IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 20, 2020, accepted January 14, 2021, date of publication February 22, 2021, date of current version March 9, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3061258

Per-Hop Reversed Packet Auctions for Cooperative Routing in Mobile Wireless Networks

LUCAS S. BRITO AND MARCELO M. CARVALHO, (Member, IEEE)
Department of Electrical Engineering, University of Brasilia, Brasilia 70910-970, Brazil

Corresponding author: Marcelo M. Carvalho (mmcarvalho@ene.unb.br)

This work was supported by the Coordenaçã de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), Brazil, Finance Code 001.

ABSTRACT Many applications could benefit from multi-hop communications through users' mobile devices. A key issue is how to incentivize users to cooperate in both routing and relay of messages by sharing their device's precious resources. Previous works on the subject have either tackled cooperation in the relay of messages alone or in both routing and relay functionalities. In the latter case, path selection is usually carried out at the destination node, which renders significant delays because the selected path needs to be conveyed all the way back to the source node before any data packet can be transmitted. This is certainly unsuitable in mobile scenarios. This paper presents the performance of the "Tightness" strategy, which allows the routing and relaying of messages "on-the-go", via per-hop reversed packet auctions. At each hop, the sender asks for bids from potential relays according to a "budget" attached to the data packet, through which the auction winner gets paid and can pay for others in subsequent auctions. The auction winner is chosen not only based on bid value, but also on the estimated relay's likelihood to deliver the packet to destination. Likewise, each potential relay makes a bid considering its own chances to deliver the packet to destination. A fine is also announced in every auction, that must be paid by all relays if the packet is not delivered to destination within a "deadline" expressed in number of hops. The performance of the Tightness strategy is evaluated for both static and mobile scenarios and compared to two baseline strategies according to different performance metrics.

INDEX TERMS Auctions, cooperation, routing, wireless networks.

I. INTRODUCTION

A number of services and applications under today's ever-increasing mobile traffic demands could benefit from multi-hop communications through mobile devices such as smartphones, tablets, and wearable devices. In particular, in many periods of the day, the owner of a mobile device does not make any actual use of it, and the device just stays on a table or in a pocket while its owner walks to school or work. In other occasions, people gather in large-scale events, such as concerts, games, and demonstrations, or visit theme parks and zoos, while their devices stay idle for significant fractions of time. So, what if these "idle times" were used to perform useful work to benefit *other* mobile users while, at the same time, the cooperative user would receive some form of reward for sharing part of her device's precious resources (e.g., battery energy and bandwidth)? Examples of potential applications include coverage extension of Internet access services [1], mobile data offloading [2]–[5] (and references therein), phone-to-phone communications [6], [7], or even wireless sensor networks [8], to name a few. In most of these applications, however, network participants are expected to act independently of each other and to make their own decisions regarding cooperating in the network operation. Consequently, multi-hop networking through users' devices remains a great challenge because users are most likely to avoid relaying someone's traffic if they do not receive any rewards or compeling incentives for that. For instance, in the context of device-to-device data offloading, Rebecchi *et al.* [2] have argued that the issue of how to incentivize users to cooperate in the offloading infrastructure is a key aspect that has not received much attention in the literature.

Given this fundamental problem, a number of works have looked at ways to promote cooperation in the formation of infrastructure-less multi-hop wireless networks, especially in

The associate editor coordinating the review of this manuscript and approving it for publication was Javed Iqbal.

the routing and packet forwarding tasks, which are some of the key functions in the deployment of such networks. In particular, some works have considered mechanisms based on monetary compensation (in general, virtual transfers) to incentivize nodes to participate in both routing (i.e., to find least cost path to destination) and relay of messages [9]–[11], or in the relay of messages alone [12]–[15]. The works that have focused on the relay of messages alone implicitly assume that nodes already cooperate in the running of an underlying routing protocol (e.g., DSR or TORA). On the other hand, all proposals that have tackled both routing and relay of messages have mainly assigned the path selection task to the destination node of the data packets. As a result, the information about the selected path needs to be conveyed all the way back to the source node before any data packet is transmitted. Consequently, these approaches are generally not well suited for operation in mobile networks, and they have been mainly evaluated under static scenarios. Last, but no least, some other approaches [15]–[17] have assumed the existence of a control center that takes care of all the routing/assignment decisions, which is clearly not in the spirit of the application scenarios previously discussed.

Based on these observations, and other considerations, we present the *Tightness* strategy to promote cooperative routing and relaying of messages in multi-hop mobile wireless networks. A distinctive feature of the *Tightness* strategy is the fact that both routing and relay of messages are executed ''on-the-go,'' which means that it avoids the delay incurred by path selection only at the destination node. To incentivize nodes to cooperate, the *Tightness* strategy is based on per-hop *reversed* auctions, by which potential relay nodes (sellers) within range of the sender (buyer), make their bid to carry out the forwarding task based on a ''budget'' value announced by the sender. The bid values are created by following a *bidding strategy*, and the sender selects the auction winner according to its *preference function* (i.e., an utility function). The sender immediately forwards the data packet once it decides the auction winner, and the process repeats itself all the way until the packet finally reaches its final destination (this is what we denote as ''recursive auctions''). At each hop, the auction winner retains some amount of the budget to itself, and uses the remaining budget to run its own auction if it needs the help of other nodes to forward the packet to its final destination.

The core idea of the *Tightness* scheme relies on the fact that the auction winner (i.e., the one indicated by the preference function) is not only based on the minimum bid value (as it is usually done in many auction-based solutions [11], [12]), but also on the *likelihood* of a given relay node to deliver the packet to its final destination. In other words, it also depends on how ''tight'' a given relay node is in accomplishing its mission (in terms of number of hops) if the repetitive bidding process continues down a path starting from it (while nodes may move around). Conversely, in order to increase the chances of data delivery and to avoid a greedy behavior, the *bidding* strategy also takes into account how likely the relay node itself is in delivering the data packet if it wins the auction. In our system model, all nodes that participate in the routing and forwarding tasks should run the risk of paying a *fine* if the packet is not delivered to its final destination. Hence, together with the budget, a fine and a ''deadline'' (in terms of number of hops) are also announced in every auction (the data source defines the deadline). The announced fine is always smaller than the fines agreed upon in previous hops, while the ''deadline'' is decremented every time the packet is forwarded to someone else. This way, not only the buyer (sender) tries to make the best decision with respect to the best relay of its packet, but also the relays themselves participate more aggressively (or not) in the announced auctions depending on their own assessment of the risks in taking the job. The association of a penalty (''fine'') to the packet-delivery job constrained to a maximum number of hops has not been considered in previous works.

The concept behind the *Tightness* strategy has been introduced previously [18], which was originally designed for a data offloading scenario within the context of the *Mobile Ad Hoc Networking Interoperability and Cooperation (MANIAC) Challenge* 2013 [19], [20]. However, its performance has not been presented before and, in this paper, we present a comprehensive evaluation of the *Tightness* strategy for both static and mobile scenarios, under different node speeds. Hence, different from proposals who assume a 2-connected topology for their solution to work (i.e., a topology with at least two node-disjoint paths from any node to destination) or other constraints on topology [13], [14], our proposal does not require any assumption about the topology, and the network scenario considers the possibility of both channel errors and packet collisions under the operation of an actual MAC protocol, the IEEE 802.11. The performance study is based on discrete-event simulations carried out with the ns-3 simulator [21]. For comparison purposes, two baseline strategies are also investigated: one that prioritizes packet delivery over budget gains (by applying shortest-path routing regardless of the bid values), and a greedy one, where the auctioneer always picks the relay node who issued the lowest bid regardless of its likelihood of delivering the packet within the deadline. All strategies are evaluated with respect to packet delivery ratio, average budget per node, budget fairness, and average number of hops needed for the packet to reach its destination.

The remainder of the paper is divided as follows. Section II describes the system model, auction rules, and the concept of a credit clearance center. Section III presents the details of the *Tightness* strategy and the adopted baseline schemes used for performance comparison. Section IV describes the simulation scenarios, and Section V presents the simulation results. Section VI discusses related work, while Section VII contains the conclusions of this work.

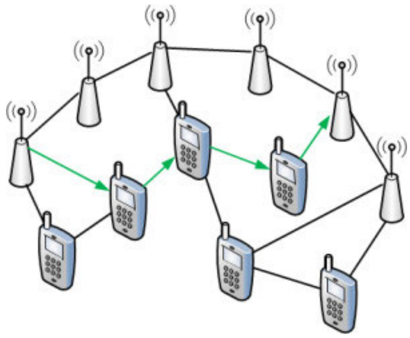


FIGURE 1. Data offloading scenario. The nodes relay packets from one source AP to a destination AP through multi-hop recursive auctions.

II. SYSTEM MODEL

A. AUCTION RULES

The scenario we consider in this paper assumes the data offloading problem where a mobile operator wants to send a portion of its traffic through an infrastructure-less multi-hop network formed by its clients' devices (we will use the term "T2T network" for short, in a broad sense [4]). Hence, all data packet transmissions initiate from one of the operator's access points (APs). The incentive for the operator is to decrease traffic through its backbone infrastructure (decreased costs), while the incentive for customers is to receive discounted monthly fees, for example, in the form of some credit or virtual payment. The operator's access points (APs) offload data packets targeted to specific destination devices. In this paper, we assume that all data packets have another AP as their destination (e.g., an AP that is the actual Internet gateway, while the other APs are just part of an extended distributed system for coverage extension purposes). Therefore, the job of the mobile devices is to deliver these packets to destination APs indicated by source APs.[1] Figure 1 depicts the application scenario for recursive per-hop auctions to achieve multi-hop routing and relay of messages. The lines connecting the devices indicate wireless connectivity, and the arrows indicate a path that a data packet might take to traverse the network formed by mobile devices.

A backbone AP initiates a data forwarding request by sending a special MAC-level broadcast packet we name *Request For Bids* (RFB). The AP's RFB contains the following information:

- Initial *budget* B_0 of "credits" for the payment of nodes involved in the successful delivery of the data packet to its target destination;
- "Deadline" for packet delivery, translated into a *maximum number of hops* H_0 allowed for a packet to traverse the T2T network before reaching the target destination;
- A *fine* F_0 to be paid if the data packet is not delivered to the destination within the "deadline" of H_0 hops.

The initial values B_0 and F_0 depend on the application scenario and the operator's approach to reward participants in the data offloading task (e.g., discounted monthly fees, credits on mobile data plans, etc.). Therefore, similar to previous works on incentive schemes [4], [22], [23], we do not propose a specific technique to set up such values, and we treat them as arbitrary. Certainly, the definition of these values are key to motivate users to join the T2T network: there should be enough credits to be shared among successful forwarding nodes, and the fine should be set high enough to inhibit reckless or greedy bidding behavior. Likewise, to set up the deadline H_0, the AP would need to take into account the application's requirements on maximum acceptable delay per packet, and translate that into a number of hops. For that, some parameters would likely be considered, such as the number of participants in the network, coverage area, link-layer transmission range, expected delay per auction execution, mobility of nodes, etc. The study of the optimal assignment for B_0, F_0, and H_0 is out of scope of this work.

Every node, neighbor to the source AP that receives the RFB must participate in the auction by making a bid with the sending of a *bid* packet. After waiting for a time interval t_0 long enough to receive the neighbors' bids, the source AP decides for the auction winner by *always* choosing the node with the *lowest bid* $b_i \leq B_0, \forall i \in \mathcal{B}$, where $\mathcal{B}$ is the set of nodes whose bids were received by the auctioneer within the time interval t_0. Note that this is the only case when the auction winner is selected without using the *preference functions* to be defined later (from the point of view of the operator, the lowest bid is always the best). Then, the source AP forwards the packet to the auction winner. From this point on, per packet hop-by-hop recursive auctions are performed in order to forward each data packet towards the destination AP. It is assumed that all nodes that accept to join the *per-hop auction application* must comply to the requirement that they must always send a bid as a response to *any* received request for bids (RFB), regardless of their individual assessment of how advantageous is their participation in the particular announced auction. Each device has the freedom to deliver the packet to the destination AP either via the T2T network or the provider's infrastructure backbone of APs (if within range, of course). Using any backbone AP for delivery guarantees 100% packet delivery, but a node that bypasses the T2T network by using the backbone must pay a price equal to the initial maximum budget B_0 (when the packet was first introduced in the network).

Each node advertises its own maximum budget and fine in its RFB, except for the maximum number of hops ("deadline"), which decreases from the original H_0 every time the packet is forwarded one hop. Also, some constraints must be obeyed: in the n-th auction for a given packet, the fine must always be smaller than or equal to the budget defined in the RFB, i.e., $F_n \leq B_n$. Likewise, the advertised fine,

[1] Note that this assumption does not limit the application of our strategy. The target destination could be any mobile device. In Section IV we also consider mobile scenarios, and the only fixed nodes are the APs. Different pairs of source/destination APs are used in simulations to create different packet flows and possible paths.

in every auction, must be smaller than or equal to the fine agreed upon in the previous hop, i.e., $F_n \leq F_{n-1}$, where n indicates the nth auction for a given packet. After receiving the bids, a node chooses the winner downstream node based on its own strategy. A node that wins an auction is allowed to drop the packet based on its own strategy. In order to avoid routing loops, a device is not allowed to bid for a data packet it has already forwarded once. An upstream node pays the agreed budget to the chosen downstream node if the packet is *successfully delivered to the destination AP*. Otherwise, the downstream node must pay the agreed fine to the upstream node if the data packet does not reach the AP destination within H_0 hops (and then, successively, all the way upstream). A node's balance may be temporarily negative. It is assumed that all nodes share connectivity information by executing some underlying protocol for topology dissemination only (notice that this is different from running a topology control protocol, as it is done by COMMIT [13]).

