A Forwarding Cooperation Protocol for Plain and Cluster-based Ad Hoc Networks

Helena Rifà-Pous and Jordi Herrera-Joancomartí

***Abstract*—In ad hoc networks, due to the lack of a dedicated network infrastructure, members have to collaborate ones with the others to support the basic networking functions that allow them to communicate. The main challenge of this model is combating the intrinsic selfish behavior of the participants, which are usually equipped with handheld and mobile devices with limited resources. In this paper, a forwarding protocol is presented that stimulates the cooperation through a mechanism that combines both credit and reputation-based solutions. A micropayment protocol is used to charge and reward the applier and forwarders of a transmission respectively. The credits obtained for collaboration not only are a mean to pay for network services, but are a symbol of the cooperative range of a node. Using this information, the presented model benefits most cooperative nodes with preferential transmission channels and a higher quality of service. The model is suited for plain and cluster-based ad hoc networks.**

***Index Terms*—Multihop ad hoc networks, cooperation, forwarding, payment, clusters.**

I. Introduction

The functioning of an ad hoc network is based on the supportive contributions of all of its members. Nodes cooperate to form a communication infrastructure that extends the wireless transmission range of every terminal without using any dedicated network device. To ensure and spur the cooperative behavior of ad hoc network members, an incentive mechanism is required that regulates the resources spent and given to the community.

Protocols to stimulate cooperation can be divided in two groups: reputation-based and credit-based. The former treat packet forwarding as an obligation and isolate and punish those nodes that do not behave as expected, while the latter consider it as a service that can be valued and charged. For a detailed comparison of different cooperative protocols we refer to [2] where the most relevant proposals are summarized.

Reputation-based schemes define a method for keeping track of nodes' actions in order to classify reliable and unreliable nodes [3]–[6]. The main problem of this approach is distinguishing misbehaving nodes from those that can not retransmit packets due to energy constraints, channel fadings or simply natural disconnections. The assumption that a node shall forward always all the packets it receives is too hard for a network formed of -beyond others- small and handheld devices. On the other hand, nodes on some strategic points of the network will have more transmission requests than those on the periphery, and it will be unfair to punish them if they can not hold all the transport.

In credit-based schemes, virtual currency is introduced to stimulate each node to behave cooperatively. Nodes that generate traffic have to pay to those ones that help forwarding the data. In this category, a distinction can be done regarding the nature of the payment: money-based schemes and token-based schemes.

Money-based schemes [7]–[9] use money as the payment token. The drawbacks of that kind of currency models is that the costs of managing financial information have a considerable legal and administrative overhead. Furthermore, the minimization of selfish nodes is not guaranteed since users without economical concerns can behave selfishly in the net and pay whatever is needed to have its packets transmitted.

Token-based schemes generally require the nodes have a balanced number of packets transmitted and relayed [10], [11]. Nodes increase the number of stored tokens when they forward packets, and decrease them proportionally to the number of hops when sending messages. A node shall forward packets until it earns enough to send its owns, so this kind of protocols can be sometimes limiting the capacity of the network if the average token level is too low. On the other hand, if it is too high, tokens no longer suppose an incentive to cooperate and the mechanism does not fulfill its purposes any more.

Present research in credit-based mechanisms is basically focused on how much a node should be paid for forwarding messages. One research direction is finding a fair incentive algorithm that rewards the nodes for the resources used in the forwarding connection [11]–[13]. The circumstances and resources employed by relying parties (battery level, transmission energy, position within the network topology, mobility, bandwidth, ..) are considered to calculate the cost of a certain path. Although theoretically these kind of algorithms are very attractive, they are too complex for mobile ad hoc networks.

Manuscript received December, 2007. This research was partially supported by the Spanish Ministry of Science and Education under grant TSI2007-65406-C03-03 E-AEGIS and CONSOLIDER CSD2007-00004 ARES. A preliminary version of this work was presented at the International Conference on Next Generation Teletraffic and Wired/Wireless Advanced Networking (NEW2AN) 2007 [1].

H. Rifà-Pous is with the Computer Science and Multimedia Studies, Universistat Oberta de Catalunya, Barcelona, Spain, (e-mail: {hrifa}@uoc.edu).

J. Herrera-Joancomartí is with the Computer Science and Multimedia Studies, Universitat Oberta de Catalunya, and with the Information and Communications Engineering Department, Universitat Autònoma de Barcelona, Barcelona, Spain, (e-mail: {jherrera}@deic.uab.cat).

The real cost of a transmission changes for every transferred packet so the overhead involved for sending a message is barely affordable. Too hard protocols may provoke a contrary effect on the nodes, not willing to participate on the network.

In this paper, we present a **F**orwarding Sp**u**rring P**r**otocol for Mult**i**hop Ad Hoc N**e**twork**s** (FURIES), a simple credit-based scheme that provides incentives to selfish mobile nodes to cooperate. The proposed protocol seeks to foster the traffic through a fair protocol, but instead of trying to pay for the resources spent in a connection, it rewards with a high quality of service those constant collaborative nodes. The protocol fits in plain and hierarchical topologies. A model for spurring the cooperation and improve the data delivery performance in ad hoc networks is described. An evaluation of the system through a simulation analysis is also presented.

The contributions of the proposal are the following. In spite of previous approaches, that try to spur the system though a payment model that rewards the nodes based on its utility function, FURIES uses a payment protocol to categorize nodes' behavior. Nodes are prone to collaborate in order to obtain a better quality of service. One of the novelties of this protocol with respect to the previous ones proposed in the literature is that introduces an incentive factor to prize the forwarding of packets of high ranked people. Moreover, an efficient adaptation of this cooperative encouraging mechanism for cluster-based ad hoc networks is presented.

The rest of the paper is organized as follows. In section II we introduce the protocol and give an overview of the proposed architecture. Section III describes the protocol details and analyzes some interesting aspects to spur traffic in multihop networks. Section IV presents a forwarding model for cluster-based ad hoc networks and incorporates the FURIES protocol in these architectures. Section V evaluates the solution based on simulation results. Finally, we conclude the paper in section VI.

II. FURIES General Description

We present in this section the general description of our **F**orwarding Sp**u**rring P**r**otocol for Mult**i**hop Ad Hoc N**e**twork**s** (FURIES). FURIES is a credit-based protocol that combines properties of both credit-based and reputation-based incentive models. On one hand, it uses payment mechanisms to charge/reward the forwarding of packets through the net. On the other hand, it manages user reputation status to classify reliable from unreliable nodes. Packets of both high and low reputed nodes are prone to be sent, however nodes with higher reputation take preference to get their data forwarded, i.e. they favor of a better quality of service.

