LIP: A Light-weighted Session-based Incentive Protocol for Multi-hop Cellular Networks

Hao Yue*, Miao Pan*, Rongsheng Huang*, Hongxia Zhao† and Yuguang Fang*
*Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611
†Huawei Technologies Co., Ltd.
Email: {hyue@, miaopan@, rshuang@}ufl.edu, zhaohongxia@huawei.com, fang@ece.ufl.edu

***Abstract*—The multi-hop cellular network (MCN) is an evolved paradigm for mobile communications, which integrates the ad hoc characteristics into the conventional cellular systems. Similar to ad hoc networks, the performance of MCNs relies on the hypothesis that each node accepts to forward traffic for the benefit of others, which may not hold with the possible presence of selfish users. In order to stimulate the collaboration among mobile nodes in MCNs, in this paper, we propose a light-weighted secure incentive protocol (LIP). We introduce a novel reward model, in which not the source and/or the destination but the network operator credits the forwarding nodes. It is shown that our model is much more realistic for MCNs in practice and simplifies the payment scheme design as well. LIP exploits a reactive receipt-submission mechanism to identify node behavior, which significantly reduces the communication overhead. Security analysis shows that LIP can resist various attacks. The efficiency of LIP is validated through the performance evaluation.**

I. Introduction

In the last few decades, mobile communications have had an unprecedented development. On the one hand, the number of subscribers in the world has been increasing exponentially, which will reach 3.96 billions by the year 2011 according to the market study [1]. On the other hand, various data applications have a blooming growth and gradually become the dominant services provided by the network operators. It leads to higher requirements on bandwidth, data rate and the quality of service (QoS). Under such circumstances, cellular systems, which were primarily designed for supporting simple voice communications, are exposed with more and more drawbacks and face a great challenge.

In 2000, an advanced wireless mobile communication paradigm, known as multi-hop cellular networks (MCNs), was presented [2] and has drawn great attention from both academia and industry lately. MCNs allow mobile nodes acting as relays to forward traffic for other nodes or base stations (BSs), which not only preserves the characteristics of traditional SCNs, such as the widest deployment and the mobility management, but also incorporates the flexibility of ad hoc networks [3], [4]. Compared to SCNs, MCNs have several attractive advantages. For example, MCNs extend the coverage area, improve the spatial reuse, reduce total transmission power, and increase the system capacity as well as data rate. Therefore, MCNs are considered as a promising candidate for the 4G wireless communication systems. Fig. 1 shows a generic architecture of MCNs.

This work was supported in part by the U.S. National Science Foundation under grant CNS-0721744 and CNS-0716450.

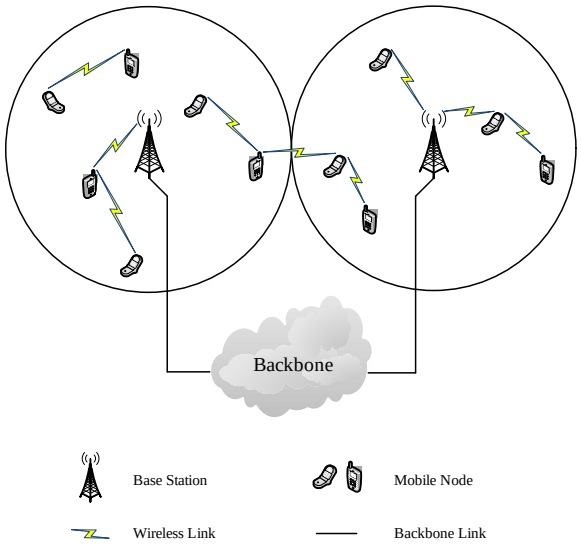

Fig. 1. **A generic architecture of MCNs.**

However, multi-hop transmission also brings in a new problem. In MCNs, packets need to be forwarded by multiple intermediate nodes to reach the destination. Therefore, similar to ad hoc networks, the performance of MCNs highly relies on the collaboration among mobile nodes to relay packets for each other. However, some participating nodes in MCNs may be self-interested in nature, and they will refuse to do so because it consumes their own limited communication and computational resources without any immediate return. In that case, the benefits of multi-hop transmissions will diminish and the network even cannot properly function. Therefore, an incentive mechanism to stimulate resource sharing and data forwarding among nodes is indispensable for the multi-hop cellular systems.

