Two Layer Secure Network Coding - (2-LSNC)

Mehdi M. Hassanzadeh, Mohammad Ravanbakhsh, and Øyvind Ytrehus
Dept. of Informatics, University of Bergen,
N-5020 Bergen, Norway
Email: {mehdi.hassanzadeh,mohammad.ravanbakhsh,oyvind}@ii.uib.no

***Abstract*—Two important parameters for network users are security and cost of network resource utilization. From a cost perspective, network coding promises a higher gain compared to ordinary routing. In this paper, a new model for secure network coding is proposed. This model is named *Two Layer Secure Network Coding* or 2-LSNC. The advantage of our new model is three-fold: 1) The number of links required for the wiretapper to extract the secret is improved. 2) For scalability analysis we have established a metric called *level of security*, and we demonstrate that our model scales well with respect to this metric. 3) The tradeoff between cost and the level of security is analyzed. In our model, this cost is lower than in Cai and Yeung's model [3] when the network size and the number of sinks reaches a critical point. Our approach is compared to other models by simulation.**

Keywords: network coding, secure network coding, secret sharing.

I. Introduction

Network coding was introduced by Ahlswede *et al.* in [1]. In this communication paradigm, network nodes are allowed not only to forward unmodified packets, as routers in a classical store-and-forward network are restricted to, but also to modify and combine incoming packets prior to forwarding them. In [2], Li *et al.* proved that linear network coding suffices to *multicast* information from a single source to a fixed set of receivers at a rate equal to the minimum (over all receivers) of the min-cut of the network flow from the source to each receiver.

In [3], Cai and Yeung presented a certain security problem that can be alleviated by network coding. They introduced a model for secure linear network coding that achieves perfect information security against a wiretapper who can eavesdrop on a limited number of network links. Their method is based on using secret sharing ideas combined with constraints on the field size for secure network coding. The constraints were based on the network topology. The fundamental limit for this approach lies in the required field size for secure network coding. Due to the requirement for a large field size, this model is computationally impractical. In the rest of paper, we refer to this model as the *C&Y model*.

In [4], Feldman *et al.* showed that finding a matrix for the construction of an optimal secure network code is equivalent to finding a linear code with certain generalized distance properties. Additionally, they improved the lower bound of the field size. They showed that if we give up a small amount in overall capacity, a secure random network coding scheme is achievable with comparably smaller field size than the C&Y model. In this model, increasing the level of security results the growth of the required field size. This property makes these models inefficient in practice.

In [5], Lima et al considered a different approach to providing secure network coding. They showed that linear network coding is sufficient to set up a secure network coding scheme when a network is constructed by imposing a limit on the input degree of nodes. Additionally, they showed that the security of the model depends fully on the network topology, which is one of its weak points [5]. Also in this model, increasing the level of security results in the growth of the field size and the number of random processes. Similar to the previous models, this property makes this model inefficient in practice [5]. The model in [5] is used for providing security in sensor networks [7].

In this paper, a new model for secure network coding is proposed. This model is named *Two Layer Secure Network Coding* or 2-LSNC. By this model, we improve the number of links that a wiretapper needs to access in order to extract the secret message, the level of security and the cost for increasing the level of security. Our model is *scalable*, which means that the efficiency improves as network size grows. We observed from our simulations that this property does not hold for previous approaches. Improvement in the security in [3] and [5] is a function of field size. In other words, resistant against a more powerful wiretapper can be achieved by using larger field size. However, our method has no constraint on the field size and it is only necessary to use a field that gives a feasible network coding solution.

The paper is organized as follows: Section II starts with an introduction about network coding and secret sharing. This section follows by a detailed description of the C&Y model. A new model for secret sharing is presented in Section III. Our proposed model for secure network coding (2-LSNC) is described in section IV. Section V contains the simulation results. The paper concludes in Section VI.

