• U  C •

FCTUC FACULDADE DE CIÊNCIAS
E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

DEPARTAMENTO DE
ENGENHARIA
ELETROTÉCNICA E DE
COMPUTADORES

Esquemas de Codificação para Segurança Adaptativa na Camada Física

Dissertação apresentada para a obtenção do grau de Mestre em Engenharia Eletrotécnica e de Computadores na Especialidade Telecomunicações

Miguel Santarém Duque Carreira

Orientadores

Doutor Marco Alexandre Cravo Gomes
Doutor João Paulo da Silva Machado Garcia Vilela

Júri

Presidente: Doutor Luís Alberto da Silva Cruz

Orientador: Doutor Marco Alexandre Cravo Gomes

Vogal: Doutor Pedro António Amado Assunção

Setembro de 2018

Agradecimentos

Começo por agradecer aos professores responsáveis que me acompanharam desde o primeiro dia, ao professor Marco Gomes e ao professor João Vilela, pela orientação e oportunidade por eles dada que foi um contributo relevante para um crescimento pessoal e intelectual. É importante referir também o professor Willie Harrison, que com todo o seu conhecimento foi um enorme contributo para o desenrolar do trabalho.

Ao Instituto de Telecomunicações que proporcionou todo o material e meios para a execução do mesmo e também a todos os meus colegas do mesmo que proporcionaram sempre carácter profissional em ambiente descontraído.

Ao Departamento de Engenharia Eletrotécnica e de Computadores, e em particular ao Instituto de Telecomunicações, onde foi possível todo o desenrolar de um percurso académico até esta etapa, bem como a todos os docentes presentes no mesmo.

A toda a minha família, que além de acreditarem em mim, me deram estas oportunidades e tudo o mais, que me permitiu chegar a esta etapa.

Por fim, e não menos importante, a todos os meus amigos que fazem parte da minha vida social e profissional,

Muito Obrigado.

Resumo

O forte crescimento observado na última década, da presença das comunicações sem fios no dia-a-dia das pessoas, trouxe consigo a necessidade de assegurar e melhorar a confidencialidade nas transmissões de dados. Esta segurança é normalmente garantida nas camadas superiores da pilha protocolar do sistema de transmissão, pelo uso de métodos criptográficos. Mas, procurando soluções alternativas, a comunidade científica tem vindo a apostar cada vez mais no desenvolvimento de técnicas de segurança na camada física como complemento à criptografia.

O desafio deste trabalho foi essencialmente uma proposta de codificação para a segurança adaptativa na camada física, tendo por base as técnicas de ICSHK (ou *Interleaved Coding for Secrecy with Hidden Key*) e SCSHK (ou *Scrambling Coding for Secrecy with Hidden Key*). Foi também realizada uma comparação entre as referidas técnicas, no âmbito da qual foi proposta uma alteração ao esquema SCSHK ao nível do algoritmo de *Scrambling*, e sua dependência da chave de baralhamento. Por último, por forma a realizar uma avaliação prática fidedigna de segurança na camada física aquando o uso de códigos lineares, foi proposta a redefinição da métrica Intervalo de Segurança (ou *Security Gap*).

Com estes resultados, podemos conferir que ambos os esquemas propostos são capazes de garantir uma transmissão segura e fidedigna, proporcionando assim adaptatividade perante diversos pontos de operação tanto para o recetor legítimo como para o adversário, demonstrando mais uma vez que podem vir a ser um fator importante no complemento da segurança das comunicações.

Palavras-chave

Segurança Adaptativa na Camada Física; Blocos de Comprimento Curto; Fiabilidade; Baralhamento; Perfuração.

Abstract

With the increasing use of wireless communications mostly in the last decade, came a growing concern for obtaining confidentiality in the data transmission. This security is provided by top security layers of protocol stack in transmission systems, using cryptographic methods. Although, in a seek of finding new solutions, scientific community focus some experiments on physical layer in a way to complement cryptography.

The core of the work consists in the propose of coding scheme for adaptive physical layer security, more specifically based on ICSHK (or *Interleaved Coding for Secrecy with Hidden Key*) and SCSHK (or *Scrambling Coding for Secrecy with Hidden Key*) methods. In this work, a comparation is made between the methods mentioned before, based on which it is proposed a different implementation of the SCSHK scheme, and its shuffling key self-dependence. In a way to implement a reliable evaluation on physical layer security using short length codes, a new redefinition of the metric Security Gap was proposed.

In conclusion, we can confirm that both of proposed methods are capable to provide a transmission secure and with fidelity, proportionating in this way the adaptive characteristic in various operation points on both receptors, demonstrating once again they can become an important factor for adding security to communications.

Keywords

Adaptive Physical Layer Security; Short Block-Length Coding; Reliability; *Interleaving*; *Scrambling*; Puncturing.

Índice

ÍNDICE DE FIGURAS

LISTA DE ACRÓNIMOS

ARQ	Pedido de Informação Automático (*Automatic Repeat Request*)
AWGN	Ruído Branco Aditivo Gaussiano (*Additive White Gaussian Noise)*
BCH	Código Bose-Chaudhuri-Hocquenguem
BER	Taxa de Erro de Bit (*Bit Error Rate*)
BPSK	*Binary Phase-Shift Keying*
BSC	Canal Binário Simétrico (*Binary Symmetric Channel*)
CDF	Função de distribuição Cumulativa (*Cumulative Distribution Function*)
DMC	Canal Discreto sem Memória (*Discrete Memoryless Channel*)
ECC	Código Corretor de Erros (*Error Correcting Code*)
PMF	Função Massa de Probabilidade (*Probability Mass Function*)
ICS	*Interleaved Coding for Secrecy*
LDPC	*Low-Density Parity-Check Code*
SCS	*Scrambled Coding for Secrecy*
SNR	Relação Sinal-Ruído (*Signal-to-Noise Ratio*)

1. INTRODUÇÃO

Com o rápido desenvolvimento das redes sem fios, e a massificação da Internet, colocando ao dispor dos utilizadores o mais variado tipo de serviços (multimédia, serviços bancários, saúde online, redes sociais, etc…) novos mecanismos de segurança tiveram que ser desenvolvidas a fim de fornecer comodidade e confidencialidade aos utilizadores. Estes mecanismos, são fulcrais para que haja uma comunicação exclusiva entre remetentes e destinatários, sem qualquer tipo de envolvimento de terceiros, também designados *Eavesdroppers*. Tradicionalmente a segurança das comunicações é garantida nas camadas superiores da pilha protocolar [22] (segundo o modelo OSI) por meio de técnicas criptográficas cada vez mais sofisticadas sendo assim necessário recorrer a sistemas com maior capacidade de processamento. Mais recentemente, começaram a ser desenvolvidas técnicas de segurança na camada física [23], que tiram partido da natureza aleatória e errônea do canal de comunicação, providenciando uma camada adicional de segurança de menor complexidade em complemento às técnicas de criptografia. Por outro lado, o aparecimento da Internet das Coisas, suportado em dispositivos de baixa complexidade e capacidade de processamento limitada contribuí ainda mais para o aumento da informação a circular autonomamente, fomentando assim um interesse acrescido por técnicas práticas de segurança na camada física.

Com vista a encontrar soluções a este problema, várias alternativas não convencionais foram impostas a fim de proporcionar mais barreiras, sendo uma delas o estudo da camada física como complemento à criptografia. Recorrendo ao estudo de Wyner [2] [11], onde se destacou o modelo de canal *Wiretap Channel*, que consiste numa comunicação entre dois utilizadores, e um terceiro ilegítimo que vai auferir de alguma desvantagem em relação ao recetor legítimo. Neste trabalho, Wyner provou a existência de códigos de canal que providenciam fiabilidade para o recetor legítimo, bem como confidencialidade contra o adversário. No entanto, a descoberta de tais códigos revelou-se difícil, tendo apenas recentemente havido alguns avanços nesse sentido [11], mas de aplicabilidade limitada a modelos de canais idealizados. Nesse sentido, surge a necessidade de mecanismos de codificação práticos, aplicáveis a modelos de canais mais realistas [7] [9]. Trabalhos do mesmo tema realizados no mesmo tema [11] [18], sobre a camada física, bem como a técnicas de embaralhamento de mensagens vão proceder ao estudo e analise da informação transmitida entre um transmissor (Alice), um recetor (Bob) e um espião (Eve).

Com a vantagem necessária do ponto de vista de Bob em relação a Eve, veio um interesse acrescido no desenvolvimento de técnicas de segurança na camada física, tornando-se assim adaptativas, ou seja, tendo por base a condição de canal de Bob, é permitido o ajuste da região de operação sobre a qual é possível garantir segurança contra Eve. No entanto, estas técnicas tipicamente assumem que o recetor legítimo e o espião se encontram a operar num ponto de específico de relação sinal-ruído (SNR). Num cenário real, estas técnicas terão que ser adaptativas, por forma a se ajustarem aos diversos níveis de SNR que os dispositivos podem experimentar no decurso das comunicações. Esse é então, o objetivo deste trabalho: desenvolvimento de técnicas de segurança na camada física, adaptativas, ou seja, tendo por base a condição de canal de Bob, é permitido o ajuste da região de operação sobre a qual é possível garantir segurança contra o *Eavesdropper*.

De entre as diversas técnicas práticas de segurança na camada física [17] [18], apenas duas vão ser consideradas para este trabalho, sendo estas: ICSHK e SCSHK – *Interleaved/Scrambling Coding for Secrecy with Hidden Key*. Estas técnicas consistem na aplicação de mecanismos de baralhamento (*interleaving* ou *Scrambling*) às mensagens, antes de estas serem enviadas pelo canal. Este baralhamento é feito com recurso a uma chave que é gerada aleatoriamente para cada mensagem, sendo essa chave depois codificada em conjunto com a mensagem para posterior envio pelo canal. Antes do envio da informação, a chave é removida (perfuração). Desta forma, a informação sobre a chave encontra-se apenas armazenada nos bits de redundância do código de canal utilizado. A vantagem do recetor legítimo Bob sobre o espião Eve vai permitir ao primeiro obter a chave original e desembaralhar a mensagem, enquanto que Eve terá dificuldades nesse processo, devido ao canal degradado que possui.

1.1. **Objectivos**

Esta dissertação teve, como principal objetivo o estudo comparativo e proposta de alteração às técnicas ICSHK e SCSHK por forma a poder fornecer segurança adaptativa na camada física. A ideia base consiste no estudo do desempenho destas técnicas face à alteração do número de bits perfurados e na determinação do melhor padrão de perfuração por forma a garantir adaptabilidade dos esquemas às condições de operação do recetor legítimo e do espião.

Foi também avaliado o uso de códigos de comprimento curto e menor complexidade, por forma a avaliar a aplicação destes esquemas a dispositivos mais limitados, conforme é comum na, Internet das Coisas.

1.2. **Contribuições**

Atendendo ao objetivo da dissertação e ao trabalho desenvolvido para os alcançar, as contribuições aqui feitas podem-se resumir nos seguintes pontos:

- Proposta de técnicas de segurança na camada física adaptativa baseadas em métodos ICSHK e SCSHK por alteração de número de bits e padrão de perfuração;
- Desenvolvimento de simuladores de demonstração em Matlab para teste das referidas técnicas;
- Estudo da influência do processo de perfuração incidente na chave *versus* mensagem;
- Proposta de alteração à técnica SCSHK com vista à melhoria do seu desempenho, fazendo corresponder a chave de baralhamento da mensagem aos coeficientes do polinómio de *Scrambling*, por oposição à anterior proposta em que a mesma definia o estado inicial do baralhador;
- Proposta de uma adaptação à métrica denominada Intervalo de Segurança, baseada não apenas no BER (ou Taxa de Erro de Bit), mas também na função de distribuição cumulativa da probabilidade de erro por mensagem transmitida.
- Análise da crítica dos resultados de simulação obtidos e proposta para definição de adaptativa com vista à região válida para uma comunicação segura, em termos da relação Sinal/Ruido (SNR).

Este trabalho visa ser um contributo para o avanço no estudo de técnicas de segurança adaptativa na camada física, pois com os resultados obtidos demonstra-se que se pode conseguir segurança contra a Eve (espião) num cenário real, mantendo a fiabilidade da transmissão para o Bob, permitindo o posicionamento de um dado intervalo de segurança (definido pelo código corretor escolhido) ao longo de uma dada gama de valores de SNR.

1.3. **Estrutura da dissertação**

Este documento está estruturado em 5 capítulos. A esta Introdução, segue-se o Capítulo 2 que introduz conceitos base de segurança na camada física e transmissão fiável de dados. Este capítulo vai ser o mais longo pois é aqui que está assente toda a informação necessária para os futuros capítulos. Aqui, estão expostos os modelos, bem como as análises destes na presença de outras métricas e padrões que perfuração de bits. Vai ser ainda detalhado como é que a mensagem é transmitida/recebida em todas as etapas do canal, os modos de operação, os esquemas e progressivas transformações na mensagem por etapas.

É ainda estudado nos sistemas o uso de outros códigos curtos, de diferentes proporções ou de outra espécie.

O capítulo 3 vai ser exclusivamente dedicado às métricas. Por isto mesmo vai ser curto e essencial, pois é a partir deste onde vai ser possível analisar os dados obtidos e gerar dados estatísticos para as conclusões finais. Vai ser ainda explicado o conceito de *Security Gap*, que vai ser fulcral para outras conclusões do trabalho.

De seguida, no capítulo 4, é explicado como funciona a segurança adaptativa e como é que esta varia conforme a estratégia aplicada na variação da perfuração dos bits. É aqui que vão ser introduzidos os resultados obtidos, são tiradas conclusões e é feita uma confrontação com os resultados esperados.

Finalmente no capítulo 5 são apresentadas as conclusões retiradas fruto da análise de dados e apontadas algumas direções para o desenvolvimento futuro do trabalho.

2. CODIFICAÇÃO PARA SEGURANÇA E FIABILIDADE NA CAMADA FÍSICA

As bases de segurança na camada física foram lançadas por Aaron Wyner em 1975, que propôs um modelo de canal grampeado (*Wiretap Channel*) [2] apresentado na figura 1, que pressupõe a existência de um canal perfeito entre o transmissor e o recetor legítimos (também designados por Alice e Bob), e de um canal degradado entre Alice e um recetor ilegítimo, também designado por Eve (ou, *Eavesdropper*). Wyner provou ser possível [2] [7] o desenho de códigos capazes de garantir:

$$I(\tilde{Z}; M) = 0 \tag{1}$$

i.e., informação mútua nula entre a mensagem M enviada por Alice e a palavra $\tilde{Z}$ recebida por Eve, garantindo assim segurança perfeita.

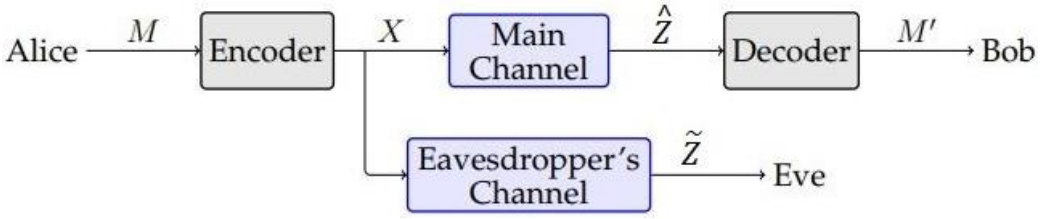


Figura 1: Canal grampeado ou *Wiretap Channel.*

Apesar de ter sido provada a existência de códigos *Wiretap* capazes de garantir (1), a sua aplicação em contexto prático em canais reais é limitada. Nesta dissertação, consideramos a generalização do modelo de canal *Wiretap* gaussiano, ilustrado na figura 2, com características mais próximas dos canais reais em que se assume que ambos os canais entre Alice → Bob e Alice → Eve são gaussianos.

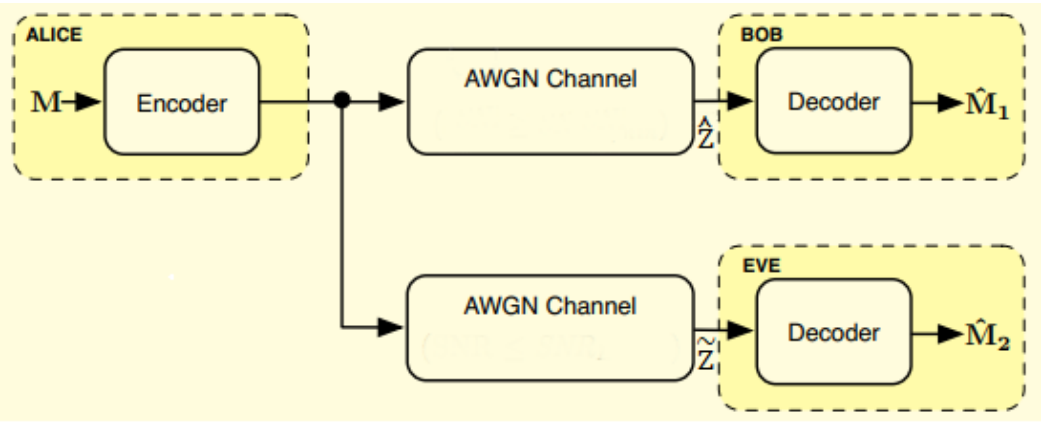


Figura 2: O canal de escuta (The Wiretap Channel). [2]

Neste cenário, assume-se ainda que o canal de Eve é mais ruidoso que o canal principal e que Eve é passivo, ou seja, que o Eve está à escuta sem se fazer notar, nem se fazer passar por outro interveniente. Considera-se ainda, que ambos os recetores (Bob e Eve) usam o mesmo algoritmo de descodificação e possuem idênticas capacidades de processamento. Apesar disso, é sabido que Eve sofre de um canal degradado (exemplo: este poder se encontrar por detrás de uma parede/obstáculo) sendo esta desvantagem explorada com vista ao desenvolvimento de técnicas de codificação capazes de providenciar fiabilidade na comunicação Alice → Bob e segurança/fiabilidade com respeito a Eve.

