Resisting Malicious Attacks via Secure Network Coding and Incentive Compatible Mechanism in Multihop Wireless Networks

Siguang Chen*
College of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing, China
Email: siguang1984@126.com

Meng Wu and Weifeng Lu
College of Telecommunications & Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing, China
Email: {wum, luwf}@njupt.edu.cn

***Abstract*—Network coding substantially improves the throughput of a network and possesses security superiority, but the security and performance of a network coding based wireless network are highly susceptible to malicious attacks such as pollution attack, dropping attack, lack of cooperation and selfish nodes collusion. This paper designs a secure network coding scheme that works in an adversarial environment, which can resist pollution attack, decrease the compromising probability of the message, and achieve tradeoff between security and performance by integrating the multipath and optimizing the coding packets allocation. Meanwhile, it also improves the fault tolerance of link failure or dropping attack by adding the finite redundancy coding packets. In addition, considering the selfish nodes of noncooperation communication, we design an incentive compatible protocol to stimulate forwarding packets and prove that only following the protocol honestly can obtain maximum utility. Furthermore, it is also proved that there is no collusion that can profit all colluding players. Finally, the simulation evaluation and security analysis confirm our theoretical results.**

***Index Terms*—Network coding, malicious attacks, multipath routing, game theory, collusion**

I. INTRODUCTION

Network coding is a novel approach proposed by Ahlswede et al. in [1] which allows network nodes to mix incoming data packets rather than simply forward them. Recently, network coding caught significant attention in the research community because compared to other traditional methods, it can improve network throughput, be done in a distributed network with low complexity and potentially possess security superiority [2], [3], [4].

However, most of previous papers mainly pursue good performance, while security aspects are disregarded, which induces numerous security vulnerabilities in system. Although several literature relevant to the security of network coding has existed, most of them focus on packet pollution attack. Current solutions to packet pollution attack in network coding systems can be categorized into cryptographic approaches, approaches based on network error correction coding, and information theoretic approaches [5].

Cryptographic approaches rely on augmenting the network coded packets with additional verification information; this allows intermediate nodes to verify the validity of coded packets and filter out polluted packets such as [6], [7], [8], [9]. But as we know, cryptography will consume mass resource in system, especially limiting its application on energy constraint system such as sensor networks.

Approaches based on a network error correcting coding theory are used by detecting and correcting corrupted packets in network coding systems [10], [11]. However, these approaches have limited error correcting ability.

Information theoretic approaches do not filter out polluted packets at intermediate nodes; instead, they add enough redundant information into packets which allows receivers to detect the presence of polluted packets [12], or use a distributed protocol which allows receivers to tolerate pollution and recover original packets [13], [14]. But these solutions do not consider the generalized scene of lossy links or interference from third party which will cause the failure of recovering original packets by destination node.

In addition to the pollution attack in network coding, there are several other attacks or threats such as: compromising attack, dropping attack, lack of cooperation, selfish nodes collusion attack and so on. Little research work was systemically done to resist all these attacks simultaneously. Especially on stimulate selfish nodes to forward packets and prevent nodes collusion for more payment. Conventionally, the measure

*Corresponding author. Tel.: +86 151 9598 8790.

doi:10.4304/jcp.5.11.1706-1714

of incentive compatible packet forwarding can be categorized into work use credit or virtual money to encourage [15], [16] and work based on reputation systems [17]. On resisting selfish nodes colluding attack, S. Zhong et al. [18] present a systematic study of collusion-resistant routing in noncooperative wireless ad hoc networks. Results from experiments show that their solution is collusion-resistant and has good performance. Whereas, these existing schemes are all designed for conventional routing protocol, and thus no longer fully fit for the routing based on network coding.

Currently, Incentive compatible opportunistic routing based on network coding was first studied by F. Wu et al. [19], who rigorously prove that their technique guarantees that it is a strict dominant strategy for each user node to behave honestly. They also design an enhanced protocol to prevent cheating not only in reporting loss rates but also in measuring them. Formally, they show that, with this enhanced protocol, it is a strict Nash equilibrium for each user node to behave honestly in both measuring and reporting. Such scheme is vulnerable to confront an adversarial environment, and there are no measures to resist malicious attacks.

Intuitively, the research of secure network coding is still preliminarily. As such, designing a systemic scheme protect against various attacks will be a major challenge.

The objective of this paper is to address the above issues. In this paper, we design a secure network coding system and incentive compatible mechanism for multihop wireless networks that can effectively limit malicious attacks. Our major contributions can be summarized as follows:

First, we are the first to systemically design a mixed secure network coding scheme for wireless networks to combat against various attacks such as: pollution attack, compromising attack, dropping attack, lack of cooperation, selfish nodes collusion attack and so on.

Second, our scheme decreases the compromising probability of the message by integrating the multipath and optimizing the coding packets allocation when design the secure network coding. Meanwhile, scheme improves the fault tolerance of link failure or dropping attack by adding the finite redundancy coding packets in source node.

Third, according to a large extent of cooperation among nodes that network coding based wireless network requires, we design an incentive compatible mechanism to stimulate intermediate nodes forwarding packets. We construct competition relationship among multiple paths for traffic transmission, and then prove that it is a Nash equilibrium for each player to follow the protocol faithfully. It means that no strategies can get more utility than in the Nash equilibrium state when other players are honest.

Fourth, we further prove that players collusion group C, the strategy profile s^* in which all players follow the protocol honestly is a group Nash equilibrium. This means that players cannot gain more utility from collusion than group Nash equilibrium the system converges to.

Finally, simulation results and security analysis show that our scheme improves the security of network coding based wireless network and has a good performance in the presence of above mentioned attacks.

The rest of this paper is organized as follows. The preliminaries and notations are presented in Section Ⅱ. In Section Ⅲ we describe the secure network coding scheme. Incentive compatible mechanism is given in Section Ⅳ. In Section Ⅴ, we discuss the security issues. Simulation results are showed and analyzed in Section Ⅵ, and conclusions are drawn in Section Ⅶ.