B. THE CREDIT CLEARANCE CENTER

The virtual currency can be implemented using the idea of a "Credit Clearance Center" (CCC) [22]–[25], which is a server connected to the Internet that nodes can access whenever they go online. All nodes in the (offline) T2T network are assumed to have registered to the CCC prior to participation. The CCC is responsible for the storage and management of the nodes' accounts, as well as the generation of private/public key pairs and certificates with unique identifications for each node. Every packet auctioned by the source node contains a header field that stores the signatures of all nodes that are responsible for its relay to the destination, including the source node. Additionally, this header contains the packet ID, the destination IP address, a time stamp for the packet's first auction, and the values B_0, F_0, and H_0.

At each hop, the n-th auctioneer generates a record for itself about the number of hops the packet has traversed so far, the budget and fine values (B_n, F_n) announced in its RFB, as well as the bid value (O^*) and IP address of the auction winner. The auction winner keeps a record of the agreed fine (F_n), its winning offered bid (O^*), the number of hops the packet has traversed so far, and IP address of the previous node (auctioneer). This bookkeeping is repeated at each hop, as the data packet advances to the target destination. Once the destination receives the data packet successfully (i.e., within the announced deadline H_0), it sends the information embedded in the packet to the CCC. In addition to the information contained in every packet delivered successfully, the CCC updates every nodes's account as soon as it receives their own records from packet transactions. This happens every time a node accesses the CCC through the Internet (e.g., when the node accesses the Internet through its operator, who happens to run the CCC as well). The CCC can then consolidate each node's balance based on its records and the stored information about each packet.

In case a data packet is delivered to the destination beyond the deadline H_0, the destination reports it to the CCC, which applies the corresponding fines to the relay nodes. In case a node decides to deliver the packet to the wrong AP (as allowed by the rules), no fine is applied, but the node who decides for this action is charged with the initial budget B_0 for not delivering it to the target destination. If a node fails to relay a data packet because either 1) it realizes it will not be able to deliver the packet within the deadline, 2) it does not have any neighbor around it; or 3) the packet is lost due to channel errors (no ACK received), the node registers the packet drop/loss, and the CCC applies the associated fines to involved relay nodes after receiving this node's records. Finally, note that the payments and fines are not applied immediately, but they happen off-line, in batches, after a given time period of data collection (e.g., a day, week or month-worth of participation).

III. AUCTION PARTICIPATION STRATEGIES

In Section II-A we presented the general rules for participating in the auctions, i.e., the rules by which any node, implementing any strategy, should obey. In this section, we present the specific strategies we propose for the devices to participate in the recursive auctions. Each strategy comprises three sub-strategies: the *bidding* strategy, which defines how to set the value of a bid for a given RFB received from a neighboring node, the *budget-and-fine* setup strategy, that defines how a node, who just won an auction, sets the budget and fine values of its own RFB, and the *decision-making* strategy, which defines how an auctioneer picks the winner of its announced RFB. Each of these sub-strategies can be specified in different ways, according to different goals.

The first strategy we introduce is actually a *class* of strategies that is based on the central idea of "tightness" with respect to packet delivery within a given deadline, i.e., how much "room" a node has (before reaching the deadline) to absorb eventual bad forwarding decisions resulted from the unpredictable outcomes of other downstream auctions. Therefore, we actually present a set of "Tightness Strategies," and we differentiate them in this paper with respect to the *preference function* used in the *decision-making* strategy (the other two sub-strategies are kept the same, for the sake of evaluation). The idea of the tightness strategy was introduced previously [18]. For completeness, we reproduce the main ideas in this paper, followed by presentation of the other two new variations in the *decision-making* sub-strategy. Then, two other strategies are introduced for purpose of performance evaluation and comparison. These two strategies do not make use of the "tightness" concept, and they differ with respect to the *bidding* and *decision-making* strategies.

A. TIGHTNESS STRATEGIES

When a *source access point* (AP) announces its *request for bids* (RFB), it announces the budget B_0 along with a fine F_0 to be paid in case the data packet is not delivered to the destination AP within the deadline H_0 (measured in number of hops). Given that all nodes have access to topology information (i.e., connectivity), let hc_i denote the number of hops (or "hop count") of the *shortest path* computed from node i to the destination AP. Also, let p_i denote the number of hops traversed by a packet from the source AP to a given node i in the network. A key metric in the tightness strategy is the definition of a "tightness function" Δ_i for a node i in the network, i.e., Δ_i measures how "tight" a node i is with respect to making the deadline H_0 imposed by the source AP. In other words, given the deadline H_0 announced by the source AP, and the number p_u of hops already traversed by the data packet all the way to node i's upstream node u (the one who issues the RFB), Δ_i measures the "surplus" or "deficit" (in number of hops) that node i possess with respect to the deadline H_0 *if the data packet were forwarded through its shortest path to the destination AP*. In other words,

$$\Delta_i = (H_0 - p_u - 1) - hc_i, \quad \forall i \in \mathcal{N}(u), \tag{1}$$

where $\mathcal{N}(u)$ is the set of nodes who are able to overhear the RFB from node u, i.e., the neighbors of node u. Therefore, if $\Delta_i < 0$, node i cannot deliver the data packet within the deadline (even if the data packet follows node i's shortest path to the destination AP). On the other hand, if $\Delta_i = 0$, node i needs *exactly* the number of hops contained in its shortest path to the destination AP in order to make the deadline. This is a "tight" situation for node i, since it relies on the unpredicted outcome of other downstream auctions for the packet to arrive within the deadline. Finally, if $\Delta_i > 0$, node i has a higher chance to deliver the data packet within the deadline because the packet may even deviate from its shortest path to the destination AP, but it has a "surplus" of hops before the deadline is up.

1) BIDDING STRATEGY

The rationale for making the bid value takes into account the likelihood of fulfilling the task of delivering the packet to destination within the deadline. Otherwise, a fine will be paid to the operator. Hence, each node needs to assess how likely it is to deliver the packet as compared to other auction contenders (or competitors). For that, we first need to determine the set $\mathcal{N}(u)$ of neighbors of the upstream node u, the auctioneer. This set contains *our competitors* in the upcoming auction, and it can be easily found because all nodes have complete knowledge of the network topology. For every node $i \in \mathcal{N}(u)$, we compute Δ_i according to (1). Based on the values of Δ_i, we create a subset $\mathcal{S}(u) \subseteq \mathcal{N}(u)$ that contains all nodes in $\mathcal{N}(u)$ such that $\Delta_i \geq 0$, i.e., the set $\mathcal{S}(u)$ contains all nodes that are actually able to deliver the packet within the deadline and, therefore, they are the ones most likely to win the auction announced by node u (our actual competitors). Observe that, we are assuming that node u will usually prefer not to pay a fine. Given $\mathcal{S}(u)$, we want to estimate how competitive we are in terms of packet delivery from the point of view of node u. It is reasonable to expect that the likelihood of successfully delivering a packet will play a key role in any decision making by any node. Therefore, we choose to find out how competitive we are by using our "tightness function." Specifically, we compute how "tight" we are with respect to the *average tightness* $\overline{\Delta}$ of nodes in $\mathcal{S}(u)$, defined as

$$\begin{aligned}\overline{\Delta} &= \frac{1}{|\mathcal{S}(u)|}\sum_{i\in\mathcal{S}(u)}(H_0 - p_u - 1) - hc_i \\ &= (H_0 - p_u - 1) - \overline{hc},\end{aligned} \tag{2}$$

where $|\mathcal{S}(u)|$ is the cardinality of $\mathcal{S}(u)$, and $\overline{hc}$ is the average optimal hop count over all $i \in \mathcal{S}(u)$, i.e., the average *shortest path* to the destination AP computed for each node $i \in \mathcal{S}(u)$. Once the average tightness $\overline{\Delta}$ is found, we compute our *relative tightness* c_n, defined by

$$c_n = \frac{\Delta_n}{\overline{\Delta}}, \tag{3}$$

where the subscript n is used to identify ourselves. It is important to mention that the above computation will only happen if our tightness function is such that $\Delta_n > 0$ and $|\mathcal{S}(u)| > 0$. Otherwise, we have specific rules for making our bid (explained later). Observe that, if $c_n < 1$ and $\Delta_n > 0$, then our competitors are better positioned than us (on average, with respect to a surplus of hop counts). Therefore, there is a high chance that they become more aggressive to win the bidding, since they may feel that they can deliver the packet in time. At the same time, since $c_n < 1$, it means that we are running a higher risk on not having the packet delivered to its final destination, compared to others. Therefore, we may want to set a higher bid (closer to the budget value B_u) because the risk should not be worth taking it. In case $c_n \approx 1$, we have similar conditions than other competitors and, therefore, we should try to win the auction with a lower bid compared to previous case. However, if $c_n > 1$, it means that we are better positioned than the average of our competitors. Therefore, we should strive to win the bid by offering a very attractive price (closer to the fine F_u).

In addition to c_n, another important metric to take into account is how Δ_n (the value of our tightness function) compares to the *biggest* value of Δ_i for $i \in \mathcal{S}(u)$. This is because, if $\Delta_n > \Delta_{\max} = \max_{i\in\mathcal{S}(u)} \Delta_i$, it means that we are the best choice for the upstream node u in terms of a positive surplus of hop counts towards destination. Therefore, we should strive to win the auction by becoming as aggressive as possible in our bid (i.e., to set lower values for the bid to make sure we win the auction). Otherwise, if $\Delta_n \ll \Delta_{\max}$, we should have very low expectations to win the auction and, therefore, we should not make dramatic changes in our bid for different values of c_n. Based on that, we define the parameter a_n that compares our tightness value with the best tightness value in $\mathcal{S}(u)$, i.e.,

$$a_n = \frac{\Delta_n}{\Delta_{\max}}. \tag{4}$$

Given the array of values $\boldsymbol{x} = [c_n\ a_n\ B_u\ F_u]$, where B_u and F_u are the budget and fine values announced by the upstream node u, respectively, and since $F_u \leq B_u$ (according to the

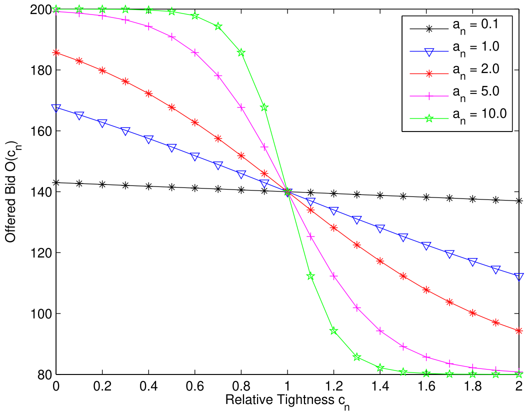


FIGURE 2. **Examples of offered bid curves $O(c_n)$ for different values of the parameters a_n and c_n, when $B_u = 200$ and $F_u = 80$.**

auction rules), our *offered bid* $O(\boldsymbol{x})$ is given by

$$O(\boldsymbol{x}) = (B_u - F_u)\left[1 - \frac{1}{1 + e^{-a_n(c_n-1)}}\right] + F_u, \qquad (5)$$

which means that $F_u \leq O(\boldsymbol{x}) \leq B_u$ because we opt for never making a bid less than the announced fine F_u. Figure 2 shows an example of the offered bid function for different values of a_n when $B_u = 200$ and $F_u = 80$. Note that the logistic function is centered at $c_n = 1$ and the *steepness* of the curve is controlled by a_n.

Finally, if $\Delta_n < 0$, we discourage the upstream node from choosing us by setting our bid equal to the budget B_u. Likewise, if there is no competition, i.e., we are the only node reachable by the upstream node, we set our bid to the maximum value B_u (the auctioneer has no option), and if $\Delta_n = 0$, it means that we are very ‘‘tight’’ and, therefore, we should set our bid to B_u (high risk).

