University of Baghdad
College of Engineering
JOURNAL OF ENGINEERING

Journal of Engineering

journal homepage: www.joe.uobaghdad.edu.iq

Volume 29 Number 2 February 2023

Double-Staged Syndrome Coding Scheme for Improving Information Transmission Security over the Wiretap Channel

Sarwat Ali Ahmed*
MSc. Student in Communication Engineering
College of Engineering - Sulaimani Polytechnic University
Technical Institute of Sulaimani - Sulaimani Polytechnic University.
Sulaymaniyah, Iraq
sarwat.ali.a@spu.edu.iq

Asaad Mubdir Jassim Al-Hindawi
Ph.D. Radio and Microwave Communication Engineering
College of Engineering - Sulaimani Polytechnic University.
Sulaymaniyah, Iraq
asaad.jasim@spu.edu.iq

ABSTRACT

This paper presents a study of a syndrome coding scheme for different binary linear error-correcting codes that refer to the code families such as BCH, BKLC, Golay, and Hamming. The study is implemented on Wyner's wiretap channel model when the main channel is error-free and the eavesdropper channel is a binary symmetric channel with crossover error probability ($0 < Pe \leq 0.5$) to show the security performance of error correcting codes while used in the single-staged syndrome coding scheme in terms of equivocation rate. Generally, these codes are not designed for secure information transmission, and they have low equivocation rates when they are used in the syndrome coding scheme. Therefore, to improve the transmission security when using these codes, a modified encoder which consists of a double-staged syndrome coding scheme, is proposed. Two models are implemented in this paper: the first model utilizes one encoding stage of the conventional syndrome coding scheme. In contrast, the second model utilizes two encoding stages of the syndrome coding scheme to improve the results obtained from the first model. The C++ programming language, in conjunction with the NTL library, is used for obtaining simulation results for the implemented models. The equivocation rate results from the second model were compared to both the results of the first model and the unsecured transmission (transmission of data without encryption). The comparison revealed that the security performance of the second model is better than the first model and the insecure system, as the equivocation for all the simulated codes over the proposed model reaches at least %97 at the $P_e = 0.1$.

Keywords: Wiretap channel, Syndrome coding, Binary linear codes, BCH, BKLC, Golay, Hamming.

*Corresponding author
Peer review under the responsibility of University of Baghdad.
https://doi.org/10.31026/j.eng.2023.02.08

Article received: 2/8/2022
Article accepted: 2/9/2022
Article published: 1/2/2023

مخطط التشفير المتلازمة المتكونة من مرحلتين تشفير لتحسين أمن نقل المعلومات عبر قناة التنصت

سه روه ت على احمد
طالب ماجستير في هندسة الاتصالات–كلية الهنسة–جامعة السليمانية التقنية
قسم الاتصالات–المعهد التقني في السليمانية– جامعة السليمانية التقنية

اسعد مبدر جاسم الهنداوي
دكتوراه في هندسة الاتصالات اللاسلكية والميكروويف
كلية الهندسة– جامعة السليمانية التقنية

الخلاصة

يقدم هذا البحث دراسة لمخطط ترميز المتلازمة لمختلف رموز تصحيح الخطأ الخطي الثنائي التي تشير إلى عائلات الكود مثل BCH و BKLC و Golay و Hamming. تم تنفيذ الدراسة على نموذج Wyner لقناة التنصت على المعلومات ، عندما تكون القناة الرئيسية خالية من الأخطاء وقناة التنصت هي قناة ثنائية متماثلة مع احتمال اضافة الخطأ بنسبة ($0<Pe\leq0.5$) ، لإظهار الأداء الأمني من حيث معدل المراوغة لرموز تصحيح الخطأ أثناء استخدامها في مخطط ترميز متلازمة المتكون من مرحلة الواحدة. بشكل عام ، لم يتم تصميم هذه الرموز لنقل المعلومات بشكل آمن ، ولها معدلات مراوغة منخفضة عند استخدامها في مخطط ترميز المتلازمة. لذلك ، لتحسين أمان الإرسال عند استخدام هذه الرموز ، يُقترح تشفير معدل يتكون من مخطط تشفير متلازمة متكون من مرحلتين تشفير. تم تنفيذ نموذجين في هذا البحث: النموذج الأول يستخدم مرحلة ترميز واحدة لمخطط ترميز المتلازمة التقليدية ، بينما يستخدم النموذج الثاني مرحلتين من مخطط ترميز المتلازمة لتحسين النتائج التي تم الحصول عليها من النموذج الأول. تستخدم لغة البرمجة C++ جنبًا إلى جنب مع مكتبة NTL للحصول على نتائج محاكاة للنماذج المنفذة. تمت مقارنة نتائج معدل التباس من النموذج الثاني مع كل من نتائج النموذج الأول ونتائج الإرسال غير الآمن (نقل البيانات بدون تشفير). كشفت المقارنة أن الأداء الأمني للنموذج الثاني أفضل من النموذج الأول وأيضًا النظام غير الآمن ، حيث أن معدل المراوغة لجميع الأكواد اللتي تم استخدامهم على النموذج المقترح تصل على الأقل 97٪ في $Pe = 0.1$.

الكلمات الرئيسية: قناة التنصت ، ترميز المتلازمة ، الرموز الخطية الثنائية ، BCH، BKLC، Golay، Hamming.

1. INTRODUCTION

Physical layer security of wireless communication is a problem that has been studied by specialists who were working on ensuring the confidentiality of data transmission between legitimate users. The issue of secure communication was first studied by Shannon **(Shannon, 1948)** from an information theoretical perspective. To achieve secure communication in Shannon's model, a secret key is shared between the legitimate users while it is concealed from the eavesdropper.

Secure and reliable communication is made using coding techniques for data transmission. The basic physical layer model that contains theoretical foundations to capture the essence of communication security and reliability is called the wiretap channel, which Wyner introduced **(Wyner, 1975)**. In Wyner's model, the transmitter (Tr.) transmits confidential

information to the legitimate receiver (Re.) through the main channel without sharing a secret key between them, and the eavesdropper (Ev.) obtains information from another channel called the wiretap channel which is assumed to be a degraded version of the main channel. To increase transmission security, Ozarow and Wyner **(Ozarow & Wyner, 1984)** proposed the syndrome coding scheme for a special case of the wiretap channel when the main channel is noiseless, and the eavesdropper channel is a Binary Symmetric Channel (BSC).

In the syndrome coding scheme, the message is conveyed as the syndrome of the code. The transmission security is measured in terms of the equivocation rate at the eavesdropper's side. There are codes designed especially for a syndrome coding scheme that give high equivocation rates **(Gazi, 2020)**. Error-correcting codes are used for reliable communication between legitimate users **(Kadum, et al., 2020)**. Furthermore, they can be used in a syndrome coding scheme to increase transmission security. Still, they cannot give high equivocation rates as specially designed codes for syndrome coding schemes **(Moon, 2021)**. Thus, to increase the security performance of error-correcting codes while used in syndrome coding schemes, additional encoding techniques or modifications to the code can be made at the encoder which makes the equivocation rate at the eavesdropper's side higher and the amount of information leakage to the eavesdropper lower.

Numerous researchers have investigated data transmission over the wiretap channel and its coding schemes for different tapping channels. The problem of strong secrecy has been studied over arbitrarily varying wiretap channels **(Chen, et al., 2022)**. From an information theoretic perspective for providing security of data transmission and on the bases of randomized coset coding, finite length codes have been studied for Gaussian wiretap channel **(Nooraiepour, et al., 2020)**. Additionally, **(Harrison, et al., 2019)** achieved reliable and secure communication over the Gaussian wiretap channel have been studied by considering the pros and cons of applying different keyless coding layers. A technique to extend the columns of the parity-check matrix was proposed **(AL-Hassan, et al., 2014)**. The utilized technique increased the equivocation rate compared with codes in Grassl's online database, which is available at **(Grassl, 2007)**. **(Zhang, et al., 2013)** used an encoder with two encoding stages to improve the security of data transmission. The first stage of syndrome coding was with (23, 12, 7) Golay code, and the second with modified McEliece public-key encryption. Increasing the security in this proposal brought down the information rate on the eavesdropper's side. Additionally, **(Al-Hassan, et al., 2013)** presented a system of twencoding stages to minimize the information leakage and maximize the equivocation rate at the eavesdropper's side. The first stage of the encoder was based on the syndrome coding scheme of (23, 12, 7) Golay code, and the second stage employed two models using the technique of McEliece cryptosystem with two types of Best Known Linear Codes (BKLC). Compared with the results of **(Zhang, et al., 2013),** the equivocation results of **(Al-Hassan, et al., 2013)** showed better equivocation rates.

This paper investigated the syndrome coding scheme with a single-staged encoder for the binary linear error correcting codes such as Bose–Chaudhuri–Hocquenghem (BCH), BKLC, Golay, and Hamming codes for the wiretap channel. The system model in **Fig. 1** shows that the main channel is error-free, and the eavesdropper channel is a BSC with a crossover probability ($0 < Pe \leq 0.5$). Then, we propose a system that employs an encoder with two encoding stages of the syndrome coding scheme: the first stage with the abovementioned codes and the second with different BKLCs compatible with the output from the first stage. Compared with the syndrome coding system of single-stage encoding and the unsecured

system, the proposed system with two encoding stages has a higher equivocation rate and lower information leakage. By this outcome, this study adds another secure system to be ready for implementation in real wireless communication applications.

