

UNIVERSIDADE ESTADUAL PAULISTA
"JÚLIO DE MESQUITA FILHO"
CAMPUS DE SÃO JOÃO DA BOA VISTA

VANESSA BEATRIZ MARTÃO

Codificação de canal para redes 5G utilizando códigos polares

São João da Boa Vista

2018

Vanessa Beatriz Martão

Codificação de canal para redes 5G utilizando códigos polares

Trabalho de Graduação apresentado ao Conselho de Curso de Graduação em Engenharia de Telecomunicações do Campus de São João da Boa Vista, Universidade Estadual Paulista, como parte dos requisitos para obtenção do diploma de Graduação em Engenharia de Telecomunicações .

Orientador: Profa^o Dra. Cintya Wink de Oliveira Benedito

São João da Boa Vista

2018

Martão, Vanessa Beatriz

Codificação de canal para redes 5G utilizando códigos polares / Vanessa Beatriz Martão. -- São João da Boa Vista, 2018.

61 p. : il. color.

Trabalho de Conclusão de Curso – Câmpus Experimental de São João da Boa Vista – Universidade Estadual Paulista “Júlio de Mesquita Filho”.

Orientador: Profa. Dra. Cintya Wink de Oliveira Benedito

Bibliografia

1. Códigos de controle de erros (Teoria da informação) 2. Sistemas de comunicação sem fio 3. Telecomunicações 4. Teoria da informação

CDD 23. ed. – 621.382

Ficha catalográfica elaborada pela [Biblioteca-BJB](#)

Bibliotecário responsável: João Pedro Alves Cardoso – CRB-8/9717

UNIVERSIDADE ESTADUAL PAULISTA
“JÚLIO DE MESQUITA FILHO”
CÂMPUS EXPERIMENTAL DE SÃO JOÃO DA BOA VISTA
GRADUAÇÃO EM ENGENHARIA DE TELECOMUNICAÇÕES

TRABALHO DE CONCLUSÃO DE CURSO

CODIFICAÇÃO DE CANAL PARA REDES 5G UTILIZANDO CÓDIGOS POLARES

Aluno: Vanessa Beatriz Martão

Orientador: Prof^ª. Dr^ª. Cintya Wink de Oliveira Benedito

Banca Examinadora:

- Prof^ª. Dr^ª. Cintya Wink de Oliveira Benedito
- Prof. Dr. Carlos Hércules Moraes de Lima
- Prof. Dr. Edgar Eduardo Benitez Olivo

A ata da defesa com as respectivas assinaturas dos membros encontra-se no prontuário do aluno (Expediente nº 02/2018)

São João da Boa Vista, 08 de fevereiro de 2018

À minha mãe, Sirlei Rafael Martão,
e ao meu pai, Emerson Rogério Martão,
que sempre foram minha maior fonte de inspiração e força,
sou grata por acreditarem e apoiarem meu sonho.
Ao Leonardo Diogo Bueno Bobadilla,
obrigada por ser tão companheiro e pelo incentivo nas hora difíceis.

AGRADECIMENTOS

Agradeço à UNESP, por me proporcionar um ambiente amigável para os estudos. Sou grata à cada membro do corpo docente, principalmente à professora Cintya Wink de Oliveira Benedito, que fez toda a diferença nesse ano, por todo apoio, paciência e companheirismo nesses poucos meses de muito trabalho. Agradeço também à direção e a administração dessa instituição de ensino.

*“Não importa o que aconteça, continue a nadar”
(Walters, Graham; Procurando Nemo, 2003)*

RESUMO

Códigos Polares são códigos de bloco lineares com baixo custo computacional que tem grande potencial para serem utilizados na tecnologia 5G, devido a comprovação de que atingem o limite da capacidade de canal de Shannon. Esses códigos são essencialmente binários, sendo implementados em canais simétricos e sem memória. O objetivo desse trabalho foi o estudo das estratégias de codificação e decodificação dos códigos polares visando aplicação nas redes 5G. Para esse fim, fizemos uma introdução dos elementos da teoria da informação e codificação. Após isto fizemos um estudo da técnica de polarização do canal que separa os canais bons dos ruins. Foram estudados estratégias de codificação e decodificação para esses códigos, e feito uma implementação computacional para verificação de sua eficiência. Os resultados mostraram que conforme o tamanho N da palavra-código é aumentada, mais próximo do limite dado pelo teorema da capacidade de canal de Shannon os sinais serão.

PALAVRAS-CHAVE: Teoria da Informação. Códigos Polares. Redes 5G.

ABSTRACT

Polar codes are linear block codes with low computational cost that have great potential to be used in 5G technology, due to evidence that it achieves the Shannon's channel capacity. These codes are essentially binary, being implemented in symmetric and memoryless channels. The objective of this work was study the coding and decoding strategy of the polar codes for the application in the 5G networks. To this end, we have introduced the elements of information theory and coding. After that we did a study about the channel polarization technique that separates the good channels from the bad ones. We studied coding and decoding strategies for codes, and a computational implementation to verify its efficiency. The results showed that the larger the N size of the codeword, the closer the signal is to the limit of the Shannon's channel capacity theorem.

KEYWORDS: Information Theory. Polar Codes. 5G Networks.

LISTA DE ILUSTRAÇÕES

Figura 1	Entropia.	16
Figura 2	Canal Ideal.	17
Figura 3	Canal Binário Simétrico.	18
Figura 4	Canal Binário de Apagamento	18
Figura 5	Canal Gaussiano.	19
Figura 6	Código de Bloco Linear.	22
Figura 7	Polarização de Canal.	28
Figura 8	W^N : N cópias do canal W	29
Figura 9	Canal Binário Simétrico	30
Figura 10	Canal Binário de Apagamento	31
Figura 11	Canal W	31
Figura 12	Construção do Canal W_2	32
Figura 13	Construção de W_4	33
Figura 14	Construção do canal W_8	34
Figura 15	Construção de W_N a partir de duas cópias de $W_{N/2}$	36
Figura 16	$I(W_1)$	38
Figura 17	$I(W_2)$	39
Figura 18	$I(W_4)$	39
Figura 19	$I(W_8)$	40
Figura 20	Gráfico canais polarizados.	41
Figura 21	Construção de W_N alternativa.	46
Figura 22	Circuito para implementação da transformação $F^{\otimes 3}$	48
Figura 23	Codificação u'	50
Figura 24	Codificação u''	50
Figura 25	Codificação com $N = 2$	51
Figura 26	Arquitetura Decodificador SC $N = 8$	54
Figura 27	Decodificador SC, $N = 8$	55
Figura 28	Decodificador SC $N = 8$ após todos os passos.	56
Figura 29	Diagrama de Blocos Codificação/Decodificação.	56
Figura 30	Gráfico BER por E_b/N_0	58
Figura 31	Gráfico BLER por E_b/N_0 para diferentes codificações.	59

LISTA DE ABREVIATURAS E SIGLAS

1G	Primeira Geração das Comunicações Móveis
2G	Segunda Geração das Comunicações Móveis
3G	Terceira Geração das Comunicações Móveis
4G	Quarta Geração das Comunicações Móveis
5G	Quinta Geração das Comunicações Móveis
Gbps	Gigabits por segundo
ITU	International Telecommunication Union (União Internacional de Telecomunicações)
GSM	Global System for Mobile Communications (Sistema Global para Comunicações Móveis)
TDMA	Time Division Multiple Access (Acesso Múltiplo por Divisão no Tempo)
SMS	Short Message Service (Serviço de Mensagens Curtas)
UMTS	Universal Mobile Telecommunications System (Sistema Universal de Telecomunicações Móveis)
WiMAX	Worldwide Interoperability for Microwave Access (Interoperabilidade Mundial para Acesso de Micro-Ondas)
LTE-A	Long Term Evolution - Advanced (Evolução a longo prazo - Avançada)
3GPP	3rd Generation Partnership Project (Projeto de Parceria de 3ª Geração)
eMBB	Enhanced Mobile Broadband (Banda Larga Móvel Melhorada)
mMTC	massive Machine Type Communications (Conexões Massivas de Comunicação entre Máquinas)
URLLC	Ultra-Reliable Low Latency Communication (Comunicações Ultra Confiáveis e de Baixa Latência)
LDPC	Low-Density Parity-Check Codes (Códigos de Verificação de Erros de Paridade de Baixa Densidade)
SNR	Signal-to-Noise Ratio (Relação Sinal-Ruído)
BER	Bit Error Rate (Taxa de Erro de Bit)

BSC	Binnary Simmetric Channel (Canal Binário Simétrico)
BEC	Binary Erasure Channel (Canal Binário com Apagamento)
AWGN	Additive White Gaussian Noise (Ruído Gaussiano Aditivo Branco)
V.A.	Variável Aleatória
SC	Successive Cancellation (Cancelamento Sucessivo)
WCDMA	Wide-Band Code-Division Multiple Access (Acesso Múltiplo por Divisão de Código de Banda Larga)
BP	Belief Propagation (Propagação de Crenças)
CRC	Cyclic Redundancy Check (Verificação de redundância Cíclica)
CA-SCL	Cyclic Redundancy Check Successive Cancellation List (Verificação de Redundância Cíclica na Lista de Cancelamento Sucessivo)
aCA-SCL	Cyclic Redundancy Check aided Successive Cancellation List (Verificação de Redundância Cíclica Auxiliar na Lista de Cancelamento Sucessivo)

SUMÁRIO

1	INTRODUÇÃO	13
2	REVISÃO DE CONCEITOS	15
2.1	Teoria da Informação	15
2.1.1	Capacidade do canal	17
2.2	Códigos de bloco lineares	20
3	POLARIZAÇÃO DE CANAL	28
3.1	Conceitos Preliminares	28
3.2	Combinação de Canal	31
3.3	Divisão de Canal	37
3.4	Exemplo de Polarização de Canal para BEC	38
4	CODIGOS POLARES	42
4.1	Codificação dos Códigos Polares	42
4.1.1	Expressões Algébricas	44
4.1.2	Exemplo Codificação	50
4.2	Decodificação dos Códigos Polares por SC	51
4.2.1	Decodificador de tamanho 2	51
4.2.2	Decodificador de tamanho N	53
4.2.3	Passos para decodificação SC	54
4.2.4	Exemplo decodificação SC para N=8	55
4.3	Implementação Computacional	56
5	CONCLUSÃO	60
	REFERÊNCIAS	61

1 INTRODUÇÃO

Durante as décadas de 1970 e 1980 foi desenvolvida e colocada em uso a primeira geração das comunicações móveis (1G), tais sistemas essencialmente analógicos, foram projetados para trafegar somente voz e tinham como principais desvantagens interfaces não padronizadas, baixa capacidade, baixa qualidade nas ligações e baixa segurança na transmissão das informações. (KUMAR; RAO, 2015). Da necessidade de atender à crescente demanda pelo serviço móvel, surgiu em seguida a segunda geração das comunicações móveis (2G), baseado na tecnologia GSM (*Global System for Mobile Communications*), entrando em atividade no início da década de 1990 e sendo responsável pela digitalização da telefonia móvel combinado com a tecnologia de acesso múltiplo por divisão no tempo (*Time Division Multiple Access*, TDMA) permitindo os serviços de troca de mensagens de texto curtas (*Short Message Service*, SMS), essa tecnologia também foi responsável pela padronização na telefonia móvel.

Os avanços tecnológicos culminaram na terceira geração das comunicações móveis (3G) em meados de 2001, proporcionando a primeira grande experiência de banda larga móvel e tendo como tecnologia padrão principal o UMTS (*Universal Mobile Telecommunications System*) e provendo diversas vantagens em comparação a seus antecessores, possuindo cobertura com qualidade superior, maior velocidade de tráfego de dados, suporte a aplicações multimídia e maior imunidade a interferências. A quarta geração das comunicações móveis (4G) foi implementada comercialmente por volta de 2010, tendo como tecnologias padrões o WiMAX (*Worldwide Interoperability for Microwave Access*) e o LTE-A (*Long Term Evolution - Advanced*), sendo esta segunda a adotada no Brasil e oferecendo um serviço de fácil compatibilidade com redes utilizadas anteriormente, maior velocidade, maior largura de banda, menor latência, melhor cobertura e maior qualidade de rede em comparação às gerações anteriores. É previsto a implementação da quinta geração (5G) da telefonia móvel a partir de 2020 (SHAFI et al., 2017), essa nova geração necessitará de novas tecnologias para atender a alta demanda de dispositivos utilizando a rede, pois com a internet das coisas (*Internet of Things*, IoT), haverá um crescimento exponencial na utilização de dados e um grande aumento em dispositivos que utilizarão essa rede.

No ano de 2016 a empresa chinesa Huawei conseguiu uma velocidade de download de 27 Gbps (Gigabits por segundo) utilizando os códigos polares em uma simulação do 5G (HUAWEI, 2016). Nessa simulação foi demonstrado que a tecnologia dos códigos polares é capaz de atingir simultaneamente os três casos típicos definidos como padrão do 5G pela ITU (*International Telecommunication Union*) como eMBB (*Enhanced Mobile Broadband*) acima de 20 Gbps, uRLLC (*Ultra-Reliable Low Latency Communication*) com latência de 1 ms e mMTC (*massive Machine Type Communications*) com bilhões de conexões. Nesse mesmo ano, 3GPP (*3rd Generation Partnership Project*) padronizou os códigos polares como código dominante para controle das funções de canal no cenário eMBB e é possível que os códigos polares possam atuar nos cenários mMTC e uRLLC também pois ainda está em decisão sobre qual irão adotar (ISCAN; LENTNER; XU, 2016). Códigos turbo não poderão ser utilizados nesses cenários pois para uma comunicação confiável não é aceitável possuir um patamar de ruído,

no qual os códigos turbo possuem. Códigos LDPC (*Low-Density Parity-Check Codes*) têm uma performance inferior para comprimentos de blocos menores que 400 bits e para taxas de códigos menores que 1/3, que são as características para URLLC e mMTC (SHARMA; SALIM, 2017).

