Adaptive Linear Secrecy Codes with Feedback

ADAPTIVE LINEAR SECRECY CODES USING FEEDBACK

David Hunn
Department of Electrical and Computer Engineering
Brigham Young University
Provo, UT 84602
dhunn@byu.edu

Faculty Advisor:
Willie K. Harrison

ABSTRACT

We identify a novel method of using feedback to provide enhanced information-theoretical security in the presence of an eavesdropper. This method begins with a fixed linear coset code providing both secrecy and error detection/correction, as has been described by several authors. The legitimate receiver then sends the syndrome information for the received codeword, and based on this feedback, the transmitter can provide further error correction information specifically tailored to assist only the legitimate receiver. We show that this method allows secure communication with the legitimate receiver even when the eavesdropper's channel is superior to that of the legitimate receiver.

INTRODUCTION

Introduced in 1975 by Wyner, wiretap coding is the technique of encoding data for transmission with the goal of keeping the encoded data secure from an eavesdropper [1]. Although wiretap coding was conceived of and developed for many years as an alternative to encryption, recent work has focused on hybrid systems using both encryption and coding. These systems reduce the costs of encryption [2] while providing the benefits of secrecy coding such as information-theoretic security guarantees [3].

A. Coset Secrecy Codes

Among the first practical implementations of wiretap coding was a class of codes called *coset codes* [4]. In coset coding, a base linear code $\mathcal{C}$ is defined, and each of the possible messages m is assigned to one of the cosets of $\mathcal{C}$. Then to transmit a message, a codeword is selected at random from the appropriate coset. Coset codes have been designed and analyzed at both asymptotic [4, 5] and finite [6] blocklength.

The correct approach for providing both secrecy and reliability via coset codes was described in [4] and expanded upon in [7] and involves first selecting an error-correcting code $\mathcal{C}^*$, then selecting a subcode $\mathcal{C}$ of $\mathcal{C}^*$ as the base code, with all the cosets of $\mathcal{C}$ within $\mathcal{C}^*$ being mapped to

the different messages m. In this work, we use the notation and conventions detailed in [7]. To implement a combined secrecy and reliability code for blocklength n, message size k, and l bits of obfuscation, we first define a base code $\mathcal{C}$ with $l \times n$ generator matrix G. Next, we define a $k \times n$ auxiliary generator matrix G' such that the combined matrix G^* given by

$$G^* = \begin{bmatrix} G \\ G' \end{bmatrix} \tag{1}$$

has full row rank. The message m is then prefixed to an l-bit auxiliary message m'. (m' is chosen at random to confuse the eavesdropper and contains no useful information for the legitimate receiver). The resulting combined message $m^* = \begin{bmatrix} m & m' \end{bmatrix}$ is multiplied by G^* to obtain the n-bit codeword

$$x = \begin{bmatrix} m & m' \end{bmatrix} \begin{bmatrix} G' \\ G \end{bmatrix} = m^* G^*. \tag{2}$$

The combined generator matrix G^* has size $(k+l) \times n$, and therefore $n-k-l$ bits of overhead are used for error control. The parity check H^* of the code defined by G^* has size $(n-k-l) \times n$. An auxiliary parity check matrix H'' of size $l \times n$ may also be defined such that the combined matrix H is a parity check matrix for G and is given by

$$H = \begin{bmatrix} H* \\ H'' \end{bmatrix}. \tag{3}$$

B. Secrecy Via Feedback

Among the difficulties associated with wiretap coding is the dependency on the specifics of the legitimate receiver's channel and the eavesdropper's channel (factors which may change rapidly in telemetering applications). In particular, in the original formulation of wiretap coding, it was shown that security requires that the legitimate receiver must have a superior channel to that of the eavesdropper [1]. Subsequent works have attempted to overcome this difficulty by various methods, including the use of a feedback channel from the legitimate receiver to the transmitter. Recent advances in newtork-based telemetry architectures make the use of feedback a potential resource for security enhancements [8].

A natural approach for utilizing feedback is to allow a request for retransmission of certain bits needed by the legitimate receiver. This is particularly simple in, e.g., the case of the binary erasure channel (BEC), where the locations of the errors are known explicitly. In channels such as the additive white Gaussian noise channel (AWGN), it is also simple to select the bits that are less reliable. In channels such as these, the hybrid automatic repeat request (HARQ) methodology designed for data reliability may be employed to provide security. A number of authors have used various implementations of HARQ to allow for secure communication even when an eavesdropper's channel is superior to that of the legitimate receiver [9].