II. Preliminaries

A. The Network Coding Model

We represent a communication network with a directed graph $G = (V, E)$, where V is the set of nodes (a single source, routers, and sinks/receivers) and E is the set of edges or links. All nodes are shown with a number that range from 0 (which, for convenience, is reserved for the source node) to $|V|-1$. Each link (i, j) represents a lossless point-to-point link

978-1-4244-2751-2/08/$25.00 ©2008 IEEE

from node i to node j. $\Gamma_I(i)$ and $\Gamma_O(i)$ refer to the number of incoming and outgoing links for node i respectively.

The goal of a *multicast session* is to convey a sequence of information symbols generated at a single source[1] to a set T of nodes (referred to as the set of *sinks*). The maximum amount of transferable directed flow between a source and a sink in a directed graph is known as the *max-flow*, which, by the celebrated *max-flow theorem* is identical to the *min-cut* between the source and the sink. In a multicast where a source node sends information to all sink nodes; it is possible to reach *max-flow* for each sink by applying network coding [1]. Without network coding this is not always possible.

In network coding intermediate nodes not only copy and forward their received packets but can also combine them. Establishing a predetermined network code consists of two steps: 1) Finding a subgraph for transferring the *max-flow* of information, and 2) Given this subgraph, finding a specific method of *encoding*; that is, a detailed procedure for how each node shall combine its received packets at its outgoing links. In [8] a deterministic method is proposed for encoding and in [9] this algorithm is extended for subgraphs with cycles (flow cycles).

B. Secret Sharing

In cryptography, *secret sharing* refers to any method for distributing a secret (with a *dealer*) amongst a group of n participants (*players*), each of which is allocated a share of the secret. In an (n, t_{ss})-threshold secret sharing scheme, the secret can only be reconstructed if at least t_{ss} shares are combined together. Secret sharing was invented independently in 1979 by A. Shamir [10] and G. Blakley [11]. The goal of *secret sharing* is to divide a secret S into n shares $s_1, \ldots, s_n$ in such a way that:

1) Knowledge of any t_{ss} or more s_i makes S *easily* computable.
2) Knowledge of any $t_{ss}-1$ or fewer s_i leaves S *completely undetermined*.

In [10] a simple (n, t_{ss})-threshold scheme based on polynomial interpolation is proposed. The polynomials can be replaced by any other collection of functions which are easy to evaluate and to interpolate.

Consider the *trivial* (n, n)-threshold scheme, i. e. t_{ss} is equal to n. In this scheme, $n-1$ random number $(r_1, \ldots, r_{n-1})$ are generated as $n-1$ shares and for the last share we have: $r_n = S \oplus r_1 \oplus r_2 \oplus \ldots \oplus r_{n-1}$, where $\oplus$ is any discrete formal addition. It is straightforward to see that S could be reconstructed with knowledge of all the shares, while no subset of $n-1$ or fewer shares can reconstruct the secret S.

C. Description of Cai and Yeung's model

In this subsection, the C&Y model is described. Suppose a wiretapper has access to a limited number of links denoted by μ. Since the wiretapper can access any subset of μ links, specifically, $h = R - \mu$ symbols can be transmitted securely, where

$$R = \min_{t \in T} \{\textit{max-flow}(t)\} \tag{1}$$

Interpretation: For a sink node $t \in T$, if μ channels in the network are wiretapped, the number of "secure paths" from the source node to T is still at least $h = R - \mu$ (although the legal network users do not know *which* paths are secure.) Consequently, a vector of h symbols $x = (x_1, \ldots, x_h)$ can be sent securely to each destination. Hence, μ independent random numbers $(r_1, \ldots, r_\mu)$ are generated and concatenated to X to construct the vector $y = (x_1, \ldots, x_h, r_1, \ldots, r_\mu)$. Finally, the source node uses a matrix $\mathbf{M}$ to produce the vector $c = \mathbf{M}y$ and multicast the vector c to all sink nodes by using network coding.

III. Concatenated Threshold Secret Sharing

In this section we present a new type of secret sharing. It is designed by serial concatenation of the trivial scheme and another threshold scheme. A secret S is first divided into n shares $(s_1, s_2, \ldots, s_n)$ by use of the (n, n)-secret sharing scheme, so that a player needs access to all shares $(s_1, s_2, \ldots, s_n)$ for reconstructing the secret S.

