# COMMIT: A Sender-Centric Truthful and Energy-Efficient Routing Protocol for Ad Hoc Networks with Selfish Nodes

Stephan Eidenbenz, Giovanni Resta, Paolo Santi

*Abstract*— **We consider the problem of establishing a route and sending packets between a source/destination pair in ad hoc networks composed of rational selfish nodes, whose purpose is to maximize their own utility. In order to motivate nodes to follow the protocol specification, we use side payments that are made to the forwarding nodes. Our goal is to design a fully distributed algorithm such that: $(i)$ a node is always better off participating in the protocol execution (individual rationality), $(ii)$ a node is always better off behaving according to the protocol specification (truthfulness), $(iii)$ messages are routed along the most energy-efficient path, and $(iv)$ the message complexity is reasonably low. We introduce the COMMIT protocol for individually rational, truthful, and energy-efficient routing in ad hoc networks. To the best of our knowledge, this is the first ad hoc routing protocol with these features. COMMIT is based on the VCG payment scheme, in conjunction with a novel game-theoretic technique to achieve truthfulness for the sender node. By means of simulation, we show that the inevitable economic inefficiency is small. As an aside, our work demonstrates the advantage of using a cross-layer approach to solving problems: leveraging the existence of an underlying topology control protocol, we are able to simplify the design and analysis of our routing protocol, and to reduce its message complexity. On the other hand, our investigation of the routing problem in presence of selfish nodes disclosed a new metric under which topology control protocols can be evaluated: the *cost of cooperation*.**

## I. Introduction

Ad hoc networks are expected to revolutionize wireless communications in the next few years: by complementing more traditional networking paradigms (Internet, cellular networks, satellite communications), they can be considered as the technological counterpart of the concept of "ubiquitous computing". However, in order for this scenario to become reality, several issues raised by ad hoc networking must be adequately addressed. One of these issues, which may be one of the reasons for the lack of commercial applications based on ad hoc networks so far, is how to stimulate cooperation between the network nodes. In fact, the nodes of an ad hoc network are in general owned by different authorities (private users, professionals, companies, and so on), and a voluntary and "unselfish" participation of the nodes in the execution of a certain network-wide task cannot be taken for granted.

One of the fundamental tasks any ad hoc network must perform is routing: since the network is in general multi-hop, a routing protocol is needed in order to discover and maintain routes between far away nodes, allowing them to communicate along multi-hop paths. Unless carefully designed, routing protocols are doomed to perform poorly in presence of "selfish" node behavior: in general, a network node has no interest in forwarding a packet on behalf of another node, since this action would only have the effect of consuming its resources (energy, and available bandwidth). Thus, if many of the nodes act selfishly few multi-hop communications can take place, and the network functionality is compromised.

In order to circumvent this problem, several authors have recently proposed to stimulate cooperation using incentives. These incentives can take the form either of reputation systems (basically, "bad behaving" nodes are detected and isolated from the rest of the network) [4], [5], or of (sometimes virtual) monetary transfer (basically, the sender of a message pays a certain amount of money to the relay nodes to motivate them to forwarding its message) [1], [2], [6], [7], [8], [19].

However, most of the approaches proposed in the literature, such as those presented in [19], are focused on the packet forwarding phase of a routing protocol: the route to the destination is already known, and the goal is to identify strategies that motivate nodes to forward packets along this route. Relatively little attention has been devoted to the problem of stimulating cooperation in the *route discovery phase* of a routing protocol. Clearly, this is a prerequisite for the actual implementation of any of the packet-forwarding schemes introduced in the literature.

To the best of our knowledge, the only paper that addresses this problem is [1], where the authors present the Ad Hoc-VCG routing protocol. This protocol is based on monetary transfer, and has several nice features: it discovers the most energy-efficient path between the source and the destination, and it is *truthful*, i.e., it stimulates the nodes to behave according to the protocol specification[1]. However, Ad Hoc-VCG suffers from two major problems: $(i)$ it assumes that the source cannot act strategically (i.e., the source node follows the protocol specification by assumption), and $(ii)$ the number of messages that must be exchanged in order to find the route to the destination is quite high – in the order of $O(n^3)$, where $n$ is the number of network nodes.

In this paper, we present COMMIT, a protocol for route discovery and packet forwarding in ad hoc networks that enjoys the same nice features as Ad Hoc-VCG (energy-efficiency and truthfulness). Contrary to [1], in our model we

Stephan Eidenbenz is with the Los Alamos National Labs, USA.
Giovanni Resta and Paolo Santi are with IIT-CNR, Pisa, Italy.

[1]This is a very informal definition of truthfulness. A more formal definition of this notion will be given in Section III-B.

allow the sender to act strategically, and we prove that the protocol remains truthful also in this scenario.

COMMIT leverages the existence of an underlying topology control protocol which determines, for every node $v$ in the network, a transmit power level $l(v)$ used to send and forward packets. As we shall see, this assumption simplifies the game theoretical analysis of the protocol, and it reduces the message complexity to $O(|M|^2 d)$, where $|M| \leq n-2$ and $d$ is the maximum node degree in the communication graph. Considering that most topology control algorithms build communication graphs with small degree ($d = O(\log n)$, or even $d = O(1)$ in some cases [3], [18]), this is a significant improvement over the $O(n^3)$ message complexity of Ad Hoc-VCG.