Códigos Polares são tidos como promissores para os casos URLLC e mMTC, pois oferecem uma excelente performance com variedade nas taxas de códigos e comprimentos de códigos por meio de simples punçionamentos e encurtamento de código, respectivamente (WANG; LIU, 2014). Devido à ausência de chão de erro, códigos polares podem suportar 99,999% de confiabilidade, que é necessário para requerimentos de aplicações ultra confiáveis. Usando uma simples codificação e um algoritmo de decodificação baseado em cancelamento sucessivo diminui o consumo de energia em 20 vezes que os códigos turbo, para uma mesma complexidade (ISCAN; LENTNER; XU, 2016). Portanto aumentando a vida útil de baterias para aplicações na internet das coisas, que necessitam de um consumo muito baixo de energia. Além disso, códigos polares tem menores requerimentos de SNR (*Signal-to-Noise Ratio*) do que os outros para taxas de códigos equivalentes, sendo assim provê alto ganho de codificação e um aumento na eficiência espectral. Esses códigos podem atingir altas taxas de transferência e taxa de erro de bit (*Bit Error Rate*, BER) melhorada em relação à tecnologias anteriores e com todas essas características os códigos polares são atrativos para muitos cenários no 5G (HUAWEI, 2016).

A Codificação Polar é um tipo de codificação que foi desenvolvida em 2009 pelo professor de Engenharia Elétrica e Eletrônica, Erdal Arıkan (ARIKAN, 2009), essa codificação é a primeira no qual foi comprovado que atinge a capacidade simétrica em canais binários discretos e sem memória (*Binary-Input Discrete Memoryless Channel*, B-DMC), essa capacidade atingida é conhecida como capacidade de Shannon (SHANNON, 1948). Isso é um grande feito, dado que essa codificação possui um baixo custo computacional para codificação e decodificação. Sua principal estratégia é fazer uma polarização do canal com o intuito de transmitir bits de informação em canais sem ruído e bits fixos (congelados) em canais ruidosos.

O objetivo central deste trabalho é a apresentação de estratégias de codificação e decodificação de códigos polares utilizando a técnica de polarização de canal. Para a codificação será feita uma construção indutiva utilizando a combinação de canais que será apresentada na polarização de canal e também através de expressões algébricas utilizando o produto de Kronecker de matrizes. A estratégia de decodificação será dada utilizando a técnica de cancelamento de sucessivo (*successive cancellation*, SC). Uma simulação computacional das estratégias de codificação e decodificação também serão apresentadas neste trabalho.

Nesse trabalho serão abordados os seguintes assuntos: o Capítulo 2 fará uma breve revisão de conceitos referentes à teoria da informação e codificação, com foco nos códigos de bloco lineares, estes sendo necessários para o entendimento da codificação utilizada no trabalho. No Capítulo 3 será abordada a técnica de polarização do canal, exemplificando sua implementação para canais binários com apagamento. O Capítulo 4 terá seu foco nas técnicas de codificação e decodificação do canal, explicando equações, operações com matrizes utilizadas na codificação e demonstrando as equações necessárias para a implementação da decodificação, exemplificando como esta será feita. Ainda no Capítulo 4, será apresentada a demonstração de resultados referentes à simulação computacional realizada. E por fim, no Capítulo 5, será feita uma breve conclusão desse trabalho.

2 REVISÃO DE CONCEITOS

Este capítulo será dedicado ao estudo de conceitos básicos da teoria da informação e codificação, que serão necessários para o desenvolvimento do trabalho. Na Seção 2.1 apresentamos os conceitos de entropia e informação mútua para podermos definir a capacidade de um canal, e exemplificamos estes conceitos através de alguns canais importantes como o canal binário simétrico (BSC), o canal binário com apagamento (BEC) e o canal gaussiano (AWGN). Já na Seção 2.2, focamos no estudo dos códigos de bloco lineares, iniciando com algumas definições básicas matemáticas como definição de corpo, de espaço vetorial e corpo finito. Feito isso, apresentaremos o peso de Hamming, matriz geradora e um breve exemplo de aplicação desses conhecimentos em um código de Hamming. Para um estudo mais aprofundado recomenda-se a leitura da bibliografia : (HAYKIN, 2001), (COVER, 2012), (RYAN; LIN, 2009) e (HEFEZ, 2008).

2.1 TEORIA DA INFORMAÇÃO

O estudo da teoria da informação envolve os limites fundamentais no desempenho de um sistema de comunicação, através da especificação do número mínimo de bits por símbolo necessário para representar completamente a fonte, e da especificação da taxa máxima, chamada capacidade do canal, à qual a transmissão de informação pode ocorrer através do canal.

De forma mais precisa, informação é a quantificação da incerteza de um processo em termos probabilísticos, essa incerteza é uma surpresa na ocorrência de um evento inesperado, sendo assim gerando informação. Matematicamente, um evento $x = x_k$ com probabilidade $p(x_k)$ tem a quantificação da informação relacionada com o inverso da probabilidade de ocorrência.

Definição 2.1.1 *Definimos a quantidade de informação obtida após observarmos um evento $x = x_k$ com probabilidade $p(x_k)$, como a função logarítmica*

$$I[x_k] = \log_2 \left[\frac{1}{p(x_k)} \right] = -\log_2 p(x_k). \quad (2.1)$$

A média desta medida de informação é dada por:

$$E[I(x_k)] = -\sum_{i=1}^k p(x_i) \log_2 [p(x_i)]. \quad (2.2)$$

Sendo esta a medida de informação média por símbolo. Assim, podemos definir a entropia como esta informação média associada às observações relativas à variável aleatória (V.A.), ou seja, a entropia é uma medida de incerteza média associada à variável.

Definição 2.1.2 *Seja X uma fonte discreta e sem memória, com função massa de probabilidade:*

$$p(x) = P[X = x]. \quad (2.3)$$

A Entropia de X é definida por:

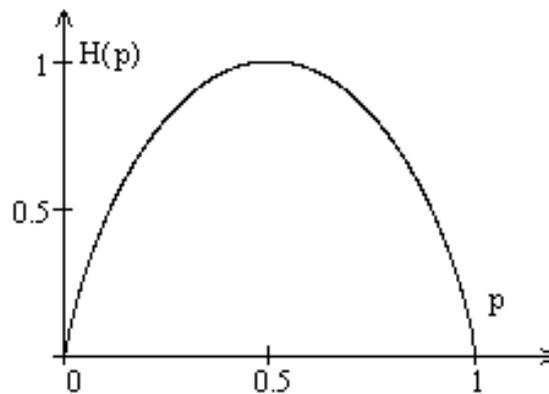
$$H(X) = - \sum p(x) \log[p(x)]. \quad (2.4)$$

Exemplo 2.1.1 Com uma V.A. assumindo dois valores: $x = a$ e $x = b$. Para $P[x = a] = p$ e $P[x = b] = 1 - p$ o valor de entropia será de:

$$H(p) = -p \log_2(p) - (1 - p) \log_2(1 - p). \quad (2.5)$$

Para valores de $p=0,5$, ou seja, que cada valor possua a mesma probabilidade de ocorrência do outro, $H(x) = 1$ bit, sendo assim a informação média de observação será 1 bit, como ilustra a figura Figura 1.

Figura 1 – Entropia.



fonte: Própria autora.

No entanto, para valores de p diferentes, por exemplo $p = 0,7$, $H(x) = 0,8813$ bit a informação obtida será menor. O fato de um valor ser mais provável que o outro não acarretará em tanta surpresa e então a entropia será cada vez menor. Quanto mais uma probabilidade se aproxima de 1, mais próximo de 0 será a sua entropia.

Informação mútua é uma indicação de quanto a incerteza associada a uma variável se reduziu, por meio de informações trazidas por outra variável.

Definição 2.1.3 Sejam X e Y variáveis aleatórias, a informação mútua associada a esse par de variáveis é dada por:

$$I(X, Y) = \sum_x \sum_y p(x, y) \log_2 \left[\frac{p(x, y)}{p(x)p(y)} \right]. \quad (2.6)$$

Por meio de simplificações obtém-se:

$$I(X, Y) = H(X) - H(X/Y) = H(Y) - H(Y/X). \quad (2.7)$$

As variáveis X e Y precisam ser estatisticamente independentes entre si para haver a redução na incerteza. Para essas variáveis temos que $p(x, y) = p(x)p(y)$ sendo quantizado o grau de dependência estatística entre as variáveis.

É possível a utilização da informação mútua entre a entrada e a saída de um canal para avaliar quão eficiente será o envio dessas mensagens. Caso as informações recebidas sejam independentes das enviadas é possível notar que esse canal é totalmente destrutivo, sendo a informação mútua nula. Porém, se as informações recebidas forem fortemente dependentes das enviadas é possível estabelecer uma conexão.

Definição 2.1.4 Para variáveis aleatórias contínuas a entropia diferencial será definida, podendo assumir valores negativos, pela seguinte fórmula:

$$H(X) = - \int_x p(x) \ln[p(x)] dx. \quad (2.8)$$

A Informação mútua é análoga:

$$I(X, Y) = \iint_{x,y} p(x, y) \cdot \ln \left[\frac{p(x, y)}{p(x)p(y)} \right] dx dy. \quad (2.9)$$

2.1.1 Capacidade do canal

Essa grandeza utiliza-se da informação mútua para indicar a condição de maior dependência (melhor fluxo de informação) entre transmissor e receptor, ou seja, é a quantidade máxima de informação capaz de ser transmitida. Para isso manipula-se as probabilidades de envio.

A capacidade do canal C é dada por:

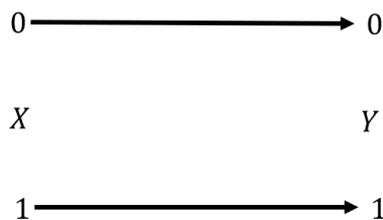
$$C = \max_{p(x)} I(X, Y). \quad (2.10)$$

A capacidade do canal é medida em bits por utilização do canal.

A seguir veremos alguns exemplos importantes de canais apresentando o cálculo de suas capacidades.

Exemplo 2.1.2 Para um canal ideal, no qual não possui erros.

Figura 2 – Canal Ideal.

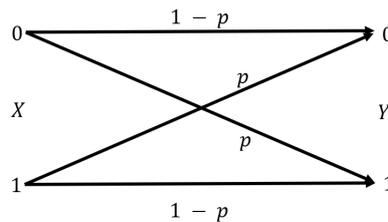


fonte: Própria autora.

Para esse canal $P(Y = 1|X = 1) = P(Y = 0|X = 0) = 1$ e $P(Y = 0|X = 1) = P(Y = 1|X = 0) = 0$, como o canal não gera erros, $P(X = 1) = P(X = 0) = 0,5$, com isso é possível obter 1 bit de informação enviado, sendo essa a capacidade do canal.

Exemplo 2.1.3 O Canal Binário Simétrico (BSC), possui esse nome devido à probabilidade p de receber um bit 1 supondo que foi transmitido 0 ser igual à probabilidade de receber um bit 0 supondo a transmissão de um bit 1. A Figura 3 exemplifica esse canal.

Figura 3 – Canal Binário Simétrico.



fonte: Própria autora.

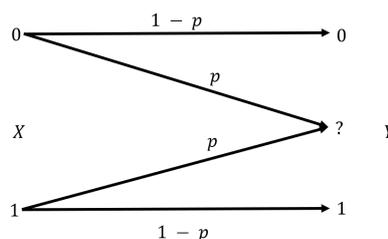
Para esse canal $P(Y = 1|X = 1) = P(Y = 0|X = 0) = 1 - p$ e $P(Y = 0|X = 1) = P(Y = 1|X = 0) = p$. A capacidade desse canal é $C = 1 - H(p)$, com $H(p)$ é a entropia de uma V.A. binária dada no exemplo 2.1.1.

Caso a probabilidade p seja 0, o canal se torna ideal, porém se p for correspondente a 0,5 a capacidade será nula, pois haverá uma incerteza sobre qual bit será transmitido e qual será recebido, nesse caso toda informação transmitida será destruída.

Exemplo 2.1.4 Um Canal Binário de Apagamento (Binary Erasure Channel, BEC) possui esse nome devido a sua condição de que, apesar de um canal de entrada X ter a possibilidade de enviar apenas dois tipos de informações diferentes, sendo estas bits 0 ou 1, durante a recepção, para um canal de saída Y existe a probabilidade de ele receber três tipos de informações diferentes, sendo estas bits 0, bits 1 ou então um alerta, representado como o símbolo ?, informando que não foi recebido nenhum bit, ou seja, o bit foi apagado durante a sua propagação no canal.

Portanto, nesse canal a entrada X é correspondente à $\{0, 1\}$ e a saída Y a $\{0, 1, ?\}$, onde ? corresponde à condição de apagamento no qual não foi recebido um símbolo válido.

Figura 4 – Canal Binário de Apagamento



fonte: Própria autora.