The interchange currency used in the FURIES payment protocol is not money but credit to transmit data. The unit of credit is a token that represents 1 packet of 2346 bytes [1]. Credit tokens exchanged in a transmission session are used to state the reputation of a user and categorize its involvement in the net. Nodes that generate traffic loose tokens and reputation, while the ones that forward it, gain them. However, payments and collections are not balanced. The cost of sending a packet depends on the hop distance to the destiny. On the other hand, the reward is based on the credit level of the sender, that is, its participation status. Thus, nodes earn more credit for forwarding packets of highly reputed and credited nodes.

A. FURIES Entities

An ad hoc network can be represented by an undirected graph $G = (V, E)$, where $V = \{v_1, v_2, ..., v_N\}$ is the set of vertices of the graph, formed by the nodes in the network, and $E = \{e_1, e_2, ..., e_M\}$ is the set of edges which correspond to the communication links between the nodes. Two nodes v_i and v_j are termed *neighbors* if there is an edge $e_l = (v_i, v_j)$ connecting them in the graph.

In this paper we consider a node v_i that wants to connect to another one who is not in his transmission range, so a multihop route has to be established. We assume a routing protocol that provides information of available routes. Opposed to other credit-based protocols for ad hoc networks, FURIES does not require that the source node knows the complete path to the destination but only the hop distance. FURIES will stimulate the transmission through the discovered routing paths.

Credit-based schemes require the use of tamper-proof hardware or a trusted third party (TTP) to manage the tokens. We make use of a TTP to securely store the credit account of nodes and give memory to the system, that is, credit tokens earned or spent in a session are taken in consideration further the lifetime of a particular ad hoc network.

FURIES architecture is composed of the following entities:

- Certification Authorities (CA) that issue identity certificates for the participants of ad hoc networks. The recognized CAs are the ones accepted in the Internet Community and that follow some established security policies.
- Reputation Authority (RpA), a TTP that is used to manage the users' credit account. Such information is contained in a reputation certificate that will be implemented as an attribute certificate according the standard X.509.

All users in our model are registered in a well known CA that issues them a certificate which binds their identity with their public key. With this certificate, users can sign on the RpA that will manage their credit. The RpA is an independent entity not related to any specific CA. It can deal with CAs of different providers as long as it accepts its certification policies. Moreover, the RpA does not need to be centrally controlled but can be a distributed entity under the control of a world-wide community [15], [16].

Reputation certificates are used to classify users and fix the rewarding tokens of a forwarding. For this reason it is important that these certificates hold updated information at any time. Therefore, reputation certificates are short live certificates, with a validity that we fix in 10 days. It is assumed

[1] This value is the maximum size of an IP packet over a 802.11 [14]

that users that enter an ad hoc network have online connectivity with the RpA at most 10 days before, and they have had the opportunity to renew its reputation certificate.

B. Incentive Factor IF

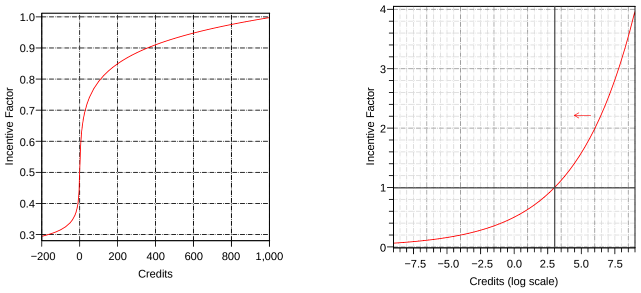
The FURIES protocol introduces an Incentive Factor (IF) element to prioritize the forwarding of packets from collaborative nodes and thus provide them a good quality of service. Nodes do not need to pay more to receive a better service, the incentives a router receive for forwarding a packet are intrinsically stated in the protocol based on the profile of each payer.

The incentive factor modulates the credit (c_{v_i}) that an intermediate node has to receive for its job such that $c_{v_i} = IF_{v_0} \cdot d$, where IF_{v_0} is the incentive factor of the sender node v_0, and d is the number of transmitted packets. We have designed the incentive factor of a node as a function of the credit it holds such that it asymptotically tends to 0 when its credit balance grows in negative values, and increases polynomially otherwise. Since the amount of data transmitted in ad hoc networks can range from a few Kb when the devices are very small and limited, up to hundreds of Mb when the net has access to the Internet, the gradient of the incentive factor function is bigger for values around 0 (see figure 1(a)). This allows to clearly make a distinction between selfish and unselfish nodes. The IF function on the credit is the following:

$$IF_{v_i}(c_{v_i}) = A \cdot abs(c_{v_i})^{(signum(c_{v_i})/B)}$$

Through simulations we have heuristically approximated two values for A and B, resulting in $A = 1/2$, and $B = 10$ (see figure 1(b)).

$$IF_{v_i}(c_{v_i}) = 1/2 \cdot abs(c_{v_i})^{(signum(c_{v_i})/10)}, -10^9 < c_{v_i} < 10^9 \tag{1}$$



(a) IF for low range credit

(b) IF function (log axis)

Fig. 1. Incentive factor function

The charges and rewards of a transmission are not balanced, so we have limited the range of the accumulated tokens to $[-10^9, 10^9]$ in order to avoid the saturation of a node in an extreme position. When the credit rate of a node is 0, its incentive factor is $A = 1/2$, which is lower than 1. This discourages nodes from indiscriminately registering themselves with a new identity to reset their record. The neutral incentive factor ($IF = 1$), that is, when a forwarder receives the same amount of tokens for a carried packet that the ones it would have to pay in case it initiates a transaction, is when the accumulated credit of a node is 10^3 packets which is a little more than $2,2MBytes$ of data.

III. FURIES Credit-based Protocol

FURIES stimulates cooperation through a credit mechanism that regulates nodes' transmissions based on their reputation. In this section we detail such mechanism, that can be divided in three phases:

- Initialization phase
- Contract establishment and communication, driven by a micropayment scheme
- Charging and Rewarding phase

A. Initialization

In order to initiate a transmission in a multihop network a node needs to hold a reputation certificate that states its forwarding parameters. In particular, the reputation certificate sets two main attributes:

- **Credit (c)**: Accumulated credit tokens of a node at the time of certificate generation.
- **Incentive Factor (IF)**: The result of applying the IF function (equation 1) over c.