In channels such as the binary symmetric channel (BSC), the challenge of designing a HARQ protocol for security is more difficult, as there are no a priori indications of which bits are unreliable. In such cases, authors have resorted to the use of frame-based retransmission, where entire frames are retransmitted if required [10] or using LDPC codes with belief propagation decoding to

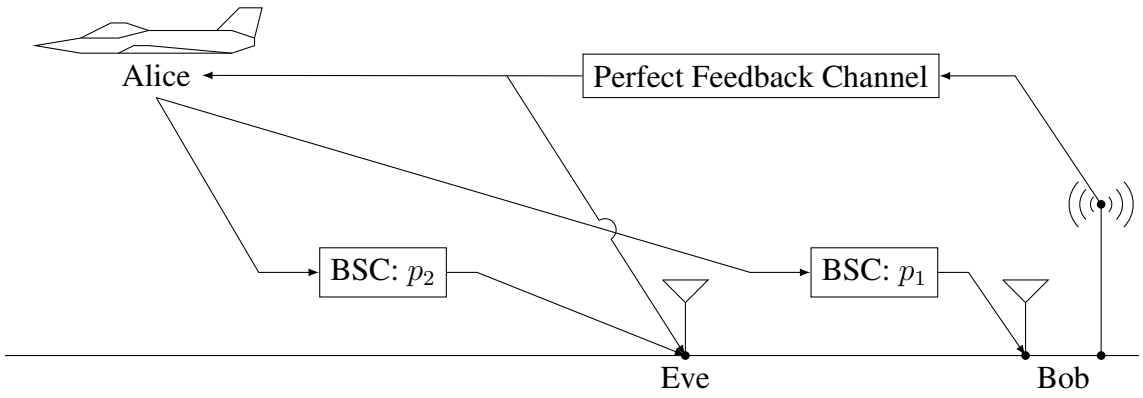


Figure 1: Wiretap Channel Model

identify unreliable bits [11]. In all of these cases, it is possible that the retransmitted bits or frames may give some useful information to the eavesdropper. This problem is mitigated, however, by the fact that the eavesdropper's unreliable data is often reliably received by the legitimate receiver. Thus the retransmitted data benefits the legitimate receiver more than the eavesdropper.

ADAPTIVE LINEAR SECRECY CODE CONCEPT

In this work, we introduce a method for secure communication over the wiretap channel illustrated in Figure 1. In this channel, the sender "Alice" sends a binary length-n codeword x to the legitimate receiver, "Bob". Bob receives the codeword through a binary symmetric channel (BSC) with transition probability p_1. Bob's received codeword y is then equal to x plus an error pattern e_{B}. (As usual, binary vector addition indicates the XOR operation.) An eavesdropper, "Eve", receives the codeword through a different BSC with transition probability p_2. Eve's codeword z is then given by $z = x + e_{\mathrm{E}}$, with e_{E} being Eve's error pattern. A perfect feedback channel is also available from Bob to Alice, and Eve also receives everything sent on the feedback channel. The approach we introduce here involves Alice using Bob's feedback to design supplementary transmissions tailored to preferentially aid Bob's error correction over that of Eve. Unlike the HARQ-based approaches described above, however, Alice does not simply retransmit part of the original codeword. Instead, her transmissions are linear functions of the original combined message m^*.

In particular, the approach outlined here involves three essential components. First, a standard linear code providing both secrecy and error correction is used to encode and transmit the message. Second, Bob transmits the syndrome $y^{\intercal}H^*$ of the received message y back to Alice. Third, Alice selects a supplementary message as a linear function of the original combined message and transmits it. To illustrate the benefits of this approach, consider the following simplified example of a length-8 code with a one-bit message and a three-bit auxiliary message. The initial code is defined by the generator matrix