Several cooperation incentive proposals have been presented for MCNs. In [5], Jakobsson et al. introduce a micro-payment scheme to encourage node collaboration in an asymmetric cellular network with a multi-hop uplink and a single-hop downlink. Due to the probabilistic payment technique exploited, forwarders may not receive their deserved credits for the relayed packets. Salem et al. extend the asymmetric network model to the case that both the up-stream and the down-stream paths are potentially multi-hop, and propose two incentive protocols [6], [7] which could withstand a wide range of rational and malicious attacks. In order to prevent

the payment evasion, their work requires the source and/or the destination to pay for all the packets, no matter whether they are received by the destination or not. But it is not fair for the honest payers to spend credits on the failed packet transmissions. CASHnet, proposed by Weyland and Braun in [8], is another cooperation and accounting strategy for MCNs. Each node relies on the ACK from the next hop to remunerate itself. CASHnet assumes the existence of a tamper-proof device in each node and a decentralized account management mechanism, but these assumptions ignore the merit of the infrastructure environment in MCNs. In [9], Mahmoud and Shen propose a receipt-submission based scheme DSC. They use the hash chain to aggregate receipts and reduce the overhead. However, when packet loss happens, the aggregation technique becomes ineffective and the number of the submitted receipts will increase.

In this paper, we propose a light-weighted virtual-currency based incentive protocol (LIP) to motivate node cooperations in MCNs. Compared with previous schemes, LIP has a number of unique appealing features:

- LIP is based on a novel reward model, in which the network operator pays for the packet forwarding service. Compared to the previous models where the source and/or the destination remunerate the relay nodes, our model is much more realistic for MCNs in practice and simplifies the incentive scheme design.
- LIP does not require any tamper-proof hardware. It relies on the existing infrastructure in cellular systems to fairly conduct accounting and prevent attacks.
- LIP is applicable to all transmission scenarios in MCNs, with or without involvement of the BSs.
- LIP exploits a reactive receipt-submission strategy to identify the forwarding behavior of the relay nodes, which not only retains the advantages of the traditional receipt-submission approaches but also significantly lowers the communication overhead.

II. System Model

A. Network Model

Consider a cellular network which consists of a collection of BSs and mobile nodes. The BSs are connected with each other by a secure and fast backbone network. Each one controls a certain geographical area, called a cell. We assume that the BSs and mobile nodes have the same transmission range, which equals to the radius of a cell. In other words, the BS can communicate with any mobile node in the same cell over one-hop. Note that this assumption is different from some prior work [6], [7], in which the coverage of the BS is smaller than the cell size and the nodes near the boundary of a cell can only connect to the BS with the relay of other nodes. In cellular systems, BSs have to exchange control messages periodically with each mobile node. Routing these packets in a multi-hop way will cost too much resource and lead to an unexpected delay. Additionally, when the node density is low, the coverage of the BS cannot be guaranteed only by the multi-hop transmission. Therefore, our assumption is more reasonable. Mobile nodes move freely in the network, but the mobility is not so high and there always exists a contemporaneous path from the source to the destination.

B. Transmission Model

According to the proposed implementations of multi-hop cellular architecture [2]–[4], we assume three kinds of transmission modes in MCNs:

- **Uplink mode:** In this case, packets are transmitted from a mobile node to the closest BS through multiple short hops.
- **Downlink mode:** The BS sends a batch of packets to a mobile node in the same cell with multi-hop relays.
- **Ad hoc mode:** The communications between two mobile nodes do not involve any infrastructure of the cellular system. It always happens when the source and the destination are close to each other.