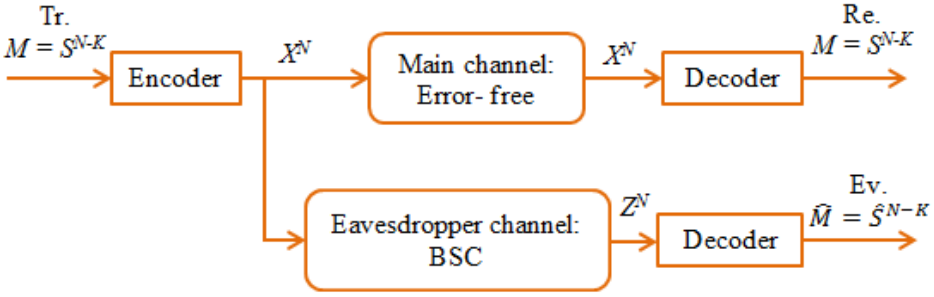

Figure 1. The block diagram of the syndrome coding scheme over the wiretap channel

2. METHODS AND MATERIALS

2.1 The Syndrome Coding Scheme Concept

A syndrome coding scheme over the wiretap channel is employed to increase the security of data transmission. Error-correcting codes having parameters (*n, k, d*) are used mainly for reliable communication between the transmitter and the legitimate receiver and to combat noisy transmission problems when information is transmitted over noisy channels. These codes can also be used in syndrome coding schemes to increase the security of information transmission. In the syndrome coding scheme, the message is encoded as the syndrome of the code and then transmitted. To measure the security level for error-correcting codes when they are used in the syndrome coding scheme, the equivocation rate (secrecy capacity) or the information leakage (channel capacity) at the eavesdropper side should be measured. The wiretap channel model presented in this paper has three users; the transmitter (Tr.), the legitimate receiver (Re.), and the eavesdropper (Ev.). It is assumed that the channel between the transmitter and the legitimate receiver is error-free, while the eavesdropper obtains a degraded version of the transmitted codewords from a BSC. Two models of encoder have been used over the wiretap channel. The first model consists of one encoding stage of the syndrome coding scheme, which is used to know how binary error correcting codes such as BCH, BKLC, Hamming, and Golay codes perform in terms of equivocation rate. The second model is implemented to improve the obtained results from the first model. This model consists of an encoder with two encoding stages: the first stage consists of a syndrome coding scheme using the binary linear BCH, BKLC, Hamming, and Golay codes, and the second stage consists of a syndrome coding scheme using BKLCs (those are compatible with the output from the first stage). The equivocation rate of the utilized codes for both models has been compared with the insecure system results. The NTL library, which is a C++ portable library with high performance in mathematical calculations and arithmetic, is used with C++ coding language for writing the simulation codes. The results are plotted using NTL in conjunction with GMP (the GNU multi-precision and high-performance tool for plotting).

The system is running (Ubuntu 20.04.1 LTS) operating system. Moreover, the Magma software suite obtained the generator and parity-check matrices for all codes.
Based on the crossover probability of the BSC, the maximum transmission rate which can be achieved from this channel when perfect secrecy has been maintained is defined as the secrecy capacity of the wiretap channel. Wyner **(Wyner, 1975)** showed the secrecy capacity as:

$$C_{S-BSC} = -P_e.\log_2 P_e - (1-P_e).\log_2(1-P_e) \quad (1)$$

Where; P_e is the error probability of transmission through the BSC. Then, the BSC capacity can be shown as:

$$C_{BSC} = 1 - C_{S-BSC} = 1 + P_e.\log_2 P_e + (1-P_e).\log_2(1-P_e) \quad (2)$$

For binary (n, k, d) error correcting codes the block length of the codeword c is *n-bits*, the block length of the information m is *k-bits*, the length of the parity bits that will be added to the information is ($s = n - k$), and the minimum hamming distance of the code is d. On the receiver side, the error detection and correction capability of the code can be obtained from d. Error-correcting codes can correct errors up to t bits, and

$$t \leq \left\lfloor \frac{d_{min} - 1}{2} \right\rfloor \quad (3)$$

Where t is the number of errors that occurred in the error pattern, and ⌊-⌋ is the floor function. Error-correcting codes can be classified into perfect and non-perfect codes. The perfect codes satisfy the Hamming bound (i.e., the number of codewords is equal to the Hamming bound); however, non-perfect codes are not. The hamming bound and several correctable error patterns are given by Eq. (4) and (5).

$$|C| \leq \frac{2^n}{\sum_{i=0}^{t} \binom{n}{i}} \quad (4)$$

$$\sum_{i=0}^{t} \binom{n}{i} = \sum_{i=0}^{t} \frac{n!}{i!\ x\ (n-i)!} \quad (5)$$

For binary linear code C, which has parameters (n, k, d), the generator matrix [G], and the parity-check matrix [H] over the field F_2, the codeword $c \in F_2^n$ and the syndrome $s \in F_2^{n-k}$ of $c \in C$ over F_2 can be defined as:

$$c = m.[G] \quad (6)$$

$$s = c.[H^T] \quad (7)$$

The generator matrix and the parity-check matrix of the linear code are orthogonal, therefor $[G] \times [H^T] = [0]$. The length of a binary linear code syndrome is equal to the length of the parity bits added to the information bits ($s = n - k$). The received vector R at the receiver combines a transmitted codeword C and an unwanted error pattern E as the effect of adding noise by the channel during transmission (i.e., $R = C + E$). The syndrome results from a parity

check performed on *R* to know whether *R* is a vector of the codeword set or not. If *R* is a valid codeword, the value of *S* will be an all-zero vector. If *R* contains errors, the syndrome will contain some non-zero elements. The non-zero syndrome will allocate the particular error pattern if the detectable error is correctable, and the forward error correcting decoder corrects the error. The parity check for the received codeword is as follows:

$$s = R.H^T = (C + E).H^T = 0 \; when \; R \in C \; (i.e. E = 0) \quad (8)$$

$$s = R.H^T \neq 0 \; when \; R \neq C \; (i.e. E \neq 0) \quad (9)$$

Each syndrome corresponds to an error pattern. The number of syndromes for binary error correcting code is $S = 2^{n-k}$. Therefore, there are 2^{n-k} correctable error patterns. The codeword set of the binary linear (*n, k, d*) code is F_2^nIt contains 2^k codewords called the coset, and the cost leader which is the lightest weight codeword in this coset is the all-zero codeword. If any correctable error pattern is added to these codewords by the noisy channel during transmission, the decoder will detect it by parity-check procedure on the received vector.
In this paper syndrome coding scheme has been implemented. In the syndrome coding scheme, the message is encoded as the syndrome of the error correcting code (i.e., the length of the message in this scheme is set to be ($m = s = n - k$). The transmitter encodes the information as the syndrome of binary error correcting code that has the length of *m*-bits, generates and transmits *n*-bits codeword *C(i)* to the legitimate receiver. The information sequence is *m(1)... m(S=2^m)*, and the codeword sequence is *c(1)... c(S)*. As the channel between the transmitter and the legitimate receiver is noiseless, the receiver receives the same codeword *C(i)* that Alice transmits. However, the eavesdropper obtains corrupted codeword *Z* through the BSC as the BSC adds an error pattern E_{BSC} of length *n*-bits to the transmitted codewords according to the P_e. So, the received vector by the eavesdropper is $Z(i) = C(i) + E_{BSC}(i)$, for *i* to be an instant time of transmission and equal to *S*.
To send all 2^m syndromes of the binary (*n, k, d*) code in the traditional syndrome coding scheme, a look-up table containing all syndromes and corresponding error patterns must be created and saved by all users. If the number of messages to be encoded by syndrome coding is equal to the denominator of the Hamming bound (i.e. $2^m = \sum_{i=0}^{t} \binom{n}{i}$), the code is said to be perfect in the syndrome coding scheme; otherwise, it is a non-perfect code.
For codes with large numbers of syndromes and error patterns, the look-up table is impractical as the creation of the look-up table is complex and needs large memory space. Also, for non-perfect codes, when the look-up table is created, there are messages which cannot be sent, as the match between the messages and error patterns is not one-to-one. For those reasons, the parity-check matrix of the binary code is put on the standard form [I_m | P^T_k], the generator matrix is put on the standard form [P_m | I_k] and the corresponding error pattern of the syndrome (message) can be created by padding the message by *k-bits* of zero. The standard form of the parity-check matrix will be sufficient for the encoding and decoding procedures.

2.2 The First Model

The model that is shown in **Fig. 2** implements one stage of syndrome coding for different binary linear BCH, BKLC, Hamming, and Golay codes with code parameters (*n, k, d*). The users

of this model are applying special algorithms to deal with their data. The binary code parameters of this stage will be recognized as (n_1, k_1, d_1). The length of the codeword C_1 is n_1-bits, the message length of error correcting code is k_1-bits, the minimum hamming distance of the code is d_1-bits, and the syndrome length of the code is m-bits which equals (n_1 - k_1). It is essential to state that the standard form of the generator and the parity-check matrices are used in this model.