$$G = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}, \tag{4}$$

Error Pattern	Original Codeword	Message	Auxiliary Message	Probability
1 1 0 0 0 0 0 0	**0 0 0 0 0 0 0 0**	**0**	**0 0 0**	$\mathbf{p_1^2(1-p_1)^6}$
1 1 1 1 1 0 0 1	0 0 1 1 1 0 0 1	0	0 0 1	$p_1^6(1-p_1)^2$
1 0 0 1 0 1 0 1	0 1 0 1 0 1 0 1	0	0 1 0	$p_1^4(1-p_1)^4$
1 0 1 0 1 1 0 0	0 1 1 0 1 1 0 0	0	0 1 1	$p_1^4(1-p_1)^4$
0 1 0 1 0 0 1 1	1 0 0 1 0 0 1 1	0	1 0 0	$p_1^4(1-p_1)^4$
0 1 1 0 1 0 1 0	1 0 1 0 1 0 1 0	0	1 0 1	$p_1^4(1-p_1)^4$
0 0 0 0 0 1 1 0	**1 1 0 0 0 1 1 0**	**0**	**1 1 0**	$\mathbf{p_1^2(1-p_1)^6}$
0 0 1 1 1 1 1 1	1 1 1 1 1 1 1 1	0	1 1 1	$p_1^6(1-p_1)^2$
0 1 0 0 1 1 0 1	1 0 0 0 1 1 0 1	1	0 0 0	$p_1^4(1-p_1)^4$
0 1 1 1 0 1 0 0	1 0 1 1 0 1 0 0	1	0 0 1	$p_1^4(1-p_1)^4$
0 0 0 1 1 0 0 0	**1 1 0 1 1 0 0 0**	**1**	**0 1 0**	$\mathbf{p_1^2(1-p_1)^6}$
0 0 1 0 0 0 0 1	**1 1 1 0 0 0 0 1**	**1**	**0 1 1**	$\mathbf{p_1^2(1-p_1)^6}$
1 1 0 1 1 1 1 0	0 0 0 1 1 1 1 0	1	1 0 0	$p_1^6(1-p_1)^2$
1 1 1 0 0 1 1 1	0 0 1 0 0 1 1 1	1	1 0 1	$p_1^6(1-p_1)^2$
1 0 0 0 1 0 1 1	0 1 0 0 1 0 1 1	1	1 1 0	$p_1^4(1-p_1)^4$
1 0 1 1 0 0 1 0	0 1 1 1 0 0 1 0	1	1 1 1	$p_1^4(1-p_1)^4$

Table 1: Bob's possible error patterns, most likely options shown in bold.

with auxiliary generator matrix

$$G' = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}. \tag{5}$$

The resulting G^* is self-dual, so the parity check matrix for the code is

$$H^* = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}, \tag{6}$$

Suppose Alice sends message $m = \begin{bmatrix} 0 \end{bmatrix}$ with auxiliary message $m' = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}$. Then the resulting codeword is the all-zero codeword. Now suppose Bob's received codeword has the first two bits flipped, so $y = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$. Eve then receives a codeword z with the first and third bits flipped, so $z = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$. Bob's possible error patterns, their likelihood, and their corresponding codewords and message values are shown in Table 1, with similar information for Eve shown in Table 2.

Bob and Eve both have four most-likely codewords and four most-likely values for m^*, with two each for a message value of zero and one. At this point, Bob and Eve both have a message equivocation of one bit, so neither has obtained any information on m. Furthermore, from Bob's perspective, all of the eight bits of y are equally likely to be in error. Thus if Bob requests retransmission of individual bits, the retransmitted bits are equally likely to help Eve as Bob. For example, suppose Bob requests retransmission of bits sequentially. Knowledge of the first bit will reveal to Bob that the message is likely $m = \begin{bmatrix} 0 \end{bmatrix}$, but it will reveal the same information to Eve.