O desenho de técnicas de codificação capazes de providenciar simultaneamente fiabilidade e confidencialidade em contexto prático, revelou-se, pois, um importante desafio de investigação com alguns trabalhos a serem publicados recentemente [11] [17] [18] [27]. Muitos dos trabalhos propostos propõem, no entanto técnicas de codificação apenas aplicáveis para canais *Wiretap* discretos e sem memória, e em que requere ruído quase inexistente no canal de Bob.

Entre as diversas técnicas propostas iremos focar o nosso estudo nos métodos ICSHK e SCSHK [17] [18] [24] no contexto de desenvolvimento de técnicas de segurança adaptativas e capazes de providenciar simultaneamente fiabilidade com respeito a Bob e segurança em relação à Eve.

2.1. **ICSHK**

Apesar dos obstáculos na transmissão de informação, os erros introduzidos por um canal ruidoso conseguem, em geral, ser bem detetados e corrigidos pelos códigos corretores de erros existentes (LDPC, códigos turbo, códigos convolucionais, etc. [1]), desde que aqueles sejam esporádicos e independentes. No entanto, quando estes mesmos erros tendem em surgir por forma de rajadas, o que é comum em canais multipercurso com desvanecimento, tal resulta numa falha de resposta por parte do código, i.e., descodificação catastrófica. Neste contexto é comum o uso de Interleaving [18], i.e., baralhamento dos bits, por forma a espalhar esses erros facilitando o processo de descodificação. No entanto tal processo do baralhamento/ desembaralhamento pressupões o uso de uma chave previamente conhecida pelas partes, o que na ausência de conhecimento da chave leva ao aumento da confusão. É este o princípio base na génese da técnica ICSHK (*Interleaved Coding for*

Secrecy with Hidden Key) desenvolvido com vista a providenciar simultaneamente fiabilidade para o Bob e segurança contra o Eve, e que se encontra representado na figura 3.

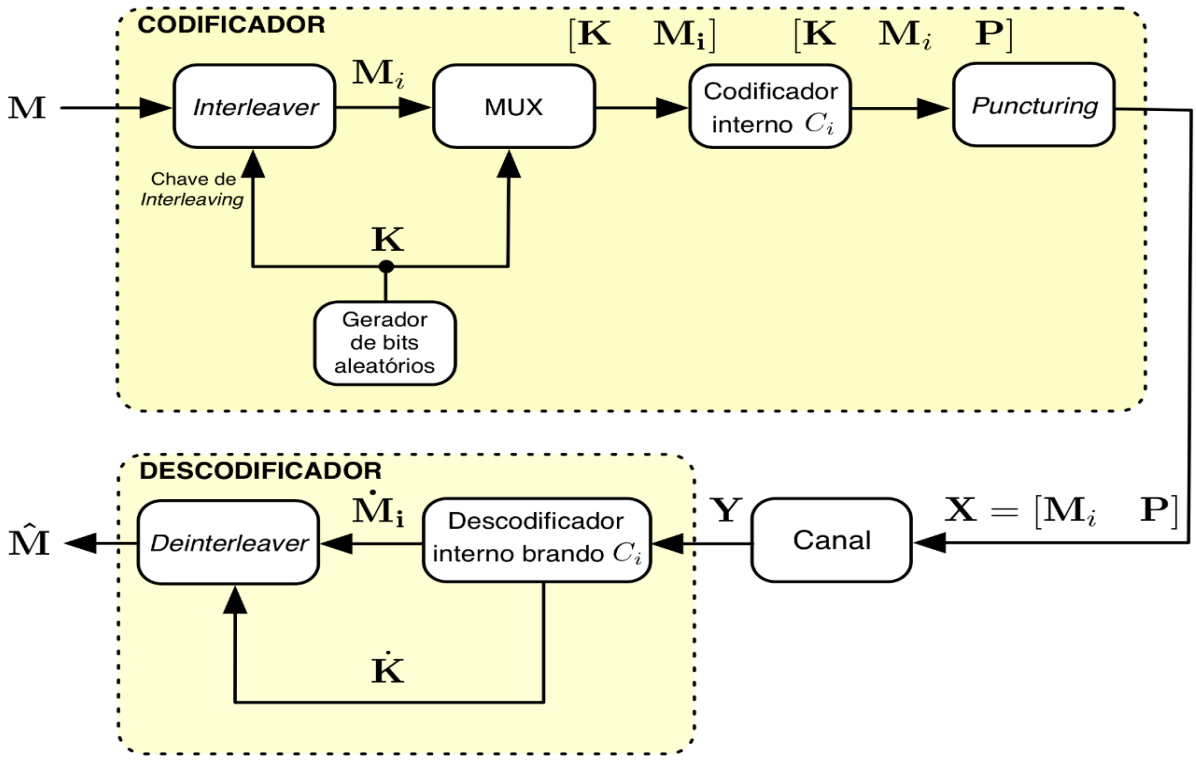


Figura 3: Modelo de Interleaved Coding for Secrecy with a Hidden Key. [17]

Relativamente ao seu funcionamento, neste modelo, é gerada uma chave aleatória K^k, que é usada no baralhamento (*Interleaving*) da mensagem a ser transmitida, M, resultando na mensagem embaralhada M_i. A chave de *Interleaving* é de seguida concatenada com M_i e codifica-se $[K\ M_i]$ com o código C_i (código interno) sistemático, e tamanho $(n,\ k+m)$. De seguida, antes de enviar esta sequência para o canal, são perfurados (Puncturing) os bits correspondentes aos bits de chave, sendo apenas enviados os bits que não foram perfurados e os bits de paridade introduzidos pelo código interno C_i,. Assim, a chave de ambos os recetores (Bob e Eve) é escondida, encontrando-se a informação sobre a mesma embebida nos bits de paridade [17] [18].

Após a passagem pelo canal, já no lado do Descodificador, procede-se à descodificação por ordem inversa inicialmente faz-se apenas uma descodificação com C_i, obtendo uma estimativa da mensagem com *Interleaving*, $\dot{M}_i$, e da chave, $\dot{K}$, que é, então, usada no desembaralhamento de $\dot{M}_i$ obtendo-se por fim uma estimativa da mensagem transmitida, $\widehat{M}$. Para boas condições de relação Sinal-Ruído (SNR) é expectável a

recuperação sem erros de M_i e K, e logo da mensagem M desembaralhada. Já para piores SNR, a existência de erros sobre K levará ao desembaralhamento errado de M_i e ao aumento da confusão, i.e., um aumento de segurança face a um Eavesdropper nessas condições.

2.1.1. Definição de *Interleaving*

Como já referido, com o intuito de corrigir o problema das rajadas de erros, foram desenvolvidas técnicas específicas que visam solucionar em parte deste problema, transformando rajadas de erros num conjunto de erros independentes dispersos ou em rajadas mais curtas.

O *Interleaving* ou baralhamento, é uma das referidas técnicas [11] que consiste numa reordenação, no transmissor após a etapa de codificação de canal (para correção de erros), de bits singulares ou conjuntos destes, dentro de uma mesma palavra de código ou entre diversas palavras de código. Por outro lado, na parte do recetor, é aplicado a técnica de *deinterleaving*, i.e., desembaralhamento que vem espalhar os erros presentes nas rajadas de erros resultando assim em vários erros individuais e/ou rajadas de menor comprimento. Para facilitar a compreensão desta técnica, é usada como exemplo o seguinte código de baralhamento de comprimento 4, presente na figura 3.

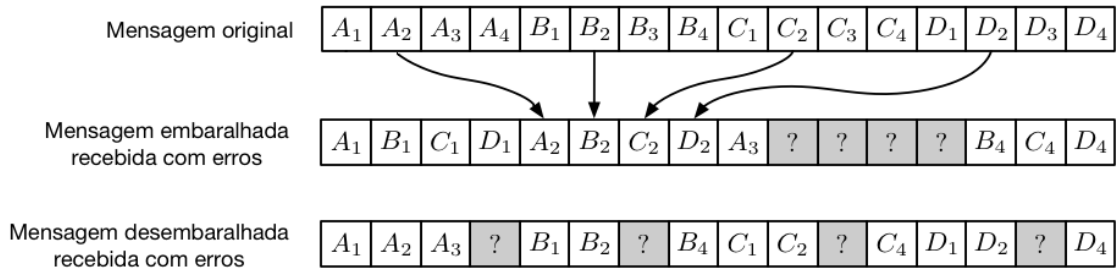


Figura 4: Representação do espalhamento de erros (representados pelos blocos a cinza) através de *Interleaving*.

No contexto da técnica ICSHK, o método de *Interleaving* desempenha um papel diferente a nível de segurança de transmissão, que consiste em baralhar cada mensagem que vai ser transmitida com uma diferente chave de *Interleaving*, gerada aleatoriamente.A função do *Interleaving*, não é neste caso espalhar os erros para aumentar a fiabilidade da comunicação, mas o oposto, o de aumentar o nível de confusão na presença de erros sobre a chave de baralhamento. Existem vários tipos de implementação para o *Interleaving* como: baralhamento de blocos, convolucional ou aleatório [13]. Como já referido no método

ICSHK é usado *interleaving* aleatório, devido à necessidade de maximizar o número de chave de baralhamento possível, para um dado comprimento de chave, obrigando no entanto ao tabelamento do alfabeto de chaves usadas no transmissor e no recetor.

2.2. **SCSHK**

Outra técnica que por vezes pode ser confundida com a anterior é o *Scrambling* [3], visto que para ambas a tradução para português é "baralhamento" o que pode gerar alguma confusão. No entanto, teoricamente, as técnicas são diferentes, podendo a técnica de *Scrambling* ser descrita como um baralhamento com combinação linear. O Scrambling é muitas vezes usado a fim de resolver eventuais problemas de sincronização no recetor com a função de eliminar sequências longas de bits iguais, aumentando também a densidade de transição dos mesmos. Também neste caso, o Scrambling pode ser usado com o objetivo oposto de aumentar a confusão para efeitos de segurança, tendo sido proposto uma variante ao esquema ICSHK, com substituição do bloco de Interleaver por um Scrambler. O novo esquema representado na figura 5 foi designado por SCSHK (*Scrambling Coding for Secrecy with Hidden Key)* [19]

A proposta de uso de scrambler em substituição da técnica de Interleaving surgiu, com a intenção de contornar algumas fragilidades do modelo de *Interleaving*, como por exemplo o fato de aquando da obtenção pelo Eve da chave K sem erros. Os erros obtidos na mensagem desembaralhada $\widehat{M}$ resultam apenas do reposicionamento dos erros existentes em M_i não havendo, pois, um mecanismo de propagação destes erros.

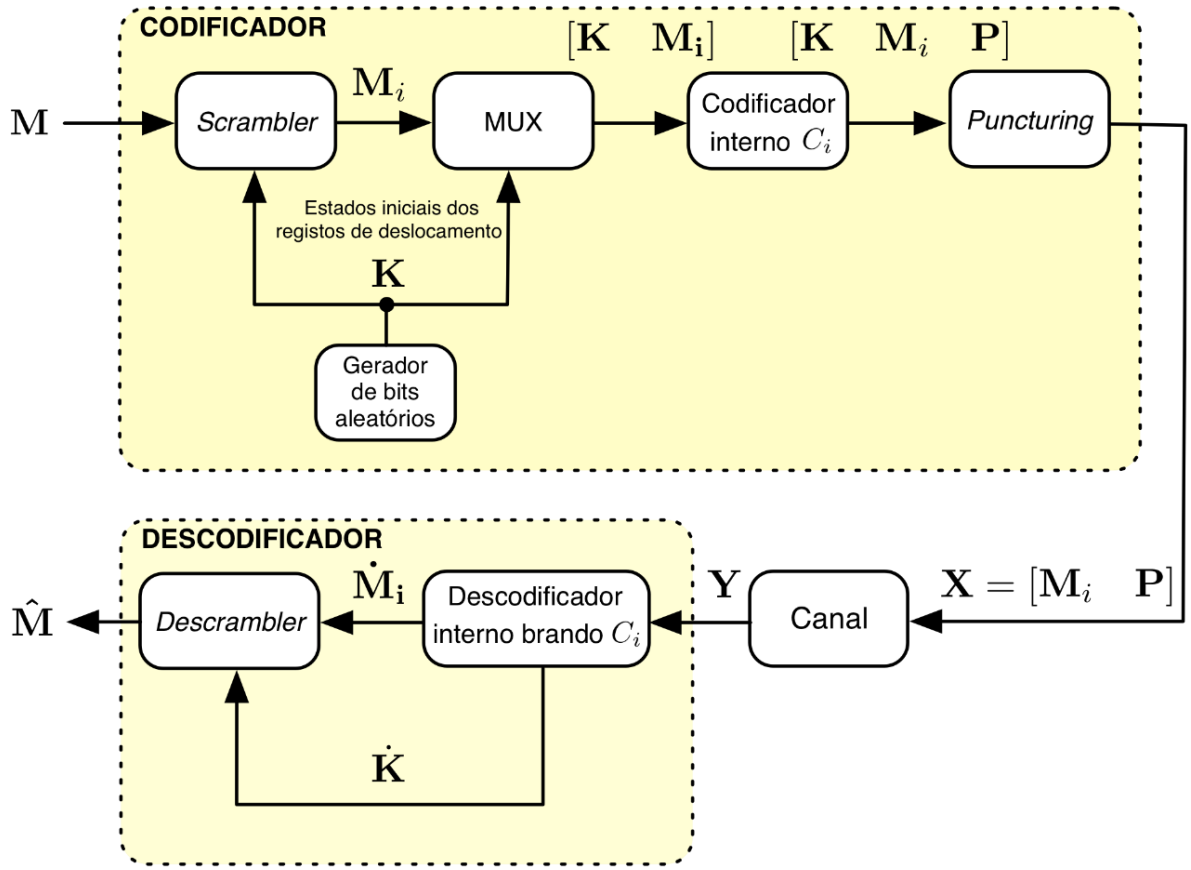


Figura 5: Modelo de *Scrambled Coding for Secrecy with a Hidden Key*.[17]

No entanto SCSHK funciona exatamente nos mesmos moldes da técnica ICSHK, descrita na secção 2.1, com a função de Interleaving/Deinterleaving, a ser substituída pela função de Scrambling/Descrambling, pelo que a descrição detalhada da figura 5 é aqui omitida.

2.2.1. Definição de *Scrambling*

Os *Scramblers* são construídos com registos de deslocamento com realimentação linear, como apresentado na Figura 6. Nesta, observa-se um *Scrambler* genérico, com *M* registos, definido por um polinômio $[1\ p_1\ p_2\ \ldots\ p_{M-1}\ p_M]$, que indica a posição dos interruptores que afetam a soma (mod 2) colocada à saída, e com estados iniciais $[k_1\ k_2\ \ldots\ k_{M-1}\ k_M]$, i.e., onde a saída binária do scrambler é:

$$b_n^{(s)} = \sum_{i=0}^{M} \oplus \; p_i b_{n-i} \tag{2}$$

, em que $\sum\oplus$ representa uma soma módulo 2 (i.e., XOR lógico) e em que b_n e $b_n^{(s)}$ representam respetivamente os bits à entrada e saída do scrambler.

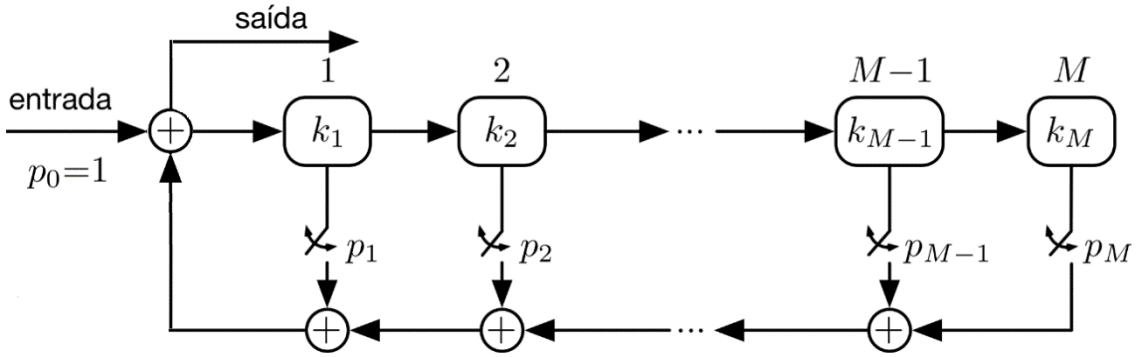


Figura 6: *Scrambler* com M registos.

Como podemos observar nesta figura, num *Scrambler*, a cada iteração os valores dos registos são deslocados para o registo seguinte, sendo o último descartado. Por outro lado, no lado do recetor, o *deScrambler* anula as imposições realizadas pelo *Scrambler*, ou seja, a saída num dado instante b_n é o resultado da diferença entre o símbolo recebido $b_n^{(s)}$ e a combinação linear dos *M* valores recebidos anteriormente, conforme representado na Figura 7., i.e. Em que a sequência binária desembaralhada é:

$$b_n = b_n^{(s)} - \sum_{i=1}^{M} \oplus \; p_i b_{n-i}^{(s)} \tag{3}$$

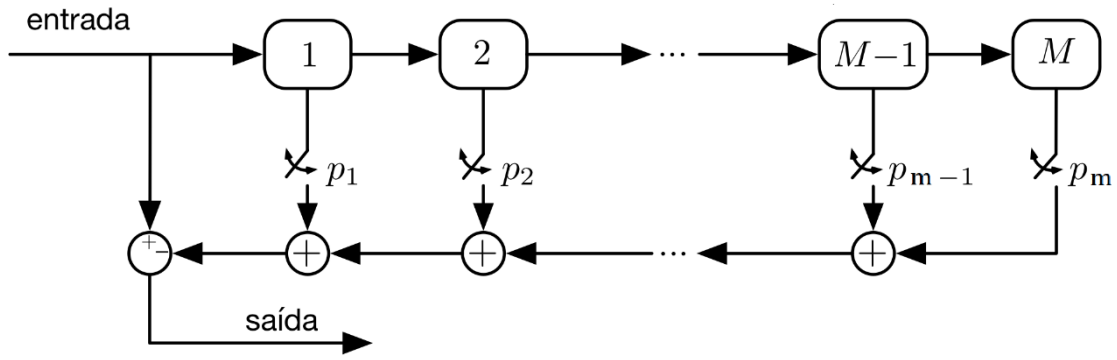


Figura 7: De*Scrambler* com M registos.

