A Secure Incentive Scheme for Delay Tolerant Networks

Haojin Zhu, Xiaodong Lin, Rongxing Lu and Xuemin (Sherman) Shen
Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario, Canada
{h9zhu, xdlin, rxlu, xshen}@bbcr.uwaterloo.ca

Abstract— **Delay tolerant networks (DTNs) provide a promising solution to support delay tolerant applications in areas where end-to-end network connectivity is not available. In DTNs, the intermediate nodes on a communication path are expected to store, carry and forward the in-transit messages (bundles) in an opportunistic way, which is also named as opportunistic data forwarding. Opportunistic data forwarding depends on the hypothesis that each individual node is ready to forward packets for others. This, however, might be easily violated due to the existence of selfish nodes or even malicious ones, who may be reluctant to serve as the bundle relays to save their precious wireless resources. To address this problem, we propose a secure credit based incentive scheme to stimulate bundle forwarding cooperation among DTNs nodes. The proposed scheme can be implemented in a fully distributed way to thwart various attacks without relying on any tamper-proof hardware. In addition, we introduce several efficiency optimization techniques to improve the overall efficiency by exploiting the unique characteristics of DTNs. Extensive simulations confirm the efficacy and efficiency of the proposed scheme.**

I. Introduction

Most of the popular Internet applications are built on the existence of a contemporaneous end-to-end link between the source and destination. However, there are many cases for which such an "existence" is invalid, for instance, space communication and networking in sparsely populated areas [1], vehicular ad hoc networks [2], [3] and underwater networks [4]. These newly emerging networks characterized by long propagation delays and/or intermittent connectivity are often referred to as Delay Tolerant Networks (DTNs). In DTNs, the in-transit messages, also named as bundles, could be sent over an existing link, get buffered at the next hop until the next link in the path appears (e.g., a new node moves in range or an existing one wakes up). This message propagation process is usually referred to as "store-carry-and-forward" strategy and the routing is made in "opportunistic" fashion.

Previously reported studies have focused on opportunistic data propagation in DTNs [4], [5], which depends on the hypothesis that each individual node is ready to forward packets for others. This hypothesis, however, might be easily violated by the existence of selfish nodes or even malicious ones, who may be reluctant to serve as the bundle relays to save their precious wireless resources. Existing research shows that the presence of such selfish nodes may significantly degrade the overall performance of a non-cooperative communication scenario [6]. Thus, to deploy a DTN in real-world scenarios, we should these selfish or malicious nodes into consideration.

One of the promising ways to address the selfish issue and stimulate cooperation among those selfish nodes in DTNs is by using the *incentive scheme*, which basically falls into two categories: reputation based schemes and credit-based schemes. Reputation based schemes rely on the individual nodes to monitor neighboring nodes' traffic and keep track of the reputation of each other so that uncooperative nodes are eventually detected and excluded from the networks [7], while credit based schemes introduce some form of virtual currency to regulate the packet-forwarding relationships among different nodes [8]–[10]. The previously reported incentive schemes, which were proposed for the traditional mobile ad hoc networks, may not be suitable for DTNs due to the following reasons. Firstly, a common assumption adopted in existing incentive schemes is that an end-to-end connection between the source and the destination is established before the data forwarding occurs. This assumption does not hold in DTNs, which makes it infeasible for the source to track the forwarding path or remunerate the forwarding nodes online [8]. Secondly, the reported schemes are mainly designed for single path forwarding. However, multi-copy forwarding or even flooding are often adopted to enhance the reliability of DTN communication, which represents a great challenge for most of existing incentive schemes. Lastly and most importantly, existing schemes fail to consider the unique security characteristics of DTNs such as fragmentation and resource-scarcity [11], which is also one of major motivations of this work.

With the security concerns and uncooperative nature among users, to realize an applicable DTN, this paper presents a Secure Credit based Incentive (SCI) scheme. SCI is based on the notion of *layered coin*, a virtue electronic credit, to charge and reward the provision of data forwarding in DTNs. A layered coin is comprised of several layers, each of which is generated by the source/destination or an intermediate node. The first layer, also named as *base layer*, is generated by the source to indicate payment rate (credit value), remuneration conditions, class of service (CoS) requirement and other rewarding policy. During the following bundle propagation process, each intermediate node will generate a new layer based on previous layer by appending a non-forged digital signature. This new layer is also named as *endorsed layer*, which implies that this forwarding node agrees to provide forwarding service under the predefined CoS requirement and will be rewarded according to the rewarding policy in the future. With endorsed layers, it is easy to track the propagation path and determine each intermediate node by checking the signature of each endorsed layer. In the rewarding and charging phase, if the forwarding service provision satisfies remuneration conditions defined in the predefined rewarding policy, each forwarding node along a or multiple path(es) will

share the credit defined in this coin depending on different data forwarding algorithms (single-copy/multi-copying forwarding) and the actual forwarding results (bundle delivered along one or multiple pathes).