Error Pattern	Original Codeword	Message	Auxiliary Message	Probability
1 0 1 0 0 0 0 0	**0 0 0 0 0 0 0 0**	**0**	**0 0 0**	$\mathbf{p_1^2(1-p_1)^6}$
1 0 0 1 1 0 0 1	0 0 1 1 1 0 0 1	0	0 0 1	$p_1^4(1-p_1)^4$
1 1 1 1 0 1 0 1	0 1 0 1 0 1 0 1	0	0 1 0	$p_1^6(1-p_1)^2$
1 1 0 0 1 1 0 0	0 1 1 0 1 1 0 0	0	0 1 1	$p_1^4(1-p_1)^4$
0 0 1 1 0 0 1 1	1 0 0 1 0 0 1 1	0	1 0 0	$p_1^4(1-p_1)^4$
0 0 0 0 1 0 1 0	**1 0 1 0 1 0 1 0**	**0**	**1 0 1**	$\mathbf{p_1^2(1-p_1)^6}$
0 1 1 0 0 1 1 0	1 1 0 0 0 1 1 0	0	1 1 0	$p_1^4(1-p_1)^4$
0 1 0 1 1 1 1 1	1 1 1 1 1 1 1 1	0	1 1 1	$p_1^6(1-p_1)^2$
0 0 1 0 1 1 0 1	1 0 0 0 1 1 0 1	1	0 0 0	$p_1^4(1-p_1)^4$
0 0 0 1 0 1 0 0	**1 0 1 1 0 1 0 0**	**1**	**0 0 1**	$\mathbf{p_1^2(1-p_1)^6}$
0 1 1 1 1 0 0 0	1 1 0 1 1 0 0 0	1	0 1 0	$p_1^4(1-p_1)^4$
0 1 0 0 0 0 0 1	**1 1 1 0 0 0 0 1**	**1**	**0 1 1**	$\mathbf{p_1^2(1-p_1)^6}$
1 0 1 1 1 1 1 0	0 0 0 1 1 1 1 0	1	1 0 0	$p_1^6(1-p_1)^2$
1 0 0 0 0 1 1 1	0 0 1 0 0 1 1 1	1	1 0 1	$p_1^4(1-p_1)^4$
1 1 1 0 1 0 1 1	0 1 0 0 1 0 1 1	1	1 1 0	$p_1^6(1-p_1)^2$
1 1 0 1 0 0 1 0	0 1 1 1 0 0 1 0	1	1 1 1	$p_1^4(1-p_1)^4$

Table 2: Eve's possible error patterns, most likely options shown in bold.

On the other hand, suppose Bob instead transmits his syndrome, $s = yH^{*\intercal} = \begin{bmatrix}1 & 1 & 1 & 0\end{bmatrix}$, to Alice. (Note that Bob's syndrome reveals no message information to Eve.) Bob's syndrome is sufficient to inform Alice of Bob's possible error patterns and their relative likelihood. Now Alice transmits a supplemental bit that is a linear combination of the bits of m^*. The option that provides the best information to Bob about m (without actually broadcasting m) is to send the sum of the first and second bits of m'. This has the overall effect of appending the column $\begin{bmatrix}0 & 1 & 1 & 0\end{bmatrix}^\intercal$ to the combined generator matrix G^*. This additional bit, if received correctly, allows Bob to rule out both of the most-likely codewords associated with $m = 1$. For Eve, this additional bit leaves two most-likely messages with different values for m. Thus, this arrangement provides more message-distinguishing help to Bob than to Eve. Note also that all the information needed to select the supplemental column for G^* is available to both Bob and Alice, so Alice does not need to communicate the specifics of the linear combination selected for the supplemental codeword bit.

IMPLEMENTATION METHODS

This section details the methods used to generate a proof of concept test of adaptive linear secrecy coding. A simple adaptive linear secrecy coding algorithm is implemented, along with a HARQ-based retransmission algorithm for comparison. The purpose of these tests is to show that adaptive linear secrecy codes can provide secrecy even when Eve's channel is superior to Bob's, and furthermore, that this advantage is greater than that provided by simple bit retransmission methods.

The adaptive linear secrecy coding algorithm uses a small initial $(k + l) \times n$ code defined by generator matrix $G^*_{(0)}$. The message m and auxiliary message m' are selected, and the combined codeword m^* is then encoded and sent to Bob and Eve through their respective channels. Several

stages of supplementary transmission are then performed. At each stage, a single column $g_{(i)}$ is appended to $G^*_{(i-1)}$, where i is the stage number. The supplementary codeword bit x_{n+i} is then calculated, transmitted, and received by Bob and Eve. Bob then calculates the supplemental syndrome bit s_{n-k+i} associated with the new bit and transmits it back to Alice.

The main point still to be specified in this implementation of adaptive linear secrecy coding is the method for selecting the columns $g_{(i)}$ to be added to the combined generator matrix for the supplementary transmissions. In this work, we simply use a brute force search of all possible columns at each stage of transmission to find the optimal column at that stage. The metric on which optimality is defined is simply the difference between Bob's equivocation and Eve's equivocation. The pseudocode for the entire implementation is given in Algorithm 1.