When a node v_i first requests a certificate in an RpA it is issued a certificate with $c_{v_i} = 0$ and $IF_{v_i} = 1$. Its IF will be 1 until the node starts transmitting data or its accumulated credit is equivalent to an incentive factor greater that 1. We give new nodes an IF of 1 to not prejudice their first transactions. At the same time, we spur nodes first to give resources to the net and then take the profit.

B. Micropayment Scheme

The micropayment scheme we use in this paper is highly inspired on PayWord [17], a light protocol that allows offline verification of the payment proofs. The micropayment protocol is divided in two parts: Contract Establishment and Data Transmission. Figure 2 depicts all the steps.

Contract Establishment

When node v_0 wants to send data to node v_n, assuming the path will go through nodes $v_1, \cdots, v_{n-1}$:

1) v_0 generates payment tokens in the following way: Node v_0 generates a long fresh chain of paywords $w_0, w_1, ..., w_m$ by choosing w_0 at random and by applying a hash function h iteratively such that $w_j = h(w_{j-1})$ for $j = 1, 2, \cdots, m$, where m is the maximum number of possible payments during the session.
2) v_0 prepares a contract offer. The offer includes the sender and receiver identifiers, I_{v_0}, I_{v_n}, the serial number of the sender reputation certificate, SN_{v_0}, and its validity

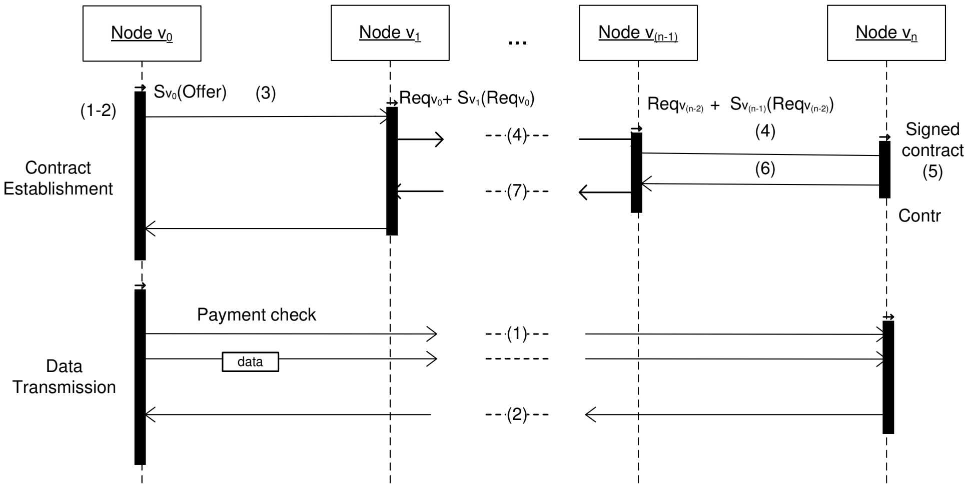


Fig. 2. Micropayment protocol

period V_{v_0}, the number of hops of the route n and the top hash chain value w_n:

$$\text{Offer} = \{I_{v_0}, I_{v_n}, SN_{v_0}, V_{v_0}, n, w_m\}$$

3) Node v_0 sends a forwarding request toward v_n that contains the contract offer and its digital signature of it, together with its reputation certificate $RCert_{v_0}$:

$$Req_{v_0} = \{\text{Offer}, S_{v_0}[\text{Offer}], RCert_{v_0}\}$$

4) The request is read by intermediate nodes of the path $(v_1, \cdots, v_{n-1})$. If they are not interested in forwarding the packet because for them the expense is not worthwhile, they send a reject response to v_0. Otherwise, they enclose in the request a signed attachment with information about its identity.

$$Req_{v_i} = \{Req_{v_{i-1}}, S_{v_i}[Req_{v_{i-1}}], I_{v_i}\},$$
$$\text{for } i = 1, \cdots, n-1$$

After forwarding an offer request, a node v_i waits $(n-i)\cdot timeout$ seconds for a response, either positive or negative, from v_{i+1}. If it not arrives, it sends a break up chain message to v_0.

5) Node v_n receives the request of transmission from node v_0 along with the information of the relaying parties v_i for $i = 1, \cdots, n-1$. v_n verifies the signatures and checks that the number of hops stated in the contract offer is at most n.

6) If all data is correct and node v_n accepts the transmission from A_0, it generates a contract with the data of the received offer and an appendix with the list of recruited routing nodes, and signs the overall information. It sends the contract to node v_0 using the same bidirectional path as the one used in the reception.

$$Rep_{v_n} = Contr = \{Req_{v_{n-1}}, S_{v_n}[Req_{v_{n-1}}], I_{v_n}\}$$

7) All routing nodes verify the signature of the node v_0 in the contract (because it is the one who pays), keep a copy of it and resend it to the next node in the path toward v_0. Node v_0 receives the contract, verifies the signature of node v_n to check it has contacted with the right destination, and the contract establishment phase ends.

Data Transmission

At the end of the contract set up phase, data transmission can be started.

1) If v_0 wants to send d packets of data to v_n, it will transmit to v_1 the data packets along with a payment check. The payment check consists of the d next hash values of the chain. In fact, presenting the highest hash is enough. For instance, for the first d packets v_0 has to send the chain value w_{n-d}.

$$info = \{packets, w_{n-d}\}$$

2) v_1 verifies the payment, checking that $w_n = h^d(w_{n-d})$, where d is obtained from the number of transmitted packets. v_1 keeps a copy of the w_{n-d} value and forwards the $info$ to the next node. Such operation is performed at each intermediated node v_i, for $i = 1, \cdots, n-1$.

3) Finally, v_n obtains the packet $info$.

C. Charging and Rewarding Model

Charging and rewarding is performed using a protocol between the routing nodes involved in the transmission and the reputation authority, RpA. This phase must be executed anytime after the data transmission session and within the validity period of the contract, when the nodes have online connection with the RpA.

It is important to notice that the possession of a payment proof by a node v_i does not entail that this particular node v_i has forwarded the data, just that it has received it. However, it is clear that v_j for all $0 \leq j \leq i-1$ indeed forwarded the data packets. For that reason, when a routing node v_i with $i \neq n$ reports a payment proof to the RpA, it only receives half of the full router rate, while the lower nodes of the path can be completely rewarded.

