Analysis of Selfish behavior in energy consumption model based Multihop cellular networks

Prof. K. Mathivanan[1] Devi Ramalingam[2] Prof. R. Anbarasu [3] Ramya. S.[4]
[1,3] Associate Professor, Dept. of Computer Science and Engineering
[2,4]M. E., PG Scholar (Computer Science And Engineering)
[1, 2, 3, 4] Selvam College Of Technology, Namakkal.

Abstract — Many nodes would not transmit during data transmission and they are considered to be in cooperative. To make them cooperative a fair charging policy is used by charging the since and destination nodes so that both of them can benefit from the communication and it can secure the payment. Charging since and destination nodes almost computationally free, and significantly reduce the number of generated and submitted checks. In this way, each intermediate node earns some credits and the destination node pays the total packet relaying cost. To implement this charging policy efficiently, hashing operations are used in the ACK packets to reduce the number of public-key-cryptography operations. Moreover, reducing the overhead of the payment checks is essential for the efficient implementation of the incentive mechanism due to the large number of payment transactions.

Keywords: Network-level security and protection, wireless communication, payment schemes, hybrid systems.

I. INTRODUCTION

Multihop Cellular Network (MCN) has been used to incorporate the flexibility of ad hoc networks into traditional cellular networks. MCN realizes multihop transmission through peer-to-peer communication among mobile stations. They are capable of achieving much higher throughput than Single hop Cellular Networks (SCNs). Consequently, MCN-type system is considered as a promising candidate of fourth generation (4G) wireless network for future mobile communications. In the multihop cellular network (MCN) architecture, all cells use the same data and control channels. The MS and BS data transmission pointer is reduced to half the cell radius to enable multiple simultaneous transmissions using the same channel. It is argued that this reduction factor of two represents a compromise between increasing the spatial reuse and keeping the number of wireless hops to a minimum. The transmission pointer in the control channel corresponds to the cell radius, and the MSs use this channel to send information about their neighbors to the BS. To ensure reliable connectivity information at the BS, the nodes recognize their neighbors using a contention-free beacon protocol. When an MS wants to connect to a given destination, it sends a route request to the BS on the control channel. Then, using the topology information, the BS finds the shortest path between the source and destination, nd sends back a route reply with the shortest path to the source node. Upon receiving the route reply, the source node inserts the route into the packet and begins its transmission. In addition, the nodes cache route information to eliminate the control overhead. When a node detects that the next hop is unreachable, it sends a route error packet to the BS and buffers the current packet. The BS responds with a route reply to the node that generated the route error and also sends a correct route packet to the source node Service providers must construct an infrastructure with many fixed bases or access points to encompass the service area. By doing so, mobile stations can access the infrastructure in a single hop. In a densely populated metropolitan area, to support more connections, the area that a single base, i.e. a cell, covers is shrunk and the number of bases increases. This phenomenon unfortunately leads to (1) a high cost for building a large number of bases, (2) total throughput limited by the number of cells in an area, and (3) high pointer consumption of mobile stations having the same transmission range as bases. Notably, (1) and (2) trade off each other. If a higher throughput in a geographical area is decimal, more bases. I.e. cells must be constructed in that area. Another kind of network, commonly referred to as packet radio or ad-hoc networks is available in which no infrastructure or wireless backbone is required. In these networks, packets may be forwarded by other mobile stations to reach their destinations in multiple hops. If the source and the destination ***are*** in the same cell, other mobile stations can be used to relay packets to the destination, which achieves multihop routing within a cell. If not in the same cell, packets are sent to the base first, probably in multiple hops, and then be forwarded **to** the base of the cell where **the** destination resides MCN has several merits: (1) the number of bases or the transmission ranges of both mobile stations and base can be reduced, (2) connections are still alloitd without base stations. (3) Multiple packets can be simultaneously transmitted within a cell of the corresponding SCN, and (4) paths are less vulnerable than the ones in adhoc networks because the bases can help reduce the wireless hop count.

Methods to construct Multichip Cellular Networks

- MCN-b- In this method the number of bases is reduced such that the distance between two neighboring bases becomes kb times of that in SCN.
- MCN-p- In this method the transmission range of both bases and mobile stations is reduced to 1/kp of that in SCN.