The contribution of this paper can be summarized as follows. Firstly, we propose a secure credit based incentive framework to stimulate the cooperations among selfish nodes in DTNs. The proposed scheme achieves flexibility by considering different data forwarding algorithms which may be adopted in DTNs. Secondly, to ensure the security of layered coin without requiring a tamper-resistant hardware, we introduce a novel *concatenated layer* technique to prevent the malicious users to cheat credits. Thirdly, we introduce SCI, a one way, non-interactive protocol for DTNs, where interactive communication suffers from long round-trip delays and frequent disconnection [1]. Lastly, we propose two performance optimization techniques to minimize the computation and transmission overhead.

The remainder of the paper is organized as follows. In Section II, we present the network model, node model, and the design goals. In Section III, the proposed SCI scheme is presented in detail. We propose two performance optimization methods in Section IV. Performance evaluation is given in Section V, followed by the conclusion in Section VI.

II. System Model and Design Goals

This section describes our system model and design goals.

A. Delay Tolerant Network Model

We model a DTN as a directed graph $G = (V, E)$, where V and E represent the set of nodes and edges, respectively. A source node $\mathcal{S}$ can deliver packets to a destination node $\mathcal{D}$ via one or multiple pathes depending on data forwarding algorithm. The existing data forwarding strategies in DTN can be categorized into single copy scheme and multi-copy scheme, where single-copy schemes such as [5] only route one copy per message while multi-copy schemes such as flooding or spray routing [4] use more than one copy per message, which can achieve better efficiency and reliability at the expense of extra transmission overhead. In this paper, we consider a general multi-copy data forwarding scheme: as shown in Fig. 1, for every bundle B originating at the source node $\mathcal{S}$, L copies of B are initially spread by the source and then, at every subsequent forwarding node, L message copies will be opportunistically propagated to the next hops. It is worth pointing out that both of single copying forwarding and flooding can be treated as a special case of this network architecture since L can be seen as 1 when single copy forwarding is adopted while L can be seen as a large number when flooding based forwarding algorithm is in place.

Similar to other credit based schemes such as [9], we assume that there exist an *Offline Security Manager (OSM)* and a *virtue bank* (VB) in our scheme, which are responsible for key management and clearance, respectively. Before joining the DTN network, every DTN node should register to the OSM and obtain its public key certificate. At the clearance phase, the DTN nodes submit the collected layered coins to VB for rewarding.

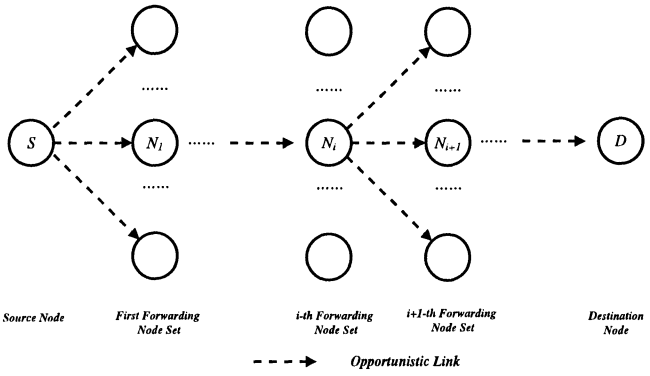

Fig. 1. Network Architecture

B. Node Model

DTNs have two types of uncooperative nodes: selfish nodes and malicious nodes. Selfish nodes are reluctant to forward packets destined for other nodes without gain or seek to economically maximize their own profit, but malicious nodes try to attack the system by interrupting the operations of the network. A well designed incentive system should be able to deal with both selfish nodes and malicious ones. We may encourage the cooperation among selfish nodes in the DTN networks with an incentive scheme. However, the incentive design may also open the door to the malicious attackers. For example, malicious attackers may collude with each other to fraud extra credits for the work they did not do or more than they deserve. Thus, to stimulate cooperation among DTN nodes while preventing attackers from disrupting the systems, it is necessary to pursue a secure incentive scheme.