No contexto de segurança [9] [19], e em particular no esquema SCS-HK o *Scrambler* é usado para efeitos de propagação de erros na presença de erros ocorridos durante transmissões. A finalidade é que para uma transmissão livre de erros (por via de uma descodificação com sucesso do código interno), a recuperação da mensagem seja feita com sucesso; por outro lado, para uma transmissão com ruído considerável, ou seja, com mais erros do que o esperado (em que o código interno é incapaz de recuperar todos os erros de transmissão). Assim pretende-se que o *Scrambler* potencie o número de erros não corrigidos, uma vez que cada bit à saída do *deScrambler* vai depender do número dos M bits anteriores.

. No entanto, a proposta inicial da técnica SCSHK [24], a chave de Scrambling gerada aleatoriamente para cada mensagem a transmitir, era usada como estado inicial [K_1, … ,K_M] do scrambler/descrambler (figura 6 e 7). No entanto facilmente se conclui que por observação do circuito descrambler, que na ausência de bits errados sobre a mensagem baralhada recebida $\dot{M}_\iota$ e de uma chave com erros $\dot{K}$, é possível recuperar corretamente os bits de mensagem com exceção dos primeiros $\widehat{M}$ bits.

Assim, durante o desenvolvimento deste trabalho, foi proposto uma alteração ao esquema de *SCSHK*, em que a chave gerada aleatoriamente é usada como polinómio [p_1, …, p_n] de Scrambling potenciando efetivamente a propagação de erros aquando da receção de um bit errado sobre a chave $\dot{K}$.

2.3. **Códigos lineares sistemáticos**

Os esquemas de codificação ICSHK e SCSHK pretendem providenciar simultaneamente fiabilidade e segurança. A fiabilidade depende em grande medida das capacidades de correção do código de canal usado no codificador interno.

A codificação de canal é tida como um processo em que redundâncias são introduzidas antes da transmissão, com o objetivo de permitir que, no recetor, a semelhança entre o sinal que foi transmitido e o sinal que foi reproduzido seja a máxima possível.

No caso das técnicas ICSHK e SCSHK impõe como restrição o uso de códigos de bloco linear sistemáticos, i.e., em que há uma segmentação no vetor código sendo eles: um segmento composto pelos $(n-k)$ bits de redundância e outro segmento correspondente aos k bits da mensagem que gerou os bits de redundância. O comprimento do código é assim n e o rácio do código (code rate) é $\frac{k}{m}$. Com vista a C_i poder recuperar os erros produzidos pela perfuração, torna-se importante o uso de códigos com elevada capacidade de correção. Entre estes destacam-se os códigos LDPC (Low Density Check Codes) [7] caracterizados por matrizes de teste de paridade esparsas e o uso de algoritmos de descodificação iterativos do tipo Belief Propagation. No contexto deste caso foi feito uso de um código (n, k) de comprimento maior LDPC (1536,1280) da norma Wimax [15] e outro de menor comprimento LDPC (256,128) para satélite [25], tendo sido usado o algoritmo soma de produto no domínio logarítmico com base na rotina implementada no Matlab [4].

2.4. **Perfuração**

A Perfuração tem normalmente por objetivo o aumento da taxa de informação de um código, por supressão de alguns bits da mensagem codificada, o que provoca uma diminuição da redundância da mesma. No contexto das técnicas de ICSHK e SCSHK, o seu papel é aumentar a segurança por eliminação dos bits da chave antes da transmissão (figura 8) por forma a ocultar esta. Neste trabalho, tendo em vista o desenvolvimento de técnicas de segurança na camada física adaptativas iremos estudar o desempenho das técnicas ICSHK e SCSHK aquando a eliminação de um número variável de bits de chave/mensagem, ou seja, apagando apenas bits da chave, apenas da mensagem ou de ambos (i.e. 25% dos bits de

perfuração na chave e 75% dos bits de perfuração na mensagem). De referir que neste estudo iremos considerar que ao perfurador, apenas é indicado em que parte da palavra do código os bits são apagados (sobre a chave ou mensagem) sendo que a sua posição nestes blocos é aleatória. Este processo, como seria de esperar, pressupõe a inserção do lado do recetor, do mesmo número de amostras a zero nas posições que foram eliminadas.

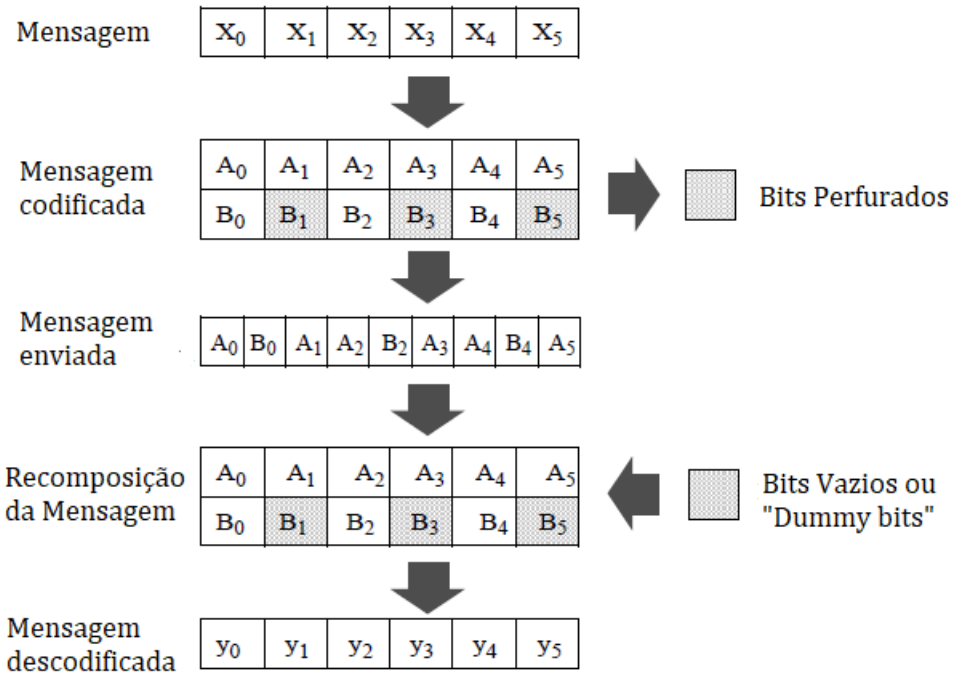


Figura 8: Procedimento de Perfuração e inserção de bits de uma mensagem.

De modo a complementar o exemplo dado pela figura anterior, foram usadas mascaras para posicionamento da perfuração. Assim, para além de o utilizador poder indicar o número de bits que quer perfurar, também poderá indicar o intervalo de índices onde este fenómeno ocorrerá no código, sendo assim possível, como foi dito antes, que a perfuração vá ocorrer apenas sobre a chave, mensagem ou ambos.

3. MÉTRICAS

Neste capítulo vão ser abordadas as métricas práticas utilizadas para aferir o nível de segurança obtido pelos mecanismos de codificação apresentados. Neste contexto será proposto uma variante à definição da métrica intervalo de segurança (ou *security gap*). As métricas aqui apresentadas permitem analisar a segurança de comunicação providenciada pelo método de codificação usado através da análise dos resultados de simulação de *Monte Carlo* e identificar as limitações dos referidos métodos.

3.1. Métricas Analíticas

Em 1949, foi proposto por Claude Shannon a primeira métrica de segurança denominada por segurança perfeita (ou *Perfect Secrecy*) [12]. Tal métrica impõe como requisito, para garantia de segurança, que a informação mútua entre a mensagem *M* e a palavra de código obtido á saída do codificador, X^n, seja igual a zero, ou seja:

$$I(M; X^n) = 0 \tag{4}$$

Quando ocorre a *Perfect Secrecy*, é necessário mencionar que tanto *M* como X^n, são estatisticamente independentes o que quer dizer que o conteúdo de X^n não revela qualquer, informação adicional sobre *M*. Posto isto, Shannon conclui que para alcançar este estado de segurança é necessária a utilização de uma chave secreta de dimensão igual à mensagem a enviar [2].

Alguns anos mais tarde, em 1975, e perante a dificuldade de encontrar códigos de comprimentos finito capazes de garantir (4) foi proposta por Aaron Wyner uma métrica de segurança mais fraca em relação à existente. Esta métrica, também conhecida por segurança fraca (ou *Weak Secrecy*), atua na condição de que a informação mútua entre *M* e a palavra de código observada no lado do espião, $\tilde{Z}$, tenda para zero, quando o comprimento do código, n, tende para valores infinitos, ou seja:

$$\lim_{n\to\infty} \frac{1}{n} I\left(M; \tilde{Z}\right) = 0 \tag{5}$$

Visto isto, em vez de a palavra de código, X^n, garantir que não fornece nenhuma informação sobre *M*, este critério garante que caso a informação de *M*, involuntariamente fornecida por $\tilde{Z}$, seja suficientemente pequena, para que o fator de $\frac{1}{n}$ possa reduzir o seu resultado para 0. Esta condição é válida se partirmos do princípio que o recetor legítimo tem vantagem contra o seu adversário, o espião.

A métrica de *Weak Secrecy* revela falhas em termos de segurança na construção do código não sendo, desta forma totalmente satisfatória como métrica de segurança [26]. Assim, Ueli Maurer [25] introduziu o conceito de *Strong Secrecy*, em que o critério de segurança consiste na informação mútua entre *M* e $\tilde{Z}$ ser assintoticamente nula para n, ou seja:

$$\lim_{n\to\infty} I(M;\tilde{Z}) = 0 \tag{6}$$

As métricas teóricas mencionadas em cima, são de aplicabilidade limitada em modelos de canais reais (i.e., modelos de Gaussian ou de *fading)* onde o cálculo analítico da informação mútua (ou mesmo, a tentativa de estimar a mesma por simulação de *Monte Carlo*) é uma tarefa quase impossível.

Outra grande limitação destas métricas é que teoricamente são impostas condições de garantias de segurança apenas para os casos em que o comprimento da palavra de código tende para o infinito. Isto, representa obviamente uma deficiência aquando aplicado no contexto de sistemas reais em que o uso códigos de comprimento médio ou curto é desejável. Assim, apesar de as métricas baseadas em informação mútua serem as que dão mais garantias de segurança, elas são de difícil aplicabilidade a códigos reais. Neste sentido, surgiu a necessidade de usar medidas práticas, sendo que, alguns autores propuseram analisar a segurança dos sistemas de codificação propostas para segurança, através do BER [7] e [9], medido na saída do descodificador.

3.2. **Métricas práticas baseadas em BER**

Na sequência da necessidade de medidas de segurança prática, os autores de [7] introduziram o conceito de Intervalo de Segurança que consiste num intervalo de segurança de BER. Este intervalo corresponde respetivamente a uma diferença em dB, para dois valores diferentes de SNR respetivamente um SNR mínimo ($SNR_{B,min}$) com garantias de fiabilidade

para o Bob e um SNR máximo ($SNR_{E,max}$) com garantias de segurança para com o Eve. Esta métrica apresenta-se útil para analisar o desempenho de um sistema, pois avalia a vantagem que o recetor legítimo precisa à partida em relação ao recetor não legítimo, bem como os valores de *threshold* limitativos para a fidelidade e segurança desejados. Na figura 9, encontra-se ilustrado o conceito de Intervalo de Segurança, o qual será apresentado com mais detalhe no próximo subcapítulo..

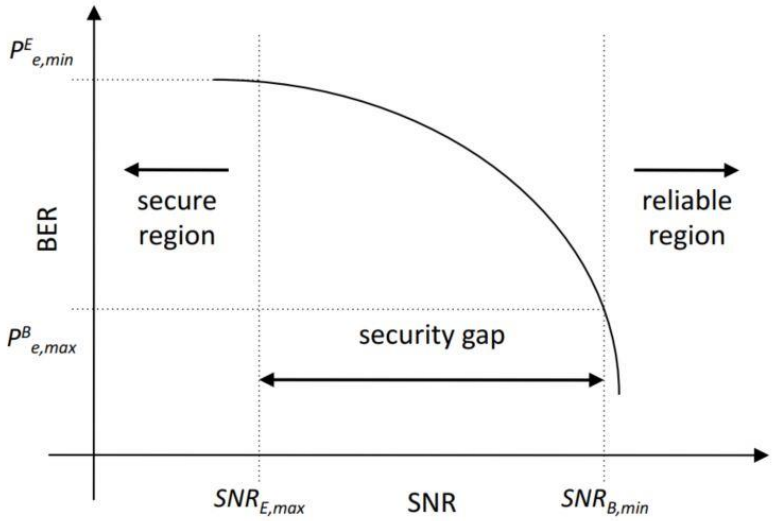


Figura 9: Conceito de "Security Gap", ou Intervalo de Segurança. [7]

Através da seguinte figura 9, é possível ter uma noção generalista do que se pretende com o conceito de segurança "adaptativa", pois a segurança adaptativa permite ajustar a operação dos esquemas de codificação para segurança desenvolvidos às condições de operação dos dispositivos envolvidos. Esses dispositivos podem operar a diferentes níveis de relação sinal-ruído (SNR), desta forma, tendo que se ajustar os esquemas de codificação por forma a garantir adequados níveis de fiabilidade (para o Bob, a operar a um nível mínimo de $SNR_{B,min}$) e confidencialidade (contra Eve, a operar a um nível máxima de $SNR_{E,max}$).

Tendo esta ideia, de segurança adaptativa em mente, é preciso elaborar um pouco mais o conceito de BER. Apesar de ser uma métrica consistente para a análise de segurança, esta também possui lacunas, ou seja, o facto de rácios muito altos de BER não implicar necessariamente que não tenha vazado informação. Na verdade, a criptografia em si, é regida através da segurança computacional e esta não revela qualquer informação sobre a mensagem, pois atualmente não existe poder computacional suficiente que seja capaz de tal descodificação, sem recorrer ao uso da chave.

Assim, enquanto o BER, eventualmente, poderá fornecer informação útil sobre a informação recebida ao espião, é necessário ter a perceção que os cálculos de BER são feitos através de uma média de uma grande quantidade de dados. Com isto dito, chegamos á conclusão que, teoricamente, quanto melhor for a distribuição dos erros, menor vai ser a questionabilidade de segurança.

O que se pretende com o BER é uma taxa de erros muito baixa para Bob e o mais próximo possível de 0.5 (ou 50%) para Eve. No entanto, usando apenas o BER para análise de performance nem sempre é o mais aconselhável visto o BER ser uma medida média o que significa que quando ocorrem muitos erros na transmissão, tal não implica necessariamente que o Eve não consiga extrair informação das palavras recebidas, isso porque poderão chegar palavras sem qualquer erro, ou seja, não se pretende que a taxa de erros seja elevada ou baixa, mas sim de valores intermédios. Como a análise de BER é feita através de simulação, a melhor maneira para contrariar este fenómeno é realizar um grande número de simulações a fim de obter uma melhor estimativa.

Na definição das referidas métricas iremos usar como referência a figura 10 [11], que representa de forma genérica as etapas de descodificação levadas a cabo por Bob e Eve aquando a transmissão através de um canal Wiretap Gaussiano.

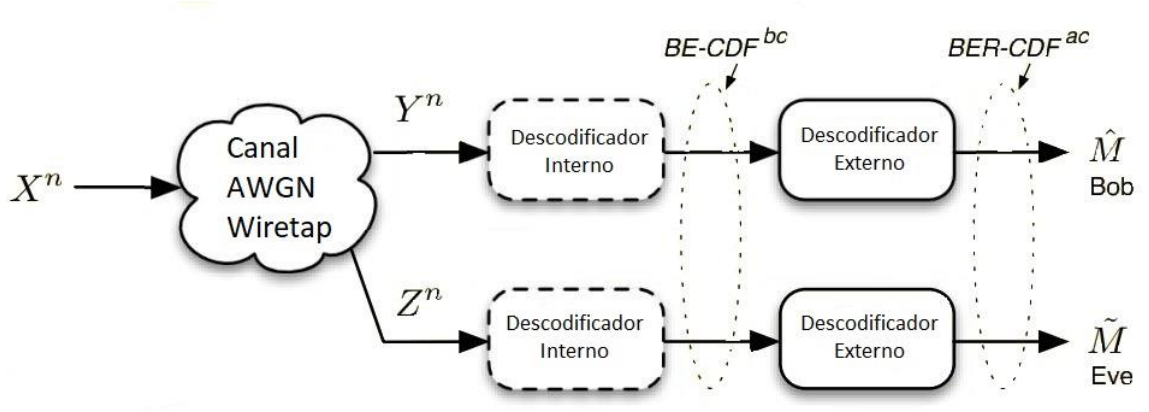


Figura 10: Esquema de um modelo de canal. Ilustração do *Outer Coder* (ou codificador externo) e do *Inner Coder* (ou codificador interno), que garantem a segurança e a fidelidade respetivamente. [17]

3.2.1. Função de distribuição cumulativa de número de erros

Tomemos como base ainda a figura 10, onde o descodificador externo desempenha um papel fundamental na segurança, pois o desempenho deste no sucesso ou

falhanço na descodificação é que vai garantir que a mensagem estimada seja igual à mensagem original, $M($ como desejável para Bob), ou apresente uma elevada taxa de erros (como desejável contra Eve). A primeira métrica, função de distribuição cumulativa de número de erros (*Bit Error Cumulative Distribution Function*, ou BE-CDF), encontra aplicação neste contexto, quando o código externo é um código corretor até $t - erros$ (ex.: um código com distância mínima de $2t + 1$).