Most of the existing incentive mechanisms for MCNs could only be applied to the model where packets are transmitted between two mobile nodes with multi-hop relays and via at least one BS. Note that it can be considered as a hybrid of uplink and downlink modes described above. Our LIP does not make any restriction on the communication scenarios in MCNs, and is able to support not only the hybrid case, but also the pure ad hoc transmissions.

C. Reward Model

The network operator maintains an account for each registered user, and remunerates or charges one appropriately based on its behavior. Mobile nodes will get rewards if they forward packets for the benefit of others. The monetary rewards can appear in terms of cash back, a discount on the user's subscription, or free promotion to an advanced service plan. In this paper, we use credit increment to represent all the possible forms of rewards.

One important issue in the reward model is who should provide financial support to the multi-hop relay service. Originating from the incentive proposals for ad hoc networks where no operator exists, the work in the literature for MCNs also asks the source and/or the destination to remunerate the forwarders. However, we argue that this strategy is neither justified nor efficient in practice.

In MCNs, mobile nodes are encouraged to transmit packets with multi-hop relay to improve the whole system's performance. However, if the operator asks the source and/or the destination to reward the intermediate nodes, they will be reluctant to use the multi-hop forwarding service and prefer the traditional single-hop transmission. If it prevails, the benefits of MCNs will no longer exist and the system will degenerate to the SCNs.

Additionally, since nodes are selfish, the source and/or the destination may attempt to escape from the payments after the packets have been delivered. The relay nodes are also likely to get rewards by cheating without truly forwarding the packets. The network operator has to supervise the behavior of both sides. Consequently, the complexity and overhead of the schemes are always quite high.

With the above considerations, in our scheme we designate the network operator to reward the forwarding nodes. The

resulting benefit on scheme design and overhead reduction will be shown in Section VI.

D. Trust and Adversary Model

We postulate that one single operator operates all the BSs and the backbone network. The operator is fully trusted by all the mobile nodes to fairly manage the accounting system. The BSs are trusted to correctly transmit packets and collect information for the operator.

Mobile nodes in the network are controlled by autonomous clients. They are not trusted to follow the protocol honestly and voluntarily. Basically, they have four characteristics:

- **Selfish:** Nodes are always self-interested and unwilling to sacrifice their limited resources to relay packets for others with no return.
- **Greedy:** Confronting the temptation of financial rewards, a mobile node may try everything possible, even cheating, to gain more credits.
- **Rational:** Rational means nodes determine their behavior according to the profit resulting from it. They attempt to cheat only if the expected benefit is greater than that of acting honestly. Note that this is quite different from malicious attacks, which aim to disrupt the network normal operation.
- **Collusive:** Nodes might collude together if they are able to benefit from doing so. Collusion makes some misbehavior much harder to identify.

III. Design of the LIP Protocol

In this section, we will use the ad hoc transmission mode as an example to describe the design of our LIP. In section IV, we will briefly describe how LIP works for the uplink and down transmission scenarios.

A. Initialization

After registering to the cellular network, each mobile node i will receive an ID-based public/private key pair (ID_i, PK_i) and a symmetric key K_i from the operator. The symmetric key K_i is only shared between node i and the operator.

In our scheme, we use ID-based cryptography to achieve the identity authentication and protect the message integrity. Each intermediate node can check the validity of the received messages, which prevents the invalid packets from propagating along the route. Compared to the standard public-key cryptography, our approach could avoid the exchange and verification of public-key certificates, which is much more efficient for MCNs.

When two mobile nodes intend to communicate, they first need to set up an end-to-end session, in which all the intermediate nodes are authenticated and accept to forward packets. The source node sends a session setup request message $SREQ_S$ to the destination, which is in the following format:

$$\begin{aligned} SREQ_S &= < ID_S, ID_D, SSN, PATH, QUAL, \\ &\quad INFO > \end{aligned}$$

Here, ID_S and ID_D carry the IDs of the source and the destination nodes, respectively. SSN is the session sequence number determined by the source. The pair (ID_S, SSN) can uniquely identify any session in the system. The fields $PATH$ and $QUAL$ are initially empty. Each forwarder along the route will append its ID to the $PATH$ field sequentially, and record the link quality into the $QUAL$ field, such as the delivery probability. Information about the session traffic can be found in the $INFO$ field, such as the traffic type, the overall amount and the QoS requirement, etc.