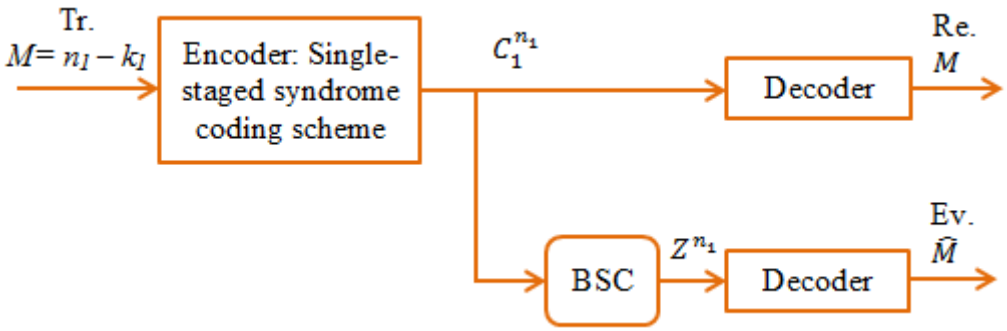

Figure 2. Single-stage syndrome coding scheme over the wiretap channel

The block diagrams and the algorithms for the transmitter's encoder, the legitimate receiver, and the eavesdropper's decoders are shown and explained as follows:

2.2.1 The transmitter's encoder

The transmitter starts encoding *m-bits* message block of length ($n_1 - k_1$) to generate C_1-bits codeword of length (n_1) such that $M(i) = C_1(i).[H_1^T]$. The block diagram in **Fig. 3** shows the single-staged encoder using the syndrome coding scheme for binary linear BCH, BKLC, Golay, and Hamming codes. At first, the transmitter's encoder generates the binary block messages of length ($M = n_1 - k_1$), and generates the error pattern E_1 of length n_1 for each message either from the syndrome look-up table or by padding the message block by k_1-*bits* of zero, generates a random vector C_{R1} of length n_1 by multiplying the random data vector DR_1 of length k_1 by the generator matrix of the linear code [G_1], and finally adds E_1 with C_{R1} to

generate the codeword C_1 of length n_1 and sends it to the legitimate receiver. The encoding process by the transmitter follows algorithm#1.

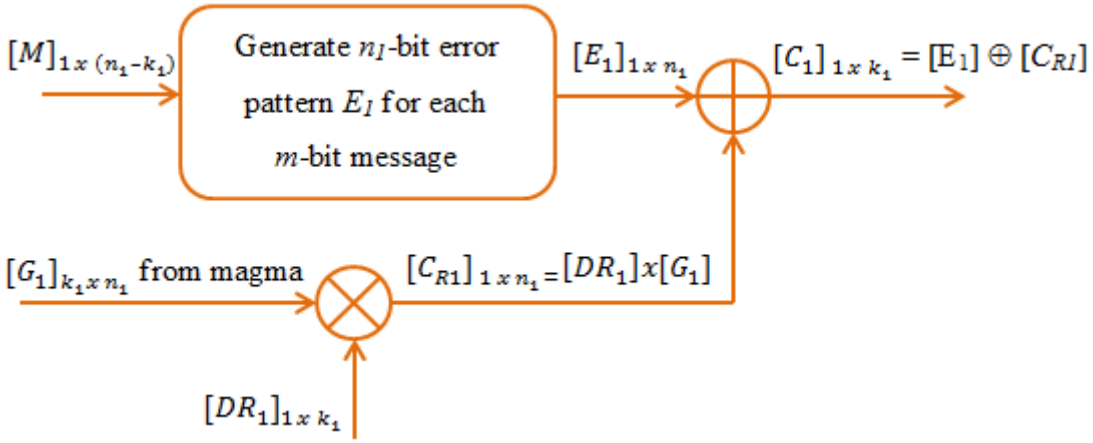

Figure 3. The transmitter's single-staged syndrome coding encoder

Algorithm#1: The encoding process for the binary linear (n_1, k_1, d_1) code.

Input: Generator matrix $[G_1]$, transpose of the parity-check matrix $[H_1^T]$, and random data vector $[DR_1]$ are required for the following encoding steps:

[1] Obtaining $[G_1]_{k1 x n1}$ and $[H_1]_{(n1-k1) x n1}$ from the Magma software suite and putting them on the standard form of $[P_{k1}/I_m]$ and $[I_m/P^T_{k1}]$, respectively.
[2] Calculating the transpose of the parity-check matrix $[H_1^T]_{n1x(n1-k1)}$
[3] Generating $[DR_1]_{1 x k1}$: (a random data of length k_1 will be generated)
[4] Calculating $[C_{R1}]_{1 x n1}$

$$[C_{R1}]_{1 x n1} = [DR_1]_{1 x k1} x\ [G_1]_{k1 x n1} \tag{10}$$

[5] Generating the error pattern $[E_1]_{1 x n1}$ for each message $[M]_{1 x (n1 - k1)}$ from:

- Either the syndrome look-up table: The error pattern that satisfies this equation for perfect codes was chosen.

$$[M]_{1x(n1-k1)} = [E_1]_{1xn1} x\ [H_1^T] \tag{11}$$

- Or by padding the message vector by k_1-bits of zero: for non-perfect codes

$$[E_1]_{1 x n1} = M_{(n1 - k1)} | \ 00..00_{k1} \tag{12}$$

[6] Calculating the codeword to be transmitted $[C_1]_{1 x n1}$

$$[C_1]_{1 x n1} = [E_1]_{1 x n1} \oplus [C_{R1}]_{1 x n1} \tag{13}$$

Output: Return $[C_1]_{1 x n1}$ that is $C_1(i) = E_1(i) + DR_1(i).[G_1]$

2.2.2 The legitimate receiver's decoder

The legitimate receiver's decoder is shown in **Fig. 4**. The legitimate receiver receives the same transmitted codeword C_1 of length n_1-bits due to an error-free communication channel, and he performs syndrome decoding using the transpose of the parity-check matrix $[H_1^T]$ to obtain the original message M (i.e., $m = n_1 - k_1$). The receiver utilizes algorithm#2, which explains the step-by-step decoding procedure of the received codewords.

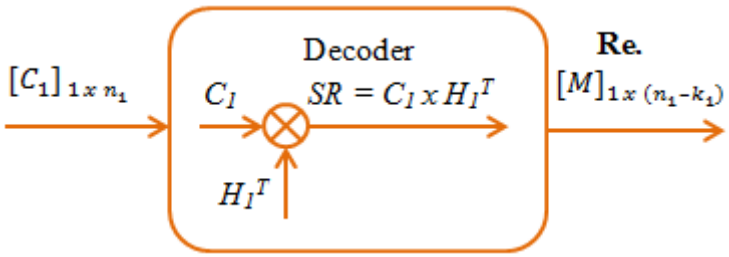

Figure 4. Single-stage legitimate receiver's decoder

Algorithm#2: The decoding algorithm of the received binary codeword vector C_1.

Input: The transpose of the standard form of the parity-check matrix $[H_1^T]$ and the received codeword C_1 are required for the following decoding steps:

[1] Obtaining $[H_1]_{(n1-k1) x n1}$ from Magma, then put it on the standard form $[I_m \mid P^T{}_{k1}]$
[2] Generating the transpose of the parity-check matrix $[H_1^T]_{n1 x (n1-k1)}$ from $[H_1]_{(n1-k1) x n1}$
[3] Calculating the received syndrome $[SR]_{1 x (n1-k1)} = [C_1]_{1 x n1} \times [H_1{}^T]_{n1 x (n1-k1)}$
[4] Setting $[M]_{1 x (n1-k1)} = [SR]_{1 x (n1-k1)}$

Output: Return $[M]_{1 x (n1-k1)}$

Because E_1 is generated either from the look-up table or padding M by k_1-bit zeros, and the parity check matrix is put on the standard form, the receiver's recovery of the original encoded message M by the transmitter can be proved from algorithm#2 and Eq. (10, 11, 12 and 13) as follows:

$$SR(i) = C_1(i)\ x\ [H_1^T]$$
$$SR(i) = \left(E_1(i) + C_{R1}(i)\right)\ x\ [H_1^T]$$
$$SR(i) = (E_1(i) + DR_1(i)\ x\ [G_1])\ x\ [H_1^T]$$
$$SR(i) = E_1(i) x\ [H_1^T] + DR_1(i)\ x\ [G_1]\ x\ [H_1^T], \text{ because } [G] \times [H^T] = [0]$$
$$SR(i) = E_1(i)\ x\ [H_1^T] + 0$$
$$SR(i) = M(i)$$

2.2.3 The eavesdropper's decoder

The block diagram of the eavesdropper's decoder is shown in **Fig. 5.** Eve receives the degraded vector Z of length n_1 instead of C_1 as a result of adding random error sequence E_{BSC} of length n_1 to the transmitted codeword C_1 by the eavesdropping channel that is a BSC. The *BSC produces the E*BSC based on the crossover probability of the channel ($0 < P_e \leq 0.5$).

$$[Z]_{1\,x\,n1} = [C_1]_{1\,x\,n1} \oplus [E_{BSC}]_{1\,x\,n1} \tag{14}$$

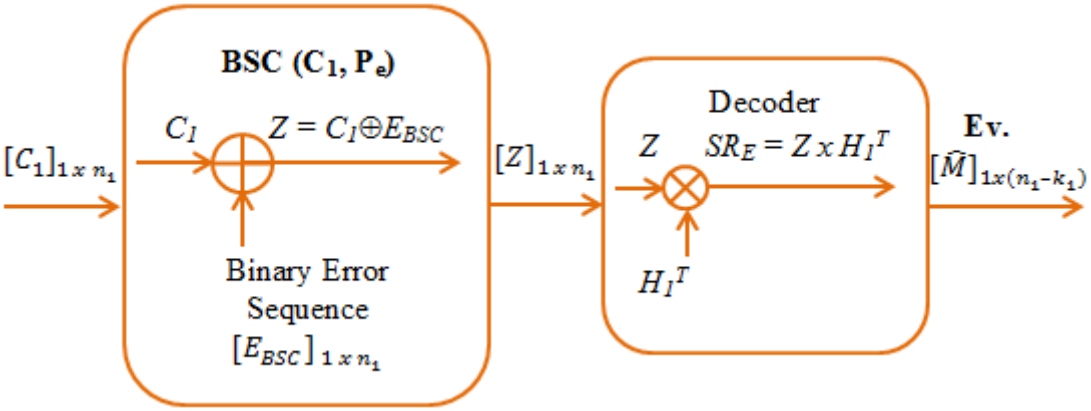

Figure 5. The block diagram for the eavesdropper's channel and decoder

Eve follows the step-by-step procedure in algorithm#3 to decode the corrupted vector Z.

Algorithm#3: The decoding algorithm of the received binary vector Z from the BSC.