C. Design Goals

The design goals include:

- *Effectiveness*: The scheme should be effective in stimulating the selfish nodes.
- *Efficiency*: The incentive scheme should be performed in an efficient way to reduce the communication and transmission overhead.
- *Security*: The incentive scheme should be secure and robust from the various attacks.

III. A Secure Credit based Incentive Scheme

In this section, we first provide some preliminary background and then present the secure incentive scheme in detail.

A. Pairing Technique

The proposed scheme is based on bilinear pairing which is briefly introduced as below. Let $\mathbb{G}$ be a cyclic additive group and $\mathbb{G}_T$ be a cyclic multiplicative group of the same order q, i.e., $|\mathbb{G}| = |\mathbb{G}_T| = q$. Let P be a generator of $\mathbb{G}$. We further assume that $\hat{e} : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ is an efficient admissible bilinear map with the following properties:

- Bilinear: for $a, b \in \mathbb{Z}_q^*$, $\hat{e}(aP, bP) = \hat{e}(P, P)^{ab}$.

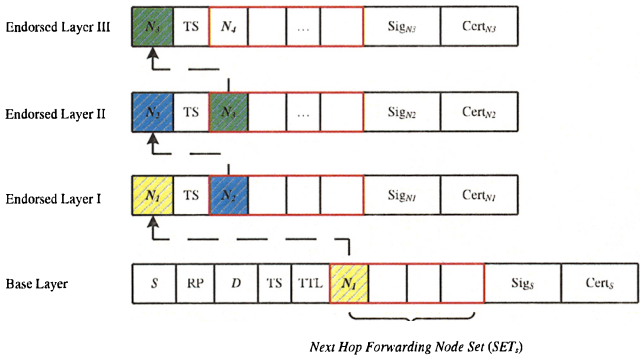

Fig. 2. An example of layered coin for a single forwarding path

- Non-degenerate: $\hat{e}(P,P) \neq 1_{\mathbb{G}_T}$.
- Computable: there is an efficient algorithm to compute $\hat{e}(P_1, Q_1)$ for any $P_1, Q_1 \in \mathbb{G}$.

According to [13], such an admissible bilinear map $\hat{e}$ can be constructed by Weil or Tate pairings on the elliptic curves.

B. Modeling of Layered Coin

The proposed scheme uses layered coin as incentives to stimulate packet forwarding. A layered coin may be comprised of a *base layer* and multiple *endorsed layers*, which serve different purposes in the SCI scheme. The architecture of a base layer generated by the source node can be shown in Fig. 2, where S and $Cert_S$ are the identity and public key certificate of source node, respectively, *RP* refers to the CoS requirements and rewarding policy proposed by the source, D is the identity of destination node, *TS* and *TTL* refer to the bundle creation timestamp and time-to-live information, respectively, *forwarding node set (SET)* includes all the possible forwarding nodes in the next hop and *Sig* is the signature generated by the source node to protect the authenticity and integrity of above information. Similar to base layer, an endorsed layer includes node identity, TS, forwarding node set and the signature.

One possible attack toward the layered coin is *layer removing attack*, in which one misbehaving node on the forwarding path attempts to remove previous layers to cheat more credits or enable the source to pay less rewarding credits. To thwart this attack and ensure the security of layered coin, we adopt the *layer concatenation* technique [12], which tries to concatenate different layers with each other by injecting the information of the next layer into the pervious layer. The basic idea of layer concatenation can be seen from Fig. 2. Starting from the source node, each node stores the next forwarding node set in its layer. It is obvious that, with layer concatenation technique, the different layers can form a linkable layer chain. Each following node can easily detect the layer removing attacks by checking the linkability of this layer chain.

C. Basic SCI Design

A basic SCI scheme includes "System Initialization", "Bundle Generation", "Bundle Forwarding" and "Charging and Rewarding" steps.

1) System Initialization: OSM adopts bilinear pairing system parameters $(q, \mathbb{G}, \mathbb{G}_T, \hat{e}, P)$ as the system parameters. In addition, two hash functions are formed: $H : \{0,1\}^* \rightarrow \{0,1\}^*$ and $H_2 : \{0,1\}^* \rightarrow \mathbb{G}$. The system parameters $(q, \mathbb{G}, \mathbb{G}_T, \hat{e}, P, H, H_2)$ will be preloaded in every DTN node. For any DTN node $\mathcal{N}$ which is going to join the DTN system, it randomly chooses $sk_{\mathcal{N}} \in \mathbb{Z}_q^*$ as its private key which corresponds to the public key expressed as $PK_{\mathcal{N}} = sk_{\mathcal{N}}P$. Then it contacts the OSM to obtain its corresponding public key certificate.

