}}$, where α is the signal loss exponent and $c_{\max}$ ($c_{\min}$) is the maximum (minimum) cost-of-energy declared by the nodes on the most cost-efficient path. Thus, the protocol guarantees that all payments made will be less than this factor times the cost incurred by routing along the most cost-efficient path.

Our protocol implements a variation of the well-known VCG mechanism, named after Vickrey [34], Clarke [11], and Groves [17], which – in a sense – is a generalized second best sealed bid auction, we thus call it Ad hoc-VCG. The main challenges lie in showing that the protocol remains truthful despite the fact that we deviate from the standard mechanism design model in which the agents know their own type. In our setting, the type can only be determined through interaction with neighboring nodes. The key idea of the VCG mechanism is to make cheating unattractive by making payments as high as a node could possibly expect to obtain by cheating. The VCG-payment for an intermediate node v_i on the shortest path from a source S to a destination D is equal to its own declared cost for forwarding a packet plus a premium, which is defined to be the difference of the overall cost of the shortest path from S to D that does not have v_i as an intermediate node and the shortest path from S to D with v_i (and its declared cost) on it. The payment is defined in such a way that node v_i will get the same amount independent of what it declares as its forwarding cost.

Ad hoc-VCG is a reactive routing protocol, which only takes action and starts computing routing paths when a network node initiates a session. Ad hoc-VCG uses a DSR-like route discovery protocol that channels all information regarding shortest paths to the destination node. The destination node computes the shortest path and all the VCG-payments that need to be made and sends this information back to the source. In the *data transmission* phase, the source sends data packets along with electronic payments to the destination along the shortest path. We focus on the mechanism of the protocol: its truthfulness, cost-efficiency and limited overpayment. The problem of actually making an electronic payment (rather than computing the correct amount) is a different research thread, which is addressed in [7, 35] among others. Ad hoc-VCG is robust against a single cheating node; it may fail in the presence of coalitions of nodes who try to maximize their total payments. However, the fact that each node is owned by a separate profit-maximizing entity may help prevent coalition forming: assume two nodes form a coalition and increase their combined payoff by an amount B; the question of how to split up this additional amount between the two nodes is not trivial; in fact, as each node is selfish, the first node will argue that the second node would not have been able to get any additional payoff without the help of the first node, thus the first node should get all of B; the second node argues accordingly for itself; thus, the coalition may never form as the gain distribution question is hard to settle.

As a last contribution, we briefly analyze a very natural protocol in our setting, in which nodes (starting with the destination) iteratively conduct second best sealed bid auctions among their neighbors to find a cheap path to the destination. In fact, this protocol corresponds to the packet purse model with auctions suggested in [8] adapted into our setting. Somewhat against our intuition, it turns out that our adapted protocol of the packet purse model with auctions does not compute the most cost-efficient path and is not truthful.

Organization

The remainder of this paper is organized as follows: We give pointers to related work in Section 2. We present our mobile ad hoc network model in more detail in Section 3. We propose Ad hoc-VCG in detail in Section 4 and prove its correctness in terms of efficiency and truthfulness in Section 5. A theoretical bound on the overpayment as well as experimental results on the overpayment can be found in Section 6. Section 7 presents a short analysis of the packet purse model with auctions [8] in our game-theoretic setting. We conclude in Section 8. Appendix A contains an alternative version of the *route discovery* phase that pays network nodes even for participating in this first phase (we assume cooperative behavior in this phase in standard Ad hoc-VCG).

2. RELATED WORK

Game theory in general and mechanism design in particular have been used with great success in analyzing routing protocols as well as protocols on other layers in wire-line settings. Nisan and Ronen [26] introduced the concept of algorithmic mechanism design. More closely related to our work is an analysis of TCP/IP as a game [1]. An excellent survey on the state of the art in distributed algorithmic mechanism design can be found in [15]. The work most closely related to our work is [14], where the authors propose a distributed VCG-type mechanism for lowest-cost routing based on BGP; in contrast to our work, their setting is a wireline setting, they assume all pairs of nodes wish to communicate, and their nodes know their own types. A suite of papers have addressed the problem of routing selfish traffic in a network [28, 29].

Selfishness has begun to be studied in wireless networks only recently (see [4] for a survey). Most approaches fall into one of two main categories: approaches rewarding cooperative nodes and approaches punishing non-cooperative nodes. In the first category nodes forwarding packets get monetary incentives for their service. In Buttyan and Hubaux [8] the sender (or the destination) uses a virtual currency called NUGLETS to pay intermediate nodes; the paper proposes different payment models using these NUGLETS. In a follow-up work Buttyan and Hubaux [9] enhanced their results: they introduce a model with a credit counter in each node and analyze four rules when a node forwards packets for other nodes. The ad hoc Participation Economy (APE) [3] uses a different payment system with a central control in the form of "banker nodes". Recently, Zhong, Chen and Yang presented SPRITE [35], which contains a payment scheme such that every node acts truthfully in a game-theoretic sense. For multi-hop cellular networks, [19] uses micro payments to stimulate cooperation. In the second category non-cooperative nodes are identified based on a reputation system and circumvented in the routing process. [21, 5, 23, 24] propose different repudiation systems with respect to how this information is propagated within the network.

A lot of work has been done in the context of malicious nodes in ad hoc networks. Malicious nodes are different from selfish nodes in the sense that it is not their primary concern to get their own messages through the network. Their concern is to disrupt the network. For an overview of the efforts in this context, see [4].

3. OUR MODEL

We have already outlined the basic components of our network and communication model in the previous section. In this section, we present the model in a more formal way.

A mobile ad hoc network $N = (V, E, w)$ consists of a set of nodes $V = \{v_1, \ldots, v_n\}$ that represent mobile devices, a set $E \subseteq V \times V$ of directed edges (v_i, v_j) that connect two nodes, and a weight function $w : E \longrightarrow \mathbb{R}$ for each edge (v_i, v_j) that indicates the cost of transmitting a data packet from node v_i to node v_j. Each node has a unique identification number (for our purposes, this will be the index i of node v_i), but it is not a priori known which nodes are currently in the network, nor is edge set E or weight function w known. The nodes are embedded in the plane.

Each node v_i has an individual parameter c_i indicating its cost-of-energy. As discussed above, the value for c_i can change over time and may depend on the current battery level of the node and its cost for recharging the battery. Parameter c_i models the level of inconvenience caused to the node by asking it to forward a message. A node is willing to accept payment in conventional dollars for forwarding a message and it is greedy: it wants to maximize utility. To this end, it will declare false values for c_i, if this increases the (expected) utility.

As for the behavior of source node S, we propose two alternative models:

- *Source model*
  In this model, we assume that if a source node S wants to send a message to a destination node D over several intermediate nodes that need to forward its message, source node S is willing to pay to the intermediate nodes a premium in addition to their true cost for forwarding the message, if these nodes form a minimum cost path from S to D. Moreover, we assume that source S acts truthfully when reporting its own cost-of-energy parameter and emission energy.[3]

- *Central-bank model*
  In this model, source node S is only willing to pay to the intermediate nodes on the shortest path their true cost for forwarding a message. The intermediate nodes receive their premium from a central bank. The central bank (similar to the concept introduced in [35]) manages accounts for all network nodes by crediting and debiting them as necessary. Nodes communicate with the central bank periodically when they have a good connection to it. The central bank, in turn, periodically debits the accounts of all nodes evenly in order to compensate for the premiums that it paid to intermediate nodes. Evenly distributing the premium payments over all nodes (even those not involved in the communication session) can be regarded as paying a tax or fee for being part of the network. See [16] for a justification and analysis of this approach.[4]

Since we allow individual values c_i for the cost-of-energy for each node, we compute the most cost-efficient path, which might not be equivalent to the most energy-efficient path; since parameter c_i models true cost of energy, however, the most cost-efficient path is more desirable even from the point of view of the mechanism designer who aims to achieve

[3]This assumption is quite strong as it essentially requires the source node to act non-selfishly, whenever it is a source; however, there is a tradition in mechanism design to treat the source (or auctioning agent) in such a special way (see Chapter 23.C, pp. 880 - 881 in [22]).

[4]In most VCG-type mechanisms, the mechanism itself, i.e., an external entity, is assumed to pay the premiums; the authors of [16] argue that distributing this resulting negative surplus evenly to all agents works well in practice and provide mathematical evidence.

a social optimum. Of course, if we require all nodes to have the same cost-of-energy, our protocol will compute the most energy-efficient path.