As probabilidades desse canal são:

$$P(Y = 1|X = 1) = P(Y = 0|X = 0) = 1 - p \text{ e } P(Y = ?|X = 0) = P(Y = ?|X = 1) = p$$

A probabilidade p corresponde à probabilidade de ocorrer um apagamento no canal.

O canal BEC é considerado um canal livre de erros, pois ao receber um bit 0 ou 1 tem-se a certeza de que esses foram os bits enviados.

A capacidade desse canal é $C = R(1 - p)$, sendo R a taxa de bits de entrada, nessa capacidade tem-se a certeza de que os bits enviados estão corretos.

Exemplo 2.1.5 Para um Canal Gaussiano, $X(t)$ é um processo estocástico ergódico de média zero e limitado em banda, a sua amostragem uniforme $X(t)$, $i = 1, \dots, k$ gera uma V.A contínua X_i com amostras transmitidas em t segundos por meio de um canal ruidoso.

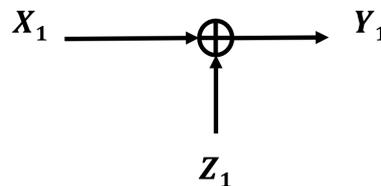
A saída desse canal corresponde a Y_i e possui um ruído aditivo gaussiano branco (Additive White Gaussian Noise, AWGN) com média 0 e variância N , Z_i e N . Portanto,

$$Y_i = X_i + Z_i.$$

$$Z_i \sim \mathcal{N}(0, N).$$

Este canal é designado como gaussiano discreto no tempo e sem memória, com o ruído Z_i independente da entrada X_i

Figura 5 – Canal Gaussiano.



fonte: Própria autora.

Para valores de ruído iguais à zero o sinal de saída é igual ao da entrada, portanto o canal pode transmitir um número real arbitrário sem erro e também a capacidade desse canal é infinita.

Para ruído diferente de zero, porém sem restrições na entrada e com entradas arbitrariamente distantes de modo que a saída consegue distingui-las com uma pequena probabilidade de erro, diz-se que a capacidade do canal também é infinita.

Já para canais com variância do ruído diferente de zero e restrição de potencia, obtém-se :

$$E[X^2] \leq P,$$

$$\sigma^2 = N.$$

Como a entrada do canal e do ruído são independentes, tem-se portanto:

$$E[Y^2] = E[X^2] + E[Z^2] \leq P + N.$$

A capacidade de informação desses canais gaussianos, é:

$$\begin{aligned} C &= \max I(X, Y) \\ &= \frac{1}{2} \log \left(1 + \frac{P}{N} \right). \end{aligned}$$

Sendo que, para um canal limitado em banda B , esta capacidade é:

$$C = B \log \left(1 + \frac{P}{N} \right),$$

Onde, P/N é a razão sinal-ruído geralmente expressa em dB, com P e N correspondentes à Watts ou volts². B é a banda em Hertz e C é a capacidade do canal em bits/segundo.

2.2 CÓDIGOS DE BLOCO LINEARES

Nesta seção serão apresentados alguns conceitos de códigos de bloco lineares, pois como veremos no Capítulo 4 um código polar é um código de bloco linear. Para isso será preciso apresentar algumas definições dadas a seguir.

Definição 2.2.1 Um corpo F é um conjunto que possui duas operações: adição "+" e multiplicação "." satisfazendo as condições:

1. *Associativa:*

$$(a + b) + c = a + (b + c).$$

$$a(bc) = (ab)c.$$

2. *Comutativa:*

$$a + b = b + a.$$

$$ab = ba.$$

3. *Elemento neutro:*

$$\text{Existe } 0 \text{ tal que } a + 0 = a = 0 + a,$$

$$\text{existe também } 1 \text{ tal que } a \cdot 1 = 1 \cdot a = a.$$

4. *Distributiva:*

$$a(b + c) = ab + ac.$$

$$(a + b)c = ac + bc.$$

5. *Elemento imerso:*

$$\text{Existe } -a \text{ tal que } a + (-a) = 0,$$

$$\text{existe também } a^{-1} \text{ tal que } a \cdot a^{-1} = 1, a \neq 0.$$

Exemplo 2.2.1 *Números racionais \mathbb{Q} , números reais \mathbb{R} e números complexos \mathbb{C} são exemplos de corpos. Já o conjunto de números inteiros \mathbb{Z} não é um corpo, pois não possui inverso multiplicativo.*

Um corpo com uma quantidade finita de elementos é chamado de corpo finito. Esses corpos serão utilizados na construção de códigos.

Exemplo 2.2.2 *Para um número primo p . O conjunto $Z_p = \{0, 1, \dots, p - 1\}$ forma um corpo finito.*

- $Z_2 = \{0, 1\}$.
- $Z_3 = \{0, 1, 2\}$.

Definição 2.2.2 *Seja F um corpo e seja V um conjunto com as operações:*

$$\begin{aligned} + : V \times V &\longrightarrow V \\ \cdot : F \times V &\longrightarrow V \end{aligned}$$

V é chamado de espaço vetorial sobre F se as operações satisfazem:

1. $(u + v) + w = u + (v + w), \forall v \in V$.
2. *Existe $0 \in V$ tal que:*
 $v + 0 = 0 + v = v, \forall v \in V$.
3. $u + v = v + u, \forall u, v \in V$
4. *Para cada $v \in V$ existe v' tal que:*
 $v + v' = 0 = v' + v$.
5. $a, b \in F$ e $v \in V$, *então:*
 $a(bv) = (ab)v$.
6. *Existe $1 \in F$ tal que:*
 $1 \cdot v = v \cdot 1 = v, \forall v \in V$.
7. $a \cdot (v + u) = av + au, \forall a \in F$ e $u, v \in V$.
8. $(a + b)v = av + bv, \forall a, b \in F$ e $v \in V$.

Exemplo 2.2.3 $V = \mathbb{R}^2$ é um espaço vetorial sobre o corpo \mathbb{R} :

$$\mathbb{R}^2 = \{(x, y); x, y \in \mathbb{R}\}$$

Com as seguintes operações:

- $(x_1, y_1) + (x_2, y_2) = (x_1 + x_2, y_1 + y_2)$.
- $\alpha(x, y) = (\alpha x, \alpha y)$.

Esse exemplo satisfaz as oito condições.

Exemplo 2.2.4 F_q^n é um espaço vetorial sobre o corpo finito F_q (conjunto de todas as n -uplas de elementos de F_q). Se $x, y \in F_q^n$, então para $x = (x_1, x_2, \dots, x_n), y = (y_1, y_2, \dots, y_n)$.

A partir dessas relações, as operações são:

- $x + y = (x_1, x_2, \dots, x_n) + (y_1, y_2, \dots, y_n) = (x_1 + y_1, \dots, x_n + y_n)$.
- $\alpha x = \alpha(x_1, \dots, x_n) = (\alpha x_1, \dots, \alpha x_n), \alpha \in F_q$.

Definição 2.2.3 Seja F_q um corpo finito com q elementos. Dizemos que \mathcal{C} é um código de bloco linear, de comprimento n e dimensão k sobre F_q se \mathcal{C} for um subespaço vetorial com dimensão k de F_q^n .

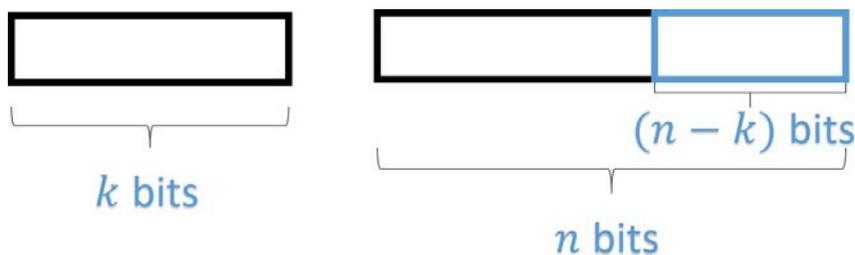
Um código de bloco linear \mathcal{C} pode ser caracterizado pelos parâmetros (n, k) ; Para valores de $v \in \mathcal{C}$, v é uma palavra-código de \mathcal{C} ; Caso \mathcal{C} seja um código sobre F_q , este será considerado alfabeto do código \mathcal{C} ; para valores de $q = 2$ esse código será binário; Para valores de $q = 3$ esse código será ternário e a quantidade de palavras-código de um código $\mathcal{C} = (n, k)$ sobre F_q é q^k .

Exemplo 2.2.5 Para F_2^3 obtém-se um código de bloco linear $\mathcal{C} = (000, 001, 010, 011, 100, 101, 110, 111)$, com parâmetros (n, k) iguais a $(3, 2)$.

Definição 2.2.4 Denomina-se taxa de código a razão $R = \frac{k}{n}$, que é interpretada como o número médio de bits de informação.

O código (n, k) é denominado código de bloco pois na codificação utiliza-se uma sequência de k bits no qual serão adicionados $n - k$ bits de redundância gerando uma nova palavra com n bits de comprimento, sendo $n > k$. Esses códigos de bloco são ilustrados na Figura 6.

Figura 6 – Código de Bloco Linear.



fonte: Própria autora.

Definição 2.2.5 Dados dois vetores u e v , a distância de Hamming d denotada como $d(u, v)$ é obtida por meio do número de coordenadas que cada vetor possui diferente do outro.

Exemplo 2.2.6 $d(001, 111) = 2$

$$d(000, 111) = 3$$

Propriedades:

1. $d(u, v) \geq 0$;
2. $d(u, v) = d(v, u)$;
3. $d(u, v) \leq d(u, w) + d(w, v)$.

Definição 2.2.6 Para códigos de bloco \mathcal{C} a distância mínima de Hamming é caracterizada como:

$$d_H(\mathcal{C}) = \min\{d(u, v) | u, v \in \mathcal{C}/u \neq v\}. \quad (2.11)$$

Para um código $\mathcal{C} = \{0000, 0110, 1001, 1111\}$, as distâncias de Hamming entre cada um dos vetores é:

$$d_H(0000, 0110) = 2$$

$$d_H(0000, 1001) = 2$$

$$d_H(0000, 1111) = 4$$

$$d_H(0110, 1001) = 4$$

$$d_H(0110, 1111) = 2$$

$$d_H(1001, 1111) = 2$$

Ou seja,

$$d_H = \min\{d_H(u, v) | u, v \in \mathcal{C}\} = \min\{2, 4\} = 2.$$

Portanto, compreende-se que a distância mínima de Hamming de um código \mathcal{C} com características $\{2, 4\}$ a distância mínima será de 2 bits.

Definição 2.2.7 O peso de Hamming $w_H(v)$ de um vetor \mathbf{u} é o número de componentes não nulos contidos no vetor. O peso mínimo de Hamming $w_H(\mathcal{C})$ de um código \mathcal{C} é equivalente ao menor peso das palavras-código.

Exemplo 2.2.7 Para o código \mathcal{C} utilizado anteriormente. Obtém-se

$$w(0110) = 2$$

$$w(1001) = 2$$

$$w(1111) = 4$$

Sendo assim,

$$w(\mathcal{C}) = \min\{w(v), v \in \mathcal{C}, v \neq 0\} = 2. \quad (2.12)$$

Teorema 2.2.1 Para um código \mathcal{C} e $u, v \in \mathcal{C}$:

1. $d_H(u, v) = w_H(u - v)$.
2. $d_H(\mathcal{C}) = w_H(\mathcal{C})$.

Para um código de bloco (n, k) com distância mínima de Hamming d , classifica-se \mathcal{C} como um código (n, k, d) .

Exemplo 2.2.8 Utilizando o código \mathcal{C} novamente, $n = 4$, $k = 2$ e $d = 2$. Portanto \mathcal{C} é um código de bloco caracterizado por $(4, 2, 2)$

Para códigos de bloco nos quais a distância mínima de Hamming for correspondente a d , esse código poderá corrigir até $\lfloor \frac{d-1}{2} \rfloor$ erros e detectar até $(d - 1)$ erros.

Em casos de números binários, é possível obter a distância de Hamming utilizando-se da operação mod-2 entre ambos os vetores e avaliando o peso do vetor resultante, $d(u, v) = w(u \oplus v)$.

Exemplo 2.2.9 $d(u, v) = w(u \oplus v) = w(10110 \oplus 10101) = w(00011)$

$$d(u, v) = 2$$

Definição 2.2.8 Seja F_q um corpo finito e $\mathcal{C} \subset F_q^n$ um código de bloco linear com parâmetros (n, k, d) e q^k palavras-código.

Seja $\beta = \{v_1, \dots, v_k\}$ uma base de \mathcal{C} e considere G a matriz cujas linhas são os vetores $v_i = (v_{i1}, \dots, v_{in})$, $i = 1, \dots, k$, ou seja,

$$G = \begin{bmatrix} \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_k \end{bmatrix}, = \begin{bmatrix} v_{11} & \cdots & v_{1n} \\ \vdots & \ddots & \vdots \\ v_{k1} & \cdots & v_{kn} \end{bmatrix}. \quad (2.13)$$

A matriz G é chamada de matriz geradora de \mathcal{C}

Exemplo 2.2.10 A matriz geradora do código $\mathcal{C} = \{0000, 0110, 1001\}$ é:

$$G = \begin{bmatrix} 0110 \\ 1001 \end{bmatrix}.$$

Matrizes geradoras podem ser um codificador natural, pois é possível a construção de códigos de bloco as utilizando.