II. PRELIMINARIES AND MODEL

In this section, we first introduce the preliminary knowledge of game theory, follows by network model, adversarial model, and notations.

A. Nash Equilibrium

Nash equilibrium is a solution concept of a game involving two or more players, in which each player is assumed to know the equilibrium strategies of the other players, and no player has anything to gain by changing only his or her own strategy unilaterally. If each player has chosen a strategy and no player can benefit by changing his or her strategy while the other players keep theirs unchanged, then the current set of strategy choices and the corresponding payoffs constitute a Nash equilibrium [20], [21].

B. Network Model

In this paper, the topology of the wireless network is represented by the directed graph $G = (N, L)$, where N is the set of nodes and L is the set of directed links. Our analysis is based on a data session between source node S and destination node D. We use P to denote the set of paths between S and D. The number of shares from S to D that traverses along path $i \in P$, the reliability probability of link $l \in L$ and the compromised probability of path $i \in P$ are denoted by f_i, r_l, and p_i, respectively. Let d denote the packet delivery ratio and let q denote tradeoff coefficient between security and performance.

C. Adversarial Model

This model assumes omniscient adversaries, i.e., adversaries with the ability to eavesdrop all links, and know the encoding and decoding algorithm between S and D. They also can inject their corrupt packets into any links of the network and pretend they are part of the data flow from S to D. Whereas, we suppose the maximum number of packets that adversaries can inject is bounded. Moreover, we assume that the path set is computable by adversaries using traffic analysis and estimation [22].

D. Notations

Some important notations used in network coding scheme can be found in Table 1.

TABLE 1

Symbols used in network coding scheme

Notations	Definition
Matrix X	k original packets
Matrix A	z corrupt packets
m	Length of packet
k	Number of packets in a batch
z	Total number of corrupt packets that adversaries can inject in a batch

III. DESIGN OF SECURE NETWORK CODING

This design builds on the scheme of [13], which introduced distributed polynomial-time rate-optimal network codes that work in the presence of Byzantine nodes. But we add two major contributions to it: first, we decrease the compromising probability of the message by employing multipath and optimizing the coding packets allocation. Second, we improve the fault tolerance for link failure by adding redundancy coding packets.

Each block of the information stream is partitioned into k packets as a batch, where we focus only on one batch for simplicity. We only consider communication between a single S and a single D for simplicity, but our scheme can be generalized to multicast flow.

A. Optimal Coding Packet Allocation on Paths

In this subsection, we assume that totally there are $|P|$ node-disjoint paths, path 1, path 2 … path $|P|$, available from the source to the destination node. Our focus is how to select security paths and allocate message packets on these chosen disjoint paths so that the overall message security risk is minimized and the delivery ratio obtains an ideal value.

Definition 1. *We define our routing protocol as the dependent path routing protocol [23] which uses multiple node disjoint paths in which data traversing separate paths are jointly coded and secured.*

From definition 1, which means in our protocol, a set of coded packets must be jointly decoded in order to recover the original message.

We formalize the packet allocation in path set to minimize the routing security risk while limiting the delivery ratio under an ideal value. Nevertheless, the attackers make efforts to maximize this risk and unreachable ratio. These can be viewed as a following minimax optimization problem:

$$r^* = \min_{f_i, i\in P} \max_{p_i, i\in P} \sum_{i\in P} f_i (0.5 + 0.5 \prod_{l\in i} r_l) p_i \qquad (1)$$

$$\text{Subject to } \begin{cases} r_l > 0.7, \forall l \in L \\ \sum_{i\in P} p_i \le t, 0 \le p_i \le 1, \forall i \in P \\ \sum_{i\in P} f_i = n, f_i \ge 0, \forall i \in P \end{cases}$$

Where a captured link may be in a random position in a path, so constant 0.5 is from an average of $(1+\prod_{l\in i} r_l)$, i.e., $(1+\prod_{l\in i} r_l)/2$, $r_l > 0.7$ denotes only the reliability probability of link l (>0.7) can reserve in path set, n equals the number of transmitted packets in S, t equals number of attackers, $r = \sum_{i\in P} f_i (0.5 + 0.5 \prod_{l\in i} r_l) p_i$.

With respect to the above optimization and following model, we define these concepts as follows:

Definition 2. *If link $l \in L$ is compromised by attacker, then any message traverse the link l will be eavesdropped or modified.*

Definition 3. *The path $i \in P$ is compromised if and only if there is at least one link $l \in i$ for which is compromised.*

Definition 4. *The entire message transmission along path set P is compromised if and only if the compromised shares equal or bigger than k.*

We employ game theory to model our optimization problem as a noncooperative game between source node and attackers, denotes as G_1. The strategy sets of source node and attackers are $\{f_i, i\in P\}$ and $\{p_i, i\in P\}$, respectively. The source node is to minimize its utility function $U_s = r$ by f_i and attackers aim to maximize its utility function $U_A = r$ by p_i. A set of strategies is a Nash Equilibrium (NE) if no player can do better by unilaterally changing his or her strategy, thus, each strategy in a Nash equilibrium is a best response to all other strategies in that equilibrium. John Forbes Nash in his article Non-Cooperative Games was to define a mixed strategy Nash Equilibrium.

Theorem 1. *In game $G = \{S_1, S_2, ..., S_n; u_1, u_2, ..., u_n\}$, there are n players, S_i and u_i denote strategy set and utility function of player i respectively, if n is finite and S_i is a finite set of strategies of every player i, then prove that at least one (mixed strategy) Nash Equilibrium must exist in G [21],[24].*

Since players can choose from finitely many strategies and only two players in G_1, we can derive a (mixed strategy) Nash Equilibrium is exist in our game model.