In both cases, a base is not always reachable from a mobile station in a single hop. Hence, multihop routing is necessary. Nevertheless, MCN-b can be visited as a special case of MCN-p. The area of a cell in MCN-b is larger than that in MCN-p. When the number of mobile stations in a cell of the two architectures is the same, the throughput will be slightly different because of different propagation delay. However when the densities of the mobile stations of the two architectures are the same, the throughput in MCN-b descends quickly. Fig. 1 'shows an SCN and two possible architectures of MCN, MCN-b and MCN-p, **as** derived from **SCN.**

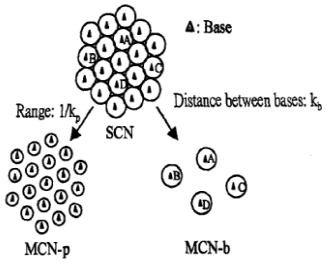

Fig.1.Examples of an SCN and two variables of MCN,MCN-P and MCN-N

Although the transmission range adopted in MCN-b is the same as that in SCN. the number of bases is reduced such that the distance between two neighboring bases becomes kt, times of the distance in SCN. Only four base stations, A, B, C. and D in the SCN are necessary in MCN-b. In MCN-p, although the number of bases is not reduced, the transmission ranges of base stations and mobile stations are reduced to l / k p of these adopted in SCN. Thus packets might be forwarded. in both MCN-b and MCN-p, by mobile stations to arrive destinations in multiple hops. Nevertheless, MCN-b can be visited as a special case of MCPc if the transmission range and the distance between hops.

A. Existing Problem

Peer-to-peer (p2p) networks, such as Napster and Bit Torrent, have become essential media for information dissemination and sharing over the Internet. Concerns about privacy, however have grown with the rapid development of P2P systems. In distributed and decentralized P2P environments, the individual users cannot rely on a trusted and centralized authority, for example, a Certificate Authority (CA) center, for protecting their privacy. Without such trustworthy entities, the P2P users have to hide their identities and behavior by themselves. Hence, the requirement for anonymity has become increasingly critical for both content requesters and providers. A number of methods to provide anonymity. Most, if not all, of them achieve anonymous message delivery via no traceable paths comprised of multiple proxies or middle agent peers. Those approaches, also known as path-based approaches, require users to setup anonymous paths before transmission. In most cases, the path is a layer-encrypted data structure. Although path-based protocols provide strong anonymity, an anonymous path has to be reconstructed, which requires the initiator to collect a large number of IP addresses and public keys. Also, an initiator has to perform asymmetric key based cryptographic encryptions, for example RSA, when wrapping the layer-encrypted packets. Both the peer collection and content encryption introduce high costs. Practically, users often expect to establish a long anonymous path and update the path periodically to defend against the analysis from attackers. In highly dynamic P2P systems, when a chosen peer leaves, the whole path fails. Unfortunately, such a failure is often difficult to be known by the initiator.

B. Simulation For Cooperation

Users have to frequently probe the path and retransmit messages so Fair, Efficient, and Secure Cooperation Incentive Mechanism is used to stimulate the node cooperation in MCN. In order to efficiently and securely charge the since and destination nodes, the light hashing operations are used in the ACK packets to reduce the number of public-key-cryptography operations. The destination node generates a hash chain and signs its root, and acknowledges message reception by releasing a hash value from the hash chain. In this way, the destination node generates a signature per group of messages instead of generating a signature per message. Furthermore, instead of generating a check per message or generating a nodal check for each intermediate node, a small-size check containing the payment data for all the intermediate nodes is generated per route. In addition, trusting one node to submit the check is not secure because this node may collude with the since and destination nodes to not submit the check. Instead of submitting the checks by all the intermediate nodes to thwart collusion attack, a Probabilistic Check-Submission scheme is proposed to reduce the number of submitted check.

II. RELATED WORK

A. Tamper Proof Device

In tamper-proof device (TPD)-based incentive mechanisms is used in which a TPD is installed in each node to manage its credit account and secure its operation. In Nuggets the self-generated and forwarding packets are passed to the TPD to decrease and increase the node's credit account, respectively. The terminode is rewarded with the nuggets Now, if a terminode wants to use a service (e.g., wants to send a message), then it has to pay for it in nuggets. This motivates each terminode to increase its number of nuggets, because nuggets are indispensable for using the network. Thus, the terminode is no longer interested in sending useless messages and overloading the network because this would decrease its number of nuggets, and it is better off providing services to other terminodes because this is the only way to earn nuggets.