In the next step, each share s_i is divided into mi shares by use of an $(m_i, t_{ss,i})$-threshold secret sharing scheme. The parameters m_i and $t_{ss,i}$ can be different for each s_i but for simplicity they are considered to be constant and equal to m and t_{ss} respectively. Therefore, nt_{ss} shares are needed for reconstructing the secret S, and each set of t_{ss} shares must be received from the corresponded s_i. We refer to this model by (n, m, t_{ss})-Concatenated Threshold Secret Sharing, or CTSS.

The structure of CTSS makes it to be useful for our model for constructing a secure network coding which is presented in section IV. It is shown that the CTSS scheme improves the level of security compared to the C&Y model.

IV. Proposed Secure Network Coding Model

In this section, we present a new model for secure network coding that can be applied to any network which has a feasible solution for network coding. In other words, no extra condition is enforced on the network in order to guarantee the existence of a feasible solution. We will demonstrate that the new model compares favorably to the C&Y model with respect to the level of security described in section V-B. We employ the *Concatenated Threshold Secret Sharing scheme* introduced in subsection III. Since this new scheme of secret sharing uses 2 steps of secret sharing in the source node and its neighbors, we refer to our new model as *2-Layer Secure Network Coding* or 2-LSNC.

In Subsection IV-A we explain how the CTSS scheme is used in the new scheme. Then we discuss the optimization process for the 2-layer secure network coding in Subsection IV-B.

[1]The single source scenario can easily be generalized to one with multiple sources. For simplicity and convenience, we consider the single source scenario only in this paper.

A. The Concatenated Threshold Secret Sharing for 2-LSNC

From now on, the *first layer of the network* refers to the source node which is indexed by 0, while the *second layer* refers to all nodes which receive input from the source directly. The nodes of the second layer are indexed from 1 to $\Gamma_O(i)$. We assume that the links $(0, i)$ are secure for $i = 1 \dots \Gamma_O(i)$. This assumption is practical in many networks and in our comparisons in section V it is considered for all methods. For convenience, we assume that $\Gamma_O(i)$ is the same for all nodes in the second layer ($\forall i : \Gamma_O(i) = \Gamma_O$).

A CTSS scheme with parameters (n, Γ_O, t_{ss}) is used in the two layers of the network, where the source node plays the role of the dealer and the sinks are the players. In other words, a secret S is divided into n shares $(s_1, s_2, \dots, s_n)$ in the first layer where $n \leqslant \Gamma_O(0)$. Thus, eventually all the sink nodes need to extract all shares $(s_1, s_2, \dots, s_n)$ that are generated in the source. Each share is assigned to one of the source outputs and is sent to the second layer. The i^{th} share is passed to the i^{th} node, which is in the second layer and divided into Γ_O shares $(s_{i,1}, s_{i,2}, \dots, s_{i,\Gamma_O})$. This means that each share generated in the source is considered as a secret in the second layer and divided into some other shares. The threshold secret sharing scheme with parameters (Γ_O, t_{ss}) is used in the second layer where $t_{ss} \leqslant \Gamma_O$. Then, for extracting s_i, each sink needs to receive at least t_{ss} shares of all shares that are generated in the i^{th} node.

In total, the secret S is divided into $n\Gamma_O$ shares and multicast to all sinks by network coding. Each sink can reconstruct the secret S from nt_{ss} of shares. It should be considered that not all nt_{ss} sets of shares can reconstruct the secret S. Only if each sink receives at least t_{ss} shares from each node in the second layer, the secret S will be extractable. It means that even if a wiretapper can capture nt_{ss} distinct shares, this captured set of shares is useful only if it corresponds exactly to the n original first layer shares, so the security of the network is improved. We will show that the network coding can guarantee that each set of t_{ss} second layer shares can be conveyed node in the second layer to each sink.

The *max-flow*(R) between the source and each sink must be bigger than or equal to nt_{ss} or $R \geqslant nt_{ss}$. For security, it is required that $(nt_{ss} \geqslant \mu)$.