As for network traffic, we assume that over a long time horizon, each note wants to send a large number of messages, thus making it attractive to collect as much money as possible in order to be able to pay for sending these messages. Moreover, we assume that whenever a source needs to communicate, it will send a large number of unit-sized packets in the communication session. Furthermore we assume that the dominant source of energy consumption are the data messages. We neglect the power needed for sending control messages, such as the messages sent in order to determine minimum emission energies and other messages in the *route discovery* phase. In fact, we assume that all nodes will participate in the *route discovery* phase without receiving payment for it. We believe that this assumption is reasonable for real-life network nodes as the potential payoff for forwarding large amounts of data compensates the nodes for their expenses during the *route discovery* phase. However, from a game-theoretic point of view it is somewhat unsatisfactory as we assume a strong adversary model, where a selfish network node might know in advance that it will not be on the shortest path of the next session and therefore will not bother to participate in the *route discovery* phase; this behavior, in turn, might lead to larger payments to the other nodes. Therefore, we propose – in Appendix A – an alternative version of the *route discovery* phase, in which the nodes get paid even for participating in the *route discovery* phase.

A node v_i is a communication device that can send data by emitting a radio signal and receive data by listening to the transmission medium. When sending data, the node can choose its emission power P_i^{emit}, which determines its range and thus the set of neighbors that receive the transmitted data. A node cannot control the direction in which it sends data, as we assume omnidirectional antennas, and thus data are broadcast to all nodes inside the chosen transmission range. The transmission range of a node v_i depends on the transmitting power P_i^{emit} of the node: the power $P_{i,j}^{rec}$ at which a node v_j at distance d to the transmitting node v_i receives the signal is [12]:

$$P_{i,j}^{rec} = \frac{K}{d^{\alpha}} P_i^{emit}, \tag{1}$$

where K is a constant and α is the distance-power gradient varying between one and six depending on the environment conditions of the network. If this power exceeds a minimum level $P_{\min}^{rec}$, a node v_j at this point can successfully receive the message and falls within the transmission range. Modern wireless cards can alter the power level for transmitting a message up to a maximum power P_{max}^{emit} and can consequently vary their transmission range. Purely for the simplicity of our presentation, we will assume $P_{\max}^{emit} = \infty$ and thus the network is in fact a complete graph. We can set $P_{\max}^{emit}$ to any other value, even for each individual node, without compromising our results (except for the upper bound on the overpayment) as long as we can guarantee that the resulting network is 2-connected, i.e. the removal of a single node does not disconnect the network. Also purely for ease of presentation, we assume $P_{\min}^{rec}$ to be constant for all nodes in the network.

Let node v_i send a message to node v_j using emission power P_i^{emit}. Upon receiving this message, node v_j can determine the signal strength $P_{i,j}^{rec}$ at which it receives the data. If node v_j additionally knows the emission power P_i^{emit} (which v_i could have included in the message), it can estimate the minimal emission power $P_{i,j}^{\min}$ required for a message from node v_i to node v_j as follows[5] using the signal loss equation Eq. 1:

$$P_{i,j}^{\min} = \frac{P_i^{emit}}{P_{i,j}^{rec}} \cdot P_{\min}^{rec} \tag{2}$$

The cost $w(v_i, v_j)$ of sending a packet from v_i to v_j is thus $w(v_i, v_j) := c_i \cdot P_{i,j}^{\min}$.

Nodes can move and their cost-of-energy parameters can change over time. However, we assume a static network during the route discovery phase. This is for ease of presentation only, in fact, this requirement can be dropped if we discretize the minimum emission energies suitably into any given number of levels and add a safety cushion, which then allows mobility even during route discovery.

4. THE AD HOC-VCG PROTOCOL

Ad hoc-VCG does not assume that the nodes have any knowledge about the network, i.e., about the edge weights of the underlying graph. However, if the nodes do have such knowledge, it will not help them to exploit it. Say w.l.o.g. a source node $S := v_0$ needs to send a message to a destination node $D := v_n$. Ad hoc-VCG first computes the most cost-efficient path and then routes the data packets from S to D along this path. Ad hoc-VCG consists of the following two phases:

1. Route discovery
2. Data transmission

Route discovery includes payment computation; the act of making payments to the intermediate nodes is included in the *data transmission* phase. We now present the phases in detail.

Route discovery

In the *route discovery* phase, we compute a minimum energy route from source to destination. It follows mainly [10, 12]. Whenever a source node $S = v_0$ wants to communicate with a destination node $D = v_n$, it initiates the route discovery process by broadcasting a `ROUTE REQUEST` packet. This packet contains:

1. A sequence number $s_{0,n}$
2. The identification 0 of the source node $S = v_0$
3. The identification n of the destination node $D = v_n$
4. The emission power P_0^{emit} of the transmitted signal
5. The cost-of-energy c_0

Every node v_j except S and D that receives the `ROUTE REQUEST` from a node v_i executes the following algorithm:

[5]Equation 2 corresponds roughly to the equations used in [10, 12], where the authors use logarithmic (decibel) measures.

$$S \rightarrow * : \langle \texttt{REQUEST}, s_{0,n}, 0, n, P_0^{emit}, c_0 \rangle$$

$$v_i \rightarrow * : \langle \texttt{REQUEST}, s_{0,n}, 0, v_n, P_{0,1}^{\min}, c_0, 1, P_{1,2}^{\min}, c_1, \ldots, i, P_i^{emit}, c_i \rangle$$

$$v_{\sigma(j)} \rightarrow v_{\sigma(j-1)} : \langle \texttt{REPLY}, s_{k,0}, \sigma(1), \ldots, \sigma(k), P_{\sigma(1),\sigma(2)}^{\min}, \ldots, P_{\sigma(k-1),\sigma(k)}^{\min}, M_{\sigma(1)}, \ldots, M_{\sigma(k)} \rangle$$

Figure 1: Packet headers during the *route discovery* phase

1. Check whether this `ROUTE REQUEST` packet contains information regarding the cost of an edge in the network graph that was not already contained in a previous `ROUTE REQUEST` packet with the same sequence number; if no such information is found, the packet is dropped, otherwise the next step is executed

2. Determine power $P_{i,j}^{rec}$ at which the packet was received

3. Estimate minimum power required for node v_i to transmit to node v_j as $P_{i,j}^{\min} = \frac{P_i^{emit}}{P_{i,j}^{rec}} \cdot P_{\min}^{rec}$

4. Replace the emission power P_i^{emit} value in the `ROUTE REQUEST` packet by $P_{i,j}^{\min}$; append the packet by adding the identification j, the emission power P_j^{emit} of the transmitted signal, and the cost-of-energy c_j; and rebroadcast the packet[6]

Node v_i chooses the value of P_i^{emit} to be whatever it wants it to be; however, a large value will increase v_i's chances to be on a shortest path. We could also require the emission power P_i^{emit} to be the same constant value for each node.

Thus, the `ROUTE REQUEST` packet contains the fields as indicated in Figure 1 for a node sequence $S, v_1, \ldots, v_i$.

Destination D collects the arriving packets and builds up the underlying graph $N = (V, E, w)$. Once it has collected all information it computes the shortest path SP from S to D, say $S, v_{\sigma(1)}, \ldots, v_{\sigma(k)}, D$ (if there is more than one shortest path then the destination randomly chooses one of them). Let $|SP|$ denote the total cost of the shortest path SP. In order to compute the VCG-payments that are to be made to the intermediate nodes $v_{\sigma(1)}, \ldots, v_{\sigma(k)}$, the destination also computes for each node $v_{\sigma(i)}$, $1 \leq i \leq k$, the shortest path $SP^{-\sigma(i)}$ from S to D that does not contain node $v_{\sigma(i)}$ as an intermediate node. The VCG-payment $M_{\sigma(i)}$ for intermediate node $v_{\sigma(i)}$ is then defined to be:

$$M_{\sigma(i)} := |SP^{-\sigma(i)}| - |SP| + c_{\sigma(i)} \cdot P_{\sigma(i),\sigma(i+1)}^{\min} \qquad (3)$$

or in words, $M_{\sigma(i)}$ is the difference of the cost of the shortest path from S to D, if node $v_{\sigma(i)}$ did not exist, and the cost of the shortest path from S to D without the cost incurred by $v_{\sigma(i)}$. The term $c_{\sigma(i)} \cdot P_{\sigma(i),\sigma(i+1)}^{\min}$ in the payment corresponds to the cost incurred by node $v_{\sigma(i)}$; the difference $|SP^{-\sigma(i)}| - |SP|$ is the (always positive) premium paid to node $v_{\sigma(i)}$.