1) Packet Purse Model (PPM)

In this model, the originator of the packet pays for the packet forwarding service. The service charge is distributed among the forwarding terminodes in the following way: When sending the packet, the originator loads it with a number of nuggets sufficient to reach the destination. Each forwarding terminode acquires one or several nuggets from the packet and thus, increases the stock of its nuggets; the number of nuggets depends on the direct connection on which the packet is forwarded for direct connection (long distance requires more nuggets). If a packet does not have

enough nuggets to be forwarded, then it is discarded Packet forwarding in the Packet Purse Model is illustrated in Figure 1. For each each terminode has 7 nuggets (1). Furthermore, let us assume that _ wants to send a packet to . In order to do so, _ loads, say, 5 nuggets in the packet and sends it to the next hop (2).takes out 1 nugget from the packet, and forwards it with the remaining 4 nuggets to (3)takes out 2 nuggets from the packet and forwards it with the remaining 2 nuggets to the final destination (4). Note that terminodes And , which forwarded the packet, increased their stock of nuggets, whereas terminode , which originated the packet, decreased its stock of nuggets.

Problem: The basic problem with this approach is that it might be difficult to estimate the number of nuggets that are required to reach a given destination. If the originator underestimates this number, then the packet will be discarded, and the originator loses its investment in this packet. If the originator over-estimates the number (like in in example above), then the packet will arrive, but the originator still loses the remaining nuggets in the packet3. The model distributed among the forwarding terminodes in the following way: When sending the packet, the originator loads it with a number of nuggets sufficient to reach the destination. Each forwarding terminode acquires one or several nuggets from the packet and thus, increases the stock of its nuggets; the number of nuggets depends on the direct connection on which the packet is forwarded (long distance requires more nuggets). If a packet does not have enough nuggets to be forwarded, then it is discarded.

2) The Packet Trade Model (PTM)

In this approach, the packet does not carry nuggets, but it is traded for nuggets by intermediate terminodes. Each intermediary "buys" it from the previous one for some nuggets4, and "sells" it to the next one (or to the destination) for more nuggets. In this way, each intermediary that provided a service by forwarding the packet, increases its number of nuggets, and the total cost of forwarding the packet is covered by the destination of the packet. As an example, let us consider Figure 2. Let us assume that originally each terminode has 7 nuggets (1). Furthermore, let us assume that wants to send a packet to sends the packet to the first hop for free (2) then sells it to the next hop for 1 nugget (3). the final destination for 2 nuggets (4). Note that terminodes and, which forwarded the packet, increased their number of nuggets, whereas the destination decreased its number of nuggets.

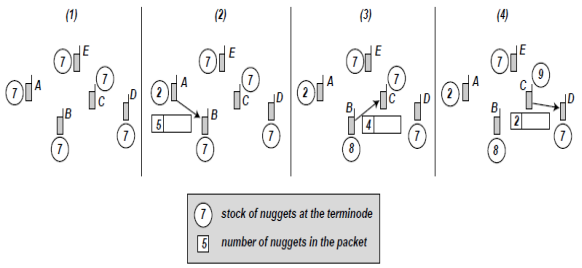

Fig 2. A packet trade model

Problems in Packet Trade Model (PTM):

– A forwarding terminode should be denied taking more Nuggets out of the packet forwarding (i.e. packet robbery should be prevented
– The integrity of the packet purse should be protected during transit.
– The replay of a packet purse should be detected
– Detachment of a packet purse from its original packet and Re-use of it with another packet should be possible
– Each terminode should be denied the re-use of the Nuggets that it spent for buying packets
– A forwarding terminode should receive the nuggets from
– An intermediary should be prevented from selling.