Only when the destination node of a packet v_n sends payment proofs to the RpA it is evidenced that the data has been delivered and all intermediate nodes are rewarded. In order to stimulate destination parties to send the proofs, they are also rewarded with a rate of 1 credit token per each packet they demonstrate they have received. The detailed protocol is the following:

1) When node v_i wants to get payed for the forwarding services, it sends to the RpA the forwarding contract, $Contr$, and the payment proof w_k, where $k = m - d$, being m the maximum number of packets that can be transmitted within that session, and d the number of forwarded packets.
2) The RpA verifies that $h^d(w_k) = w_m$, which ensures that the payment proof is valid. RpA obtains the value w_m from $Contr$, where the value is signed by the sender node v_0 and then assumed authentic.
3) Then, the RpA executes the following procedure:
   - If no proof w_k has been previously presented by any node, then RpA adds $(IF_{v_0} \cdot d)$ credit tokens to each node v_j for $1 \leq j < i$ and $(1/2 \cdot IF_{v_0} \cdot d)$ tokens to v_i, in case $i \neq n$. If $i = n$ (i.e. the reporter is the destination node) then v_n is rewarded with d tokens. In any case, the RpA also deducts $(i \cdot d)$ tokens from the credit of node v_0.
   - If v_j, for some $1 \leq j < i$, has already presented the proof w_k to the RpA, then the RpA adds $(1/2 \cdot IF_{v_0} \cdot d)$ credit tokens to v_j, $(IF_{v_0} \cdot d)$ tokens to each node v_k for $j+1 \leq k < i$ and $(1/2 \cdot IF_{v_0} \cdot d)$ tokens to v_i, in case $i \neq n$. If $i = n$, then v_n is rewarded with d tokens. In any case, the RpA also deducts $((i - j) \cdot d)$ tokens from the credit node v_0.
   - If v_j, for some $i < j \leq n$, has already presented the proof w_k to the RpA, then the RpA informs to v_i that it has already been rewarded for such operation.

Since the incentive factor of a node can suffer changes in short periods of time, the rewarding IF_{v_0} to be used in step 3 is the one stated in the reputation certificate which serial number SN_{v_0} appears in the forwarding contract $Contr$. However, when the transmission path is short ($n < 5$), rewarding IF_{v_0} can not exceed 1. This prevents fake nodes to create looping traffic between them in order to increase their credit. Then,

$$\text{Rewarding } IF_{v_0} = \begin{cases} 1, & \text{if } n < 5, IF_{v_0} > 1 \\ IF_{v_0}, & \text{otherwise} \end{cases}$$

It has to be noted that the charging and rewarding model we propose is unbalanced, hence, it faces a problem of credit saturation when all nodes achieve the maximum credit level. This congestion leads the system to work as if it was a plain model that can neither prioritize transmission packets to provide a quality of service, nor offer any real incentive to the routing nodes to spur the data forwarding. To avoid such case, the RpA maintains a sliding window for each node that inspects the accumulated amount of data forwarded by each of them during the last 30 days. If the result does not exceeds 1% of its forwarding credit, this will be reduced 1% every day that passes in these conditions.

Figure 3 illustrates the charging and rewarding model with an example. Nodes only transmit packets from initiators whose incentive factor is greater than a threshold. Node v_0 has two connection routes to node v_5, however, it can not use the shortest one to send data to v_5 because its reputation value is not high enough to encourage the intermediate nodes of this path to forward its packets. Nodes in the shortest path are centrally located in the network, receive a lot of forwarding requests, and only relay packets of nodes which are very collaborative and have a high reputation level. As a result, v_0 has to select the longest path for the transmission, which is more expensive (it costs 5 tokens/packet instead of 3 tokens/packet), but offers the required availability.

IV. Incentives for Cluster-based Ad Hoc Networks

As the networks grow in size, they are more difficult to manage due to the high dynamism of the nodes, which cause frequent changes in the available routes between peers. Route discovery and data dissemination protocols based on flooding incur in sever message overheads when the networks are unstructured and information has to reach all nodes in a lot of independent branches.

To overcome these problems networks have to be structured based on the connectivity properties of their members. In the 1980s was first introduced the idea of creating a virtual backbone [18] to provide distributed control in mobile radio networks. In virtual backbone architectures, nodes v_i are grouped in a collection of clusters $C = \{cl_1, cl_2, ..., cl_l\}$, and each cluster cl_i has a clusterhead h_i responsible for the transmission arrangement and data forwarding. Clusterheads are connected with one another directly or by means of clustergateways g, so that the union of clusterheads and clustergateways constitute a connected backbone that is used for the network management. A node v_i is a clustergateway if $v_i \in cl_r \cap cl_t$, with $r \neq t$.

Therefore, nodes v_i of a clusterbased ad hoc network can be qualified as clusterheads h, clustergateways g, or cluster members n, that is $V = \{H, G, N\}$, with $H = \{h_1, h_2, ..., h_l\}$ the set of clusterheads, $G = \{g_1, g_2, ..., g_w\}$ the set of clustergateways, and $N = \{n_1, n_2, ..., n_p\}$ the set of plain cluster members.

Cluster-based architectures allow that networks appear smaller and more stable from the point of view of cluster members because changes in the configuration of a particular cluster do not affect the network in its entirety. Cluster members do not need to manage routing information themselves but can directly communicate with their clusterhead that gathers information about the location and available resources of each node in the cluster. Routing is carried out through the spine of the network so when a node needs to communicate to a remote peer, only the clusterheads and clustergateways are involved in the search of a transmission path. Then, different

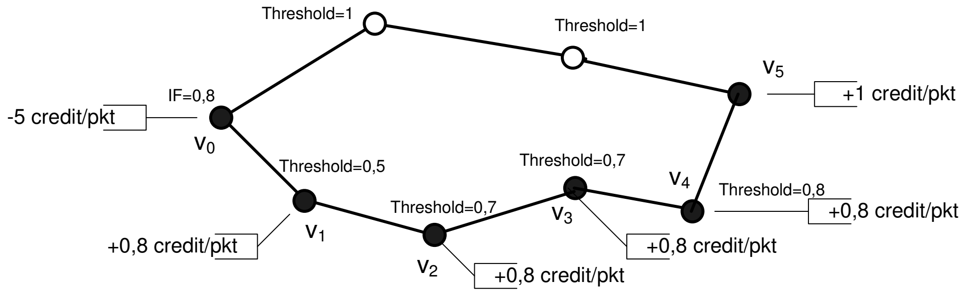


Fig. 3. Illustration of the payment charges

transfer models can be used to encourage the forwarding of data through that path.

We define two transfer models suited for forwarding protocols that are session based, like FURIES. That is, a channel has to be set for being able to carry data through it. The proposed models are the End-to-end Session Model and the Layered Session Model.