Definição 1 (**Função de distribuição cumulativa de número de erros** de um código C_i também designada de **BE-CDF^{bc} (t, SNR, S_m, C_i)**, fornece a probabilidade de obter à saída do descodificador de C_i, t erros ou menos por palavra descodificada, i.e., Pr (E ≤ t), em função do SNR para mensagens de comprimento S_m, codificadas pelo código C_i [11]).

Tomando como referência a figura 10 e considerando C_i como sendo o código interno, esta métrica permite prever a taxa de sucesso e falha no descodificador externo, quando este é um código corretor até $t - erros$, sendo, portanto, bastante relevante para a construção do sistema bem como para aferir a fidelidade e segurança do mesmo.

Ao trabalhar com a distribuição do número de erros é facilmente ultrapassado uma das falhas de BER enquanto medida média, quando são usados códigos corretores de até $t - erros$. De facto, a utilização do BER à saída do descodificador interno (código C_i) para avaliar a probabilidade de o descodificador externo ser bem ou malsucedido pressupõe que a distribuição de erros obtida na descodificação de C_i é uniforme o que não é assim tão credível pois os códigos apresentam cumprimento curto.

Após escolhidos possíveis pontos de operação de SNR tanto para o recetor legítimo como para o recetor ilegítimo, e aplicando a métrica BE-CDF^{bc}, esta fornece informação útil sobre o comportamento do canal e do codificador interno. Por outro lado, a métrica pode ser usada na escolha do código externo, i.e., na necessária capacidade de correção, tomando como referência pontos espectáveis de funcionamento $SNR_{B,min}$ e $SNR_{E,max}$.

A métrica BE-CDF permite, pois, avaliar o sucesso da descodificação do codificador externo em termos probabilísticos permitindo assim julgar se o sistema é viável e seguro. No entanto, no caso de ocorrer uma falha no descodificador externo não é legítimo garantir ainda assim que o recetor ilegítimo não vá obter parte da mensagem. Com vista a procurar soluções para este problema outra métrica apresentada no seguinte subcapítulo.

3.2.2. Função de Distribuição Cumulativa de Taxa de Erros

Como referido anteriormente, uma falha por parte do descodificador externo não garante que o Eve não possa obter parte dos bits da mensagem. Assim, foi proposta uma nova métrica, a função de distribuição cumulativa da taxa de erros, que avalia a probabilidade de o descodificador externo vazar informação quando é observado à sua saída um BER, para a mensagem estimada, próximo de 0.5.

Definição 2 (Função de Distribuição Cumulativa da Taxa de Erros [11]): Também designada BER-CDF$^{ac}(\delta, \frac{E_b}{N_0}, Sb, C)$ é a probabilidade, função de $\frac{E_b}{N_0}$, de a taxa de erros $\hat{P}_b$ calculada por mensagem estimada obtida após descodificação para um código C (ou, opcionalmente, concatenação de códigos) ser superior a $0.5 - \delta$, com $\delta \ll 0.5$.

$$\Pr(\hat{P}_b > 0.5 - \delta) \tag{7}$$

O BER é uma métrica bem aceite para a fiabilidade, mas o propósito de BER-CDF é garantir segurança contra Eve, privando-o de informação relevante. Esta métrica prática obtida por simulação permite uma avaliação mais fidedigna da segurança obtida.

À semelhança da métrica BE-CDF^{bc} em que era especificada um nível de segurança baseado no parâmetro t, para a métrica BER-CDF^{ac}, o utilizador deve especificar o nível de segurança, i.e., definir um dado δ. De notar também, que a nomenclatura *bc* e *ac*, indicam que as métricas atuam *before* (antes) e *after* (depois) do código externo, respetivamente conforme ilustrado na figura 10.

Em suma, o BER-CDF^{bc} tem como utilidade a identificação das regiões de operação para o Bob, em termos de SNR, que visam obter uma taxa alta de descodificação bem-sucedida, alcançando assim confiabilidade. Também fornece informação sobre as regiões de operação para Eve, de modo a garantir uma probabilidade alta de o descodificador falhar. Por outro lado, o BER-CDF^{ac}, permite avaliar de forma mais fidedigna a segurança providenciada face a um Eavesdropper aquando falha do descodificador do código externo que apresenta um BER de aproximadamente 0.5.

No próximo capítulo, ambas as métricas são aplicadas na análise de desempenho das técnicas de segurança adaptativa para a camada física propostas nesta dissertação. Será também usada uma variante da métrica intervalo de segurança (ou *Security Gap*) que é definido na próxima secção.

3.3. **Intervalo de Segurança (nova definição)**

As limitações inerentes ao uso da medida de BER para avaliação da segurança, acorrentam idênticas limitações relativas á utilidade da métrica intervalo de segurança conforme definida em [9] [11].

Nesta dissertação é proposta uma redefinição da medida intervalo de segurança tendo por base o uso da métrica BER-CDF^{ac} para determinação do $SNR_{E(\max\ )}$ máximo que garante segurança face ao Eve, dado um dado valor δ; e o uso do BER para determinação do $SNR_{B(\min\ )}$ que garante fiabilidade para com o Bob considerando um dado valor máximo de BER_{Max} admissível na receção por parte do Bob.

$$\text{i.e., } SG_{new} = SNR_{B(\min\ )} - SNR_{E(\max\ )}$$

ou

$$SG_{new} = f(BER_{Max}) - f(BER - CDF^{ac})$$

Esta métrica pode ser também expressa de forma equivalente em função de $\frac{E_b}{N_0}$ com E_b a energia gasta por bit de informação enviado e N_0 a densidade espetral de potência de ruído, através da simples relação entre $\frac{E_b}{N_0}$.

$$SNR = \frac{E_b}{N_0}(\log_2 M)R \tag{8}$$

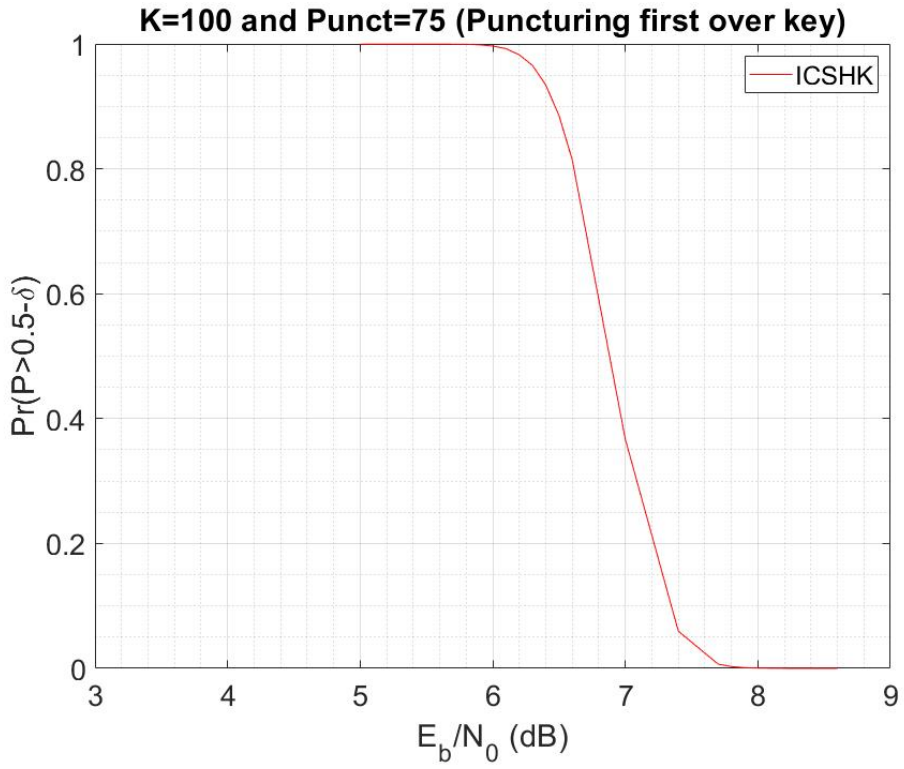


Figura 11: Analise do gráfico da Probabilidade de Erro, usando o modelo de Interleaving, para uma chave de 100 bits e perfuração de 75 bits sobre a chave (com δ=0.05)..

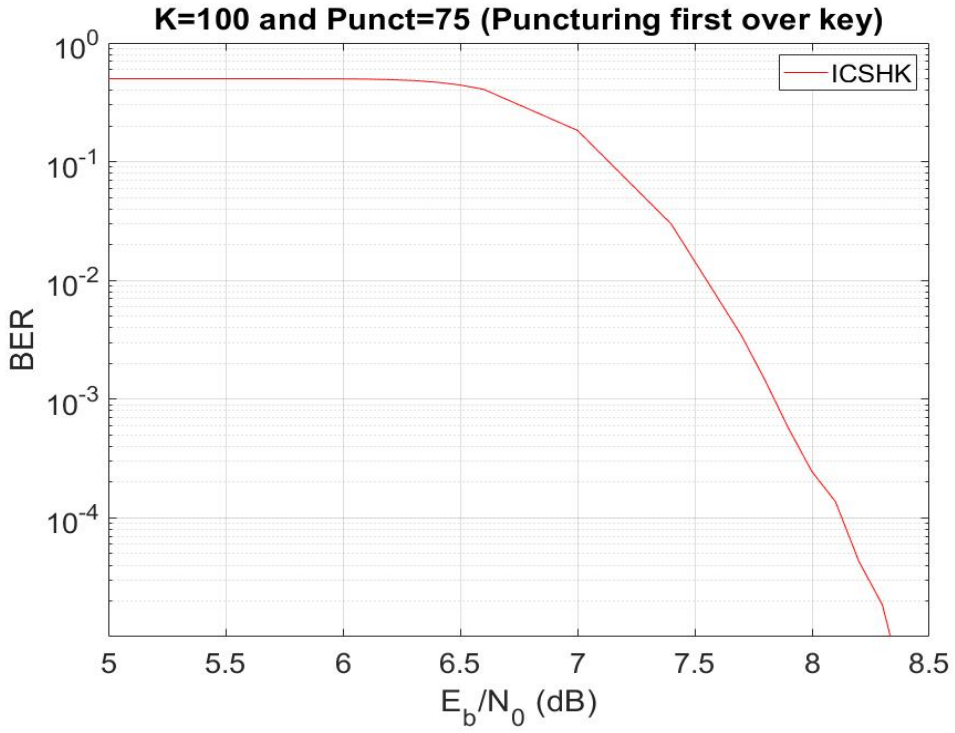


Figura 12:Analise do gráfico de BER, usando o modelo de Interleaving, para uma chave de 100 bits e perfuração de 75 bits sobre a chave.

Onde M é a ordem da modulação usada na transmissão e R o rácio de codificação empregue.

De modo a facilitar a compreensão da nova definição de *Security Gap*, ou traduzindo à letra "Intervalo de Segurança", foram gerados dois gráficos, figura 11 e figura 12, que dizem respeito ao Bit Error Rate (BER) e à BER-CDF, respetivamente.

Assim, através da análise do gráfico na figura 11, onde está presente o BER no eixo vertical, o nosso ponto de interesse para referência vai ser onde o BER é pequeno de forma a garantir fiabilidade sem perda de generalidade tomando como referência de fiabilidade um BER= 10^{-4}, para o Bob verifica-se da figura 11 que o mesmo é atingível para um $(E_b/N_0)_{B,min} \approx$ 8,12 dB. Já no que respeita à segurança, tomando como referência um δ=0.05 e uma $\Pr(\hat{P}_b > 0.5 - \delta) > 99\%$, na figura 12, que tal é garantido para o Eve com $(E_b/N_0)_{E,max} \approx$ 6,12 dB.

Agora que já temos os dois pontos de interesse, neste caso para o modelo de *Interleaving*, é só fazer o mesmo para o modelo de *Scrambling*, e a partir a partir de aí construir o gráfico de intervalo de segurança desejado.

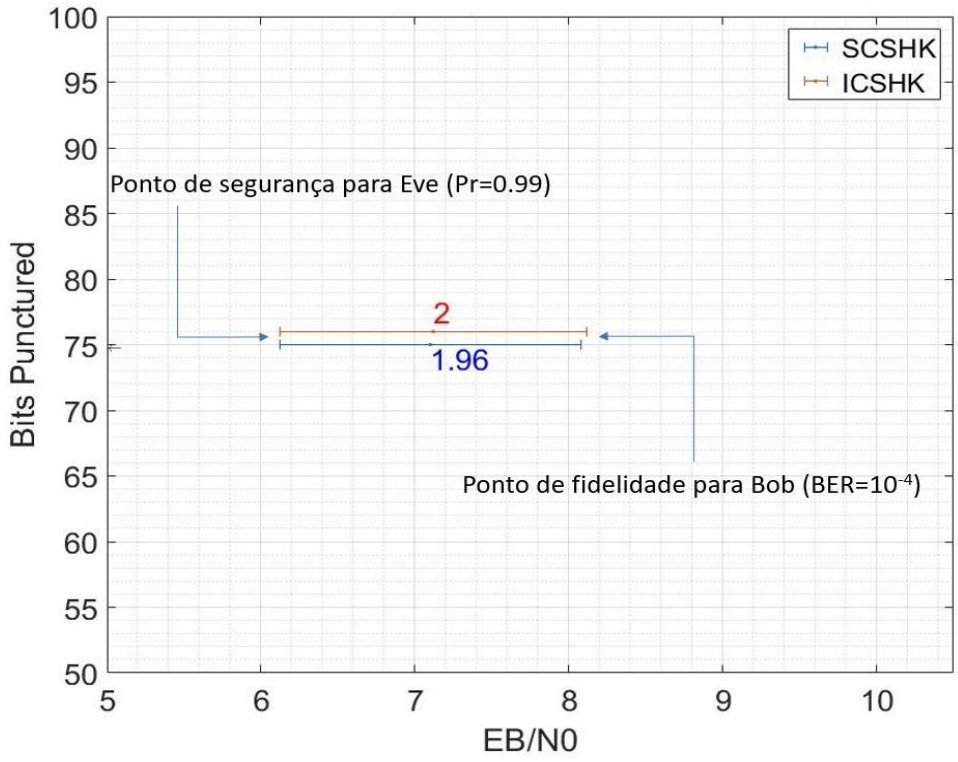


Figura 13:Analise do gráfico do Intervalo de Segurança, usando os modelos de Interleaving (a vermelho) e de Scrambling (a azul), para uma chave de 100 bits e perfuração de 75 bits sobre a chave.

Como podemos analisar através da figura 13, de ambos os gráficos gerados (figura 11 e figura 12), vai ser retirado um único ponto de interesse para a construção do gráfico de Security Gap, gerando posteriormente, um intervalo para o modelo de *Interleaving* e outro para o modelo de *Scrambling*. Este intervalo, é representado por uma cor que faz corresponder ao modelo utilizado em questão (como é indicado na legenda da figura 13) e por um número da mesma cor (correspondente ao valor do intervalo do Intervalo de Segurança). Este valor, em $\frac{E_b}{N_0}$, do intervalo de Segurança é obtido através da subtração do ponto de fidelidade em Bob pelo ponto de segurança em Eve. Como exemplo, e usando as medidas obtidas anteriormente, podemos verificar que: 8,12dB - 6,12dB = 2dB.

4. SEGURANÇA ADAPTATIVA NA CAMADA FÍSICA

As métricas presentes na secção anterior procuram fornecer técnicas para análise que têm como objetivo, para cada modelo em questão, validar a sua segurança. Neste capítulo as mesmas serão usadas no desenvolvimento e avaliação de técnicas de segurança adaptativa que pretendem ajustar a operação de esquemas de codificação para segurança desenvolvidos às condições de operação dos dispositivos envolvidos. Esses dispositivos podem operar a diferentes níveis de relação Sinal-Ruído (SNR), desta forma, tendo que se ajustar os esquemas de codificação por forma a garantir adequados níveis de fiabilidade para o Bob a operar a um dado $SNR_{B,min}$ e confidencialidade contra Eve, a operar noutro nível de $SNR_{E,max}$ (onde $SNR_{E,max} < SNR_{B,min}$).

Iremos focar o nosso estudo de segurança adaptativa nas técnicas ICSHK e SCSHK apresentadas no capítulo 2.

4.1. Perfuração para segurança adaptativa

Uma possibilidade para ajustar a região de operação dos mecanismos de codificação ICSHK e SCSHK desenvolvidos consiste em ajustar o nível/quantidade de perfuração que é efetuado sobre a chave de *Interleaving/Scrambling* e/ou sobre a mensagem.

O primeiro passo consiste em definir como implementar a perfuração. Recorrendo às mesmas configurações dos esquemas originais [11] [17], nos quais é utilizado um código LDPC com dimensões (1536,1280) e uma chave de 100 bits, o primeiro passo consistirá em avaliar o efeito da perfuração sobre a chave ou sobre a mensagem. Para tal, iremos efetuar testes com níveis de perfuração de 50, 100 e 150 bits. Para cada nível de perfuração iremos testas diversas configurações que permitam avaliar o efeito de efetuar perfuração sobre a chave apenas, sobre a mensagem apenas, ou sobre ambas, simultaneamente.

4.1.1. Estratégia de Perfuração

Para aferir o impacto de perfurar os bits da chave e/ou da mensagem, para cada nível de perfuração vamos testar diversas configurações. Em particular, configurações em que fazemos sempre perfuração sobre a chave primeiro, sobre a chave e sobre a mensagem,

ou sobre a mensagem apenas. Por exemplo, consideremos o caso em que vamos perfurar 50 bits. Para este nível de perfuração, iremos testar perfuração de 50 bits sobre a chave (key50), 50 bits sobre a mensagem (Msg50), 25 sobre a chave e 25 sobre a mensagem (25key25Msg), 13 sobre a chave e 37 sobre a mensagem (13key37Msg), bem como 38 sobre a chave e 12 sobre a mensagem (38key12Msg).

Com isto dito, as condições de simulação que foram as seguintes:

Técnica de Codificação para Segurança	ICSHK e SCSHK
Código Interno	LDPC (1536,1280); LDPC (256,128)
Número de bits de Perfuração	N_{punct}=50, 100, 150
Canal	AWGN
Tamanho da Simulação	20×10^6 bits
Plataforma de desenvolvimento	Matlab R2018a

Tabela 1: Quadro Resumo incidente nas condições de simulação.