B. Subgraph Optimization

In the 2-LSNC approach, different sets of information streams are produced at the second layer nodes. Each of these streams must be sent to each sink nodes over disjoint paths. By routing, the problem is to find disjoint Steiner trees [13], which is known to be NP-hard; and the resulting solutions are comparatively wasteful with respect to network resources. For even a moderate number of sink nodes, it may be impossible to allocate such trees. The now obvious alternative approach is to use network coding. In this subsection, we explain how to optimize the subgraph for the network coding multicast.

Based on the work by Lun *et al.* [12], we have investigated a linear program (LP) to minimize the cost and to provide our objectives. The goal here is to search a subgraph that includes the required flow for each sink with minimum cost. A fixed cost and unit capacity is considered for each link.

The CTSS for our approach (2-LSNC) uses some of the outgoing links of the source node (n outputs) to apply the first step of secret sharing. Each node in the second layer constructs t_{ss} shares by using threshold secret sharing. A sink node must receive t_{ss} flows from each of the second layer nodes. If we virtually assign capacity t_{ss} to those n outgoing links from the source node, this problem is similar to send a flow of value nt_{ss}. Now we have a single source network coding and the optimization problem is a linear programming (LP) shown in (2)

$$\begin{aligned} &\text{minimize} \sum_{(i,j)\in E} b_{ij} z_{ij} \\ &\text{subject to:} z_{ij} \leqslant c_{ij} \quad \forall (i,j) \in E \\ &0 \leqslant x_{ij}^{(t)} \leqslant z_{ij} \quad \forall (i,j) \in E \ \forall t \in T \\ &\sum_{\{j|(i,j)\in E\}} x_{ij}^{(t)} - \sum_{\{j|(j,i)\in E\}} x_{ji}^{(t)} = \sigma_i^{(t)} \quad \forall i \in V \ \forall t \in T \end{aligned} \tag{2}$$

Here b_{ij}, z_{ij} and c_{ij} are cost, flow, and capacity of link (i, j) respectively. $x_{ij}^{(t)}$ is the variable which shows the amount of flow destined to sink t passing through link (i, j). After solving this LP problem it is possible to see which links are used in subgraph by looking at the z_{ij} values, and it is also possible to see which links are used for a specific sink by checking the value of $x_{ij}^{(t)}$ for all links. In the LP problem, we have:

$$\sigma_i^{(t)} = \begin{cases} R, & \text{if } \ i = s \\ -R, & \text{if } \ i = t \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

where R is equal to nt_{ss}.

The trick of virtually increasing the capacity of outgoing links from the source node enables the sink nodes to utilize their full receiving capacity, which in turn can improve the max-flow to the sinks. This improvement is also confirmed in simulations.

Fig. 1 shows an example of how our model improves security compare to the C&Y model in worst case $(h = 1)$. In this example, the same graph is used for both models. In this graph we have one source (S) and two sinks $(t1, t2)$. Links carrying coded packets are shown with dashed lines.

In Fig. 1(a), the C&Y model is applied and the result has a (2,2)-secret sharing. In Fig. 1(b) our approach utilizes the complete available resources. The resulted security is a (2,2,2)-secret sharing. In the C&Y model, the wiretapper requires to at least access two links in order to extract the secret. However, in our case wiretapper requires to access 4 links.

V. Performance of the 2-LSNC Model and Simulation Results

In this section, we describe some criteria for evaluating and comparing the secure schemes. The 2-LSNC model sends one secret at each time instance, while the C&Y model can send h symbols. Hence, the 2-LSNC model can be compared with the

