entropy

MDPI

Article

Secure Physical Layer Network Coding versus Secure Network Coding †

Masahito Hayashi [1,2,3,4]

[1] Shenzhen Institute for Quantum Science and Engineering, Southern University of Science and Technology, Nanshan District, Shenzhen 518055, China; hayashi@sustech.edu.cn
[2] International Quantum Academy (SIQA), Futian District, Shenzhen 518048, China
[3] Guangdong Provincial Key Laboratory of Quantum Science and Engineering, Southern University of Science and Technology, Nanshan District, Shenzhen 518055, China
[4] Graduate School of Mathematics, Nagoya University, Furocho, Chikusa-ku, Nagoya 464-8602, Japan

† Parts of this paper were presented at the 2018 IEEE Information Theory Workshop (ITW), Guangzhou, China, 25–29 November 2018.

Abstract: When a network has relay nodes, there is a risk that a part of the information is leaked to an untrusted relay. Secure network coding (secure NC) is known as a method to resolve this problem, which enables the secrecy of the message when the message is transmitted over a noiseless network and a part of the edges or a part of the intermediate (untrusted) nodes are eavesdropped. If the channels on the network are noisy, the error correction is applied to noisy channels before the application of secure NC on an upper layer. In contrast, secure physical layer network coding (secure PLNC) is a method to securely transmit a message by a combination of coding operation on nodes when the network is composed of set of noisy channels. Since secure NC is a protocol on an upper layer, secure PLNC can be considered as a cross-layer protocol. In this paper, we compare secure PLNC with a simple combination of secure NC and error correction over several typical network models studied in secure NC.

Keywords: secrecy analysis; secure communication; untrusted relay; network coding; physical layer security; cross-layer protocol

Citation: Hayashi, M. Secure Physical Layer Network Coding versus Secure Network Coding. *Entropy* **2022**, *24*, 47. https://doi.org/10.3390/e24010047

Academic Editors: Alex Dytso and Luca Barletta

Received: 24 November 2021
Accepted: 23 December 2021
Published: 27 December 2021

1. Introduction

Wireless communication networks with relay nodes have a risk for information leakage to untrusted relays. To resolve this problem, several studies [1–6] considered the relay terminals untrustworthy based on the result of secure computation-and-forward (CAF) tests [7–11], which is the main topic of the secure extension of physical layer network coding (PLNC), in short, secure PLNC. However, this type of security can be realized by the secure extension of network coding (NC), in short, secure NC, which is an upper-layered protocol to securely transmit a message via a noiseless network when a part of the edges and/or a part of the intermediate (untrusted) nodes are eavesdropped [12–17]. Since a wireless channel is disturbed by noise, an error correction needs to be applied to the channel. Then, secure NC is applied to noiseless channels virtually implemented by an error correction. In other words, the error correction and secure NC are separately performed in the different layers under the above scenario. In contrast, since secure PLNC combines both parts, it can be considered as a cross-layer protocol. In order to clarify the advantage of this cross-layer protocol, it is needed to compare secure PLNC with a simple combination of secure NC and error correction over wireless channels, and this comparison has not been studied yet. That is, this type of comparison is strongly required in the viewpoint of wireless communication networks.

Secure PLNC is based on PLNC [18–20], which efficiently transmits the modulo sum of two transmitters' messages via a Gaussian channel. To guarantee the security, the preceding studies [7–11,21] invented a secure extension of PLNC, i.e., secure PLNC, which is a scheme

to securely transmit a message by a combination of coding operations on nodes when the network is given as a set of noisy channels. Secure PLNC can be classified into two types. In the first case, secure NC is applied to the noiseless CAF process realized by PLNC. This method can be considered as a simple combination of secure NC and PLNC. The other type is a direct method to realize security in the PLNC. The typical example is secure CAF. The code of the latter type cannot be made by such a simple combination. All existing studies [7–11,21] belong to the latter case and address only a two-hop relay scheme or its simple extension, the multi-hop relay scheme, which are based on secure CAF to securely transmit the modulo sum of two input message when the channel is a noisy multiple access channel (MAC). Indeed, a secure NC can guarantee the secrecy for the eavesdropper that eavesdrops the channels. Several typical secure NCs cannot guarantee the secrecy when one of the intermediate (untrusted) nodes is eavesdropped. In this way, secure PLNC has an advantage under attacks on intermediate (untrusted) nodes.

However, the network models studied in secure NC are more advanced and more complicated, and no study discussed secure PLNC over such typical network models in secure NC. That is, the network models studied in secure PLNC is too limited and too primitive in the comparison with typical network models in secure NC. In other words, no prior study investigated the application of secure PLNC to such typical network models. In order for secure PLNC to overcome secure NC, we need to demonstrate that secure PLNC can be used in more advanced network models. At least, it is needed to study secure PLNC over typical network models in secure NC.

Since no existing paper has made the comparison between secure PLNC and the simple combination of secure NC and error correction, this paper aims to make this type of comparison under typical network models in secure NC. That is, this paper is the first study for secure PLNC over typical network models in secure NC, the butterfly network model and the network model composed of three source nodes under certain assumptions for the attack. Unfortunately, secure PLNC has a completely different mathematical structure from the simple combination of secure NC and error correction. Hence, it is quite difficult to construct a general theory to compare them. Due to this reason, we address two typical network models in the area of secure NC, the butterfly network [22] and a network with three source nodes, which is a special network model studied in [23].Then, we make the above comparison numerically over these two networks. Indeed, many existing studies [7–11] for secure PLNC employed lattice codes. Only the reference [21] studied it with BPSK modulation. Notice that the QPSK modulation can be considered as twice the use of the BPSK modulation.

For PLNC, references [24–27] discussed CAF based on lattice codes. Indeed, 2^n-phase shift keying (PSK) modulation works for practical systems such as conventional satellite communications with LDPC codes [28]. In addition, references [29,30] demonstrated the efficiency of the CAF scheme composed of binary LDPC codes under the BPSK modulation. Reference [31] compared the BPSK modulation and the method based on lattice codes for CAF. Hence, to adopt the existing communication system, we focus on the BPSK modulation.

Although, this paper is the journal version of the preceding conference paper [32], this paper is different from the conference version as follows. First, the conference version gave the secure NC protocol only when q is not a power of 2. This paper additionally gives the secure NC protocol when q is a power of 2 (not 2). This kind of extension enables us to consider the new protocol given in Section 3.3.2. Second, the conference version discussed only one type of secure NC protocol. This paper additionally considers another type of protocol in secure PLNC (See Sections 3.3.2 and 4.2.2). Totally, this paper discusses two types of protocols in secure PLNC. This additional protocol clarifies the merit of use of CAF. Third, the conference version compared the number of times slots only for two cases: secure NC without Gaussian MAC and secure PLNC with Gaussian MAC. Also, it did not considered the protocol in Sections 3.3.2 and 4.2.2. This paper additionally considers another case: secure NC with Gaussian MAC. Further, this version discussed the transmission time by considering the information transmission rate when the asymptotically best code is employed. To make this additional comparison, analytical

discussions are newly made in this version by using the mutual information. Also, this version newly contains numerical graphs (Figures 3 and 5) for this comparison. Due to this additional comparison, we can compare the transmission time.

The rest of this paper is organized as follows. First, Section 2 reviews the results in CAF and secure CAF, which is a typical example of secure PLNC. Next, Section 3 considers how secure communication can be implemented over the butterfly network based on secure PLNC. Finally, Section 4 discusses how secure communication can be implemented over a network with three source nodes based on secure PLNC. That is, Sections 3 and 4 are devoted to our contribution.

2. CAF and Secure CAF

2.1. CAF

As the first step, we review existing results for secure CAF. For this aim, we prepare an important notation. The symbol $\oplus$ expresses the arithmetic sum over a finite field, and the symbol $+$ denotes the sum over the real numbers. A typical setting for secure CAF has two transmitters, V_1 and V_2, and one receiver, R. Suppose that Transmitter V_i has message $M_i \in \mathbb{F}_q$, and Receiver R is linked by a (noisy) MAC that has two input variables from the two transmitters V_1 and V_2. In this scheme, Receiver R is required to obtain the modulo sum $M_1 \oplus M_2$ via the (noisy) MAC, as depicted in Figure 1.