Thus, we can transform our minimax optimization problem to following form:

$$\min_{f_i, i\in P} \max_{p_i, i\in P} r = U_s[f_i^*(i\in P), p_i^*(i\in P)] \qquad (2)$$

Where (f_i^*, p_i^*) is a mixed strategy NE of G_1.

Definition 5. *In our protocol, every choice path has at most one attacker.*

From Ref. [22], we can know that the path set is computable by attacker using traffic analysis and estimation, in addition, the choice paths are node disjoint paths, the intelligent choice of collusion attackers is attack one path at most one attacker.

Obviously, solve the minimax optimization problem equal to find the NE in G_1. We cite the following lemma of mixed strategy NE to compute the optimal solution.

Lemma 1. *Every action in the support of any player's NE mixed strategy yields the same payoff [21], [25].*

If $\sum_{i\in P} p_i = t$, then we apply the lemma 1, we can derive the following equation:

$$r^* = \min_{i\in P^*} nt / \sum_{i\in P^*}[1/(0.5+0.5\prod_{l\in i} r_l)] \quad (3)$$

$$\text{Subject to} \prod_{l\in i} r_l \geq 2t / \sum_{j\in P^*}[1/(0.5+0.5\prod_{l\in j} r_l)]-1 \quad (4)$$

Afterward, NE (f_i^* , p_i^*) can be figured out as follows:

$$f_i^* = n/[(0.5+0.5\prod_{l\in i} r_l)\sum_{j\in P^*} 1/(0.5+0.5\prod_{l\in j} r_l)] \quad (5)$$

$$p_i^* = t/[(0.5+0.5\prod_{l\in i} r_l)\sum_{j\in P^*} 1/(0.5+0.5\prod_{l\in j} r_l)] \quad (6)$$

Meanwhile, we define the n 's value as follows:

$$n = k\sum_{j\in P^*} 1/(0.5+q\prod_{l\in j} r_l)/|P^*| \quad (7)$$

Where $|P^*|$ denotes the number of paths which are selected to transmit shares.

We introduce an algorithm to find our path set P^* such that $r^* = \min_{i\in P^*} nt / \sum_{i\in P^*}[1/(0.5+0.5\prod_{l\in i} r_l)]$.

1: Input: path set P which was established by multipath routing finding algorithm [26].

2: For each path $i \in P$ do.

3: If path $i \in P$ satisfies the constraints (4), reserve this path in P .

4: Else: delete this path from P .

5: End for.

6: Return new P which $P^* = P$.

Algorithm 1: Optimal path set computation algorithm

B. Network Coding Scheme

The packet format is denoted as:

Batch identifier	Data segment

A packet contains m symbols from the finite field F_q , $j\,m$ symbols are added as redundancy. One batch can be taken as a matrix X as follows:

$$X = \begin{pmatrix} a_{11} & a_{12} & \mathbf{K} & a_{1(m-k)} & 1 & 0 & \mathbf{K} & 0 \\ a_{21} & a_{22} & \mathbf{K} & a_{2(m-k)} & 0 & 1 & \mathbf{K} & 0 \\ \mathbf{M} & \mathbf{M} & \mathbf{O} & \mathbf{M} & \mathbf{M} & \mathbf{M} & \mathbf{O} & \mathbf{M} \\ a_{k1} & a_{k2} & \mathbf{K} & a_{k(m-k)} & 0 & 0 & \mathbf{K} & 1 \end{pmatrix} \quad (8)$$

The i^{th} row in the matrix denotes the i^{th} packet in the batch, right side of X is a $k\times k$ identity matrix. The z packets that adversaries inject into each batch can be described as:

$$A = \begin{pmatrix} b_{11} & b_{12} & \mathbf{K} & b_{1m} \\ b_{21} & b_{22} & \mathbf{K} & b_{2m} \\ \mathbf{M} & \mathbf{M} & \mathbf{O} & \mathbf{M} \\ b_{z1} & b_{z2} & \mathbf{K} & b_{zm} \end{pmatrix} \quad (9)$$

1) Source node encoding

From matrix X we can deduce that the original message length of one batch is $(km-j\,m-k^2)$, the redundancy $j\,m$ column vector computed by solving the following matrix equation.

$$R\vec{X} = 0 \quad (10)$$

Where R is a $j\,m\times km$ matrix defined as a redundancy matrix. R is chosen from the independent and uniformly random symbol in finite field F_q and is known to participant nodes including adversaries. $\vec{X}$ is obtained by stacking the columns of X one after the other.

Source node encodes this batch X into n (the value of n according to the equation (7)) transmitted packets according to the following equation.

$$\begin{pmatrix} e_{11} & e_{12} & \mathbf{K} & e_{1k} \\ e_{21} & e_{22} & \mathbf{K} & e_{2k} \\ \mathbf{M} & \mathbf{M} & \mathbf{O} & \mathbf{M} \\ e_{n1} & e_{n2} & \mathbf{K} & e_{nk} \end{pmatrix} X = \begin{pmatrix} e_{11}B_1 + e_{12}B_2 + \mathbf{K} + e_{1k}B_k \\ e_{21}B_1 + e_{22}B_2 + \mathbf{K} + e_{2k}B_k \\ \mathbf{M} \\ e_{n1}B_1 + e_{n2}B_2 + \mathbf{K} + e_{nk}B_k \end{pmatrix} \quad (11)$$

Where B_i denotes the i^{th} row of matrix X , $e_{ij}(i=1,2\,\mathbf{K}\,n;\ j=1,2\,\mathbf{K}\,k)$ denotes the coefficients of random linear combination of original packets.

Finally, source node transmits n coding packets to destination node, the number of packets allocate on path i according to the value of f_i^* .

2) Intermediate node encoding

If $f_i^* \geq 2$, intermediate node of path i random linear combines the received packets and outputs corresponding number of transmitted packets, otherwise, intermediate node doesn't apply linear transform to received packet.