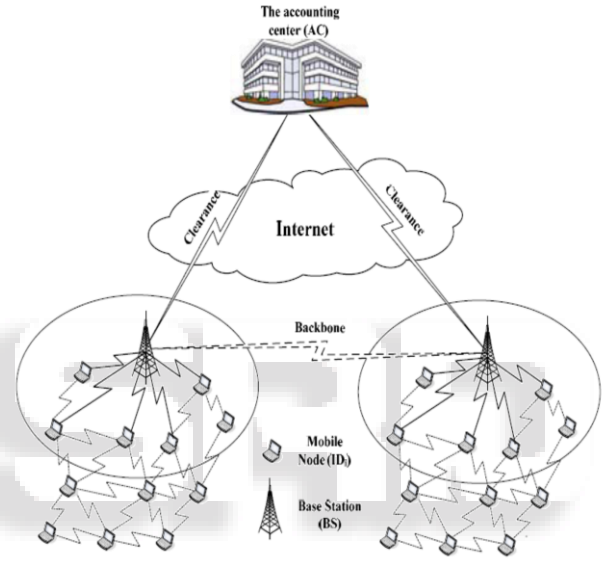

Fig.3 A multihop cellular network architecture

In order to eliminate the need for TPDs, a central bank called he AC can be used to store and manage the nodes' accounts. In the source node appends a payment token to each transmitted packet, and each intermediate node uses its secret key to check whether the token corresponds to a winning icket. Winning tickets are submitted to the AC to reward the winning nodes. The source and destination nodes are charged per packet but the intermediate nodes are rewarded per winning ticket. In a security flaw, the colluding nodes can exchange tokens to be checked in each node to steal credits. In our earlier work instead of submitting payment checks to the AC, each node submits an activity report containing its alleged charges and rewards of different sessions. The AC uses a reputation system to identify the cheating nodes that report false charges and/or rewards to steal credits. However due to the nature of the reputation systems, some honest nodes may be falsely identified as cheaters and the colluding nodes

III. GENERAL ASSUMPTIONS

Tamper resistant security module**:** It assumes that each terminode has a tamper resistant security module, such as,

for instance, a special chip or a smart card, that is used for the management of cryptographic parameters (e.g., keys) and nuggets. It assume that this security module functions correctly and its behavior cannot be modified by the user of the terminode or other attackers. Contrary to the security module, other parts of the terminode hardware and software are not tamper resistant and their behavior can be modified by anybody who has physical access to the device. It understands that regular users usually do not have the required level of knowledge and skills to modify their terminodes. Criminal organizations, however can have enough interest and resources to reverse engineer a terminode and sell tampered terminodes with modified behavior on a large scale. Users may be interested in buying these tampered devices if they offer advantages over correctly behaving ones (e.g., longer battery lifetime). Our design goal is to distribute the terminode functions between the tamper resistant security module and the rest of the terminode device, which can be altered by an attacker, in a way that modification of the latter cannot give any advantages to the attacker.

1) Threat and Trust Models

Since the mobile nodes are autonomous and self-interested, the attacker has full control on his node, and thus he can change its operation. The attackers work individually or collude with each other under the control of one authority to launch sophisticated attacks. The attackers are rational in the sense that they is behave when they can achieve more benefits than behaving honestly. Specifically, the attackers attempt to steal credits, pay less, and communicate freely. The mobile nodes are probable attackers because they are motivated to misbehave to increase their welfare, but the AC is fully secure. It is impossible to realize secure payment between two entities without trusted third party For the trust models, the network nodes fully trust the AC to correctly perform billing and auditing, but the AC does not trust any node or base station in the network. The network nodes also trust their operators' base stations but do not trust those of other operators