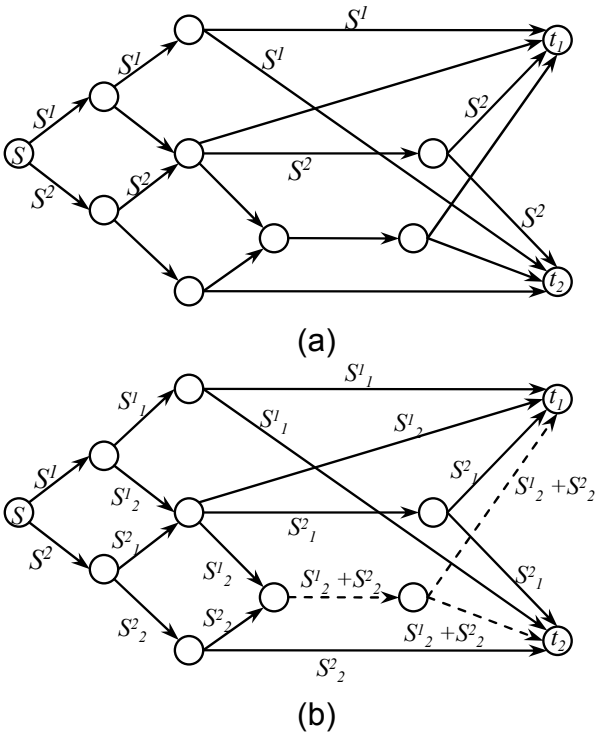

Fig. 1. Secure Network Coding Example

C&Y model when $h = 1$. In this section, parameters without and with an asterisk $(*)$ refer to the 2-LSNC model and to the C&Y model, respectively.

In our simulation, we generate a vast number of random geometric graphs with output degree for each 7 to 8 node. The number of nodes ranges from 40 to 100 nodes and for up to eight sinks. The C&Y and 2-LSNC models are applied to these graphs and the average results are computed and demonstrated in Fig. 2-6.

A. The Number of Links Accessible by Wiretapper

In this subsection, the maximum access to links by wiretapper (Maximum value for μ) that the model is still resistant against, is compared for two models. Wiretapper requires at least $\mu + 1$ links in order to retrieve the secret. For sake of simplicity, μ and μ^* are used for their maximum obtainable value.

In the C&Y model, μ^* is equal to $R^* - 1$, because $h^* = R^* - \mu^*$ and it is assumed that $h^* = 1$. In 2-LSNC model μ must be less than nt_{ss} where $nt_{ss} \leqslant R$, so μ is $nt_{ss} - 1 = R - 1$. According to subsection IV-B we have $R > R^*$, hence $\mu > \mu^*$. Thus the new model is more secure against wiretappers. The simulation results, described in Figure 2, confirm that μ is increased in our model.

B. The Level of Security

The *Level of Security*, denoted by Φ depends on μ and the number of input flows for the intermediate nodes, which is denoted by λ. The larger λ is, the fewer intermediate nodes are needed by the wiretapper to collect the interesting

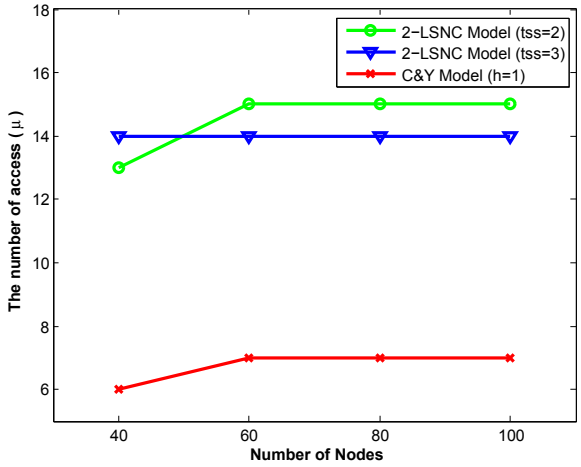

Fig. 2. Simulation results for μ

information. The parameter λ shows the level of security when the wiretapper is supposed to access intermediate nodes (node-wise security). For all nodes, λ is less than or equal to μ except for the sink nodes where $\lambda \geqslant (\mu + 1)$. Φ will grow by increasing μ and decreasing λ . Consequently, Φ is defined in the following:

$$\Phi = (1 - \frac{\lambda}{\mu}), 0 \leqslant \Phi \leqslant 1 \tag{4}$$

Fig. 3 and 4 show that the level of security in 2-LSNC model is better than in C&Y model. In Fig. 3 the level of security (Φ) is sketched for the fixed number of nodes when the number of sinks increases. Fig. 4 follows the same role when the number of sinks is considered fixed.