2) Bundle Generation: When a bundle sender $\mathcal{S}$ is going to send a bundle B to the destination $\mathcal{D}$, after determining the next hop forwarding node set $SET_{\mathcal{S}}$, $\mathcal{S}$ signs on the bundles with its private keys $sk_{\mathcal{S}}$ by computing $Sig_{\mathcal{S}} \leftarrow sk_{\mathcal{S}}H_2(B||\mathcal{S}||RP||D||TS||TTL||SET_{\mathcal{S}})$. Here, we use the Boneh, Lynn and Shacham signature [14] as the underlying building block to generate the supporting signature. Thus, $\mathcal{S}$ obtains the base layer as $B\_layer = (S, RP, D, TS, TTL, SET_s, Sig_{\mathcal{S}}, Cert_{\mathcal{S}})$. Then $\mathcal{S}$ forwards the bundle as well as the base layer to the next forwarding nodes as follows:

$$\mathcal{S} \rightarrow SET_{\mathcal{S}} : B, B\_Layer$$

It is important to note that, in a multi-copy opportunistic data forwarding algorithm, a bundle may be forwarded along with multiple pathes. Each forwarding path may form its layered coin even though the generated coins share a same base layer. Without loss of generality, in the following section, we will take a single forwarding path $\mathcal{S} \rightarrow \mathcal{N}_1 \rightarrow \mathcal{N}_2 \rightarrow \ldots \mathcal{N}_i \ldots \rightarrow \mathcal{N}_m \rightarrow \mathcal{D}$ as an example to show the details of SCI scheme, where $\mathcal{N}_m$ represents the last intermediate node.

3) Bundle Forwarding: When an intermediate node $\mathcal{N}_i$ receives the bundle as well as the layered coin which includes a base layer and multiple endorsed layers, it performs the following steps to authenticate the layered coin:

1) Check if the bundles are in their lifetime.
2) Check the linkability of the layer chains.
3) Verify the sender's certificate and check the supporting signature of base layer by verifying if $\hat{e}(P, sig_s) = \hat{e}(PK_{\mathcal{S}}, H_2(B||\mathcal{S}||RP||D||TS||TTL||SET_{\mathcal{S}}))$ holds.
4) Verify the intermediate nodes' certificates and check the endorsed layers one by one.

After performing above verifications and determining the next hop forwarding node set $SET_{\mathcal{N}_i}$, $\mathcal{N}_i$ creates an additional endorsed layer by computing $Sig_{\mathcal{N}_i} \leftarrow sk_{\mathcal{N}_i}H_2(B||B\_Layer||\mathcal{N}_i||TS||SET_{\mathcal{N}_i})$ and thus obtain the $i\text{-}th$ endorsed layer $E\_Layer_i = (\mathcal{N}_i, TS, SET_{\mathcal{N}_i}, Sig_{\mathcal{N}_i}, Cert_{\mathcal{N}_i})$. Then $\mathcal{N}_i$ forwards the bundle as well as the layered coin to the next forwarding node set as follows:

$$\mathcal{N}_i \rightarrow SET_{\mathcal{N}_i} : B, B\_Layer, E\_Layer_1, \ldots, E\_Layer_i$$

The verification of the supporting signature of $i\text{-}th$ endorsed layer be performed by computing if $\hat{e}(P, Sig_{\mathcal{N}_i}) = \hat{e}(\mathcal{N}_i, H_2(B||B\_Layer||\mathcal{N}_i||TS||SET_{\mathcal{N}_i})$ holds.

The similar steps are also be conducted by each intermediate node before the bundles reach the destination $\mathcal{D}$. When

the destination receives the bundles, it may also check the bundles' lifetime, senders and forwarders' certificates and the layered coins one by one. If the verification passes, it may generate a special endorsed layer as the receipt: $Sig_{\mathcal{D}} \leftarrow sk_{\mathcal{D}}H_2(B||B\_Layer||\mathcal{D}||TS)$. Thus it obtains the endorsed layer $E\_Layer_{\mathcal{D}} = (\mathcal{D}, TS, Sig_{\mathcal{D}})$. Then $\mathcal{D}$ sends it to $\mathcal{N}_m$ as follows

$$\mathcal{D} \rightarrow \mathcal{N}_m : B, E\_Layer_{\mathcal{D}}$$

Thus, the last intermediate node obtains a complete layered coin $B, B\_Layer, E\_Layer_1, \ldots, E\_Layer_i, \ldots, E\_Layer_m$, E_Layer_D, which will be submitted to the virtue bank for clearance in the future.