Algorithm 1: Adaptive Linear Secrecy Coding

Input: p_1, p_2, k, l //Channel and code properties
Input: m //Message to be sent
Input: $n_{\text{start}}, n_{\text{end}}$ //Initial and final codeword length
Input: $G^*_{(0)}, H^*_{(0)}$ //Initial combined generator matrix and parity check matrix
$G^* \leftarrow G^*_{(0)}$ //Augmented generator matrix for the adaptive linear secrecy code.
$m' \leftarrow$ Random l-bit string
$x \leftarrow m^* G^*$
Alice transmits x
Bob receives y
$s \leftarrow yH^{*\mathsf{T}}$ //Set syndrome to the initial syndrome.
for $i = 1$ **to** $n_{\text{end}} - n_{\text{start}}$ **do**
  $\beta \leftarrow -1$ //Best equivocation differential so far for g
  for $g \in \mathbb{F}_2^{k+l}$ **do** //Find best new column g to append to G^*.
    if $L([G^* \mid g], s, p_1) - L([G^* \mid g], [\,], p_2) > \beta$ **then**
      $b \leftarrow g$
      $\beta \leftarrow L([G^* \mid g], s, p_1) - L([G^* \mid g], [\,], p_2)$
    end
  end
  $G^* \leftarrow [G^* \mid b]$ //Update G^*.
  $H^* \leftarrow$ Parity Check Matrix of G^* //Update H^*.
  $x_i \leftarrow m^* b$
  Alice transmits x_i
  Bob receives y_i
  $y \leftarrow [y|y_i]$
  $s \leftarrow yH^{*\mathsf{T}}$
end

The retransmission algorithm starts with the same generator matrix $G^*_{(0)}$ used for the adaptive linear secrecy code. At each stage, Bob identifies the bit of his received codeword y which most reduces his own message entropy and requests retransmission of that bit. (Note that retransmitting a bit of the original codeword is equivalent to appending the corresponding column of the original generator matrix.) The pseudocode for the retransmission algorithm is given in Algorithm 2.

Algorithm 2: Retransmission-based Secrecy Method

Input: p_1, k, l //Channel and code properties
Input: m //Message to be sent
Input: $n_{\text{start}}, n_{\text{end}}$ //Initial and final codeword length
Input: $G^*_{(0)}, H^*_{(0)}$ //Initial combined generator matrix and parity check matrix
$G^* \leftarrow G^*_{(0)}$ //Augmented generator matrix for the adaptive linear secrecy code.
$m' \leftarrow$ Random l-bit string
$x \leftarrow m^* G^*$
Alice transmits x
Bob receives y
$s \leftarrow yH^{*\intercal}$ //Set syndrome to the initial syndrome.
for $i = 1$ **to** $n_{\text{end}} - n_{\text{start}}$ **do**
  $\alpha \leftarrow k$ //Lowest message entropy so far g
  for $\gamma \in$ columns of $G^*_{(0)}$ **do** //Find best column γ of $G^*_{(0)}$ to append to G^*.
    if $L([G^* \mid \gamma], s, p_1) < \alpha$ **then**
      $a \leftarrow \gamma$
      $\alpha \leftarrow L([G^* \mid \gamma], s, p_1)$
    end
  end
  $G^* \leftarrow [G^* \mid a]$ //Update G^*.
  $H^* \leftarrow$ Parity Check Matrix of G^* //Update H^*.
  $x_i \leftarrow m^* a$
  Alice transmits x_i
  Bob receives y_i
  $y \leftarrow [y|y_i]$
  $s \leftarrow yH^{*\intercal}$
end

In both Algorithm 1 and Algorithm 2, the function $L(G^*, s, p)$ returns the expected message equivocation resulting from transmitting m^* using G^* over a BSC with transition probability p, given that the received codeword has syndrome s. If the length of s is less than the length of G^*, then the expected equivocation across all possible syndromes that match s up to its length is returned.

A combined test algorithm is also implemented to perform a comparison of the adaptive linear secrecy code and the retransmission algorithm. This combined algorithm provides exact results averaged across all possible transmission error patterns for both Bob and Eve. To accomplish this, the algorithm first considers each possible initial syndrome, then it implements Algorithm 1 and 2 stage by stage, branching at each possible value of each supplemental syndrome bit. Each time the last stage is reached, the overall equivocation at each preceding stage is calculated and multiplied by the overall probability of the final combined syndrome. These stage by stage equivocation values are then accumulated, and once all possible syndromes have been considered, the equivocation totals are returned. The pseudocode for the overall combined test is given in Algorithm 3.