A. End-to-end Session Model

In the End-to-end Session model, the initiator establishes a transmission channel with the correspondent node that passes through a set of clusters. In FURIES, establishing a session means signing a contract between some actors that is the base of an agreement for carrying the traffic. For each cluster there are two stipulated nodes involved in the forwarding, the clusterhead and a clustergateway (see Fig. 4a). A contract is signed between the initiator, correspondent, and all selected clusterheads and clustergateways in the path that facilitates the transmission. If a breakdown occurs, another end-to-end session has to be established using new available intermediate nodes.

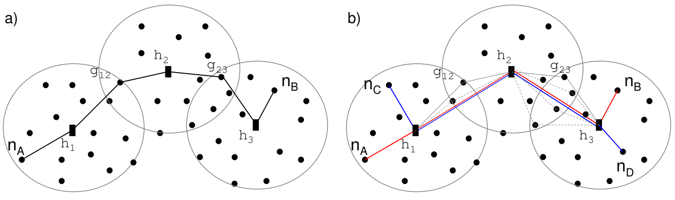


Fig. 4. Forwarding models. (a) End-to-end, (b) Layered

Clustering in ad hoc networks is performed when networks grow in size, so transmission channels in these environments are usually long. We approximate the duration τ_k of a link e_k in an ad hoc network by an exponential distribution (equation 2) with an average staying time $\overline{\tau_k} = \mu = 1/\lambda$, and the composition of an n-hop path P by the joint distribution of all of them (equation 3). In [19] Sadagopan et al. empirically observe that the probability distribution function of the duration time of a multihop path is an exponential function.

$$f_{T_k}(\tau_k) = \lambda \cdot e^{\lambda \tau_k} \tag{2}$$

$$f_P(t) = f_{T_1 T2..Tn}(\tau_1, \tau_2, .., \tau_n) = \lambda^n \cdot e^{\lambda(\tau_1+\tau_2+..+\tau_n)} \tag{3}$$

Considering the duration of each link is independent from the others, the probability that an n-hop path stays active over a stated time T is:

$$P(t \geq T) = \int_T^{\infty} f_P(t)\ d\tau_1 d\tau_2..d\tau_n = e^{-\lambda Tn}$$

Then, the period of time a multihop path is available in a mobile ad hoc network decreases exponentially with the number of hops, and the initiator and the corresponding nodes may be required to reestablish the connection channel between them using different forwarding peers several times in a transmission. Setting up new forwarding channels in a single transmission incurs in relevant overheads of time (delays for detecting death sessions and establishing new ones), processing power (execution of the micropayment protocol), bandwidth (setting up a new session) and energy.

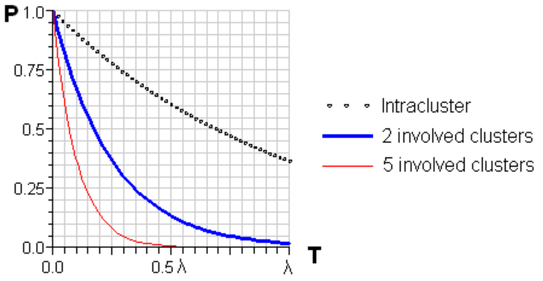


Fig. 5. Probability of a path duration in an End-to-end Session Model

Figure 5 compares the probability of a path staying more than T time for different path lengths. It is worth noting that while for an intercluster transmission the probability of getting a path with a minimum duration of μ is approximately 36.8%, for a communication in which 5 clusters are involved this probability represents an estimated duration of $\mu/10$.

B. Layered Session Model

In the light of the data presented in the previous part, we introduce a more flexible forwarding model that does not require so long and static paths. The initiator node establishes a virtual forwarding session toward the correspondent that only involves the end peers and clusterheads. This session is called *virtual* because clustergateways, the nodes that link the clusters, are not included in the contract.

For their part, clusterheads have to manage the relying operations inside their particular cluster (intercluster sessions) and are responsible to constitute connection paths among themselves using clustergateways so that the transmission from one cluster to another one is possible.

Since clusterheads can join the traffic of different virtual sessions into one single intercluster channel between them and a gateway, these kind of connections optimize network resources. On the other hand, two clusters can have different routes interconnecting them if there are several available clustergateways. This multipath ensures a better response in case of faddings, traffic congestion, etc.

The period of time a virtual channel connecting two peers is available is much longer than the transmission paths created in the End-to-end Session model. Not only the session involves less nodes (nearly the half), but the average duration time of links between clusterheads is longer than between any two cluster members. This is because the clusterheads of a network are chosen using election protocols that ensure the suitability of the picked nodes to cover this position. Apart from taking into consideration the localization properties of the nodes and their available resources, the stability and the reliability of their links is also evaluated.

Fig. 4b shows a Layered Session model. The transmission contract between node n_A and n_B, or between node n_C and n_D, involves 5 peers instead of the 7 used in the End-to-end Session. Besides, custerhead h_1 sets a contract with h_2, and h_2 another one with h_3. These clusterheads paths will be used for carrying packets of both n_A to n_B and n_C to n_D channels.

C. FURIES for Cluster-based Ad Hoc Networks

FURIES protocol can work in cluster-based ad hoc networks. The protocol fits seamlessly in the End-to-end Session Forwarding model, without requiring any change in the scheme. However, in mobile ad hoc networks it is more efficient to use the proposed Layered Session Forwarding model because is more robust provided the dynamic behavior of cluster nodes. Following, we describe the Contract Establishment process, Data Transmission and Charging and Rewarding phase for a Layered Session Forwarding model.

Contract Establishment

Ad hoc networks periodically check their configuration and topology to adapt to the environment. After the clustering formation process, clusterheads get information of their cluster members to know if there are nodes that belong to two or more clusters and thus, can act as clustergateways and extend the range of the network. If neighboring clusters are found, clusterheads initiate a contract request process with them in order to set up a session and procure intercluster communication. Contract establishment is accomplished using the FURIES micropayment scheme (see section III-B). Intercluster session contracts are always two hops long and involve two clusterheads and a clustergateway.

Then, when a node wants to communicate with another one, it asks its clusterhead to localize the correspondent and using the FURIES contract establishment protocol as before, a virtual session is set up among the initiator, the correspondent, and clusterhead nodes of the intermediate clusters.

Data Transmission

The initiator node sends data packets to the correspondent through the virtual transmission path it has established with clusterheads. Along with data packets, the initiator node sends payment checks for rewarding the intermediate clusterheads that help it reaching the correspondent.