D. Charging and Rewarding

In SCI, it is up to the source to decide the rewarding policy such as rewarding rate, conditions and payees, which is promised in the base layer of layered coin. In some cases, the source may also choose some special rewarding policies to achieve some special stimulating goals. For example, from the efficiency point of view (trying to stimulate the bundle to be propagated as soon as possible), the source may consider to remunerate the intermediate nodes of the first arrived forwarding path. On the other hand, from the fairness point of view, the source may adopt the rewarding policy that the intermediate nodes of those forwarding pathes will be remunerated only if the bundles are delivered within their lifetime. The rewarding policy is propagated together with the layered coin to each intermediate node. Those intermediate nodes which agree to forward the packets are assumed to agree with these rewarding policy. Thus, the payment agreement can be implicitly achieved.

There are different charging models which can be adopted in SCI. For example, a popular charging method in existing literatures [8] is paying per packet, which means for each successfully transmitted unitsized pack, each of N intermediate nodes should receive λ credits while the source need to pay $\lambda * N$ in total. However, we argue that this method is not suitable for opportunistic data forwarding in that it is difficult for source to predict how many copies can be successfully delivered to the destination. Therefore, we may consider the profit sharing concept, which means each forwarder which satisfies the remuneration conditions will share a piece of profit of the total credit defined in the layered coin.

Another benefit of profit sharing payment method is that it can discourage the cheating behavior of source nodes which may try to collude with the last intermediate node to cheat credits. Recall that SCI relies on the last intermediate node to submit the layered coins for rewarding. If misbehaving or colluding with the source node, the last intermediate node may refuse to submit the collected coins, which may cause the loss to other intermediate nodes along this path. We further clarify this problem from two aspects. From the last intermediate node point of view, it may also lose the chance to gain credits from this rewarding protocol. From the source node point of view, under the profit sharing payment model, it is impossible for the source node to avoid charging even by colluding with the last intermediate node only if there is at least one different data forwarding path existed on which the packets are successfully delivered to the destination. Therefore, the source node may lack of motivations to launch the collusion attacks.

After a batch of a given size of layered coins are gathered, the last intermediate node may connect to the VB and submit the collected layered coins for clearance. VB first checks the certificates of each node in the forwarding path and then verifies the legitimacy of the layered coins. VB also check that if these layered coins have been deposited before by inquiring the sender's previous record. If all verifications pass, a predefined amount of the credit will be shared by all of the forwarders under a particular predefined rewarding policy.

IV. Performance Enhanced SCI

Due to resource scarcity characteristic, the computation and transmission efficiency is a critical concern in designing a practical incentive scheme in DTNs. In this section, we propose two methods to improve the computation and transmission efficiency of SCI.

A. SCI with Aggregate Signature

The signature transmission and verification contribute to the most of transmission and computation overhead incurred by SCI transmission and verification. Therefore, reducing the signature size and increasing the verification efficiency is a major concern in the practical deployment of the SCI scheme. In this section, we take the advantage of aggregated signature to reduce the transmission and verification cost.

An aggregate signature is a digital signature that supports aggregation of n distinct signatures issued by n distinct signers to a single short signature [14]. This single signature (and the n original messages) will convince the verifier that the n signers indeed sign the n original messages. With aggregate signature, it is possible for the intermediate nodes to aggregate the received layered coins into a short one.

Step 1: Layered Coin Aggregation Let an intermediate node $\mathcal{N}_m$ receive a layered coin which is constituted with a base layer $B\_layer = (S, RP, D, TS, TTL, SET_s, Sig_S, Cert_S)$ and multiple endorsed layer $E\_Layer_i = (\mathcal{N}_i, TS, SET_{\mathcal{N}_i}, Sig_{\mathcal{N}_i}, Cert_{\mathcal{N}_i})|1 \leq i \leq m-1$, where $\mathcal{S} \rightarrow \mathcal{N}_1 \ldots \rightarrow \mathcal{N}_i \ldots \rightarrow \mathcal{N}_m$ is the current forwarding path. For the simplicity of presentation, we assume that $M_0 = B||\mathcal{S}||RP||D||TS||TTL||SET_S$ and $M_i = B||B\_Layer||\mathcal{N}_i||TS||SET_{\mathcal{N}_i}$, where $1 \leq i \leq m-1$. Thus, the layered coin signatures can be represented as $Sig_S \leftarrow sk_S H_2(M_0)$ and $\{Sig_{\mathcal{N}_i} \leftarrow sk_{\mathcal{N}_i} H_2(M_i)|1 \leq i \leq m-1\}$. To aggregate the layered coin, node $\mathcal{N}_m$ can compute and obtain the aggregate signature: $Sig_{agg} \leftarrow Sig_S \prod_{i=1}^{m-1} Sig_{\mathcal{N}_i}$. In the subsequent bundle forwarding process, node $\mathcal{N}_m$ could transmit aggregate signature Sig_{agg} rather than transmit the signatures one by one. Therefore, the transmission overhead can be reduced.