## II. Application scenario and motivation

We consider a wireless network used to access a certain service (e.g., internet access). In principle, ad hoc networking could be used to increase the service coverage: instead of requiring each customer to be directly connected to the base station, customers could be allowed to reach the base station along multi-hop paths, using the wireless devices (laptop, PDA, and so on) of other customers as intermediate nodes.

We remark that the mechanisms described in this paper can be used to establish any type of connection between a service provider and a customer along wireless, multi-hop paths, where the relay nodes are in general other customers. In the following, we will conventionally call the customer who wants to establish a connection to the service the 'sender', the intermediate wireless nodes the 'relays', and the service provider the 'destination' of the communication, regardless of the actual data flow between the sender and the destination. For instance, in case the provided service is internet access, most of the traffic is likely to be downlink (i.e., from the destination to the sender, according to our terminology). Nevertheless, the data session is initiated by the customer with a route discovery (or service discovery) phase, and the customer will pay for both the ingoing and outgoing traffic. For this reason, we have adopted the terminology introduced above.

In order to implement such wireless multi-hop access service successfully, intermediate nodes should be motivated to act "unselfishly", relaying packet on behalf of other nodes.

Since in this scenario the newcomer does not know the route to the access point, incentives must be given also to perform route discovery. So, routing according to the Ad Hoc-VCG protocol seems a reasonable choice. Ad Hoc-VCG is based on the following idea [1]: The sender starts a route discovery process, declaring the destination of its packets. As a result of the route discovery phase, the sender receives a message indicating the path $P$ to the destination (if any), and the cost of sending (or receiving) the packet along $P$. The amount that the sender pays is divided among the nodes on $P$, in such a way that every node receives an amount of money that is at least equal to (actually, it is usually greater than) its real cost for forwarding the packet. In one of the two payment models presented in [1], the sender also pays the premiums (i.e., the amount of money exceeding the actual cost of sending a packet) to the intermediate nodes.

Unfortunately, Ad hoc-VCG is of little help in the application scenario described above. In fact, in Ad hoc-VCG it is assumed that both the sender and the destination of the communication act truthfully. This assumption, and in particular the assumption on the sender's behavior, is quite unrealistic in the application scenario considered. In fact, in this scenario many nodes act as sender and relay node at the same time, and the assumption above implies that a node would behave strategically when forwarding packets on behalf of someone else, but it would become a "good guy" (no strategic behavior) when it sends its own packets.

Another unrealistic aspect of Ad hoc-VCG is the fact that *it is assumed that, after the route discovery phase, the sender actually sends out/receive data packets and pays the amount of money due for sending/receiving the packets*. In other words, once the sender has started the route discovery phase, it cannot withdraw the connection request. This mechanism is fundamental for the correct execution of the routing protocol: if intermediate nodes in the selected path $P$ would not be sure that the payment will actually take place, they would loose their incentive to participate in the route discovery phase. In Ad hoc-VCG, when the sender issues the route discovery message, it has no idea of the amount of money that it will pay. Considering our application scenario, the above assumption would imply that a customer, once issued the request for the service (e.g., internet access), would be forced to pay an amount of money that she does not know in advance. Clearly, nobody would use such a service.

In this paper, we propose a sender-centric approach to the design of incentive compatible routing protocols for ad hoc networks, which results in a protocol called COMMIT. The basic idea is inspired by the business model of the *priceline.com* website [16]. On this website, customers declare the maximum amount of money they are willing to pay for a certain service (e.g., a hotel of a certain category in a certain city). When a customer presents the request, she is required to provide to the system all details for payment before her request is processed. If the system finds a "provider" matching the request (e.g., a hotel with the correct features and a price not exceeding the offered one), then the request is automatically accepted, and the transaction takes place.

We believe a similar approach is suitable to the application scenario described in this section: when a new customer wants to access the service, she issues a "connection request", stating the maximum amount of money she is willing to pay for it. The connection request represents a full commitment[2] of the new customer: if the connection can actually take place at a cost less than the declared price, the newcomer must pay the corresponding amount of money. This way, *the customer has always full control of the maximum amount of money she will spend for sending/receiving the packets*.

In the following we design the COMMIT routing protocol based on this idea, and we show that *it is resilient to strategic sender behavior*, thus overcoming one of the main limitations of Ad hoc-VCG. Further, we prove that COMMIT always chooses the most energy-efficient path between the source and

[2]This is why we called our protocol COMMIT.

the destination, that is truthful, and that it satisfies individual rationality. Energy efficiency is the key design criterion for any routing protocol as transceiver devices always have very limited battery power. With truthful, we mean that the best selfish strategy for every node (excluding the destination which, as in Ad hoc-VCG, is assumed to behave good) is to follow the protocol specification. With individual rationality, we mean that it is rational for the selfish node to participate in the protocol execution. Note that, given the discussion above, *executing Ad hoc-VCG is not individually rational for the sender*.

## III. THE SYSTEM MODEL

### A. Network model

We consider an ad hoc network composed of $n$ nodes. The wireless links between nodes are represented in the *communication graph* $G$. In this paper, we consider only *symmetric* wireless links; i.e., an edge between nodes $v$ and $w$ appears in $G$ if and only if $v$ is within $w$'s transmitting range, and $w$ is within $v$'s transmitting range. Further, we assume that the (symmetric) communication graph $G$ that describes the network topology is 2-connected (with respect to the destination): i.e., there exist at least two node-disjoint paths from any node to the destination node in $G$.