Input: The transpose of the parity-check matrix $[H_1^T]$ and the received vector $[Z]$ are required for the following decoding steps:

[1] Obtaining $[H_1]_{(n1-k1)\,x\,n1}$ from Magma, then put it on the standard form $[I_m \mid P^T_{k1}]$
[2] Generating the transpose of the parity-check matrix $[H_1^T]_{n1\,x\,(n1-k1)}$ from $[H_1]_{(n1-k1)\,x\,n1}$
[3] Calculating eavesdropper's received syndrome $[SR_E]_{1\,x\,(n1-k1)} = [Z]_{1\,x\,n1}$ x $[H_1^T]_{n1\,x\,(n1-k1)}$
[4] Set $[\widehat{M}]_{1\,x\,(n1-k1)} = [SR_E]_{1\,x\,(n1-k1)} = [M]_{1\,x\,(n1-k1)} \oplus [S_c]_{1\,x\,(n1-k1)}$

Output: Return $[\widehat{M}]_{1\,x\,(n1-k1)}$

Even if the eavesdropper's decoder is assumed to be the same type as the legitimate receiver's, he cannot recover the original message M that is encoded by the transmitter as a corrupted syndrome S_c will be added to the original encoded syndrome M due to random error addition by the BSC; instead, he recovers the estimate of the original message $\acute{M}$. Based on Eq. (10, 11, 12, 13, and 14), the proof to show how the eavesdropper cannot obtain the original encoded message M and can only obtain the estimation of the message is explained as follows:

$SR_E(i) = Z(i)\ x\ [H_1^T]$, from the output of the BSC: $Z(i) = C_1(i) + E_{BSC}(i)$
$SR_E(i) = (C_1(i) + E_{BSC}(i))\ x\ [H_1^T]$
$SR_E(i) = C_1(i)\ x\ [H_1^T] + E_{BSC}(i) x\ [H_1^T]$
$SR_E(i) = (E_1(i) + DR_1(i) x\ [G_1]) x\ [H_1^T] + E_{BSC}(i) x\ [H_1^T]$
$SR_E(i) = E_1(i) x [H_1^T] + DR_1(i)\ x\ [G_1]\ x\ [H_1^T] + E_{BSC}(i) x\ [H_1^T]$
$SR_E(i) = M(i) + 0 + S_C(i)$
$SR_E(i) = M(i) + S_C(i) = \widehat{M}(i)$

2.3 The Second Model

The model shown in Fig. 6 implements an encoder with two stages of syndrome coding: the first stage with different BCH, BKLC, Golay, and Hamming codes, and the second stage with different BKLCs compatible with the output of the first stage encoder. (n_1, k_1, d_1) are the parameters of the binary linear codes in the first stage encoder, [G_1] is in the standard form of [$P_m|I_{k1}$], [H_1] is in the standard form of [$I_m|P^T_{k1}$], the input to the encoder is M of length ($m = n_1 - k_1$), and the output of this stage is C_1 of length n_1. (n_2, k_2, d_2) are the parameters of the BKLC codes used in the second stage, [G_2] is in the standard form of [$P_{n1}|I_{k2}$], [H_2] is in the standard form of [$I_{n1}|P^T_{k2}$], the input to this stage is C_1 of length n_1, the output of this stage is C_2 of length n_2.

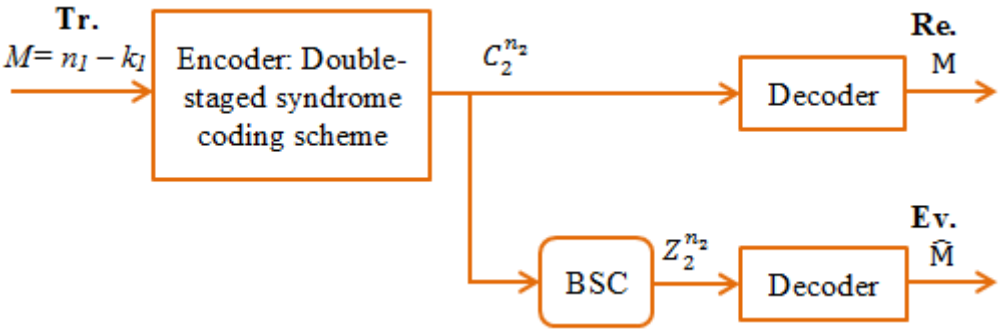

Figure 6. Double-staged syndrome coding scheme over the wiretap channel

The block diagrams and the algorithms for the transmitter's encoder, the legitimate receiver, and the eavesdropper's decoders are shown and explained as follows:

2.3.1 The transmitter's encoder

The transmitter starts encoding the message M of length m to generate the codeword C_2 of length n_2 to be transmitted over the channel to the legitimate party using two stages of the syndrome coding scheme. The first stage encoder uses binary linear BCH, BKLC, Hamming, and Golay codes with (n_1, k_1, d_1) parameters. The second stage encoder encodes the output of first stage C_1 of length n_1 using a syndrome coding scheme of appropriate BKLC codes with parameters (n_2, k_2, d_2) such that $n_1 = n_2 - k_2$. The block diagram for the transmitter's encoder is shown in **Fig. 7**. For encoding steps, the transmitter follows algorithm#4.

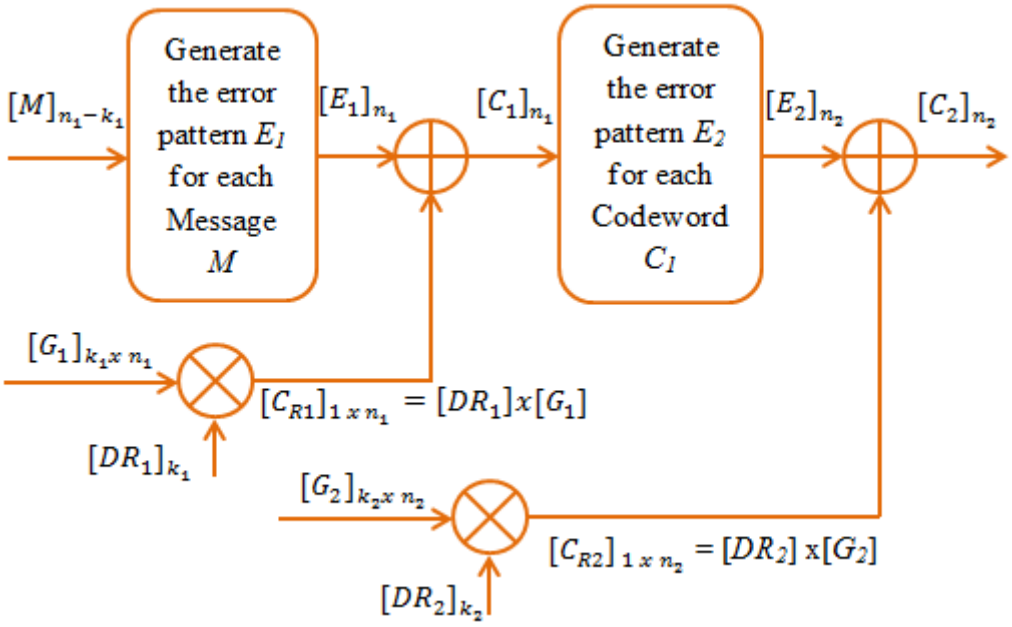

Figure 7: The transmitter's double-staged syndrome coding encoder

Algorithm#4: The encoding process for the binary message M by the transmitter.

Input: Generator matrix $[G_1]$, transpose of the parity-check matrix $[H_1^T]$, and the random data vector $[DR_1]$ is required for the first stage, and the Generator matrix $[G_2]$ and the random data vector $[DR_2]$ are required for the second stage to perform the following encoding steps:

[1] Obtaining $[G_1]_{k1\,x\,n1}$, $[H_1]_{(n1\text{-}k1)\,x\,n1}$, and $[G_2]_{k2\,x\,n2}$ from the Magma software suite and put them on the standard form
[2] Calculating the transpose of the parity-check matrix $[H_1^T]_{n1x(n1-k1)}$
[3] Generating $[DR_1]_{1\,x\,k1}$: (a random data vector of length k_1 will be generated)
[4] Calculating $[C_{R1}]_{1\,x\,n1}$

$$[C_{R1}]_{1\,x\,n1} = [DR_1]_{1\,x\,k1}x\,[G_1]_{k1\,x\,n1} \tag{10}$$

[5] Generating the error pattern $[E_1]_{1\,x\,n1}$ for each message $[M]_{1\,x\,(n1\,-\,k1)}$ from:

- Look-up table: for perfect codes, choose the error pattern that satisfies Eq. (11)

$$[M]_{1x(n1-k1)} = [E_1]_{1xn1}x\,[H_1^T] \tag{11}$$

- Padding the message vector by k_1-bits of zero: for non-perfect codes

$$[E_1]_{1\,x\,n1} = M_{(n1\,-\,k1)}|\,00..00_{k1} \tag{12}$$

[6] Calculate the output of the first stage $[C_1]_{1\,x\,n1}$

$$[C_1]_{1\,x\,n1} = [E_1]_{1\,x\,n1} \oplus [C_{R1}]_{1\,x\,n1} \quad (13)$$

[7] Generating $[DR_2]_{1\,x\,k2}$: (a random data vector of length k_2 will be generated)
[8] Calculating $[C_{R2}]_{1\,x\,n2}$

$$[C_{R2}]_{1\,x\,n2} = [DR_2]_{1\,x\,k2} x\, [G_2]_{k2\,x\,n2} \quad (15)$$

[9] Generating the error pattern $[E_2]_{1\,x\,n2}$ for each output of first stage $[C_1]_{1\,x\,n1}$ from padding the codeword vector by k_2-bits of zero

$$[E_2]_{1\,x\,n2} = [C_1]_{1\,x\,n1} | \, 00..00_{k2} \quad (16)$$

[10] Calculating the output of the second stage $[C_2]_{1\,x\,n2}$, which is the output of double stage encoder

$$[C_2]_{1\,x\,n2} = [E_2]_{1\,x\,n2} \oplus [C_{R2}]_{1\,x\,n2} \quad (17)$$

Output: Return $[C_2]_{1\,x\,n2}$ that is $C_2(i) = E_2(i) + DR_2(i).[G_2]$

2.3.2 The legitimate receiver's decoder

To recover the original encoded message by the transmitter, the legitimate receiver uses a double-stage decoder, as it is shown in **Fig. 8**.