Step 2: Layered Coin Batch Verification Given the aggregate signature Sig_{agg}, the message M_0 and $\{M_i|1 \leq$

$i \leq m-1\}$ on which it is based, public keys $PK_{\mathcal{S}}$ and $\{PK_{\mathcal{N}_i}|1 \leq i \leq m-1\}$, node $\mathcal{N}_m$ can verify the aggregate signature by checking if $\hat{e}(Sig_{agg}, P) = \hat{e}(PK_{\mathcal{S}}, H_2(M_0)) \prod_{i=1}^{m-1} \hat{e}(PK_{\mathcal{N}_i}, H_2(M_i))$.

It is observed that the computation cost that the intermediate node spends on verifying m signatures is reduced from $2m$ pairing operations to $m+1$ pairing operation, where pairing operation is the most computational expensive operation in SCI scheme. Thus, this batch verification can dramatically reduce the verification delay, particularly when verifying a large number of layered coins.

B. Merkle Hash Tree based SCI

In DTNs, when a message is large, it may not be possible to send the entire message at once. One possible solution is to split the message into smaller pieces and let each become its own bundle, or "fragment bundle", and send some pieces of a large message through the current link and rest of the message through another link later to make the best use of limited resources. To support layered coin based fragment authentication in SCI, one possible way is to make each fragment self-authenticating by attaching a layered coin to the end of each fragment separately. However, this approach may lead to a more serious performance issue since the intermediate nodes have to spend more computational efforts on verifying a growing number of signatures.

The Merkle tree [15] (also called binary hash tree) is a complete binary tree equipped with a function hash and an assignment Ω, which maps a set of nodes to a set of fixed-size strings. In a Merkle tree, the leaves of the tree contain the data, and the value of an internal tree node is the hash value of the concatenation of the values of its two children. Merkle tress have been applied in DTNs to realize efficient bundle authentication [16]. Here, we extend it to support efficient implementation of credit based incentive scheme, or an Merkle Hash Tree based SCI scheme (MHK-SCI).

Building Merkle Tree: To build a Merkle tree for our problem, the sender constructs N leaves $\{\Omega_i = H(F_i)|i = 1, \ldots, m\}$ with each leaf corresponding to a fragment bundle, where $\{F_i, |i = 1, \ldots, m\}$ refer to m fragments. The bundle sender then builds a complete Merkle tree with these leaves. The Ω value of each node is defined as the following:

$$\Omega(V) = H(\Omega(V_{left})||\Omega(V_{right})))$$

where we use V to denote an internal tree node, and V_{left} and V_{right} to denote V's two children. Fig. 3 shows an example to construct such a Merkle Tree. To add credit based incentive scheme to these bundles, the bundle sender only needs to generate a layered coin based on the root of the Merkle tree, which replaces the original bundle as the signed message.

Fragment Authentication with Merkle Tree based Incentive Scheme: To authenticate a particular fragment such as F_1, the intermediate node needs the set of hash value $\Omega_2, \Omega(B), \Omega(D)$ and the base layer which is a signature on the root $\Omega(E)$. The verifier can calculate each hash in the path from F_1 leaf node to the root node, and finally check the validity of raw coin. Note that to verify m fragments, it only performs one signature verification operation. In contrast, the original SCI scheme should verify m signatures in total.