To establish the communication graph, nodes execute a topology control protocol, at the end of which every node $v$ has determined its transmitting range $r_v$. The power required to achieve a transmitting range $r_v$ is generally believed to be proportional to $r_v^\alpha$, where $\alpha$ is a constant between one and six. We remark that $v$ will transmit with range $r_v$ independently of the actual 1-hop neighbor to which the packet is directed. These transmitting ranges imply a directed connection graph (possibly) with non-symmetric links. Since we only consider symmetric links, data will never be transmitted along links that only work in a single direction. Using symmetric links only is a standard assumption in the topology control community [3], [18], since it offers a variety of conveniences such as the fact that sending ACKs is always possible.

The topology control protocol is executed periodically to update $r_v$ but, in the period of time between consecutive topology checks, the same transmitting range $r_v$ is used for any transmission. For the sake of clarity of illustration, we assume that no link failures (due to node mobility) occur during the route discovery phase and the subsequent data session before the topology control protocol executes its next round. Our model of topology updates is realistic for real-life hardware (such as the CISCO Aironet wireless cards [9]) and significantly reduces the message complexity when compared with Ad hoc-VCG [1].

Any topology control strategy can be used in combination with our routing protocol. In particular we have conducted simulation experiments using the following strategies.

a) KNeigh: the node's transmitting range is computed using the KNeigh protocol [3]: every node considers the $k$ closest neighbors, and sets the transmitting range to the value needed to reach the farthest *symmetric* neighbor amongst the $k$ closest nodes.
b) CBTC: the node's transmitting range is computed using the CBTC protocol [18]: every node sets its transmit power to the minimum value such that at least one neighbor is present in any cone of degree $\rho$ centered at the node. The communication graph is then restricted to the symmetric links.
c) CTR: a degenerated topology control mechanism, in which all the nodes have the same trasmitting range, but the value of the common range is carefully chosen to ensure connectivity with high probability [14], [17].

In order to simplify the presentation, in the following we assume that nodes can transmit using different power levels (e.g., 1mW, 5mW, 20mW, 30mW, 50mW and 100mW as in the CISCO Aironet 350 wireless card [9]). At the end of the topology control phase, every node chooses one of the power levels as its transmit power, which is retained until the next topology check. Choosing the power levels from a discrete set of values is not a requirement for COMMIT, but it is much more realistic to do so.

### B. Modeling routing as a game

In this paper, we model the process of establishing a route between a source and a destination node as a game. The players of the game are the network nodes. With respect to a given data session, any node can play only one of the following roles: *source*, *relay (or intermediate) node*, or *destination*. We denote by $S$ the sender, with $v$ (or sometimes $v_i$) an arbitrary relay node, and with $D$ the destination.

Although in principle our approach can be used for establishing a generic connection between arbitrary source/destination pairs, in the remainder of this paper we specialize our protocol to deal with the case in which the destination node is fixed, and provides some service (e.g., internet access) to the other network nodes. In this scenario, it is reasonable to assume that the service provider is a trustworthy third party, which has no interest in cheating. Thus, the destination node in our model is not actually part of the game, but it is rather a "neutral referee", whose goal is to correctly compute the minimum energy $(S, D)$ path, and the payment/premiums for $S$ and the intermediate nodes.

The assumption that the service provider is trustworthy is quite common in the literature on incentive compatibility in ad hoc networks and it is also commonly used in the literature on game theory [15].

We recall that in our model the goal is to establish a path between the sender and the destination, along which traffic packets *in both directions* will be routed (this is always possible since we are assuming that wireless links are bi-directional). The sender will pay for both the packets sent and received during the data session.

The sender $S$ has a private information (its *type*), i.e., its willingness to pay for establishing a connection to the destination. Assuming that $m$ is the maximum per-packet price that $S$ is willing to pay for the connection, we can model the *utility* of player $S$ if the communication takes place as $u_S = m - c_S(D)$, where $c_S(D)$ represents the actual per-packet amount of money that $S$ will pay. In case the connection cannot be established, we have $u_S = 0$.[3]

Let us now consider an arbitrary relay node $v$. In this case, the private type of the node is its power level $l(v)$ which, as described in the previous section, is assumed to be constant during the route discovery and data session phase, but is not known to the other nodes. In general, the cost $c_v$ incurred by node $v$ to relay a packet sent by $S$ is determined by $l(v)$ and by other factors (e.g., the remaining energy in the battery, the bandwidth currently used by the node for its own connections, or any other type of consideration influencing $v$'s willingness to relay $S$'s packet). For the sake of simplicity, in this paper we assume that $c_v = l(v)$. However, our approach remains valid if $c_v$ is an arbitrary function of $l(v)$ and, say, the battery level of node $v$. The utility of node $v$ if it takes part in the data session is $u_v = pay(v) - l(v)$, where $pay(v)$ is the per-packet payment that $v$ receives for relaying $S$'s packets. In case $v$ does not take part in the data session, it gets 0 utility.

In accordance with standard game-theoretic settings (see [15]), we assume that nodes act selfishly and are rational. In other words, we assume that each player in the game plays the strategy that maximizes her utility. As part of the strategy, a node might decide to lie about its type, or to drop/modify messages, and so on. Of course, one of the possible strategies for the nodes is to follow the protocol specification, i.e., declaring the true type and sending/relaying messages as prescribed. Using game theory terminology, we call this strategy *truth-telling*[4].