2) Payment Model

A fair charging policy is to support cost sharing between the since and destination nodes when both of them benefit from the communication. In order to make FESCIM flexible, the payment-splitting ratio is adjustable and service dependent, e.g., a DNS server should not pay for name resolution. For rewarding policy, some incentive mechanisms, such as consider that a packet relaying reward is proportional to the incurred energy in relaying the packet. It is difficult to implement this rewarding policy in practice without involving complicated route discovery process and calculation of route individual payments. Therefore, similar to it use fixed rewarding rate, e.g., _ credits per unit-sized packet. In MCNs, packet loss may occur normally due to node mobility, channel impairment, etc. Ideally, any node that has ever tried to relay a packet should be rewarded no matter whether the packet eventually reaches its destination or not because relaying a packet consumes the node's resources. However it is difficult to corroborate an intermediate forwarding action without involving too much overhead, e.g., all the intermediate nodes have to submit all the checks in Moreover, rewarding the nodes for relaying route establishment packets or packet retransmissions significantly increases the number of checks because a large number of nodes may relay route establishment packets and packet retransmission frequently happens in wireless networks. Therefore, the AC charges the since and destination nodes for every transmitted message even if the message does not reach the destination, but the AC rewards the intermediate nodes only for the delivered messages. For fair rewarding policy, the value is determined to compensate the nodes for relaying route establishment packets, packet retransmission, and undelivered packets it will argue that in charging and rewarding policies can thwart rational attacks and encourage the nodes' cooperation. Similar to the VISA system and the incentive mechanisms in the nodes communicate first and pay later. The AC issues certificates to enable the nodes to transact by issuing digital checks without the need for direct verification from the AC to avoid frequently contacting the AC and thus creating a bottleneck at the AC. The nodes at the network border cannot earn as many credits as those at other locations because they are less frequently selected by the routing protocol. In order to communicate, they can purchase credits with real money. However it do not consider this as a fairness problem because the philosophy behind incentive mechanisms is that packet relay is a service not an obligation. This service may not be requested from some nodes, i.e., the customers (since and destination nodes) request the packet-relay service from the best service providers (shortest route nodes). If the traffic is directed through the border nodes, obviously, it sacrifices the network performance because the routes may be long.

IV. PROPOSED SYSTEM

It has proposed a fair, efficient, and secure cooperation incentive mechanism for MCN. In order to fairly and efficiently charge the since and destination nodes, the lightweight hashing operations are used to reduce the number of public-key-cryptography operations. Moreover, to reduce the overhead of the payment checks, one small-size check is generated per session instead of generating a check per message, and the Probabilistic-

1) Check-Submission scheme: It is proposed to reduce the number of submitted checks and protect against the collusion attack. Extensive analysis and simulations have demonstrated that in incentive mechanism can secure the payment and significantly reduce the overhead of storing, submitting, and processing the checks. In addition, replacing the destination node's signatures with the hashing operations can charge the since and destination nodes almost computationally free In order to establish an end-to-end route, the since node broadcasts the Route Request Packet (RREQ) containing the identities of the since (IDS) and the destination (IDD) nodes, the route establishment time stamp (TS), and the payment-splitting ratio (Pr). The since node is charged the ratio of Pr of the total payment and the destination node is charged the ratio of 1 Pr. A network node appends its identity and broadcasts the packet if the time stamp is within a proper range. The RREQ packet is relayed by BSS to BSD (if the destination node resides in a

different base finally, the destination node sends back the Route Reply Packet (RREP) to establish the route. The since node initiates a new route discovery phase if the route is broken. The destination node generates a hash chain with size of Z þ 1 by iteratively hashing a random value called seed (HD 0ðiÞ) Z times to obtain a final hash value called root where and i is the hash-chain number. Note that multiple hash chains may be used in one route. the RREP packet contains the session identifier (Si), the destination node's certificate, the root of the first hash chain (HZ Dð1Þ), and the destination node's signature . Si contains the identities of the nodes in the route, TS, and Pr, e.g., Si ¼ IDS, ID1, ID2, BSS, ID3, ID4, IDD, TS, Pr for the route shown the destination node's signature authenticates the node and proves its approval to pay for the session. The signature also proves that the hash chain has indeed been created by the destination node and links it to the route. Upon receiving the RREP packet, each intermediate node relays the packet if the signature is correctly verified, and the since node starts data transmission.

2) Data Generation and Relay Phase

for the Xth data packet, that the since node appends the message MX and its signature Signing the hash of the message instead of the message can significantly reduce the check size because the smaller size and not MX is attached to the check. The since node attaches its certificate to the first data packet to enable the intermediate and destination nodes to verify its signatures. Before relaying a data packet, each intermediate node verifies the attached signature to ensure the message's integrity and authenticity and to verify the payment data that include Si and X. The intermediate nodes store only the last since node's signature to be used in the check composition, i.e., after receiving the Xth data packet the intermediate nodes delete and store that is enough to prove that X messages have been transmitted and X1 messages have been delivered. Each node in the route restarts a timer each time the node transmits or relays a packet. The route is considered broken when the timer expires. After receiving the ACK of the last message, the since node sends End of Session (EoS) packet to close the session.