From Fig. 3 and 4, we can make two observations. First, the 2-LSNC model is always more secure than the C&Y model. Second, increasing the number of sinks or nodes results in a lower lever of security for the C&Y model, while in the 2-LSNC this is almost constant.

C. The cost of security

Since there is always a trade-off between level of security and the total cost (P), we provided a new metric (Ω) to compare the cost per level of security. According to (2), cost is the objective value from the optimization process. We have defined Ω in (5) and simulation results are shown in Fig. 5 and 6. Fig. 5 shows Ω according to the number of sinks for the fixed number of nodes and Fig. 6 shows Ω according to the number of nodes for the fixed number of sinks.

$$\Omega = \frac{P}{\Phi \cdot 100} \tag{5}$$

Based on Fig. 5 and 6 we observe that for a small number of nodes or sinks, the C&Y is more cost efficient (according to (5)) then the new model. However, the new model scales better and is more efficient in networks with a large number of nodes or sinks.

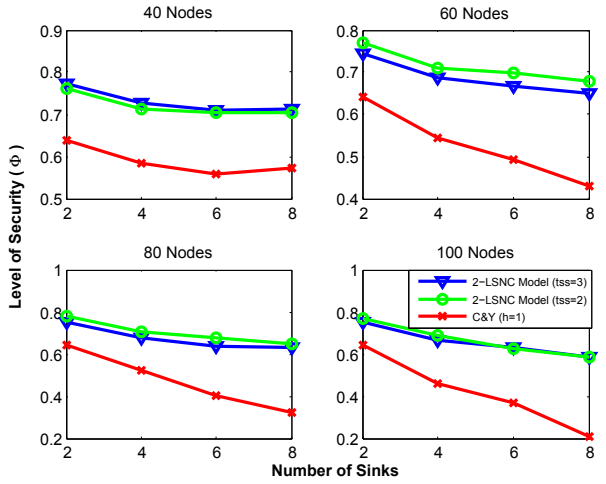

Fig. 3. The level of security (Φ) for fixed number of nodes

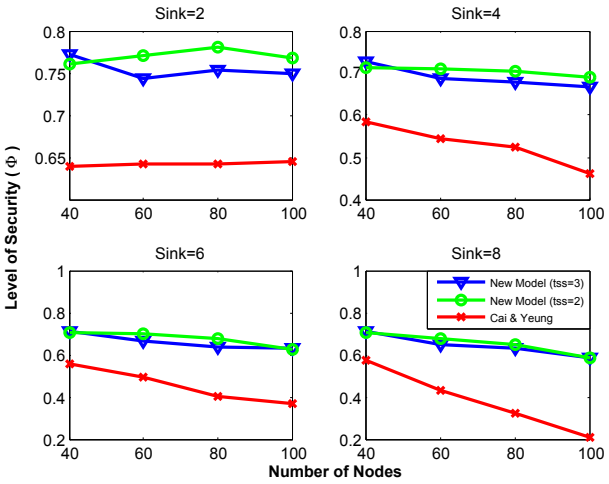

Fig. 4. The level of security (Φ) for fixed number of sinks

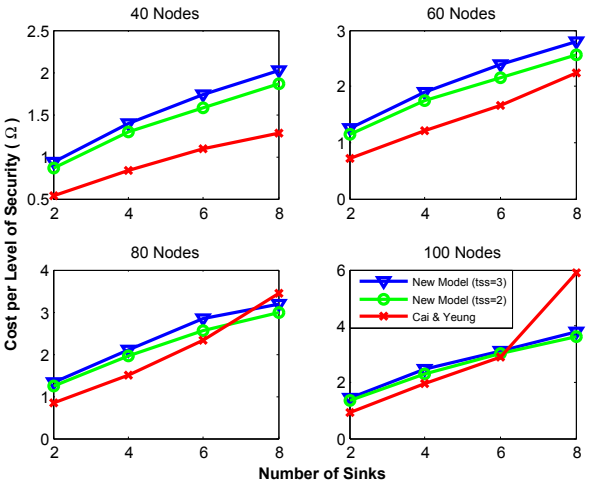

Fig. 5. The cost per security (Ω) for fixed number of nodes

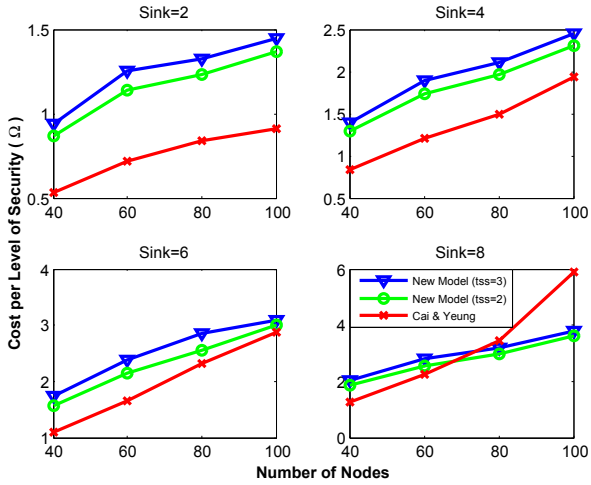

Fig. 6. The cost per security (Ω) for fixed number of sinks

VI. Conclusion

The new model, Two Layer Secure Network Coding (2-LSNC), improves several security parameters compared to previous approaches. These parameters include the maximum wiretapper resistance, the level of security and the cost-to-security-level ratio.

Our model is resistant against more powerful wire-tappings, and the cost for increasing the level of security becomes less than Cai and Yeung's model when the network size and the number of sinks reaches a critical point. Our simulation results confirmed these improvements.

References