Our goal is to devise a mechanism such that a globally desirable goal (selecting the most energy-efficient path between source and destination) is achieved or optimized, and nodes participating in the protocol are always better off behaving in accordance with the protocol specification (truth-telling).

Finally, we outline that in this paper we are not concerned with malicious node behavior and with coalition formation. In case of malicious nodes, players are allowed to choose irrational strategies (e.g., strategies leading to negative utility), as long as this is detrimental for the system. In case of coalitional games, players are allowed to coordinate their cheating behavior in order to fool the system. If this coordinated behavior increases the overall utility of the coalition, the surplus can be shared among its participants, which will then have an incentive to deviate from truth-telling. How to extend/modify our protocol in order to take malicious nodes and collusion into account is matter of ongoing research.

## IV. THE COMMIT PROTOCOL

In this section, we describe the COMMIT protocol for incentive compatible and energy-efficient routing in ad hoc networks.

[3]In general, the utility of $S$ if there is no connection is $0 - \bar{c}_S(D)$, where $\bar{c}_S(D)$ is the price paid by $S$ when the connection is not possible. As we shall see, our protocol sets $\bar{c}_S(D) = 0$, so the overall utility of $S$ in case of no connection is 0.

[4]Indeed, in standard (non distributed) game theory, the strategy of a player is simply her declared type. For this reason, the strategy in which the player behaves honestly is called truth-telling. In the distributed context, the player must also participate in the protocol by exchanging messages. By analogy, we call the honest node behavior truth-telling also in this case.

### A. Design guidelines

The design goals of our protocol are: $a)$ individual rationality; $b)$ truthfulness; $c)$ energy efficiency; $d)$ limited message overhead.

A mechanism satisfies the individual rationality property if a node that executes the protocol never gets a negative utility. This property clearly ensures that nodes are motivated to take part in the protocol. *This fundamental property is not satisfied by Ad Hoc-VCG [1], which is the only truthful routing mechanism for ad hoc networks introduced so far.* The motivations for goal $b)$ are clearly described in the previous sections. With energy efficiency, we mean that the communication between $S$ and $D$ (if feasible) will take place along the path of minimum energy cost. The energy cost of a path $P$ is defined as $\sum_{v \in P, v \notin \{S,D\}} l(v)$. Since nodes are battery operated, energy efficiency is a key property in ad hoc networks. Finally, the protocol should minimize the overall number of messages exchanged in the session setup phase.

In order to ensure properties $a) - c)$, our mechanism will use side payments to some of the relay nodes (those in the winning $(S, D)$ path). The mechanism we design must perform the following tasks:

– *winner determination*: determine the winning path (if any) along which the communication will take place.
– *payment computation*: in case the winning path exists, determine the price that $S$ must pay for transmitting/receiving the packets, and the payments for the nodes in the winning path;
– *billing*: if the communication takes place, charge $S$ and pay the nodes in the winning path according to the prices previously determined.

In our protocol, winner determination and payment computation are performed by the destination node $D$, based on the information provided by the network nodes; billing is done when the actual data session begins. Similarly to [1], in this paper we focus on the problem of winner determination and payment computation, leaving the details on how the payments are actually delivered to the nodes unspecified. Indeed, the problem of implementing electronic payments in ad hoc networks is a research thread in itself, which is addressed, for instance, in [8], [19]. In principle, any of the electronic payments methods presented in the literature can be used in combination with our routing protocol.

Technically, COMMIT implements a distributed reverse second-price single item auction with reserve price. The auctioneer is the sender $S$, which wishes to buy an item (a path to the destination $D$) at a maximum price of $m$ (the reserve price). On the other side, there is a set of sellers (the relay nodes). The reverse auction is second-price, since the price paid by the sender to the winning group (the nodes in the most energy-efficient path) depends on the price of the second best offer. As we will see later, the choice of implementing a second price auction is dictated by the truthfulness requirement.

### B. The pricing scheme

Before presenting the protocol specification, we describe the pricing scheme used by COMMIT, since the choice of the pric-

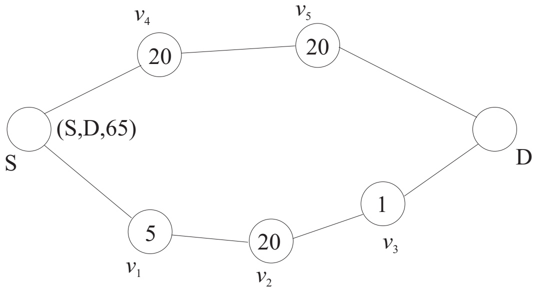


Fig. 1. Example of cheating node behavior if $c_S(D)$ would be defined as $c_S(D) = \sum_{v \in MP, v \neq \{S,D\}} pay(v)$. The nodes are labeled with their true types.

ing scheme determines the minimum amount of information which must be communicated to the destination node (which is in charge of computing the payments).