Exemplo 2.2.11 Para um código $(n, k) = (5, 3)$ em F_2^5 com vetores 10011, 01001, 00111 linearmente independentes. A matriz geradora será:

$$G = \begin{bmatrix} 10011 \\ 01001 \\ 00111 \end{bmatrix}. \quad (2.14)$$

Esse código possui $2^3 = 8$ palavras-código dado em (2.14).

Para construir um código \mathcal{C} em F_2^5 utilizando a matriz geradora G se considera uma transformação linear dada por

$$f : F_2^3 \mapsto F_2^5.$$

$$u \mapsto uG.$$

Sendo $F_2^3 = \{000, 001, 010, 100, 110, 011, 101, 111\}$, em F_2^5 o código obtido corresponde a uG .

Por exemplo:

$$uG = (101) \begin{pmatrix} 10101 \\ 11010 \\ 11111 \end{pmatrix} = (01010).$$

Com (101) e (01010) sendo o código fonte e o código de canal, respectivamente.

A decodificação desse código pode ser feita encontrando valores de u no qual uG serão correspondentes ao código gerado.

Exemplo 2.2.12 $uG = (10101)$

$$(u_1 u_2 u_3) \begin{pmatrix} 10101 \\ 11010 \\ 11111 \end{pmatrix} = (10101).$$

Fazendo a multiplicação entre as matrizes obtém-se:

$$u_1 + u_2 + u_3 = 1$$

$$u_2 + u_3 = 0$$

$$u_1 + u_3 = 1$$

$$u_2 + u_3 = 0$$

$$u_1 + u_3 = 1$$

Portanto, $u_1 = 1, u_2 = 0$ e $u_3 = 0$.

E então, (10101) é decodificada em (100).

Nem sempre essa é a melhor maneira de decodificar um canal, uma melhor alternativa à esse método é utilizar G na sua forma padrão, sendo assim possível a decodificação, apenas observando o início das palavras.

A forma padrão de uma matriz geradora G de um código C possui a seguinte forma:

$$G = (I_k | P). \quad (2.15)$$

No qual I_k é a matriz identidade de ordem k e P uma matriz $k \times (n - k)$.

Para o caso da matriz $G = \begin{pmatrix} 10101 \\ 11010 \\ 11111 \end{pmatrix}$, utiliza-se de algumas operações elementares e permutações

em suas colunas para resultar em sua forma padrão.

Após essas ações a matriz G corresponderá a:

$$G = \left(\begin{array}{ccc|cc} 000 & 00 \\ 010 & 10 \\ 001 & 01 \end{array} \right).$$

Sendo as três primeiras colunas correspondentes à I_3 e as duas últimas à P .

Definição 2.2.9 *Matriz controle de paridade é uma matriz no qual se relaciona com uma mensagem do código para definir se essa mensagem pertence ou não ao código.*

Para uma matriz geradora $G = (I_k|P)$ com código \mathcal{C} com (n, k) . A matriz controle de paridade será $H = (-P^t|I_{n-k})$, sendo P^t a matriz transposta de P . A operação transposição (t) de uma matriz é feita trocando as linhas pelas colunas.

A possibilidade de utilização da matriz H para verificar se uma palavra pertence ou não ao código se dá pelo fato de que a operação de multiplicação entre as matrizes G e H^t deverá ser nula,

$$GH^t = (I_k|P_{K \times (N-K)}) \begin{pmatrix} -P_{k \times (n-k)} \\ I_{(n-k)} \end{pmatrix} = 0.$$

Portanto, se a palavra-código pertencer ao código, ao efetuar sua multiplicação pela matriz H^t o resultado deverá ser nulo.

Exemplo 2.2.13 *Para um código binário $(6, 3)$ com matriz geradora*

$$G = \begin{pmatrix} 100|111 \\ 010|011 \\ 001|010 \end{pmatrix}.$$

A matriz controle de paridade será:

$$H = \begin{pmatrix} 100|100 \\ 111|010 \\ 110|001 \end{pmatrix}.$$

Para códigos $v = (100111)$ e $v' = (010101)$

Verificação para v :

$$vH^t = (100111) \begin{pmatrix} 111 \\ 011 \\ 010 \\ 100 \\ 010 \\ 001 \end{pmatrix} = (000).$$

Verificação para v' :

$$v'H^t = (010101) \begin{pmatrix} 111 \\ 011 \\ 010 \\ 100 \\ 010 \\ 001 \end{pmatrix} = (110).$$

Como é possível observar, a palavra v após a verificação se anulou, portanto pertence ao código. Já a palavra v' retornou uma resposta não nula, portanto ela não faz parte do código. Um exemplo de aplicação de códigos de bloco é o código de Hamming.

Exemplo 2.2.14 *Código de Hamming.*

Um código de Hamming de ordem m sobre \mathbb{F}_2 com matriz teste de paridade H_m , possui ordem $m \times n$, com colunas sendo elementos de $\mathbb{F}_2^m \setminus \{0\}$ utilizando uma ordem qualquer. Sendo assim, o comprimento de um código de Hamming de ordem $n = 2^m - 1$ tem dimensão correspondente a $k = n - m = 2^m - m - 1$. Verifica-se que $d = 3$, pois é visível a existência de três colunas linearmente dependentes. A matriz abaixo exemplificará numericamente:

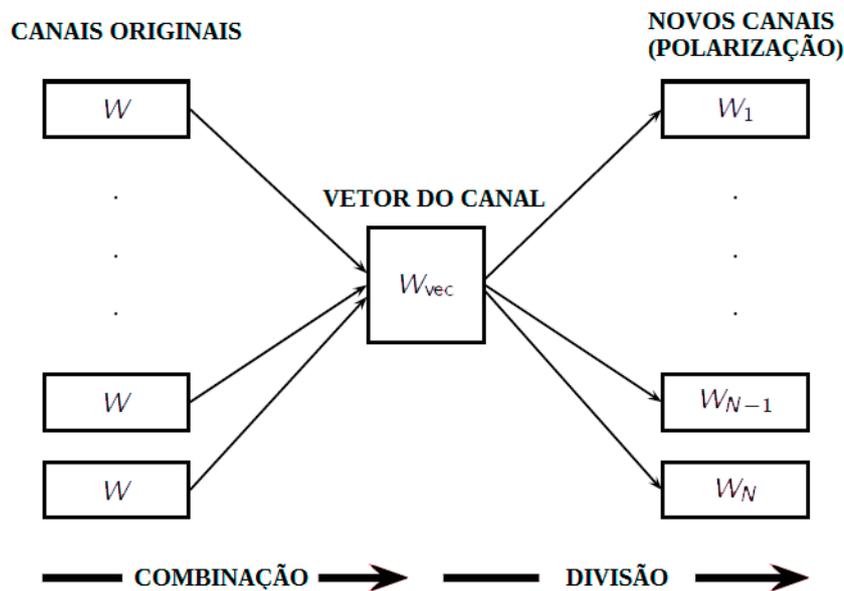
$$H_3 = \begin{pmatrix} 1011100 \\ 1101010 \\ 0111001 \end{pmatrix}.$$

Essa matriz representa um código de Hamming com $m = 3$.

3 POLARIZAÇÃO DE CANAL

Nesse capítulo será demonstrado como é feita a polarização de um canal, um processo que refere-se ao fato de que é possível sintetizar N cópias independentes de um canal W discreto e sem memória com fonte binária (*B-DMC: Binary Discrete Memoryless Channel*), em um segundo conjunto de entradas binárias $\{W_N^{(i)} : 1 \leq i \leq N\}$. Para isso, na Seção 3.1, serão apresentados alguns conceitos preliminares sobre os canais B-DMC e alguns parâmetros. Esse processo de polarização é feito em duas etapas: a primeira consiste na combinação de canais que será demonstrada na Seção 3.2 e a segunda etapa consiste na divisão destes canais separando um canal original em canais bons (sem ruídos) e em canais ruins (ruidosos), por meio da polarização desses canais, essa etapa será exemplificada na Seção 3.3. Os bits de informação que são os bits nos quais não podem sofrer interferência são enviados pelo canal sem ruídos, enquanto que os outros bits são fixados em zero e transmitidos pelos canais ruidosos, esses bits que serão fixados em zero são chamados de bits congelados. Chamamos de polarização o fato da capacidade do canal $I(W)$ convergir para 0 ou 1 quando N tende ao infinito. Na Seção 3.4 será apresentado um exemplo de polarização para um canal BEC. Este capítulo está baseado basicamente nas referências (ARIKAN, 2009) e (NIU et al., 2014).

Figura 7 – Polarização de Canal.



fonte: Própria autora.

3.1 CONCEITOS PRELIMINARES

Seja

$$W : X \rightarrow Y$$

um canal B-DMC no qual utiliza-se um alfabeto de entrada X , um alfabeto de saída Y e probabilidades de transição

$$W(y|x) = P(Y = y|X = x),$$

sendo que o alfabeto de entrada será sempre $X = \{0, 1\}$, o alfabeto de saída e as probabilidades de transição serão arbitrários.

Iremos considerar W^N como sendo o canal resultante de N cópias do canal W , ou seja,

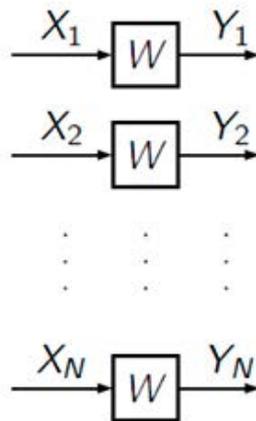
$$W^N : X^N \rightarrow Y^N,$$

com probabilidades de transição dada por

$$W^N(y_1^N|x_1^N) = \prod_{i=1}^N W(y_i|x_i), \quad (3.1)$$

onde $y_1^N = (y_1, \dots, y_N)$ e $x_1^N = (x_1, \dots, x_N)$. A Figura 8 ilustra o canal resultante W^N .

Figura 8 – W^N : N cópias do canal W



fonte: Própria autora.

Para um canal B-DMC, W , existem dois parâmetros de interesse primário, a saber, a capacidade do canal e o parâmetro de Bhattacharyya. Esses parâmetros são utilizados para fazer medidas de taxa e confiabilidade, respectivamente.

A capacidade do canal $I(W)$ é a maior taxa que comunicações confiáveis conseguem alcançar por W , usando entradas de W com frequências iguais, podendo ser calculada por:

$$I(W) = \sum_{y \in Y} \sum_{x \in X} \frac{1}{2} W(y|x) \log \frac{W(y|x)}{\frac{1}{2}W(y|0) + \frac{1}{2}W(y|1)}. \quad (3.2)$$

Já o parâmetro de Bhattacharyya $Z(W)$ é o limite superior da probabilidade de decisão de erro de máxima verossimilhança quando W é utilizado somente para transmitir 0 ou 1. Este parâmetro é

definido por:

$$Z(W) = \sum_{y \in Y} \sqrt{W(y|0)W(y|1)}. \quad (3.3)$$

Temos que $Z(W)$ utiliza valores em $[0, 1]$, utilizaremos logaritmos na base 2 e consequentemente $I(W)$ irá utilizar valores em $[0, 1]$. A unidade das taxas de código e capacidade do canal serão *bits* e a relação entre estes dois parâmetros é dada na Proposição 3.1.1.

Proposição 3.1.1 (ARIKAN, 2009) Para qualquer B-DMC W , temos que:

$$I(W) \geq \log \frac{2}{1 + Z(W)}, \quad (3.4)$$

$$I(W) \leq \sqrt{1 - Z(W)^2}. \quad (3.5)$$

Através da Proposição 3.1.1, podemos concluir que:

- $I(W) \simeq 1$ se, e somente se, $Z(w) \simeq 0$.
- $I(W) \simeq 0$ se, e somente se, $Z(w) \simeq 1$.

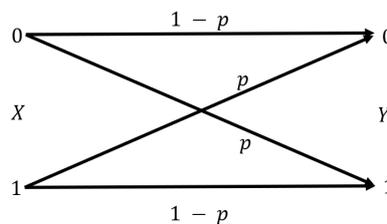
A capacidade $I(W)$ se iguala a capacidade de Shannon quando W é um canal simétrico, isto é, um canal no qual existe uma permutação π na saída do alfabeto Y tal que:

- $\pi^{-1} = \pi$
- $W(y|1) = W(\pi_y|0)$ para todo $y \in Y$.

Como exemplos de canais simétricos temos o canal binário simétrico (*BSC-Binary Symmetric Channel*) e o canal binário com apagamento (*BEC-Binary Erasure Channel*) definidos em 2.1.3 e 2.1.4, os quais relacionamos com a polarização de canal nos exemplos abaixo.

Exemplo 3.1.1 O canal BSC é um B-DMC com $Y = \{0, 1\}$.

Figura 9 – Canal Binário Simétrico



fonte: Própria autora.

Temos que

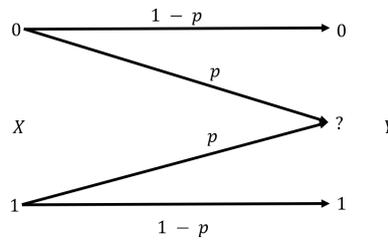
$$W(0|0) = W(1|1) = 1 - p \text{ e } W(1|0) = W(0|1) = p.$$

Exemplo 3.1.2 O canal BEC é um B-DMC se para cada $y \in Y$, tivermos

$$W(y|0)W(y|1) = 0 \text{ ou } W(y|0) = W(y|1),$$

no segundo caso, y é chamado de símbolo de apagamento e a soma de $W(y|0)$ sobre todos os símbolos de apagamento y é chamada de probabilidade de apagamento do BEC.

Figura 10 – Canal Binário de Apagamento



fonte: Própria autora.