Dito isto, e de encontro a satisfazer um dos objetivos propostos inicialmente foram estabelecidos gráficos comparativos com o mesmo número total de bits de perfuração e as mesmas percentagens aplicadas no código, para os dois tipos de modelos: *Interleaving* e *Scrambling.* Na legenda das figuras abaixo expostas, as legendas nos gráficos ilustram a configuração de bits em causa, i.e., para o caso de uma configuração "13key37Msg", sabemos que a perfuração atua sobre 13 bits de chave e 37 de mensagem.

Resultados de simulação BER função da perfuração variável:

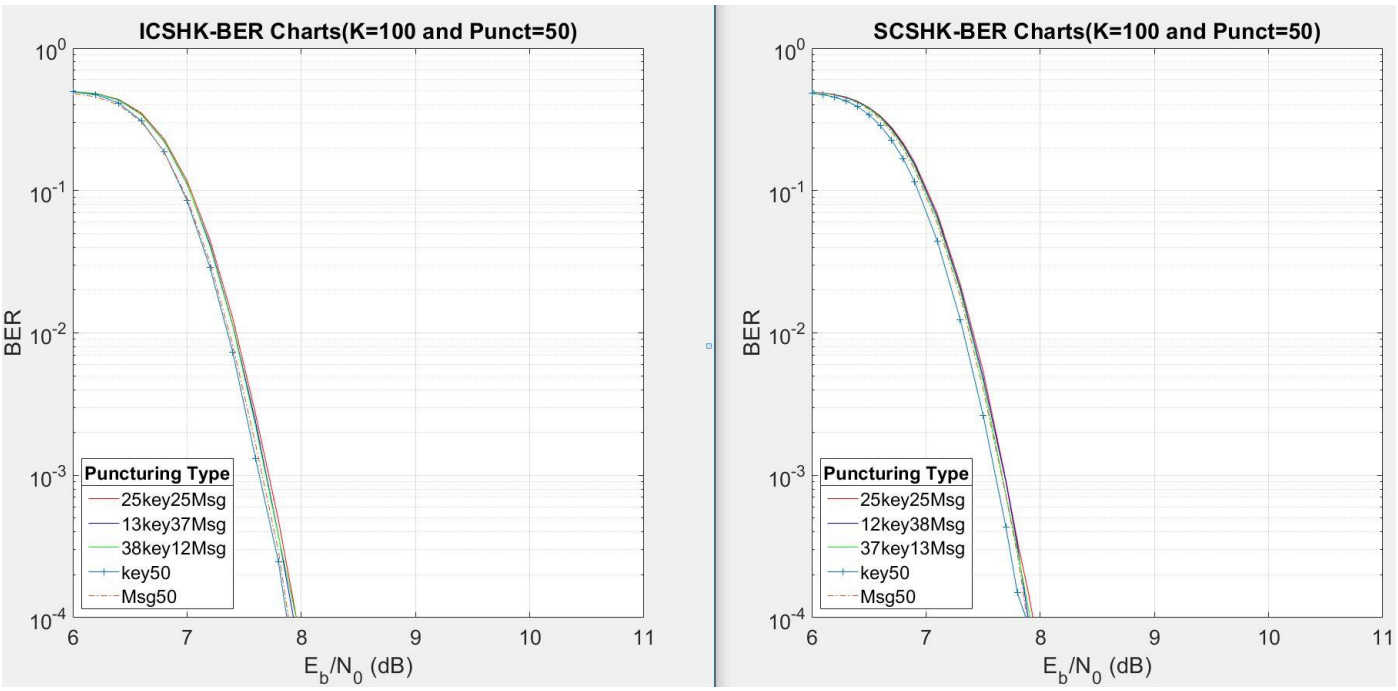


Figura 14: Comparação dos gráficos de BER, em *Interleaving* e Scrambling, para chave de 100 bits e 50 bits de perfuração.

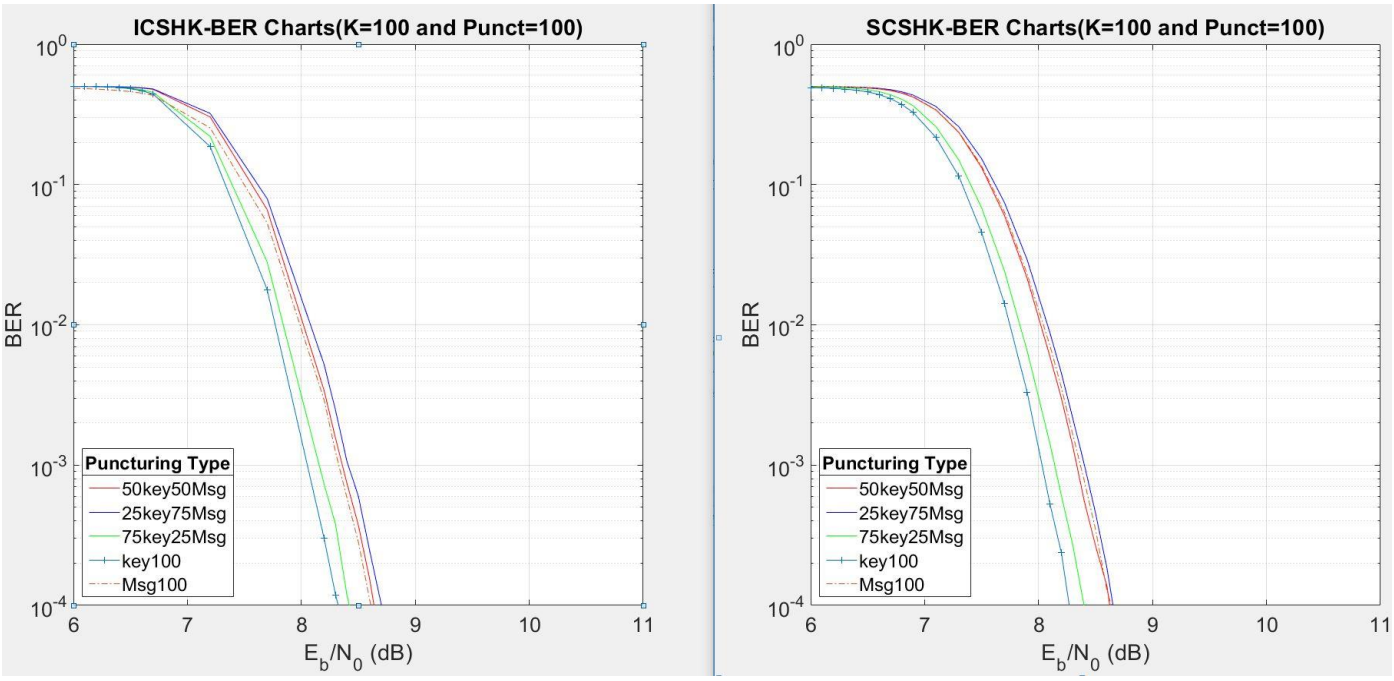


Figura 15: Comparação dos gráficos de BER, em Interleaving e Scrambling, para chave de 100 bits e 100 bits de perfuração.

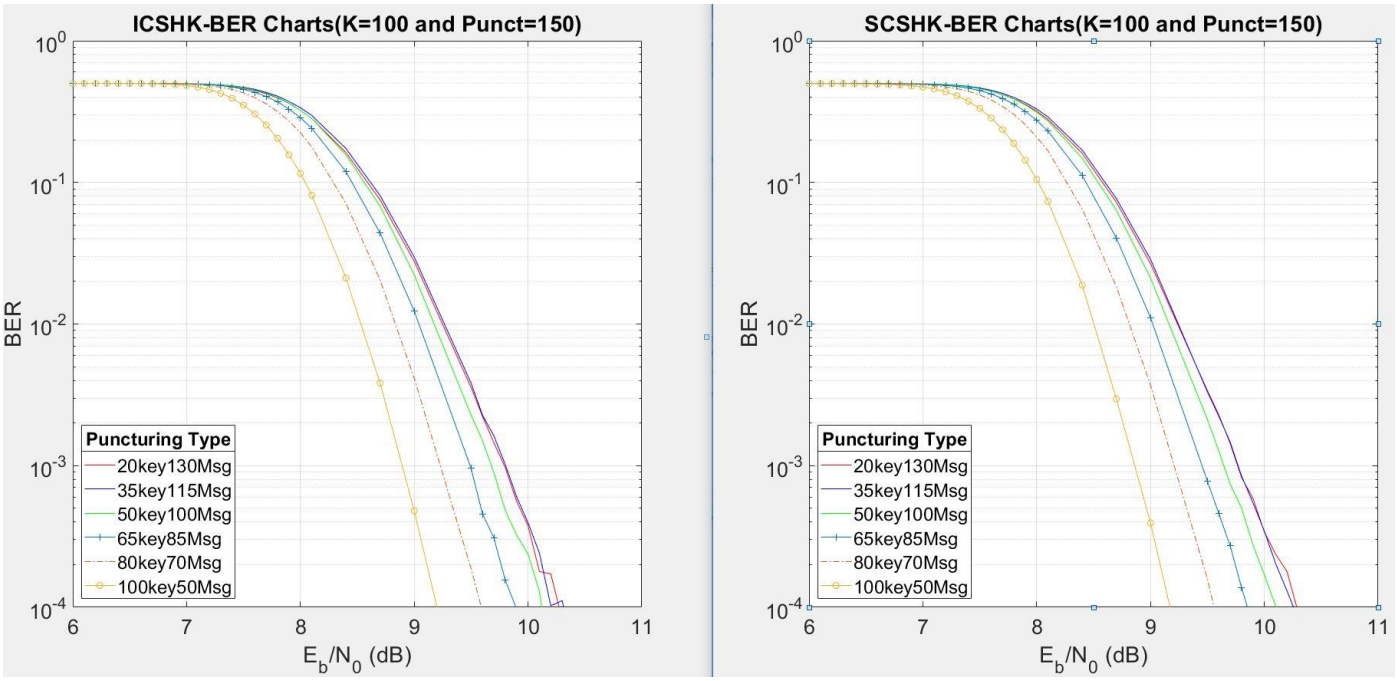


Figura 16: Comparação dos gráficos de BER, em Interleaving e Scrambling, para chave de 100 bits e 150 bits de perfuração.

Resultados de simulação em termos de BER-CDF^{ac} função de perfuração variável:

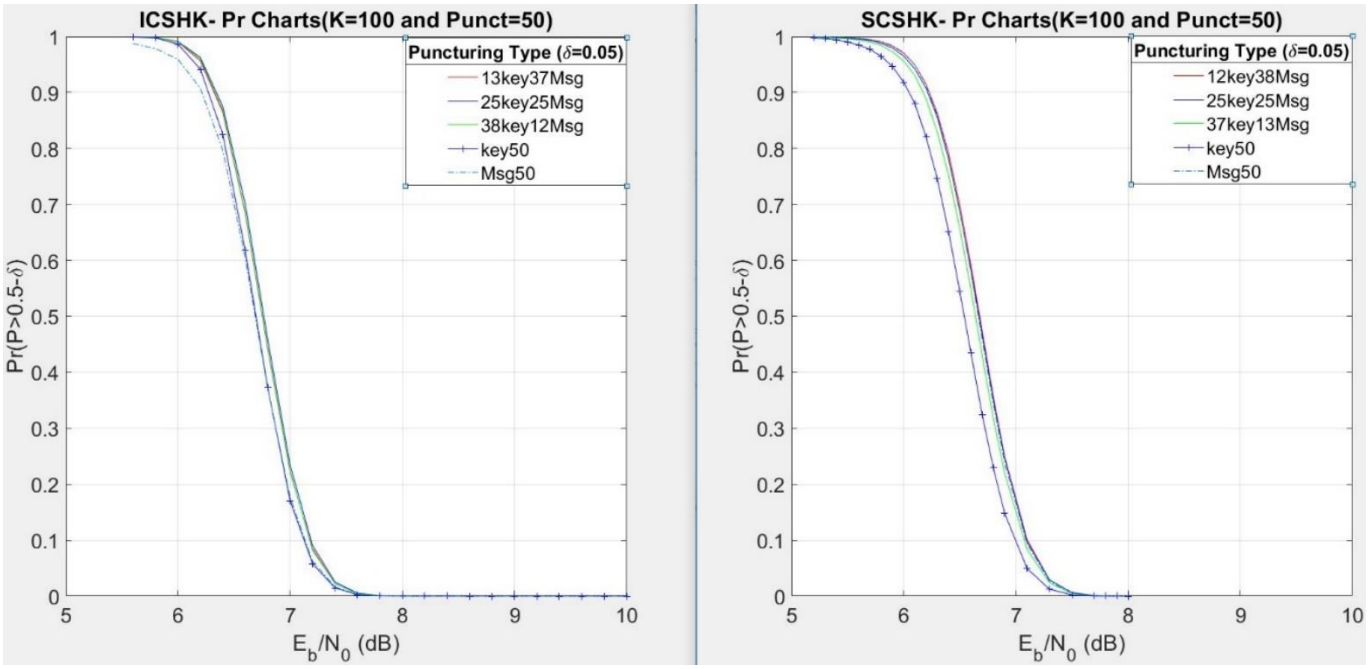


Figura 17: Comparação dos gráficos da Probabilidade de Erro, em Interleaving e Scrambling, para chave de 100 bits e 50 bits de perfuração.

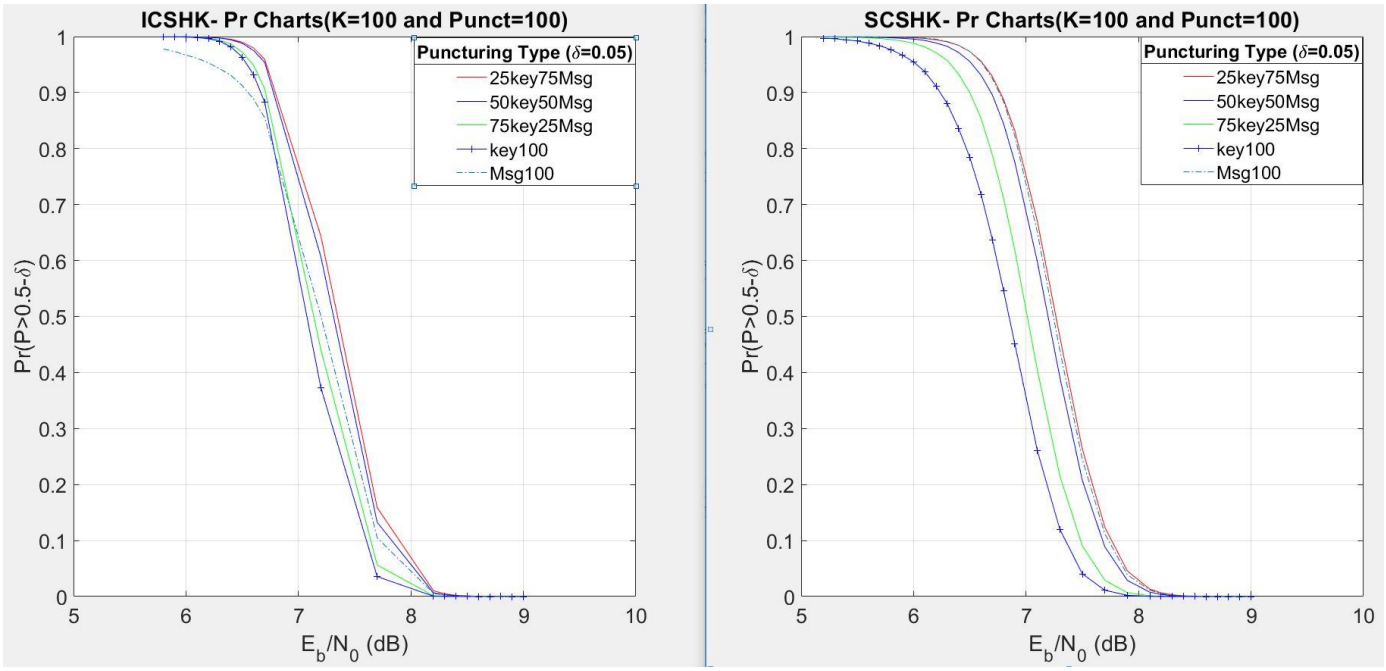


Figura 19:Comparação dos gráficos da Probabilidade de Erro, em Interleaving e Scrambling, para chave de 100 bits e 100 bits de perfuração.