[6]Alternatively, we could only rebroadcast parts of the path contained in the packet that are actually new information and dropping other edges.

Figure 2 shows an example of how payments are calculated. In this small network consisting of six nodes, the edge-weights are written on the edges. The most cost-efficient path from source S to destination D is

$$SP = S, v_2, v_3, D$$

with $|SP| = 5+2+3 = 10$. The shortest path without node v_2 is

$$SP^{-2} = S, v_1, v_4, D$$

with cost $|SP^{-2}| = 7 + 3 + 4 = 14$. The shortest path without node v_3 is

$$SP^{-3} = S, v_2, v_4, D$$

with cost $|SP^{-2}| = 5 + 3 + 4 = 12$. Thus, we have the VCG-payments

$$M_2 = 14 - 10 + 2 = 6$$
$$M_3 = 12 - 10 + 3 = 5$$

Source node S has to send the data messages to node v_2 incurring a cost of 5. In the *source model*, the source also needs to pay amount M_2 and M_3 to nodes v_2 and v_3 respectively for their forwarding service resulting in an overall cost of $5 + 6 + 5 = 16$ for source S; in the *central-bank model*, the source only pays the true cost to the intermediate nodes resulting in an overall cost of $5 + 2 + 3 = 10$, the intermediate nodes receive their premiums from the central bank, who collects it evenly from all network nodes.

Destination node D then creates a `ROUTE REPLY` packet containing the sequence $\sigma(1), \ldots, \sigma(k)$ of the node identifications on the shortest path together with the corresponding minimal required transmit powers $P_{\sigma(i),\sigma(i+1)}^{\min}$ as well as the computed VCG-payments $M_{\sigma(i)}$ for each intermediate node on the shortest path. The `ROUTE REPLY` packet is sent back to source node S along the reversed order of the discovered route. In order to prevent intermediate nodes from altering the information in the `ROUTE REPLY` packet to their advantage, the destination signes this packet with a digital signature, which ensures that the source receives the correct data.

As an aside, note that, unlike [10, 12], an intermediate node rebroadcasts the `ROUTE REQUEST` whenever a new edge is detected (rather than a shorter route). Moreover, we do not consider techniques to shorten the *route discovery* phase such as storing routes in cache etc. as such techniques would – with non-zero probability – prevent us from computing all required edge costs with respect to the shortest path, which in turn would compromise the truthfulness of the protocol.

As for the overhead of the *route discovery* phase, all nodes essentially need to store a local copy of their view of the graph in order to determine whether an incoming `ROUTE REQUEST` message contains new information that needs to be forwarded. Thus, we need a cache structure similar to the *energy aware link cache* introduced in [12]. Each of the n nodes node (except for S and D) may need to forward $O(n^2)$ `ROUTE REQUEST` messages containing at least one new edge-weight, resulting in a total of $O(n^3)$ messages sent in this phase. The destination needs to compute the shortest paths and all replacement paths, which can be done in polynomial time. In order for the VCG mechanism to be truthful against an omniscient adversary, we must guarantee to compute all edge-weights; this partly explains the rather large overhead.

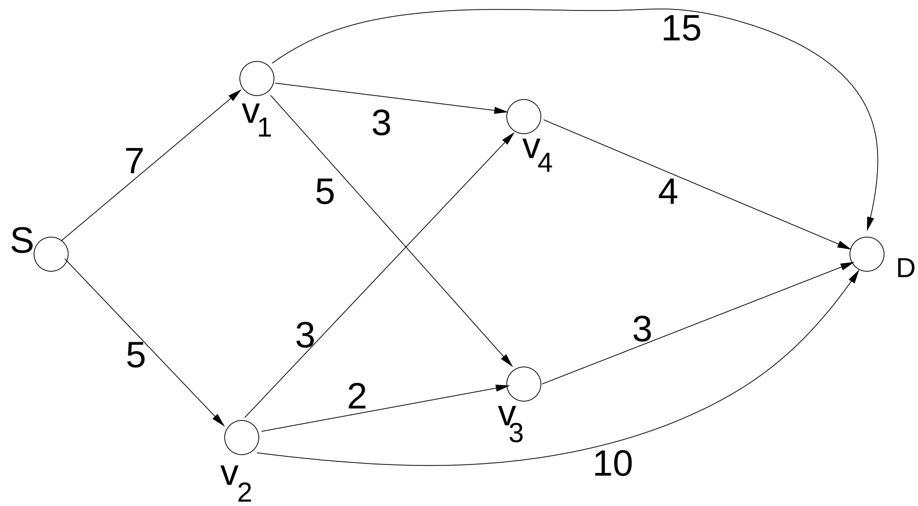

Figure 2: An example network with edge-weights

Data Transmission

In the *data transmission* phase, the data packets are sent along the shortest path $S, v_{\sigma(1)}, \ldots, v_{\sigma(k)}, D$. All nodes use the minimum power required to forward the packages to the next node on the shortest path.

In the *source model*, the source S attaches to each data packet the payments $M_{\sigma(1)}, \ldots, M_{\sigma(k)}$ that it owes to the intermediate nodes $v_{\sigma(1)}, \ldots, v_{\sigma(k)}$. Several models for making such payments have been proposed and any model could be applied here. A first suite of models (such as [35]) requires a universally accepted financial institution that issues digital money that can then be transferred from one agent to another. A second type of model calls for a tamper-proof hardware item that is included in the communication device and in which the money is stored (see [8] for an example).

In the *central-bank model*, the payments are made in a similar way as proposed in [35]: the central bank keeps an account for each node in the network. Whenever a network node is the destination D in a communication session, it records for each intermediate node v_i how much it money it is supposed to receive (i.e., D counts the number of successfully received packets and multiplies it with the computed VCG payment for node v_i); similarly the destination keeps a record of the total that source S owes to other nodes, which is only the truly incurred costs without the premiums. Whenever this node D has a fast connection to the central bank (see [35] for details), it forwards all this information to the central bank, who then acts as a clearing house and credits and debits the accounts accordingly. As no node ever reports credits or debits on its own account to the central bank, such a scheme is truthful. The central bank then pays the premiums to the intermediate nodes; periodically, the central bank sums up all premiums that it has paid out since the last update, say this sum amounts to K, and debits the account of each of the n network nodes by the amount K/n, thus distributing the cost of the premiums evenly across all nodes.

Route Recovery

During the data transmission phase of a communication session, the cost-of-energy parameter c_i of an intermediate node may change, for example if the session is very long and depletes v_i's battery. Also, intermediate nodes may move and thus render the computed minimum emission energies invalid. Additionally, an intermediate node may know (for whatever reason) that its surrogate path has become more expensive and it should therefore receive a larger premium. If any of these events occur, the corresponding intermediate node sends a control message `BROKEN LINK` to the source along the inverse routing path. Upon receiving a `BROKEN LINK` packet or – as a matter of fact – whenever the source deems it necessary, the source initiates a new route discovery phase, in which the destination lets the source know the sequence number of the last packet that it successfully received.

5. ANALYSIS

In this section, we show that Ad hoc-VCG meets the design requirements of cost-efficiency and truthfulness in the presence of selfish nodes.

Cost-efficiency of Ad hoc-VCG follows immediately from the description of Ad hoc-VCG, if we can guarantee truthfulness. Since the destination node in Ad hoc-VCG collects all edge-weights of the network graph, it computes the most cost-efficient routing path from the source node S to the destination node D. Thus,

OBSERVATION 1. *Ad hoc-VCG routes along the most cost-efficient route between source and destination node, if truthfulness is guaranteed.*

If we additionally assume that all nodes have the same cost-of-energy (i.e., c_i is constant for all nodes), then the mechanism also chooses the most energy-efficient path.

As an aside, we note that Ad hoc-VCG guarantees voluntary participation for all nodes (except the source in the source model) because a node only participates in the protocol when it has a non-negative utility in the equilibrium outcome. Since the utility of a node is defined as the payment minus the cost a node on the most cost-efficient path has a non-negative utility. Every other node has no cost and no payment and thus a utility of zero.

Showing that Ad hoc-VCG is truthful is more involved. Truthfulness is given if and only if it is a dominant strategy for each node v_j to always

1. Declare its true cost-of-energy c_j and its true emission power P_j^{emit}
2. Correctly compute and declare the minimum emission power $P_{i,j}^{\min}$ of all its predecessors v_i
3. Correctly rebroadcast all edge-weight information contained in the ROUTE REQUEST packets received without alterations or intentional dropping

We call these items "cheating possibilities".