3) Destination node decoding

The decoding process is similar to the Ref. [13]. The transformation of one batch packets in the network can be showed as:

$$Y = [T \mid T_a]\begin{bmatrix} X \\ A \end{bmatrix} \quad (12)$$

Where X is the original packets sent by source node, Y is the received encoding packets, T denotes the linear transform from source to destination node, T_a denotes the linear transform from adversary to destination node.

Destination node selects $k+z$ linearly independent columns from Y , and denotes as Y_i . The corresponding columns in X and A can be denoted as X_i and A_i , respectively. Equation (12) can be written as:

$$Y_i = [T \mid T_a]\begin{bmatrix} X_i \\ A_i \end{bmatrix} \quad (13)$$

Therefore, Y can be denoted as:

$$Y = [T \mid T_a]\begin{bmatrix} X_i E \\ A_i E \end{bmatrix} \quad (14)$$

$$\text{Subject to} \quad Y = Y_i E$$

If $[T \mid T_a]$ is invertible, then we can obtain:

$$X = X_i E \quad (15)$$

We denote the first $m-k$ columns of matrix E as E', X as $X=[X_1 \quad X_2 \quad X_3]$, where X_1 corresponds to the first z columns of X, X_3 to the last k columns of X. Thus, we can write the equation (15) into

$$[X_1 \quad X_2] = X_i^{1} E_1' + E_2' \tag{16}$$

Where X_i^{1} denotes the first z columns of X_i, E_1' denotes first z rows of E' and E_2' denotes the last k rows of E'.

Combining the equation (10) and (16), we can get

$$B\begin{bmatrix} \vec{X}_1 \\ \vec{I} \\ X_2 \end{bmatrix} = \begin{pmatrix} \vec{E}_2' \\ -R_2 \vec{I} \end{pmatrix} \tag{17}$$

Where R_2 denotes the last $k*k$ columns of R and R_1 denotes the remainder of R, $\vec{X}_1$ denotes the vector obtained by stacking the columns of matrix X_1 one after the other. And matrix B can be denoted as:

$$B = \left(\begin{array}{cccc|c} (1-e_{1,1}')I & -e_{2,1}'I & \mathbf{K} & -e_{z,1}'I & \\ \mathbf{M} & \mathbf{M} & \mathbf{M} & \mathbf{M} & 0 \\ -e_{1,z}'I & -e_{2,z}'I & \mathbf{L} & (1-e_{z,z}')I & \\ \hline -e_{1,z+1}'I & -e_{2,z+1}'I & \mathbf{K} & -e_{z,z+1}'I & \\ \mathbf{M} & \mathbf{M} & \mathbf{M} & \mathbf{M} & \mathbf{I} \\ -e_{1,m-k}'I & -e_{2,m-k}'I & \mathbf{K} & -e_{z,m-k}'I & \\ \hline \multicolumn{4}{c|}{R_1} & \end{array}\right) \tag{18}$$

Where e_{ij}' denotes the $(i,j)^{th}$ entry of the matrix E_1', identity matrix I has dimension k, zero matrix $\mathbf{0}$ has dimension $zk \times k(m-z-k)$, identity matrix $\mathbf{I}$ has dimension $k \times (m-z-k)$.

Destination node received packets are related to source node and adversary's transmitted packets, if destination node receives at least $k+z$ packets from source node in one batch and matrix B has full column rank, then equation (18) has a unique solution. The destination node can decode the packets successfully even where adversaries exist.

IV. DESIGN OF INCENTIVE COMPATIBILITY MECHANISM

The former section assumes that the nodes are very cooperative, but in fact a selfish node may be unwilling to spend its resources on forwarding packets if there is no corresponding compensation, even though this node expects other nodes to forward its packets to the destination. Therefore, we should design an incentive mechanism to encourage the nodes to forward packets. This mechanism must maintain the low cost for transmitting message and at the same time effectively resist the nodes collusion.

A. Definition

We model the routing based on network coding as game strategy.

Definition 6. *A strategy profile s^* of all players' strategies is a Nash equilibrium if $s_i \neq s_i^*$ and*

$$u_i(s_i^*, s_{-i}^*) > u_i(s_i, s_{-i}^*) \tag{19}$$

Definition 7. *A strategy profile s^* is a group Nash equilibrium if for all nonempty subset C of players, for all profile s_c of strategies in C, there is a player $n_i \in C$ which*

$$u_i(s_C^*, s_{-C}^*) \geq u_i(s_C, s_{-C}^*) \tag{20}$$

Where s_i denotes the strategy that player i takes and s_{-i} denotes the strategies of all players except node i. Here, u_i is the utility function of player i. In this paper, we ignore the cost of control packets for the small overhead compare to the data packets.

B. Incentive Design

We integrate incentive mechanism into network coding based routing protocol to constitute a mixed secure network coding scheme. The aims of our mechanism are to enable the optimal strategies of all players to converge to an equilibrium state and maintain players' collusion can not obtain more utility than following protocol honestly.

First of all, incentive mechanism is necessary to encourage players to forward packets. Every player who participates in transmission packets must be paid to compensate the cost of forwarding packets by source node S. A player n_i 's utility is given by

$$u_i = f_j(\overline{p}_i - c_i),\ j \in P \tag{21}$$

$$\text{Subject to } \begin{matrix} \overline{p}_i = \overline{p}_i(a_i) \\ c_i = c_i(s_i) \end{matrix}$$

Where f_j denotes the number of data packets send by player n_i (including injecting packets from adversaries), $\overline{p}_i$ denotes the unit payment for player n_i, c_i is the real cost of player n_i sending a unit of data, a_i is the claimed cost of player n_i sending a unit of data. If player n_i is not in the path set P^*, then the values of both $\overline{p}_i$ and c_i are 0.