A seguir veremos como são feitas as etapas de combinação e divisão do canal utilizadas na polarização do canal.

3.2 COMBINAÇÃO DE CANAL

Nesta etapa, combina-se N cópias de um canal W para produzir o vetor de canal $W_N : X^N \rightarrow Y^N$, sendo N uma potência de 2, ou seja, $N = 2^n$, $n \geq 0$.

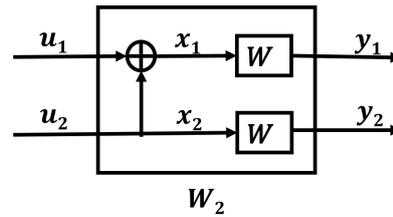
A seguir apresenta-se uma construção indutiva do vetor de canal W_N que é dado por N cópias do canal W . A construção indutiva se caracteriza pela demonstração da construção do canal para $N=1, 2, 4, 8$ e então a construção para $N=2^n$. Inicialmente, para $N=1$, será gerado somente um canal $W = W_1 = W^1$.

Figura 11 – Canal W .



fonte: Própria autora.

Para $N=2$ serão geradas duas cópias de $W = W_1$, obtendo assim o canal $W_2 : X^2 \rightarrow Y^2$ descrito na figura abaixo.

Figura 12 – Construção do Canal W_2 .

fonte: Própria autora.

A partir da construção do canal W_2 apresentada na Figura 12, as probabilidades de transição do canal serão obtidas pela combinação das entradas u_1 e u_2 que geram as saídas y_1 e y_2 , da seguinte forma:

$$W_2(y_1, y_2 | u_1, u_2) = W(y_1 | u_1 \oplus u_2)W(y_2 | u_2). \quad (3.6)$$

Mapeando $u_1^2 \mapsto x_1^2$, onde u_1^2 são as entradas de W_2 e x_1^2 são as entradas de W^2 , observe que esta construção também pode ser representada pela seguinte matriz:

$$G_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}, \quad (3.7)$$

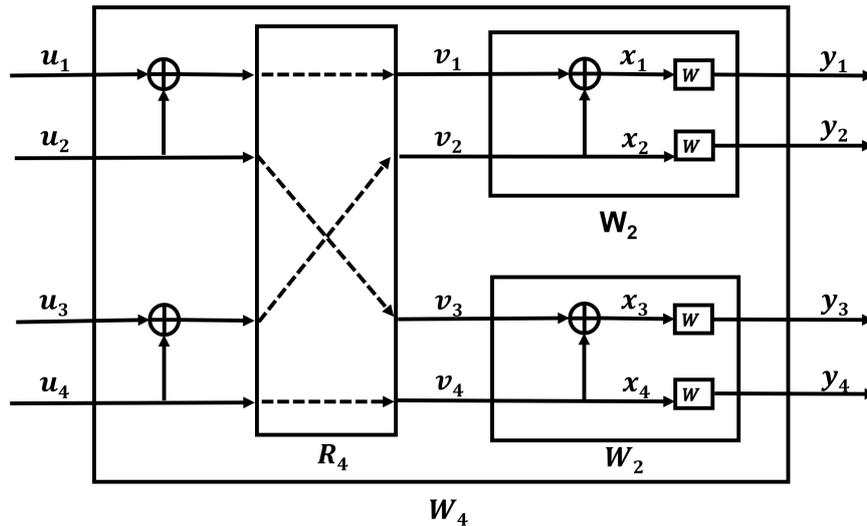
obtendo assim

$$x_1^2 = u_1^2 G_2.$$

A relação entre as probabilidades de transição de W_2 dada em (3.6) e de W^2 dada em (3.1) para $N = 2$, é dada por:

$$W_2(y_1^2 | u_1^2)$$

Agora, para $N = 4$, teremos o canal $W_4 : X^4 \rightarrow Y^4$ obtido pela combinação de duas cópias independentes de W_2 . A construção do canal W_4 é apresentada a seguir na Figura 13:

Figura 13 – Construção de W_4 .

fonte: Própria autora.

Assim, as probabilidades de transição serão dadas por:

$$W_4(y_1^4|u_1^4) = W_4(y_1, y_2, y_3, y_4|u_1, u_2, u_3, u_4) = W_2(y_1^2|u_1 \oplus u_2, u_3 \oplus u_4)W_2(y_3^2|u_2, u_4). \quad (3.8)$$

Porém, a partir de $N = 4$ será necessária a utilização de uma operação de permutação para a combinação dos canais, como podemos observar na Figura 13, onde a operação R_4 permuta as entradas dos canais W_2 . Essa operação de permutação, que será denotada por R_N , irá separar as entradas de índices pares das ímpares, as entradas ímpares estarão posicionadas anteriormente das pares. Essa operação utiliza-se de uma matriz de permutação para inverter essas posições. No caso de $N = 4$, temos a seguinte permutação das entradas:

$$s_1^4 = (s_1 s_2 s_3 s_4) \longrightarrow v_1^4 = (s_1 s_3 s_2 s_4),$$

E assim, a matriz de permutação é dada da seguinte forma

$$R_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (3.9)$$

Mapeando $u_1^4 \mapsto x_1^4$, onde u_1^4 são as entradas de W_4 e x_1^4 são as entradas de W^4 , esta construção também pode ser representada pela seguinte matriz:

$$G_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}, \quad (3.10)$$

obtendo assim

$$x_1^4 = u_1^4 G_4.$$

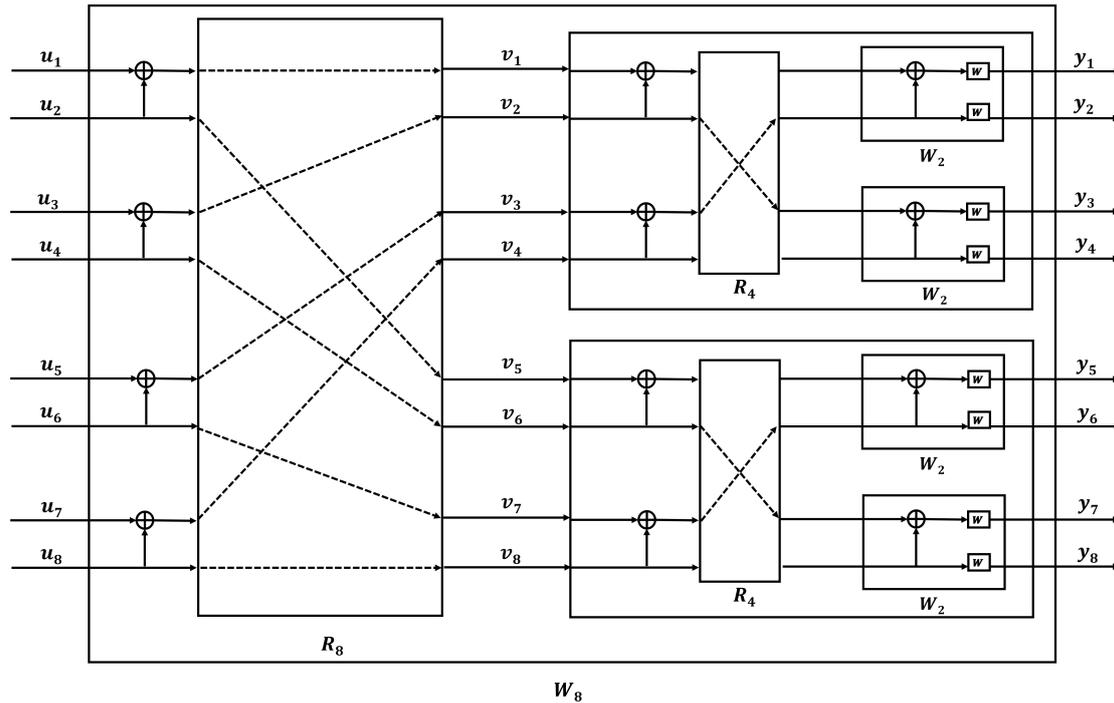
Tem-se a relação

$$W_4(y_1^4|u_1^4) = W^4(y_1^4|u_1^4 G_4)$$

entre as probabilidades de transição de W_4 dado em (3.8) e de W^4 como em (3.1) para $N = 4$.

Já para $N = 8$ gera-se o canal $W_8 : X^8 \rightarrow Y^8$ cuja construção é dada pela figura abaixo.

Figura 14 – Construção do canal W_8 .



fonte: Própria autora.

Assim, as probabilidades de transição serão dadas por

$$W_8(y_1^8|u_1^8) = W(y_1^4|u_1 \oplus u_2, u_3 \oplus u_4, u_5 \oplus u_6, u_7 \oplus u_8)W(y_5^8|u_{1,e}^8). \quad (3.11)$$

A operação de permutação R_8 irá mapear a entrada da seguinte forma

$$s_1^8 = (s_1 s_2 s_3 s_4 s_5 s_6 s_7 s_8) \mapsto v_1^8 = (s_1 s_3 s_5 s_7 s_2 s_4 s_6 s_8),$$

que pode ser representada pela seguinte matriz

$$R_8 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (3.12)$$

Mapeando $u_1^8 \mapsto x_1^8$, onde u_1^8 são as entradas de W_8 e x_1^8 são as entradas de W^8 , esta construção também pode ser representada pela seguinte matriz:

$$G_8 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \quad (3.13)$$

obtendo assim

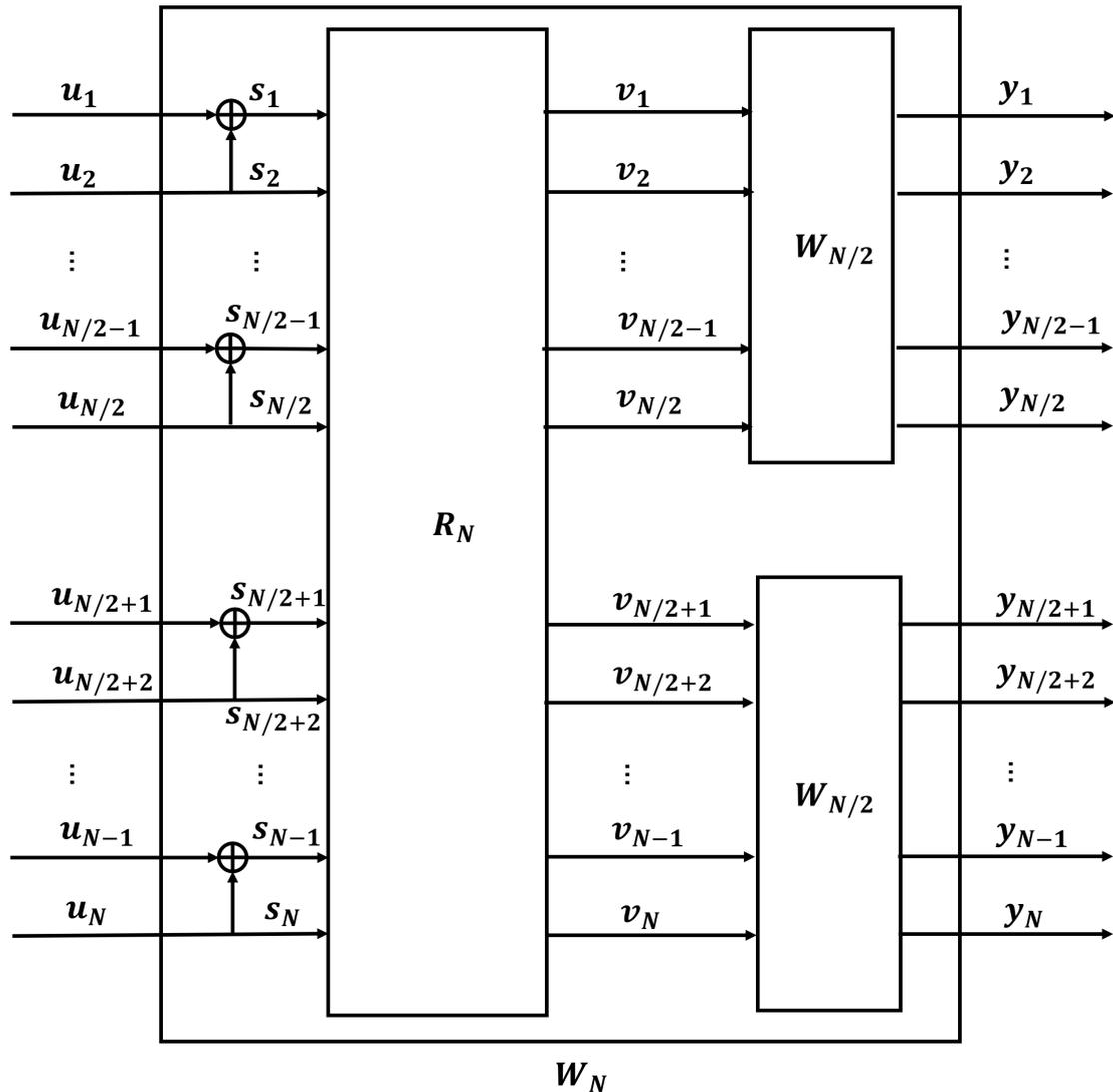
$$x_1^8 = u_1^8 G_8.$$

Novamente, tem-se a relação

$$W_8(y_1^8|u_1^8) = W^8(y_1^8|u_1^8 G_8)$$

entre as probabilidades de transição de W_8 dado em (3.11) e de W^8 como em (3.1) para $N = 8$.