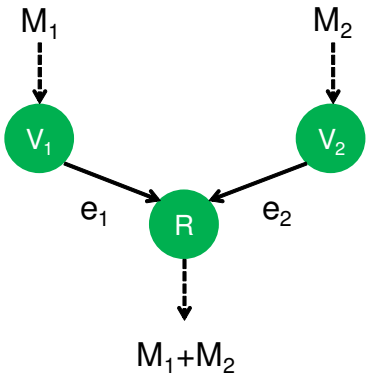

Figure 1. CAF (Computation-and-forward).

Many papers proposed a protocol for CAF over a Gaussian MAC. Suppose that the transmitter V_i sends the complex-valued variable X_i for $i = 1, 2$. When the channel fading coefficients are given as $h_1, h_2 \in \mathbb{C}$, Receiver R receives the complex-valued variable Y as:

$$Y = h_1 X_1 + h_2 X_2 + N, \tag{1}$$

where N is a complex Gaussian random variable with zero mean and a variance of one. The remaining part of this section assumes multiple uses of the above Gaussian MAC.

References [24,33,34] obtained an achievable rate under the energy constraint by using lattice codes. This rate is called the computation rate. Here, to seek a practical scheme, we consider the BPSK scheme, in which X_i is coded to $(-1)^{A_i}$ with $A_i \in \mathbb{F}_2$. Hence, (1) can be rewritten as:

$$Y = h_1(-1)^{A_1} + h_2(-1)^{A_2} + N. \tag{2}$$

The reference [35] showed that the rate $I(Y; A_1 \oplus A_2)_{\text{Equation(2)}}$ is achieved when the task of CAF is imposed, where the mutual information is given by the independent and uniform random numbers A_1 and A_2. (More precisely, the quantity $I(Y; A_1 \oplus A_2)_{\text{Equation(2)}}$ is defined as the mutual information when A_1 and A_2 are independently subject to the uniform distribution. This rule will be applied later when the equation number such as Equation (2) is given as a subscript of a mutual information). Then, references [29,30]

studied LDPC codes, in particular, spatial coupling LDPC codes and regular LDPC codes, to achieve this task under the BPSK scheme. In fact, the method introduced by references [29,30] can be efficiently implemented with a rate close to $I(Y; A_1 \oplus A_2)_{\text{Equation(2)}}$. Furthermore, the recent reference [36] studied its quantum extension.

2.2. Secure CAF

Next, we consider the secrecy condition for each message to Receiver R in addition to the correct decoding. This problem setting is called secure CAF. Here, Receiver R is required to obtain the modulo sum $M_1 \oplus M_2$ while the variable Y in Receiver R's hand is required to be independent of M_1 and M_2. References [7–11] proposed an approach using lattice codes. Using an efficiently implementable algebraic for CAF given in [29,30], the reference [21] proposed an efficiently implementable code for secure CAF. (Here, a code is called an algebraic code when the encoding map preserves algebraic operation. For example, Reed Solomon codes and LDPC codes are algebraic codes.) It also showed that the rate $2I(Y; A_1 \oplus A_2)_{\text{Equation(2)}} - I(Y; A_1, A_2)_{\text{Equation(2)}}$ is achievable in the BPSK scheme ([21], (29)), where the mutual information is given with the independent and uniform random numbers A_1 and A_2. That is, when the channel (2) is prepared and Receiver R colludes with no transmitter, secure CAF guarantees no information leakage of each message to Receiver R while Receiver R can recover the sum $M_1 \oplus M_2$. In the code in [21], $I(Y; A_1 \oplus A_2)_{\text{Equation(2)}}$ is the rate of CAF, and $I(Y; A_1, A_2)_{\text{Equation(2)}} - I(Y; A_1 \oplus A_2)_{\text{Equation(2)}}$ is the rate of sacrifice bits for the privacy amplification. Hence, the achievable rate of secure CAF is the difference between these two rates.

In fact, all the references [7–11,21] for secure CAF addressed only the case when the number of transmitters is two. Only reference [37] addresses secure CAF when the number of transmitters is larger than two. Unfortunately, these existing studies proposed no application for secure CAF except for a secure two-way relay channel with untrusted relays. The remaining part of this paper discusses its further application.

2.3. Concrete Expressions for Mutual Information

In this paper, we employ mutual information $I(Y; A_1, A_2)_{\text{Equation(2)}} - I(Y; A_1 \oplus A_2)_{\text{Equation(2)}}$ when $h_1 = h_2 = h$. Although their concrete descriptions were presented in ([21], Section IV-A), we give these concrete descriptions here. Assume that ϕ_a is the Gaussian distribution with average a and a variance of one. By using the differential entropy H, the mutual information $I(Y; A_1 \oplus A_2)_{\text{Equation(2)}}$ is calculated as:

$$H\left(\frac{\phi_0 + 2\phi_h + \phi_{2h}}{4}\right) - \frac{1}{2}H\left(\frac{\phi_0 + \phi_{2h}}{2}\right) - \frac{1}{2}H(\phi_h). \tag{3}$$

when $n \to \infty$, this value goes to $\log 2$.

In addition, the mutual information $I(Y; A_1, A_2)_{\text{Equation(2)}}$ is calculated as:

$$H\left(\frac{\phi_0 + 2\phi_h + \phi_{2h}}{4}\right) - H(\phi_h). \tag{4}$$

when $n \to \infty$, this value goes to $\frac{3}{2}\log 2$.

3. Butterfly Network

3.1. Conventional Protocol

A typical method for NC is the butterfly NC [22], which efficiently transmits information in the crossing way as explained in Figure 2. The goal of this problem setting is composed of the following two requirements: One is the reliable transmission of the message M_1 from V_1 to V_6, and the other is the reliable transmission of the message M_2 from V_2 to V_5. When each channel transmits only one element of $\mathbb{F}_q$, the bottleneck of this network is the channel e_3 from V_3 to V_4. Here, no signal is transmitted between disconnected nodes. Hence, no cross talk occurs between disconnected nodes. However, cross talk occurs between e_5 and e_6 if the signal on e_5 is different from that on e_6. Hence,

if they are different, the transmission on e_5 has to be performed on a different time from the transmission on e_6. However, when they are the same, these transmissions can be performed simultaneously. In this network model, only the node V_3 has a freedom to choose the transmitted information because other nodes receive only one information so that they have no other choice for the transmitted information except for transmitting the received information. To resolve the bottleneck in e_3, the node V_3 transmits the modulo sum to the node V_4 via channel e_3. Then, both destination nodes can recover their respective intended messages while the information transmission over e_3 is performed only once. That is, the destination node V_5 decodes the message M_2 from the received information M_1 and $M_1 \oplus M_2$. Similarly, the other destination node V_6 decodes the message M_1 from the received information M_2 and $M_1 \oplus M_2$.

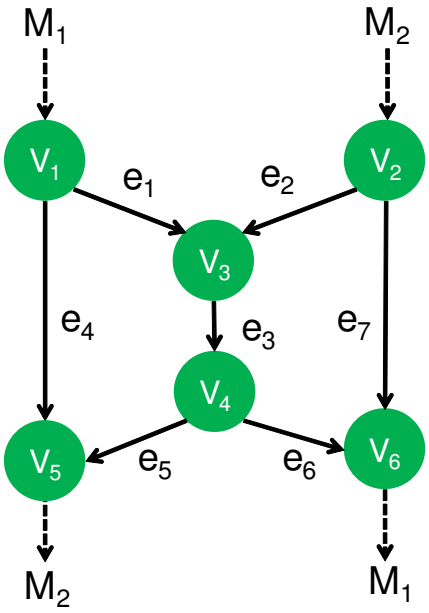

Figure 2. Butterfly NC.