In accordance with the definition of dominant strategies, we assume an adversarial node to be omniscient: an omniscient node knows all edge-weights of the network graph and it also knows the source and destination of the next communication session. Our protocol is truthful only if even such a powerful node cannot exploit its knowledge to its own advantage. The rationale for using such a strong adversarial model is to prevent real-life nodes from taking bets based on their partial knowledge of the network.

For the analysis we distinguish three types of nodes: the source node S, the destination node D and other nodes. In the *source model*, we simply assume the source to be truthful when declaring its own cost-of-energy c_i and its emission energy. If we did not make this strong assumption, the source node might have reason to underdeclare.[7] Thus, the source will follow the protocol.

In the *central-bank model*, the source is asked to pay for the true costs that are incurred when routing along the shortest path, thus it is in the source's interest that the true shortest path is computed and it will declare its parameters truthfully.

As for the destination D, the destination has no cost (except for sending control messages) and it is in its interest to receive the data in our model; thus it will act truthfully.

For the truthfulness of all other nodes (subsequently called nodes), we treat each cheating possibility in a lemma and then argue that even combining cheating possibilities does not help.

Lemma 1. *Each node v_j declares its true cost-of-energy c_j and its true emission power P_j^{emit}.*

Proof. We show that the payment that v_j receives will not increase if it cheats and, thus, truth-telling is a dominant strategy. We distinguish two cases: node v_j underdeclares the emission energy (i.e., claiming its emission energy is $\overline{P_j^{emit}}$ when it is in fact P_j^{emit} with $P_j^{emit} > \overline{P_j^{emit}}$) or the cost-of-energy (i.e., claiming its cost-of-energy is $\overline{c_j}$ when it is in fact c_j with $c_j > \overline{c_j}$). By underdeclaring these values, node v_j makes its outgoing edges appear cheaper than they are in reality. Since the payments are computed according to Eq. 3, this does not help v_j: if v_j is on the most cost-efficient path with P_j^{emit} and c_j, then it still is on it with either false lower declaration $\overline{P_j^{emit}}$ or $\overline{c_j}$; node v_j gets the same payment in both cases and has the same cost. If v_j is not on the minimum cost path with P_j^{emit} and c_j, but moves itself onto the minimum cost path by declaring $\overline{P_j^{emit}}$ or $\overline{c_j}$, the utility or gain of v_j becomes negative as it will incur costs that are higher than the payment it receives.

Alternatively, v_j can overdeclare the emission power (i.e., $P_j^{emit} < \overline{P_j^{emit}}$) or its cost-of-energy (i.e., $c_j < \overline{c_j}$). Either action makes all outgoing edges of v_j appear more expensive than they actually are. If v_j is not on the minimum cost path when declaring the truth, this action will certainly not move it there. If v_j is on the minimum cost path when declaring P_j^{emit} and c_j, it may either no longer be on it by overdeclaring thus resulting in a loss of revenue, or it may still be on the minimum cost path, but the payment that it receives does not change. □

Lemma 2. *Each node v_j correctly computes and declares the minimum emission power $P_{i,j}^{\min}$ of all its predecessors v_i.*

Proof. We again show that cheating will not increase the utility of a node. Node v_j can either overdeclare or underdeclare the minimum emission power $P_{i,j}^{\min}$ for its predecessor v_i. Underdeclaring (i.e., declaring $\overline{P_{i,j}^{\min}}$ instead of $P_{i,j}^{\min}$ with $\overline{P_{i,j}^{\min}} < P_{i,j}^{\min}$) may move v_j onto the minimum cost path if it is not already on it, but it will also cause predecessor node v_i to use $\overline{P_{i,j}^{\min}}$ as emission energy when forwarding data to v_j in the *data transmission* phase. If v_i uses $\overline{P_{i,j}^{\min}}$ as emission energy, node v_j is outside the transmission range, which will prevent successful communication between source S and D and thus also prevent payment delivery.

Overdeclaring (i.e., declaring $\overline{P_{i,j}^{\min}}$ instead of $P_{i,j}^{\min}$ with $\overline{P_{i,j}^{\min}} > P_{i,j}^{\min}$) may result either in v_j no longer being on the minimum cost path or v_j still being on the minimum cost path but receiving a smaller payment (as $|SP|$ appears larger) with the same cost, thus lowering the utility for v_j. If v_j is not on the minimum cost path and overdeclares $P_{i,j}^{\min}$, it does not influence its payment. □

We also have to guarantee that no node can gain an advantage of the distributed manner of the protocol [25, 14]. The declarations are not sent directly to the destination node (which computes a minimum cost path and the payments) but forwarded by other nodes. These nodes can potentially manipulate the declarations of all preceding nodes in the path of the ROUTE REQUEST packet.

Lemma 3. *Each node v_j correctly rebroadcast all edge-weight information contained in* ROUTE REQUEST *packets received without alterations or intentional dropping.*

Proof. Suppose node v_j decreases an edge-weight information $w(v_g, v_h)$ contained in a ROUTE REQUEST packet. We distinguish the following cases:

- If edge (v_g, v_h) is on the minimum cost path before the weight decrease, it will also be on it after the alteration, but the communication will fail as v_h is out of the transmission range of v_g.

[7]As an example for such a situation, assume that the omniscient source node knows that it will end up paying a large amount due to the true network topology, which exceeds even the true cost that it would incur by communicating with the destination in a single hop; if the source then underdeclares its cost of energy, it can turn the single hop connection to the destination into the shortest path, and thus end up paying less.

- If node v_j is on the minimum cost path SP and edge (v_g, v_h) is on the minimum cost path SP^{-j} without node v_j before the decrease, the weight decrease will result in a smaller payment for node v_j, thus reducing its utility.
- If node v_j is on the minimum cost path SP and edge (v_g, v_h) is not on SP nor on SP^{-j}, the decrease will either have no effect or knock node v_j off the minimum cost path, thus reducing its utility.
- If node v_j is not on SP, reducing the edge weight may put v_j onto the shortest path, but the communication will fail as v_h is out of the transmission range of v_g.

Thus, node v_j cannot increase its utility by decreasing an edge-weight. Now, suppose node v_j increases edge-weight information $w(v_g, v_h)$. This action might increase v_j's utility if either:

- node v_j lies on SP and edge (v_g, v_h) lies on SP^{-j} before the increase, or
- node v_j does not lie on SP and edge (v_g, v_h) lies on SP before the increase.

However, in the first case, all nodes on SP^{-j} will forward their packets containing edge-weight $w(v_g, v_h)$ truthfully and cheating node v_j is not on SP^{-j}, thus enabling the destination node to simply ignore the increased edge-weight. Similarly, in the second case, the nodes on SP will forward their packets containing edge-weight $w(v_g, v_h)$ truthfully and cheating node v_j is not on SP.

Finally, if node v_j intentionally drops packets, all the information in these packets (except for edge-weights of incoming and outgoing edges from v_j) will find its way to the destination through path SP^{-j}. $\square$

In order to see that even combining the cheating possibilities does not help node v_j, first assume that v_j is not on the minimum cost path SP. Then, in order to move itself onto the minimum cost path, node v_j can either try to increase the cost of the true minimum cost path, which will fail as the nodes on the true SP will report truthfully, or v_j can try to decrease the cost of the shortest path that contains v_j, which will result either in a communication failure as a node on the path is out of the range of its predecessor or in a negative utility for v_j if it underdeclares its own cost. Now assume that v_j is on the minimum cost path SP. Then, in order to increase its utility, it can either try to increase the cost of path SP^{-j}, which will fail as the nodes on SP^{-j} report truthfully, or node v_j can try to underdeclare the cost of other nodes on SP while overdeclaring its own cost, which will – once again – result in a communication failure as a node will be out of transmission range for its predecessor. Thus, we have shown:

THEOREM 1. *Ad hoc-VCG is truthful.*

6. OVERPAYMENT

In the *source model*, Ad hoc-VCG forces the source node S that initiates a communication to destination D to pay the premiums to all intermediate nodes that lie on the shortest path

$$SP = \{S = v_{\sigma(0)}, v_{\sigma(1)}, \ldots, v_{\sigma(k)}, D = v_{\sigma(k+1)}\}.$$

In the *central-bank model*, the premiums are paid by the central bank, who periodically debits all network nodes evenly for all premiums. In this section, we study how large the sum P of the premiums can get.