Second, incentive mechanism is important to further prevent behavior of dropping or dropping part of forwarding packets in players. Accordingly, a punishment will be applied to players' noncooperation behavior. If the malicious action of one player is detected, the path where this player lies will be removing from the transmission path set P^*. The malicious action can be detected via following formula:

$$f_i(0.5 + 0.5\prod_{l \in i} r_l)(1-p_i), (i \in P^*) \tag{22}$$

Where f_i denotes the pure data recovered from the adversaries inject data. Because destination node transmits the corresponding numerical value and the number of injecting data packets from adversary to source node as an acknowledge packet, source node can detect the malicious action by discovering large margin between each other.

Third, the incentive mechanism must preserve the player's claimed cost lies with in a reasonable range,

with only slightly higher than the real cost being allowed. We construct competition relationship among multiple paths. The path will be deleted from path set P^* if total amount of the claimed cost of this path are the highest. The algorithm is presented below.

1: Input: path set P^* which was established by algorithm 1.

2: Computing $i = \arg\max_{j \in P^*} A(j)/[h(j)-1]$, $A(j)$ denotes the total amount of claimed cost of path j , $h(j)$ denotes hops of path j .

3: Deleting path i from P^*.

4: Updating P^*.

Algorithm 2: Lowest cost path set optimization algorithm

Player can maximize its claimed cost to the extent that the path where this player lies won't be removed from transmission path set. If the sum of players' claimed cost in one path is quite high, then this path will be deleted from path set P^*. If one player $n_i \in j, j \in P^*$, the claimed cost a_i of this player can equal to or be only slightly higher than c_i. Then this player can gain the optimal claimed cost a_i^* through algorithm 3.

Initially, $a_i^* = c_i$, while $j \in P^*$.

1: For each player $n_i \in j$ do.

2: $a_i^* = a_i^* + a$.

3: If the value of $\sum_{n_i \in j} a_i^*$ is acceptable by the system which means path j will still exist in path set P^*, then go to step 2.

4: Recover the previous value of a_i^* , perform $a_i^* = a_i^* - a$.

5: End for.

6: Update a_i^* .

Algorithm 3: Optimal claimed cost obtain algorithm

Theorem 2. *In our scheme, payments are provided to incent every player n_i to forward packets. It is a Nash equilibrium for all players to obey protocol honestly.*

Proof: First of all, the players denote the nodes in the path set P^*. Here, s^* is the strategy profile of all players that follow the protocol honestly. and s_{-i}^* is the strategy profile of all players except player n_i . If player $n_i \in j, j \in P^*$ selects strategy s_i^* and claims cost a_i^* , then the expected utility of player n_i is

$$\begin{aligned} u_i(s_i^*, s_{-i}^*) &= f_j(\overline{p}_i - c_i) \\ &= f_j[\overline{p}_i(a_i^*) - c_i(s_i^*)] \end{aligned} \tag{23}$$

If player $n_i \in j, j \in P^*$ selects strategy s_i ($s_i \neq s_i^*$) and claims cost a_i ($a_i \neq a_i^*$), then the expected utility of player n_i is

$$\begin{aligned} u_i(s_i, s_{-i}^*) &= f_j(\overline{p}_i - c_i) \\ &= f_j[\overline{p}_i(a_i) - c_i(s_i)] \end{aligned} \tag{24}$$

Payment function $\overline{p}_i$ increases with the value of claimed cost rise. According to algorithm 3, we can deduce that $a_i^* > a_i$, thus, $\overline{p}_i(a_i^*) > \overline{p}_i(a_i)$. In order to maximize the utility, player will make every endeavor to reduce the real cost of itself, therefore, the optimal strategy $c_i(s_i^*) < c_i(s_i)$. Combining the above two results, we can educe the following expression:

$$u_i(s_i^*, s_{-i}^*) > u_i(s_i, s_{-i}^*) \tag{25}$$

This means strategy profile s^* of all players is a Nash equilibrium. □

C. Collusion Analysis

Theorem 3. *For players collusion group C, the strategy profile s^* , in which the protocol that all players honestly follow is a group Nash equilibrium.*

Proof: This theorem means that the players' collusion can not obtain more payment than honestly following protocol. We divide the utilities' calculation into two cases.

Case 1: $a_i > a_i^*, n_i \in C$, if $a_i > a_i^*$, from algorithm 3 we can find the path where the player n_i lies can be found then deleted from path set P^* , when P^* becomes to be an empty set, system will initiate the route discovery process to dynamically find new paths to destination node. Thus, we can get

$$\begin{aligned} &u_i(s_C^*, s_{-C}^*) \geq 0 \\ &u_i(s_C, s_{-C}^*) = 0 \\ &\Rightarrow u_i(s_C^*, s_{-C}^*) \geq u_i(s_C, s_{-C}^*) \end{aligned} \tag{26}$$

Case 2: $a_i \leq a_i^*, n_i \in C$, if $a_i \leq a_i^*$, we can get

$$\begin{aligned} &\overline{p}_i(a_i) \leq \overline{p}_i(a_i^*) \\ &u_i(s_C^*, s_{-C}^*) = f_j[\overline{p}_i(a_i^*) - c_i(s_i^*)] \\ &u_i(s_C, s_{-C}^*) = f_j[\overline{p}_i(a_i) - c_i(s_i)] \\ &\Rightarrow u_i(s_C, s_{-C}^*) \leq u_i(s_C^*, s_{-C}^*) \end{aligned} \tag{27}$$

Combining the above two expressions, we can see that players can not increase their utility by colluding. □

V. SECURITY ANALYSIS

In this section, we will analyze the security of our scheme via evaluating its robustness in the presence of some attacks described above.

Compromising attack: Multiple paths are used in our scheme, which makes the eavesdropping attack maximally difficult as the attackers would have to eavesdrop on all possible paths. The optimal coding packets allocation mechanism is implemented in our network coding scheme, which further decreases the compromising risk of transmission information.

From the perspective of the quantitative analysis, simulation results of next section confirm the above stated theoretical analysis. The detailed presentation of numerical results can be referred to next section (subsection A). Meanwhile, as data transmission is based on network coding, attackers also have to decrypt the intercepted data. It is difficult to successfully decrypt

data unless there is enough eavesdropped information.