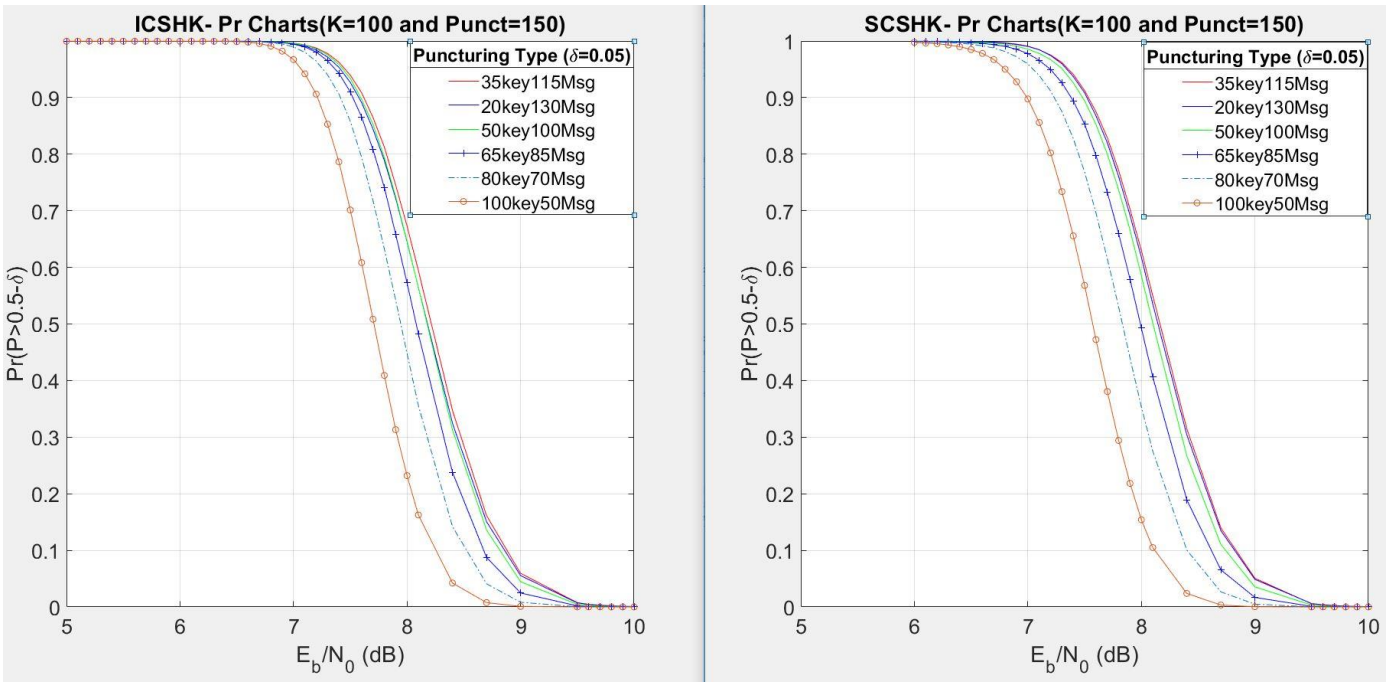


Figura 18:Comparação dos gráficos da Probabilidade de Erro, em *Interleaving* e Scrambling, para chave de 100 bits e 150 bits de perfuração.

NOTA: Para evitar a saturação de resultados, outros gráficos foram colocados em Anexo.

Perfuração	Configuração	E_b/N_0 @ $BER = 10^{-4}$	E_b/N_0 @ $BER - CDF = 0.99$	Intervalo de Segurança (dB)
50 bits	13key37Msg	7.93; 7.89	6; 6	1,93; 1,89
	25key25Msg	7.94; 7.94	6.03; 5.95	1,91; 1,99
	38key12Msg	7.95; 7.91	6.01; 5.9	1,94; 2,01
	key50	7.87; 7.87	5.93; 5.95	1,94; 1.92
	Msg50	7.89; 7.88	5.55; 5.7	2,34; 2,18
100 bits	25key75Msg	8.7; 8.64	6.5; 6.3	2,20; 2,34
	50key50Msg	8.64; 8.62	6.46; 6.47	2,18; 2,15
	75key25Msg	8.43; 8.4	6.35; 6.27	2,08; 2,13
	key100	8.32; 8.28	6.3; 6,24	2,02; 2,04
	Msg100	8.62; 8.63	5.4; 5,3	3,22; 2,33
150bits	20key130Msg	10.24; 10.29	7.13; 7.13	3,11; 3,16
	35key115Msg	10.2; 10.25	7.16; 7,07	3,04; 3,18
	50key100Msg	10.1; 10.1	7.14; 6.91	2,96; 3,19
	65key85Msg	9.9; 9.85	7.08; 6.8	2,82; 2,95
	80key70Msg	9.6; 9.46	7; 6.85	2,60; 2,61
	100key50Msg	9.2; 9.19	6.82; 6.83	2,38; 2,36
	150Msg	10.17; 10.34	5.25; 5,6	4,92; 4,74

Tabela 2:Dados relativos aos gráficos representados anteriormente para as técnicas ICS-HK (resultados a azul) e SCS-HK (resultados a vermelho).

Os resultados anteriores encontram-se sumariados na Tabela 2 tomando os limiares de fiabilidade e segurança anteriormente referidos e para os quais foi calculado o intervalo de Segurança.

Relativamente a esta, os resultados para ICSHK encontram-se a azul e para SCSHK a vermelho, e, os melhores valores para o intervalo de segurança para cada tipo de perfuração encontram-se a sombreado.

Após a análise dos gráficos anteriores, chegou-se à conclusão que a melhor estratégia de perfuração, que conduz consistentemente a valores menores de intervalo de

segurança é realizar a perfuração somente na chave até preencher todos os bits nesta, e só depois nos bits de mensagem. Por exemplo: quando o objetivo é fazer perfuração de 90 bits, todos eles vão incidir na chave (K=100); mas quando o objetivo é fazer perfuração de 120 bits para a mesma chave (K=100), todos os bits da chave vão ser perfurados e ainda outros 20 bits na mensagem, escolhidos aleatoriamente.

A escolha desta estratégia de perfuração no ponto de vista teórico é fundamentada nas seguintes observações:

- Os resultados de BER relevantes para o recetor legítimo ao nível da avaliação da fiabilidade mostra que se verifica que as combinações correspondentes a um mesmo número de bits de perfuração que conduzem a curvas com um melhor desempenho, i.e., um menor Eb/N0 são, consistentemente, aquelas nas quais a perfuração sobre a chave é a maior;
- Os resultados de BER-CDF, relevantes para Eve em termos de segurança são aquelas em que se observa a obtenção de um $\Pr(\hat{P}_{\mathrm{b}} > 0.5 - \delta) \approx 1$ para valores de Eb/N0 mais elevados que não são necessariamente em que a perfuração sobre a chave é maior. No entanto aquando da análise combinada, a fiabilidade vs segurança, em termos de intervalo de segurança (ou S.G.) verifica-se conforme observado na tabela 2 que os menores valores de intervalo de segurança inferiores são obtidos para os casos onde a perfuração sobre a chave é maior. Para além disso observa-se ainda como expectável que o aumento do número de bits de perfuração conduz a uma degradação de desempenho com a deslocação das curvas para a direita.

De facto por análise da tabela 2 podemos observar que, os melhores resultados surgem quando a perfuração é mais incidente nos bits da chave. Posto isto, foi proposta a elaboração de um gráfico onde a perfuração era incidente primeiro sobre a chave só depois sobre a mensagem, daí resultando o seguinte gráfico, onde também os valores dos intervalos de segurança são expostos.

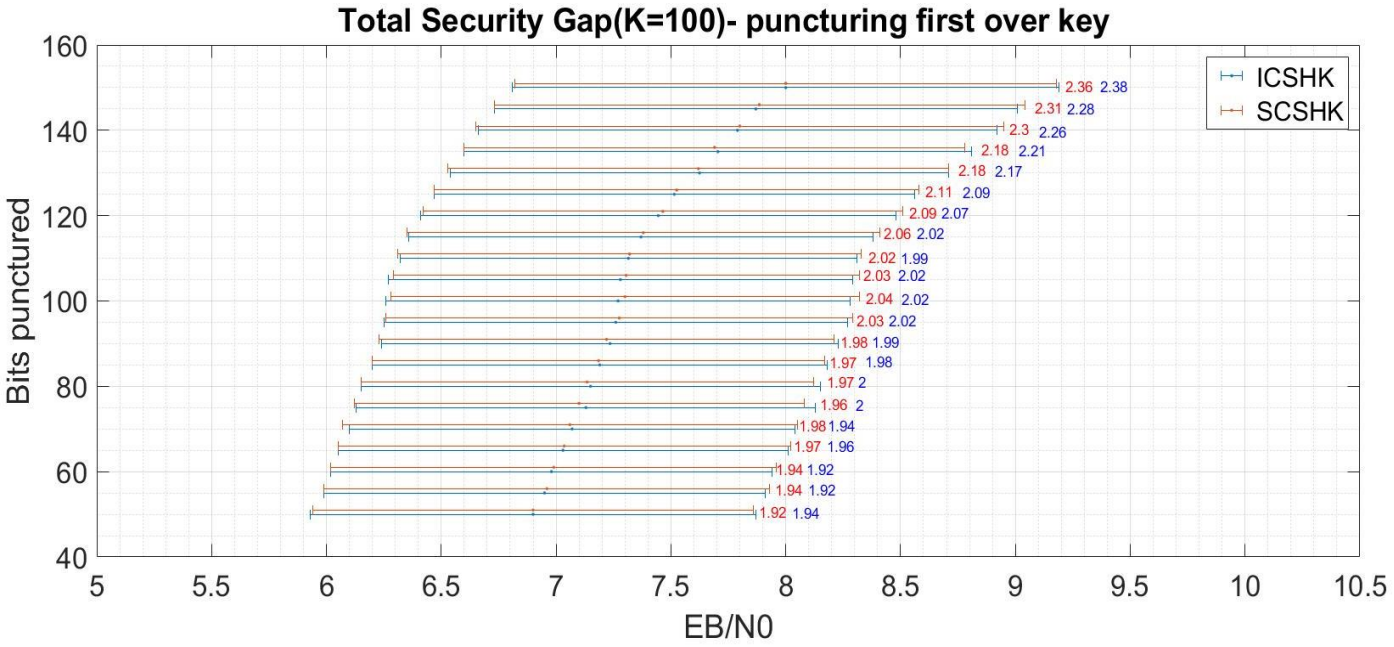


Figura 20: Ilustração de progressão do Intervalo de Segurança em código (1536,1280), fazendo perfuração primeiro apenas sobre a chave e só depois, adicionando bits de mensagem.

Estes resultados de intervalo de segurança podem ser observados na figura 20, que mostra a evolução do intervalo de segurança de acordo com o número de bits de perfuração. Cada linha horizontal no gráfico representa o intervalo de segurança, no qual o ponto mais à esquerda corresponde ao ponto de segurança para Eve, enquanto o ponto mais à direita ao ponto de fiabilidade para Bob. A sua diferença corresponde ao intervalo de segurança, cujo valor é indicado à direita da linha horizontal para os esquemas ICS-HK e SCS-HK.

Através da figura 20, podemos verificar alguns factos interessantes começando pelo facto de tanto num modelo como no outro no outro o comportamento do *Intervalo de Segurança* ser bastante similar, o que acaba por ser uma contribuição significativa neste trabalho. Isto porque, apresentando ambas as técnicas, um desempenho idêntico em termos de segurança na camada física, a técnica de SCS-HK apresenta vantagens de implementação em hardware pela simples ativação/desativação dos interruptores relativos ao polinómio de *Scrambling* (ver figura 6), ao passo que a técnica de SCS-HK obriga ao tabelamento de todas as sequências de baralhamento no transmissor e recetor.

Outra das conclusões, é que o intervalo de segurança além de se deslocar progressivamente para a direita também o seu valor aumenta pois à medida que são introduzidos mais bits de perfuração observa-se uma degradação mais acentuada da capacidade de correção do código, face ao aumento da região segura, o que é também

concluído pela diminuição do declive da região de “waterfall” do código observados nos gráficos de BER.

Se for feita uma análise minuciosa, podemos verificar que, na figura 20, os pontos do lado esquerdo referentes à análise dos gráficos da probabilidade de Erro têm um desvio menor entre os mesmos sendo mais próximos entre si, do que os pontos do lado direito refentes à taxa de bits errados, que se dispersam mais, com a perfuração de mais bits na chave. Outro reparo é que sensivelmente na zona de perfuração de 120 bits, o Intervalo de segurança começa a aumentar ligeiramente em relação a um padrão que é visível anteriormente. Isto deve-se aos pontos de fidelidade (do lado direito), avançarem mais face aos pontos de segurança (do lado esquerdo), como resultado da perfuração de bits da mensagem (que ocorre a partir dos 100 bits de perfuração, já que a chave tem tamanho de 100 bits). Isto valida novamente a escolha de efetuar perfuração sobre a chave antes de perfurar a mensagem.

Após obtenção dos resultados anteriores aquando do uso de um código de comprimento médio-longo (1536,1280) pretendia-se confirmar os resultados e conclusões se mantinham aquando o uso de códigos de comprimento curto. Assim, foi feito uso de um código LDPC (256,128), considerando uma chave composta apenas por 64 bits.

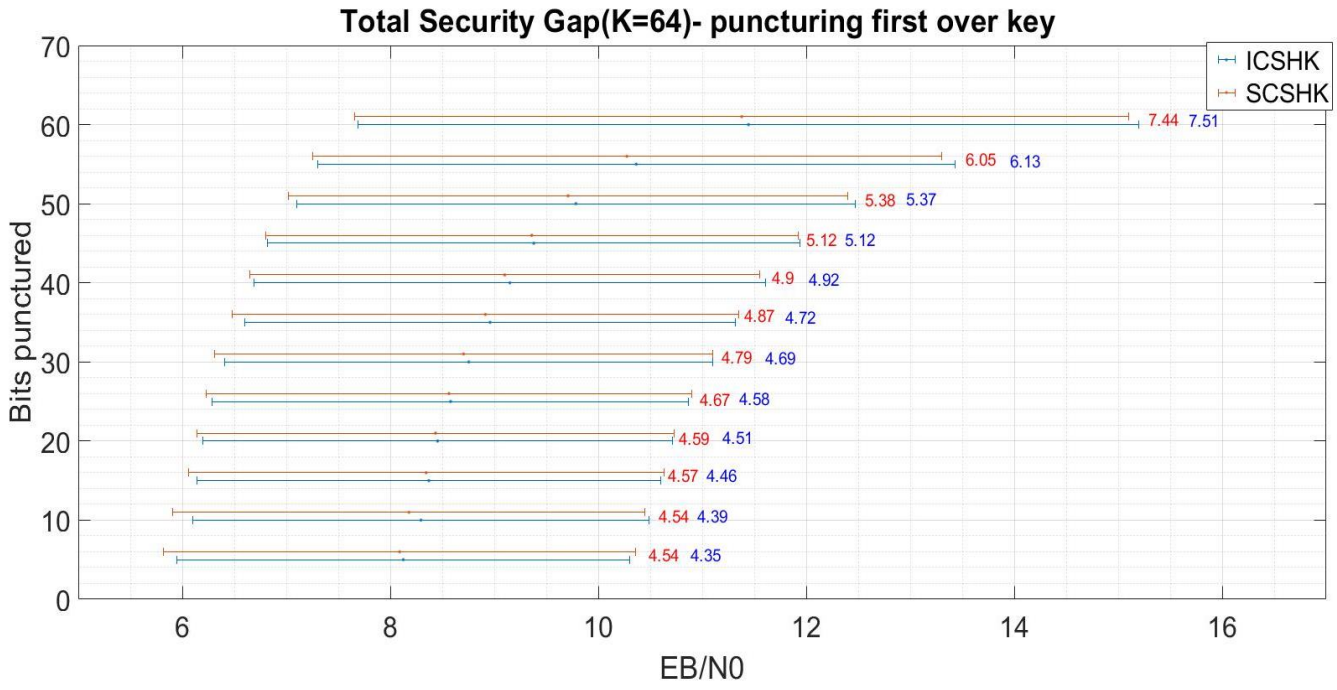


Figura 21:Ilustração de progressão do Intervalo de Segurança em código (256,128), fazendo perfuração primeiro apenas sobre a chave e só depois, adicionando bits de mensagem.

Através de uma análise, primeira da figura 21, é observado que ambos os modelos apresentam um comportamento bastante idêntico, o que é positivo, ainda que o modelo ICSHK apresente ligeiramente melhores valores para segurança, que o seu par.

Assim, tal como observado anteriormente para o código longo, para um aumento do nº de bits perfurados o Intervalo de segurança começa a aumentar ligeiramente em relação a um padrão que é visível anteriormente. Uma das razões para o intervalo de segurança ser superior é devido ao impacto da chave ser também superior, uma vez que o tamanho da chave tem um tamanho muito mais próximo tamanho da mensagem, em relação ao caso do código longo. Isto explica também que o intervalo de segurança comece a aumentar quando o nível de perfuração é ainda inferior ao tamanho da chave (perfuração de 55 bits), ao contrário do caso do código longo, no qual o intervalo de segurança só começa a aumentar mais quando a perfuração está nos 120-130 bits.

Comparativamente, entre os gráficos da figura 20 e 21, é notável que o caso em que o código é menor, os valores do intervalo de segurança são mais elevados, e como já tínhamos visto anteriormente quanto menor fosse o intervalo de segurança melhor ia ser o desempenho do código. De certa maneira, era esperado este tipo de observação pois o código ao ser menor e como a chave é uma porção significativa do mesmo (em relação a códigos de maior comprimento), os métodos de embaralhamento e perfuração no código vão atuar sobre menos bits, ou seja, de forma mais limitada, remetendo assim para piores valores de fidelidade e segurança.

Outro fator diferenciador, que poderá influenciar os resultados é que o tamanho da chave além de ser menor, ainda é uma porção significativa do código, consequentemente o rácio de codificação ser menor.

5. CONCLUSÕES

Esta dissertação tem como uma das finalidades a comparação dos esquemas propostos inicialmente, *Interleaving* e *Scrambling*, enquanto submetidas a estados iniciais iguais.

A aplicabilidade dos mecanismos de segurança na camada física depende da existência de esquemas que se possam adaptar às condições de canal para que o recetor legítimo possa beneficiar de uma taxa de erros almejada, sem que o adversário Eavesdropper possa obter mais informação do que a desejada.

Esta dissertação mostra que os esquemas de segurança na camada física baseados em baralhamento de informação (interleaving e Scrambling) e perfuração de bits, podem ser utilizados para facilitar a operação em condições de canal variáveis, tanto para o recetor legítimo como para o adversário. Essa adaptabilidade é obtida através da variação do nível de perfuração aplicado pelos esquemas ICSHK e SCSHK. Os resultados obtidos permitiram demonstrar que é possível adaptar a operação do sistema a diversos níveis de relação sinal-ruído tanto do recetor legítimo como do adversário. Foi também determinado que a abordagem com melhores resultados em termos de diminuir o intervalo de segurança consiste em efetuar perfuração primeiro sobre os bits de chave e apenas depois sobre os bits de mensagem. Por fim, e não menos importante, foi possível atribuir a característica de "adaptativa" à segurança da camada física pois é possível garantir uma transmissão segura e fidedigna, perante diversos pontos de operação tanto para o recetor legítimo como para o adversário.