2) BUDGET-AND-FINE SET UP STRATEGY

Once an auction is won, the strategy to set the new values for the budget B_n and fine F_n to be announced in an RFB is based on a fixed rule. Given that an upstream node has paid a node n an amount equal to the winning offer O^*, the budget B_n and fine F_n values set by node n are given by

$$B_n = 0.95 \times O^* \quad \text{and} \quad F_n = 0.4 \times B_n. \qquad (6)$$

The rationale for using this heuristic approach is the following: every auction winner should keep a fraction of its winning offered bid to itself (its ‘‘payment’’) as part of its reward to participate in the cooperative network. Then, the remaining budget value is used to formulate the next announced budget for that packet, since there should be enough budget for downstream nodes to perform their own auctions. Note that the propagated budget values should not decrease too sharply along the route. Otherwise, a small budget value would remain to the last nodes in the path. This is why we propose that each auction winner should keep only 5% of the received budget, while the remaining 95% of it is used to formulate the new announced budget. But, given that downstream nodes can only bid values that are equal or smaller than the announced budget, the auctioneer may still keep some extra ‘‘credits’’ after the execution of its auction.

Another consideration is that upstream nodes are more distant from the target destination and, therefore, they should receive a reward that is proportional to the early risk involved in relaying a packet that may not be delivered successfully by downstream nodes. On the other hand, downstream nodes are closer to destination and, therefore, can have a better assessment of how ‘‘tight’’ they are to deliver the packet to destination because they are likely to have a more updated topology information regarding the destination node. Therefore, such nodes have less uncertainty regarding the likelihood of packet delivery. Consequently, they should receive a smaller compensation compared to upstream nodes. At the same time, downstream nodes should not pay a high fine due to bad decisions made by upstream nodes. Therefore, if downstream nodes find themselves in a very tight condition that prohibits them to deliver the packet within the deadline, they should pay a fine that is proportionally smaller than the ones paid by upstream nodes. The proposed fine value corresponding to 40% of the announced budget seeks to balance both encouragement to participate in the announced auction, but also the need for careful bid making. Finally, the proposed budget-and-fine setup strategy is very simple, which can be easily and efficiently implemented, without incurring further computational burden to nodes.

3) DECISION-MAKING STRATEGY

In order to determine who wins an auction, an auctioneer considers both the bid b_i and *relative tightness* c_i of each node i who has replied to the announced RFB within a given time period (a timeout value is set after which a decision is made). Let $\mathcal{B}$ denote the set of bidders that reply to the announced RFB. The node to which the packet is relayed is based on the outcome of a *preference function* P evaluated on the set $\{(b_i, c_i) | i \in \mathcal{B}\}$. The winner bidder is the one that provides the largest P value, i.e.,

$$\text{auction winner} = \arg\max_{i \in \mathcal{B}} P(b_i, c_i). \qquad (7)$$

Notice that, for a given RFB, $F_n \leq b_i \leq B_n$, and $c_i \leq c_{\max}$, where $c_{\max}$ depends on the largest Δ_i for all $i \in \mathcal{B}$. Therefore, the auctioneer needs to compute c_i for all $i \in \mathcal{B}$ in order to decide the winner. In this work, two types of preference functions are used, based on which *three* different strategies are defined. The first preference function is based on a *hyperplane*, and the second is based on a *Gaussian* function.

a: HYPERPLANE PREFERENCE FUNCTION

the main motivation for a hyperplane as a preference function is its simplicity and low computational complexity. Also, by setting appropriate constant values, the plane can be tilted to reflect a certain weight towards b_i or c_i in the decision-making process. To define the hyperplane, we pick some points of interest and assign specific values to them. For instance, the lowest preference should be given to bidders with $c_i = 0$ and $b_i = B_n$, since these are nodes that charge the most to relay a packet in a very tight condition (no room for mistakes in the forwarding process). Hence, we set $P_n(0, B_n) = 0$. On the other hand, the highest preference should be given to bidders with $c_i = c_{\max}$ and $b_i = 0$, i.e., they have a "surplus" of hops before reaching the deadline (they are less tight), and they relay the packet for free. Other interesting cases are $P_n(0, 0)$, where the bidder is "tight," but it relays for free, and $P_n(B_n, c_{\max})$, where the bid is maximum, but the bidder has the lowest tightness. Hence, if we let $P_n(0, 0) = k_1$ and $P_n(B_n, c_{\max}) = k_2$, we may choose $0 < k_1 < k_2$ to reflect our tendency to favor packet delivery as opposed to increase our budget. The plane that intersects these points define $P_n(b_i, c_i)$, given by

$$P(b_i, c_i) = k_2 \left(\frac{c_i}{c_{\max}}\right) - k_1 \left(\frac{b_i}{B_n}\right) + k_1. \tag{8}$$

Notice that, the input values to the hyperplane are based on the *relative* values $c_i/c_{\max}$ and b_i/B_n. Therefore, this preference function is designed to work with any auction in the network, regardless of the specific RFB and bid values. Figure 3 shows an example of a hyperplane preference function with $B_n = 20$, $c_{\max} = 3$, $k_1 = 2$, and $k_2 = 3$.

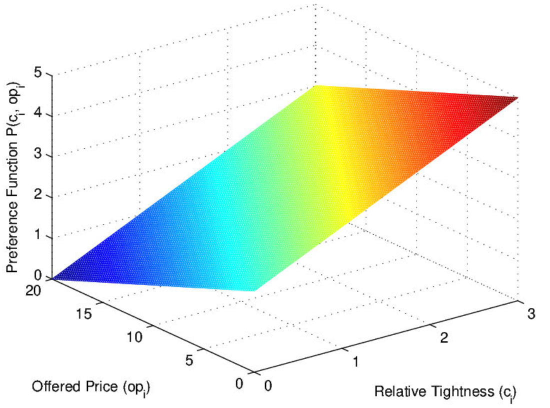


FIGURE 3. Preference function for $B_n = 20$, $c_{\max} = 3$, $k_1 = 2$, and $k_2 = 3$.

b: GAUSSIAN PREFERENCE FUNCTION

For the second preference function, we want to investigate a function that has a global maximum at a given *local operating point*. For that, we use a two-dimensional Gauss-like function because the operating point can be easily set up and we want to have its shape modified according to specific bid and RFB values of an auction (so, not only the operating point, but also the shape of the function is modified in every auction). Hence, $P(b_i, c_i)$ is given by

$$P(b_i, c_i) = \frac{1}{2\pi\sigma_b\sigma_c\sqrt{1-\rho^2}} \exp\left\{-\left[\frac{(b_i - b^*)^2}{2\sigma_b^2(1-\rho^2)} - \frac{2\rho(b_i - b^*)(c_i - c^*)}{2\sigma_b\sigma_c(1-\rho^2)} + \frac{(c_i - c^*)^2}{2\sigma_c^2(1-\rho^2)}\right]\right\}, \tag{9}$$

where (b^*, c^*) is the desired operating point, and σ_b, σ_c, and ρ control the shape of the function. Hence, given a set of $n = |\mathcal{B}|$ bid values, σ_b^2 and σ_c^2 are computed as

$$\sigma_b^2 = \frac{1}{n-1}\sum_{i=1}^{n}\left(b_i - b^*\right)^2, \; \sigma_c^2 = \frac{1}{n-1}\sum_{i=1}^{n}\left(c_i - c^*\right)^2, \tag{10}$$

i.e., σ_b and σ_c express the root-mean-square deviation from the operating point (b^*, c^*). Likewise, borrowing from the definition of correlation,

$$\rho = \frac{\sum_{i=1}^{n}(b_i - b^*)(c_i - c^*)}{(n-1)\sigma_b\sigma_c}, \tag{11}$$

which gives an idea of how "correlated" the sets $\{b_i\}$ and $\{c_i\}$ are, and define the shape of $P(b_i, c_i)$. Figure 4 shows an example of a preference function generated from data drawn from one of the auctions performed in simulations.

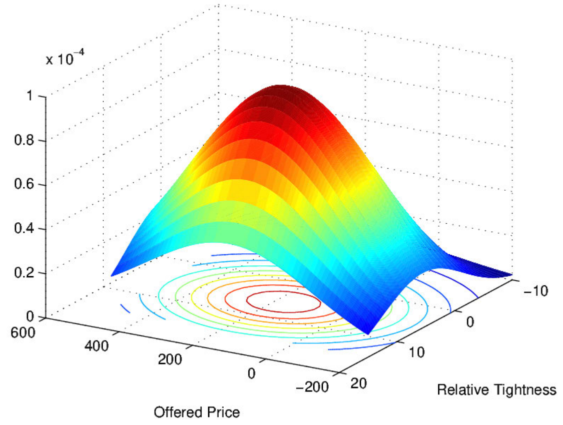


FIGURE 4. Example of Gaussian preference function.

Observe that, for each auction, one operating point is chosen, and all bids and tightness values are compared to the optimal case in that particular auction. The auction winner is the node whose bid and tightness values are closer to the operating point. In simulations, we investigate two operating points.

B. BASELINE STRATEGIES

In this section we define two baseline strategies for purposes of performance evaluation. The first strategy is designed to investigate what happens if the goal of every auctioneer is to deliver the packet to the destination no matter the values of the bids. In this case, the decision-making strategy of every auctioneer is simply to use shortest-path routing, i.e., to always relay the packet to the bidder in the shortest path towards destination, regardless of its bid. In addition, because the value of the bid is not taken into consideration, it is assumed that each node has its own, unknown, *bidding* strategy. To represent the collective behavior of every node having its own bidding strategy, we make every node to bid a value uniformly drawn from the interval

TABLE 1. Table of symbols and definitions used in this work.

Symbol	Definition
H_0	Maximum number of hops the packet can traverse
B_0	Initial budget value
F_0	Initial fine value
B_n	Budget value announced by the n-th auctioneer
F_n	Fine value announced by n-th auctioneer
Δ_i	Tightness function of node i
hc_i	Hop count in the shortest path from i to destination
p_u	Number of hops traversed up to upstream node u
$\mathcal{N}(u)$	Set of nodes that can receive RFB from node u
$\mathcal{S}(u)$	Subset of $\mathcal{N}(u)$ such that $\Delta_i \geq 0\ \forall i \in \mathcal{S}(u)$
$\overline{\Delta}$	Average tightness value $\forall i \in \mathcal{S}(u)$
$\overline{hc}$	Average optimal hop count $\forall i \in \mathcal{S}(u)$
c_n	Relative tightness w.r.t. average tightness: $c_n = \Delta_n/\overline{\Delta}$
$\Delta_{\max}$	Maximum value of Δ_i in $\mathcal{S}(u)$
a_n	Relative tightness w.r.t. maximum: $a_n = \Delta_n/\Delta_{\max}$
$O(\boldsymbol{x})$	Offered bid, where $x = [c_n\ a_n\ B_u\ F_u]$
$\mathcal{B}$	Set of bidders that reply to RFB
$c_{\max}$	Maximum relative tightness in the set $\mathcal{B}$

$[F_u, B_u]$, where F_u and B_u are the fine and budget values announced in the received RFB. Finally, the *budget-and-fine setup* strategy follows the same one defined in Section III-A. Henceforth, this strategy will be referred to as *Shortest Path* strategy.

The other strategy we investigate assumes that every auctioneer always relay the packet to the node whose bid is the lowest among the nodes $i \in \mathcal{B}$. Therefore, with this strategy, we investigate what happens if every auctioneer is greedy, and always want to increase its own budget regardless of packet delivery. Similar to *Shortest Path*, we assume that nodes run different bidding strategies that are collectively represented by random values chosen in $[F_u, B_u]$. Finally, the *budget-and-fine* setup strategy follows the same one defined in Section III-A. Henceforth, this strategy will be referred to as *Lowest Bid* strategy.

Finally, for ease of reference, Figure 5 presents a flowchart describing the overall operation of an arbitrary node that participates in the T2T network, and Table 1 summarizes all variables used in the description of the strategies, along with their corresponding definitions.

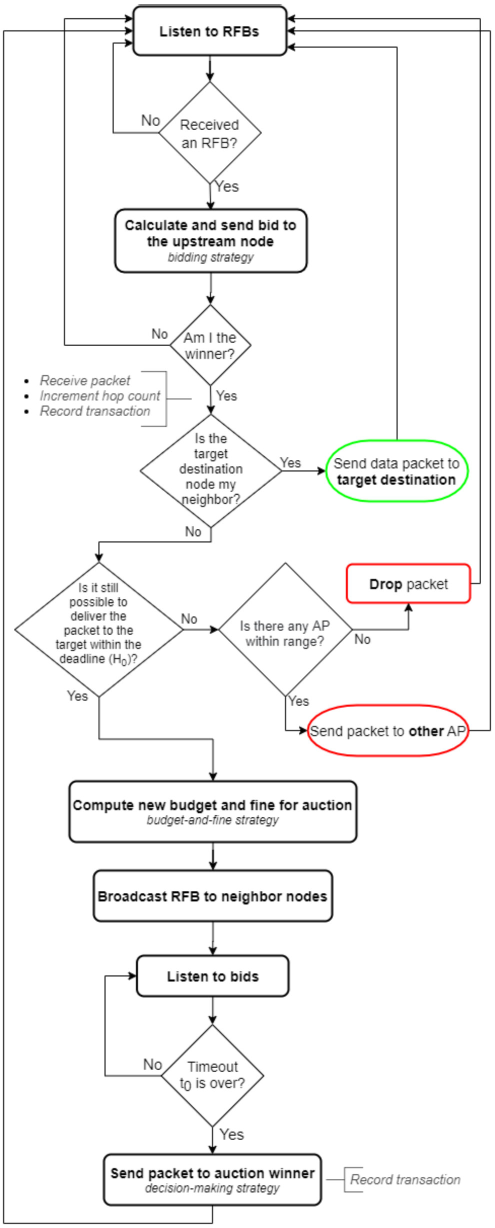

Listen to RFBs
No
Received an RFB?
Yes
Calculate and send bid to the upstream node
bidding strategy
No
Am I the winner?
Yes
• Receive packet
• Increment hop count
• Record transaction
Is the target destination node my neighbor?
Yes
Send data packet to target destination
No
Is it still possible to deliver the packet to the target within the deadline (H0)?
No
Is there any AP within range?
No
Drop packet
Yes
Send packet to other AP
Yes
Compute new budget and fine for auction
budget-and-fine strategy
Broadcast RFB to neighbor nodes
Listen to bids
No
Timeout t0 is over?
Yes
Send packet to auction winner
decision-making strategy
Record transaction

FIGURE 5. Flowchart of operation of a general node in the T2T network.