The total cost-of-energy used on the shortest path is

$$|SP| = \sum_{i=0}^{k} c_{\sigma(i)} \cdot P^{\min}_{\sigma(i),\sigma(i+1)}.$$

The total amount VCG paid out in VCG-payments (plus the emission costs of source S) corresponds to $P + |SP|$ and it is

$$\begin{aligned} VCG &= c_S \cdot P^{\min}_{S,\sigma(1)} + \sum_{i=1}^{k} M_{\sigma(i)} \\ &= c_S \cdot P^{\min}_{S,\sigma(1)} - (k-1)|SP| + \sum_{i=1}^{k} |SP^{-\sigma(i)}| \end{aligned}$$

We prove that the overpayment is limited by providing an upper bound on the ratio $\frac{VCG}{|SP|}$.[8]

THEOREM 2. *Let $c_{\max}$ be the maximum, let $c_{\min}$ be the minimum cost-of-energy declared by any network node on the shortest path SP, and let α be the exponent to the distance with which signal strength is lost, then*

$$\frac{VCG}{|SP|} \leq 2^{\alpha+1} \frac{c_{\max}}{c_{\min}}$$

.

PROOF. The key idea of the proof is to bound the length and cost of the shortest paths $SP^{-\sigma(i)}$ without intermediate node $v_{\sigma(i)}$ by analyzing the path

$$\{S, v_{\sigma(1)}, \ldots, v_{\sigma(i-1)}, v_{\sigma(i+1)}, \ldots, v_{\sigma(k)}, D\},$$

which is equivalent to the shortest path SP, but overhops node $v_{\sigma(i)}$ by linking its predecessor $v_{\sigma(i-1)}$ on SP directly to its successor $v_{\sigma(i+1)}$.[9] Thus, we have

$$\begin{aligned} |SP^{-\sigma(i)}| \leq\ & |SP| + c_{\sigma(i-1)} \cdot P^{\min}_{\sigma(i-1),\sigma(i+1)} \\ & - c_{\sigma(i-1)} \cdot P^{\min}_{\sigma(i-1),\sigma(i)} - c_{\sigma(i)} \cdot P^{\min}_{\sigma(i),\sigma(i+1)} \end{aligned}$$

We find a lower bound for $P^{\min}_{\sigma(i-1),\sigma(i+1)}$ as follows, where $d_{i,j}$ is the Euclidean distance between nodes v_i and v_j:

$$\begin{aligned} P^{\min}_{\sigma(i-1),\sigma(i+1)} &= \frac{P^{rec}_{\min}}{K} \cdot d_{\sigma(i-1),\sigma(i+1)}{}^{\alpha} \\ &\leq \frac{P^{rec}_{\min}}{K} \cdot (d_{\sigma(i-1),\sigma(i)} + d_{\sigma(i),\sigma(i+1)})^{\alpha} \\ &\leq \frac{P^{rec}_{\min}}{K} \cdot 2^{\alpha} (d_{\sigma(i-1),\sigma(i)}{}^{\alpha} + d_{\sigma(i),\sigma(i+1)}{}^{\alpha}) \\ &= 2^{\alpha} (P^{\min}_{\sigma(i-1),\sigma(i)} + P^{\min}_{\sigma(i),\sigma(i+1)}) \end{aligned}$$

If we combine these inequalities, we obtain $VCG \leq c_S \cdot P^{\min}_{S,\sigma(1)} + 2^{\alpha} \sum_{i=1}^{k} c_{\sigma(i-1)} (P^{\min}_{\sigma(i-1),\sigma(i)} + P^{\min}_{\sigma(i),\sigma(i+1)})$. For the

[8] In terms of the sum of the premiums P, this ratio is $1 + \frac{P}{|SP|}$.

[9] Unfortunately, a mechanism that always pays according to this rule is not truthful as a simple example shows. However, for our analytical purpose we can use this path; this requires that this edge actually exists, which is given, if $P^{emit}_{\max} = \infty$, but not necessarily for any 2-connected graph as claimed in Section 3.

Number of nodes	10						100						500					
Exponent α	1.5	2	3	4	5	6	1.5	2	3	4	5	6	1.5	2	3	4	5	6
Overpayment ratio																		
average	1.16	1.40	1.94	2.65	4.15	5.99	1.25	1.52	2.10	2.82	3.684	4.834	1.25	1.53	2.01	2.72	3.33	4.358
std dev	.14	.30	.75	1.44	2.86	5.08	.10	.20	.47	.89	1.46	2.42	.08	.10	.36	.70	.86	1.70
maximum	1.62	2.45	5.06	9.68	20.09	32.90	1.59	2.24	4.49	8.57	13.55	31.32	1.50	2.09	4.13	7.85	10.32	16.76
Cheap direct communication	28	42	43	53	53	57	9	9	8	12	5	10	0	0	0	0	0	1

Table 1: Overview of experimental results

overpayment ratio, we thus have:

$$\frac{VCG}{|SP|} \leq 2^{\alpha} \frac{c_{\max}}{c_{\min}} \frac{\sum_{i=1}^{k} P^{\min}_{\sigma(i-1),\sigma(i)} + \sum_{i=0}^{k} P^{\min}_{\sigma(i),\sigma(i+1)}}{\sum_{i=0}^{k} P^{\min}_{\sigma(i),\sigma(i+1)}}$$

$$\leq 2^{\alpha+1} \frac{c_{\max}}{c_{\min}}$$

□

Since exponent $\alpha \leq 6$ is usually assumed and since the ratio $\frac{c_{\max}}{c_{\min}}$ between the maximum and minimum cost-of-energy can be assumed to be reasonably low (in fact, we could impose a minimum and maximum cost-of-energy upon all nodes without loosing truthfulness), Theorem 2 gives quite a good upper bound for the overpayment ratio; we will further substantiate this claim by looking at simulation results. Theorem 2 is in contrast to results from wire-line networks, where it is known that such a constant bound does not exist for general network graphs [2]. [10]

In order to compute the average overpayment ratio, we have conducted experiments in the following setup: in a first set of experiments, we randomly placed 10 nodes onto a rectangular grid of size 1500×500. We then computed the resulting graph with edge-weights, where an edge between two nodes v_i and v_j was assigned the weight ${d_{i,j}}^{\alpha}$ (i.e., the Euclidean distance between the two nodes v_i and v_j taken to the power of α). We ran experiments for signal loss exponents $\alpha = \{1.5, 2, 3, 4, 5, 6\}$. This setup corresponds to a setup, where all nodes have the same cost-of-energy. We randomly picked a source-destination pair among these nodes and computed the overpayment ratio for this pair. In a second and third set of experiments, we placed 100 nodes and 500 nodes, all other parameters remained the same. We ran all three sets of experiments one thousand times for each value of α. Table 6 presents overview results, giving the average overpayment ratio, the standard deviation and the maximum for each exponent α and scenario. It also indicates the number of source-destination pairs chosen for which the source would pay less by communicating directly with the destination without intermediate nodes. Figure 3 shows the distribution and density functions of the overpayment ratio that we obtained from our experiments.

From Table 6, we see that the average overpayment ratio roughly varies between 1.16 and 1.25 for $\alpha = 1.5$ (i.e., the source needs to pay 16 to 25 per cent more than the cost of a shortest path on average in the *source model*). As expected, the average increases with increasing path loss exponent α as surrogate "SP^{-i}"-paths become more expensive. The average values decrease with increasing node numbers, which is to be expected as more nodes imply more possibilities for (and thus cheaper) surrogate paths; however, this observation only holds for $\alpha > 5$, for smaller α the values do not change much with increasing number of nodes. The standard deviation of the overpayment ratio ranges from 0.08 to 5.08. It increases with the path loss exponent and clearly decreases as the number of nodes increases; thus having more nodes seems to decrease the spread of the overpayment ratios. The maximum overpayment ratios that were paid during the 1000 experiments are clearly below the theoretical bound of $2^{\alpha+1}$ from Theorem 2; they tend to be approximately one fourth of the theoretical upper bound, which shows that our theoretical analysis is not terribly loose, but still does leave some room for improvement. The maxima clearly fall with increasing number of nodes, which is further evidence that having a large number of nodes offers better surrogate paths. The number of experiments in which the source would have had less expenses by communicating with the source directly in a single hop heavily depends on the number of nodes. Once we have 100 nodes, these cases become very rare.