5.1. Trabalho futuro

Visto que a partir dos resultados gerados foi possível retirar resultados interessantes, faria sentido aplicar as mesmas métricas a códigos de diferentes dimensões, como por exemplo os códigos polares ou BCH. Outro possível rumo, é a implementação dos esquemas apresentados em plataformas de Software Defined Radio, com proposta de um esquema de estimação de canal Alice → Bob com vista ao ajuste dinâmico do nº de bits de Puncturing face à qualidade do canal.

REFERÊNCIAS

[1] S. Lin e D. J. Costello, Error Control Coding, Prentice Hall, 2004

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

[3] A. B. Carlson, P. B. Crilly e J. C. Rutledge, Communication Systems, McGraw-Hill, 2002

[4] R. G. Gallager, "Low-Density Party-Check Codes, *IRE Transactions on Information Theory*, pp. 21-28, Janeiro 1962.

[5] D. J. C. MacKay, "Good Error- Correction Coded Based on Very Sparse Matrices", *IEEE Transactions on Information Theory*, vol.45, nº42, pp. 339-431, 1999.

[6] IEEE Standard for Local and metropolitan area networks, IEEE Standard 802.16e,2005.

[7] D. Klinc, J. Ha, S. W. McLaughlin, J. Barros, and B.-J.Kwak, "LDPC codes for the Gaussian wiretap channel", *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 3, pp. 532-540, Sept 2011.

[8] E. Arıkan, "Channel polarization: A method for constructing capacity achieving codes for symmetric binary-input memoryless channels", IEEE Trans. Inf. Theory, vol. 55, no. 7, pp. 3051–3073, 2009, ISSN: 0018-9448.
DOI: 10.1109/TIT.2009.2021379

[9] M. Baldi, M. Bianchi, and F. Chiaraluce, "Coding with scrambling, concatenation, and harq fot the awgn wire-tap channel: A security gap analysis", IEEE Transactions on Information Forensics and Security, vol. 7, no. 3, pp. 883-894, 2012.

[10] W. Ryan, An Introduction to Low-Density Parity-Check Codes, Universidade do Arizona, 2001.

[11] W. K. Harrison, D. Sarmento, J. P. Vilela, and M. Gomes, "Analysis of Short Blocklength Codes for Secrecy," ArXiv e-prints, Sep. 2015, [Online]. Available at http://arxiv.org/abs/1509.07092

[12] C. E. Shannon, "Communication theory of secrecy systems", *Bell System Technical Journal*, vol.28, pp. 656-715, Oct.1949.

[13] S. S. Haykin, Digital Communication Systems. Wiley, 2014.

[14] C. Ling, L. Luzzi, J.-C. Belfiore, and D. Stehle, “Semantically secure lattice codes for the gaussian wiretap channel,” IEEE Transactions on Information Theory, vol. 60, no. 10, pp. 6399–6416, 2014.

[15] A. Thangaraj, S. Dihidar, A. R. Calderbank, S. W. McLaughlin, and J. -M. Merolla, “Applications of LDPC codes to the wiretap channel”, IEEE Trans. Inf. Theory, vol. 53, no. 8, pp 2933-2945, August 2007.

[16] A. T. Suresh, A. Subramanian, and A. Thangaraj, “Strong secrecy for erasure wiretap channels”, IEEE Information Theory Workshop, 2010.

[17] J. P. Vilela, M. Gomes, W. K. Harrison, D. Sarmento e F. Dias, “Interleaved Concatenated Coding for Secrecy in the Finite Blocklenght Regime”, IEEE Signal Processing Letters, vol.23, nº 3, pp. 356-360, Março 2016.

[18] D. Sarmento, J. P. Vilela, W. K. Harrison e M. Gomes, “Interleaved Coding for Secrecy with a Hidden Key”, em IEEE Globecom Workshop on Trusted Communications with Physical Layer Security, San Diego, CA, USA, 2015.

[19] M. Baldi, M. Bianchi, e F. Chiaraluce “Coding With Scrambling, Concatenation, and HARQ for the AWGN Wire-Tap Channel: A Security Gap Analysis”, IEEE Transactions on Information Forensics and Security, vol. 7, no. 3, June 2012

[20] R. M. Tanner, “A Recursive Approach to Low Complexity Codes”, IEEE Transactions on Information Theory, Vols. %1 de 2IT-27, nº25, pp. 553-547, September 1981.

[21] R. Chien, “Cyclic decoding procedures for Bose- Chaudhuri-Hocquenghem codes”, IEEE Transactions on Information Theory (Volume: 10, Issue: 4, Oct 1964)

[22] G.Robinson, “Communication Networks with Layered Architectures” IEEE802 N-WEST Standards Meeting for Broadband Wireless Access Systems, 9 March 1999

[23] L. Wang, “Physical Layer Security in Wireless Cooperative Networks”, Wireless Networks, DOI 10.1007/978-3-319-61863-0_2, USA 2018

[24] C. Martins; T. Fernandes; M.Gomes; J.Vilela.” Testbed Implementation and Evaluation of Interleaved and Scrambled Coding for Physical-Layer Security”, IEEE 87th Vehicular Technology Conference (VTC Spring) 2018

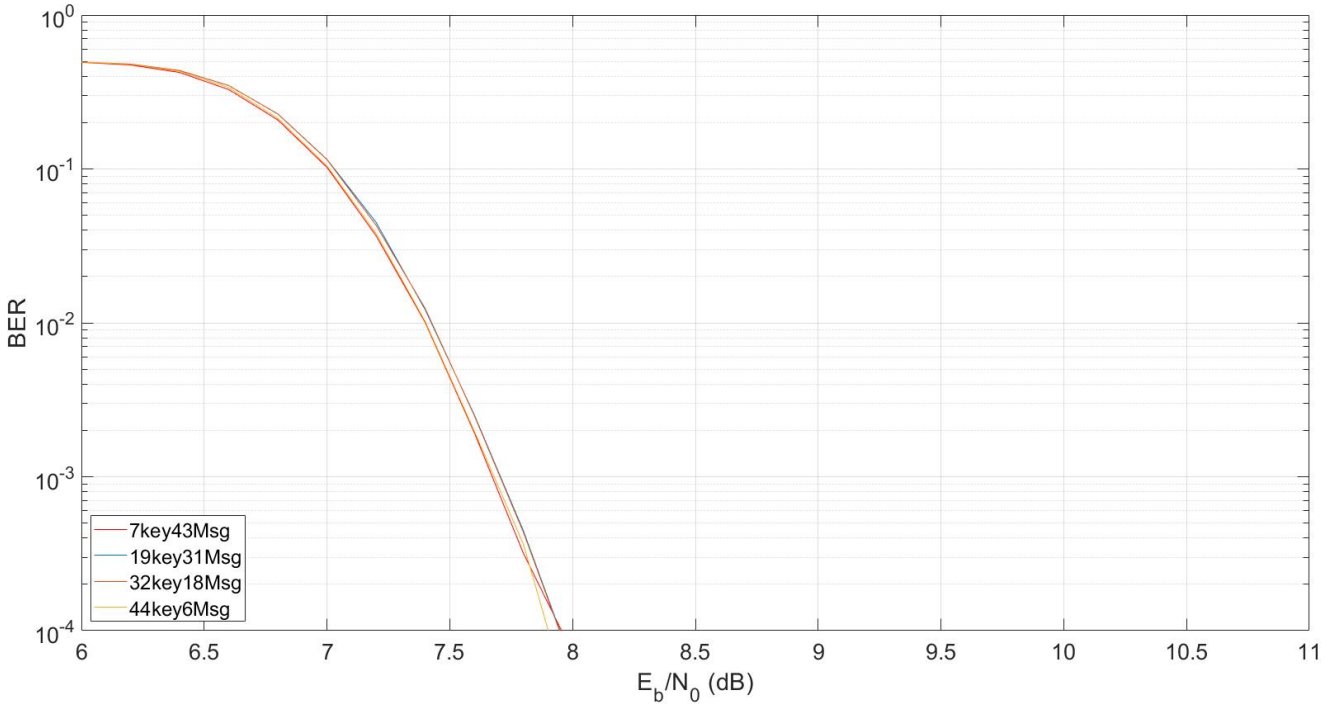
[25] “Short Block Length LDPC Codes for TC Synchronization and Channel Coding, Experimental Specification, Issue 1”, CCSDS 231.1-O-1, April 2015

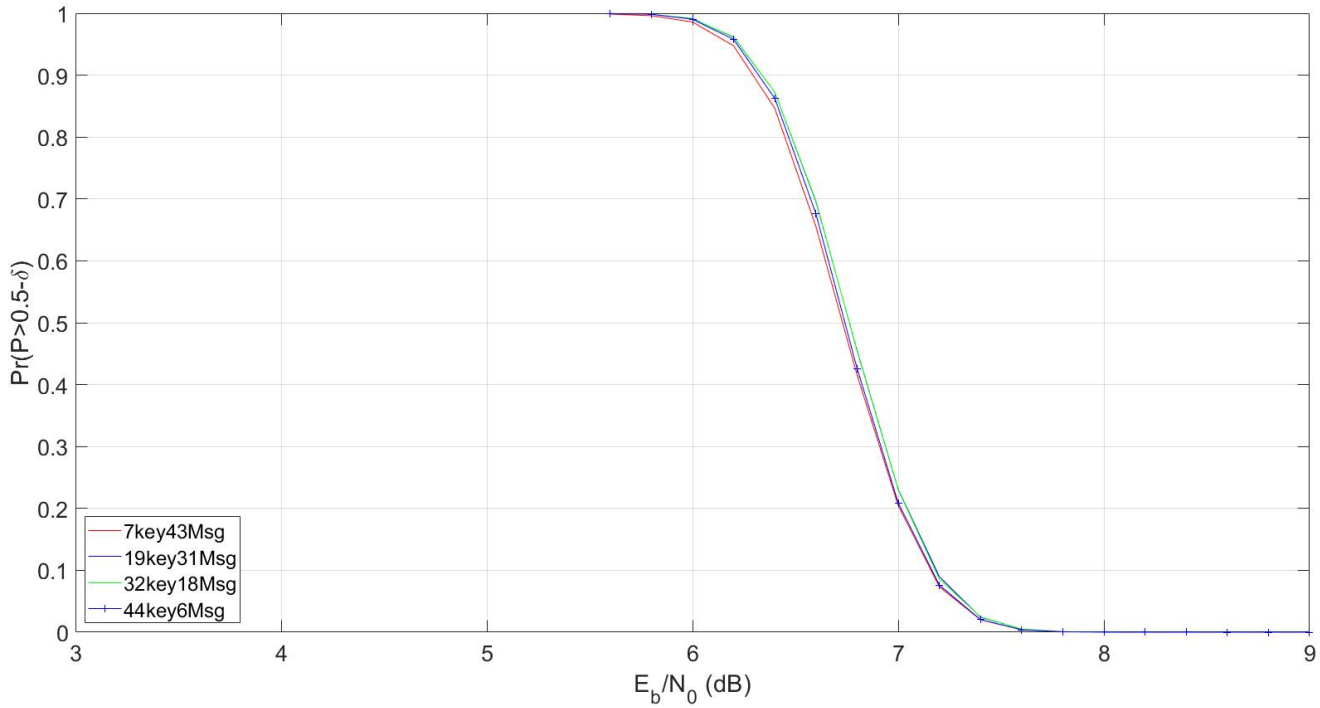
[26] M.Bloch and J. Barros, *Physical Layer Security: From Information Theory to Security Engineering*. Cambridge University Press, 2011.

[27] Harrison, W. K., J. Almeida, M. R. Bloch, S. W. McLaughlin, and J. Barros, "Coding for Secrecy: An Overview of Error-Control Coding Techniques for Physical-Layer Security", {IEEE} {S}ignal {P}rocessing {M}agazine, vol. 30, no. 5, pp. 41–50, September, 2013.

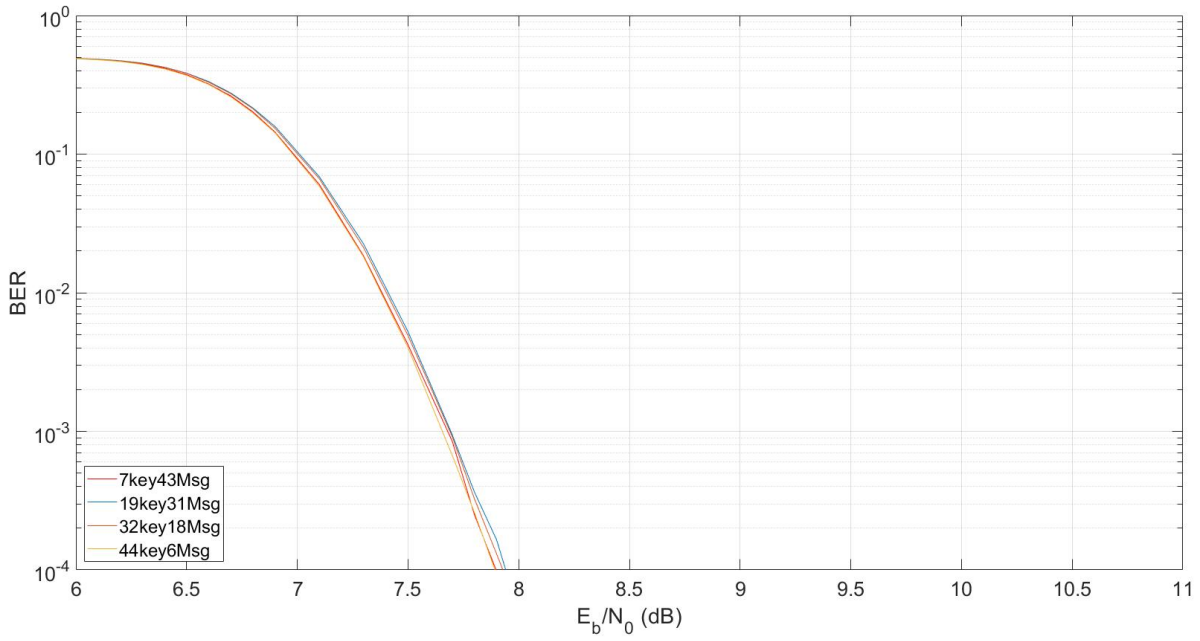
ANEXO

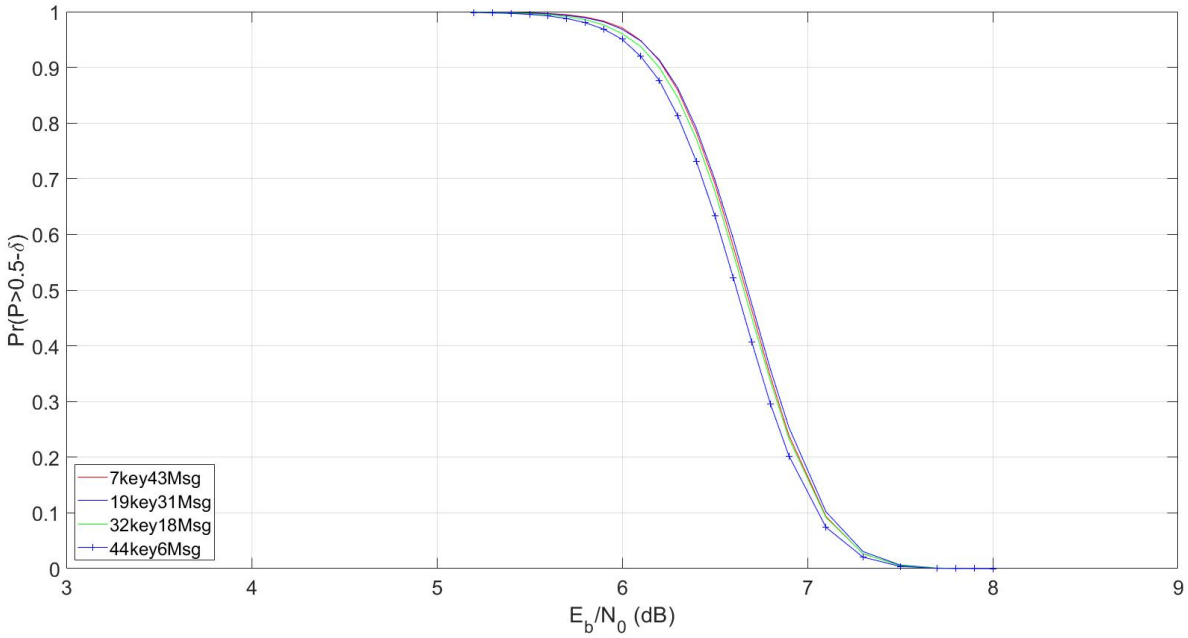
- ICSHK Charts (K=100 e Perfuração=50)