Pollution attack: This attack means that adversaries inject corrupted packets to our transmission links. If the size of corrupted packets is close to or exceeds the network capacity, it is difficult to resist this type of attack. But using our network coding scheme, system can endure finite pollution attack and retrieve original information from mixed corrupted information.

Whereas the fault tolerance capabilities sacrifice partial throughput of the network, the throughput that can be achieved by our scheme is upper-bounded by the information theoretic optimal rate of $f-2z-(n-k)$, where f is the network capacity between source and destination nodes. The payload of one batch packets is $km-j\,m-k^2$. We can see that resisting pollution does reduce throughput and payload of packet because pollution packets are injected and the error-correcting decoding algorithm is used. However, this is the tradeoff. We argue that for security critical applications, the concern of network efficiency might not be as critical as network security.

Dropping attack: Dropping attack falls into two cases: one of them indicates the attacker's malicious action. As we add the redundancy coding packets in source node, our coding scheme has well fault tolerance capability against dropping attack. The maximum number of packets that allow to be dropped is *n-k* when the links of network remain lossless.

The other indicates the inner nodes' selfish action. An incentive compatible mechanism is designed to defend against selfish action. The last section (subsection B) demonstrates that only honestly behaving can gain maximum payment from source node; meanwhile, the simulation results in next section (subsection B) also show that forwarding packets can maximize the node's utility. As such, this incentive mechanism can prevent the inner nodes' dropping attack.

Collusion attack: The attack means that inner nodes collude to cheat more payment from source node. The theorem 3 shows that our incentive measure can effectively resist collusion attack. The simulation figure also validates theorem 3 in next section (subsection B). In summary, our incentive compatible mechanism can prevent the nodes collusion attack.

VI. SIMULATION RESULTS

Our simulation is implemented in OPNET [27]; the network coverage area is a 1000m*1000m square with 100 mobile nodes. Each node has radio power range of 200m. The channel capacity is 2 Mbps. The IEEE 802.11 wireless LAN standard is used as the MAC layer protocol. The network topology is generated randomly. We evaluate the performance and security of network by the following metrics:

• Utility per batch: the total payment received from source node minus the real cost of player forwarding packets per batch.

• Security risk: the probability of message which is compromised.

• Delivery ratio: the ratio of packets reaches *D* to the total packets generated at the *S*.

A. Security Risk and Delivery Ratio

In this setting, the reliability of Link $l \in L$ is generated through a normal distribution $N(m=0.7, s\,2=0.2)$. The simulation assumes that our transmission exist in the worst case in which the attackers know the path set and every path allocates at most one attacker. We observe the fact that the number of maximum paths in this experiment is around 6 and the maximum routing overhead is around 8 (routing overhead: the ratio of the total hop count from source to destination node in our multipath routing to the minimum hop count in single-path routing).

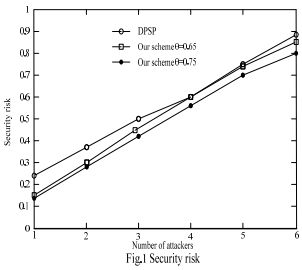

Fig.1 Security risk

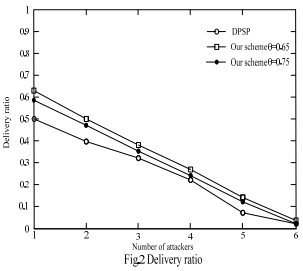

Fig.2 Delivery ratio

"DPSP" denotes the algorithm of Ref. [28]. "Our scheme" denotes the protocol proposed in this paper. Fig.1 depicts the security risk at different number of attackers. Figure shows that the security risk rises significantly with the increase of the number of attackers, and the risk decreases with the q rising from 0.65 to 0.75. Obviously, the security risk of our scheme is less than DPSP, and these results are not the best of our algorithm. We can obtain a better value of security risk by adjusting tradeoff coefficient q, as the theoretic analysis of equation (7) the security risk and delivery ratio will decrease with the increase of q. The simulation results demonstrate that our scheme further enhances the security of the network subject to the worst case. It also confirms that the message will nearly be compromised if the number of attackers $t \geq |P^*|$.

We consider that the attack not only compromise our packets, but also disrupt the communication. Fig.2 shows that the delivery ratio reduces saliently with the increase of the number of attackers, and also decreases with the increase of q. The delivery ratio of our scheme is still higher than DPSP, although in order to keep the low security risk, we sacrifice part of the delivery ratio. Combination Fig.1 and Fig.2 we can deduce that our scheme is more robust and flexible than DPSP's; we can also derive that choosing the reliability path and maximizing the paths can avoid or at least reduce the compromising probability and increase the delivery ratio.

In conclusion, simulation results show that the design of the scheme further improves routing security and fault tolerance, and helps to flexibly achieve tradeoff between security risk and delivery ratio via q. Moreover, since the scheme is based on network coding and only performs several linear combination operations on packets, it has lower complexity than cryptography-based network

coding scheme and conventional routing scheme.

B. Player's Utility

We assume that one player n_i 's optimal strategy is $a_i^* = 0.3/unit, c_i = c_i(s_i^*) = 0.26/unit$; when this player deviates from protocol, we suppose the claimed cost a_i generates from a uniform distribution $U[0.25, 0.35], a_i \neq 0.3/unit$ and $c_i = c_i(s_i) = 0.26/unit$, simulation implements 50 times with randomly selected a_i 's value from uniform distribution. Other players honestly follow the protocol. We set batch size k to 30 packets and $a = 0.001$.

Fig.3 demonstrates the utility of different strategies of one randomly selected player. Optimal strategy denotes the strategy of Nash equilibrium state, simulation results show that the utility gain of any players deviating from protocol is always less than utility gained via honestly obeying the protocol. The negative utility means the claimed cost is less than the real cost for forwarding packets. Fig.3 also further confirms that no strategy can get more utility than in the Nash equilibrium state when other players are honest.