3.2. Secure NC

Under the network code given in Section 3.1, the node V_3 obtains both messages M_1 and M_2. The destination node V_5 recovers the unintended message M_1 as well as the intended message M_2, and the other destination node V_6 the unintended message M_2 as well as the intended message M_1. Next, we impose the secrecy against an attack to one of the intermediate (untrusted) nodes. In other words, the information of all intermediate (untrusted) nodes are required to be independent of M_1 and M_2, and the information of destination node V_5 (V_6) is required to be independent of the unintended message M_1 (M_2). This kind of secrecy can be realized under the following assumption. When messages M_1 and M_2 are elements of $\mathbb{F}_q$ and q is not a power of 2 in the following way ([38], Figure 2):

(A1) Two source nodes V_1 and V_2 share a secret number L,

when the information Z_i transmitted on the edge e_i is given as:

$$Z_1 = 2M_1 \oplus L,\ Z_4 = -(M_1 \oplus L), \tag{5}$$
$$Z_2 = 2M_2 \oplus L,\ Z_7 = -(M_2 \oplus L), \tag{6}$$
$$Z_3 = Z_1 \oplus Z_2 = 2M_1 \oplus 2M_2 \oplus 2L, \tag{7}$$
$$Z_5 = Z_6 = Z_3/2, \tag{8}$$
$$\hat{M}_2 = Z_5 \oplus Z_4 = M_2,\ \hat{M}_1 = Z_6 \oplus Z_7 = M_1, \tag{9}$$

where $\hat{M}_2$ ($\hat{M}_1$) is the recovered message by V_5 (V_6). Any intermediate edge and any intermediate node obtain no information about the messages M_1 and M_2. In addition, the destination node V_5 (V_6) obtains no information for the message M_1 (M_2) while it obtains the message M_2 (M_1). Hence, this code guarantees the following types of security:

(S1) When the eavesdropper attacks only one of the edges, she obtains no information for each message M_i.

(S2) When the nodes do not collude, each node obtains no information for the unintended messages.

When $q \geq 4$ is a power of 2, the above code can be modified as follows. We choose an element $e \in \mathbb{F}_q$ such that $e^2 \oplus e \neq 0$, i.e., $e \neq 1, 0$.

Then, we define our code as:

$$Z_1 = (1 \oplus e)M_1 \oplus L,\ Z_4 = -(M_1 \oplus L), \tag{10}$$

$$Z_2 = (1 \oplus e)M_2 \oplus eL,\ Z_7 = -(M_2 \oplus L), \tag{11}$$

$$Z_3 = Z_1 \oplus Z_2 = (1 \oplus e)(M_1 \oplus M_2 \oplus L), \tag{12}$$

$$Z_5 = Z_6 = Z_3/(1 \oplus e), \tag{13}$$

$$\hat{M}_2 = Z_5 \oplus Z_4 = M_2,\ \hat{M}_1 = Z_6 \oplus Z_7 = M_1. \tag{14}$$

This modification realizes the required security in this case.

3.3. Secure PLNC

3.3.1. Use of Secure CAF

If no shared secret number is assumed between V_1 and V_2, it is difficult to realize the type of secrecy for the butterfly network presented in Section 3.2 under the problem setting of secure NC. Then, we consider the following assumption:

(A2) The pairs (e_1, e_2), (e_4, e_5), and (e_6, e_7) are given as Gaussian MACs such as (2).

In the network model given in Figure 2, only the channel e_3 is a Gaussian channel with a single input. To achieve secrecy under the assumption (A2), we employ secure CAF in the Gaussian MACs appearing in this network model in the following way: In the Gaussian MAC (e_1, e_2) at V_3, the node V_3 receives only the information $M_1 \oplus M_2$. Then, the node V_3 forwards the received information to the node V_4, and the node V_4 receives the information $M_4 := M_1 \oplus M_2$. In the Gaussian MAC (e_4, e_5) at V_5, the node V_5 receives only the information $M_4 \oplus (-M_1) = M_2$. In the same way, the node V_6 receives only the information $M_4 \oplus (-M_2) = M_1$. That is, we employ secure CAF in the three Gaussian MACs at V_3, V_5, and V_6. In this way, these uses of secure CAF realize the security (S2) under this method.

3.3.2. Use of CAF

As another kind of secure PLNC, we attach the CAF to the decoding operations on nodes V_3, V_5, and V_6 in the protocol with $q = 4$ given in Section 3.2. In this protocol, an element of $\mathbb{F}_4$ is regarded as a vector over the finite field $\mathbb{F}_2$. While this protocol saves the time, it still requires the secure shared randomness L. This protocol can be regarded as a simple combination of secure NC and PLNC.

The assumptions and the realized types of security are summarized in Table 1. Only the protocol given in Section 3.3.1 can realize security (S2) without requiring a secure shared randomness between two source nodes. This is a big advantage for secure PLNC.

Table 1. Comparison for protocols in butterfly network.

Protocol	Assumption	Security	Type
Section 3.2	(A1)	(S1) (S2)	Secure NC
Section 3.3.1	(A2)	(S2)	Secure PLNC with secure CAF
Section 3.3.2	(A1) (A2)	(S1) (S2)	Secure PLNC with CAF and secure NC

3.4. Comparison

To implement the above discussed protocols as wireless communication networks, we compare the transmission rates of the protocols given in Sections 3.2 and 3.3 when each edge is given as the BPSK scheme of a two-input Gaussian channel as (2) or a single-input Gaussian channel:

$$Y = hX + N, \tag{15}$$

where $h \in \mathbb{C}$ are the channel fading coefficients, N is a complex Gaussian random variable with zero mean and a variance of one, and X is coded as $(-1)^A$ with $A \in \mathbb{F}_2$. Hence, (16) is rewritten as:

$$Y = h(-1)^A + N, \tag{16}$$

In this comparison, for simplicity, we assume that $h_1 = h_2 = h$. We assume that T is the time period of transmitting one Gaussian signal on each edge. Additionally, we assume that ideal codes are available as follows. The mutual information rate $I(Y;A)_{\text{Equation(16)}}$ is achievable over the channel (16), the rate $I(Y;A_1 \oplus A_2)_{\text{Equation(2)}}$ is achievable for CAF in the channel (2), and the rate $2I(Y;A_1 \oplus A_2)_{\text{Equation(2)}} - I(Y;A_1,A_2)_{\text{Equation(2)}}$ is available for secure CAF in the channel (2). Notice that the relation $I(Y;A_2|A_1)_{\text{Equation(2)}} = I(Y;A_1|A_2)_{\text{Equation(2)}}$ holds in this case. In addition, the mutual information rate pair $(I(Y;A_1A_2)_{\text{Equation(2)}}/2, I(Y;A_1A_2)_{\text{Equation(2)}}/2)$ is available in the MAC channel (2) when both transmitters intend to send their own message to the receiver. (Generally, the symmetric rate $(I(Y;A_1,A_2)/2, I(Y;A_1,A_2)/2)$ is achievable when the symmetric rate $(I(Y;A_1,A_2)/2, I(Y;A_1,A_2)/2)$ belongs to the interval between $(I(Y;A_1), I(Y;A_2|A_2))$ and $(I(Y;A_1|A_2), I(Y;A_2))$. Our case with $h_1 = h_2 = h$ satisfies this condition.) In the above discussion, the random variables A_1, A_2, and A are subject to the uniform distribution independently.

The secure NC protocol given in Section 3.2 needs to avoid a crossed line when Gaussian MAC is not used. Now, we consider how much time is needed for this protocol. In this protocol, we need to repeat several processes, each of which is composed of the encoding, wireless communication, and decoding. In the protocol given in Section 3.2, the first step can make the simultaneous transmissions on e_1 and e_4. However, the simultaneous transmission on e_1 cannot be performed simultaneously in order to avoid the cross line on the receiving on V_3. Hence, the second step makes the simultaneous transmissions on e_2 and e_7. We say that the time period for the first step is the time slot of Time i, and the time period for the second step is the time slot of Time ii. That is, each time span for the process composed of the encoding, wireless communication, and decoding is called a time slot. Now, to evaluate the required number of time slots, we assume that all players have only one transmitting antenna, which can broadcast the transmitting signal. Then, we find that the whole network has five time slots at least as presented in Table 2. When the length of the transmitted message is G, the transfer time for each time slot is $\frac{GT}{I(Y;A)_{\text{Equation(16)}}}$. Therefore, the total transfer time in this case is calculated to be $\frac{5GT}{I(Y;A)_{\text{Equation(16)}}}$.

Table 2. Secure NC without Gaussian MAC.