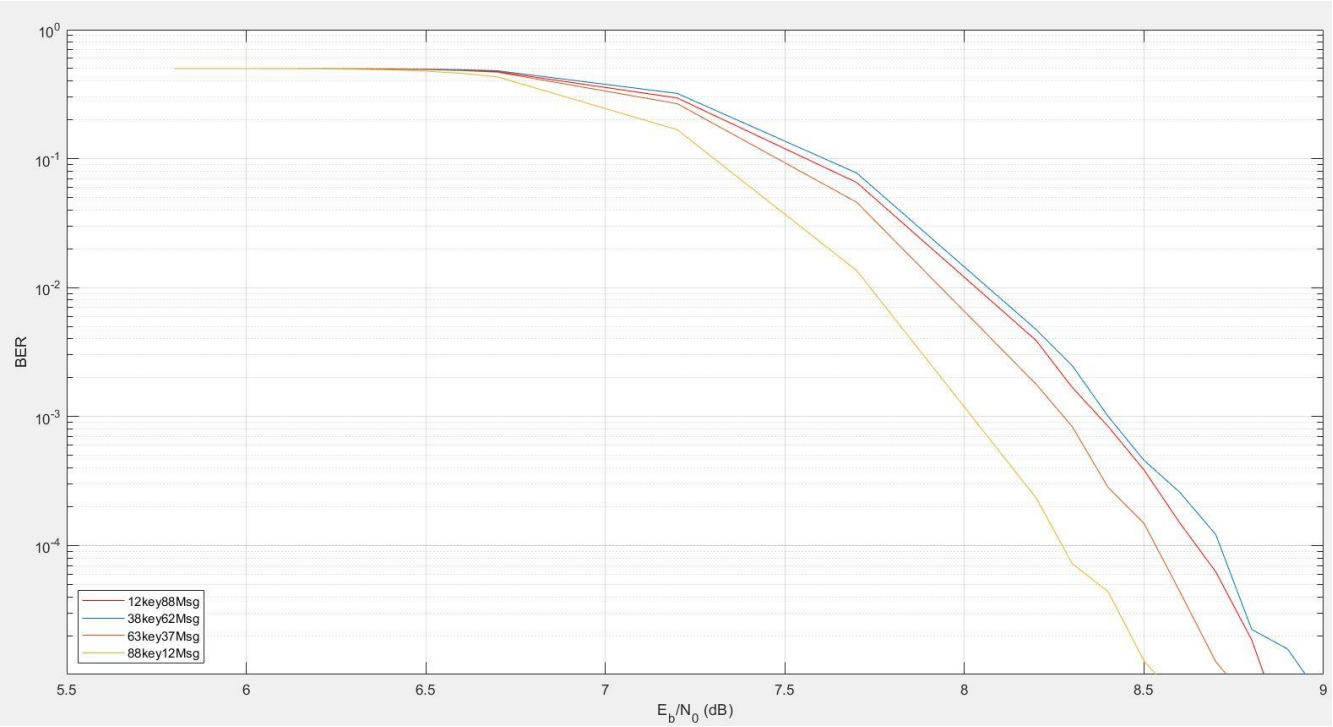


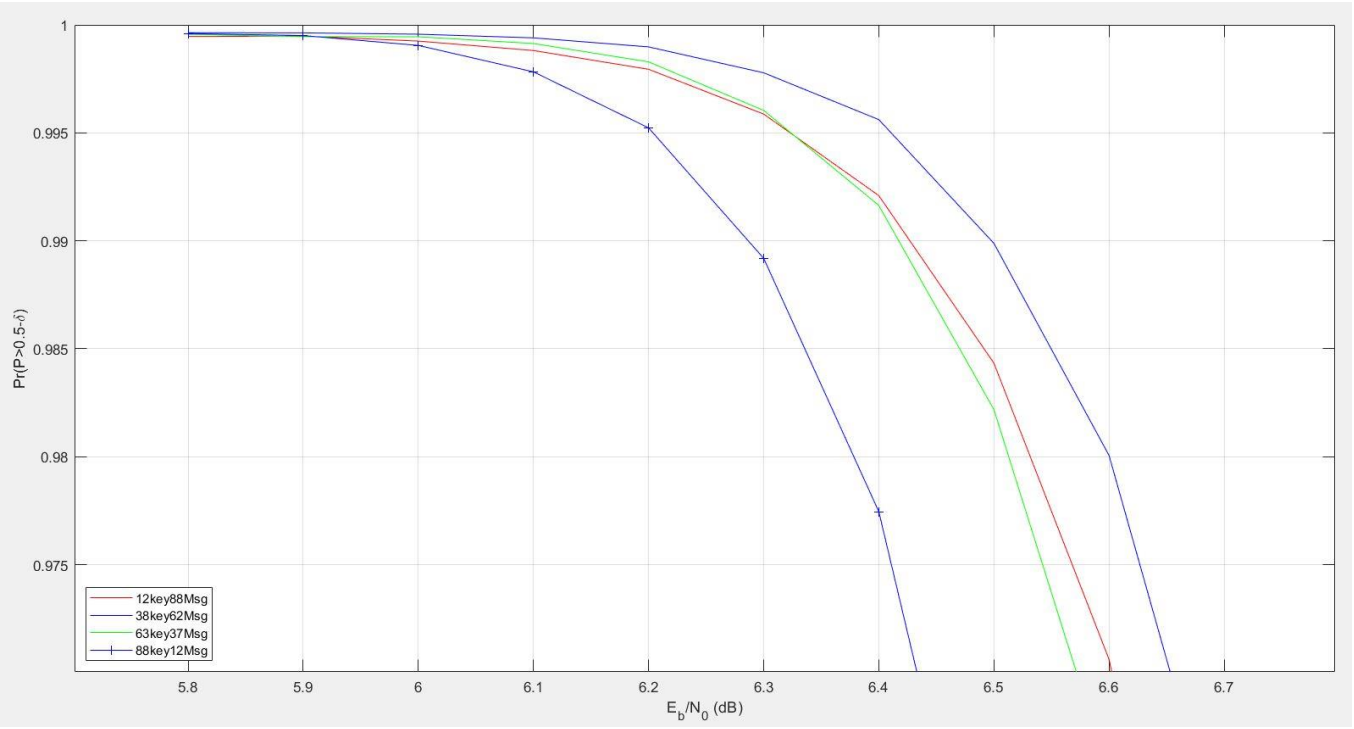
- SCSHK Charts (K=100 e Perfuração=50)



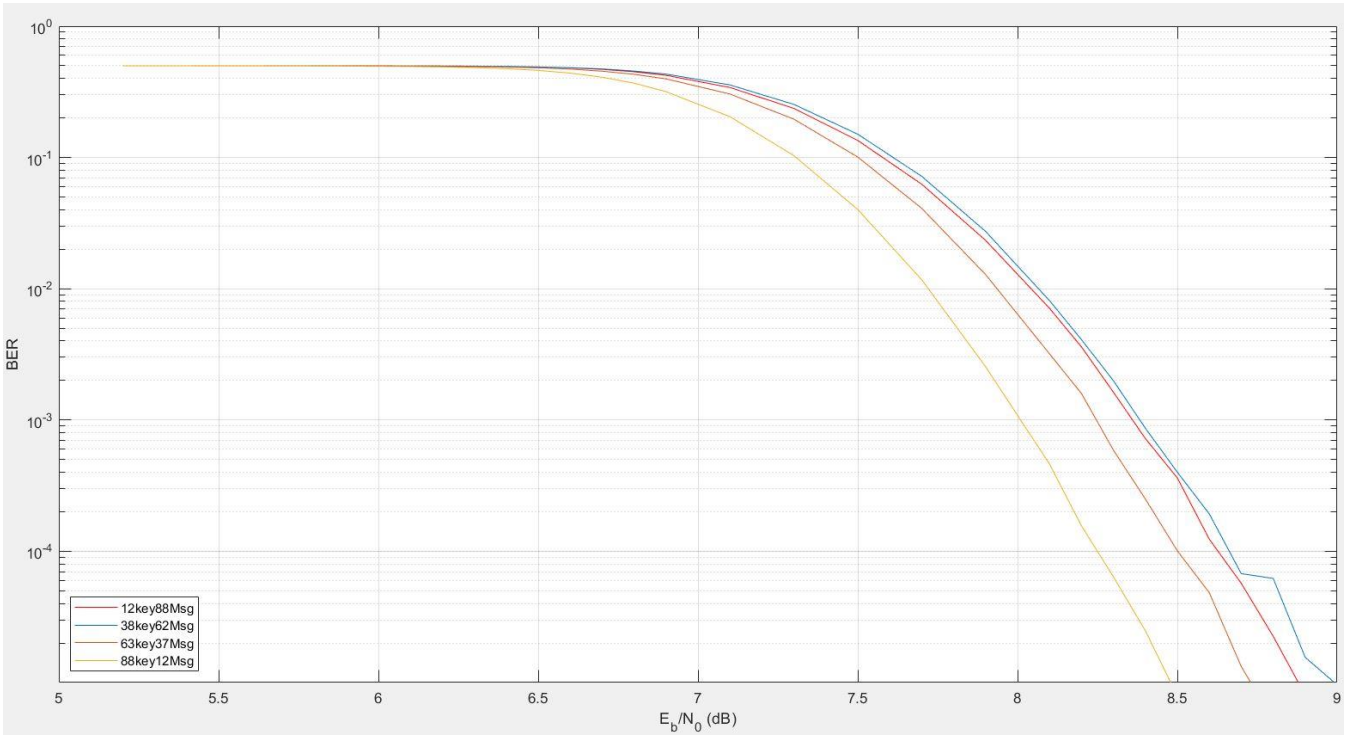


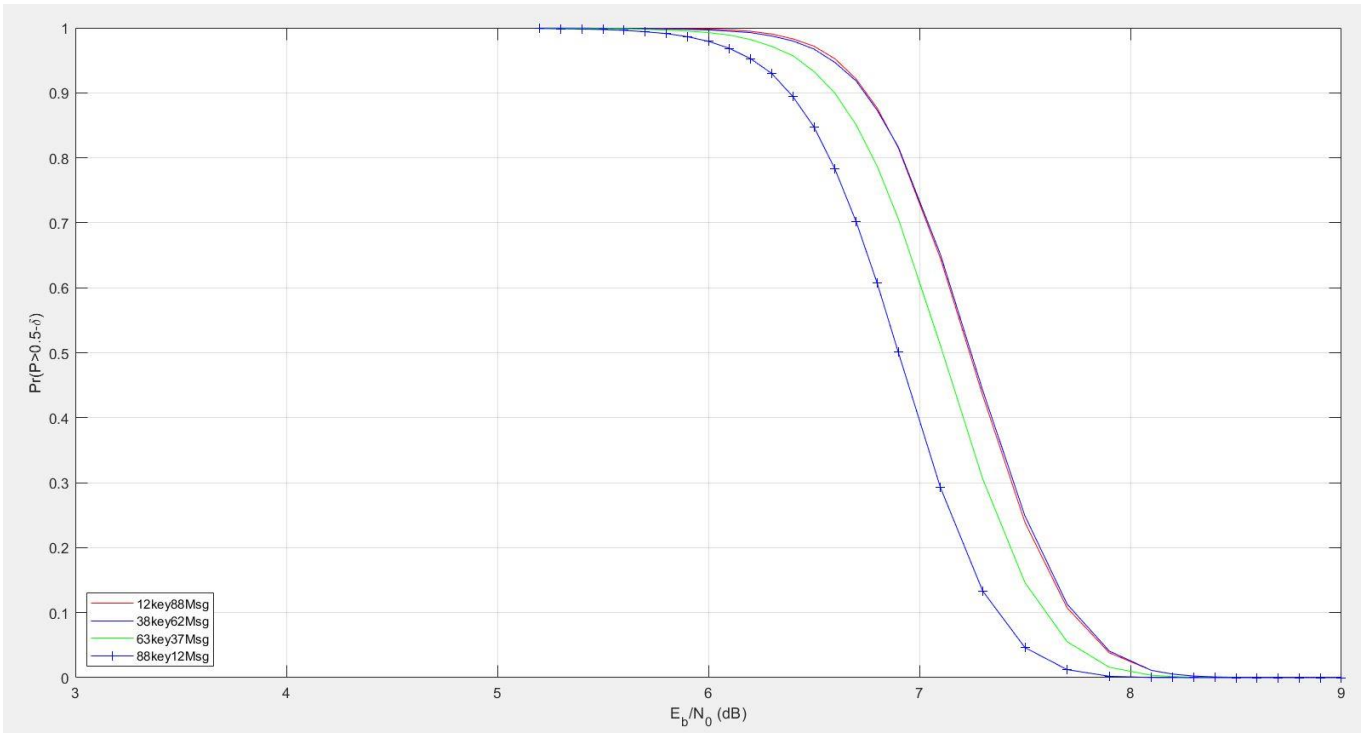
- ICSHK Charts (K=100 e Perfuração=100)



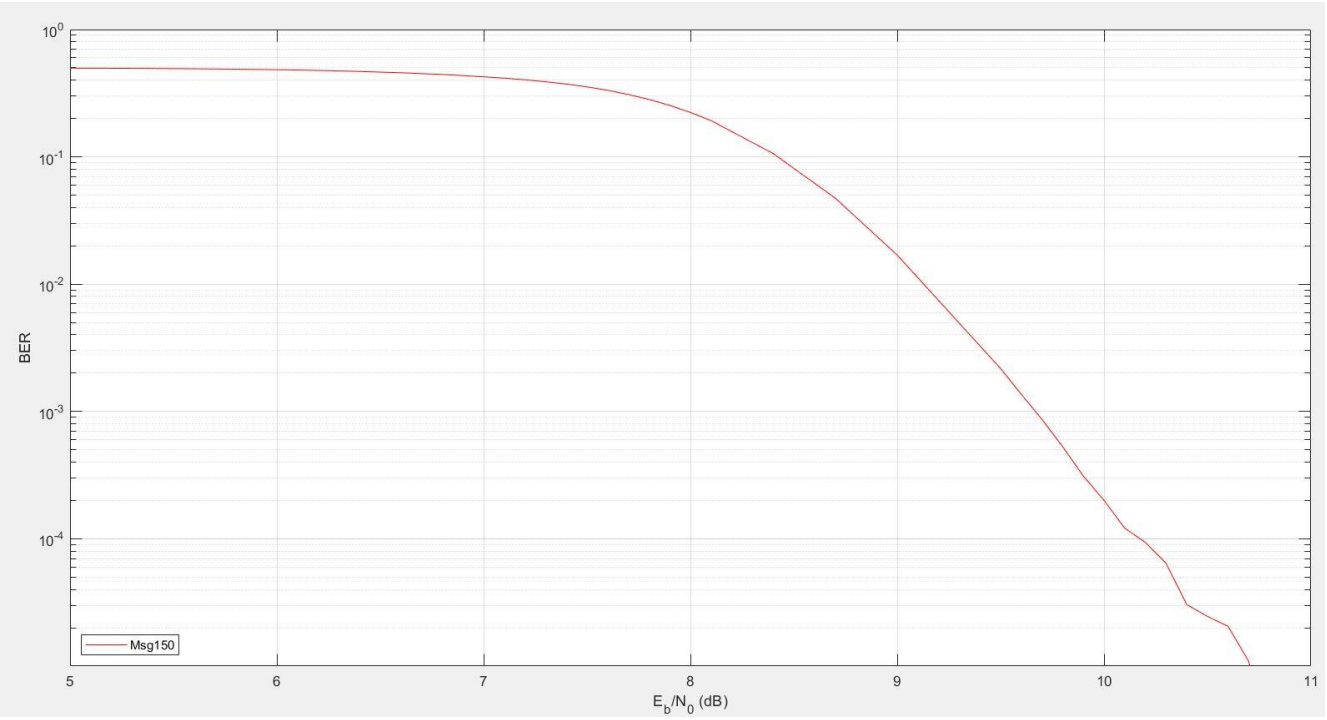


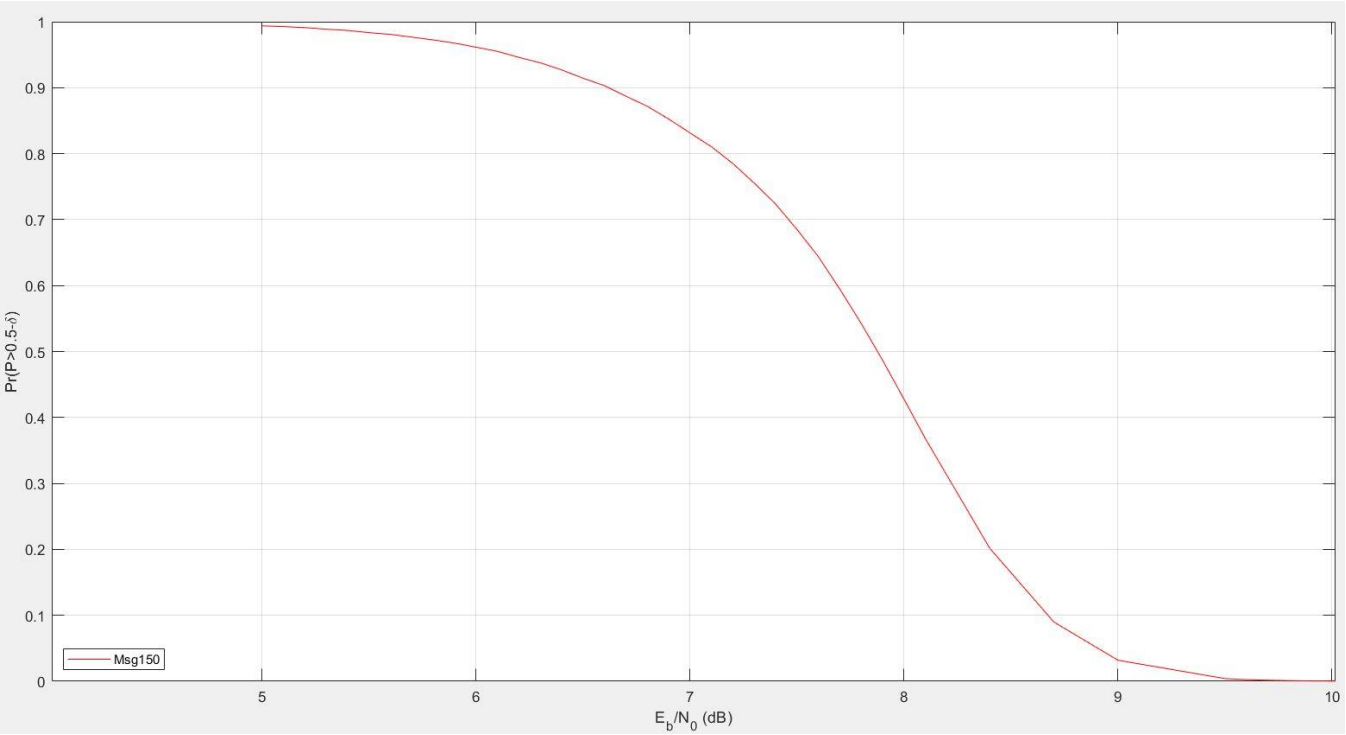
- SCSHK Charts (K=100 e Perfuração=100)





- ICSHK Charts (K=100 e Perfuração=150)





- SCSHK Charts (K=100 e Perfuração=150)