Para o caso geral, duas cópias independentes de $W_{N/2}$, sendo $N/2$ correspondente a metade das N cópias do canal W que serão combinadas produzindo um canal $W_N : X^N \rightarrow Y^N$. A construção deste canal é dada na figura abaixo.

Figura 15 – Construção de W_N a partir de duas cópias de $W_{N/2}$ 

fonte: Própria autora.

Para descrevermos as probabilidades de transição de W_N , observe que o vetor de entrada u_1^N para W_N é transformado primeiramente em s_1^N de tal forma que

$$s_{2i-1} = u_{2i-1} \oplus u_{2i} \text{ e } s_{2i} = u_{2i}, \text{ para } 1 < i < N/2.$$

Assim, o operador R_N age na entrada s_1^N produzindo $v_1^N = (s_1 s_3, \dots, s_{N-1}, s_2 s_4, \dots, s_N)$ que se tornarão as entradas para as duas cópias de $W_{N/2}$.

Prosseguindo por indução, mapeando $u_1^N \mapsto x_1^N$, da entrada do vetor do canal W_N para a entrada do canal W^N pode ser representada pela matriz G_N tal que

$$x_1^N = u_1^N G_N, \quad (3.14)$$

onde, $N = 2^n, n \geq 0$.

A matriz G_N é chamada de matriz geradora de tamanho N . Dessa forma, as probabilidade de

transição dos dois canais W_N e W^N são relacionados por:

$$W_N(y_1^N | u_1^N) = W^N(y_1^N | u_1^N G_N). \quad (3.15)$$

3.3 DIVISÃO DE CANAL

Nesta etapa de divisão, os canais sintetizados são reagrupados pelo grau de polarização, ou seja, transformamos novamente W_N em um conjunto de N canais com coordenadas binárias

$$W_N^{(i)} : X \longrightarrow X^N \times X^{i-1}, \text{ com } 1 \leq i \leq N,$$

definido pelas probabilidades de transição

$$W_N^{(i)}(y_1^N, u_1^{i-1} | u_i) = \sum_{u_{i+1}^N \in X^{N-i}} \frac{1}{2^{N-1}} W_N(y_1^N | u_1^N),$$

onde u_i denota a entrada e (y_1^N, u_1^{i-1}) denota a saída de $W_N^{(i)}$.

Por exemplo, para $N = 1$, escrevemos

$$(W, W) \mapsto (W_2^{(1)}, W_2^{(2)}),$$

onde por (3.6), temos que

$$W_2^{(1)}(y_1^2 | u_1) = \sum_{u_2} \frac{1}{2} W_2(y_1^2 | u_1^2) = \sum_{u_2} \frac{1}{2} W(y_1 | u_1 \oplus u_2) W(y_2 | u_2)$$

e

$$W_2^{(2)}(y_1^2, u_1 | u_2) = \frac{1}{2} W_2(y_1^2 | u_1^2) = \frac{1}{2} W(y_1 | u_1 \oplus u_2) W(y_2 | u_2).$$

Para um valor qualquer de $N = 2^n$, com $n \geq 0$ e $1 \leq i \leq N$, generalizamos da seguinte forma:

$$(W_N^{(i)}, W_N^{(i)}) \mapsto (W_{2N}^{(2i-1)}, W_{2N}^{(2i)}),$$

com probabilidades de transição dadas por:

$$W_{2N}^{(2i-1)}(y_1^{2N}, u_1^{2i-2} | u_{2i-1}) = \sum_{u_{2i}} \frac{1}{2} W_N^{(i)}(y_1^N, u_{1,o}^{2i-2} \oplus u_{1,e}^{2i-2} | u_{2i-1} \oplus u_{2i}) W_N^{(i)}(y_{N+1}^{2N}, u_{1,e^{2i-2}} | u_{2i}) \quad (3.16)$$

e

$$W_{2N}^{(2i)}(y_1^{2N}, u_1^{2i-1} | u_{2i}) = \frac{1}{2} W_N^{(i)}(y_1^N, u_{1,o}^{2i-2} \oplus u_{1,e}^{2i-2} | u_{2i-1} \oplus u_{2i}) W_N^{(i)}(y_{N+1}^{2N}, u_{1,e^{2i-2}} | u_{2i}). \quad (3.17)$$

Os canais $\{W_N^{(i)}\}$ também serão considerados na Seção 4.2, quando apresentamos a estratégia de decodificação por cancelamento sucessivo em que o i -ésimo elemento de decisão estima u_i após observar y_1^N e as entradas anteriores u_1^{i-1} . O efeito da combinação e divisão de canais na polarização

de canal é sintetizado no resultado a seguir.

Teorema 3.3.1 (ARIKAN, 2009) *Para qualquer canal B-DMC, W , os canais $\{W_N^{(i)}\}$ polarizam no sentido que, quando N tende para o infinito, para qualquer $\delta \in (0, 1)$, a fração de índices $i \in \{1, \dots, N\}$ para os quais $I(W_N^{(i)}) \in (1 - \delta, 1]$ tende para $I(W)$ e a fração de índices $i \in \{1, \dots, N\}$ para os quais $I(W_N^{(i)}) \in [0, \delta)$ tende para $1 - I(W)$.*

A polarização de canal pode ser recursivamente implementada transformando múltiplos usos independentes de um determinado B-DMC em um conjunto de usos sucessivos de canais de entrada binária sintetizados. Para os cálculos de $\{I(W_N(i))\}$ são utilizadas as Equações (3.16) e (3.17). Porém, segundo (ARIKAN, 2009), não se tem algoritmos eficientes conhecidos para o cálculo de $\{I(W_N(i))\}$ para um canal B-DMC, W , geral.

Na próxima seção, exemplificamos os cálculos de $\{I(W_N(i))\}$ para um canal BEC.

3.4 EXEMPLO DE POLARIZAÇÃO DE CANAL PARA BEC

Considere um canal binário com apagamento BEC, ver def. Figura 10, com probabilidade de transição $p = 0,5$. Por (ARIKAN, 2009), os valores $\{I(W_N(i))\}$ podem ser calculados através das relações recursivas

$$I(W_N^{(2i-1)}) = I(W_{N/2}^{(i)})^2 \quad (3.18)$$

e

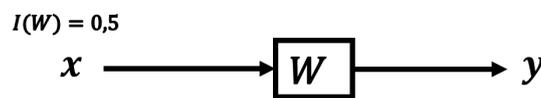
$$I(W_N^{(2i)}) = 2I(W_{N/2}^{(i)}) - I(W_{N/2}^{(i)})^2, \quad (3.19)$$

com $I(W_1^{(1)}) = 1 - p$.

Iniciando com um canal, $N = 2^0$, a correspondente capacidade do canal será

$$I(W) = 1 - p = 0,5.$$

Figura 16 – $I(W_1)$.

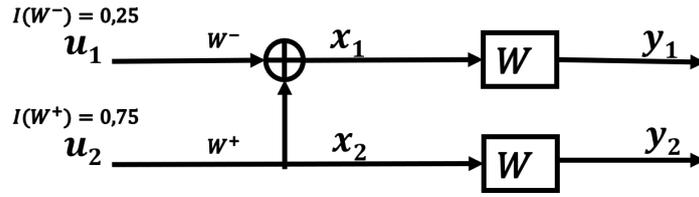


fonte: Própria autora.

Agora, combinando dois canais BEC independentes (W, W) , $N = 2^1$, a capacidade deste canal resultante será

$$2I(W) = 2 \times 0,5 = 1,$$

que será dividida entre os dois canais sintetizados, conforme ilustra a Figura 17, os quais denotaremos como W^- , com entrada u_1 e saídas y_1 e y_2 e, como W^+ com entrada u_2 e saídas u_1 , y_1 e y_2 .

Figura 17 – $I(W_2)$.

fonte: Própria autora.

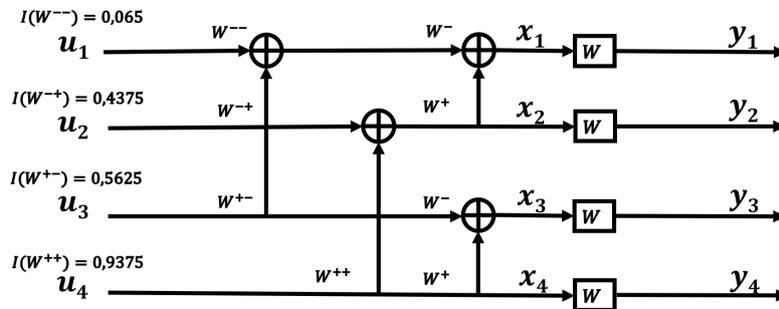
Assim, após a combinação e divisão do canal teremos

$$I(W^-) + I(W^+) = 2I(W) = 1,$$

com $I(W^-) \leq I(W) \leq I(W^+)$. Logo, por (3.18) e (3.19), temos que

$$\begin{aligned} I(W^+) &= 2I(W) - I(W)^2 = 1 - 0,5^2 = 0,75 \\ I(W^-) &= I(W)^2 = 0,5^2 = 0,25. \end{aligned} \quad (3.20)$$

Prosseguindo com 4 usos independentes do BEC, $N = 2^2$, teremos inicialmente as 4 cópias de W divididas em 2 grupos, e os dois BECs de cada grupo são transformados em canais polarizados W^- e W^+ . Dessa forma, os canais W^{--} e W^{+-} são derivados do canal W^- e os canais W^{+-} e W^{++} são derivados do canal W^+ .

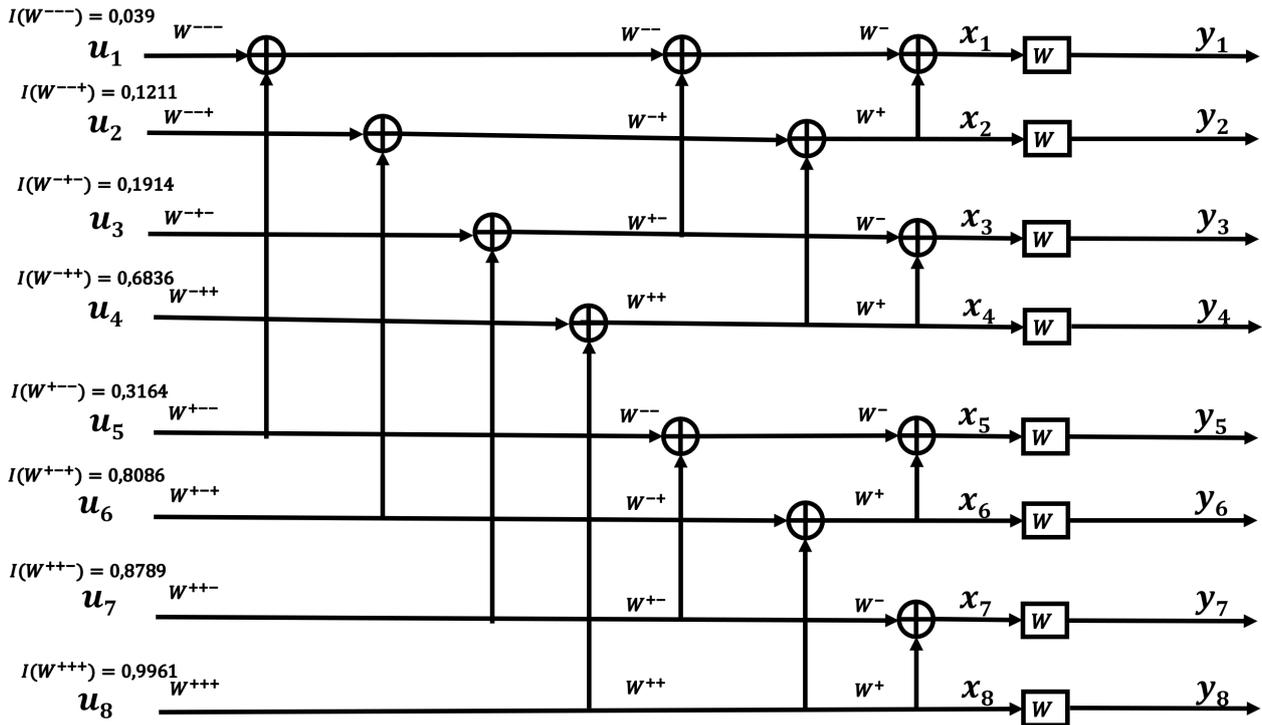
Figura 18 – $I(W_4)$.

fonte: Própria autora.

A capacidade dos canais serão dadas, utilizando (3.20), por

$$\begin{aligned} I(W^{--}) &= I(W^-)^2 = 0,25^2 = 0,0625 \\ I(W^{-+}) &= 2I(W^-) - I(W^-)^2 = 2 \times 0,25 - 0,25^2 = 0,4375 \\ I(W^{+-}) &= I(W^+)^2 = 0,75^2 = 0,5625 \\ I(W^{++}) &= 2I(W^+) - I(W^+)^2 = 2 \times 0,75 - 0,75^2 = 0,9375 \end{aligned} \quad (3.21)$$

Com 8 usos independentes do BEC, $N = 2^3$, dividimos inicialmente em 2 grupos de 4 canais, os quais cada um dos 2 são divididos em 2 grupos de 2, e os dois de cada grupo são transformados em dois canais polarizados W^- e W^+ .

Figura 19 – $I(W_8)$.

fonte: Própria autora.