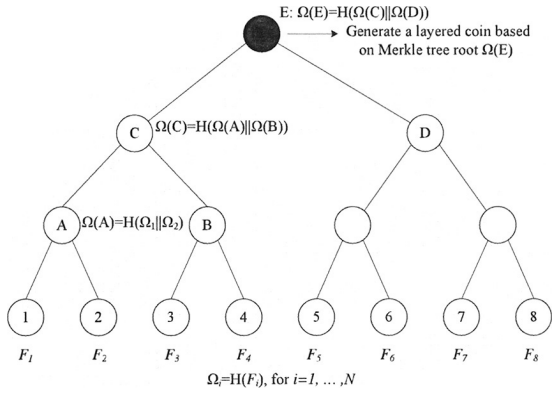

Fig. 3. An example of Merkle Tree Building

V. Performance Evaluation

In this section, we evaluate the performance of the proposed SCI scheme in terms of the resultant communication cost and computation overhead. The evaluated scheme includes the original SCI, Agg-SCI and MKH-SCI. Note that MKH-SCI can be seen as the integration of Agg-SCI and the Merkle Hash Tree method.

A. Communication Cost

One of the major advantages of SCI is the reduction of transmission cost. It is observed that the communication cost of layered coin is dominated by the size of supporting signatures generated by the intermediate nodes. To ensure the security of the protocol, the elements in $\mathbb{G}$ could be up to 160 bits. We summarize the approximated length of components of a layered coin in SCI shown in Table I, where BL represents base layer and EL is endorsed layer. Note that L refers to the number of copies adopted in the bundle forwarding scheme. In the following performance analysis section, we take $L = 4$ as an example.

TABLE I
THE SIZE OF EACH COMPONENT OF LAYERED COIN(BYTES)

BL	SRC	RP	D	TS	TTL	SET	Sig	Cert	Total
Size	4	10	4	4	4	4L	20	20	66+4L
EL	ID	Ts	SET	Sig	Cert	Total			
Size	4	4	4L	20	20	48+4L			

For m layered coins corresponding to m bundle fragments, each of which is accompanied with n endorsed coins, the total size of the layered coins (including both of the base layers and endorsed layers) without aggregation should be $82m + 62mn$. However, in our aggregate SCI scheme, the total size can be reduced to $82m + (42n + 20)m$ by taking advantage of aggregation signature. Under the same parameter, if every k fragments can be rebuilt with a Merkle hash tree, the total

size of MKH-SCI can be further reduced to $82m/k + (42n + 20)m/k$.

B. Computation Cost

The computation costs are measured by the most expensive pairing (Pair) and point multiplication (Pmul) operation. In the original SCI scheme, a Pmul operation is involved for each raw coin or endorsed coin generation while two pairing operations are necessary for verification. To investigate the performance of proposed SCI scheme, we first study the time for (Pmul) operation and Pair operation. We evaluate the delay of cryptographic operations on an Intel Pentium 4 3.0 GHz machine with 1 GB RAM running Fedora Core 4 based on cryptographic library MIRACL [17], which is shown in the Table II.

TABLE II
CRYPTOGRAPHIC OPERATIONS EXECUTION TIME

	Descriptions	Execution Time
T_{pmul}:	The time for one point multiplication in G	0.86 ms
T_{pair}:	The time for a pairing operation	4.14 ms

Here, we focus on the cost of verifying operation in SCI since the verification operation will be operated at each hop. Based on the execution time results, we have the verification cost for the $n-th$ intermediate node in the original SCI as $T_{\text{SCI}} = 2 * mn * T_{\text{pair}}$, where m and n refer to the number of fragments. In the aggregate SCI scheme, by using aggregate signature and batch verification technique, the verification cost can be reduced to $T_{\text{agg-SCI}} = m * (n+1)(T_{\text{pair}} + T_{\text{pmul}})$. The verification cost can be further reduced in the MKH-SCI scheme. Given every k fragments can be rebuilt with a Merkle hash tree, the total verification cost of MKH-SCI can be further reduced to $T_{\text{MKH-SCI}} = m/k * (n+1)(T_{\text{pair}} + T_{\text{pmul}})$.

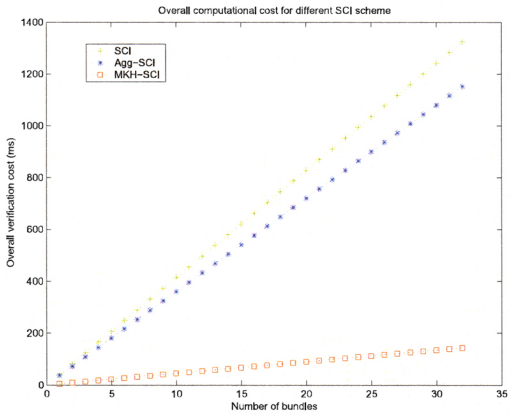

Fig. 4. Overall computational cost for different SCI scheme

From Fig. 4, we can observe that by adopting both of the MKH-SCI and Agg-SCI schemes, the computation cost of the SCI scheme can be dramatically reduced. It is worth to point out that the verification cost MHK-SCI shown in the Fig. 4 is calculated based on Agg-SCI, both of which are more efficient than the original SCI scheme.