When node i receives a $SREQ$ message, it checks the traffic information described in the $INFO$ field. With the concern of its available resources and the wireless link condition, node i makes a decision on joining this session or not. If it decides to participate in the forwarding, node i adds its ID into the $PATH$ field, computes a new $\mathcal{HMAC}$ with its symmetric key K_i, replaces the old $\mathcal{HMAC}$ and transmits the updated session request message to the next hop. $\mathcal{HMAC}$ is the keyed-hash message authentication code for data integrity check. Otherwise, it just discards the received $SREQ$ message.

$$\begin{aligned} SREQ_i &= < ID_S, ID_D, SSN, PATH, QUAL, \\ &\quad INFO, \mathcal{HMAC}_{K_i}(SREQ_{i-1}) > \end{aligned}$$

Each intermediate node processes the request message in the same way until it reaches the destination of the traffic. The destination node evaluates the quality of the path based on the information in the $QUAL$ field to determine whether it is able to support the traffic. If no, the destination node discards the $SREQ$ message. Otherwise, it updates the $\mathcal{HMAC}$ field and directly sends the pair $(PATH, \mathcal{HMAC}_{K_D})$ to the BS through the traditional one-hop uplink. The BS repeats all the $\mathcal{HMAC}$ computations according to the order of the node IDs listed in the $PATH$ field and checks the result against the received one from the destination. If correct, the BS sends a confirmation message $CFIR$ along the route,

$$\begin{aligned} CFIR &= < ID_S, SSN, PATH, \\ &\quad \mathcal{SIG}_{BS}(ID_S || SSN || PATH) > \end{aligned}$$

where $\mathcal{SIG}$ is the ID-based digital signature, and $||$ denotes message concatenation.

Each node verifies the $\mathcal{SIG}$ field in the $CFIR$ message. If valid, it saves ID_S, SSN and the next hop in the $PATH$ field in the routing table.

The initialization phase is completed. In the rest of the paper, we assume that the established session contains M packets to be delivered. The BS creates and maintains a table for each active session indicated by the pair (ID_S, SSN). If no such a route can be found, the transmission request will be denied.

B. Packet Sending

When the packet sending phase starts, the source could transmit packets to the destination. The LIP packet contains the data payload and a few LIP fields, the format of which is as follows:

$$\begin{aligned} PKT &= < ID_S, ID_D, SSN, PSN, DATA, \\ &\quad \mathcal{SIG}_S(ID_S || SSN || PSN || \mathcal{H}(DATA)) > \end{aligned}$$

where PSN represents a non-decreasing session-related packet sequence number set by the source node. The triple

(ID_S, SSN, PSN) can uniquely identify a packet. $\mathcal{H}$ is a hash function. We use the hash value instead of the payload to reduce the computational overhead of the digital signature.

When node i receives a LIP packet, it performs the following operations:

1) Check whether the packet is new or not. If the packet with sequence number (ID_S, SSN, PSN) has been received before, the node just drops it since it is a duplicate.
2) Verify the $\mathcal{SIG}$ field. If it is incorrect, abort the processing and dump the packet.
3) Store the information in the packet to generate a receipt.
4) Output and forward the LIP packet to the next hop on the route.

Upon receiving a packet, the destination checks the validity of the $\mathcal{SIG}$ field and sends an end-to-end acknowledgement back to the source. Each forwarder will delete the information it stored about one packet when it receives the corresponding acknowledgement. It only keeps the receipts for the lost packets.