The nodes in this path to the correspondent are clusterheads that are not in the transmission range ones from the others. So, they have to make use of intercluster channels for providing the forwarding functions they have accorded with the initiator. Clusterheads are responsible of intercluster transmissions and have to pay for the services they command the clustergateways. For this reason, clusterheads, before forwarding data to the next hop, have to extract the intercluster payment check they have received from the previous cluster, and attach a check bounded with the intercluster session contract they are going to use.

On the other hand, clustergateways receive payment checks of both virtual and intercluster channels. However, they will only be able to charge the checks associated to their contract with the clusterhead.

Charging and Rewarding

The charging and rewarding is performed between the nodes involved in the transmission and the RpA that manages the credit bags in a similar way as in plain ad hoc networks. Whenever the nodes have connectivity to the Authority, they have to send it the forwarding contract and the payment checks they have received bounded to that contract. Holding a payment check does not proof that the node has forwarded some data, so when a node sends a payment proof to the Authority, it only receives half of the payment it can get, while the previous nodes of the same transmission path can be completely rewarded. At the same time, the initiator node is charged for the number of intermediate nodes that it can be assured that have carried the data.

The initiator node of a virtual channel is charged with 2 tokens/packet per each traversed cluster. Note the difference with payments in plain ad hoc networks, in with the sender is charged with 1 token/packet per each traversed node. The rate of 2 tokens/packet is due to the crossing of a cluster implicates two forwarders, the clusterhead and a clustergateway. Clusterheads are rewarded based on the contract with the initiator, bearing in mind that the IF of that node modulates the rate of the job, and the higher the reputation of the initiator, the bigger the revenues for forwarding its packets. Finally, clustergateways earn tokens from their contract with clusterheads. However, because the short transmission paths in which clustergateways are involved, the maximum revenue they will get is 1 credit per forwarded packet. This is a FURIES mechanisms to avoid the creation of fake looping routes between friends inside a network.

Besides, in order to stimulate destination parties to send the proofs, they are also rewarded with a rate of 1 credit for each packet they demonstrate they have received.

D. Example

Following, an example of carrying a FURIES transmission in a cluster-based ad hoc network is presented. The architecture of the network is shown in Fig. 4b, and a scheme of the transmission sessions that have to be established is in Fig. 6.

The network is composed of three clusters, each of which with a clusterhead responsible of routing and transmission management. These clusterheads set up intercluster channels between them through the clustergateways they share. This way, clusterhead h_1 establishes two transmission sessions with h_2 through two different clustergateways, and h_2 sets up four channels with h_3.

The cluster memeber n_A in the network wants to send some information to member n_B. The initiator sets a contract through clusterheads h_1, h_2 and h_3 to reach the destination. Once the channel has been established, node n_A can start transmitting data. Table I shows a summary of the payments and profits nodes receive in this example forwarding session.

Let's assume n_A has an Incentive Factor $IF = 1,4$, and it sends 5 data packets through the channel. Since the path traverses 3 clusters, node n_A has to pay $2 \cdot n \cdot d = 2 \cdot 3 \cdot 5 = 30$ tokens for the delivery. On the other hand, each clusterhead in the path earns $1,4$ credit tokens per forwarded packet, that is, $IF \cdot 5 = 1,4 \cdot 5 = 7$ tokens for all the traffic. Finally, the correspondent node n_B also receives 1 token per each data packet it reports to have received. The reporting of 5 data packets gives it 5 tokens.

Intercluster transmissions, in their turn, also entail some payments. Clusterheads h_1 and h_2 have to pay 5 tokens to their clustergateways to send data packets to the neighboring cluster. Clusterhead h_1 uses an intercluster session that passes through clustergateway g_{12} to reach clusterhead h_2, and the path from h_2 to h_3 goes by g_{23}. Assuming the IF of clusterheads h_1 and h_2 is positive, the clustergateways g_{12} and g_{23} will earn 1 token per forwarded packet, that is, a total of 5 tokens. Besides, clusterheads h_2 and h_3 receive tokens for reporting payment evidences to the reputation authority. In particular, they earn 5 tokens.

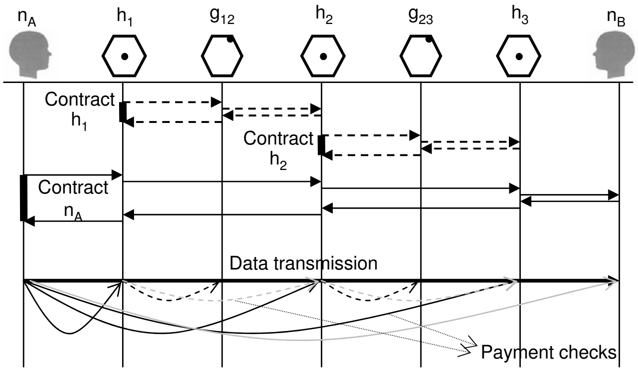


Fig. 6. Session Establishment and Data Transmission

TABLE I
CHARGING AND REWARDING EXAMPLE

Node	Payment	Profit	Total
n_A	30	-	-30
h_1	5	7	2
g_{12}	-	5	5
h_2	5	7+5	12
g_{23}	-	5	5
h_3	5	7+5	12
n_B	-	5	5

Table I presents a summary of the payments and profits nodes receive in the above example. It can be observed that clusterhead h_1 is the node that receives less profits for its job, only 2 tokens. Anyway, this does not put a brake in the forwarding rate of a network because the first clusterhead of a path is the manager of the group in which the initiator belongs, and a clusterhead is interested in giving good services to its members because they are the ones that will help it in intercluster operations. Thus, clusterheads, beyond the motivation of earning tokens for the forwarding, always take a special consideration for the packets of their cluster members. If a clusterhead has good collaborators and the cluster is able to carry traffic from and to other parts of the network, the clusterhead will be the one that will get the most profits. This is manifested in the example, clusterheads h_2 and h_3 earn 12 tokens each one for carrying 5 data packets, which is much higher than the 5 tokens that an initiator node has to pay for sending 5 data packets through a single node.

V. EVALUATION

Simulations of FURIES were conducted to evaluate the general characteristics of the protocol and provide a proof of concept. We used a self-developed application that considers network layer factors and allows us to make qualitative appraisals. However, we do not model the problems of physical and link layers, so that quantitative performances can not be directly extracted from the tests.

We simulated two different payment models in an unstructured ad hoc network: a plain payment protocol without incentives, such as [10] (that is, sending one packet through 3 hops costs 3 credit tokens, and the intermediate nodes get 1 token each one), and the proposed FURIES protocol with the incentive factor defined in section II-B.