[1] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network Information Flow", *IEEE Transactions on Information Theory,* Vol. 46, April 2000, pp. 1204-1216.

[2] S.-Y. R. Li, R. W. Yeung, and N. Cai, "Linear network coding", *IEEE Transactions on Information Theory,* Vol. 49, February 2003, pp. 371-381.

[3] N. Cai, and R. W. Yeung, "Secure network coding", *Proceedings of IEEE Symposium on Information Theory, 2002*

[4] J. Feldman, T. Malkin, R. A. Servedio, and C. Stein, "On the Capacity of Secure Network Coding", *Proc. 42nd Annual Allerton Conference on Comunication, Control and Computing,* September 2004.

[5] L. Lima, M. Medard, and J. Barros, "Random linear network coding: a free cipher?", *Proceedings of IEEE Symposium on Information Theory, 2007*

[6] T. Ho, M. Medard, J. Shi, M. Effros, and D.R. Karger, "On randomized network coding", *Proc. 41st Annual Allerton Conference on Comunication, Control and Computing,* Oct. 2003.

[7] F. Lu, L. Geng, L. Chia, and Y. Liang, "Secure Multi-path in Sensor Networks", *Proc. the 5th international conference on Embedded networked sensor systems,* 2007.

[8] S. Jaggi, P. Sanders, P.A. Chou, M. Effros, S. Egner, K. Jain and L.M.G.M. Tolhuizen,"Polynomial Time Algorithms for Multicast Network Code Construction", *IEEE Transactions on Information Theory,* Vol. 51, June 2005, pp. 1973-1982.

[9] Á. Barbero and Ø. Ytrehus, "Cycle-logical Treatment of 'Cyclopathic' Networks", *IEEE Transactions on Information Theory,* Vol. 52, June 2006, pp. 2795-2805.

[10] A. Shamir, "How to share a secret", *Communications of the ACM,* Vol. 22(1), 1979, pp. 612-613.

[11] G. R. Blakley, "Safeguarding cryptographic keys", *Proc. the National Computer Conference,* Vol. 48, 1979, PP. 313-317.

[12] D. S. Lun, N. Ratnakar, M. Médard, R. Koetter, D. R. Karger, T. Ho, E. Ahmed, and F. Zhao "Minimum-Cost Multicast over Coded Packet

Networks", *IEEE Transactions on Information Theory,* Vol. 52, Issue 6, June 2006, pp. 2608– 2623.
[13] P. Winter, "Steiner problem in networks: A survey,", *Networks*, vol. 17, Wiley-Interscience New York, 1987, pp. 129-167.