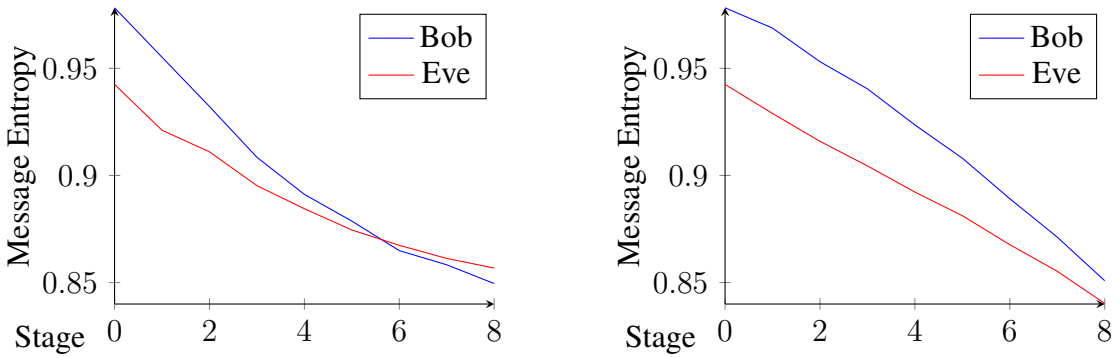
Algorithm 3: Combined Test Algorithm

Input: p_1, p_2, k, l
Input: $n_{\text{start}}, n_{\text{end}}$ //Initial and final codeword length
Input: $G^*_{(0)}$ //Initial combined generator matrix
$G^* \leftarrow G^*_{(0)}$ //Augmented generator matrix for the adaptive linear secrecy code.
$\Gamma^* \leftarrow G^*_{(0)}$ //Augmented generator matrix for the retransmission algorithm.
for $s_{(0)} \in \mathbb{F}_2^{n-k-l}$ **do** //For each possible initial syndrome
  $s \leftarrow s_{(0)}$ //Set syndrome to the initial syndrome.
  $i \leftarrow 0$ //Stage number.
  while $i \geq 0$ **do**
    $\lambda_i \leftarrow L(G^*, s, p_1)$ //Bob's message entropy for G^* with syndrome s.
    $\omega_i \leftarrow L(G^*, [\,], p_2)$ //Eve's message entropy for G^*.
    $\rho_i \leftarrow L(\Gamma^*, s, p_1)$ //Bob's message entropy for Γ^* with syndrome s.
    $\delta_i \leftarrow L(\Gamma^*, [\,], p_2)$ //Eve's message entropy for Γ^*.
    if $i > n_{\text{end}} - n_{\text{start}}$ **then** //If we have a completed codeword
      $\Lambda \leftarrow \Lambda + \Pr(s)\lambda$ //Record Bob's message entropy for G^*.
      $\Omega \leftarrow \Omega + \Pr(s)\omega$ //Record Eve's message entropy for G^*.
      $R \leftarrow R + \Pr(s)\rho$ //Record Bob's message entropy for Γ^*.
      $\Delta \leftarrow \Delta + \Pr(s)\delta$ //Record Eve's message entropy for Γ^*.
      while $s_{i+n_{\text{start}}-k-l} = 1$ **do** //Backtrack
        $i \leftarrow i - 1$
      end
      $s_{i+n_{\text{start}}-k-l} \leftarrow 1$
    else
      Find best new columns b and a for G^* and Γ^* using Algorithms 1 and 2.
      $G^* \leftarrow [G^* \mid b]$ //Update G^*.
      $\Gamma^* \leftarrow [\Gamma^* \mid a]$ //Update Γ^*.
      $s \leftarrow [s \mid 0]$ //Set new syndrome bit to zero initially.
      $i \leftarrow i + 1$ //Move to next stage.
    end
  end
end

RESULTS

Algorithm 3 was executed using parameters $p_1 = 0.25$, $p_2 = 0.2$, $k = 1$, $l = 4$, $n_{\text{start}} = 8$, $n_{\text{end}} = 16$, and initial combined generator matrix

$$G^*_{(0)} = \left[\begin{array}{cccccccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \end{array}\right]. \tag{7}$$



(a) Adaptive Linear Secrecy Code

(b) Retransmission Method

Figure 2: Performance of secrecy coding methods, calculated using Algorithm 3.