In [12], it is shown that any pricing scheme that achieves individual rationality, truthfulness, energy efficiency, and pays only the nodes in the winning path must be based on the VCG mechanism[5]. When adapted to our setting, the VCG mechanism [15], which optimizes the socially desirable goal of energy efficiency, defines the following rules to determine the winning path and the relative payments. Let $c(P)$ denote the energy cost of an arbitrary $(S, D)$-path $P$ (i.e., a path from $S$ to $D$),where $c(P) = \sum_{v \in P, v \notin \{S,D\}} l(v)$. The winning path is the path of minimum energy cost, denoted by $MP$. For any node $v$ in the winning path, let us denote with $c(P^{-v})$ the cost of the minimum energy $(S, D)$-path $P^{-v}$ that does not include $v$. Since we are assuming that the communication graph is 2-connected, this alternative path, which we call the *replacement path*, always exists. The payment for a node $v$ in the winning path $MP$ is defined as follows:

$$pay(v) = c(P^{-v}) - c(MP) + l(v) \ .$$

The payments for the nodes which are not on the winning path are set to 0.

A key novel feature of COMMIT (novel even in the broader context of distributed mechanism design and game theory) lies in the definition of the price $c_S(D)$ that $S$ must pay for sending the packets along $MP$ and in the subsequent definition of who makes which payments. Price $c_S(D)$ defines the *decision rule*, which determines whether the communication takes place or not. A trivial choice would be to set $c_S(D) = \sum_{v \in MP, v \notin \{S,D\}} pay(v)$. However, due to the presence of the reserve price $m$, this choice would leave space for a strategic behavior of the nodes in $MP$, that could declare a false type in order to drive $c_S(D)$ below $m$.[6]

This subtle example of strategic node behavior is depicted in Figure 1. The sender wants to establish a connection to the destination paying at most 65 for each packet. If all the nodes behaved truthfully, the communication would not take place. In fact, we have $MP = \{v_1, v_2, v_3\}$, $c(MP) = 26$, and $c(P^{-v_1}) = c(P^{-v_2}) = c(P^{-v_3}) = 40$, which implies the following payments for the nodes in $MP$:

$$pay(v_1) = 40 - 26 + 5 = 19, \ pay(v_2) = 40 - 26 + 20 = 34,$$

$$pay(v_3) = 40 - 26 + 1 = 15 \ .$$

It follows that the total payment is $68 > 65$, and the communication does not take place, yielding a 0 utility for all the players. Let us now assume that node $v_2$ falsely declares power level 30. The winning path $MP$ would remain the same, as well as the replacement path for all the nodes in $MP$. However, the payments would change as follows:

$$pay(v_1) = 40 - 36 + 5 = 9, \ pay(v_2) = 40 - 36 + 30 = 34$$

$$pay(v_3) = 40 - 36 + 1 = 5 \ .$$

Thus, the total payment is now $48 < 65$, and the communication would take place, yielding an utility of $34 - 20 = 14$ for node $v_2$. Since $v_2$ would increase its utility by reporting a false type, it follows that defining $c_S(D)$ as $\sum_{v \in MP, v \notin \{S,D\}} pay(v)$ would result in a non-truthful mechanism.

In order to circumvent this problem, we set $c_S(D) = c(P^{-MP})$, where $c(P^{-MP})$ denotes the cost of the minimum energy path that does not contain *any* of the nodes in $MP$. We call this path the *global replacement path*. It is immediate to see that with this definition of $c_S(D)$ any false declaration of the nodes in $MP$ would have no effect on $c_S(D)$. Thus, the truthfulness of the mechanism is not impaired.

Observe that the assumption of 2-connected communication graph does not imply that a global replacement path always exists. Indeed, this is a stronger property, since we require that one of the at least two node-disjoint paths that exist between $S$ and $D$ (because of 2-connectivity) is the minimum-energy path $MP$. We call this property *minimum-energy 2-connectivity*. To make the distinction between 2-connectivity and minimum-energy 2-connectivity clearer, consider the graph in Figure 1, and suppose there exists an extra edge between units $v_3$ and $v_4$. From the point of view of nodes $S$ and $D$, the graph is 2-connected; however, if it happens that $MP = \{S, v_4, v_3, D\}$ is the minimum-energy path, then the graph is not minimum-energy 2-connected, since removing $v_3$ and $v_4$ from the graph would make it disconnected.

We have conducted simulation experiments to determine whether the communication graphs produced by the topology control protocols listed in Section III-B satisfy minimum-energy 2-connectivity on the average. The experimental results, which are not reported due to lack of space, show that global replacement paths exist with high probability: for CBTC, this probability always exceeded 99%, for other topology control protocols it was always above 92% and quickly increases to more than 99% as we increase the number of nodes.

Given the pricing scheme, we can define the winning path $MP$ as *feasible* if $c_S(D) < m$. If this condition does not

[5]Although this result is proved with reference to a routing problem on the Internet, it can be easily adapted to our scenario.

[6]The reader could question whether an *explicit* reserve price (an implicit reserve mechanism is needed to ensure individual rationality of the sender) is needed at all. An implicit reserve mechanism could be implemented, for instance, by having the sender aborting the connection if the requested price is too high. However, this solution would require exchanging several (useless) control messages, resulting in a waste of resources. Our solution of having an explicit reserve price ensures that a minimal amount of control messages is exchanged to establish the connection (see also Section IV-D).