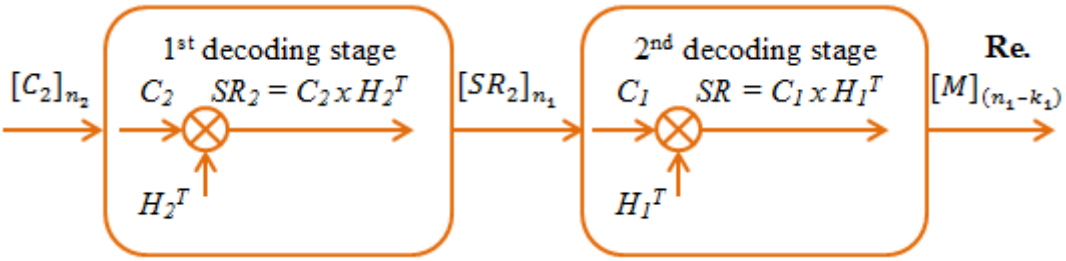

Figure 8: Double-staged legitimate receiver's decoder

The legitimate receiver receives the same transmitted codeword bits C_2 of length n_2, which is the output of the double-stage encoder, due to an error-free communication channel. The first decoding stage performs a parity-check multiplication over C_2 with the transpose of the parity-check matrix of the BKLC that is used for encoding in the second encoding stage (i.e. H_2^T). This stage of the decoder returns SR_2 of length n_1 equal to C_1. The second decoding stage performs a parity-check multiplication over C_1 with the transpose of the parity-check matrix of the binary linear code that is used for encoding in the first encoding stage (i.e. H_1^T). This stage of the decoder returns M of length $(m = n_1 - k_1)$. To perform this procedure, the legitimate receiver follows the steps explained in algorithm#5.

Algorithm#5: The legitimate receiver's double-staged decoding procedure to return M.

Input: The transpose of the parity-check matrices $[H_2^T]$, $[H_1^T]$, and the received codeword C_2 is required to perform the following decoding steps:

[1] Obtaining $[H_2]_{(n2-k2)\,x\,n2}$ and $[H_1]_{(n1-k1)\,x\,n1}$ from Magma then put them on the standard form $[I_{n2-k2} \mid P^T_{k2}]$ and $[I_m \mid P^T_{k1}]$ respectively
[2] Generating the transpose of the parity-check matrix $[H_2^T]_{n2\,x\,(n2-k2)}$ from $[H_2]_{(n2-k2)\,x\,n2}$
[3] Calculating the output of the decoder's first stage $[SR_2]_{1\,x\,n1}$

$$[SR_2]_{1\,x\,n1} = [C_2]_{1\,x\,n2}\ x\ [H_2^T]_{n2\,x\,(n2-k2)} \tag{18}$$

[4] Setting $[C_1]_{1x\,n1} = [SR_2]_{1\,x\,n1}$
[5] Generating the transpose of the parity-check matrix $[H_1^T]_{n1\,x\,(n1-k1)}$ from $[H_1]_{(n1-k1)\,x\,n1}$
[6] Calculating the output of the decoder's second stage $[SR]_{1\,x\,(n1-k1)}$

$$[SR]_{1\,x\,(n1-k1)} = [C1]_{1\,x\,n1}\ x\ [H_1^T]_{n1\,x\,(n1-k1)} \tag{19}$$

[7] Set $[M]_{1\,x\,(n1-k1)} = [SR]_{1\,x\,(n1-k1)}$

Output: Return $[M]_{1\,x\,(n1-k1)}$

From Eq. (10, 11, 12, 13, 15, 16, 17, 18, and 19), the legitimate receiver's recovery of the original message M can be proved as follows:

$$SR_2(i) = C_2(i)\ x\ [H_2^T]$$
$$C_2(i) = E_2(i) + C_{R2}(i) = C_1(i)_{n1}|00..0_{k_2} + DR_2(i).[G_2]$$
$$SR_2(i) = (C_1(i)_{n1}|00..0_{k_2} + DR_2(i).[G_2])\ x\ [H_2^T]$$
$$SR_2(i) = C_1(i)_{n1}|00..0_{k_2} x\ [H_2^T] + \underbrace{DR_2(i).[G_2]\ x\ [H_2^T]}_{0}$$
$$SR_2(i) = C_1(i)_{n1}$$
$$SR(i) = SR_2(i) x\ [H_1^T] = C_1(i)\ x\ [H_1^T]$$
$$C_1(i) = E_1(i) + C_{R1}(i) = E_1(i) + DR_1(i).[G_1]$$
$$SR(i) = (E_1(i) + DR_1(i).[G_1]) x\ [H_1^T]$$
$$SR(i) = E_1(i)\ x\ [H_1^T] + \underbrace{DR_1(i).[G_1] x\ [H_1^T]}_{0}$$
$$SR(i) = M(i)_{n1-k1}$$

2.3.3 The eavesdropper's decoder

The eavesdropper captures a corrupted version of the transmitted vector Z_2 of length n_2 due to obtaining information through a BSC. After that, he tries to obtain the original encoded message by implementing the same legitimate receiver's double-staged decoder. However, he cannot recover the original encoded message by the transmitter, as the BSC adds random error sequence E_{2BSC} to the transmitted codeword through the channel based on the crossover error probability of the channel. The randomness is added by the BSC as in Eq. (20). The block diagram for the eavesdropper's channel and decoder is shown in **Fig. 9**. To perform the decoding steps, the eavesdropper follows algorithm#6.

$$[Z_2]_{1\,x\,n2} = [C_2]_{1\,x\,n2} \oplus [E_{2BSC}]_{1\,x\,n2} \quad (20)$$

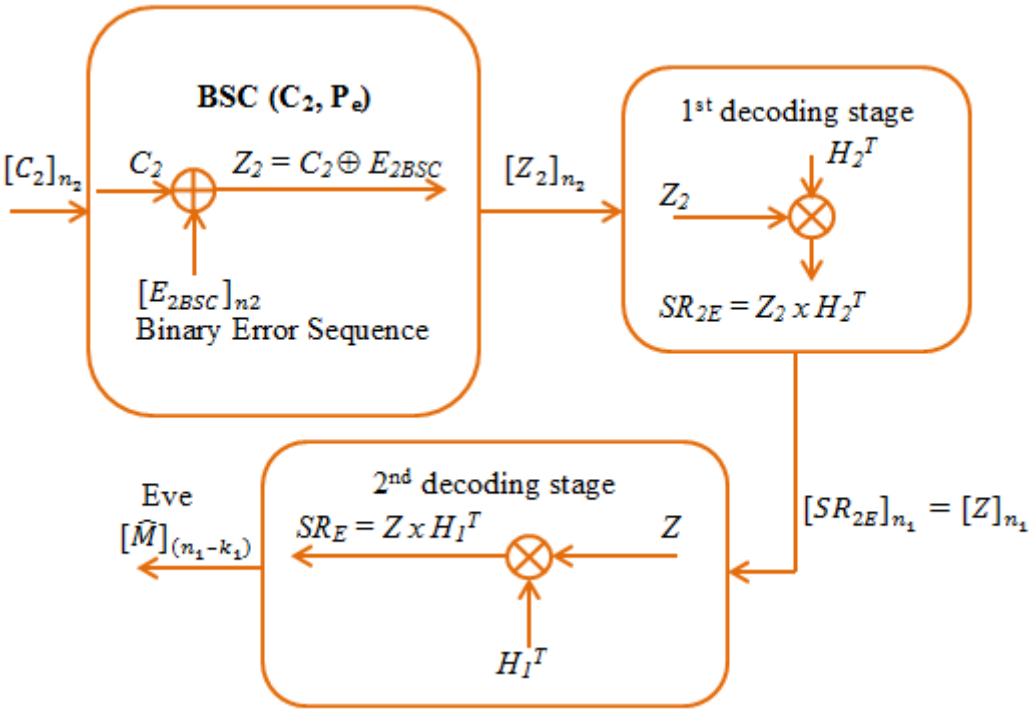

Figure 9: The eavesdropper's communication channel and double-staged decoder

Algorithm#6: Eavesdropper's attempt to obtain *M*.