The simulated networks are composed of 100 nodes that move randomly in a square area of $1000m^2$. The transmission range is $70m$. Each node starts, on average, 2 transmissions a day of messages the size of which is uniformly distributed from 1Kb to 10Mb. The application is run during a simulation period of a year. 100 simulation runs have been performed.

Table II compares the results of a population attempting to send data through a multihop network giving the mean and variance over 100 simulations. We have modeled the nodes willingness to forward packets based on their available resources (i.e. battery level), and the profits they can make for the action. Relaying parties do not transmit if the battery level is below 20%. However, our assumption is that between 20% and 50% they will resend packets if they obtain a credit rate over the cost price, in particular, a benefit more than 30%. If the remaining battery is above 50%, nodes will transmit if the reward is at least the 90% of what they offer. Despite the battery level, we also assume that nodes with a negative credit balance will accept any forwarding request. Otherwise, when the forwarding is rejected, the initiator has to search another routing path. It tries it up to five times.

First of all, it has to be noted from the first row of Table II, that the number of accepted transmissions in FURIES is greater than in the plain payment protocol. This is one of the goals of incentive protocols, and FURIES achieve it. By offering appropriate incentives -a good reputation status that, as we state in the next point, provides a quality of service-, FURIES can take profit of the maximum forwarding capacity of nodes and thus improve the overall throughput of the network.

The service of forwarding packets is rewarded with credit tokens, and the accumulation of tokens increases the reputation status. The second and third rows of Table II show that in plain mode the reputation level of nodes which packets are accepted or rejected is not relevant since its average is the same as the rest of the population. That is, in spite of its accumulated tokens, the sending of any node can be blocked. Nevertheless, in FURIES accepted traffic is from people who hold a better profile (8% better than the average), and rejected one is from those nodes that tend to behave more selfishly (its reputation is 12% worse than the average). Hence connectivity of cooperative nodes takes priority and such nodes receive a better quality of service.

FURIES spurs cooperation, but does not enforce it. There are multiple reasons for which a node can not collaborate in a determinate moment (lack of resources, bandwidth..). What is not acceptable is a continuous selfish behavior, and thus is penalized. Moreover, when users enter in the FURIES system, they start with a negative reputation level in order to prevent sybil attacks that cause the unfair exploitation of the system.

In general, the advantage of FURIES in front of other credit based mechanisms [7]–[9] is that tokens have a double use: being the exchange currency of the payment protocol and, moreover, being the hook that attracts nodes to relay packets of certain nodes. The accumulation of tokens is awarded, and because tokens can not be obtained by external means, nodes have to provide resources to the net if they want to benefit of its services.

TABLE II
FORWARDING SIMULATION: PLAIN PROTOCOL VS. FURIES

	Plain protocol		FURIES	
Ratio of accepted transmissions	$E(X) = 69\%,$	$\sigma^2 = 0.95$	$E(X) = 83\%,$	$\sigma^2 = 1.82$
Reputation accept vs. average	$E(X) = 0\%,$	$\sigma^2 = 4.64$	$E(X) = 8\%,$	$\sigma^2 = 0.64$
Reputation reject vs. average	$E(X) = 0\%,$	$\sigma^2 = 5.23$	$E(X) = -12\%,$	$\sigma^2 = 2.59$

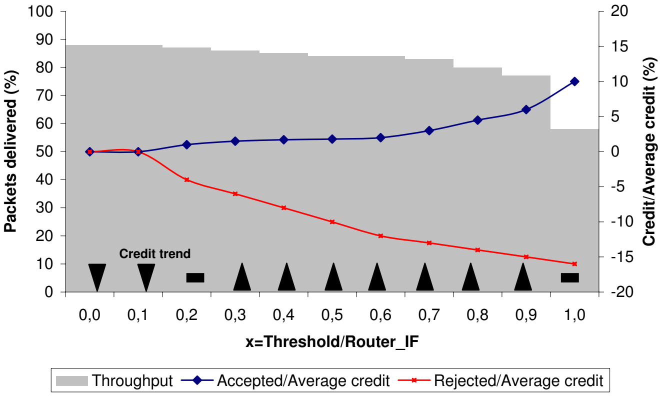


Fig. 7. Forwarding response of an ad hoc network

The evolution of an ad hoc network depends on the behavior of each of its members and how they react to the proposed incentives. We made a simulation of FURIES to analyze the performance of a network relative to the threshold used to trigger the forwarding services. We assume nodes always reject to forward when their battery level is below 20% of its capacity. Otherwise, they accept the transmission if the incentive factor of the initiator stated in its reputation certificate IF_{v_0}, is greater in a certain factor x than its own IF, that is, $IF_{v_0} \geq IF_{v_i} \cdot x$, where v_i is the forwarding node.

Figure 7 shows the results of the simulation based on parameter x, that is, the quotient between the triggering threshold and the incentive factor of the forwarding node. The background columns of the figure depict the percentage of packets accepted to transmit. It is shown that the throughput of the network is nearly constant whatever the threshold. However, when we harden the condition and require the IF_{A_0} is equal to the IF_{A_i} of the forwarding node ($x = 1$), the throughput gets down to 58%. If we would increment the threshold a little more, the throughput will continue to fall toward 0%.

With this result it may seem that the best x to choose is a low one. However, for very low values of x we can not offer quality of service, the probability to get a packet rejected is hardly the same for all kind of nodes. It is worth noting the lines in the figure that show the relation between the reputation level of the nodes which packets are accepted or rejected, and the average level. The more these lines are separated, a better quality of service is offered because the reputation of a node most influence the forwarding acceptance decision.

Moreover, the figure depicts with black arrows the credit storage trend of a group of people whose initial credit level was 0. It is shown that when x is low, the credit storage of the group tends to decrease, so in the long run people will not have credit to transmit.

Therefore, there is a compromise to get the best results. Setting thresholds with low values increases the performance at short term but the network gets unhealthy: less credit tokens, no quality of service, and so, at last, less motivation to do the forwarding. On the other hand, high thresholds can reduce the throughput of the net. Consequently, there is no fixed optimum threshold, it depends on the resources of the node, its eagerness to transmit and so the necessity to obtain tokens, etc. The threshold is a variable that has to be adjusted in every case to get the expected reactions. However, the adjustment can be done automatically to meet the requirements of a specific environment.