C. COMMENTS ON ENERGY CONSUMPTION

None of the strategies introduced previously considered the energy level in the batteries of users' devices as an input parameter. Indeed, during certain periods of network operation, some nodes may participate in the auctions more frequently than others as a result of their geographical proximity to a data source or target destination, for instance. Consequently, there can exist a heterogeneous drain of energy battery among devices, which may affect network connectivity and, ultimately, its longevity. Therefore, one might argue that the rate at which devices consume energy should be taken into account not only when a bid is formulated, but also when the auction winner is decided. However, accounting for energy consumption in incentive schemes for packet forwarding is not an easy task.

Previous works on incentive schemes that considered energy battery levels in the computation of payments and routes (e.g., [8]–[10], [13]) have generally relied on a two-step forwarding process: first, the target destination must receive all required information (i.e., energy levels, neighborhood information, etc.) from *every* node in the network in order to compute the best route and associated payments to relay nodes. Then, this information must be sent *back* to the source node to start with the actual data packet transmission over the selected route. Clearly, such solutions not only consume extra energy due to the two-step forwarding process, but they also become practically infeasible in mobile scenarios, since 1) the selected route and payment information may never reach the data source, or 2) there can be significant topology changes that invalidate the selected routes and associated payments. Not surprisingly, the aforementioned works dealt with static or semi-static topologies.

Additionally, we note that the vast majority of works that have considered the battery's energy level as a parameter in their incentive schemes have not addressed battery usage/network lifetime too. For instance, Anderegg and Eidenbenz [9] considered minimum-energy routes, but they did not present any results on battery usage. Zhong *et al.* [10] took into account the nodes' energy levels, but they only focused on investigating the relation between energy consumption and credit gains, i.e., they basically reported that nodes who participate more actively in the network (and earn more credits) are also the ones who spend more energy. However, they did not address how fast their scheme drains the nodes' energy battery and, consequently, the extent of network lifetime. Eidenbenz *et al.* [13] considered energy-efficient paths, but they did not present any results concerning battery usage/energy consumption, while Xu *et al.* [23] considered the remaining energy of each node as a parameter but they did not present any results concerning energy battery usage.

In this paper, we opted for not considering the energy level of users' devices in the auction strategies because there is an inherent trade-off between delivering a packet within a given deadline and saving energy consumption (let alone maximizing profits). In the one hand, the selection of nodes that lead to shortest paths (to satisfy a deadline) may generally incur faster energy depletion of their batteries, especially if such nodes are "popular" due to their location in the network. On the other hand, selecting the nodes that can prolong network lifetime (by saving energy consumption) may likely lead to routes that can make it hard to meet the given deadlines (e.g., by selecting rarely used nodes due to their unfavorable location in the network). Therefore, because our work is based on per-hop auctions, achieving a good compromise among all *three* goals in the long run, i.e., energy efficiency, deadline fulfillment, and profitable (and fair) operation, becomes a very challenging problem.

IV. SIMULATION SCENARIOS

As presented in Section III-A there are many possibilities to set up each sub-strategy in the "tightness strategies" class. Therefore, we focus on *three* specific setups, which are defined according to the chosen *decision-making* strategy and respective parameters. To differentiate them, the following nomenclature is used:

- *Tightness*: this is the tightness strategy based on the hyperplane preference function with parameters $k_1 = 2$, and $k_2 = 3$, i.e., a slightly higher weight is given to the ratio $c_i/c_{\max}$ as opposed to b_i/B_n (packet delivery is considered more important than budget);
- *Gauss*: Gaussian preference function with operating point $(F_u, c_{\max})$, i.e., highest preference is given to the bid that is the closest to the smallest possible value (the announced fine F_u), and whose node has a tightness value closest to $c_{\max}$. This would locally maximize the budget and the likelihood of deliverying the packet within the deadline (surplus of hops before deadline is reached);
- *Gauss*$_1$: Gaussian preference function with operating point $(F_u, 1)$. In this case, the highest preference is given to the bid that is the closest to F_u, but whose bidder has a tightness value equal to the average tightness ($c_n = 1$). This is a more relaxed situation, where the surplus of hops to destination is not considered so critical to make a decision on the auction winner;

The performance of each of the considered strategies (*Tightness*, *Gauss*, *Gauss*$_1$, *Shortest Path*, and *Lowest Bid*) is evaluated via discrete-event simulations based on the ns-3 network simulator [21]. Ten topologies are used with 100 nodes forming the T2T network, and 32 other nodes acting as Wi-Fi access points (APs) to the operator's backbone. All AP nodes are fixed located and evenly spaced surrounding a terrain of 800 m × 800 m, which gives an overall network density of 0.00021 nodes/m^2. Such node density is less than or about the same as the ones found in previous works on this subject: 0.002 nodes/m^2 [11], [22], 0.0004 nodes/m^2 [8], [26], and 0.0002 nodes/m^2 [8]. We chose this node density[2] to strike a balance between connectivity, sparsity, and, more importantly, to insure some level of competition among nodes in announced auctions. After all, the auction strategies cannot be properly evaluated if there is low competition among nodes to relay the data packets. Figure 6 depicts an example of a random topology used in simulations, where the green lines indicate connectivity between nodes (transmission range).

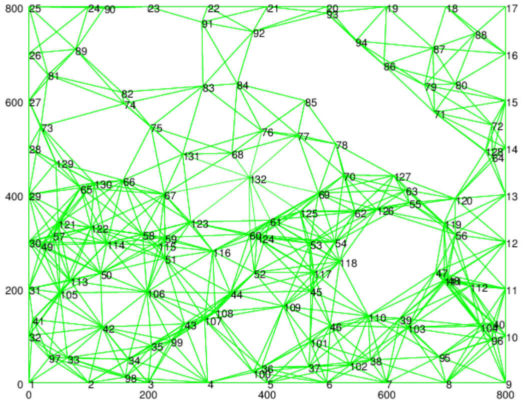


FIGURE 6. Example of random topology used in simulations. The green lines indicate connectivity between nodes based on transmission range.

Two scenarios are investigated: *static* and *mobile*. In the static scenario, one of the topologies is based on a grid of nodes forming the T2T network, while the other topologies are based on nodes randomly placed on the terrain. In the

[2] Incidentally, this node density corresponds to a number of users equivalent to just 1% of the maximum capacity of a theme park as big as Universal Studios Hollywood.

mobile scenario, all nodes move according to the random walk mobility model available in the ns-3 simulator. Three mobile scenarios are investigated, based on three different speeds: 0.5 m/s, 0.75 m/s, and 1.0 m/s. In all three scenarios, nodes change direction every 10 m (randomly). The chosen speeds reflect *walking* behavior, which is an appropriate scenario for per-packet recursive auctions, and also because the investigated strategies rely on the knowledge of network topology (apart from *Lowest Bid*): topology information becomes less reliable as mobility becomes too high. Note that, when an auction happens, it incurs the announcement of the auction (RFB), the wait for the reception of bids from neighbors, the decision on auction winner, and data packet transmission to the corresponding node. Incidentally, most of previous works on incentive schemes for multihop networks have generally considered *static* or *semi-static* networks in order to evaluate their solution (see discussion in Section VI). Few works have dealt with mobility. For instance, Xu *et al.* [23] have considered node speeds within the range [0.5, 2.5] m/s, while Buttyán and Hubaux [26] have considered node speeds within the range [1.0, 3.0] m/s. For topology dissemination among nodes, we use the scheme embedded in the OLSR protocol [27]. We also use OLSR to implement the *Shortest Path* strategy over the topology information. Every simulation has a "warm-up" period of 30 s before any auction happens, during which the nodes start moving around and OLSR operates. This is to allow dissemination of topology information before the beginning of any auction.

As far as traffic generation is concerned, each AP node offloads a total of 50 packets into the T2T network. But, each AP only starts its auctions when its neighbor AP finishes the auction of all 50 packets, i.e., AP nodes transmit consecutively, one after the other. Also, each AP node has a fix destination AP to which all of its 50 packets are addressed. The destination AP is roughly located in the opposite direction of the transmitting AP in the topology, so that the number of hops to destination is maximized (to make the offloading job more challenging). This traffic generation pattern aims to provide a somewhat fair distribution of data flows in the network, while avoiding location-specific interpretation of results (one of the goals of this study is to understand fairness issues between strategies).

The time interval between the issue of requests for bids is 3.0 s. The *auction timeout*, i.e., the time interval that a node waits before deciding for the winner of an RFB is 50 ms. Consequently, the next AP in sequence waits for 160 s before issuing its first RFB (time for offloading all 50 packets plus a guard interval of 10 s). The simulation is over once every AP node finishes offloading all of its packets. This happens after 5,200 seconds of network operation (simulated time), which includes an extra time interval to guarantee the relay of any packet still traveling in the network. Each packet has an initial "deadline" (H_0) of 10 hops, an initial budget (B_0) of 1000, and an initial fine (F_0) of 400 (arbitrary initial values are also used in [4], for instance). Finally, for the MAC- and PHY-layer parameters, all network nodes operate according to the IEEE 802.11g ad hoc mode in the 2.407 GHz frequency channel. All frames are transmitted at 1 Mb/s, and no RTS/CTS frames are used. Energy detection threshold is set to -67.5785 dBm, while the CCA threshold is set to -71.1003 dBm. Transmit power is 16.0206 dBm, which corresponds to a transmit range of 150 m under the Friis large-scale channel propagation model. No small-scale fading was implemented, since we wanted to minimize the occurrence of errors due to channel impairments (and have a better idea of packet delivery by each strategy). But, errors due to large-scale propagation effects (path loss) could still occur, as well as packet collisions, especially with OLSR broadcast messages or simultaneous bids (under CSMA/CA operation, of course).

The strategies are investigated based on four performance metrics: *packet delivery ratio* (PDR), defined as the ratio of the number of packets delivered to destinations to the total number of packets offloaded to the network; the *relative average budget* (RAB), defined as the ratio of the average accumulated budget per node to the initial budget announced by every access point (i.e., $B_0 = 1000$). In this case, we compute a relative value because the initial budget is just a symbolic value. Therefore, it makes more sense to understand the average accumulated budget per node as a *gain* over the announced budget per packet. The other two metrics are *fairness*, defined according to Jain's fairness index [28]

$$\mathcal{J}(x_0, x_1, x_2, \ldots, x_n) = \frac{\left(\sum_{i=0}^{n} x_i\right)^2}{n \sum_{i=0}^{n} x_i^2}, \tag{12}$$

where x_i is the final budget at node i; The idea of this metric is to understand how fair each strategy is regarding budget distribution among nodes. Finally, we investigate the *average number of hops* (ANH) traversed by all packets that are offloaded to the network (successfully transmitted or not).

V. SIMULATION RESULTS

In the following, we present simulation results divided into two groups, according to node mobility: static and mobile topologies.