The resulting equivocation by stage for Bob and Eve are given for the adaptive linear secrecy code in Figure 2(a) and for the retransmission algorithm in Figure 2(b). The results shown in these figures indicate that the security gain provided by adaptive linear secrecy coding is superior to that of the retransmission-based secrecy method.

OPEN QUESTIONS AND FUTURE WORK

While the findings presented above show the potential of the adaptive linear secrecy coding, the results presented in this work are only an initial proof of concept for this new technique. There are many opportunities to optimize the algorithm presented here, especially in relation to practicality of implementation. A few of these possible optimizations are listed here:

- The current form of Algorithm 1 requires a brute force search of columns to extend G^*, a procedure which is not practical at higher code dimensions.
- The metric β used in Algorithm 1 provides no guarantees regarding Bob's successful decoding probability.
- After the initial transmission, Algorithm 1 transmits one bit at a time on the main and feedback channels. This likely presents latency problems for a practical implementation.

In addition to the optimizations listed above, many opportunities also exist to generalize the technique as presented here and to potentially gain even greater performance benefits over retransmission-based methods. Several of these are listed here:

- In addition to *extending* the code by adding columns to G^*, it may be valuable to *lengthen* the code by adding both columns and rows to G^* to further improve secrecy gain.
- The initial generator matrix $G^*_{(0)}$ could have all-zero rows corresponding to the bits of the message m. The transmission of any information about the message would then be put off until later stages of transmission, potentially further decreasing information leaked to Eve.
- The dependency of the new columns on Bob's feedback could be altered with the goal of making it resilient to errors in the feedback channel.

CONCLUSIONS

In this work, we introduce the technique of adaptive linear secrecy coding and present a demonstration of its effectiveness in principle. Specifically, we show that the overall performance (in terms of legitimate vs. eavesdropper equivocation) of adaptive linear secrecy coding exceeds that of retransmission-based methods such as HARQ when used for secrecy. The results presented here demonstrate the value of further development and study of adaptive linear secrecy coding.

REFERENCES

[1] A. D. Wyner, "The wire-tap channel," *The Bell System Technical Journal*, vol. 54, no. 8, pp. 1355–1387, 1975.

[2] A. Cohen, R. G. L. D'Oliveira, K. R. Duffy, and M. Médard, "Partial encryption after encoding for security and reliability in data systems," in *Proc. IEEE International Symposium on Information Theory (ISIT)*, pp. 1779–1784, 2022.

[3] M. R. Bloch, M. Hayashi, and A. Thangaraj, "Error-control coding for physical-layer secrecy," *Proceedings of IEEE*, vol. 103, pp. 1725–1746, Oct. 2015.

[4] A. Thangaraj, S. Dihidar, A. R. Calderbank, S. W. McLaughlin, and J.-M. Merolla, "Applications of LDPC codes to the wiretap channel," *IEEE Transactions on Information Theory*, vol. 53, no. 8, pp. 2933–2945, 2007.

[5] H. Mahdavifar and A. Vardy, "Achieving the secrecy capacity of wiretap channels using polar codes," *IEEE Transactions on Information Theory*, vol. 57, no. 10, pp. 6428–6443, 2011.

[6] J. Pfister, M. A. C. Gomes, J. P. Vilela, and W. K. Harrison, "Quantifying equivocation for finite blocklength wiretap codes," in *Proc. IEEE International Conference on Communications (ICC)*, pp. 1–6, 2017.

[7] M. Shoushtari and W. K. Harrison, "New dual relationships for error-correcting wiretap codes," in *Proc. IEEE Information Theory Workshop (ITW)*, pp. 1–6, 2021.

[8] M. A. Carvalho, N. P. O. Leite, and R. d'Amore, "iPCM telemetry protocol: Reliability and bandwidth improvement for pcm irig-106," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 56, no. 1, pp. 155–166, 2020.

[9] A. Roongta and J. Shea, "Reliability-based hybrid ARQ using convolutional codes," in *Proc. IEEE International Conference on Communications*, vol. 4, pp. 2889–2893 vol.4, 2003.

[10] Y. Omar, M. Youssef, and H. E. Gamal, "ARQ secrecy: From theory to practice," in *Proc. IEEE Information Theory Workshop*, pp. 6–10, 2009.

[11] L. Wang and D. Guo, "Secure communication based on reliability-based hybrid ARQ and LDPC codes," in *Proc. Prognostics and Health Management Conference (PHM-Besançon)*, pp. 304–308, 2020.