C. Rewarding

Suppose different services have been classified into several categories based on the QoS requirement. In this paper, we only consider the number of packets delivered to the destination, though LIP can be easily extended to other measurements like the maximum transmission delay. Every kind of services has a required packet delivery ratio and a predetermined standard rewarding rate, which are denoted by W and α, respectively. In the initialization phase, each forwarder can obtain the information about the traffic type and the corresponding requirement from the $INFO$ field in the session setup request message $SREQ$.

When the session is closed, the destination transmits a REP message along the path to report the delivery achievement of the session:

$$\begin{aligned} REP_D &= < ID_S, SSN, NumAck, \\ &\quad \mathcal{HMAC}_{K_D}(ID_S, SSN, NumAck) > \end{aligned}$$

Here, $NumAck$ denotes the number of the packets acknowledged by the destination.

When the REP message arrives at the relay node i, it checks every receipt it stored for the lost packets. If the sequence number has appeared in the REP message, node i will delete the corresponding receipt. Otherwise, it appends the sequence number to the REP message. Let $\overline{PSN}_i = (PSN_1^i, PSN_2^i, ..., PSN_j^i)$ denote the set of the sequence numbers inserted into the REP message by node i. The updated REP message is

$$REP_i = < REP_{i+1}, \overline{PSN}_i, ID_i >$$

When the upstream nodes and the BS receive the REP message, the pair $(\overline{PSN}_i, ID_i)$ will notify them that node i takes the responsibility to submit the receipts for the packets with sequence numbers in $\overline{PSN}_i$. In this way, LIP reduces the overhead on identifying intermediate nodes' behavior for each lost packet by only making the last receiver submit the receipt.

When the REP message is delivered to the source node, it adds another field $NumSnd$, which contains the number of packets the source sends out when the session is closed. Then, it forwards the REP directly to the BS. The BS records the information about $NumSnd$, $NumAck$, which packets are lost and which nodes will report the receipts for them in the table it maintains for this session. It also calculates the achieved packet delivery ratio, which is denoted by μ, as the ratio of $NumAck$ to $NumSnd$, i.e., $\mu = NumAck/NumSnd$. Based on them, the performance of the multi-hop packet forwarding service has three classes with different rewarding rates:

- **Completed:** All packets in the session are transmitted, and the delivery requirement is satisfied. Packet loss is mainly due to the random error of the wireless links. Since the packet forwarding job is well finished, each relay node will gain αM credits for the whole session.
- **Interrupted:** The achieved packet delivery ratio μ exceeds the requirement W, but only a part of packets have been sent out from the source node when the session is closed, i.e., $NumSnd < M$. This can be caused by the sudden exit of some relay nodes. In this case, the rewarding rate will decrease to β because the session is not finished. Each forwarder will get $\beta NumSnd$ credits in total.
- **Failed:** When the session is closed, only a few packets are received by the destination and the required delivery ratio is not achieved. The session will be considered as failed. Because in the initialization phase the destination node selects the path which can support the session, the failure cannot be caused by the link random error and with high probability attacks happen during the transmissions. The rewarding rate is γ credits per packet, and $\gamma < \beta < \alpha$. The operator will first reward the relay nodes for the packets acknowledged by the destination, and active the receipt-submission mechanism.

  Each forwarder is informed by the BS to submit the receipts according to what they reported in the REP message. To reduce the communication overhead caused by the receipt transmissions, each forwarder, say node i, sends the receipts in an aggregated form, the format of which is shown below:

$$RCT_i = < ID_S, SSN, \overline{\mathcal{H}}_i, \overline{\mathcal{HMAC}}_{K_i} >$$

  Here, $\overline{\mathcal{H}}_i = (\mathcal{H}(DATA_1^i), \mathcal{H}(DATA_2^i), ..., \mathcal{H}(DATA_j^i))$ lists all the hash values of the lost packets which the receipt RCT_i is submitted for. $\overline{\mathcal{HMAC}}_{K_i}$ is the aggregated result of a series of $\mathcal{HMAC}$, i.e.,

$$\begin{aligned} \overline{\mathcal{HMAC}}_{K_i} &= \mathcal{HMAC}_{K_i}(\mathcal{SIG}_1^i || \mathcal{HMAC}_{K_i} \\ &\quad (\mathcal{SIG}_2^i || ... \mathcal{HMAC}_{K_i}(\mathcal{SIG}_j^i)...)) \end{aligned}$$

  where the $\mathcal{SIG}_j^i$ denotes the $\mathcal{SIG}$ field in PKT_j^i.