Time Slot	Time i	Time ii	Time iii	Time iv	Time v
Channel	e_1, e_4	e_2, e_7	e_3	e_5	e_6

When we use the Gaussian MAC, the secure NC protocol given in Section 3.2 can be implemented with three time slots as Table 3 because V_4 broadcasts the information to e_5 and e_6. When the length of the transmitted message is G, the first time slot requires transfer time $\frac{2GT}{I(Y;A_1,A_2)_{\text{Equation(2)}}}$, and the second and third time slots require transfer time $\frac{GT}{I(Y;A)_{\text{Equation(16)}}}$. Hence, the total transfer time is calculated to be $\frac{2GT}{I(Y;A)_{\text{Equation(16)}}} + \frac{2GT}{I(Y;A_1,A_2)_{\text{Equation(2)}}}$. When we design the whole process as in Table 4, the first and third time slots require transfer time $\frac{2GT}{I(Y;A_1,A_2)_{\text{Equation(2)}}}$, and the second time slot requires transfer time $\frac{GT}{I(Y;A)_{\text{Equation(16)}}}$. Hence, the total transfer time is $\frac{GT}{I(Y;A)_{\text{Equation(16)}}} + \frac{4GT}{I(Y;A_1,A_2)_{\text{Equation(2)}}}$, which is larger than $\frac{2GT}{I(Y;A)_{\text{Equation(16)}}} + \frac{2GT}{I(Y;A_1,A_2)_{\text{Equation(2)}}}$ because $\frac{I(Y;A_1,A_2)_{\text{Equation(2)}}}{2} \le I(Y;A)_{\text{Equation(16)}}$.

Table 3. Secure NC with Gaussian MAC.

Time Slot	Time i	Time ii	Time iii
Channel	(e_1, e_2)	e_3, e_4, e_7	e_5, e_6

(e_i, e_j) expresses a Gaussian MAC composed of the joint transmission on the edges e_i and e_j.

The secure PLNC protocol given in Section 3.3.1 can be performed only with three time slots as in Table 4, where the pairs (e_1, e_2), (e_4, e_5), and (e_6, e_7) are realized by secure CAF based on the Gaussian MAC channel (2). The first and third time slots require transfer time $\frac{GT}{2I(Y;A_1\oplus A_2)_{\text{Equation(2)}} - I(Y;A_1,A_2)_{\text{Equation(2)}}}$, and the second time slot requires transfer time $\frac{GT}{I(Y;A)_{\text{Equation(16)}}}$. The total transfer time is calculated to be $\frac{2GT}{2I(Y;A_1\oplus A_2)_{\text{Equation(2)}} - I(Y;A_1,A_2)_{\text{Equation(2)}}} + \frac{GT}{I(Y;A)_{\text{Equation(16)}}}$.

Secure PLNC protocol given in Section 3.3.2 can also be implemented only with three time slots as in Table 4. The first and third time slots require transfer time $\frac{GT}{I(Y;A_1\oplus A_2)_{\text{Equation(2)}}}$, and the second time slot requires transfer time $\frac{GT}{I(Y;A)_{\text{Equation(16)}}}$. The total transfer time is calculated to be $\frac{2GT}{I(Y;A_1\oplus A_2)_{\text{Equation(2)}}} + \frac{GT}{I(Y;A)_{\text{Equation(16)}}}$.

Table 4. Secure PLNC with Gaussian MAC.

Time Slot	Time i	Time ii	Time iii
Channel	(e_1, e_2)	e_3	(e_4, e_5),(e_6, e_7)

Figure 3 gives the numerical comparison among the following time periods:

$$\begin{gathered} \frac{2GT}{2I(Y;A_1\oplus A_2)_{\text{Equation(2)}} - I(Y;A_1,A_2)_{\text{Equation(2)}}} + \frac{GT}{I(Y;A)_{\text{Equation(16)}}}, \\ \frac{4GT}{I(Y;A)_{\text{Equation(16)}}}, \quad \frac{2GT}{I(Y;A)_{\text{Equation(16)}}} + \frac{2GT}{I(Y;A_1,A_2)_{\text{Equation(2)}}}, \\ \frac{2GT}{I(Y;A_1\oplus A_2)_{\text{Equation(2)}}} + \frac{GT}{I(Y;A)_{\text{Equation(16)}}}. \end{gathered} \tag{17}$$

When $h \to \infty$, these values converge to:

$$\frac{5GT}{\log 2}, \frac{4GT}{\log 2}, \frac{10GT}{3\log 2}, \frac{3GT}{\log 2}, \tag{18}$$

respectively.

The secure NC protocol given in Section 3.2 requires a shorter transfer time for the transmission than the secure PLNC protocol given in Section 3.3 in this comparison. Since the difference is not so extensive, the secure PLNC protocol given in Section 3.3.1 is useful when it is not easy to prepare secure shared randomness between two source nodes. In fact, when the direct communication between two distinct source nodes is not available, we often use the butterfly network. In this case, such a secure shared randomness requires an additional cost. However, the secure PLNC protocol given in Section 3.3.2 has no advantage over the secure NC protocol with the MAC channel. That is, a simple combination of secure NC and PLNC is not useful in this case.

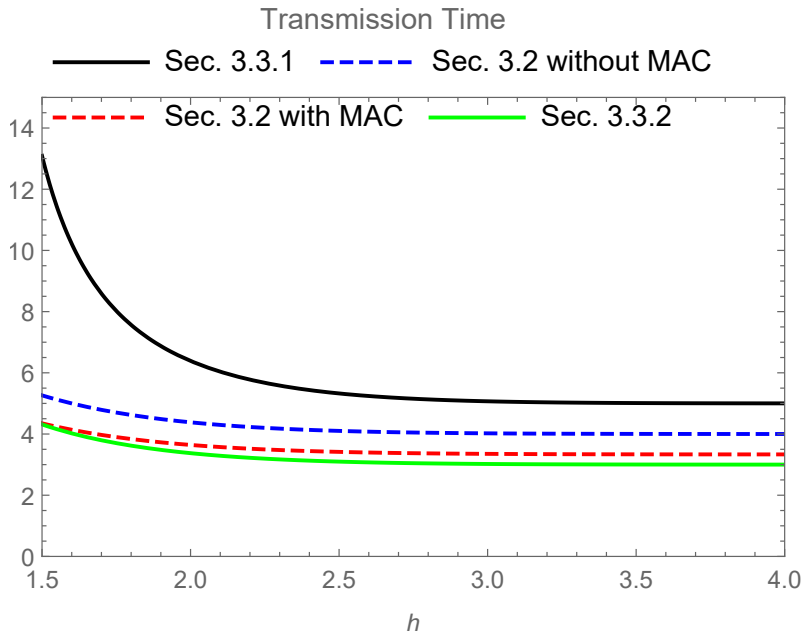

Figure 3. Transmission time for four schemes when $GT = 1$ and the base of the logarithm is 2. The upper solid line (black) expresses the time $\frac{2GT}{2I(Y;A_1\oplus A_2)_{\text{Equation(2)}}-I(Y;A_1,A_2)_{\text{Equation(2)}}} + \frac{GT}{I(Y;A)_{\text{Equation(16)}}}$ of the secure PLNC protocol given in Section 3.3.1. The upper dashed line (blue) expresses the time $\frac{4GT}{I(Y;A)_{\text{Equation(16)}}}$ of the secure NC protocol given in Section 3.2 without the MAC channel. The lower dashed line (red) expresses the time $\frac{2GT}{I(Y;A)_{\text{Equation(16)}}} + \frac{2GT}{I(Y;A_1,A_2)_{\text{Equation(2)}}}$ of the secure NC protocol given in Section 3.2 with the MAC channel. The lower solid line (green) expresses the time $\frac{2GT}{I(Y;A_1\oplus A_2)_{\text{Equation(2)}}} + \frac{GT}{I(Y;A)_{\text{Equation(16)}}}$ of the secure PLNC protocol given in Section 3.3.2.

4. Network with Three Source Nodes

Finally, we study the network topology shown in Figure 4 that is composed of three source nodes, S_1, S_2, and S_3; three intermediate nodes, I_1, I_2, and I_3; and one destination node, D. Its generalization was discussed as a multilayer network in the recent reference [23]. The goal of this network model is the secure transmission from the three source nodes to the destination node D when the source node S_i is required to send an element $M_i \in \mathbb{F}_q$ to the destination node D.