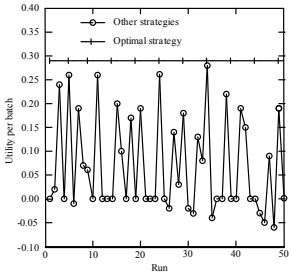

Fig.3 One player' s Utility

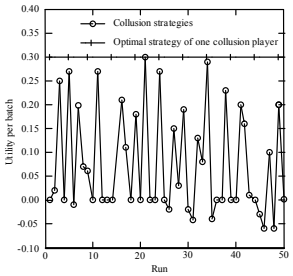

Fig.4 Random one collusion player' s Utility

In the second setting, we suppose that 20% players collude which implies that these players deviate from our protocol. We also assume the claimed cost of collusion players obeys uniform distribution $U[0.25, 0.35]$ and $c_i = c_i(s_i) = 0.26/unit$. The optimal strategy takes $a_i^* = 0.3/unit$.

Fig.4 plots one randomly selected colluding player's utility, only when the collusion strategy $a_i = a_i^*, n_i \in C$ the utility can reach to top. Experimental results also confirm that no collusion strategy can obtain more utility than group Nash equilibrium state. There is no collusion strategy that can make all colluding players benefit. On the contrary, other players do not reside in paths where colluding player resides will gain more utility for carrying more weight per batch.

C. Collusion Effect on Security Risk and Delivery Ratio

In this setting, the reliability of Link $l \in L$ is generated through a normal distribution $N(m = 0.7, s\,2 = 0.2)$ and $q = 0.75$. We demonstrate the collusion action can impact on data security transmission and delivery ration. Players collude to maximize their profit via deviation, which means players' claimed cost $a_i > a_i^*$, is a little bit different from players collusion mentioned above. Two different proportions of colluding players are 25% and 40%. Meanwhile, we observe the fact that the number of maximum paths in this experiment is around 6 and the maximum routing overhead is around 7.

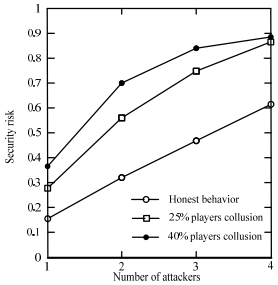

Fig.5 Effect on security risk

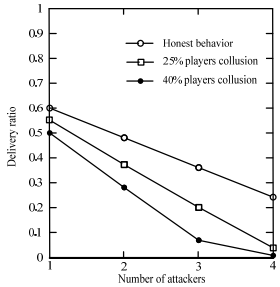

Fig.6 Effect on delivery ratio

Fig.5 illustrates the players' collusion effect on data transmission security when using our scheme. In honest behavior, players follow the protocol faithfully where the node disjoint paths are around 6. For the sake of increasing their utility, the cost that claimed by the colluding players emerges simultaneously $a_i > a_i^*$, then according to the algorithm 3 the paths where colluding players lie will be deleted from the path set. Therefore, the number of paths used to send data is less than 6 while 25% players collude. Thus, the data transmission security risk of 25% players' collusion is higher than no players' collusion. With the increase of collusion rate, the security risk is proportioned to this rate. Consequently, the effect on security risk of 40% players' collusion is highest.

The delivery ratio is calculated by following equation:

$$d = [\sum_{i \in P^*} f_i \prod_{l \in i} r_l (1 - p_i)] / k$$

Fig.6 shows the colluding players' effect on the delivery ratio if using our scheme. It can be seen from the numerical results that with the increase of collusion rate, the delivery ratio will decrease correspondingly. Because cutting down the transmission paths can augment the probability of being attacked, which type of attack not only compromises our data but also interrupts our transmission, the data delivery ratio will be reduced. Although players' collusion has adverse effect on our scheme's performance and security, from Fig.4 we can still obtain that the utility of colluding players is less than obeying the protocol. Hence, no players are willing to take collusion action for nonprofit results.

VII. CONCLUSIONS

In order to maintain the security and efficient performance of wireless networks, the network coding is selected rather than the conventional routing pattern. But random network coding is vulnerable to new threats as well as some malicious attacks. Thus, we design a secure network coding scheme, and conceive an incentive compatible mechanism to stimulate forwarding packets according to the selfish behavior of nodes. From section Ⅲ and section Ⅳ, it is obvious that our scheme achieves security for wireless networks based on network coding. Experiment results and security analysis also validate our theory.

ACKNOWLEDGMENT

This work was supported by the National Grand Fundamental Research 973 Program of China (2011CB302903), the Key Program of Natural Science for Universities of Jiangsu Province (10KJA510035), the Science and Technology Innovation Group Foundation of Jiangsu Province ("Qing and Lan" Project), and the Postgraduate Innovation Project Foundation of Jiangsu Province (CX10B-194Z).