The distribution and density functions shown in Figure 3, where the 100-node scenario is on the left and the 500-node scenario on the right, exhibit classic behavior with a clear peak at the average overpayment ratios in the density functions only for $\alpha = \{1.5, 2\}$. For $\alpha > 3$, the density curves seem quite close to each other, while the distribution curves clearly show that path loss exponent α still has a large influence on the experiment for large α.[11]

These experimental results show that the overpayment is certainly a factor that cannot be neglected. This can be considered a proof-of-concept for our model as neglectable overpayment ratios would imply that a good surrogate "SP^{-i}"-path almost always exists, thus making cheating unattractive in the first place.

7. ANALYSIS OF PACKET PURSE MODEL WITH AUCTIONS

We briefly describe a very natural routing protocol that implements the basic idea of iterative second best sealed-bid

[10]The reason for the bounded overpayment lies in the fact that in our setting the agents fulfill the so-called "agents are substitutes"-property as defined in [18] respectively have frugality ratio 1 as defined in [32].

[11]The corresponding curves are not shown for the 10-node scenario as they do not provide additional information. However, they are somewhat different as a lot more experiments achieve an overpayment ratio of 1, which is due to the fact that the direct path is very often the shortest path with only ten nodes in the network.

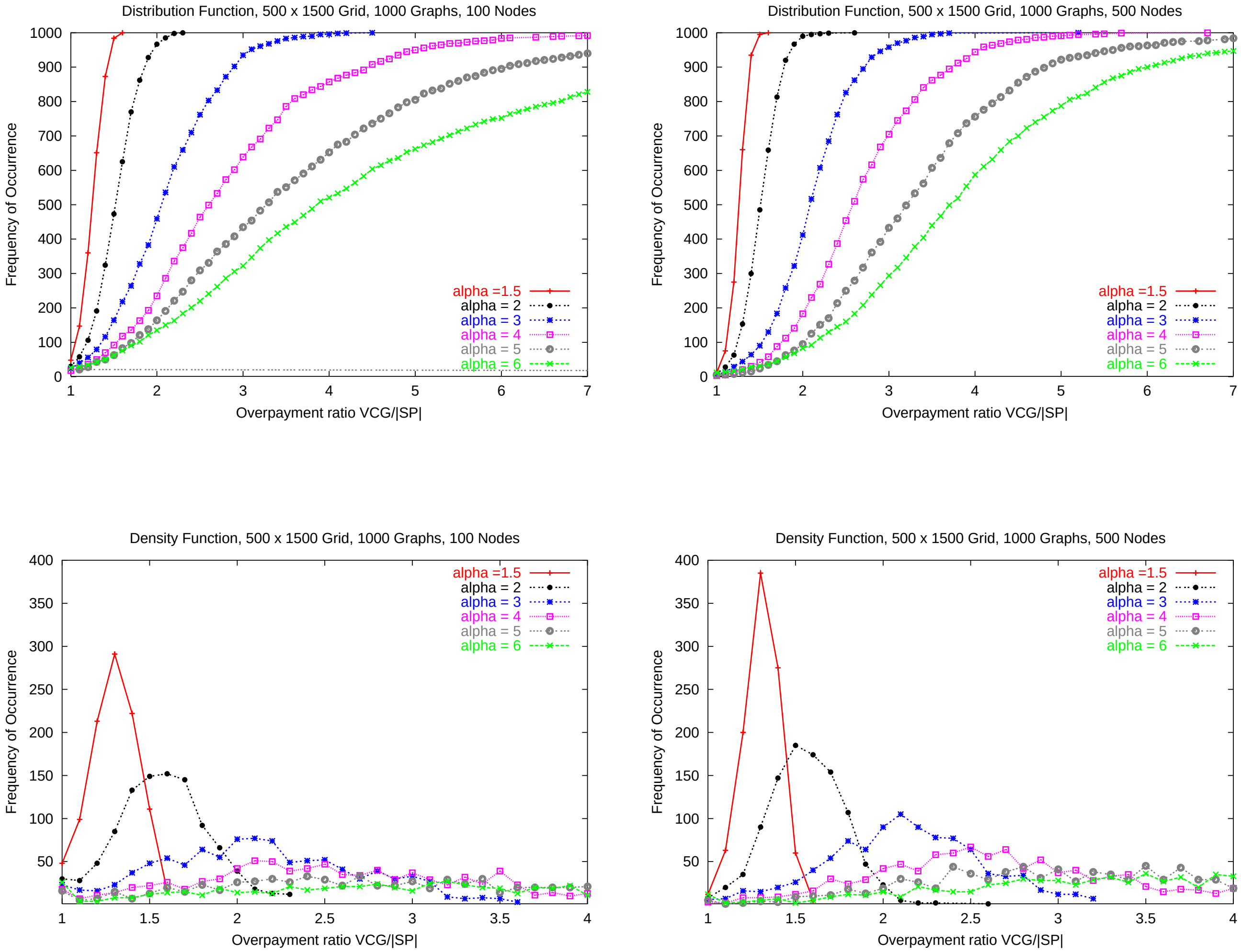

Figure 3: Overpayment ratio distribution and density functions of our experiments

auctions, which in its spirit is the Packet Purse Model with Auctions from [8], and represents a possible way of adopting their protocol to our model. The fact that the nodes can choose the emission power levels in our model is the crucial distinguishing feature.

Suppose a source node S wants to send a message to a destination node D. In order to discover a route, S conducts a second-best sealed bid auction among its neighbors by asking them to name the smallest amount of money for which they would be willing to forward the message (possibly over several intermediate nodes) to the destination D.

The neighbors $v_1^s, \dots v_k^s$ of S make offers $off_1, \dots, off_k$ to the source and also let the source know the energy level at which they received the signal from the source, thus enabling the source to compute the minimum emission energies $P_{S,v_i^s}^{min}$ required to reach its neighbors. Source S then adds its own cost of forwarding to its neighbors to the offers resulting in values $off_i' = off_i + c_S P_{S,v_i^s}^{min}$ for all neighbors v_i^s. The neighbor v_i^s with minimum off_i' wins the auction and the source forwards the message to neighbor v_i^s. The payment that node v_i^s receives from the source is the difference of the second-lowest offer $off_{i'}'$ minus the lowest offer off_i' plus the original offer off_i of v_i^s; in other words: the payment is equivalent to the second-lowest offer $off_{i'}'$ minus the cost of the source for forwarding $c_S P_{S,v_i^s}^{min}$. Thus, the payment is "VCG-like". Before the neighbors can make their offers, they determine their cost of forwarding the message to the destination D by conducting a second-best sealed bid auction among their neighbors (just as described for the source); as before, to the result of their auction, they will add their own cost for forwarding the message to the winner of the auction. This process will continue iteratively until the destination can be reached efficiently in a single hop.

If we analyze this protocol, it is obvious that the source has to pay a premium to its neighbor similar to Ad hoc-VCG. As for the design objectives of truthfulness and cost-efficiency, Figure 4 illustrates that the protocol cannot guarantee these objectives, which may seem counter-intuitive.

In Figure 4, the numbers on the edges represent edge weights, the number above the nodes represent the offer made to the previous node(s). The most cost-efficient path would be S, v_7, v_9, D with total cost 3, but the protocol chooses path S, v_1, v_2, D with cost 12, since the offer by v_1, which is 16, is the lowest offer made to the source S. Thus, the protocol is not cost-efficient. Moreover, node v_5 can successfully cheat in the following way: node v_5 reports a