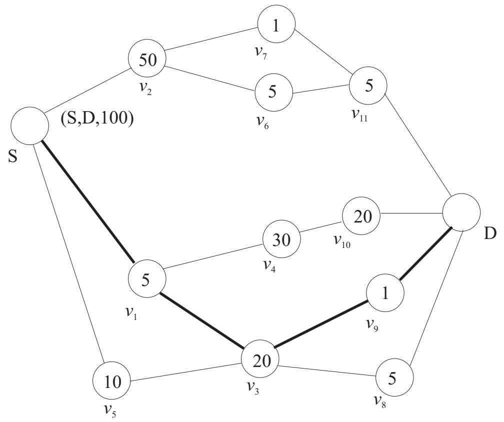


Fig. 2. Example of network topology. Intermediate nodes are labeled with the corresponding power level (type). The sender offers a price of 100 for establishing a connection to the destination. The communication will take place along the minimum-energy path (bold edges).

hold, the communication cannot take place, since the sender would be forced to pay an amount of money that exceeds $m$, violating the condition of individual rationality.

Note that in general we have $c(P^{-MP}) \neq \sum_{v\in MP, v\notin\{S,D\}} pay(v)$, i.e., the amount that the sender pays and the total amount that the intermediate nodes receive are not equal. In this case we say that budget is imbalanced.[7] In our protocol, we assume that the destination node $D$ is in charge of balancing the budget, getting the additional money if $c(P^{-MP}) > \sum_{v\in MP, v\neq S} pay(v)$, or contributing to the payments if $c(P^{-MP}) < \sum_{v\in MP, v\notin\{S,D\}} pay(v)$. This assumption is coherent with our reference scenario, in which $D$ is the service provider. Since the service provider is involved in many sessions, it is possible that its overall balance is close to 0. Even if this is not the case (for instance, because $c(P^{-MP}) < \sum_{v\in MP, v\notin\{S,D\}} pay(v)$ most of the time), the service provider can modify the price of the fixed (e.g., per-connection, or monthly) fee that the customers must pay to access the service in order to not reduce its revenue.

Let us clarify our pricing scheme with the example in Figure 2. The sender wants to establish a connection with the destination, and is willing to pay at most 100 for it. For the moment, let us assume that the information regarding the network topology and nodes' types is known to the destination. $D$ computes the winning (minimum-energy) path $MP$, the replacement paths for all nodes on $MP$ and the global replacement path $P^{-MP}$:

$$\begin{aligned} MP &= \{v_1, v_3, v_9\} & c(MP) &= 26 \\ P^{-v_1} &= \{v_5, v_3, v_9\} & c(P^{-v_1}) &= 31 \\ P^{-v_3} &= \{v_1, v_4, v_{10}\} & c(P^{-v_3}) &= 55 \\ P^{-v_9} &= \{v_1, v_3, v_8\} & c(P^{-v_9}) &= 30 \\ P^{-MP} &= \{v_2, v_7, v_1 1\} & c(P^{-MP}) &= 56\ . \end{aligned}$$

[7]The VCG mechanism is known to have imbalanced budgets [15], and in fact under reasonable assumptions no mechanism can achieve budget balance, energy-efficiency, and truthfulness simultaneously.

The price that $S$ should pay is $c(P^{-MP}) = 56 < 100$, so MP is feasible.

The payments for the nodes in the winning path are computed as follows:

$$pay(v_1) = c(P^{-v_1}) - c(MP) + l(v_1) = 31 - 26 + 5 = 10$$

$$pay(v_3) = 55 - 26 + 20 = 49\ , \quad pay(v_9) = 30 - 26 + 1 = 5\ .$$

The total payments amount to 64. Since $S$ will pay only 56, the remaining 8 units of money are paid by $D$. Note that, if the type of node $v_{11}$ is 20 instead of 5, we have $c(P^{-MP}) = 71$, with all the other costs unchanged. In this situation, $S$ would pay 71 (which is still below 100), and the 7 units of money remaining would be retained by $D$.

In order to evaluate the impact of the different topology control strategies on budged balance, we performed a set of simulations, which showed that on the average the destination node must contribute some money to support the communication. The relative amount of this contribution, which strongly depends on the number of nodes and on the topology of the communication graph, can be confined to less than 6% of the total cost: with the CTR topology control protocol, the percentage decreases with an increasing number of nodes after reaching a negative peak of about 6% at about 200 nodes.

Thus, using the global replacement path to define the payment that the sender needs to make is a novel idea in distributed game theory that, combined with a suitable topology control protocol, turns out to almost balance the budget on the average.

### C. Protocol specification

COMMIT consists of two phases:

– Route discovery:
  The communication graph is computed by a limited flooding process, the winning path $MP$ and payments are computed by destination $D$ and communicated to $S$.
– Data transmission (only if $MP$ is feasible):
  Data packets and payments are sent along the winning path $MP$ from source $S$ to destination $D$ (or vice-versa) until the sender terminates the connection or until the topology control protocol updates the topology.

In the route discovery phase, $S$ sends (using power $l(S)$) a route discovery message $RD(S, D, m)$, indicating that it wants to start a data transmission session with node $D$, and that it wishes to pay at most $m$ for this service for each data packet that is sent in this session.