Input: The transpose of the parity-check matrices $[H_2^T]$, $[H_1^T]$, and the corrupted codeword Z_2 is required to perform the following decoding steps:

[1] Obtaining $[H_2]_{(n2-k2)\,x\,n2}$ and $[H_1]_{(n1-k1)\,x\,n1}$ from Magma then put them on the standard form $[I_{n2-k2} \mid P^T_{k2}]$ and $[I_m \mid P^T_{k1}]$ respectively
[2] Generating the transpose of the parity-check matrix $[H_2^T]_{n2\,x\,(n2-k2)}$ from $[H_2]_{(n2-k2)\,x\,n2}$
[3] Calculating the output of the decoder's first stage $[SR_{2E}]_{1\,x\,n1}$

$$[SR_{2E}]_{1\,x\,n1} = [Z_2]_{1\,x\,n2}\ x\ [H_2^T]_{n2\,x\,(n2-k2)} \quad (21)$$

[4] Setting $[Z]_{1x\,n1} = [SR_{2E}]_{1\,x\,n1}$
[5] Generating the transpose of the parity-check matrix $[H_1^T]_{n1\,x\,(n1-k1)}$ from $[H_1]_{(n1-k1)\,x\,n1}$
[6] Calculating the output of the decoder's second stage $[SR_E]_{1\,x\,(n1-k1)}$

$$[SR_E]_{1\,x\,(n1-k1)} = [Z]_{1\,x\,n1}\ x\ [H_1^T]_{n1\,x\,(n1-k1)} \quad (22)$$

[7] Set $[\widehat{M}]_{1\,x\,(n1-k1)} = [SR_E]_{1\,x\,(n1-k1)}$
Output: Return $[\widehat{M}]_{1\,x\,(n1-k1)}$

From Eq. (10, 11, 12, 13, 15, 16, 17, 18, 19, 20, 21, and 22), the eavesdropper's failure to obtain the original message M can be proved as:

$SR2_E(i) = Z_2(i)\ x\ [H_2^T]$
$Z_2(i) = C_2(i) + E_{2BSC}(i) = E_2(i) + C_{R2}(i) + E_{2BSC}(i)$
$Z_2(i) = C_1(i)_{n1}|00..0_{k_2} + DR_2(i).[G_2] + E_{2BSC}(i)$
$SR2_E(i) = (C_1(i)_{n1}|00..0_{k_2} + DR_2(i).[G_2] + E_{2BSC}(i))\ x\ [H_2^T]$
$SR2_E(i) = C_1(i)_{n1}|00..0_{k_2} x\ [H_2^T] + \underbrace{DR_2(i).[G_2]x[H_2^T]}_{0} + E_{2BSC}(i)\ x\ [H_2^T]$
$SR2_E(i) = C_1(i)_{n1} + E_c(i)_{n1} = Z(i)$
$SR_E(i) = Z(i)x\ [H_1^T] = (C_1(i) + E_c(i)_{n1})x\ [H_1^T]$
$SR_E(i) = (E_1(i) + C_{R1}(i) + E_c(i)_{n1})x\ [H_1^T]$
$SR_E(i) = (E_1(i)_{n1} + DR_1(i).[G_1] + E_c(i)_{n1})x\ [H_1^T]$
$SR_E(i) = E_1(i)_{n1}x\ [H_1^T] + \underbrace{DR_1(i).[G_1]x\ [H_1^T]}_{0} + E_c(i)_{n1}x\ [H_1^T]$
$SR_E(i) = M(i)_{n1-k1} + S_{C2}(i)_{\ n1-k1}$
$SR_E(i) = M(i) + S_{C2}(i) = \widehat{M}(i)$, which is not the same as the original encoded message.

2.4 Calculations of Equivocation Rate, Channel Capacity, Equivocation Difference, Equivocation Gain, and Transmission Rate

The security of the syndrome coding scheme will be measured in terms of equivocation rate $H(M(i)|\widehat{M}(i))$ at the output of Eve's decoder. The equivocation can be calculated from:

$$H(M(i)|\ \widehat{M}(i)) = H(M(i), \widehat{M}(i)) - H(\widehat{M}(i)) \quad (23)$$

Where H (.) denotes the entropy, H (-, .) denotes the joint entropy, and H (- | .) denotes the conditional entropy.
Normalization to equivocation can be obtained after dividing the equivocation rate by the length of the original encoded message M.

$$Normalised\ Equivocation = \frac{H\left(M(i)\middle|\widehat{M}(i)\right)}{m} \quad (24)$$

Normalization puts the scale of the equivocation rate between 0 and 1, which is the mean of equivocation per bit of transmitted data. Therefore for different code parameters, normalization gives a meaningful comparison among them as the comparison is based on similar measurements.
The amount of information that Eve can obtain from Alice's transmitted codewords can be calculated from Eq. (25), and the maximum rate of obtained information is called the channel capacity.

$$I\left(M;\widehat{M}\right) = H\left(\widehat{M}\right) + H(M) - H\left(M,\widehat{M}\right) \quad (25)$$

The normalized equivocation difference between any two information transmission models (either secured or unsecured) can be calculated to show the difference between the two systems. Eq. (26) is used for the calculation of equivocation differences.

$$Normalised\ Equivocation\ Difference = Eq_{(system2)} - Eq_{(system1)} \quad (26)$$

The equivocation gain, which can be obtained by a modified model with better security than another model, can be calculated from Eq. (27).

$$Equivocation\ Gain = \frac{Equivocation_{modified\ model}}{Equivocation_{\ model\ before\ modification}} \quad (27)$$

The transmission rate of the syndrome coding scheme is calculated by dividing the length of the original encoded message ($m = n_1 - k_1$) over the length of the output of the decoder n (for the first model, $n = n_1$, and the second model $n = n_2$).

$$Transmission\ rate = \frac{m}{n} \quad (28)$$

3. RESULTS AND DISCUSSIONS

The normalized equivocation rates with different error probabilities of the BSC for the first and the second models are presented in **Tables 1, 2, 3, 4, 5,** and **6**. The normalized equivocation rates for the unsecured system are presented in **Table 7**. The BCH, BKLC, Golay, and Hamming binary linear code families have been tested for different code parameters as they are presented in the result tables. Each table shows results for two different binary codes when simulated on the first and second models. For all tables, the first column shows the error probability of the BSC, the second column is the normalized equivocation rates of the single-stage syndrome coding for the mentioned code in the column header, and the third column is the normalized equivocation rates of the double-stage syndrome coding using the code which is mentioned first for the first stage encoding and the code which is mentioned last for the second stage encoding, and the fourth and fifth columns are the same as second and third columns in definitions respectively.

The results obtained from the second model are compared with the results of the uncoded transmission for each code family. For all codes, when the BSC adds no errors to the transmitted codewords, the equivocation rate at the eavesdropper is zero, and Eve can recover the original encoded massage. In general, when the BSC adds random error sequences to the transmitted codewords with high crossover probability ($P_e > 0.2$), the equivocation rate at the eavesdropper side will be high, and the channel is inappropriate to obtain information. In other words, the equivocation rate by Eve will be high for any used system by the transmitter. Therefore, for comparing the obtained equivocation rates, we focus on the crossover probability of the BSC from (0.01 to 0.1) as the security performance of these codes appears in the low P_e of the BSC. The code that gives higher equivocation rates at low Pe performs better security.

It is very hard to compare the obtained results of the second model and present a code as the best in its family. The results show that the BCH code (31, 26, 3) with BKLC (144, 113, 9) has the best security performance in the BCH family. The BKLC (21, 15, 4) with BKLC (128, 107, 7) has the best security performance in the BKLC family. The Hamming code (15, 11, 3) with BKLC (130, 115, 5) has the best security performance in the Hamming family. The Golay code (23, 12, 7) with BKLC (100, 77, 8) has the best security performance in the Golay family. In addition, among these four code families, we represent the Hamming code (15, 11, 3) with

BKLC (130, 115, 5) as the best codes to transmit information for our proposed model as they give the highest equivocation rates.
The System with minimum information leakage is considered the best for security constraints. Besides the obtained security, the normalized information leakage can be calculated to Eve from the normalized equivocation results as (normalized information leakage = 1 – normalized equivocation rate) and presented on a graph. Moreover, the equivocation difference and the equivocation gain between our model and the unsecured system can be calculated.

Table 1. Normalized equivocation rate for BCH code family-Part 1

	Normalized equivocation results of the:			
P_e of BSC	1st model with BCH(31,26,3)	2nd model with BCH(31,26,3) BKLC(144,113,9)	1st model with BCH(31,21,5)	2nd model with BCH(31,21,5) BKLC(144,113,9)
0.01	0.43305	0.89886	0.24365	0.78649
0.02	0.65968	0.99091	0.41907	0.95704
0.03	0.79603	0.99897	0.55881	0.98775
0.04	0.87914	0.99954	0.67027	0.99200
0.05	0.92985	0.99955	0.75750	0.99245
0.06	0.96012	0.99955	0.82448	0.99247
0.07	0.97803	0.99955	0.87481	0.99247
0.08	0.98837	0.99955	0.91174	0.99249
0.09	0.99401	0.99955	0.93842	0.99250
0.10	0.99688	0.99955	0.95711	0.99250
0.20	0.99954	0.99955	0.99239	0.99250
0.30	0.99955	0.99958	0.99248	0.99250
0.40	0.99956	0.99958	0.99249	0.99250
0.50	0.99956	0.99958	0.99249	0.99250

Table 2. Normalized equivocation rate for BCH code family-Part 2

	Normalized equivocation results of the:

P_e of BSC	1st model with BCH (15,7,5)	2ndmodel with BCH (15,7,5) BKLC(130,115,5)	1st model with BCH (15,5,7)	2ndmodel with BCH (15,5,7) BKLC (130,115,5)
0.01	0.14991	0.79864	0.11994	0.74045
0.02	0.26188	0.96148	0.20995	0.93736
0.03	0.35766	0.99238	0.28792	0.98267
0.04	0.44201	0.99707	0.35791	0.99107
0.05	0.51699	0.99765	0.42197	0.99232
0.06	0.58361	0.99767	0.48103	0.99247
0.07	0.64278	0.99769	0.53562	0.99247
0.08	0.69500	0.99770	0.58610	0.99247
0.09	0.74110	0.99770	0.63269	0.99247
0.10	0.78139	0.99771	0.67555	0.99249
0.20	0.97198	0.99771	0.93093	0.99249
0.30	0.99675	0.99771	0.98851	0.99249
0.40	0.99766	0.99771	0.99247	0.99249
0.50	0.99770	0.99771	0.99249	0.99249