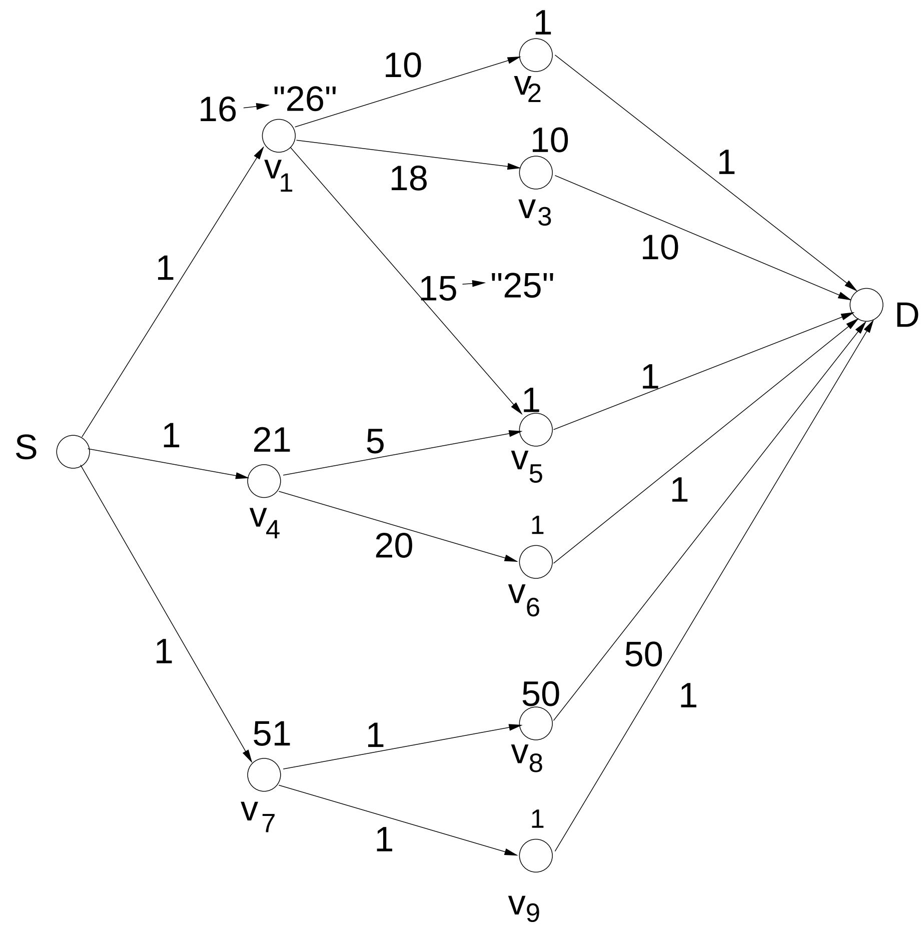

Figure 4: An example for the Packet Purse Model with Auctions

smaller receiving energy than actually true to node v_1, thus increasing the cost of edge (v_1, v_5) from 15 to 25 as indicated in the figure. This, in turn will cause node v_1 to increase its offer to source S from 16 to 26, which – in turn – will make the offer of v_4, which is 21, the most attractive for source S to choose, thus resulting in a chosen path S, v_4, v_5, D with node v_5 receiving a payment of $21 - 5 + 1 = 17$. Thus, node v_5 now makes a profit of $17 - 1 = 16$ as opposed to a profit of 0 before cheating.

Our adaptation of the packet purse model with auctions, which we consider to be a very natural protocol, therefore does not guarantee truthfulness or cost-efficiency. Analyzing other protocols designed for selfish nodes in our model is an interesting direction for future research.

8. CONCLUSION

We have introduced a game-theoretic setting for the routing layer of mobile ad hoc networks, in which the communication nodes are assumed to be selfish and in which the communication nodes need to declare their cost-of-energy in order to compute a cost-efficient communication path. We have presented Ad hoc-VCG, a reactive routing protocol for mobile ad hoc networks that is robust against individual selfishness of the communication nodes and achieves cost-efficiency and truthfulness. Ad hoc-VCG works well for settings of ad hoc networks where communication sessions between two nodes are generally long and the routing path does not change dramatically during a session. If these conditions are not met, the considerable overhead of the *route discovery* phase stalls the network.

While we have introduced Ad hoc-VCG as a reactive protocol, a proactive version could be proposed as well. In fact, showing that it is always in a nodes best interest to (i) send out a position update, whenever (and only if) it has moved or its cost-of-energy has changed, and to (ii) forward such update messages is a case analysis very similar to our proofs in Section 5. We have therefore omitted a formal introduction of a proactive version, but it can be done using similar concepts as the reactive version. In view of the rather inefficient route discovery phase, a proactive version might actually make sense in some settings as edge-weights that do not change do not get re-computed for each communication session.

We believe Ad hoc-VCG is a first step in designing a practical protocol that achieves truthfulness and cost-efficiency. In its current form, the protocol leaves room for improvement in several aspects, which should be pursued in future research.

One aspect is the excessive overhead. Ad hoc-VCG presently requires complete knowledge of the underlying graph, which inevitably creates a large overhead in the *route discovery* phase. In order to guarantee truthfulness and cost-efficiency, any protocol must not only guarantee to always route along the shortest path but also to correctly compute the weight of all shortest surrogate "SP^{-i}"-paths for the nodes on the shortest path. As soon as this is not guaranteed, an omniscient adversary node can exploit its knowledge to its advantage and make the protocol untruthful. Thus, standard techniques such as using cache methods to shorten the route discovery phase do not seem suitable here. However, designing a leaner route discovery phase (both in terms of message complexity and memory requirements) is not impossible and is in fact a promising research direction for the future. A second approach to design leaner route discovery procedures would be to reduce the power of an adversary node or to use a less strict notion of equilibria, such as Bayesian Nash equilibria, which rely on a known distribution of the types of all nodes involved. A third approach in refining Ad hoc-VCG could consist of defining lean protocols that operate locally with the property that they can guarantee to violate the design objectives of truthfulness and cost-efficiency at most by a certain factor; both positive and negative (i.e., impossibility) results would be valuable in this setting.

A second aspect is coalition-forming. While Ad hoc-VCG is truthful with respect to a single cheating node, it is not robust against coalitions of cheating nodes. Designing a protocol in our setting that is group-strategy-proof and cost-efficient at the same time remains a challenge.

As a third aspect, we have not focused on actual payment delivery, but only on payment computation. Payment delivery schemes such as Sprite [35] or Nuglets [9] could be combined with Ad hoc-VCG to achieve a fully functioning system.

Recent advances in distributed algorithmic mechanism design with more in-depth analysis of algorithmic aspects of truthfulness [30] could be applied to Ad hoc-VCG; studies in this direction promise to lead to a better understanding of implicit assumptions in the Ad hoc-VCG mechanism.

We believe that game theory and mechanism design are excellent tools to study other network layer functions (such as topology control or MAC layer; see [13] for a first step) and also other optimization functions as well. For example, the problem of finding a route in our setting such that the total payment that the source has to make is minimized is an interesting open problem.

Acknowledgements

We would like to express our sincere gratitude to anonymous referees for valuable coments that helped us improve the presentation of the paper. We also wish to thank Jeffrey Shneidman for advising us on issues of budget balance and overpayment.

9. REFERENCES

[1] A. Akella, S. Seshan, R. Karp, S. Shenker, C. Papadimitriou; Selfish Behavior and Stability of the Internet: A Game-Theoretic Analysis of TCP; *in: Proceedings of SIGCOMM'02,* 2002.

[2] A. Archer and É. Tardos; Frugal Path Mechanism; *in: Proceedings of 13th Symposium on Discrete Algorithms, ACM Press/SIAM, New York/Philadelphia,* pp. 991-999, 2002.

[3] M. Baker, E. Fratkin, D. Guitierrez, T. Li, Y.Liu, V. Vijayaraghavan; Ad hoc Participation Economy; May 2001.

[4] S. Buchegger, J. Le Boudec; Cooperative Routing in Mobile Ad-hoc Networks: Current Efforts Against Malice and Selfishness; *in: Lecture Notes on Informatics, Mobile Internet Workshop, Informatik 2002, Dortmund, Germany,* October 2002.

[5] S. Buchegger, J. Le Boudec; Nodes Bearing Grudges: Towards Routing Security, Fairness, and Robustness in Mobile Ad Hoc Networks; *in: Proceedings of the Tenth Euromicro Workshop on Parallel, Distributed and Network-based Processing,* pp. 403 - 410, IEEE Computer Society, 2002.

[6] S. Buchegger, J. Le Boudec; Performance Analysis of the CONFIDANT Protocol: Cooperation Of Nodes – Fairness in Dynamic Ad-hoc NeTworks; *in: Proceedings of IEEE/ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC), Lausanne Switzerland,* 2002.

[7] L. Buttyan and J. Hubaux; Enforcing Service Availability in Mobile Ad-Hoc WANs; *in: Proceedings of IEEE/ACM Workshop on Mobile Ad Hoc Networkind and Computing (MobiHOC), Boston,* August 2000.

[8] L. Buttyan and J. Hubaux; Nuglets: a virtual currency to stimulate cooperation in self-organized ad hoc networks; *in: Technical Report EPFL, DSC,* 2001.

[9] L. Buttyan and J. Hubaux; Stimulating Cooperation in Self-Organizing Mobile Ad Hoc Networks; *ACM/Kluwer Mobile Networks and Applications,* Vol. 8 No. 5, October 2003.