  The BS verifies the receipt RCT_i. If the verification is correct, the operator remunerates the set of all the upstream forwarders of node i along the route for their forwarding action at rewarding rate γ. The operator also reimburses node i for the transmission of the receipt RCT_i.

Table. I shows a simple example about different packet transmission results and the corresponding rewarding rates for a session. Suppose the session has 100 packets, and the required delivery ratio is 80%.

TABLE I
AN ILLUSTRATION OF THE REWARDING RATE COMPUTATION WHEN $M = 100, W = 80\%$.

$(NumSnd, NumAck)$	μ	Type	Rewarding Rate
$(100, 92)$	92%	Completed	α
$(50, 45)$	90%	Interrupted	β
$(80, 40)$	50%	Failed	γ

From the above description we could see that, in our scheme, when the impact of misbehavior is small and most of the packets are successfully received by the destination, none of the receipts are transmitted. Only when the attacks become severe and the throughput decreases, the receipt-submission mechanism is active. In this way, LIP guarantees the performance of MCNs under the appearance of selfish nodes and significantly reduces the communication overhead.

IV. LIP FOR OTHER TRANSMISSION MODE

In this section, we will extend our LIP protocol to the uplink and downlink transmission modes. Basically, they can be considered as the special cases of the ad hoc transmission mode in which the BS is the source or the destination of the traffic. Therefore, LIP will be simplified. Due to space limitations, We only emphasize on the differences.

In the uplink transmission mode, the session setup request message $SREQ$ will be sent from the mobile node toward the BS. The REP message is transmitted from the BS.

In the downlink transmission mode, $SREQ$ is generated by the BS. Since now the BS has the copies of all the transmitted packets, the format of the receipt can be simplified. Consequently, the size of the receipt is further reduced.

$$\begin{aligned} RCT_i &= < ID_S, SSN, \overline{\mathcal{HMAC}}_{K_i} > \\ \overline{\mathcal{HMAC}}_{K_i} &= \mathcal{HMAC}_{K_i}(DATA_1^i || \mathcal{HMAC}_{K_i} \\ &\quad (DATA_2^i || ... \mathcal{HMAC}_{K_i}(DATA_j^i) ...)) \end{aligned}$$

V. SECURITY ANALYSIS

In our scheme the credit payer is the network operator, so the source and/or the destination have no incentive to launch attacks for escaping from the payments, such as denial of service, not sending acknowledgements, etc. Next we will analyze how LIP prevents the attacks carried out by the dishonest forwarders. Note that we will focus on rational attacks. Malicious attacks are not considered in this paper.

A. Double Rewarding

A greedy node may try to submit the same receipt multiple times to gain undeserved credits. Our LIP only requires the last receiver to declare receipts for the lost packets to determine which nodes should be remunerated. The decision does not rely on the number of the receipts reported by each node.

B. Free Riding

Two dishonest intermediate nodes can piggyback their own messages on the relayed packets and exploit the concurrent session for their private communications. To do so, they can avoid the cost for the session establishment.

The BS is trusted, so it will not disclose any private key, or generate a $\mathcal{SIG}$ field on behalf of a certain node. Therefore, the misbehavior nodes cannot compute the correct $\mathcal{SIG}$ on the modified packet. The invalid packet will be discarded when it passes through the honest forwarders residing between the attackers.

C. Forwarding Partial Packet

Several forwarders on the path could collude together and transmit only part of a packet which is enough to generate receipts. In this way, they attempt to earn credits with minimum resource consumption.

In the downlink transmission mode, the BS has the copy of every transmitted packet. If some intermediate nodes transmit an incomplete packet to their colluders, the packet will not be acknowledged by the destination. Moreover, the colluding nodes are not capable of producing the valid receipt without the entire packet. Therefore, the attackers cannot get the credits from the operator.