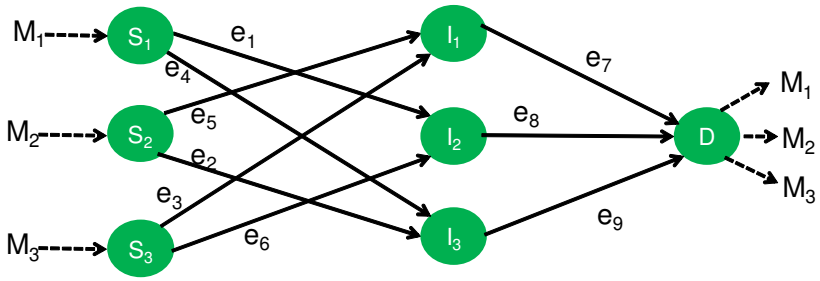

Figure 4. Network with three sources.

4.1. Secure NC

As the first step, let us study the network with three sources under the framework of the secure NC. In Figure 4, every edge expresses a noiseless channel to transmit one element of $\mathbb{F}_q$. Here, we consider the following two security requirements:

(S3) When Eve eavesdrops only one edge among three edges (channels) between the intermediate nodes and the destination node, she obtains no information about each message.

(S4) When Eve eavesdrops only one intermediate (untrusted) node among three intermediate (untrusted) nodes, she obtains no information for each message. Here, no node colludes with another node.

4.1.1. Security (S3)

The following code satisfies Security (S3) when q is not a power of 2. This code uses $1/2$, which cannot be allowed in finite field $\mathbb{F}_q$ of a power q of 2. Notice that the matrix $\begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}$ is invertible because $\begin{pmatrix} -1/2 & 1/2 & 1/2 \\ 1/2 & -1/2 & 1/2 \\ 1/2 & 1/2 & -1/2 \end{pmatrix}$ is the inverse matrix. Source node S_i sends M_i in each edge. Each intermediate node sends the sum of the received vector. Finally, applying the inverse matrix $\begin{pmatrix} -1/2 & 1/2 & 1/2 \\ 1/2 & -1/2 & 1/2 \\ 1/2 & 1/2 & -1/2 \end{pmatrix}$ to the received vector, the node D recovers all messages. In this code, each of the messages $M_1 \oplus M_2$, $M_2 \oplus M_3$, and $M_3 \oplus M_1$ are independent of anyone of M_1, M_2, and M_3. Hence, Security (S3) is satisfied. This protocol achieves the optimum transmission rate even when the secrecy condition is not imposed.

As the next step, let us proceed to the case when $q \geq 4$ is a power of 2. We choose an element $e \in \mathbb{F}_q$ such that $e^2 \oplus e \neq 0$, which implies that $\begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & e \\ e & e & 0 \end{pmatrix}$ is invertible because its determinant is $e^2 \oplus e \neq 0$. For example, when $q = 4$, since $e^2 = e \oplus 1$, the inverse matrix is: $\begin{pmatrix} e \oplus 1 & e & e \\ e \oplus 1 & e & 1 \\ e & e & 1 \end{pmatrix}$. Then, the following code is secure. Source node S_i sends M_i in each edge. The intermediate nodes I_1, I_2, and I_3 send the received information $Z_1 := M_2 \oplus M_3$, $Z_2 := M_1 \oplus eM_3$, and $Z_3 := eM_1 \oplus eM_2$, respectively. Finally, applying the inverse matrix of $\begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & e \\ e & e & 0 \end{pmatrix}$ to the received vector $\begin{pmatrix} Z_1 \\ Z_2 \\ Z_3 \end{pmatrix}$, the node D recovers all messages. In this code, each of the information symbols $eM_1 \oplus eM_2$, $M_2 \oplus M_3$, and $eM_3 \oplus M_1$ are independent of anyone of M_1, M_2, and M_3. Hence, Security (S3) is satisfied.

4.1.2. Security (S4)

To make a code satisfy Security (S4), we modify the above protocol as follows. The modified protocol uses the channels between the intermediate (untrusted) nodes and the

destination node twice. In addition, it employs the channels between the source nodes and the intermediate (untrusted) nodes only once. Source node S_i sends the scrambled variable L_i to the intermediate (untrusted) node $I_{i\oplus 1}$ via the edge e_i. Each source node S_i prepares the scrambled variable L_i and sends the variable $M_i \oplus (-L_i)$ to the intermediate (untrusted) node $I_{i\oplus(-1)}$ via the edge $e_{3\oplus i}$. Here, $i \oplus 1$ and $i \oplus (-1)$ are regarded as elements of $\mathbb{Z}_3$. Each intermediate (untrusted) node sends both received variables to the destination node by using the channel twice. Then, the destination node D can recover the messages as:

$$M_1 = (M_1 \oplus (-L_1)) \oplus L_1,\ M_2 = (M_2 \oplus (-L_2)) \oplus L_2, \tag{19}$$

$$M_3 = (M_3 \oplus (-L_3)) \oplus L_3 \tag{20}$$

because the node D obtains information L_1, L_2, L_3, $M_1 \oplus (-L_1)$, $M_2 \oplus (-L_2)$, and $M_3 \oplus (-L_3)$. The information at the intermediate (untrusted) node I_i is the pair of L_{i+1} and $M_{i-1} \oplus (-L_{i-1})$, which is independent of anyone of M_1, M_2, and M_3. Hence, this code guarantees Security (S4) as well as Security (S3).

4.2. Secure PLNC

4.2.1. Use of Secure CAF

Now, we assume the following assumption:

(A3) The channels over the pairs (e_1, e_6), (e_2, e_4), and (e_3, e_5) are Gaussian MACs as in (2).

That is, the eavesdropper is supposed to access only one of the information symbols at the intermediate (untrusted) nodes, which corresponds to Case 2 of Section 4.1. Then, using secure CAF [21], we construct our protocol.

As the first step, we discuss the case when q is not a power of 2. In the Gaussian MAC (e_1, e_6), we employ secure CAF so that the node I_2 obtains the information symbol $M_1 \oplus M_3$. Similarly, I_1 and I_3 obtain the information symbol $M_2 \oplus M_3$ and $M_1 \oplus M_2$, respectively. Hence, the information symbols at every intermediate (untrusted) node are independent of the messages M_1, M_2, and M_3. In the next step, the intermediates (untrusted) nodes I_1, I_2, and I_3 transmit their received information symbols M'_1, M'_2, and M'_3 to the destination node D via the Gaussian MACs with three input signals. Then, applying separate decoding, the destination node D recovers the information symbols M'_1, M'_2, and M'_3. Using the method presented in Section 4.1.1, the destination node D obtains the original information symbols M_1, M_2, and M_3.

When $q \geq 4$ is a power of 2, to apply the method given in Section 4.1.1, the node I_2 needs to obtain the information $M_1 \oplus eM_3$. This task for I_2 can be implemented by a secure CAF with a two-dimensional vector over the finite field $\mathbb{F}_2$ by the prior conversion from M_3 to eM_3 at the node S_3 before use of the Gaussian MAC (e_1, e_6). The same method is applied to the Gaussian MACs (e_2, e_4) and (e_3, e_5). The remaining part of this protocol can be performed in the same way as the above.

In the above way, the framework of the secure PLNC enables us to implement the secure code for an attack on an intermediate (untrusted) node by using secure CAF. That is, this code guarantees Security (S4). This protocol requires no additional random variable, unlike the protocol presented in Section 4.1.2.

4.2.2. Use of CAF

Next, we construct a protocol using CAF. In this protocol, at the node D, to recover M_1, we employ CAF on the two edges e_8 and e_9. Similarly, to recover M_2 (M_3), we employ CAF on the two edges e_7 and e_9 (e_7 and e_8). To avoid information leakage over every intermediate (untrusted) node, the transmitter applies the secure network code given in Section 4.1.2.