[10] P. Bergamo, A. Giovanardi, A. Travasoni, D. Maniezzo, G. Mazzini, M. Zorzi; Distributed Power Control for Energy Efficient Routing in Ad Hoc Networks; *Accepted for publication in WINET.*

[11] E. Clarke; Multipart Pricing of Public Goods; *in: Public Choice* 11, pp. 17-33, 1971.

[12] S. Doshi, S. Bhandare, T. Brown; An On-demand minimum energy routing protocol for a wireless ad hoc network; *in: ACM Mobile Computing and Communications Review,* vol. 6, no. 3, July 2002.

[13] S. Eidenbenz, V.S.A. Kumar, S. Zust; Equilibria in Topology Control Games for Ad Hoc Networks; *in: Proceedings of DIALM-POMC'03,* 2003.

[14] J. Feigenbaum, Ch. Papadimitriou, R. Sami, S. Shenker; A BGP-based Mechanism for Lowest-Cost Routing; *in: Proceedings of the 21st ACM Symposium on Principles of Distributed Computing (PODC),* pp. 173-182, July 2002.

[15] J. Feigenbaum, S. Shenker; Distributed Algorithmic Mechanism Design: Recent Results and Future Directions; *in: Proceedings of the 6th International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communication (Dial-M'02), ACM Press, New York,* pp. 1 -13, 2002.

[16] J. Green and J. Laffont; Incentives in Public Decision Making; *in: Studies in Public Economies, vol. 1, North Holland, Amsterdam,* pp. 65-78, 1979.

[17] T. Groves; Incentives in Teams; *in: Econometrica 41,* pp.617-663, 1973.

[18] R. Garg, V. Kumar, A. Rudra, A. Verma; Coalitional games on graphs: core structure, substitutes and frugality; *Poster session ACM Conference on E-commerce 2003,* pp. 248-249, June 2003.

[19] M. Jakobsson, J.P. Hubaux, L. Buttyan; A micro-payment scheme encouraging collaboration in multi-hop cellular networks; *In: Proceedings of Financial Crypto 2003, La Guadeloupe,* January 2003.

[20] S. Lindsey, K. Sivalingam, C. Raghavendra; Power Optimization in Routing Protocols for Wireless and Mobile Networks; *in: Handbook of Wireless Networks and Mobile Computing, I. Stojmenovic (ed.), John, Wiley and Sons, Inc., New York,* pp. 407 - 423, 2002.

[21] S. Marti, T. Giuli, K. Lai, M. Baker; Mitigating routing misbehavior in mobile ad hoc networks; *in: Proceedings of The Sixth International Conference on Mobile Computing and Networking 2000, Boston, MA,* Aug. 2000.

[22] A. Mas-Colell, M. Whinston, J. Green; Microeconomic Theory; *Oxford University Press, New York,* 1995.

[23] P. Michiardi, R. Molva; Game theoretic analysis of security in mobile ad hoc networks; *Research Report No. RR-02-070, Institut Eurecom,* 2002.

[24] P. Michiardi, R. Molva; CORE: A Collaborative Repudiation Mechanism to enforce node cooperation in Mobile Ad hoc Networks; *in: Sixth IFIP conference on security communications, and multimedia (CMS 2002), Portoroz, Slovenia,* 2002.

[25] D. Monderer, M. Tennenholtz; Distributed Games: From Mechanisms to Protocols; *in: Proceedings of the 16th National Conference on Artificial Intelligence,* pp. 32-37, 1999.

[26] N. Nisan, A. Ronen; Algorithmic mechanism design; *in: Games and Economic Behavior* 35, pp. 166-196, 2001.

[27] T. Rappaport; Wireless Communications: Principles and Practice; *Prentice Hall,* 2002.

[28] T. Roughgarden, É. Tardos; How Bad is Selfish Routing?; *in: Journal of the ACM,* 49(2), pp. 236-259, March 2002.

[29] T. Roughgarden; How unfair is optimal routing?; *in: Proceedings of the 13th annual ACM-SIAM symposium on Discrete algorithms (SODA),* pp. 203-204, San Francisco, 2002.

[30] J. Shneidman, D.C. Parkes, M. Seltzer; Overcoming Rational Manipulations in Distributed Mechanism Implementations; *Harvard Technical Report TR-12-03,* 2003.

[31] V. Srinivasan, P. Nuggehalli, C.F. Chiasserini, R.R. Rao; Energy Efficiency of Ad Hoc Wireless Networks with Selfish Users; *in: European Wireless conference, EW* 2002.

[32] K. Talwar; The Price of Truth: Frugality in Truthful Mechanisms; *in: Proceedings of the 20th Annual Symposium on Theoretical Aspects of Computer Science (STACS),* pp.608-619, Berlin, 2003.

[33] Y. Tseng, W. Liao, S. Wu; Mobile Ad Hoc Networks and Routing Protocols; *in: Handbook of Wireless Networks and Mobile Computing, I. Stojmenovic (ed.), John, Wiley and Sons, Inc., New York,* pp. 195 - 218, 2002.

[34] W. Vickrey; Counterspeculation, auctions, and competitive sealed tenders; *in: Journal of Finance 16,* pp. 8-37, 1961.

[35] S. Zhong, Yang Richard Yang, J. Chen; Sprite: A Simple, Cheat-Proof, Credit-Based System for Mobile Ad-hoc Networks; *in: Proceedings of INFOCOM 2003,* pp. 1987-1997, March 2003.

APPENDIX

A. ALTERNATIVE VERSION OF ROUTE DISCOVERY PHASE

In order to ensure participation in the *route discovery* phase, we propose an alternative version of the *route discovery* phase of Ad hoc-VCG that pays the nodes even for participating in the *route discovery* phase. The alternative protocol pays a unit amount for each time a node forwards a `ROUTE REQUEST` packet. Thus, we assume that there exists a publicly known maximum cost-of-energy $c_{\max}$ that exceeds or matches the cost-of-energy of all nodes in the network. We also assume – contrary to the model described in Section 4 – that there exists a maximum emission energy $P_{\max}^{emit}$. The unit payment made to each node for forwarding a `ROUTE REQUEST` packet that contains at least one new edge-weight is thus defined to be $c_{\max} \cdot P_{\max}^{emit}$. With this payment, it is guaranteed that every node will be able to cover its cost for forwarding the packet. In the alternative version, a source node S starts just as in Section 4 by broadcasting a `ROUTE REQUEST` packet. The forwarding network nodes execute the original algorithm with the following enhancement: whenever a network node v_j receives a `ROUTE REQUEST` packet from a neighbor v_i that contains no new edge-weight information for node v_j, it drops the packet as in the original version, but v_j also checks whether it had received packets from v_j in the past with the same sequence number that together contained the same edge-weight information as the current packet. If this is not the case, it increases a counter N_j^i for neighbor v_i by the number of new edge-weights contained in the packet that v_j has not received from v_i before. At the end of the broadcasting phase, the counter N_j^i contains the number of edges of which node v_i knows the weight and which v_i has forwarded to v_j. The destination node, upon collecting all edge weights, also executes the same steps as in Section 4, but in addition, it computes a minimum spanning tree MST of the resulting graph (it could be any spanning tree). It then communicates MST to the other nodes by sending a message containing the MST along all MST-edges that originate from the destination. The other nodes forward their information along their MST-edges. Once all nodes have received the MST-message, each leaf v_j of the MST-tree sends all counter values N_j^i that it has collected for all its neighbors v_i along the MST to the source node S authenticating this information through a digital signature. All other nodes in the tree wait until they have received the counter information from all their MST-neighbors except the one that lies on the path to the source; they then combine all counter information into a single packet which they forward to their neighbor on the MST-path to the source, also adding their own authenticated counter values. Once the source S has received all counter value information it computes the payment for a node v_i by taking the maximum N_j^i over all j. Finally, the source adds three additional unit payments to each maximum counter value (in order to pay for the three forwarding actions along the MST) and sends out the payments along the MST.

This alternative *route discovery* phase adds considerable overhead to the protocol. However, it guarantees that the nodes will participate in the *route discovery* phase even if they know in advance that they will not be on the shortest path. To see this, note that the nodes only get the money in the very end, thus they are willing to participate upto the last message. Participation in the final forwarding action (i.e., passing money on to the MST-neighbors) is implicitly ensured as the neighbors will start complaining if they do not receive payment. Moreover, forwarding more `ROUTE REQUEST` packets than necessary is discouraged by the fact that the nodes receive payment for every new edge-weight, and finally, forwarding bogus edge weights can be prevented by requiring authentication for each edge weight.