Table 3. Normalized equivocation rate for BKLC code family-Part 1

	Normalized equivocation results of the:					
P_e of BSC	1st model with BKLC(21,15,4)	2nd model with BKLC(21,15,4) BKLC(128,107,7)	1st model with BKLC(21,12,5)	2nd model with BKLC(21,12,5) BKLC(128,107,7)	1st model with BKLC(20,15,3)	2nd model with BKLC(20,15,3) BKLC(95,75,7)
0.01	0.27180	0.84304	0.18549	0.77111	0.30204	0.78015
0.02	0.45220	0.97651	0.32281	0.95073	0.49217	0.95020
0.03	0.58838	0.99667	0.43794	0.98846	0.62985	0.98988
0.04	0.69255	0.99902	0.53627	0.99490	0.73160	0.99790
0.05	0.77177	0.99923	0.62047	0.99576	0.80792	0.99931
0.06	0.83182	0.99923	0.69202	0.99586	0.86344	0.99954
0.07	0.87698	0.99923	0.75231	0.99586	0.90379	0.99956
0.08	0.91087	0.99923	0.80256	0.99586	0.93312	0.99956
0.09	0.93598	0.99926	0.84414	0.99586	0.95433	0.99956
0.10	0.95433	0.99926	0.87814	0.99586	0.96917	0.99956
0.20	0.99828	0.99926	0.99162	0.99586	0.99937	0.99956
0.30	0.99922	0.99926	0.99586	0.99587	0.99959	0.99956
0.40	0.99923	0.99929	0.99588	0.99589	0.99959	0.99957
0.50	0.99923	0.99929	0.99588	0.99589	0.99959	0.99957

Table 4. Normalized equivocation rate for BKLC code family-Part 2

	Normalized equivocation results of the:

P_e of BSC	1st model with BKLC (25,18,4)	2nd model with BKLC (25,18,4) BKLC (100,75,8)	1st model with BKLC (33,23,5)	2nd model with BKLC (33,23,5) BKLC (110,77,10)	1st model with BKLC (23,11,8)	2nd model with BKLC (23,11,8) BKLC (100,77,8)
0.01	0.27444	0.73195	0.25866	0.67829	0.15210	0.56454
0.02	0.45227	0.92970	0.44306	0.90283	0.26501	0.81691
0.03	0.58193	0.98275	0.58731	0.97048	0.36138	0.92055
0.04	0.67764	0.99554	0.69950	0.98790	0.44649	0.95733
0.05	0.74856	0.99816	0.78487	0.99167	0.52275	0.96863
0.06	0.80073	0.99859	0.84858	0.99236	0.59120	0.97162
0.07	0.83914	0.99868	0.89489	0.99246	0.65238	0.97233
0.08	0.86721	0.99869	0.92782	0.99249	0.70662	0.97245
0.09	0.88787	0.99870	0.95068	0.99249	0.75418	0.97245
0.10	0.90333	0.99871	0.96620	0.99249	0.79538	0.97246
0.20	0.95891	0.99871	0.99244	0.99249	0.96283	0.97248
0.30	0.98174	0.99871	0.99247	0.99249	0.97243	0.97248
0.40	0.99453	0.99871	0.99248	0.99249	0.97248	0.97248
0.50	0.99871	0.99873	0.99248	0.99249	0.97248	0.97248

Table 5. Normalized equivocation rate for Hamming code family

	Normalized equivocation results of the:					
P_e of BSC	1st model with Hamming (7,4,3)	2nd model with Hamming (7,4,3) BKLC (80,73,3)	1st model with Hamming (15,11,3)	2nd model with Hamming (15,11,3) BKLC (130,115,5)	1st model with Hamming (63,57,3)	2nd model with Hamming (63,57,3) BKLC (130,67,20)
0.01	0.17959	0.78746	0.28072	0.89543	0.63262	0.85579
0.02	0.30865	0.95049	0.45989	0.98938	0.85732	0.98009
0.03	0.41165	0.98934	0.59089	0.99897	0.94668	0.99726
0.04	0.49884	0.99784	0.69162	0.99971	0.98112	0.99907
0.05	0.57473	0.99953	0.76877	0.99973	0.99352	0.99924
0.06	0.63855	0.99977	0.82764	0.99973	0.99763	0.99926
0.07	0.69353	0.99983	0.87222	0.99973	0.99885	0.99926
0.08	0.74109	0.99983	0.90661	0.99973	0.99917	0.99926
0.09	0.78169	0.99989	0.93238	0.99973	0.99922	0.99926
0.10	0.81660	0.99989	0.95151	0.99973	0.99925	0.99926
0.20	0.97734	0.99989	0.99901	0.99973	0.99926	0.99926
0.30	0.99888	0.99989	0.99972	0.99979	0.99926	0.99926
0.40	0.99981	0.99989	0.99972	0.99979	0.99926	0.99926
0.50	0.99981	0.99989	0.99972	0.99979	0.99926	0.99926

Table 6. Normalized equivocation rate for Golay code family

	Normalized equivocation results of the:

P_e of BSC	1st model with Golay (23,12,7)	2nd model with Golay (23,12,7) BKLC (100,77,8)	1st model with Golay (24,12,8)	2nd model with Golay (24,12,8) BKLC (100,76,8)
0.01	0.16600	0.60235	0.15855	0.56297
0.02	0.28920	0.84965	0.27604	0.81552
0.03	0.39421	0.94324	0.37606	0.91980
0.04	0.48657	0.97422	0.46410	0.95701
0.05	0.56849	0.98315	0.54259	0.96853
0.06	0.64090	0.98535	0.61258	0.97162
0.07	0.70417	0.98583	0.67459	0.97229
0.08	0.75867	0.98590	0.72895	0.97243
0.09	0.80498	0.98593	0.77597	0.97246
0.10	0.84381	0.98593	0.81605	0.97246
0.20	0.98079	0.98593	0.96606	0.97248
0.30	0.98589	0.98594	0.97246	0.97248
0.40	0.98593	0.98595	0.97246	0.97248
0.50	0.98594	0.98595	0.97246	0.97249

Table 7. Normalized equivocation rates for transmission over an unsecured system

Pe of BSC	Normalized Equivocation
0.01	0.08017
0.02	0.14063
0.03	0.19325
0.04	0.24089
0.05	0.28503
0.06	0.32587
0.07	0.36402
0.09	0.40017
0.10	0.43433
0.20	0.46669
0.30	0.71768
0.40	0.87583
0.50	0.94499

To visualize the obtained results, as an example, **Fig. 10, 11, 12,** and **13** are drawn to show the normalized equivocation, information leakage, equivocation differences, and equivocation gain respectively for the insecure system, the first model for the Hamming code (15, 11, 3) and the second model for the Hamming code (15, 11, 3) with BKLC (130, 115, 5).

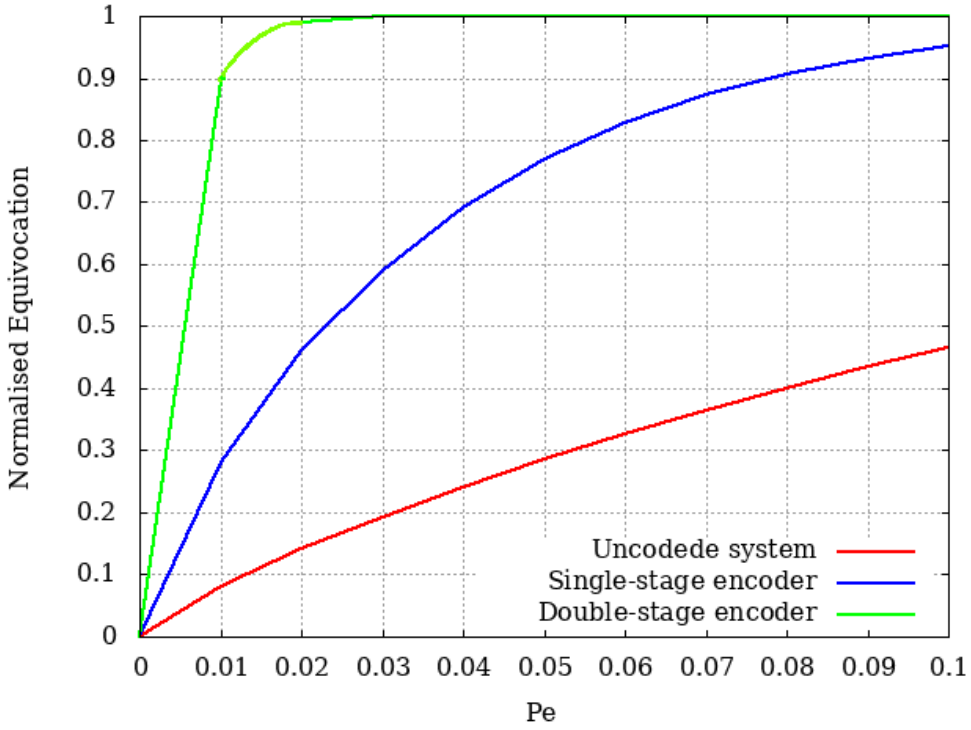

Figure 10. Normalized equivocation of unsecured system, the 1st model and the 2nd model