4.3. Comparison

All the proposed protocols are summarized in Table 5. Since the security of our interest is (S4), we compare the protocols except for the protocol given in Section 4.1.1. Only the

protocol given in Section 4.2.1 satisfies Security (S4). To implement these protocols as wireless communication network, we compare the transmission rates of the protocols given in Sections 4.1 and 4.2 when each edge is given as the BPSK scheme of a single-input Gaussian channel (16), a two-input Gaussian channel (2), or a three-input Gaussian channel (2):

$$Y = hX_1 + hX_2 + hX_3 + N, \tag{21}$$

where $h \in \mathbb{C}$ are the channel fading coefficients, N is a complex Gaussian random variable with zero mean and a variance of one, and X_i is coded as $(-1)^{A_i}$ with $A_i \in \mathbb{F}_2$. In this comparison, we make the same assumptions for h_1, h_2, and GT as the previous section. Additionally, we assume that ideal codes given in Section 3.4 are available, and that the mutual information rate triple

$$\left(\frac{I(Y;A_1A_2A_3)_{\text{Equation(21)}}}{3}, \frac{I(Y;A_1A_2A_3)_{\text{Equation(21)}}}{3}, \frac{I(Y;A_1A_2A_3)_{\text{Equation(21)}}}{3}\right)$$

is available in the MAC channel (21) when three transmitters intend to send their own message to the receiver, where the random variables A_1, A_2, and A_3 are independently subject to the uniform distribution [37]. Using this rate, we compare the secure NC protocol given in Section 4.1.2 and the secure PLNC protocol given in Section 4.2 because both protocols realize the secrecy for intermediate (untrusted) nodes.

When any Gaussian MAC is not used, the secure NC protocol given in Section 4.1.2 requires five time slots at least as shown in Table 6. In particular, the edges e_7, e_8, and e_9 need to send the information symbols twice as the remaining edges. Therefore, when the length of the transmitted message is G, the first and second time slots need transfer time $\frac{GT}{I(Y;A)_{\text{Equation(16)}}}$, and the remaining time slots need transfer time $\frac{2GT}{I(Y;A)_{\text{Equation(16)}}}$. Hence, the total transfer time is calculated to be $\frac{8GT}{I(Y;A)_{\text{Equation(16)}}}$.

When we use the Gaussian MAC, the secure NC protocol given in Section 4.1.2 can be implemented with two time slots as in Table 7. The first time slot needs transfer time $\frac{2GT}{I(Y;A_1,A_2)_{\text{Equation(2)}}}$, and the second time slot needs transfer time $\frac{6GT}{I(Y;A_1A_2A_3)_{\text{Equation(21)}}}$. Hence, the total transfer time is calculated to be $\frac{6GT}{I(Y;A_1A_2A_3)_{\text{Equation(21)}}} + \frac{2GT}{I(Y;A_1,A_2)_{\text{Equation(2)}}}$.

Table 5. Comparison for protocols in network given in Sections 4.1 and 4.2.

Protocol	Assumption	Security	Type
Section 4.1.1	–	(S3)	Secure NC
Section 4.1.2	–	(S3) (S4)	Secure NC
Section 4.2.1	(A3)	(S4)	Secure PLNC with secure CAF
Section 4.2.2	(A3)	(S3) (S4)	Secure PLNC with CAF and secure NC

Table 6. Secure NC without Gaussian MAC.

Time Span	Time i	Time ii	Time iii	Time iv	Time v
Channel	e_1, e_2, e_3	e_4, e_5, e_6	e_7	e_8	e_9

Table 7. Secure NC with Gaussian MAC.

Time Span	Time i	Time ii
Channel	(e_1, e_6), (e_2, e_4),(e_3, e_5)	(e_7, e_8, e_9)

The secure PLNC protocol given in Section 4.2.1 can be implemented only with two time slots as in Table 8, where the pairs (e_1, e_2), (e_4, e_5), and (e_6, e_7) are realized by the secure CAF based on the Gaussian MAC channel (2). The first time slot needs transfer

time $\frac{GT}{2I(Y;A_1\oplus A_2)_{\text{Equation(2)}}-I(Y;A_1,A_2)_{\text{Equation(2)}}}$, and the second time slot needs transfer time $\frac{3GT}{I(Y;A_1A_2A_3)_{\text{Equation(21)}}}$. Hence, the total transfer time is calculated to be $\frac{3GT}{I(Y;A_1A_2A_3)_{\text{Equation(21)}}}+\frac{GT}{2I(Y;A_1\oplus A_2)_{\text{Equation(2)}}-I(Y;A_1,A_2)_{\text{Equation(2)}}}$.

Table 8. Secure PLNC with secure CAF.

Time Span	Time i	Time ii
Channel	$(e_1,e_6),(e_2,e_4),(e_3,e_5)$	(e_7,e_8,e_9)

Another secure PLNC protocol given in Section 4.2.2 can be implemented only with two time slots as in Table 9, where the pairs (e_1,e_2), (e_4,e_5), and (e_6,e_7) are realized by the secure CAF based on the Gaussian MAC channel (2). The first time slot needs transfer time $\frac{2GT}{I(Y;A_1,A_2)_{\text{Equation(2)}}}$, and the second time slot needs transfer time $\frac{3GT}{I(Y;A_1\oplus A_2)_{\text{Equation(2)}}}$. Hence, the total transfer time is calculated to be $\frac{3GT}{I(Y;A_1\oplus A_2)_{\text{Equation(2)}}}+\frac{2GT}{I(Y;A_1,A_2)_{\text{Equation(2)}}}$.

Table 9. Secure PLNC with CAF.

Time Span	Time i	Time ii	Time iii	Time iv
Channel	$(e_1,e_6),$ $(e_2,e_4),(e_3,e_5)$	(e_8,e_9)	(e_7,e_9)	(e_7,e_8)

Figure 5 gives the numerical comparison among the following time periods:

$$\begin{aligned}&\frac{8GT}{I(Y;A)_{\text{Equation(16)}}},\ \frac{6GT}{I(Y;A_1A_2A_3)_{\text{Equation(21)}}}+\frac{2GT}{I(Y;A_1,A_2)_{\text{Equation(2)}}},\\&\frac{3GT}{I(Y;A_1\oplus A_2)_{\text{Equation(2)}}}+\frac{2GT}{I(Y;A_1,A_2)_{\text{Equation(2)}}}.\\&\frac{3GT}{I(Y;A_1A_2A_3)_{\text{Equation(21)}}}+\frac{GT}{2I(Y;A_1\oplus A_2)_{\text{Equation(2)}}-I(Y;A_1,A_2)_{\text{Equation(2)}}}.\end{aligned}\tag{22}$$

When $h\to\infty$, these values converge to:

$$\frac{8GT}{\log 2},\ \frac{8GT}{4\log 2-\log 3}+\frac{4GT}{3\log 2},\ \frac{13GT}{3\log 2},\ \frac{4GT}{4\log 2-\log 3}+\frac{2GT}{\log 2},\tag{23}$$

respectively.

The codes for the secure PLNC protocol given in Section 4.2.1 require shorter transfer time for the transmission than the secure NC protocol given in Section 4.1.2 in this comparison when the coefficient h is larger than about 1.7. This comparison shows that the secure PLNC protocol given in Section 4.2.1 has an advantage over the secure NC protocol given in Section 4.1.2 when the power of the signal is sufficiently large. In addition, this comparison indicates the advantage of the simple combination of secure NC and PLNC given in Section 4.2.2 over the secure NC protocol given in Section 4.1.2 with the MAC channel. In other words, if the power of the signal is not so large, the secure NC protocol given in Section 4.1.2 with the MAC channel is better than the secure PLNC protocols given in Sections 4.2.1 and 4.2.2.