3) ACK Generation and Relay Phase

Upon receiving the Xth data packet and X _ Z, the destination node sends back ACK packet containing the pre image of the last released hash value, to acknowledge receiving the message in an undeniable way. Therefore, instead of generating a signature per ACK packet, one signature is generated per Z ACKs. Payment no repudiation and non manipulation are achievable because the hash function is one-way, i.e., only the destination node could have generated the hash chain because it is not possible to compute HZ_X1 from HZ_X Each intermediate node verifies the hash value by making sure that HZ_X is generated from hashing HZ_X1 .after releasing all the hash values of the first hash chain, the destination node creates a new hash chain and authenticates it by signing all the used hash chains' roots, or HZ instead of signing only the last hash chain's root. In this way, the intermediate nodes store.

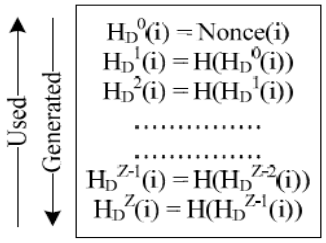

Fig.4. Hash chain generated by the destination node

4) Check Composition Phase

For each route, one check containing the payment data for all the intermediate nodes can be composed. The general format of the route check "[Y]" means that Y may not exist in some cases. A check contains two main parts: Descriptor (D) and Security Token (St). The Descriptor contains Si that has the identities of the payers and the payees, TS, and Pr. The Descriptor also contains the messages' number (X), the hash value of the last received message, the hash chains' roots and seeds, and the last released hash value The Security Token is

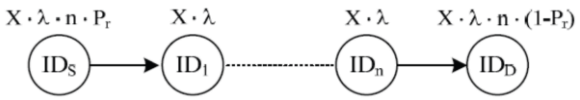

Fig.5. The charges and rewards for X messages

An undeniable proof that prevents payment repudiation and manipulation, and thus ensures that the check is undeniable, unmodifiable, and unforgivable. In order to significantly reduce the check size, the Security Token is composed by hashing the since and destination nodes' signatures instead of attaching the large-size signatures. The check size depends on the number of used hash chains because two hash values should be attached for each hash chain, and thus properly choosing the hash chain size can minimize the check size.

5) Check Clearance Phase:

The base station submits the check to the AC for redemption, but the nodes submit the check if the base station belongs to a different operator. The nodes also submit the check if the route is not complete, i.e., the EOS packet is not received, and the base station does not have correct payment information. For example, if the route is broken during relaying the ACK of MX from IDD to BSD, the BSD's check does not prove that the Xth message is delivered, and thus, the nodes are not rewarded for the last message if they do not submit the check. Once the AC receives a check, it checks that the check has not been deposited before using its unique identifier (Si). Then, the AC generates the source and destination nodes' signatures and hashes them to verify the check's credibility. The check is valid if the resultant hash value is identical to the check's Security Token. For a check (SC(X)), the number of transmitted messages (X) is signed by the source node, and the number of delivered messages can be computed from the number of hashing operation

V. PERFORMANCE EVALUATION

The checks overhead in terms of the check size and the number of generated checks and overhead of the signed and hash-chain-based ACKs in terms of energy consumption and end-to-end packet delay is evaluated

- Simulation Results
- Check Overhead

The simulation results given in Table demonstrate that although FESCIM adopts more fair charging policy than Sprite and Express, the check size can be less. This is because of hashing the source and destination nodes' signatures and signing the hash of the message instead of the message, i.e., hashing the signatures can alleviate the effect of the long RSA signature tag.

	Sprite	Express	FESCIM
Ad Hoc Mode	202.6	196	$86.6 + 40 \cdot i$
Hybrid Mode	214.53	196	$98.53 + 40 \cdot i$

Table. 1 Average Check size(Byte)

A check size depends on the number of used hash chains in the session (i) because two hash values are attached to the check per hash chain. If the hash-chain size is long enough, FESCIM can generate one fixed-size check per route. A storage area of 1 MB can store up to 8,283; 5,176; and 5,350 checks

S_{max}	Hash chain size (Z+1)	P(i = 1)	P(i = 2)	P(i = 3)	P(i > 3)
3 m/s	30	0.48	0.24	0.11	0.17
	50	0.6	0.28	0.12	0
10 m/s	30	0.89	0.11	0	0
	50	0.99	0.01	0	0