A. STATIC TOPOLOGIES

Figure 7 presents the results for the relative average budget (RAB) of all strategies. All tightness strategies perform better than *Shortest Path*. In particular, *Tightness* and *Gauss* present the best results, with RAB values of 12.20 and 12.17, respectively, while *Gauss*$_1$ performs slightly worse, with 11.74 RAB, but still better than *Shortest Path* with its 8.96 RAB. This indicates that, in the static scenario, the bell-shaped preference functions (with operating points) are as profitable as the hyperplane preference function. *Tightness* provides a gain of 36.15% over *Shortest Path*, while *Gauss*$_1$ also obtains good performance, with a 31.03% gain over *Shortest Path*. *Lowest Bid* delivers poor performance, since it always picks the node with the smallest bid, regardless of its chances to deliver the packet at destination. These results

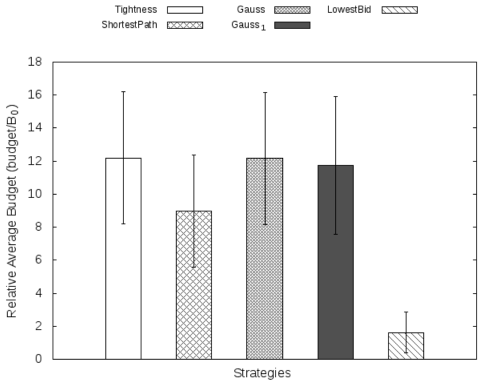


FIGURE 7. **Relative average budget of all strategies under static topologies.**

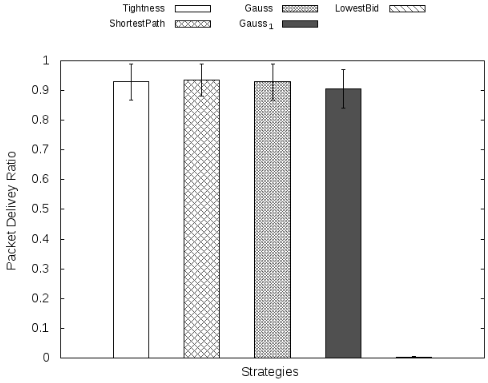


FIGURE 8. **Packet delivery ratio (PDR) under static topologies.**

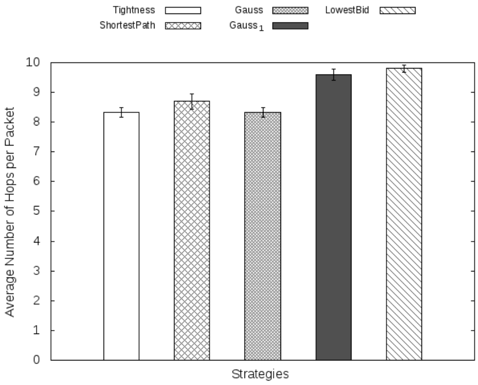


FIGURE 9. **Average number of hops per packet under static topologies.**

indicate that one of the design goals of the tightness strategies was achieved, which is that of incentivizing nodes to join the T2T network by delivering high profits to those that participate in the recursive auctions.

Figure 8 presents the results for packet delivery ratio (PDR). *Shortest Path*, *Tightness* and *Gauss* deliver similar performance, with 0.94, 0.929 and 0.928 PDR, respectively. *Gauss*$_1$ also shows competitive performance, with 0.91 PDR. These results are very important, since they show that it is possible to achieve packet delivery ratios as competitive as those provided by *Shortest Path* while, at the same time, guaranteeing higher RAB values per node. Also, it is interesting to observe the low PDR variability between topologies in all strategies (including the tightness strategies), as indicated by the standard deviation in the graphs. This indicates that, in the static scenario, and compared to budget results, all strategies tend to lead to more stable routes toward destinations (in the sense of PDR within the deadline), while budget accumulation is more dependent on the type of topology. Finally, as expected, *Lowest Bid* performs very poorly, delivering almost no packets to destination. This is because it does not aim to deliver the packets within the deadline, which leads to an excessive number of packet drops.

Figure 9 depicts the average number of hops (ANH) traversed by packets (successfully or not). *Gauss* and *Tightness* deliver the best performance with similar results: 8.33 ANH. Surprisingly, *Shortest Path* performs slightly worse than *Gauss* and *Tightness*, with 8.69 ANH This result suggests that introducing the *deadline* constraint into the bidding and decision process (in terms of number of hops) has a clear benefit to the overall ANH traversed by packets. In the tightness strategies, the nodes themselves encourage (or discourage) the reception of a packet through their bids, which take into account how tight a node is to deliver the packet within the deadline. Curiously, however, the *request-for-bids* strategy used by *Gauss*$_1$ does not provide a good ANH. This is related to the fact that *Gauss*$_1$ uses the *average* tightness $c_n = 1$ in its operating point, as opposed to $c_{\max}$ used by *Gauss*. Thus, *Gauss*$_1$ prefers to relay the packet to a node that is as *tight* as its neighbors (on average), and whose bid is close to F_u. Consequently, it is prone to deliver the packet to someone that is not so likely to deliver the packet to destination. Here, it is important to remember that all nodes use the OLSR protocol, which delivers a *partial* view of the network topology, since the paths are computed over the *multipoint relays* only [27]. Therefore, not all nodes are known to everyone, and some inaccuracies exist on shortest-path computation. Using the tightness information for the bidding and decision process, there is a reassurance of the best path since nodes may have different topology information. As expected, *Lowest Bid* has the poorest performance, with about 9.81 ANH.

Figure 10 shows the results for budget fairness among nodes. *Shortest Path* achieves the best fairness with 0.69, surpassing *Tightness* with 0.579, *Gauss* with 0.578, and *Gauss*$_1$ with 0.54. Again, *Lowest Bid* delivers the worst performance with just 0.18. To understand these results, notice that, when nodes follow tightness-based strategies, the auction winner is generally a node whose bid value is close to the announced fine F_u (one of the goals of the preference functions, as presented in Section III-A). Thus, because a winner node keeps 5% of its bid before setting up its own budget and fine values, budget gains decay along a route towards destination: the nodes close to the transmitting AP get higher gains than those close to the destination AP. This is actually a reasonable policy, since the nodes close to the transmitting AP assume

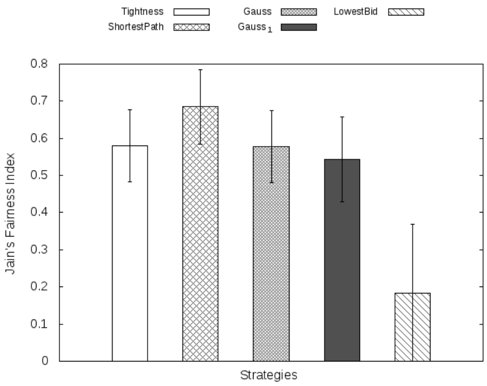


FIGURE 10. Budget fairness under static topologies.

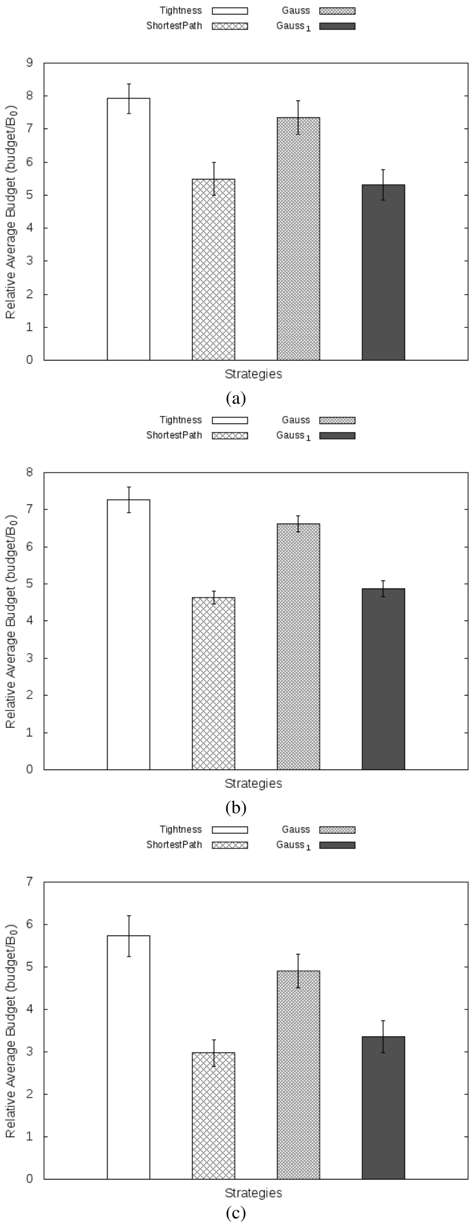
higher risks early on, with unpredictable outcomes, compared to those close to the destination AP, which have a better assessment of the likelihood of packet delivery within the deadline. Under *Shortest Path*, the budget gains still decay towards destination, since the budget-and-fine setup strategy is the same. But, the bids of the nodes are randomly distributed in $[F_u, B_u]$, and the winner bid is always the one on the shortest path towards destination. Consequently, the winning bid is not necessarily close to F_u, and this leads to larger variations on budget gains for different packets towards the same destination AP. This is why *Shortest Path* achieves higher fairness compared to the other strategies.

It is clear from previous results that *Lowest Bid* performs very poorly because it does not aim to deliver the packet within the deadline. Therefore, it performs even worse under mobile topologies. For this reason, in the following, we omit the results for this strategy.

B. MOBILE TOPOLOGIES

Figure 11 shows the results for the *relative average budget* (RAB) under mobility. Compared to the static case, the RAB values of all strategies decrease as mobility increases, as expected. The most significant decay in performance happens with *Gauss*$_1$, whose RAB decays by 54.8% just by starting moving at 0.5 m/s. Also, *Tightness* RAB decays by 35.0%, while *Shortest Path* drops by 38.7%, and *Gauss* by 39.6%. As a result, the difference between *Tightness* and both *Gauss* and *Gauss*$_1$ increase under mobility. In fact, *Tightness* dominates RAB performance in all speed scenarios. At 0.5 m/s, *Tightness* performance is 7.8% better than *Gauss*, 44.3% better than *Shortest Path*, and 49.2% better than *Gauss*$_1$. As mobility increases, all tightness strategies surpass *Shortest Path* (*Gauss*$_1$ is a bit worse than *Shortest Path* at 0.5 m/s). At the speed of 1.0 m/s, *Tightness* RAB decreases to 5.73, which is about 92.93% higher than *Shortest Path* (2.97 RAB), 16.94% higher than *Gauss* (4.90 RAB), and 70.54% better than *Gauss*$_1$ (3.36 RAB).

It is interesting to notice that having a preference function that focus on an operating (target) point does not necessarily deliver the best RAB results under mobility. Instead, the *plane* preference function works best. It is worth noting that both the *parameters* and *shape* of the Gauss-like preference functions depend on the specific bid and tightness values of a given auction. Therefore, it seems that the "best" decisions are too localized, which seems to reflect on the overall performance as mobility increases. In the case of the plane preference function, the parameters k_1 and k_2 are kept fixed in every auction performed in the network (we can interpret the ratios op_i/B_n and $c_i/c_{\max}$ as input variables to the plane). Therefore, the same preference function is applied in every single auction. The plane does not define a specific "operation point," based on which a maximum value can be drawn. It simply picks the node whose bid and tightness values lead to the maximum on the plane preference function.

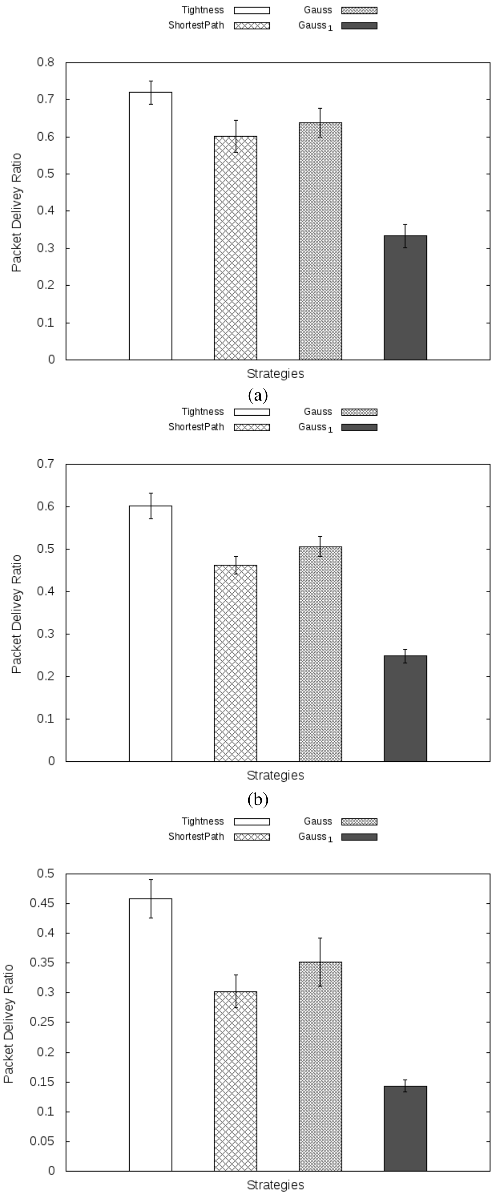


(a)

(b)

(c)

FIGURE 11. Relative average budget per node under mobility. Performance of each strategy at (a) 0.5 m/s. (b) 0.75 m/s. (c) 1.0 m/s.



(a)

(b)

(c)

FIGURE 12. Packet delivery ratio under mobility at different speeds. Performance of each strategy at (a) 0.5 m/s. (b) 0.75 m/s. (c) 1.0 m/s.