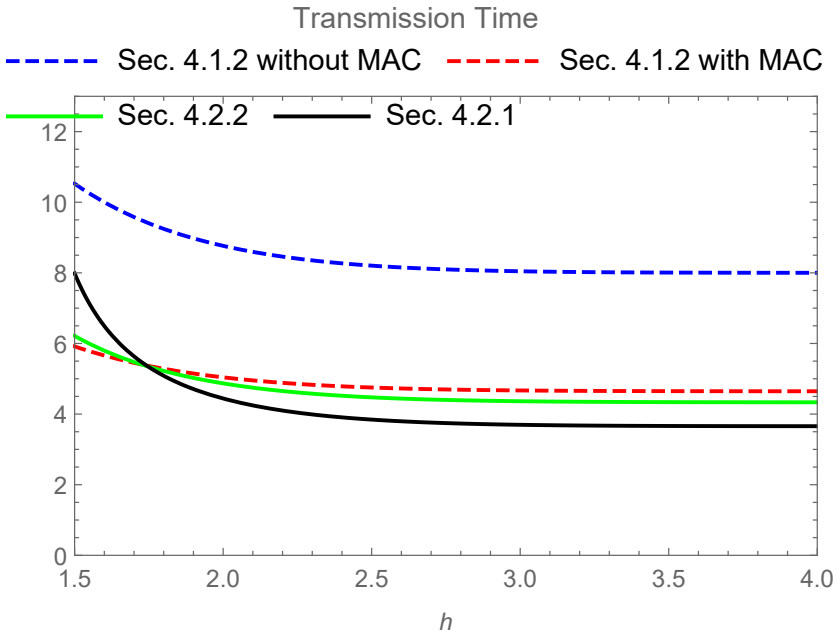

Figure 5. Transmission Time for four schemes when $GT = 1$ and the base of the logarithm is 2. The upper dashed line (blue) expresses the time $\frac{8GT}{I(Y;A)_{\text{Equation(16)}}}$ of secure NC protocol given in Section 4.1.2 without the MAC channel. The lower dashed line (red) expresses the time $\frac{6GT}{I(Y;A_1A_2A_3)_{\text{Equation(21)}}} + \frac{2GT}{I(Y;A_1,A_2)_{\text{Equation(2)}}}$ of the secure NC protocol given in Section 4.1.2 with the MAC channel. The solid line (green) expresses the time $\frac{3GT}{I(Y;A_1\oplus A_2)_{\text{Equation(2)}}} + \frac{2GT}{I(Y;A_1,A_2)_{\text{Equation(2)}}}$ of the secure PLNC protocol given in Section 4.2.2. The solid line (black) expresses the time $\frac{3GT}{I(Y;A_1A_2A_3)_{\text{Equation(21)}}} + \frac{GT}{2I(Y;A_1\oplus A_2)_{\text{Equation(2)}} - I(Y;A_1,A_2)_{\text{Equation(2)}}}$ of the secure PLNC protocol given in Section 4.2.1. The solid line (black), the solid line (green), and the lower dashed line (red) intersect around $h = 1.7$.

5. Conclusions and Discussion

We have studied the advantages of a secure PLNC over a secure NC. To investigate this type of advantage, we have focused on two typical network models. Section 3 has discussed the butterfly network model given in Figure 2, and Section 4 has discussed the network model with three source nodes given in Figure 4. We have described concrete protocols that efficiently realize the required secrecy and work over these network models. In these examples, the secure PLNC can realize the secrecy even with untrusted intermediate nodes. In particular, as summarized in Table 1, in the butterfly network, although the protocols using secure network codes require a secure shared randomness for this purpose, the secure PLNC does not need it. Comparing the transfer times of the proposed codes, we have shown that the secure PLNC has a shorter transfer time than the the simple combination of secure NC and physical layer network under a certain range of channel parameters.

As one of the main reasons of these advantages, we can list the fact that secure PLNC is a cross-layered network protocol. That is, it can be realized by a joint application of the error correction and the secure NC by using the mechanism of a physical layer while the conventional scenario can be considered as separate application of the error correction and the secure NC. In particular, the noise in the channels is utilized for keeping the secrecy in the secure PLNC. Therefore, we can conclude that the secure PLNC is useful to realize the secrecy against information leakage at intermediate (untrusted) nodes.

One might consider that the proposed method does not work for jamming attacks [39] or spoofing [40,41]. The transmitters and the receivers can detect it by attaching authentication [42–45], which can be realized by using a universal2 hash function and preshared keys.

Furthermore, the number of existing applications of the secure PLNC is quite limited. It is an important future study to find much more fruitful applications of the secure PLNC over untrusted relays. In fact, reference [21] also derived an upper bound for the amount of

the leaked information of the constructed finite-length code. Therefore, it is an interesting future topic to make finite-length analysis by applying the finite-length analysis in [21]. In addition, the analysis of this paper is based on the BPSK scheme. Since many papers on secure PLNC were based on lattice codes, a similar comparison based on lattice codes is needed. Such a comparison remains an interesting open problem.

Finally, we list three future problems. The first one is the application of the proposed method to multi-hop untrusted relaying networks [8,46]. The second one is the realization of covert communication [47,48] over the wireless networks discussed in this paper. The third one is the problem related to retransmission. In real communication, there is a possibility that we need to perform retransmission due to various reasons. While such a retransmission causes delay, our time analysis does not cover it. In addition, due to the existence of retransmissions, the network needs to prepare a certain central system that controls the status of the whole network. It is another future problem to design the implementation of our system taking care of these issues. These are challenging future studies.

Funding: This work was supported in part by the Japan Society of the Promotion of Science (JSPS) Grant-in-Aid for Scientific Research (B): 16KT0017; the Japan Society of the Promotion of Science (JSPS) Grant-in-Aid for Scientific Research (A): 17H01280; the Japan Society of the Promotion of Science (JSPS) Grant-in-Aid for Scientific Research (C): 16K00014; the Kayamori Foundation of Informational Science Advancement: K27-XX-467; and Guangdong Provincial Key Laboratory: 2019B121203002.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable.

Acknowledgments: The author is thankful to Ángeles Vazquez-Castro, Tadashi Wadayama, and Satoshi Takabe for discussions for secure PLNC. The author is also grateful to Go Kato and Masaki Owari for discussions on secure NC.

Conflicts of Interest: The author declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; nor in the decision to publish the results.

References

1. Han, Z.; Sun, Y.L. Distributed cooperative transmission with unreliable and untrustworthy relay channels. *EURASIP J. Wirel. Commun. Netw.* **2009**, *2009*, 740912. [CrossRef]
2. Sun, L.; Zhang, T.; Li, Y.; Niu, H. Performance study of two-hop amplify-and-forward systems with untrustworthy relay nodes. *IEEE Trans. Veh. Technol.* **2012**, *61*, 3801–3807. [CrossRef]
3. Sun, L.; Ren, P.; Du, Q.; Wang, Y.; Gao, Z. Security-aware relaying scheme for cooperative networks with untrusted relay nodes. *IEEE Commun. Lett.* **2015**, *19*, 463–466. [CrossRef]
4. Kim, J.-B.; Lim, J.; Cioffi, J.M. Capacity scaling and diversity order for secure cooperative relaying with untrustworthy relays. *IEEE Trans. Wirel. Commun.* **2015**, *14*, 3866–3876. [CrossRef]
5. Xiong, J.; Cheng, L.; Ma, D.; Wei, J. Destination-aided cooperative jamming for dual-hop amplify-and-forward mimo untrusted relay systems. *IEEE Trans. Veh. Technol.* **2016**, *65*, 7274–7284. [CrossRef]
6. Khodakarami, H.; Lahouti, F. Link adaptation with untrusted relay assignment: Design and performance analysis. *IEEE Trans. Commun.* **2013**, *61*, 4874–4883. [CrossRef]
7. Ren, Z.; Goseling, J.; Weber, J.H.; Gastpar, M. Secure Transmission Using an Untrusted Relay with Scaled Compute-and-Forward. In Proceedings of the 2015 IEEE Information Theory Workshop (ITW), Jerusalem, Israel, 26 April–1 May 2015.
8. He, X.; Yener, A. End-to-end secure multi-hop communication with untrusted relays. *IEEE. Trans. Wirel. Comm.* **2013**, *12*, 1–11. [CrossRef]
9. He, X.; Yener, A. Strong secrecy and reliable Byzantine detection in the presence of an untrusted relay. *IEEE Trans. Inform. Theory* **2013**, *59*, 177–192. [CrossRef]
10. Vatedka, S.; Kashyap, N.; Thangaraj, A. Secure Compute-and-Forward in a Bidirectional Relay. *IEEE Trans. Inform. Theory* **2015**, *61*, 2531–2556. [CrossRef]
11. Zewail, A.A.; Yener, A. The Two-Hop Interference Untrusted-Relay Channel with Confidential Messages. In Proceedings of the IEEE Information Theory Workshop 2015, Jeju, Korea, 11–15 October 2015; pp. 322–326.