In the route discovery phase, an intermediate node $v_k$ receives messages of the form

$$RD(S, D, m, v_1, l(v_1), \ldots, v_{k-1}, l(v_{k-1}))$$

where path $v_1, \ldots v_{k-1}$ indicates a path from sender $S$ to node $v_{k-1}$. The amount of money that is left once $v_k$ receives the message is the original offer by $S$ minus all costs along the path, i.e., $m - \sum_{i=1}^{k-1} l(v_i)$. Node $v_k$ builds up its own local view of the communication graph by receiving messages: whenever it receives a path containing information about the existence of an edge that it does not yet know, it adds this information to its local view. Node $v_k$ then appends to the message that contains new information the fields $v_k, l(v_k)$, and forwards it with power $l(v_k)$. In order to prevent other nodes from altering the fields $v_k, l(v_k)$, these fields are cryptographically signed by node $v_k$. Moreover, $v_k$ signs the field $v_{k-1}$ to acknowledge that an edge between $v_{k-1}$ and $v_k$ exists.

This flooding process is repeated until the route discovery message arrives at $D$, which does not forward messages, but other than that it acts just like a regular intermediate node: it collects the $RD$ messages arriving from different nodes, and builds up a complete view of the communication graph. Once $D$ has received all information, it computes the minimum energy path $MP = \{S, v_1, \ldots, v_k, D\}$ from $S$ to $D$, the replacement paths $P^{-v_i}$ for each intermediate node $v_i$ on the minimum energy path $MP$, and the global replacement path $P^{-MP}$. Given this, $D$ determines whether $MP$ is feasible (i.e., if $c_S(D) = c(P^{-MP}) < m$) and, in case the answer is positive, it computes the payment/premiums for $S$ and the nodes in $MP$. It then sends back this information (winning path, payments, and the global replacement path) to $S$ using the reverse of path $MP$. In order to avoid that intermediate nodes manipulate the payments, we assume that this message is cryptographically signed by $D$. Then $S$ sends a test packet along the global replacement path in order to verify that this path actually exists, asking each node $v$ in $P^{-MP}$ to sign that the two neighbors of $v$ on $P^{-MP}$ are actually neighbors. $D$ receives the signed test packet, checks all signatures, and then sends a packet along the reverse $MP$ path to $S$ to indicate that it can start the data transmission phase.

After the route discovery phase, the *data transmission* phase takes place, in which $S$ sends/receives its data packets to/from $D$ via the computed minimum energy path. With each packet, it includes an electronic payment that is due to the intermediate nodes. The nodes on $MP$ forward the data packet and collect the payments. The data transmission phase ends when $S$ has transmitted its last packet or when the topology control protocol changes the network topology in order to account for node mobility. The latter case forces the sender to initiate a new route discovery phase.

### D. Optimizations

The route discovery phase of COMMIT as described above leaves room for improvement.

A first optimization is the following. Instead of forwarding whole paths every time a new path is received, the nodes could forward only new edges that it has learned of and that give rise to new paths. This reduces the message complexity of the route discovery phase.

The second optimization is somewhat more involved. An intermediate node $v_k$ can compute whether a newly received path is feasible in the sense that it has a non-negative amount of money left at $v_k$. If the path is not feasible, there is no point in forwarding it because communication will not take place even if this path is either the minimum-energy path, or a replacement path for a node on $MP$, or the global replacement path. Thus, node $v_k$ has no economic incentive to propagate the route request, and will simply drop it. Note that this "selfish" behavior of $v_k$ turns out to be beneficial for the whole network, since the dropped message was useless. In other words, with this optimization only $RD$ messages referring to paths that have some chance to win the auction, or that are needed to compute the payments, will circulate in the network, eventually reaching the destination node $D$.

If the first optimization measure is implemented, node $v_k$ still adds the new information from the path into its local view of the communication graph and forwards this information as soon as it receives information regarding an edge that renders the path feasible.

## V. PROTOCOL ANALYSIS

We summarize the main properties of the procotol in the following three theorems, whose proofs, here omitted for space reasons, are reported in [11].

*Theorem 1:* If all nodes act truthfully, COMMIT computes the most energy-efficient route from the sender $S$ to the destination $D$.

*Theorem 2:* If the COMMIT protocol is executed in an ad hoc network to route messages, behaving truthfully is a dominant strategy and individually rational for all nodes (except for the destination).

*Theorem 3:* Assume that we implement COMMIT with both optimization options, i.e., only edges are forwarded and paths was cost exceeds $m$ are thrown away. Let $M$ be the subset of all relay nodes in the communication graph such that their minimum energy path to the sender has cost lower than $m$. Let $d$ denote the maximum node degree in the communication graph. Then the total message complexity is $O(|M|^2 d)$.

## VI. THE COST OF COOPERATION

In our protocol, as well as in Ad hoc-VCG, the payment for establishing the communication exceeds the actual cost of the minimum-energy path. This is due to the fact that, in order to motivate the intermediate nodes to cooperate, they must be given some premiums. The difference between the overall amount of these premiums and the cost of the minimum-energy path can be interpreted as the *cost of cooperation*.

The cost of cooperation is a measure of the economic inefficiency induced by the need of stimulating selfish nodes to act unselfishly. This inefficiency occurs when the minimum-energy path has a cost below the offered price $m$, but $c(P^{-MP}) > m$, causing the communication to be aborted.

From the protocol designer's point of view, the cost of cooperation should be as low as possible (note that, on the contrary, from the intermediate nodes' point of view this cost should be as high as possible). Unfortunately, unless some a-priori (probabilistic) information on the player's types is known to the destination, the VCG mechanism (which is the cause of the economic inefficiency) is essentially the only pricing scheme that achieves truthfulness, individual rationality, and routing along the minimum-energy path [12], [15].