But for the uplink and ad hoc transmission scenarios, it is hard to protect the system against the forwarding partial packet attack, since the BS has no reliable knowledge on the lost packets to verify the reported receipts. In our LIP, to launch the attack, the colluders have to forward the $\mathcal{H}(DATA)$ and the $\mathcal{SIG}$ field of each packet for receipt generation. We can reduce the rewarding rate γ to make it equal to or less than the cost of the transmission. In that case, selfish nodes will have no incentive to carry out the attack.

VI. EVALUATION

A. Communication

In this subsection, we analyze the communication overhead of our scheme. Since reducing cost caused by the receipt transmissions is essential to the implementation of a receipt-submission based scheme, we focus on the comparison of the number and size of the submitted receipts in LIP with those in other existing approaches. We consider the ad hoc transmission mode, where LIP has the highest overhead among the three transmission modes. In our evaluation, the length of node identity and the sequence number are 16 bytes and 4 bytes, respectively [6].

Table II illustrates the receipt sizes of different protocols. Here, 1024-bit RSA is selected as the digital signature scheme, and the message digest function is MD5 [10]. Express, DSC and LIP generate one aggregated receipt for a series of packets. In our comparison, we assume their aggregated receipts are for the same 10 packets.

TABLE II
THE RECEIPT SIZES OF DIFFERENT PROTOCOLS (BYTES)

Sprite	Express	DSC	LIP
276	352	160	196

In Sprite, the receipt includes a digital signature of the packet, which increases its length. To reduce the cost of the signature, the other three protocols use the hash function instead. Express has the largest receipt size because it packs the receipts of several packets into one message without aggregation. The receipt of LIP is larger than that of DSC due to the hash value of each packet in $\overline{\mathcal{H}}$ field.

In Fig. 2, we present the simulation results on the number of submitted receipts under different packet delivery ratio. Assume one established session contains 100 packets to be transmitted. According to the experiment results in [9], we suppose the number of intermediate nodes is 4. Node mobility is not considered since it has the same effect on all the protocols.

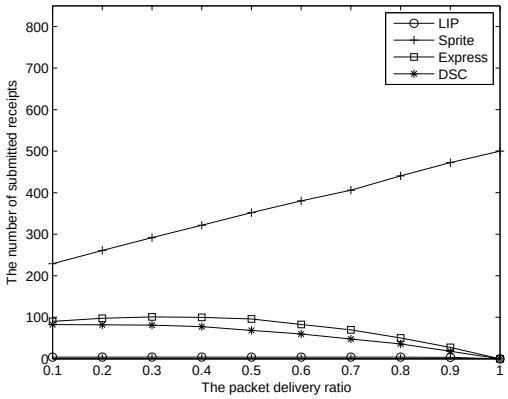

Fig. 2. **The number of submitted receipts.**

Sprite asks each relay node to submit one receipt for every packet it transmitted. As the packet delivery ratio increases, more packets are successfully forwarded by the intermediate nodes to the destination. Therefore, the number of receipts increases as well. In both DSC and Express, one receipt is generated for a batch of continually received packets, so they have almost the same number of receipts to report. But when the packet delivery ratio is low, more receipts are submitted because discrete packet receiving frequently happens. Different from other schemes, in LIP each forwarder only sends one aggregated receipt no matter how many packets in the session are not delivered to the destination. Thus, the number of the receipts is bounded by the number of the relay nodes, which is independent of the packet delivery ratio.

Fig. 3 shows the overall overhead of receipt submission in different incentive protocols. Because the receipt size is moderate and the receipt number is extremely small, LIP has the minimum overhead among all the schemes.