The strength of the ''tightness strategies'' under mobility is best appreciated if we look at the results for *packet delivery ratio* (PDR) in Figure 12. Surprisingly, *Tightness* and *Gauss* achieve better PDR than *Shortest Path* and *Gauss*$_1$ when nodes move at all speeds (0.5 m/s, 0.75 m/s and 1 m/s). This is quite interesting, since it means that the ''offered price'' dimension in the preference function has a positive impact on the achievable PDR. When nodes move at 0.5 m/s, *Tightness* achieves a PDR of 71.92%, while *Shortest Path* reaches 60.10%, *Gauss* 63.80%, and *Gauss*$_1$ 33.36%. In other words, *Tightness* and *Gauss* deliver 19.67% and 6.16% more packets than *Shortest Path*, respectively, while *Gauss*$_1$ is 44.49% worse than *Shortest Path*. When nodes move at 0.75 m/s, the performance of all strategies degrades, but *Tightness* and *Gauss* are 30.25% and 9.54% better than *Shortest Path*, respectively. Still, *Shortest Path* is 85.81% better than *Gauss*$_1$. Finally, as nodes move at 1.0 m/s, performance degrades across all strategies, but *Tightness* delivers 51.57% more packets than *Shortest Path*, while *Gauss* becomes 16.41% better than *Shortest Path*. It is important to remember that all the strategies rely on the information provided by the OLSR protocol. Therefore, as mobility increases, the routing tables at nodes become less reliable, and stale topology control information is disseminated on the network, which reflects on routing decisions (*Shortest Path*) and tightness computations.

The PDR results also show that the choice of *operation point* for the bell-shaped preference functions has a clear impact on the final performance. *Gauss* is significantly better than *Gauss*$_1$ in both RAB and PDR metrics. This means that relaying a packet to a neighbor whose tightness (c_i) is closer to the maximum possible value among competitors ($c_{\max}$) is better than relaying the packet to a node with average tightness ($c_i = 1$) (assuming that in both cases the offered price is close to the minimum possible F_u). It is also worth noting that, in spite of the lower PDR values obtained at 1.0 m/s (compared to *Shortest Path*), *Gauss*$_1$ deliver higher RAB value than *Shortest Path* at this speed (see Figure 11(c)). This means that, although *Gauss*$_1$ have delivered less packets, the nodes ended up accumulating higher profits. Under high mobility, one should expect a higher reluctance from nodes to participate in the T2T network, because of the likelihood of higher losses in a less predictable and stable environment. Therefore, it is reasonable to trade off PDR with RAB, since the nodes are assuming higher risks (this is certainly not a favorable situation to operators, but it is definitely better to participant nodes in the T2T network—the prospect of some profit under a harsh environment).

Figure 13 presents the results for the average number of hops (ANH) traversed by successful packets in each strategy. *Tightness*, *Gauss*, and *Shortest Path* present similar results, with *Tightness* delivering the best performance across all speeds. In spite of mobility, *Tightness* manages to deliver packets within 1 to 1.5 hops away from the maximum number of hops allowed to destination (on average). *Gauss*$_1$ deviates the most from other strategies, delivering slightly higher ANH values, especially at low mobility (similar to the static scenario). As mobility increases, *Tightness* ANH values increase from 8.88 (at 0.5 m/s) to 9.11 (at 1.0 m/s), a 2.6% variation. *Gauss* ANH values also increase with speed: from 9.07 (at 0.5 m/s) to 9.26 (at 1.0 m/s), which is a 2.09% variation. Gauss$_1$, however, is the only strategy whose ANH values *decrease* as mobility increases: from 9.89 (at 0.5 m/s) to 9.61 (at 1.0 m/s), a variation of −2.83%.

Lastly, Figure 14 presents the results on budget *fairness* among nodes under mobility. Following the results on the static scenario, *Shortest Path* has the best performance at speeds of 0.5 m/s and 0.75 m/s, where it achieves an average fairness of 0.73 and 0.74 respectively. This is roughly 20% better than *Gauss* and *Tightness* in both scenarios. This is

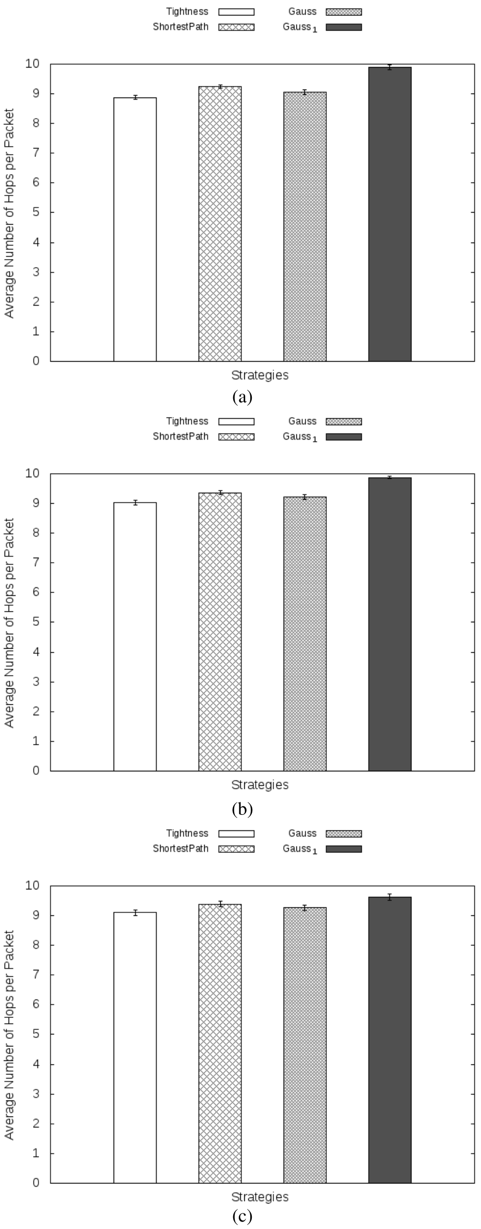


FIGURE 13. Average number of hops per packet. Performance of each strategy at (a) 0.5 m/s. (b) 0.75 m/s. (c) 1.0 m/s.

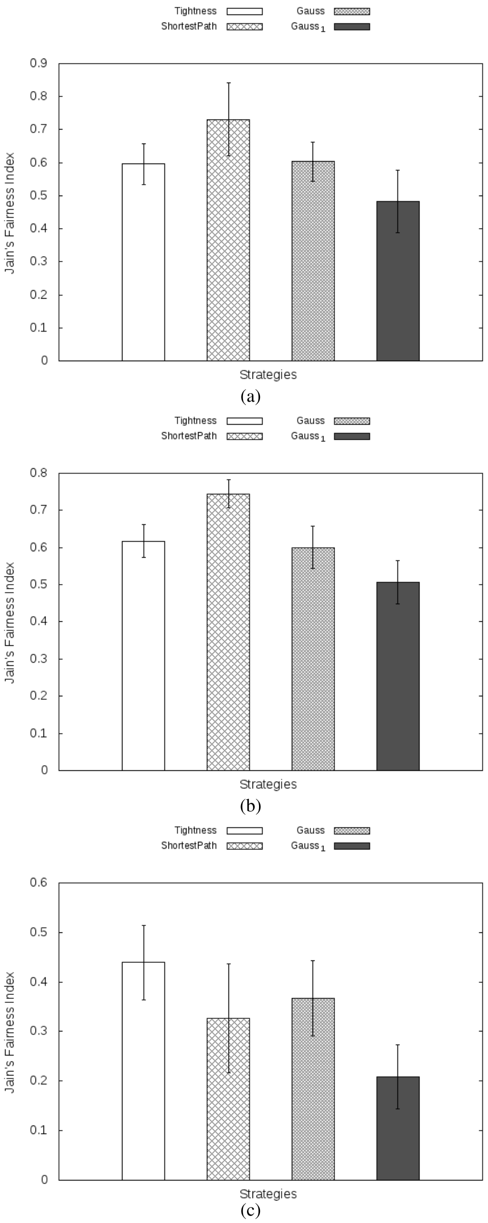


FIGURE 14. Budget fairness under mobility. Performance of each strategy at (a) 0.5 m/s. (b) 0.75 m/s. (c) 1.0 m/s.

because, when nodes follow the tightness strategies, the auction winners are those that bid values closer to the announced fine F_u, as dictated by the bidding and decision strategies presented in Section III-A. Thus, because the winner node keeps 5% of its bid before setting up its own budget and fine values (see the Budget-and-Fine set up strategy), the accumulated budget drops fast as a packet moves forward along a route (nodes that are closer to the AP gets more, since they assume a task of high risk early on, of unpredictable outcome towards destination, while nodes that are closer to destination have a much better idea of the possible success in the forwarding of a packet. Hence, they should be less rewarded, comparatively). Under *Shortest Path*, however, the auction participants offer random values within the interval $[F_u, B_u]$, and the winner node is always the one on the shortest path towards destination. This leads to a higher variation of accumulated budget along a route. Note that, nodes still obey the Budget-and-Fine set up strategy under *Shortest Path*, but the winner is no longer the one who bids a value close to the announced fine.

As far as resilience to mobility is concerned, *Tightness* presents the best performance, since its fairness varies by only 26.7% as speed changes from 0.5 m/s to 1.0 m/s. *Gauss* comes next, with a 38.3% variation, while *Shortest Path* has a variation of 54.8%, and $Gauss_1$ undergoes a 56.2% variation. In fact, the performance decay of *Shortest Path* and $Gauss_1$

is accentuated when speed changes from 0.75 m/s to 1.0 m/s. Such a significant drop in fairness is probably due to the low PDR of *Shortest Path* in this scenario. At the speed of 1.0 m/s, *Tightness* delivers the best performance, with an average fairness of 0.44, against 0.37 of *Gauss*, 0.33 of *Shortest Path*, and 0.21 of $Gauss_1$.

VI. RELATED WORK

One of the key issues to fully realize cooperative multi-hop communications is how to incentivize users to let their devices work as packet relays to benefit others. Buttyán and Hubaux [12] have tackled this problem earlier by introducing a virtual currency named *nuglets*, by which nodes can pay other nodes to forward their packets. According to their ''packet purse'' model, a source node needs to load a packet with sufficient *nuglets* to reach its destination. Each forwarding node gets some *nuglets* from the packet in order to cover its forwarding costs. A packet is discarded if it does not contain enough *nuglets* to be forwarded. To control the number of *nuglets* taken out from a packet, a sealed bid second price auction is run at each hop: each bidder determines the price for which it is willing to forward the packet, and sends it to the forwarding node in a sealed form. The price is obtained from two utility functions that are based on battery level and number of *nuglets* in the node. It is assumed that a bidding node has no information about the total number of bidders participating in the auction, and the auction winner is always the one with the lowest bid. Then, the forwarding node puts the value of the second lowest bid in the packet and sends it to the winner. For proper operation, this scheme requires the use of routing algorithms that allow nodes to have multiple entries in their routing tables with different next hops to the same destination (e.g., TORA [29]). The performance evaluation of the proposed approach has focused on packet delivery ratio for different battery energy levels over static topologies. Also, in spite of targeting cooperation, the fairness in distribution of *nuglets* among nodes has not been evaluated, and no indication has been given about the achievable average gain of *nuglets* per node. Finally, this solution did not target packet delivery within a given deadline. Later, they proposed a set of deterministic rules to decide whether a node forwards a packet without relying on auctions at each hop [26]. Their mechanism is based on a counter of *nuglets* that decreases if the node sends its own packet, and increases if the node forwards a packet. The counter must remain positive for the node to send its own packets.

Anderegg and Eidenbenz [9] have proposed Ad Hoc-VCG, a reactive routing protocol for ad hoc networks where nodes are selfish and require payments to forward data. The protocol is designed to achieve truthfulness (also known as *incentive compatibility*), i.e., the nodes reveal the true cost to forward data (measured in terms of energy to relay a packet) and cost efficiency. It consists of two phases: route discovery and data transmission. During route discovery, a minimum energy route is computed from source to destination based on a weighted graph informed to the destination node. The edge weights represent the payments a node has to receive if it transmits a packet along that edge. The destination node computes the shortest path to it and all payments needed to be made. Then, it sends this information back to the source node. In the data transmission phase, the source node sends the data packets with the electronic payments over the shortest path. For analysis, this work has focused on proving truthfulness and cost efficiency mathematically, and provided some results on experiments with random static topologies to analyze overpayment. However, the protocol has not been analyzed under mobility, and its performance has not been evaluated regarding packet delivery ratio and individual node profit gains (as well as fairness in profit distribution). Similar to [12], the protocol does not consider packet delivery under a given deadline, and its route discovery phase may stall network operation if routing paths change frequently, as pointed by the authors themselves.

Zhong *et al.* [10] have showed that Ad-Hoc VCG is flawed because it assumes that the transmitter knows the link cost beforehand (i.e., the energy to relay a packet) when, in reality, it needs the receiver's feedback to estimate it. Hence, they have showed that Ad-Hoc VCG does not handle cheating appropriately and, consequently, it does not preserve incentive compatibility. More importantly, they have showed that there does not exist a *forwarding-dominant* protocol in wireless ad-hoc networks. This means that there does not exist a protocol implementing both routing and forwarding such that, under the protocol, nodes always forward packets, and that following the protocol is a *dominant action*, i.e., no matter what other nodes do, following the protocol always brings the maximum utility. Based on this result, they have proposed the Corsac on-demand routing protocol, which adheres to weaker requirements than those for a forwarding-dominant protocol, and uses cryptographic techniques to prevent cheating. Under Corsac, a set of test signals with increasing power levels (and associated costs) are sent between neighbors until they reach the session destination. After receiving all link costs, the session destination chooses the minimum power level for each link and computes the lowest cost path (LCP) from the source using Dijkstra's algorithm. The LCP is sent back to the source for the data transmission phase, along with the payment information for each node in the LCP, which is computed as a function of the link costs. Corsac assumes a semi-static topology, ignores control overhead, and makes assumptions on network connectivity.