VI. Conclusion

In this paper, we have proposed a secure credit-based incentive (SCI) scheme to stimulate cooperation in packet forwarding for delay tolerant networks. We have also proposed two efficiency optimization methods to reduce the transmission and computation overhead. As the future research, we will study the performance of SCI in a practical delay tolerant application scenario.

Acknowledgement

This research has been supported by a joint grant from Natural Science and Engineering Research Council (NSERC) and Research In Motion (RIM), Canada.

References

[1] A. Kate, G. Zaverucha and Urs Hengartner, "Anonymity and security in delay tolerant networks," Proc. of *SecureComm 2007*, Sept. 2007.
[2] H. Zhu, X. Lin, R. Lu, P.H. Ho, and X. Shen, "AEMA: An aggregated emergency message authentication scheme for enhancing the security of vehicular ad hoc networks," Proc. of *IEEE ICC'08*, Beijing, China, May 19-23, 2008.
[3] R. Lu, X. Lin, H. Zhu, P.H. Ho and X. Shen, "ECPP: Efficient Conditional Privacy Preservation Protocol for Secure Vehicular Communications," Proc. *IEEE INFOCOM'08*, Phoenix, AZ, USA, April 14-18, 2008.
[4] T. Spyropoulos, K. Psounis and C. S. Raghavendra, "Efficient routing in intermittently connected mobile networks: the multiple-copy cast," *IEEE/ACM Trans. on Networking*, vol. 16, no. 1, Feb. 2008.
[5] T. Spyropoulos, K. Psounis and C. S. Raghavendra, "Efficient routing in intermittently connected mobile networks: the single-copy cast," *IEEE/ACM Trans. on Networking*, vol. 16, no. 1, Feb. 2008.
[6] S. Marti, T. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," Proc. of *ACM Mobicom*, Boston, Massachusetts, August 2000.
[7] Q. He, D. Wu, and P. Khosla, "SORI: A Secure and Objective Reputation-Based Incentive Scheme for Ad Hoc Networks," Proc. of *WCNC 2004*, Atlanta, GA, Mar. 2004.
[8] Y. Zhang, W. Lou, W. Liu and Y. Fang, "A secure incentive protocol for mobile ad hoc networks," *ACM Wireless Networks*, vol. 13, no. 5, pp. 569-582, October 2007.
[9] S. B. Lee, G. Pan, J-S Park, M. Gerla, Songwu Lu, "Secure incentives for commercial ad dissemination in vehicular networks," Proc. of *MobiHoc'07*, Sept. 2007.
[10] R. Lu, X. Lin, H. Zhu, C. Zhang, P.H. Ho and X. Shen, "A Novel Fair Incentive Protocol for Mobile Ad Hoc Networks," Proc. *IEEE WCNC'08*, Las Vegas, Nevada, USA, March 31 - April 3, 2008.
[11] S. Symington, S. Farrell, H. Weiss and P. Lovell, "Bundle security protocol specification," draft-irtf-dtnrg-bundle-security-05.txt, work-in-progress, February 2008.
[12] H. Zhu, X. Lin, R. Lu, Pin-Han Ho and X. Shen, "SLAB: secure localized authentication and billing scheme for wireless mesh networks," to appear in *IEEE Trans. on Wireless Communications*.
[13] D. Boneh and M. Franklin, "Identity Based Encryption from the Weil Pairing," Proc. of Crypto'01, LNCS, vol. 2139, pp. 213-229, Springer-Verlag, 2001.
[14] D. Boneh, B.Lynn and H. Shacham, "Short signatures from the weil pairing," *Journal of Cryptology*, vol. 17, no. 4, pp. 297-319, 2004.
[15] R. Merkle, "Protocols for public key cryptosystems," Proc. of *IEEE S&P*, pp. 122-133, 1980.
[16] N. Asokan, K. Kostiainen, P. Ginzboorg, J. Ott and Cheng Luo, "Applicability of identity-based cryptography for disruption-tolerant networking," Proc. of *the First International MobiSys Workshop on Mobile Opportunistic Networking (MobiOpp)*, June 2007.
[17] Multiprecision Integer and Rational Arithmetic C/C++ Library (MIRACL).