VI. Conclusions

In this paper we have presented FURIES, a new model to stimulate cooperation in multihop ad hoc networks. The novelty of the protocol is using a payment system that is based in rewards in the form of quality of service, and not in compensations for the particular efforts and resources destined to make a transaction work. The majority of the actual cooperative forwarding protocols uses this second form of reward, which is very costly in terms from achieving a fair payment for each node, and does not really suppose a clear motivation to participate in the network due to its complexity overcosts.

The FURIES model is light and simple. The charges for sending a data packet only depends on the length of the transmission path, while the payment rewards are a function of the reputation of the sender node. Thus, intermediate nodes prioritize the forwarding of high reputed users' data.

Moreover, the solution is scalable to large ad hoc networks with a layered architecture. We have analyzed the protocol and, by means of simulation, we have evaluated the functionality of the system based on the configurable parameters and we have provided proof of concept.

The results prove that FURIES fulfills its objectives: it improves the throughput of the network and reinforces a quality of service for collaborative nodes.

In terms of future work, we plan to study the performance of the protocol in real environments, evaluate its overhead in terms of energy consumption and delay, and compare it quantitatively and qualitatively with other mechanisms of incentives.

References

[1] H. Rifà-Pous and J. Herrera-Joancomartí, "A Forwarding Spurring Protocol for Multihop Ad Hoc Networks (FURIES)," in *Next Generation Teletraffic and Wired/Wireless Advanced Networking (NEW2AN)*, ser. Lecture Notes in Computer Science, vol. 4712, 2007, pp. 281–293.

[2] G. F. Marias, P. Georgiadis, D. Flitzanis, and K. Mandalas, "Cooperation enforcement schemes for MANETs: A survey," *Wirel. Commun. Mob. Comput.*, vol. 6, no. 3, pp. 319–332, 2006.

[3] S. Buchegger and J. L. Boudec, "Nodes bearing grudges: Towards routing security, fairness, and robustness in mobile ad hoc networks," in *Euromicro Workshop on Parallel, Distributed and Network-based Processing (PDP)*, 2002.

[4] P. Michiardi and R. Molva, "Core: A COllaborative REputation mechanism to enforce node cooperation in Mobile Ad Hoc Networks," *Institut Eurecom. RR-02-062*, December 2001.

[5] Y. Rebahi, V. Mujica, C. Simons, and D. Sisalem, "SAFE: Securing pAcket Forwarding in ad hoc nEtworks," *Work. on App. and Services in Wirel. Networks*, June/July 2005.

[6] Q. He, D. Wu, and P. Khosla, "SORI: A Secure and Objective Reputation based Incentive Scheme for Ad-hoc Networks," in *IEEE Wirel. Commun. and Net.*, 2004.

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

[8] L. Anderegg and S. Eidenbenz, "Ad hoc-vcg: a truthful and cost-efficient routing protocol for mobile ad hoc networks with selfish agents," in *Mob. Compt. and Net. (MobiCom)*. New York, NY, USA: ACM Press, 2003, pp. 245–259.

[9] S. Zhong, J. Chen, and Y. Yang, "Sprite: A Simple, Cheat-proof, Credit-based System for Mobile Ad Hoc Networks," in *IEEE INFOCOM*, vol. 3, 2003, pp. 1987–1997.

[10] L. Buttyán and J. Hubaux, "Stimulating Cooperation in Self-Organizing Mobile Ad Hoc Networks," *Tech.Rep.DSC*, 2002.

[11] J. Crowcroft, R. Gibbens, F. Kelly, and S. Ostring, "Modelling incentives for collaboration in mobile ad hoc networks," *Perform. Eval.*, vol. 57, no. 4, pp. 427–439, 2004.

[12] O. Ileri, S.-C. Mau, and N. Mandayam, "Pricing for enabling forwarding in self-configuring ad hoc networks," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 1, pp. 151–162, January 2005.

[13] Y. Yoo, S. Ahn, and D. Agrawal, "A credit-payment scheme for packet forwarding fairness in mobile ad hoc networks," *IEEE Intern. Conf. on Commun. (ICC)*, vol. 5, pp. 3005–3009, May 2005.

[14] P. Congdon, B. Aboba, A. Smith, G. Zorn, and J. Roese, "IEEE 802.1X Remote Authentication Dial In User Service (RADIUS) Usage Guidelines," RFC 3580, Sep. 2003.

[15] L. Zhou, F. Schneider, and R. van Renesse, "COCA: A Secure Distributed Online Certification Authority," *ACM Trans. Computer Systems*, vol. 20, no. 4, pp. 329–368, 2002.

[16] B. Zhu, F. Bao, R. H. Deng, M. S. Kankanhalli, and G. Wang, "Efficient and robust key management for large mobile ad hoc networks," *Comput. Networks*, vol. 48, no. 4, pp. 657–682, 2005.

[17] R. L. Rivest and A. Shamir, "PayWord and MicroMint: Two Simple Micropayment Schemes," in *Security Protocols Workshop*, 1996, pp. 69–87.

[18] D. J. Baker, J. A. Flynn, and A. Ephremides, "The design and simulation of a mobile radio network with distributed control," *IEEE Journal on Selected Areas in Communications*, vol. 2, pp. 226–237, January 1984.

[19] N. Sadagopan, F. Bai, B. Krishnamachari, and A. Helmy, "PATHS: analysis of PATH duration statistics and their impact on reactive MANET routing protocols," in *MobiHoc: Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*. New York, NY, USA: ACM Press, 2003, pp. 245–256.

[20] H. Tewari and D. O'Mahony, "Multiparty micropayments for ad hoc networks," *IEEE Wirel. Commun. and Net. (WCNC)*, vol. 3, pp. 2033–2040, 2003.

Helena Rifà-Pous is a professor at the department of Computer Science in the Universitat Oberta de Catalunya. She received the degree of Telecommunications Engineering by Universitat Politècnica de Catalunya in 2001 and is currently a PhD candidate working on security on ad hoc networks. Her research interests include network security, key management and mobile networks. From 2000 to 2007 she was with Safelayer Secure Communications working on PKI projects mainly for the public administration.

Jordi Herrera-Joancomartí is associate professor at Departament dÆEnginyeria de la Informació i les Comunicacions in the Universitat Autònoma de Barcelona. He is graduated in Mathematics by Universitat Autònoma de Barcelona in 1994 and he received his Ph.D. degree in 2000 from Universitat Politècnica de Catalunya. His research interests include topics in the field of computer security and more precisely in copyright protection techniques and security in ad-hoc networks. He has published more than 50 papers in national and international conferences and journals. He is also reviewer of different national and international conferences and journals. He has been main researcher in several national research projects.