Eidenbenz *et al.* [13] have proposed the COMMIT protocol, which improves over Ad-Hoc VCG and Corsac by allowing the source of a session to act strategically, i.e., the source of a session is no longer obligated to pay whatever the destination decides, and it can refuse a proposed path if, for instance, the cost is too high. COMMIT relies on the existence of an underlying periodic topology control protocol to simplify the routing protocol and to reduce its message complexity. It focuses on promoting cooperation for the establishment of a path and packet forwarding, while it is assumed that all nodes cooperate in the execution of the topology control algorithm.

As opposed to Ad-Hoc VCG and Corsac [9], [10], the costs used to compute routes are associated to nodes, and not links. On the other hand, similar to Ad-Hoc VCG and Corsac, the destination node is responsible for computing the lowest-cost path and the payments to the source and intermediate nodes. It also assumes that network topology is 2-connected, i.e., there exist at least two node-disjoint paths from any node to the destination. Moreover, it assumes no link failures result from node mobility during the route discovery phase and subsequent data session, before execution of another round of its topology control protocol. In fact, routes of the data sessions have to be recomputed from scratch after each round of execution of the topology control protocol. It is also assumed that nodes are willing to forward control packets because of potentially large payoff. Both COMMIT and Corsac assume that power levels are enough for communication, and disregard the impact of interference between nodes and channel errors. In spite of the theoretical results, the paper does not present any performance evaluations, which makes it hard to understand its actual limitations and performance.

Su *et al.* [14] have proposed the use of generalized second price (GSP) auction in multi-path routing with selfish nodes. They have adopted the Dynamic Source Routing (DSR) protocol [30] for route discovery and maintenance and, therefore, they have assumed that nodes cooperate in the running of an underlying routing protocol. Their focus is on the formulation of the payment scheme to the nodes, which is supposed to produce less over-payment than VCG-based schemes. It is assumed that each intermediate node incurs a per-packet cost to forward traffic, and this cost is private to the node. Each node's bid is equal to the cost of the outgoing link. Then, each node places an encrypted version of its bid in the route request (RREQ) message, while the route reply message (RREP) carries the network graph constructed at destination based on the information gathered from the received RREQ messages. Once the source receives the RREP messages, it determines the fraction of traffic that must go through each of the detected m node-disjoint least cost paths (LCP) by solving an optimization problem that maximizes each node's utility and minimizes overall system cost, subject to certain constraints given by a set of policies. In their analysis, the cost of each link (i.e., the bid of each node) is randomly chosen and, therefore, does not follow a specific formulation. Also, the paper only considers static topologies and, therefore, does not take into account the mobility of nodes and, consequently, route partitions. In other words, their scheme relies on the assumption that costs and node-disjoint paths are kept the same throughout the network operation.

Zhou *et al.* [16] have proposed the optimal auction-based multipath routing (OAMR) for *wired* networks where nodes are selfish to carry other nodes' flows. Hence, the source node of any source-destination (SD) pair must pay for the bandwidth provided by the intermediate nodes to carry its flows. The bandwidth from the links on the transmission paths are auctioned off to meet the communication requirements of the SD pairs, and traffic is scheduled in such a way that cost is minimized. To solve the problem, OAMR requires a control center in the network that collects routing requests with corresponding bandwidth demands to solve the optimization problem in a batching fashion. Because of its high complexity and computational time, the authors have also proposed a sequential auction-based multipath routing (SAMR) scheme where bandwidth is auctioned off sequentially along the transmission path.

Khairullah and Chatterjee [11] have looked at routing on multi-channel cognitive networks based on combinatorial auctions. In their model, secondary users (SUs) buy some bandwidth from primary users (PUs) in order to trade it among themselves for purposes of packet routing via repeated bidding over multiple hops. At each hop, a transmitter (buyer) picks the receiver (seller) who is able to sustain the bit rate requirement and offers the minimum price to execute the forwarding task. Each seller computes its bid based on the number of channels it needs to combine to sustain the buyer's bit rate requirement. For that, the seller computes the capacity of each available channel based on the measured signal-to-interference and noise ratio (SINR). Since there can be multiple bundles of channels that satisfy the bit rate requirement, the seller offers the one with minimum price. The total price of each route is accumulated through the repeated bidding process over all hops. The route with the minimum payment is chosen as the optimal solution that satisfies the sender's bit rate requirement. Consequently, it requires the knowledge of all auction outcomes, all the way to the destination, before the data packet is actually transmitted. Note that it requires not only the knowledge of network topology, but also the SINR at each available channel, at each receiver, at each hop. Because of that, the proposed algorithm assumes static network topologies. Additionally, the assigned price to each channel is treated as a given, from previous negotiations with the PU (not treated in the paper). Therefore, the price of each bid is simply the sum of pre-assigned prices of the chosen bundle of channels.

Recently, Koutsopoulos *et al.* [15] have considered multi-hop device-to-device (D2D) communications where social network ties are leveraged by MNOs to achieve efficient data transport. Their work is based on the D2D paradigm of 4G+ technologies, where end-to-end path formation and resource allocation (e.g., spectrum management) are centrally controlled by the operator. However, data forwarding decisions are left to the user, whose willingness to relay data packets (i.e., the user benefit) is related to the strength of his social ties. The problem is modeled as a constrained minimum-cost problem on the communication graph, where the constraints arise from the delivery probability derived from the social network graph. We note that, although social ties may increase one's willingness to forward the packets, a scheme for actual compensation (e.g., discounts, credits, etc.) should supplement this approach, since regardless of its social ties, all users will have to donate part of their precious resources (battery life, storage, bandwidth, etc.) to the task of data offloading.

Along the lines of leveraging social interactions for purposes of packet forwarding, Nunes *et al.* [17] have also investigated D2D multi-hop communication through social group meetings. They have argued that social-based solutions like Bubble Rap [31], and similar ideas, have the drawback of relying on the community structure of mobile social networks. Such communities are computationally expensive to detect, especially in distributed environments, and they depend too much on scenario-specific parameter tuning. Their proposal, GROUPS-NET, is based on packet forwarding through the most probable ''group-to-group'' path, where a group is defined as a group of people who are together, in space and time, for some reason. In a group meeting the message can be transmitted to all nodes. However, the message must be forwarded to the next group and so on, until it reaches a group to which the destination belongs. Their solution relies on a centralized control plane at the base station, who computes the most probable group-to-group path *P* and returns it to the source node. Therefore, such a solution (and similar ones) are not in the spirit of the one we propose here, since they assume cooperation among nodes upfront.

Xu *et al.* [23] treated the relay of ''bundles'' in opportunistic networks based on the store-carry-forward paradigm. They adopted a single-copy mechanism, where only one node has a copy of the message at any time. To promote cooperation, a bargain game is proposed for every time a node encounters a potential relay. But, the bargain game can happen only between two nodes: the buyer (sender) and the potential seller (relay), i.e., no concurrent competition among multiple relays is allowed. They assume the existence of a Credit Clearance Center (CCC) connected to the Internet to manage the virtual currency. Each node registers itself to the CCC and gets a digitally-signed receipt for each transaction of relay service, which is submitted to the CCC. A node only gets paid after the destination receives the bundle and submits an ACK to the CCC. In spite of the promising performance, this work does not specify how nodes can recognize relays along the time, i.e., no specification is given about what types of messages are exchanged (and when) for the nodes to announce themselves and perform a bargain through the alternative wireless interface (e.g., Bluetooth).

Mai *et al.* [4] propose the ''virtual bank with movement prediction'' (VBMP), which introduces slight modifications to the bargain game model previously discussed [23] for opportunistic networks. In particular, VBMP makes use of GPS-based locations to estimate the encounter probabilities between nodes. Hence, based on each neighbor's encounter probability with the destination node and their expected revenue with the given transaction, the source node either waits to transmit the information directly to the destination node or to relay it to a subset of neighbors (as opposed to a single relay as in [23]). The encounter prediction system assumes a constant speed for the nodes, and disregards the impact of channel propagation effects.

Feng *et al.* [22] proposed the incentive compatible multiple-copy packet forwarding (ICMPF), which exploits an evolutionary game framework to model the interactions among nodes in an opportunistic network. Their work assumes that an ''attraction point'' (AP) transfers multiple copies of a data packet to mobile nodes to increase the likelihood of successful delivered at destination. The nodes are divided into different classes determined by social ties and similar characteristics. Any node that receives a packet decides to forward it or drop it according to a payoff function that depends on the value of a virtual currency, a meeting probability between nodes, and the cost of a packet transmission. A credit clearance center (CCC) [23] manages the virtual currency, and a node is only paid if its copy is the first one to arrive at destination (i.e., all the other relays who deliver the same copy later are not rewarded and have their resources wasted). Unfortunately, the meeting probability used in the cost function assumes a Poisson process of known parameter, which is not a realistic assumption under arbitrary mobility patterns. Moreover, the existence of an evolutionary stable strategy is proved to networks with only 3 hops, and its solution requires the knowledge of all utility functions and exact actions adopted by all nodes in the network, which leads to huge communication overhead, as pointed out by the authors themselves. Because of that, an empirical distributed solution is also proposed that still relies on the aforementioned Poisson assumptions.

In the context of delay-tolerant networks, Seregina *et al.* [32] considered a network of two hops, where source and destination are fixed, while the relays are mobile,. The source node delivers multiple copies of a packet to relays who approach it. The relays do not seek profit, and they only accept the message if the reward proposed by the source offsets the expected cost to deliver the message, as estimated by the relays themselves when they meet the source. Only the relay who first delivers the message to destination receives the reward. Hence, every relay computes the average delivery cost by estimating the probability of successful delivery at destination. Such probability is dependent on the information the source provides at the time they first meet. Three static strategies are proposed based on whether the source tells the relay how many other relays have already received the message and at what times (i.e., by combining the availability or not of these information). Explicit expressions for the probability of successful delivery are derived for the particular case of exponential inter-contact times.

Umar *et al.* [8] addressed cooperation in wireless sensor networks where the selfish behavior of nodes can degrade end-to-end delay and lead to unfair energy consumption. A base station (BS) is responsible for distributing ''reward scores'' (RSc) to sensor nodes (i.e., values of a virtual currency) that are used to pay for the forwarding service of other sensors. The BS distributes RSc to nodes according to specific attributes computed for each node. Two routing protocols are employed concurrently: Dynamic Source Routing (DSR) [30], for the basic routing functionality, and OLSR [27] for control message dissemination and delivery of ''scores'' to nodes. To transfer data, a data source broadcasts a forward request, which is followed by the reception of ''bargaining scores'' (BSc) computed from ''individual bargain scores'' (IBSc) accumulated in each path. The source node always picks the neighbor with the smallest BSc. This means that the data transfer happens only after gathering all BScs from all possible paths. Moreover, the bargaining may run several times (a Rubinstein-Stahl game is employed), that consumes resources and decreases the benefit of involved parties at each run. Nodes are assumed to store their neighbors' location, energy levels, and demand scores, which imposes a significant control overhead. Also, because the definition of a node's neighbors is based on plain Euclidean distance, no channel propagation effects are considered for link establishment. Given such constraints, the scheme is proposed for static WSNs only.

VII. CONCLUSION AND FUTURE WORK

This paper presented a comprehensive performance evaluation of the so-called *Tightness* strategy for cooperative routing and relay of messages ''on-the-go,'' via per-hop reversed packet auctions. At each hop, the sender of a data packet (buyer) asks for bids from potential relays (sellers), according to a budget value associated to the data packet, through which the auction winner gets paid and can pay for others in subsequent auctions. In addition to the budget, a fine is also announced in every auction, which must be paid by all relay nodes of a given packet if it is not delivered within the announced deadline (expressed in number of hops). The *Tightness* strategy uses the idea of estimating how ''tight'' a node is to fulfill the job of delivering a packet to its destination within the announced deadline. In other words, a node estimates how much ''room'' it has (with respect to the deadline) to absorb eventual bad forwarding decisions resulted from the unpredictable outcomes of other downstream auctions. For tightness computations, the nodes in the T2T network rely on the sharing of topology information only. The ''tightness'' concept was used in the design of the bidding and decision-making sub-strategies, which take as input parameters the offered price and the relative ''tightness'' of the nodes.

The performance of the *Tightness* strategy was investigated for two specific preference functions used in the decision-making sub-strategy, according to different operating points (leading to three preference functions). Both static and mobile scenarios were investigated, for different node speeds, assuming channel errors and realistic MAC activity (IEEE 802.11). Two baseline strategies were also investigated for purposes of performance comparison: one that prioritizes packet delivery over budget gains (using shortest-path routing), and a greedy one, that always pick the highest bid regardless of packet delivery within the deadline. All strategies were evaluated according to packet delivery ratio, average budget per node, fairness on budget sharing fairness, and average number of hops to destination.