Table 2. The statistical distribution of the number of used hash chain

Table 2 gives the statistical distribution of the number of hash chains used for a route. The simulation results demonstrate that more hash chains are used in low node mobility because more packets are transmitted before the route is broken. It can also be seen that the probability of using only one hash chain increases with the increase of Z. Properly choosing Z can reduce the number of used hash chains, which reduces the check size and saves the destination node's resources because the unused hash values in a chain should not be used for other routes to secure the Payment. A good Z depends on the average number of transmitted packets before the route is broken, which is related to the packet transmission rate, the node speed, and the expected number of packets transmitted in the session Table 3gives the expected number of checks and the amount of paid and earned credits in Sprite, Express, and for a 300-second data transmission with different node speed. During the data transmission, a new route is established when the route is broken. For Sprite and Express, the simulation results demonstrate that the number of checks is large due to generating a check per message. The increase of the nodes' speed increases the number of checks because checks are generated for undelivered packets. It can significantly reduce the number of checks due to generating one check per route. More checks are generated at high node mobility because the routes are more frequently broken, i.e., the messages are transmitted over larger number of routes

– ACK Overhead

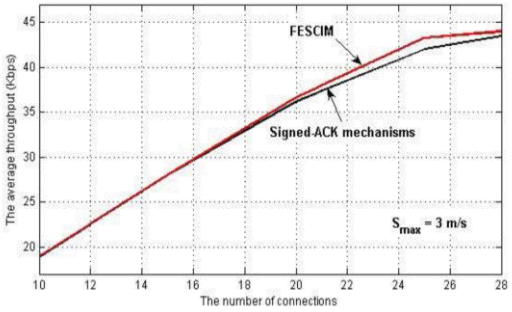

Fig 6. The End – to – end delay for the signed ACK mechanisms

For Z messages, the destination node generates Z signatures in signed ACK-based incentive mechanisms, but one signature and Z hashing operations are required in the ACK packets .From Fig. 7, it can be seen that FESCIM can significantly reduce the end-to-end delay due to replacing the destination node's signature with the lightweight hashing operations. Above 20 connection. the delay increases significantly with and without implementing the cooperation incentive mechanism due to the significant increase in the channel contention and queuing delays. The average network throughput is computed by dividing the size of the received data by all the nodes over the simulation time. As shown in Fig. 8 it can be seen that increasing the number of connections increases the network throughput t until the network reaches its capacity. The AC can learn that the data packet number X is dropped by an intermediate node A, B, or in-between node. However packets may be dropped sometime, e.g., due to mobility or bad channel, or maliciously, but frequently dropping packets is an obvious malicious behavior.

VI. CONCLUSION

In order to fairly and efficiently charge the source and destination nodes, the lightweight hashing operations are used to reduce the number of public-key-cryptography operations. Moreover, to reduce the overhead of the payment checks, one small-size check is generated per session instead of generating a check per message, and the Probabilistic-Check- Submission scheme has been proposed to reduce the number of submitted checks and protect against the collusion attack. Extensive analysis and simulations have demonstrated that our incentive mechanism can secure the payment and significantly reduce the overhead of storing, submitting, and processing the

checks. In addition, replacing the destination nodes signatures with the hashing operations can charge the source and destination nodes almost computationally free. Instead of generating two signatures per packet (one from the source and the other from the destination), it have replaced the destination node's signature with hashing operations to reduce the number of public-key-cryptography operations nearly by half. The source node attaches a signature in each data packet to ensure the payment non repudiation and to verify the message integrity at each intermediate node to thwart Free- Riding attacks. If two nodes IDA and IDB submit checks then AC can learn that the data packet number X is dropped by an intermediate node A, B, or in-between node. However packets may be dropped sometime, e.g., due to mobility or bad channel, or maliciously, but frequently dropping packets is an obvious malicious behavior. In our future work, the AC can precisely differentiate between the honest nodes and the irrational packet droppers in order to reduce the number of honest nodes that are falsely identified as irrational packet droppers.