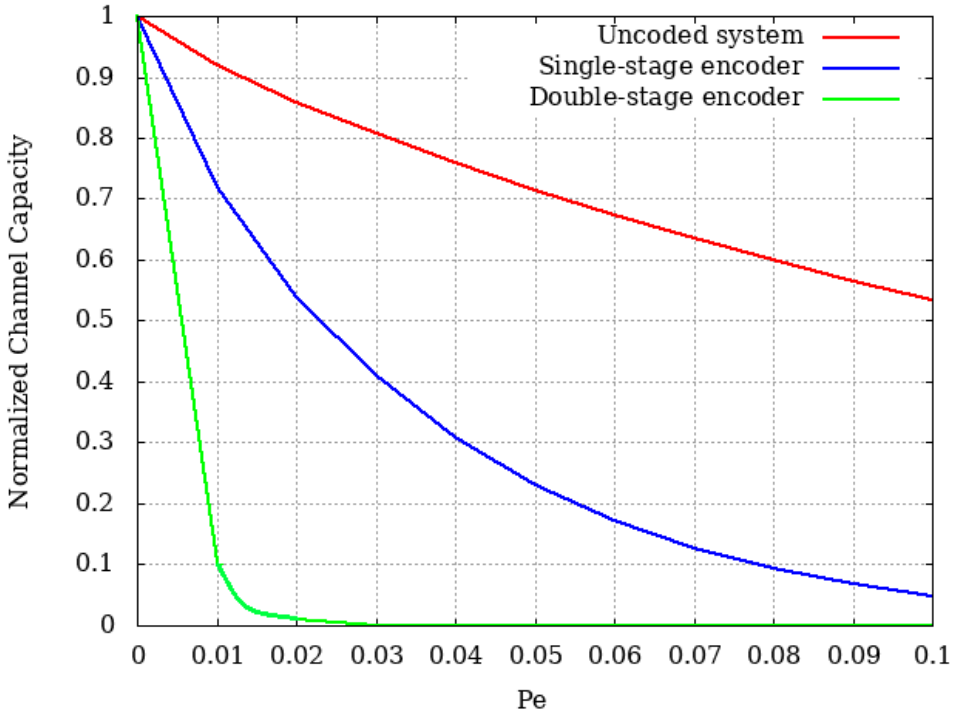

Figure 11. Information leakage to Eve through the insecure system, the 1st, and the 2nd model

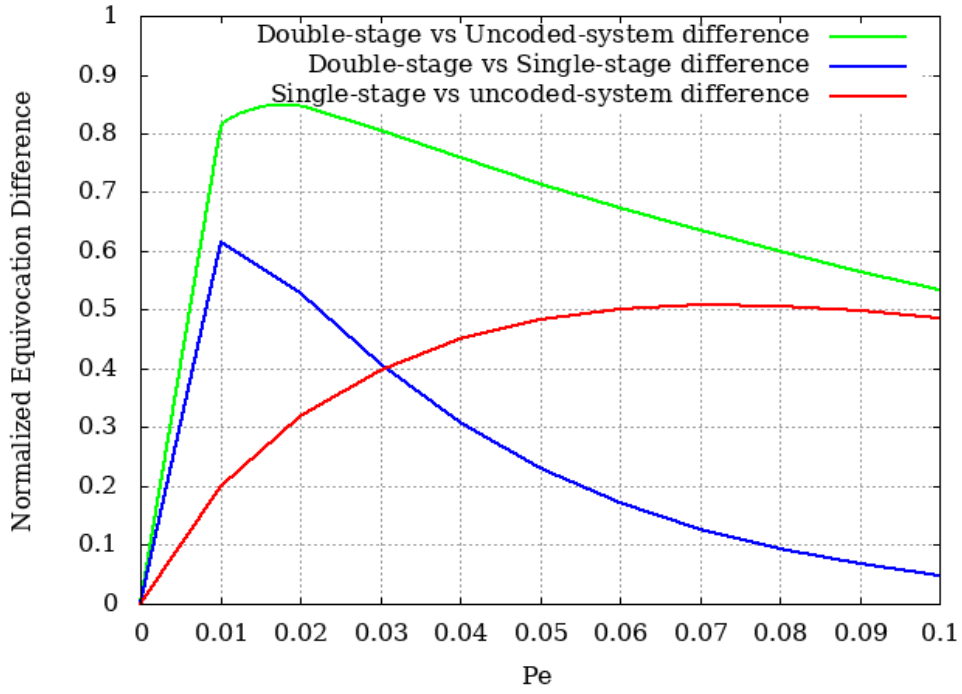

Figure 12. Normalized equivocation difference

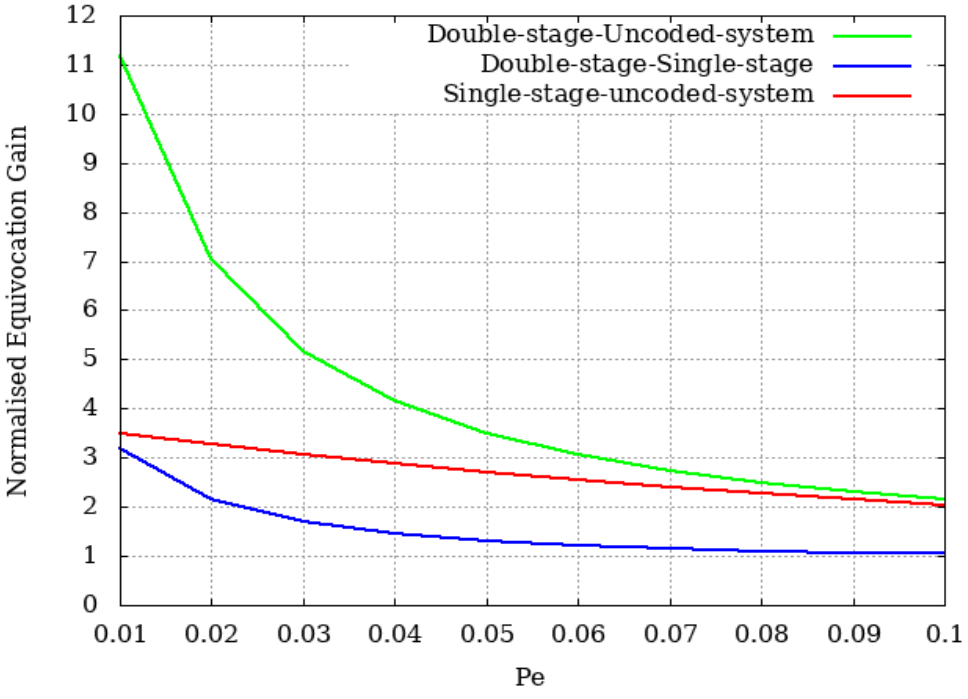

Figure 13. Normalized equivocation gain

4. CONCLUSIONS

The security constraints were investigated for different binary linear codes, such as BCH, BKLC, Golay, and Hamming, having different code parameters in a single-staged syndrome coding scheme over the wiretap channel for a special case of the error-free main channel and binary symmetric eavesdropper channel. The security of these codes is measured in terms of equivocation rates at the eavesdropper. The equivocation rate results of the utilized codes in the investigation stage showed that the security of error-correcting codes used in the syndrome coding scheme was inappropriate for secure information transmission.

To improve the security of the codes used in the single-staged syndrome coding system, a system of two stages of the syndrome coding scheme was proposed such that the output from the first encoding stage was used as the input to the second encoding stage using BKLC codes. The information leakage of the proposed system is reduced such that the eavesdropper obtains a vanishing bit of the original message at Pe = 0.1 as the equivocation for all the simulated codes over the proposed model reaches at least %97. The proposed system's obtained results showed a significant security gain in terms of equivocation rate for all simulated codes due to the use of long codes and additional complications in the second stage of the encoder.

In the future, further studies can be made by implementing different wiretap channel types such as Gaussian channel, Binary Erasure Channel, and Arbitrary Varying Channel. Finally, it is important to study different capabilities for the wiretapper in the systems that consider information transmission security as a real problem that faces wireless communication systems.

REFERENCES

Al-Hassan, S., Ahmed, M. Z., and Tomlinson, M., 2013. Secrecy coding for the wiretap channel using Best Known Linear Codes, *Global Information Infrastructure Symposium, IEEE,* pp. 1-6.

AL-Hassan, S., Ahmed, M. Z. and Tomlinson, M., 2014. Extension of the parity check matrix to construct the best equivocation codes for syndrome coding, *IEEE Global Information Infrastructure, and Networking Symposium,* pp. 1-3.

Chen, Y., He, D., Ying, C. and Luo, Y., 2022. Chen, Yiqi, Dan He, Chenhao Ying, and Yuan Luo. "Strong secrecy of arbitrarily varying wiretap channel with constraints, *IEEE Transactions on Information Theory,* 68(7), pp. 4700-4722.

Gazi, O., 2020. *Forward error correction via channel coding.* Switzerland: Springer.

Grassl, M., 2007. *Bounds on the minimum distance of linear codes and quantum codes.* [Online] Available at: http://www.codetables.de

Harrison, W. K. et al., 2019. Implications of coding layers on physical-layer security: A secrecy benefit approach, *Entropy,* 21(8), pp. 755.

Kadum, A. C., Flayyih, W. N. and Rokhani, F. Z., 2020. Reliability Analysis of Multibit Error Correcting Coding and Comparison to Hamming Product Code for On-Chip Interconnect, *Journal of Engineering,* 26(6), pp. 94-106.

Moon, T. K., 2021. *Error correction coding: mathematical methods and algorithms,* 2nd ed. New Jersey, USA: John Wiley & Sons.

Nooraiepour, A., Aghdam, S. R. and Duman, T. M., 2020. On secure communications over Gaussian wiretap channels via finite-length codes, *IEEE Communications Letters,* 24(9), pp. 1904-1908.

Ozarow, L. H. and Wyner, A. D., 1984. Wire-tap channel II. *AT&T Bell Laboratories technical journal,* 63(10), pp. 2135-2157.

Shannon, C. E., 1948. A mathematical theory of communication, *The Bell System Technical Journal,* 27(3), pp. 379-432.

Wyner, A. D., 1975. The wiretap channel, *The Bell System Technical Journal,* 54(8), pp. 1355-1387.

Zhang, K., Tomlinson, M., and Ahmed, M. Z., 2013. A modified McEliece public key encryption system with a higher security level, *IEEE Third International Conference on Information Science and Technology,* pp. 991-996.