The presented results have shown that, apart from the *Lowest Bid* strategy, which delivered very low packet delivery ratios, all strategies proved to be very suitable for T2T data offloading under recursive auctions. Overall, they have provided consistent and positive results across all performance metrics, in both static and mobile scenarios. According to the results, two of the proposed variations of the *Tightness* strategy proved to be more effective than simply using shortest-path routing without taking into account the nodes' bids in the decision-making sub-strategy. In particular, the preference functions *Hyperplane* and *Gaussian* (the one that prioritizes the lowest bid with the highest relative tightness) performed better than *Shortest Path* with respect to average budget per node, average number of hops to destination, and packet delivery ratio, especially under mobility. This result indicates that the use of the tightness concept in both bidding and decision-making sub-strategies is beneficial for cooperative behavior and better overall performance of T2T offloading under recursive auctions. This happens because the nodes who perceive a ''tight'' condition to deliver a packet within the announced deadline discourage the auctioneer from choosing them by bidding high values. As far as fairness in budget distribution among nodes is concerned, the *Tightness* strategies delivered slightly lower results than *Shortest Path* due to a higher variation of accumulated budget along a route. However, the *Tightness* strategies presented lower fairness variation across different mobility scenarios.

We envisage many future directions for this work. One such possibility is to allow multiple winners per auction (e.g., by selecting the best l choices among m bids), so that multiple replicas of the same data packet can follow different paths to the target destination in order to increase the packet delivery ratio. In this case, the sharing of the budget value among the l auction winners should be addressed carefully, so that enough budget remains to be shared among nodes over different paths. Different strategies could be devised based on the ranking of auction bidders with respect to the likelihood of delivering the packet to the destination. Related to that, applicable fines should also be treated: for instance, what happens if multiple copies of the same data packet arrive at destination, within the deadline, but at different time instants due to the different paths taken? Should the nodes pertaining to routes delivering late copies (within the deadline) pay a fine or be rewarded somehow?

Another challenging research direction is the consideration of the nodes' battery energy levels as an input parameter for the auction strategies. As discussed before, improving the overall energy consumption (i.e., network lifetime) while satisfying a given packet delivery deadline not only increases the dimension of the decision problem, but also imposes a significant trade-off that demands careful design of the preference (utility) functions. Moreover, considering the ''on-the-go'' routing nature of the proposed auction mechanism, it is unclear whether the local knowledge of the bidders' battery energy levels, in a specific auction, can lead to better (energy-wise) routes to the target destination. If not, how to ensure timely and proper broadcast of the batteries' energy levels throughout the network?

The adoption of delay-tolerant principles to scenarios where the packet delivery deadline can be relaxed (or traded somehow) so that network partitions can be better handled is also a venue of future research, along with hybrid operations based on that principle, i.e., different incentive schemes could be used based on the need to use delay-tolerant mechanisms at a given moment. Also, the design of appropriate (or optimal) values for the budget B_0, fine F_0, and deadline H_0 are of interest, especially if a specific form of incentive scheme is defined by the application, considering the actual nature of the virtual currency (i.e., monthly discounts, credits on mobile data plans, etc.), and considering constraints over the possible range of values adopted by the operator.

Last, but not least, a challenging problem is a mathematical analysis on whether the Tightness strategy converges to optimal paths and, if so, under what conditions this is satisfied, so that one can understand the scope of its application across different network scenarios. For this purpose, game-theoretic approaches could be developed based on previous works discussed in Section VI, or by looking at tools such as non-classic algebra [33], [34] which might help studying the strategy's convergence properties and optimal behavior. In any case, a key issue for analysis is the mobility of nodes. As previously discussed, most of previous works have dealt with static or semi-static networks because they rely on a two-step forwarding process: first all needed information (e.g., link costs, topology, etc.) is transferred to the destination node, who decides about the least-cost path(s) and associated payments. Then, the selected route, and associated payments, are sent back to the source node, who start the actual data packet transmission and payments. In such cases, one could possibly investigate the convergence and optimality of selected paths because all needed information is known at destination. In our case, however, an auctioneer does not know, in advance, all possible bidders of any auction over any path (and negotiated budget, fine, and bid values). In particular, due to mobility, an auctioneer cannot anticipate who will participate in a given auction down a possible path, which makes the problem particularly difficult to handle.

REFERENCES

[1] H. Luo, X. Meng, R. Ramjee, P. Sinha, and L. Li, "The design and evaluation of unified cellular and ad-hoc networks," *IEEE Trans. Mobile Comput.*, vol. 6, no. 9, pp. 1060–1074, Sep. 2007.

[2] F. Rebecchi, M. Dias de Amorim, V. Conan, A. Passarella, R. Bruno, and M. Conti, "Data offloading techniques in cellular networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 2, pp. 580–603, 2nd Quart., 2015.

[3] N. Jiang, L. Guo, J. Li, M. Ren, S. Cheng, and X. Guo, "Data dissemination protocols based on opportunistic sharing for data offloading in mobile social networks," in *Proc. IEEE 22nd Int. Conf. Parallel Distrib. Syst. (ICPADS)*, Dec. 2016, pp. 705–712.

[4] L. Mai, D. Pan, H. Song, and C. Wang, "A T2T-based offloading method: Virtual bank with movement prediction," *IEEE Access*, vol. 6, pp. 16408–16422, 2018.

[5] D. Xu, Y. Li, X. Chen, J. Li, P. Hui, S. Chen, and J. Crowcroft, "A survey of opportunistic offloading," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 3, pp. 2198–2236, 3rd Quart., 2018.

[6] E. Wang, Y. Yang, J. Wu, and W. Liu, "Phone-to-Phone communication utilizing WiFi hotspot in energy-constrained pocket switched networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 10, pp. 8578–8590, Oct. 2016.

[7] F. S. Shaikh and R. Wismuller, "Routing in multi-hop cellular device-to-device (D2D) networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 2622–2657, Jun. 2018.

[8] M. M. Umar, S. Khan, R. Ahmad, and D. Singh, "Game theoretic reward based adaptive data communication in wireless sensor networks," *IEEE Access*, vol. 6, pp. 28073–28084, 2018.

[9] L. Anderegg and S. Eidenbenz, "Ad hoc-VCG: A truthful and cost-efficient routing protocol for mobile ad hoc networks with selfish agents," in *Proc. 9th Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*, 2003, pp. 245–259.

[10] S. Zhong, L. E. Li, Y. G. Liu, and Y. R. Yang, "On designing incentive-compatible routing and forwarding protocols in wireless ad-hoc networks," *Wireless Netw.*, vol. 13, no. 6, pp. 799–816, Dec. 2007.

[11] E. F. Khairullah and M. Chatterjee, "Combinatorial auction based routing in multi-channel cognitive radio networks," in *Proc. IEEE 25th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Sep. 2014, pp. 1217–1222.

[12] L. Buttyán and J.-P. Hubaux, "Nuglets: A virtual currency to stimulate cooperation in self-organized mobile ad hoc networks," EPFL, Lausanne, Switzerland, Tech. Rep. DSC/2001/001, Jan. 2001.

[13] S. Eidenbenz, G. Resta, and P. Santi, "The COMMIT protocol for truthful and cost-efficient routing in ad hoc networks with selfish nodes," *IEEE Trans. Mobile Comput.*, vol. 7, no. 1, pp. 19–33, Jan. 2008.

[14] X. Su, S. Chan, and G. Peng, "Generalized second price auction in multi-path routing with selfish nodes," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Nov. 2009, pp. 1–6.

[15] I. Koutsopoulos, E. Noutsi, and G. Iosifidis, "Dijkstra goes social: Social-graph-assisted routing in next generation wireless networks," in *Proc. Eur. Wireless*, 2014, pp. 1–7.

[16] H. Zhou, K.-C. Leung, and V. O. K. Li, "Auction-based schemes for multipath routing in selfish networks," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2013, pp. 1956–1961.

[17] I. O. Nunes, P. O. S. Vaz de Melo, and A. A. F. Loureiro, "Leveraging D2D multihop communication through social group meeting awareness," *IEEE Wireless Commun.*, vol. 23, no. 4, pp. 12–19, Aug. 2016.

[18] G. B. Kalejaiye, J. A. Rondina, L. V. Albuquerque, T. L. Pereira, L. F. Campos, R. A. Melo, D. S. Mascarenhas, and M. M. Carvalho, "Mobile offloading in wireless ad hoc networks: The tightness strategy," *SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 3, pp. 96–102, Jul. 2014.

[19] E. Baccelli, F. Juraschek, O. Hahm, T. C. Schmidt, H. Will, and M. Wählisch, "The MANIAC challenge at IETF 87," *IETF Journal*, vol. 9, no. 2, pp. 27–29, Nov. 2013.

[20] E. Baccelli, F. Juraschek, O. Hahm, T. C. Schmidt, H. Will, and M. Wählisch, "Proceedings of the 3rd MANIAC challenge, Berlin, Germany, July 27–28, 2013," 2014, *arXiv:1401.1163*. [Online]. Available: http://arxiv.org/abs/1401.1163

[21] *The NS-3 Network Simulator*. Accessed: Oct. 15, 2020. [Online]. Available: http://www.nsnam.org

[22] L. Feng, Q. Yang, and K. S. Kwak, "Incentive-compatible packet forwarding in mobile social networks via evolutionary game theory," *IEEE Access*, vol. 5, pp. 13557–13569, 2017.

[23] Q. Xu, Z. Su, and S. Guo, "A game theoretical incentive scheme for relay selection services in mobile social networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 8, pp. 6692–6702, Aug. 2016.

[24] M. E. Mahmoud and X. Shen, "PIS: A practical incentive system for multihop wireless networks," *IEEE Trans. Veh. Technol.*, vol. 59, no. 8, pp. 4012–4025, Oct. 2010.

[25] F. Wu, T. Chen, S. Zhong, C. Qiao, and G. Chen, "A bargaining-based approach for incentive-compatible message forwarding in opportunistic networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2012, pp. 789–793.

[26] L. Buttyán and J.-P. Hubaux, "Stimulating cooperation in self-organizing mobile ad hoc networks," *Mobile Netw. Appl.*, vol. 8, no. 5, pp. 579–592, 2003.

[27] T. Clausen, P. Jacquet, A. Laouiti, P. Muhlethaler, A. Qayyum, and L. Viennot, "Optimized link state routing protocol," in *Proc. IEEE Nat. Multi-Topic Conf. (INMIC)*, Dec. 2001, pp. 62–68.

[28] R. Jain, W. R. Hawe, and D.-M. Chiu, *A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer System*. Eastern Res. Lab., Digit. Equip. Corp., Hudson, MA, USA, 1984.

[29] V. D. Park and M. S. Corson, "A highly adaptive distributed routing algorithm for mobile wireless networks," in *Proc. INFOCOM*, vol. 3, 1997, pp. 1405–1413.

[30] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing*. Boston, MA, USA: Springer, 1996, pp. 153–181.

[31] P. Hui, J. Crowcroft, and E. Yoneki, "BUBBLE rap: Social-based forwarding in delay-tolerant networks," *IEEE Trans. Mobile Comput.*, vol. 10, no. 11, pp. 1576–1589, Nov. 2011.

[32] T. Seregina, O. Brun, R. El-Azouzi, and B. J. Prabhu, "On the design of a reward-based incentive mechanism for delay tolerant networks," *IEEE Trans. Mobile Comput.*, vol. 16, no. 2, pp. 453–465, Feb. 2017.

[33] J. L. Sobrinho, "Algebra and algorithms for QoS path computation and hop-by-hop routing in the Internet," in *Proc. IEEE Conf. Comput. Commun., 20th Annu. Joint Conf. IEEE Comput. Commun. Soc. (INFOCOM)*, Anchorage, AK, USA, 2001, pp. 727–735.

[34] J. L. Sobrinho, "An algebraic theory of dynamic network routing," *IEEE/ACM Trans. Netw.*, vol. 13, no. 5, pp. 1160–1173, Oct. 2005.

LUCAS S. BRITO received the B.Sc. degree in network engineering and the M.Sc. degree in automation and electronic systems engineering from the University of Brasilia, Brazil, in 2011 and 2016, respectively. He has worked as a Researcher at the Brazilian Agriculture Research Company. He is currently working in the information technology industry.

MARCELO M. CARVALHO (Member, IEEE) received the B.Sc. degree in electrical engineering from the Federal University of Pernambuco, Brazil, in 1995, the M.Sc. degree in electrical engineering from the State University of Campinas, Brazil, in 1998, the M.Sc. degree in electrical and computer engineering from the University of California at Santa Barbara, USA, in 2003, and the Ph.D. degree in computer engineering from the University of California at Santa Cruz, USA, in 2006. He is currently an Assistant Professor with the Department of Electrical Engineering, University of Brasilia, Brazil. His research interests include wireless networks, the Internet of Things, and multimedia communications. He is also a member of the ACM Society.