REFERENCES

[1] Y. Lin and Y. Hsu, "Multihop Cellular: A New Architecture for Wireless Communications," Proc. IEEE INFOCOM, vol. 3, pp. 1273-1282, Mar. 2000.
[2] X. Li, B. Seet, and P. Chong, "Multihop Cellular Networks:Technology and Economics," Computer Networks, vol. 52, no. 9,pp. 1825- 1837, June 2008. [3] C. Gomes and J. Galtier, "Optimal and Fair Transmission RateAllocation Problem in Multi-Hop Cellular Networks," Proc. Int'lConf. Ad-Hoc, Mobile and Wireless Networks, pp. 327-340, Aug. 2009.
[3] C. Gomes and J. Galtier, "Optimal and Fair Transmission RateAllocation Problem in Multi-Hop Cellular Networks," Proc. Int'lConf. Ad-Hoc, Mobile and Wireless Networks, pp. 327-340, Aug. 2009.
[3] C. Gomes and J. Galtier, "Optimal and Fair Transmission RateAllocation Problem in Multi-Hop Cellular Networks," Proc. Int'lConf. Ad-Hoc, Mobile and Wireless Networks, pp. 327-340, Aug. 2009.
[4] H. Wu, C. Qios, S. De, and O. Tonguz, "Integrated Cellular and Ad Hoc Relaying Systems: iCAR," IEEE J. Selected Areas in Comm.,vol. 19, no. 10, pp. 2105-2115, Oct. 2001.
[5] G. Shen, J. Liu, D. Wang, J. Wang, and S. Jin, "Multi-Hop Relay for Next-Generation Wireless Access Networks," Bell Labs Technical J.,vol. 13, no. 4, pp. 175-193, 2009.
[6] R. Schoenen, R. Halfmann, and B. Walke, "MAC Performance of a 3GPP-LTE Multihop Cellular Network," Proc. IEEE Int'l Conf.Comm. (ICC), pp. 4819-4824, May 2008.
[7] 3rd Generation Partnership Project, Technical Specification Group Radio Access Network, "Opportunity Driven Multiple Access," 3G Technical Report 25.924, Version 1.0.0, Dec. 1999.
[8] S. Marti, T. Giuli, K. Lai, and M. Baker, "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks," Proc. ACM MobiCom,pp. 255-265, Aug. 2000.
[9] P. Michiardi and R. Molva, "Simulation-Based Analysis of Security Exposures in Mobile Ad Hoc Networks," Proc. EuropeanWireless Conf., Feb. 2002.
[10] J. Hu, "Cooperation in Mobile Ad Hoc Networks," Technical Report TR-050111, Computer Science Dept., Florida State Univ.,Jan. 2005.
[11] G. Marias, P. Georgiadis, D. Flitzanis, and K. Mandalas "Cooperation Enforcement Schemes for MANETs: A Survey," J. Wireless Comm. and Mobile Computing, vol. 6, no. 3, pp. 319-332,2006.
[12] C. Song and Q. Zhang, "OMH-Suppressing Selfish Behavior in Ad Hoc Networks with One More Hop," Mobile Networks and Applications, vol. 14, no. 2, pp. 178-187, Feb. 2009.
[13] D. Djenouri and N. Badache, "On Eliminating Packet Droppers in MANET: A Modular Solution," Ad Hoc Networks, vol. 7, no. 6, pp. 1243-1258, Aug. 2009.
[14] G. Bella, G. Costantino, and S. Riccobene, "Evaluating the Device Reputation Through Full Observation in MANETs," J. Information Assurance and Security, vol. 4, no. 5, pp. 458-465, Mar. 2009.
[15] L. Feeney, "An Energy-Consumption Model for Performance Analysis of Routing Protocols for Mobile Ad Hoc Networks,"Mobile Networks and Applications, vol. 3, no. 6, pp. 239-249, 2001.
[16] M. Peirce and D. O'Mahony, "Micropayments for Mobile Networks," technical report, Dept. of Computer Science, TrinityCollege, 1999.
[17] L. Buttyan and J. Hubaux, "Enforcing Service Availability in Mobile Ad-Hoc WANs," Proc. ACM MobiHoc, pp. 87-96, Aug.2000
[18] L. Buttyan and J. Hubaux, "Stimulating Cooperation in Self- Organizing Mobile Ad Hoc Networks," Mobile Networks and Applications, vol. 8, no. 5, pp. 579-592, Oct. 2004.