REFERENCES

[1] R. Ahlswede, N. Cai, S. R. Li, and R. W. Yeung, "Network information flow," IEEE Transactions on Information Theory, vol. 46, no. 5, pp. 1204-1216, Jul. 2000.
[2] R. Koetter and M. Medard, "An algebraic approach to network coding," IEEE/ACM Transactions on Networking, vol. 11, no. 5, pp. 782-795, Oct. 2003.
[3] T. Ho, R. Koetter, M. Medard, D. Karger, and M. Effros, "The benefits of coding over routing in a randomized setting," in Proc. IEEE International Symposium on Information Theory (ISIT 2003), 2003, pp. 442.
[4] D. S. Lun, M. Medard, and R. Koetter, "Efficient operation of wireless packet networks using network coding," in Proc. International Workshop on Convergent Technologies (IWCT), 2005, pp. 1-5.
[5] J. Dong, R. Curtmola, and C. Nita-Rotaru, "Secure network coding for wireless mesh networks: Threats, challenges, and directions," Computer Communications, vol. 32, no. 17, pp. 1790-1801, Nov. 2009.
[6] M. Krohn, M. Freedman, and D. Mazieres, "On-the-fly verification of rateless erasure codes for efficient content distribution," in Proc. Symposium on Security and Privacy, 2004, pp. 226-239.
[7] E. Kehdi and B. Li, "Null keys: limiting malicious attacks via null space properties of network coding," in Proc. IEEE INFOCOM, 2009, pp. 1224-1232.
[8] F. Zhao, T. Kalker, M. Medard, and K. Han, "Signatures for content distribution with network coding," in Proc. IEEE International Symposium on Information Theory (ISIT 2007), 2007, pp. 556-560.
[9] Z. Yu, Y. Wei, B. Ramkumar, and Y. Guan, "An efficient signature scheme for securing xor network coding against pollution attacks, " in Proc. IEEE INFOCOM, 2009.
[10] R. Koetter and F. R. Kschischang, "coding for errors and erasures in random network coding," IEEE Transactions on Information Theory, vol. 54, no. 8, pp. 3579-3591, Aug. 2008.
[11] D. Silva, F.R. Kschischang, and R. Koetter, "A rank-metric approach to error control in random network coding," IEEE Transactions on Information Theory, vol. 54, no. 9, pp. 3951-3967, 2008.
[12] T. Ho, B. Leong, R. Koetter, M. Medard, M. Effros, and D. Karger, "Byzantine modification detection in multicast networks using randomized network coding," in Proc. IEEE International Symposium on Information Theory (ISIT 2004), 2004, pp. 144.
[13] S. Jaggi, M. Langberg, S. Katti et al., "Resilient network coding in the presence of byzantine adversaries," IEEE Transactions on Information Theory, vol. 54, no. 6, pp. 2596-2603, Jun. 2008.
[14] L. Nutman and M. Langberg, "Adversarial models and resilient schemes for network coding," in Proc. IEEE International Symposium on Information Theory (ISIT 2008), 2008, pp. 171-175.
[15] N. B. Salem, L. Buttyan, J. P. Hubaux, and M. Jakobsson, "A charging and rewarding scheme for packet forwarding in multi-hop cellular networks," in Proc. ACM MOBIHOC, 2003, pp. 13-24.
[16] S. Zhong, L. E. Li, Y. G. Liu et al., "On designing incentive-compatible routing and forwarding protocols in wireless ad-hoc networks," Wireless networks, vol. 13, no. 6, pp. 799-816, Dec. 2007.
[17] J. J. Jaramillo and R. Srikant, "Distributed and adaptive reputation mechanism for wireless ad-hoc networks," in Proc. ACM MOBICOM, 2007, pp. 87-98.
[18] S. Zhong and F. Wu, "A collusion-resistant routing scheme for noncooperative wireless ad hoc networks," IEEE/ACM Transactions on Networking. This article has been accepted for inclusion in a future issue of this journal, Digital Object Identifier 10.1109/TNET. 2009.2030325.
[19] F. Wu, T. Chen, S. Zhong et al., "Incentive-compatible opportunistic routing for wireless networks," in Proc. ACM MOBICOM, 2008, pp. 303-314.
[20] N. John, "Equilibrium points in n-person games," in Proc. the National Academy of Sciences, vol. 36, no. 1, pp.48-49, 1950.
[21] M. J. Osborne and A. Rubinstein, A Course in Game Theory, MIT Press, Cambridge, Mass, USA.
[22] G. Danezis and R. Clayton, "Introducing Traffic Analysis," Digital Privacy: Theory, Technologies, and Practices, A. Acquisti, S. Gritzalis, C. Lambrinoudakis, and S. di Vimercati, eds., Auerbach, Dec. 2007.
[23] P. Tague, D. Slater, J. Rogers et al., "Evaluating the vulnerability of network traffic using joint security and routing analysis," IEEE transactions on dependable and secure computing, vol. 6, no. 2, pp. 111-123, Apr-Jun. 2009.
[24] N. John, "Non-Cooperative Games," The Annals of Mathematics, vol. 54, no. 2, pp. 286-295, 1951.
[25] L. Chen and J. Leneutre, "On multipath routing in multihop wireless networks: security, performance, and their tradeoff," Eurasip journal on wireless communications and networking, vol. 2009, pp. 1-13, 2009.
[26] S. Chen and M. Wu, "Secure multipath routing based on secret sharing in mobile Ad Hoc networks," in Proc. IEEE International Conference on Network Infrastructure and Digital Content, Nov. 2009, pp.119-123.
[27] http://www.opnet.com
[28] P. Papadimitratos, Z. J. Haas and E. G. Sirer, "Path set selection in mobile ad hoc networks," in Proc. the International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '02), 2002, pp. 1–11.

Siguang Chen was born in 1984. Chen received his B.S. and M.S. degree in computer science from East China Jiaotong University, Jiangsu University of Science and Technology in 2006 and 2008, respectively. He is currently pursuing his Ph.D. in the Department of Information Security at Nanjing University of Posts and Telecommunications, Nanjing, China. His research interests are in the area of secure network coding, routing security, and game theory applications in wireless networks.

Meng Wu was born in 1963. He received his B.S., M.S. and Ph.D. degrees in communication engineering and computer science from Zhenjiang University, Shanghai Jiaotong University, Southeast University, in 1985, 1990 and 1993, respectively. Currently, he is a professor of Nanjing University of Posts and Telecommunications. His main research areas are routing security of wireless networks and secure network coding, etc. He has published more than 100 papers in journals and international conferences.

Weifeng Lu received his Ph.D. degree in communication and information engineering from Nanjing University of Posts and Telecommunications, Nanjing, China, in 2007. He is an Assistant Professor with the College of Telecommunications & Information Engineering, Nanjing University of Posts and Telecommunications. His research interests include network coding and security in wireless communications.