B. Computation

In LIP, the major online processing overhead is the $\mathcal{HMAC}$ computation, the $\mathcal{SIG}$ generation and verification. According to the implementation results with the Crypto++5.6 [11], a mobile node equipped with a 2.2 GHz AMD Opteron 8354 processor under Linux operating system can perform a $\mathcal{HMAC}$ computation with SHA-1 algorithm [12] at 187 Mbytes/s. For ECNR over GF(2^n) 233, the speed of the signature and verification is 5.52 Milliseconds/operation and

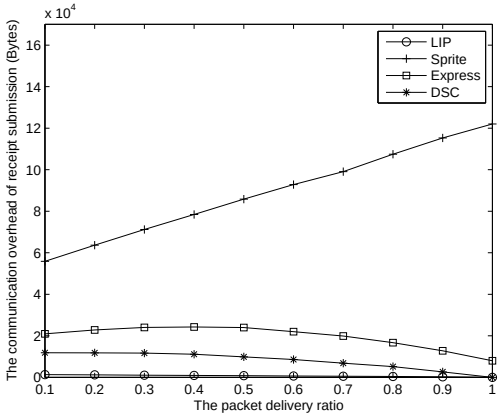

Fig. 3. **The communication overhead of receipt submission.**

6.73 Milliseconds/operation, respectively. The results could be scaled correspondingly to estimate the computational overhead for real mobile nodes.

VII. Conclusion

In this paper, we present a light-weighted virtual-currency based incentive protocol LIP to motivate node cooperation for packet forwarding in MCNs. A new reward model is proposed, which is much more practical and reduces the complexity of the scheme design. LIP has high flexibility and can be well adapted to all communication scenarios in MCNs. The performance evaluation shows that LIP is able to resist a wide range of rational attacks with low communication and computational overhead.

References

[1] The Portio Research Limited, "Worldwide mobile market forecasts 2006-2011," Market Study, UK, 2006.

[2] Y.-D. Lin and Y.-C. Hsu, "Multihop cellular: A new architecture for wireless communications," in *Proc. of INFOCOM 2000*, Tel Aviv, Israel, March 2000.

[3] H. Wu, C. Qiao, S. De, and O. To, "Integrated cellular and ad hoc relaying systems: icar," *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 10, pp. 2105–2115, October 2001.

[4] H. Lu, R. Ramjee, P. Sinha, L. Li, and S. Lu, "Ucan: A unified cellular and ad-hoc network architecture," in *Proc. of MobiCom 2003*, San Diego, CA, September 2003.

[5] M. Jakobsson, J.-P. Hubaux, and L. Buttyán, "A micro-payment scheme encouraging collaboration in multi-hop cellular networks," in *Proc. of Financial Cryptography (FC'03)*, Gosier, Guadeloupe, January 2003.

[6] N. Salem, L. Buttyán, J.-P. Hubaux, and M. Jakobsson, "A charging and rewarding scheme for packet forwarding in multi-hop cellular networks," in *Proc. of MobiHoc 03*, Annapolis, MD, June 2003.

[7] N. Salem, L. Buttyán, J.-P. Hubaux, and M. Jakobsson, "Node cooperation in hybrid ad hoc networks," *IEEE Transactions on Mobile Computing*, vol. 5, no. 4, pp. 365–376, April 2006.

[8] A. Weyland and T. Braun, "Cooperation and accounting strategy for multi-hop cellular networks," in *Proc. of the 13th IEEE Workshop on Local and Metropolitan Area Networks (LANMAN 2004)*, Mill Valley, CA, April 2004.

[9] M. Mahmoud and X. Shen, "DSC: Cooperation incentive mechanism for multi-hop cellular networks," in *Proc. of ICC 2009*, Dresden, Germany, June 2009.

[10] R.L. Rivest, "The MD5 message digest algorithm, RFC1321," April, 1992. [Online]. Available: http://andrew2.andrew.cmu.edu/rfc/rfc1321.htm

[11] W. Dai, "Crypto++ 5.6.0 benchmarks," 2010. [Online]. Available: http://www.cryptopp.com/benchmarks.html

[12] NIST, *Digital Hash Standard*. Federal Information Processing Standards Publication 180-1, 1995.