12. Cai, N.; Yeung, R. Secure network coding. In Proceedings of the IEEE International Symposium on Information Theory (ISIT 2002), Lausanne, Switzerland, 30 June–5 July 2002; p. 323.
13. Cai, N.; Yeung, R.W. Secure Network Coding on a Wiretap Network. *IEEE Trans. Inform. Theory* **2011**, *57*, 424–435. [CrossRef]
14. Yeung, R.W.; Cai, N. On the optimality of a construction of secure network codes. In Proceedings of the 2008 IEEE International Symposium on Information Theory (ISIT 2008), Toronto, ON, Canada, 6–11 July 2008; pp. 166–170.
15. Chan, T.; Grant, A. Capacity bounds for secure network coding. In Proceedings of the 2008 Australian Communications Theory Workshop, Christchurch, New Zealand, 30 January–1 February 2008; pp. 95–100.
16. el Rouayheb, S.; Soljanin, E.; Sprintson, A. Secure network coding for wiretap, networks of type II. *IEEE Trans. Inf. Theory* **2012**, *58*, 1361–1371. [CrossRef]
17. Feldman, J.; Malkin, T.; Stein, C.; Servedio, R.A. On the capacity of secure network coding. In Proceedings of the 42nd Annual Allerton Conference on Communication, Control, and Computing, Monticello, IL, USA, 29 September–1 October 2004.
18. Zhang, S.; Liew, S.C.; Lam, P.P. Hot topic: Physical-layer network coding. In Proceedings of the 12th Annual International Conference on Mobile Computing and Networking (MobiCom 2006), Angeles, CA, USA, 23–29 September 2006; pp. 358–365.
19. Nazer, B.; Gastpar, M. Computing over multiple-access channels with connections to wireless network coding. In Proceedings of the 2006 IEEE International Symposium on Information Theory (ISIT 2006), Seattle, WA, USA, 9–14 July 2006; pp. 1354–1358.
20. Popovski, P.; Yomo, H. The anti-packets can increase the achievable throughput of a wireless multi-hop network. In Proceedings of the IEEE International Conference on Communications, 2006 (ICC 2006), Istanbul, Turkey, 11–15 June 2006; pp. 3885–3890.
21. Hayashi, M.; Wadayama, T.; Vázquez-Castro, A. Secure Computation-and-Forward Communication with Linear Codes. *IEEE J. Sel. Areas Inf. Theory* **2021**, *2*, 139–148. [CrossRef]
22. Ahlswede, R.; Cai, N.; Li, S.-Y.R.; Yeung, R.W. Network information flow. *IEEE Trans. Inf. Theory* **2000**, *46*, 1204–1216. [CrossRef]
23. Cai, N.; Hayashi, M. Secure Network Code for Adaptive and Active Attacks with No-Randomness in Intermediate Nodes. *IEEE Trans. Inf. Theory* **2020**, *66*, 1428–1448. [CrossRef]
24. Nazer, B.; Gastpar, M. Compute-and-forward: Harnessing interference through structured codes. *IEEE Trans. Inf. Theory* **2011**, *57*, 6463–6486. [CrossRef]
25. Hong, S.; Caire, G. Compute-and-Forward Strategies for Cooperative Distributed Antenna Systems. *IEEE Trans. Inform. Theory* **2013**, *59*, 5227–5243. [CrossRef]
26. Wei, L.; Chen, W. Compute-and-Forward Network Coding Design over Multi-Source Multi-Relay Channels. *IEEE Trans. Wirel. Commun.* **2012**, *11*, 3348–3357. [CrossRef]
27. Nokleby, M.; Aazhang, B. Cooperative Compute-and-Forward. *IEEE Trans. Wirel. Commun.* **2016**, *15*, 14–27. [CrossRef]
28. Kienle, F.; Brack, T.; Wehn, N. A synthesizable IP core for DVB-S2 LDPC code decoding. In Proceedings of the Design, Automation and Test in Europe, Munich, Germany, 7–11 March 2005; Volume 3, pp. 100–105.
29. Takabe, S.; Wadayama, T.; Hayashi, M. Asymptotic Behavior of Spatial Coupling LDPC Coding for Compute-and-Forward Two-Way Relaying. *IEEE Trans. Commun.* **2020**, *68*, 4063–4072. [CrossRef]
30. Sula, E.; Zhu, J.; Pastore, A.; Lim, S.H.; Gastpar, M. Compute—Forward Multiple Access (CFMA) with Nested LDPC Codes. In Proceedings of the 2017 IEEE International Symposium on Information Theory (ISIT 2017), Aachen, Germany, 25–30 June 2017; pp. 2935–2939.
31. Hayashi, M.; Vázquez-Castro, Á. Physical Layer Computation as NOMA for Integrated Wireless Systems. *IEEE Trans. Commun.* **2021**, *69*, 4520–4535. [CrossRef]
32. Hayashi, M. Secure physical layer network coding versus secure network coding. In Proceedings of the 2018 IEEE Information Theory Workshop (ITW 2018), Guangzhou, China, 25–29 November 2018; pp. 430–434.
33. Nazer, B.; Gastpar, M. Compute-and-forward: A novel strategy for cooperative networks. In Proceedings of the 2008 42nd Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, USA, 26–29 October 2008; pp. 69–73.
34. Nazer, B.; Cadambe, V.R.; Ntranos, V.; Caire, G. Expanding the compute-and-forward framework: Unequal powers, signal levels, and multiple linear combinations. *IEEE Trans. Inform. Theory* **2016**, *62*, 4879–4909. [CrossRef]
35. Ullah, S.S.; Liva, G.; Liew, S.C. Physical-layer Network Coding: A Random Coding Error Exponent Perspective. In Proceedings of the 2017 IEEE Information Theory Workshop (ITW 2017), Kaohsiung, Taiwan, 5–10 November 2017; pp. 559–563.
36. Hayashi, M.; Vázquez-Castro, Á. Computation-aided classical-quantum multiple access to boost network communication speeds. *Phys. Rev. Appl.* **2021**, *16*, 054021. [CrossRef]
37. Hayashi, M. Secure Modulo Sum via Multiple Access Channel. In Proceedings of the 2021 IEEE International Symposium on Information Theory (ISIT 2021), Melbourne, Victoria, Australia, 12–20 July 2021; pp. 1397–1402.
38. Owari, M.; Kato, G.; Hayashi, M. Single-Shot Secure Quantum Network Coding on Butterfly Network with Free Public Communication. *Quantum Sci. Technol.* **2017**, *3*, 014001. [CrossRef]
39. Letafati, M.; Kuhestani, A.; Behroozi, H.; Ng, D.W.K. Jamming-Resilient Frequency Hopping-Aided Secure Communication for Internet-of-Things in the Presence of an Untrusted Relay. *IEEE Trans. Wirel. Commun.* **2020**, *19*, 6771–6785. [CrossRef]
40. Shiu, Y.-S.; Chang, S.Y.; Wu, H.-C.; Huang, S.C.-H.; Chen, H.-H. Physical layer security in wireless networks: A tutorial. *IEEE Wirel. Commun.* **2011**, *18*, 66–74. [CrossRef]
41. Zeng, K. Physical Layer Key Generation in Wireless Networks: Challenges and Opportunities. *IEEE Commun. Mag.* **2015**, *53*, 33–39. [CrossRef]
42. Carter, L.; Wegman, M. Universal classes of hash functions. *J. Comput. Syst. Sci.* **1979**, *18*, 143–154. [CrossRef]

43. Wegman, M.N.; Carter, J.L. New Hash Functions and Their Use in Authentication and Set Inequality. *J. Comput. Syst. Sci.* **1981**, *22*, 265–279. [CrossRef]
44. Krawczyk, H. *Advances in Cryptology CRYPTO 1994*; Lecture Notes in Computer Science; Springer: New York, NY, USA, 1994; Volume 893, p. 129.
45. Krawczyk, H. *Advances in Cryptology EUROCRYPT1995*; Springer: New York, NY, USA, 1995; Volume 921, p. 301.
46. Mamaghani, M.T.; Kuhestani, A.; Behroozi, H. Can a multi-hop link relying on untrusted amplify-and-forward relays render security? *Wirel. Netw.* **2021**, *27*, 795–807. [CrossRef]
47. Forouzesh, M.; Azmi, P.; Kuhestani, A.; Yeoh, P.L. Joint Information-Theoretic Secrecy and Covert Communication in the Presence of an Untrusted User and Warden. *IEEE Internet Things J.* **2021**, *8*, 7170–7181. [CrossRef]
48. Forouzesh, M.; Azmi, P.; Kuhestani, A.; Yeoh, P.L. Covert Communication and Secure Transmission Over Untrusted Relaying Networks in the Presence of Multiple Wardens. *IEEE Trans. Commun.* **2020**, *68*, 3737–3749. [CrossRef]