In case of COMMIT, the cost of cooperation depends on the distribution of the energy cost of the paths connecting to $D$: if all these paths have approximately the same cost, then the cost of coordination is relatively low; otherwise, it can be quite high.

However, differently from the case of Ad hoc-VCG, in our approach we have a way to reduce (to a certain extent) the cost of cooperation: *changing the topology of the network*. In other words, the network designer could use the underlying topology control protocol to build communication graphs with the desired feature (many paths with approximately the same energy cost), thus reducing the average cost of cooperation. The fact that topology control has a strong influence on the economic efficiency of COMMIT is supported by simulation results: by changing the topology control protocol used the average budget imbalance can be reduced by approximately 15%. We believe this observation is quite interesting, since it discloses a new metric (besides traditional metrics such as connectivity, node degree, etc.) that can be used to evaluate the performance of topology control algorithms.

Since the cost of cooperation might be quite high, a natural question to ask is the following: are side payments (or other forms of incentives) really necessary to stimulate cooperation in ad hoc networks? The results presented in a recent paper [13] seem to give a positive answer to this question. In [13], Felegyhazi et al. study the Nash Equilibria (see [15] for a definition of Nash Equilibrium) of packet forwarding strategies for ad hoc networks. Essentially, the authors discover that the conditions under which cooperation between nodes need not to be enforced (because it is a Nash Equilibrium) are very unlikely to occur in real ad hoc networks. Then, the authors of [13] conclude that, in practice, *an incentive mechanism is needed to stimulate cooperation between nodes*.

## VII. Conclusion

In this paper, we have introduced the COMMIT protocol for individually rational, truthful, and energy efficient routing in ad hoc networks. Besides presenting and analyzing our protocol, we have discussed several issues related to cooperation in ad hoc networks. In particular, we have identified a quantity that can be considered the intrinsic cost of cooperation, and pointed out that topology control can be used to curb this cost.

We hope that the results and discussions presented in this paper will stimulate further research in the field. In this paper we have assumed that the nodes execute the topology protocol truthfully; while we believe this assumption is reasonable in reality, we are sure that the design of truthful topology control protocols that can be combined with COMMIT is necessary. The NE analysis of a topology control game presented in [10] can be a good starting point in this direction.

## References


[1] L. Anderegg, S. Eidenbenz, "Ad hoc-VCG: A Truthful and Cost-Efficient Routing Protocol for Mobile Ad hoc Networks with Selfish Agents", *Proc. ACM Mobicom*, pp. 245–259, 2003.

[2] N. Ben Salem, L. Buttyan, J.P. Hubaux, M. Jakobsson, "A Charging and Rewarding Scheme for Packet Forwarding in Multi-hop Cellular Networks", *in Proc. ACM MobiHoc 03*, pp. 13–24, 2003.

[3] D. M. Blough, M. Leoncini, G. Resta, and P. Santi, "The $k$-Neigh Protocol for Symmetric Topology Control in Ad Hoc Networks", *in Proc. ACM MobiHoc 03*, pp. 141–152, June 2003.

[4] 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.

[5] 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.

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

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

[8] 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.

[9] *Cisco Aironet 350 data sheets*, available at `http://www.cisco.com/en/US/products/hw//wireless`.

[10] S. Eidenbenz, V.S. Kumar, S. Zust, "Equilibria in Topology Control Games for Ad Hoc Networks", *Proc. ACM DIALM-POMC 2003*, pp. 2–11, 2003.

[11] S. Eidenbenz, P. Santi, "COMMIT: A Sender-Centric Truthful and Energy-Efficient Routing Protocol for Ad Hoc Networks", Tech. Rep. IIT-TR01/2004, Istituto di Informatica e Telematica, Pisa - Italy, Jan. 2004.

[12] J. Feigenbaum, C. Papadimitriou, R. Sami, S. Shenker, "A BGP-based Mechanism for Lowest-Cost Routing", *Proc. ACM PODC*, pp. 173–182, 2002.

[13] M. Felegyhazi, L. Buttyan, J.P. Hubaux, "Equilibrium Analysis of Packet Forwarding Strategies in Wireless Ad Hoc Networks – the Static Case", *Proc. Personal Wireless Communications (PWC '03)*, Venice, Italy, September 2003.

[14] P. Gupta, P.R. Kumar, "Critical Power for Asymptotic Connectivity in Wireless Networks", *Stochastic Analysis, Control,Optimization and Applications*, Birkhauser, Boston, pp.547-566,1998.

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

[16] `http://www.priceline.com`

[17] P. Santi, D.M. Blough, "The Critical Transmitting Range for Connectivity in Sparse Wireless Ad Hoc Networks", *IEEE Transactions on Mobile Computing*, vol. 2, no. 1, pp. 25–39, January-March 2003.

[18] R. Wattenhofer, L. Li, P. Bahl, Y. Wang, "Distributed Topology Control for Power Efficient Operation in Multihop Wireless Ad Hoc Networks", *Proc. IEEE Infocom 2001*, pp. 1388-1397, 2001.

[19] S. Zhong, Yang Richard Yang, J. Chen, "Sprite: A Simple, Cheat-Proof, Credit-Based System for Mobile Ad-hoc Networks", *Proc. IEEE Infocom 2003*, pp. 1987–1997, 2003.