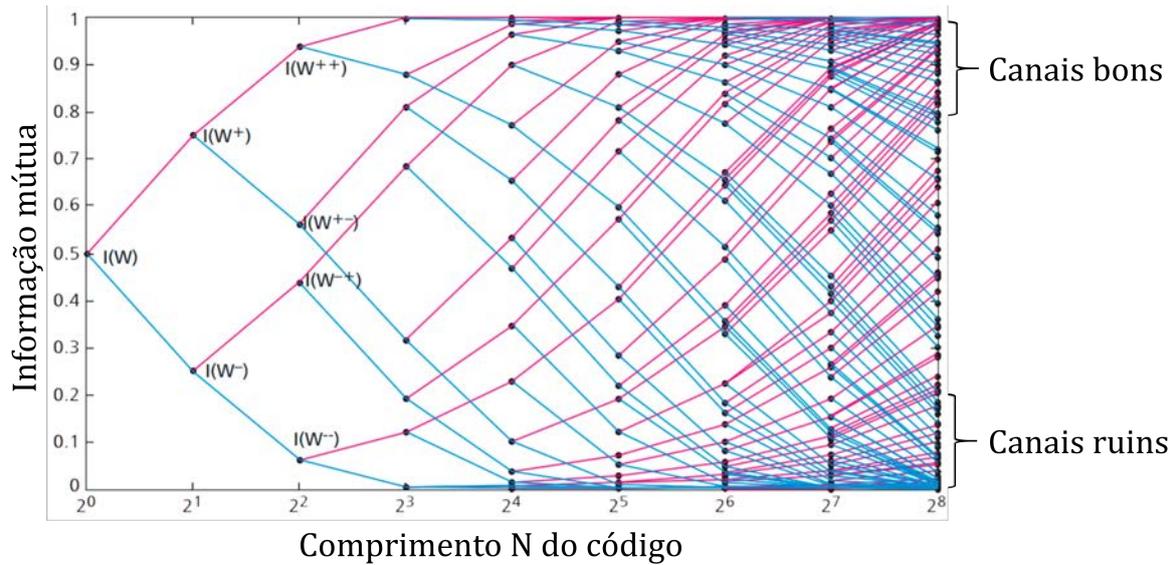
A capacidade dos canais serão dadas, utilizando (3.21), por

$$\begin{aligned}
 I(W^{---}) &= I(W^{--})^2 = 0,0039 \\
 I(W^{---+}) &= 2I(W^{--}) - I(W^{--})^2 = 0,1211 \\
 I(W^{-+-}) &= I(W^{-+})^2 = 0,1914 \\
 I(W^{-++}) &= 2I(W^{-+}) - I(W^{-+})^2 = 0,6836 \\
 I(W^{+--}) &= I(W^{+-})^2 = 0,3164 \\
 I(W^{+--+}) &= 2I(W^{+-}) - I(W^{+-})^2 = 0,8086 \\
 I(W^{++-}) &= I(W^{++})^2 = 0,8789 \\
 I(W^{+++}) &= 2I(W^{++}) - I(W^{++})^2 = 0,9961
 \end{aligned} \tag{3.22}$$

Procedendo com o mesmo raciocínio, a polarização pode ser continuamente realizada para $N = 2^n$ usos independentes do canal BEC, W , e a capacidade destes canais polarizados podem ser calculadas recursivamente.

Como o comprimento do código N tende para o infinito, os canais polarizados vão para os extremos, que correspondem a 0 e 1.

Figura 20 – Gráfico canais polarizados.



fonte: (NIU et al., 2014)

A técnica de polarização de canal, nos auxilia a separar os chamados canais bons dos chamados canais ruins, ou seja, permite identificar quais entradas serão bits de informação e quais serão bits congelados. Por exemplo, para construir um código com 8 bits de entrada sendo 4 congelados e 4 livres, analisamos a polarização $I(W_8)$ ilustrada na Figura 19 e descrita em 3.22. Assim, as entradas congeladas serão u_1, u_2, u_3 e u_5 , e as entradas livres serão u_4, u_6, u_7 e u_8 .

4 CODIGOS POLARES

Esse capítulo tratará sobre a codificação e decodificação dos códigos polares que utilizam o fenômeno de polarização do canal apresentado no Capítulo 3 para sintetizar o seu funcionamento. Na Seção 4.1 será apresentada a codificação dos códigos polares utilizando os canais W_N apresentados na Seção 3.2 e também através de expressões algébricas utilizando o produto de Kronecker de matrizes. Além disso, um exemplo dessa codificação será dado na Subseção 4.1.2. A Seção 4.2 irá demonstrar como é feita a decodificação por cancelamento sucessivo (*successive cancellation*, SC) para códigos polares. Na Subseção 4.2.1 será apresentado um exemplo do decodificador com tamanho 2 para na Subseção 4.2.2 ser apresentado o decodificador com tamanho N , contendo passo-a-passo desse processo de decodificação. A Subseção 4.2.3 exemplificará a decodificação passo-a-passo para um código de tamanho $N = 8$. Por fim, na Seção 4.3 será demonstrado como foi feita a implementação computacional de códigos polares para verificar sua codificação e decodificação gerando curvas para diferentes valores de N da BER pela relação sinal ruído. Foram utilizados para o desenvolvimento desse capítulo os seguintes artigos: (ARIKAN, 2009), (WASSERMAN, 2014), (HUILGOL, 2017), (LAMARE, 2017) e (VANGALA, 2018).

4.1 CODIFICAÇÃO DOS CÓDIGOS POLARES

Nesta seção apresentamos uma estratégia de codificação para os códigos polares através do produto tensorial de matrizes, o produto de Kronecker. Como veremos, esta estratégia está diretamente relacionada com a construção dos canais W_N apresentadas no Capítulo 3.

Foi visto na Seção 3.14, que para cada $N = 2^n$ com $n \geq 0$, a codificação de um canal com vetores de entrada u_1^N e vetores de saída x_1^N , se caracteriza por:

$$x_1^N = u_1^N G_N,$$

onde G_N é a matriz geradora de ordem N . Utilizando a polarização de canal, apresentada no Capítulo 3, para decidir quais serão os bits de informação e quais são os bits congelados, o congelamento dos bits pode ser feito a partir de uma divisão do vetor u_1^N em duas partes. Dividimos u_1^N em u_A que indica os vetores livres e u_{A^c} que indica os vetores congelados, onde A é um subconjunto arbitrário de $\{1, \dots, N\}$.

A codificação ficará da seguinte maneira:

$$x_1^N = u_A G_N(A) \oplus u_{A^c} G_N(A^c), \quad (4.1)$$

onde $G_N(A)$ indica a submatriz que é formada pelas linhas da matriz G_N com índices A , o mesmo ocorre para $G_N(A^c)$.

Fixando A e u_{A^c} e, deixando u_A livre, obtemos um mapeamento do bloco fonte u_A de comprimento K para o bloco palavra-código x_1^N de comprimento N . Dessa forma, definimos um **código polar**,

como sendo um código de bloco linear com parâmetros (N, K, A, u_{A^c}) , onde K é a dimensão do código e especifica o tamanho de A .

A seguir apresentamos um exemplo de codificação polar utilizando a Equação 4.1.

Exemplo 4.1.1 *Considere um código polar com parâmetros $(4, 2, \{2, 4\}, (1, 0))$. Pela Equação 4.1, o codificador corresponde a:*

$$x_1^4 = u_1^4 G_4 = (u_2, u_4) \cdot \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \oplus (1, 0) \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}$$

Como estamos considerando códigos binários, as entradas livres (u_2, u_4) podem assumir os valores $(0, 0)$, $(0, 1)$, $(1, 0)$ e $(1, 1)$. A seguir, descrevemos a palavra código para cada um dos casos.

- Para $(u_2, u_4) = (0, 0)$:

$$\begin{aligned} x_1^4 &= (0, 0) \cdot \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \oplus (1, 0) \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix} \\ &= (1, 0, 0, 0) \end{aligned}$$

- Para $(u_2, u_4) = (1, 0)$:

$$\begin{aligned} x_1^4 &= (1, 0) \cdot \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \oplus (1, 0) \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix} \\ &= (0, 0, 1, 0) \end{aligned}$$

- Para $(u_2, u_4) = (0, 1)$:

$$\begin{aligned} x_1^4 &= (0, 1) \cdot \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \oplus (1, 0) \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix} \\ &= (0, 1, 1, 1) \end{aligned}$$

- Para $(u_2, u_4) = (1, 1)$:

$$\begin{aligned} x_1^4 &= (1, 1) \cdot \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \oplus (1, 0) \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix} \\ &= (1, 1, 0, 1) \end{aligned}$$

4.1.1 Expressões Algébricas

Para um melhor entendimento da codificação serão demonstradas expressões algébricas correspondentes à matriz geradora G_N utilizada em códigos polares, essas formas demonstram a eficiência da implementação da geração de codificação $x_1^N = u_1^N G_N$.

Iniciamos definindo o produto de Kronecker de matrizes que será utilizado nesta estratégia de codificação.

Definição 4.1.1 *O produto de Kronecker é uma operação entre duas matrizes no qual denota-se pelo símbolo \otimes . Utilizando-se uma matriz A com dimensões $m \times n$ e uma matriz B com dimensões $p \times q$:*

$$A \otimes B = \begin{bmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{bmatrix} \quad (4.2)$$

Mais especificamente:

$$A \otimes B = \begin{bmatrix} a_{11}b_{11} & a_{11}b_{12} & \cdots & a_{1n}b_{11} & a_{1n}b_{12} \\ a_{11}b_{21} & a_{11}b_{22} & & a_{1n}b_{21} & a_{1n}b_{22} \\ & \vdots & \ddots & & \vdots \\ a_{m1}b_{11} & a_{m1}b_{12} & \cdots & a_{mn}b_{11} & a_{mn}b_{12} \\ a_{m1}b_{21} & a_{m1}b_{22} & & a_{mn}b_{21} & a_{mn}b_{22} \end{bmatrix} \quad (4.3)$$

Observação 4.1.1 *O tamanho da palavra código influenciará fortemente na complexidade da codificação. Para um tamanho de palavra código N será utilizada uma matriz $F_N = F^{\otimes n}$ onde se caracteriza por $F^{\otimes n} = F \otimes F^{\otimes(n-1)}$, no qual $n \geq 0$.*

Exemplo 4.1.2 *Seja*

$$F = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \quad (4.4)$$

A matriz $F^{\otimes n}$ varia conforme o produto de Kronecker, portanto para $n = 1$, $F^{\otimes 1} = F \otimes F^{\otimes 0}$:

Por convenção torna-se $F^{\otimes 0} = [1]$, então:

$$\begin{aligned} F^{\otimes 1} &= F \otimes F^{\otimes 0} \\ &= F \otimes [1] \\ &= \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}. \end{aligned}$$

Para $n = 2$

$$F^{\otimes 2} = F \otimes F^{\otimes 1}$$

$$F^{\otimes 2} = \left[\begin{array}{cc} \left(\begin{array}{cc} 1 & 0 \\ 1 & 1 \end{array} \right) \cdot 1 & \left(\begin{array}{cc} 1 & 0 \\ 1 & 1 \end{array} \right) \cdot 0 \\ \left(\begin{array}{cc} 1 & 0 \\ 1 & 1 \end{array} \right) \cdot 1 & \left(\begin{array}{cc} 1 & 0 \\ 1 & 1 \end{array} \right) \cdot 1 \end{array} \right]$$

$$F^{\otimes 2} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}.$$

Para $n = 3$:

$$F^{\otimes 3} = F \otimes F^{\otimes 2}$$

$$F^{\otimes 3} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

Considere o canal W_2 descrito na Figura 12, iremos denotar a sua matriz geradora dada em (3.7) por

$$F = G_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}. \quad (4.5)$$

Já para o canal W_4 descrito na Figura 13, a matriz geradora G_4 pode ser obtida por meio da seguinte expressão:

$$G_4 = (I_2 \otimes F)R_4(I_2 \otimes F), \quad (4.6)$$

onde I_2 é a matriz identidade de ordem 2 e R_4 é a matriz de permutação dada em (3.9). E, para canal W_8 descrito na Figura 14:

$$G_8 = (I_4 \otimes F)R_8(I_4 \otimes G_4), \quad (4.7)$$

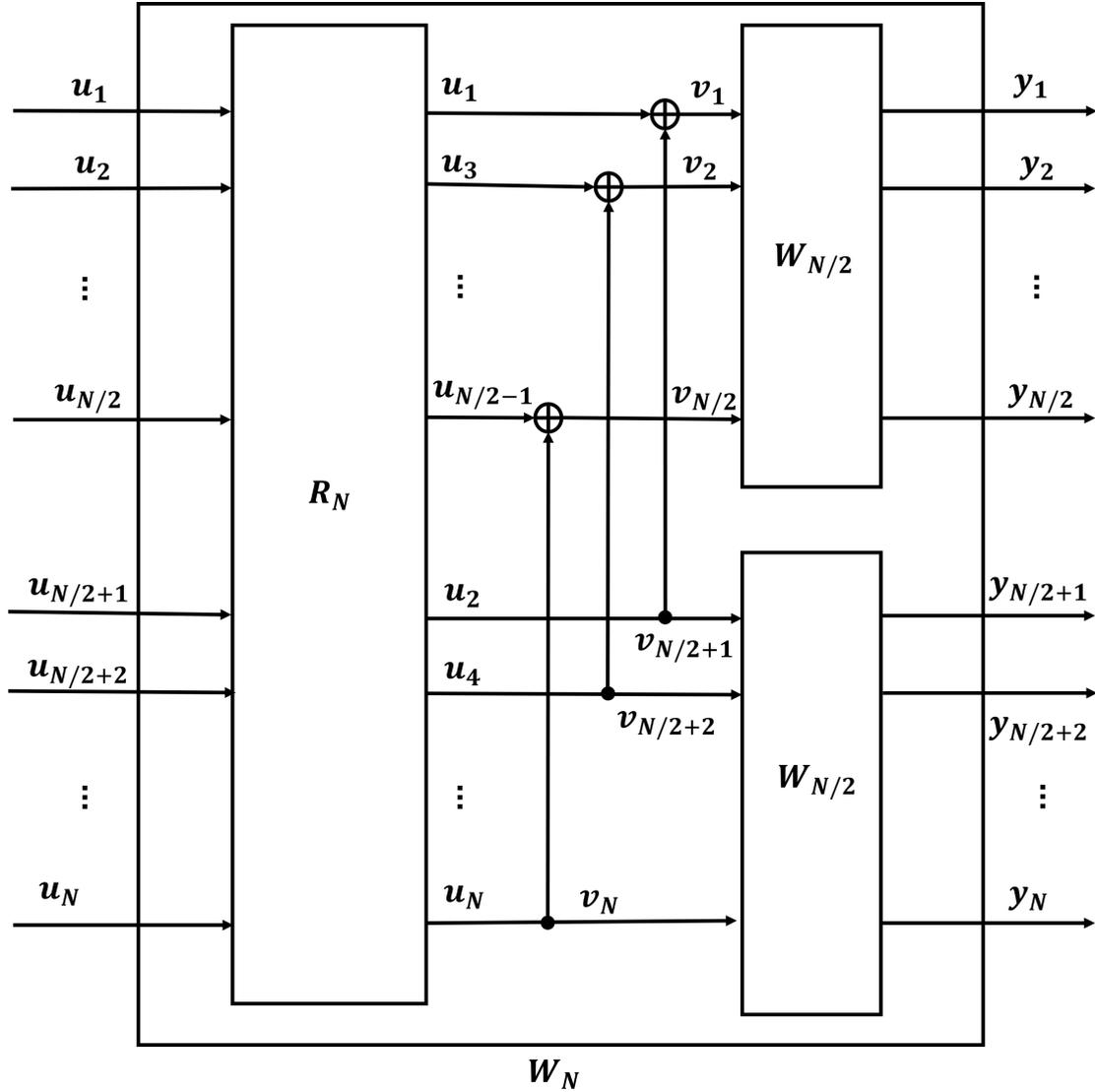
onde I_4 é a matriz identidade de ordem 4 e R_8 é a matriz de permutação dada em (3.12). Prosseguindo de modo análogo, para um canal W_N como descrito na Figura 15, a matriz geradora G_N será obtida por:

$$G_N = (I_{N/2} \otimes F)R_N(I_2 \otimes G_{N/2}) \quad (4.8)$$

onde $I_{N/2}$ é a matriz identidade de ordem $N/2$ e R_N é uma matriz de permutação de ordem N .

A combinação de canal descrita na Figura 15 pode ser modificada de modo que se obtenha uma combinação como na Figura 21.

Figura 21 – Construção de W_N alternativa.



fonte: Própria autora.

Essa modificação algebricamente corresponde a seguinte equação:

$$(I_{N/2} \otimes G_2)R_N = R_N(F \otimes I_{N/2}). \quad (4.9)$$

Logo, substituindo a relação (4.9) em (4.8) obtém-se:

$$G_N = R_N(F \otimes I_{N/2})(I_2 \otimes G_{N/2}) = R_N(F \otimes G_{N/2}) \quad (4.10)$$

Substituindo $G_{N/2} = R_{N/2}(F \otimes G_{N/4})$ em (4.10) e utilizando a identidade

$$(AC) \otimes (BD) = (A \otimes B)(C \otimes D),$$

com $A = I_2$, $B = R_{N/2}$, $C = F$ e $D = F \otimes G_{N/4}$ segue que:

$$G_N = R_N(F \otimes (R_{N/2}(F \otimes G_{N/4}))) = R_N(I_2 \otimes R_{N/2})(F^{\otimes 2} G_{N/4}) \quad (4.11)$$

Aplicando novamente (4.10), finalmente obtemos que:

$$G_N = B_N F^{\otimes n}, \quad (4.12)$$

onde

$$B_N = R_N(I_2 \otimes B_{N/2}) = B_N = R_N(I_2 \otimes R_{N/2})(I_4 \otimes R_{N/4}) \dots (I_{N/2} \otimes R_2) \quad (4.13)$$

O operador permutação dado pela matriz de permutação R_N será chamado operador de embaralhamento no qual faz o mapeamento das entradas do canal, trocando suas posições, que como vimos, é feito por meio do deslocamento das entradas de índices ímpares para as posições iniciais do canal e logo em seguida as entradas com índices pares são posicionadas, as posições dos índices são deslocadas em ordem de crescimento.

Já a matriz de permutação B_N atua como operador inversor de bits, como é possível observar em (4.13). Conforme N aumenta, B_N fará cada vez mais trocas nas posições dos vetores.

Para o valor $N = 4$, $B_4 = R_4$ dado em (3.9). Já para $N = 8$, considerando R_8 como em (3.12), segue que:

$$\begin{aligned} B_8 &= R_8(I_2 \otimes B_4) \\ &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

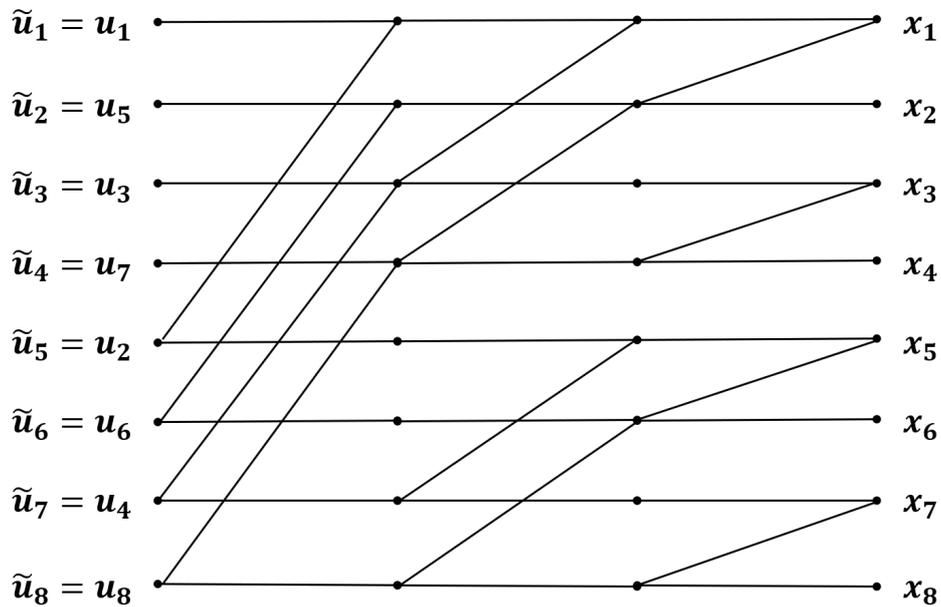
Definição 4.1.2 Para estimar a complexidade da codificação, será utilizado um modelo computacional que é uma máquina com um único processador com memória de acesso aleatório e as complexidades serão expressadas no tempo. A discussão será dada para um código de classe lateral G_N arbitrário

com parâmetros (N, K, A, u_{Ac}) .

O pior caso de complexidade de codificação sobre todos (N, K, A, u_{Ac}) com comprimento N de bloco se classificará por $\chi_E(N)$. Utilizando a complexidade de uma adição mod-2 escalar como uma unidade e a complexidade da operação de embaralhamento R_N como N unidades, é possível observar pela Figura 15 que $\chi_E(N) \leq N/2 + N + 2\chi_E(N/2)$. Com um valor inicial $\chi_E(2) = 3$, obtém-se que $\chi_E N \leq 3/2 N \log_N$ para todos $N = 2^n$, $n \geq 1$, sendo assim a complexidade da codificação é $O(N \log_N)$.

Uma implementação específica do codificador é ilustrada na Figura 22, nessa imagem utiliza-se um canal com $N = 8$.

Figura 22 – Circuito para implementação da transformação $F^{\otimes 3}$.



fonte: Própria autora.

Ao fazer uma análise detalhada dos nós do codificador da Figura 22 é possível observar que as saídas x_1^8 irão corresponder as seguintes entradas de \tilde{u}_1^8 :

- $x_1 \longleftrightarrow \tilde{u}_1, \tilde{u}_2, \tilde{u}_3, \tilde{u}_4, \tilde{u}_5, \tilde{u}_6, \tilde{u}_7, \tilde{u}_8$;
- $x_2 \longleftrightarrow \tilde{u}_2, \tilde{u}_4, \tilde{u}_6, \tilde{u}_8$;
- $x_3 \longleftrightarrow \tilde{u}_3, \tilde{u}_4, \tilde{u}_7, \tilde{u}_8$;
- $x_4 \longleftrightarrow \tilde{u}_4, \tilde{u}_8$;
- $x_5 \longleftrightarrow \tilde{u}_5, \tilde{u}_6, \tilde{u}_7, \tilde{u}_8$;
- $x_6 \longleftrightarrow \tilde{u}_6, \tilde{u}_8$,
- $x_7 \longleftrightarrow \tilde{u}_8$.

Este formato do codificador demonstra que para entradas \tilde{u}_1^8 as saídas correspondentes serão

$$x_1^8 = \tilde{u}_1^8 F^{\otimes 3},$$

sendo assim o codificador da Figura 22 será equivalente à $F^{\otimes 3}$ dada por:

$$F^{\otimes 3} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

Agora, se analisarmos os nós da Figura 22 utilizando como entradas os valores de u_1^8 , que "sofrem" a operação de reversamento de bit, será equivalente à codificação apresentada na Figura 14. Assim, a matriz utilizada para codificação será G_8 como em (3.13). Portanto, para as entradas u_1^8 as saídas correspondentes serão:

$$x_1^8 = u_1^8 G_8.$$

Uma alternativa de aplicação do codificador seria a utilização de u_1^8 na ordem natural de entrada do circuito na Figura 22, obtendo-se

$$\tilde{x}_1^8 = u_1^8 F^{\otimes 3}$$

na saída. Sendo assim, a codificação poderia ser completada por uma operação de inversão de bits após o reversamento de bit, ou seja,

$$x_1^8 = \tilde{x}_1^8 B_8 = u_1^8 G_8.$$

A complexidade dessa implementação é $O(N \log_N)$ com $O(N)$ para B_N e $O(N \log_N)$ para $F^{\otimes n}$.

Muitas alternativas de implementação para $F^{\otimes n}$ podem ser obtidas pelo circuito de codificação da Figura 22, como por exemplo utilizando N processadores, um faz a implementação "coluna por coluna" reduzindo então a latência total para \log_N , várias outras trocas são possíveis entre latência e complexidade de hardware.

Atualmente códigos polares são implementados utilizando $F^{\otimes n}$ no lugar de $B_N F^{\otimes n}$ com o mapeamento do codificador como intuito de simplificar a implementação. Para esse caso, o decodificador deve compensar decodificando os elementos do vetor de origem u_1^N na ordem indexada de reversamento de bit. Inclui-se assim, B_N como parte do codificador nesse projeto para a utilização de um decodificador que decodifica u_1^N na ordem natural de índice, simplificando a notação.

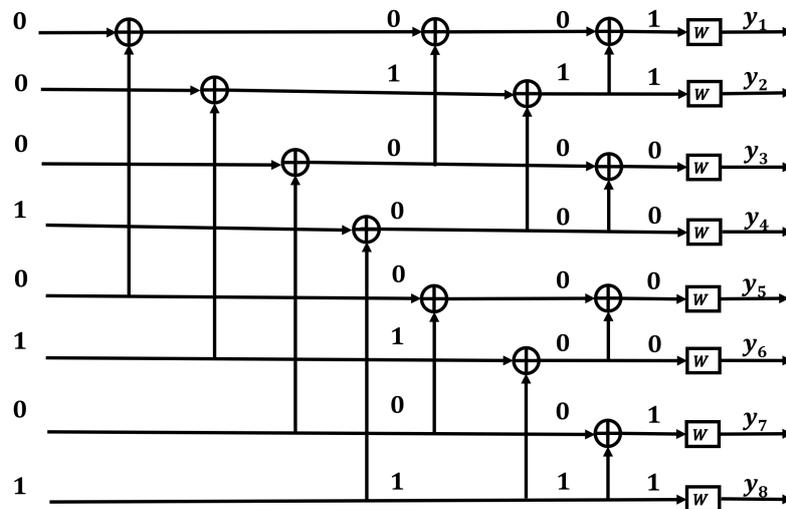
4.1.2 Exemplo Codificação

Para $N = 8$ canais, utilizando a polarização de canal exemplificado na Seção 3.3, podemos utilizar quatro entradas congeladas e quatro livres, para construir um código polar com os seguintes parâmetros: $(N, K, A, u_{A^c}) = (8, 4, \{4, 6, 7, 8\}, (0, 0, 0, 0))$. Neste caso, as entradas livres u_4, u_6, u_7, u_8 podem assumir os valores 0 ou 1.

A seguir vamos exemplificar esta codificação para dois vetores de entrada utilizando a codificação descrita na Figura 19.

- Para $u' = (0, 0, 0, 1, 0, 1, 0, 1)$ temos a seguinte codificação:

Figura 23 – Codificação u' .

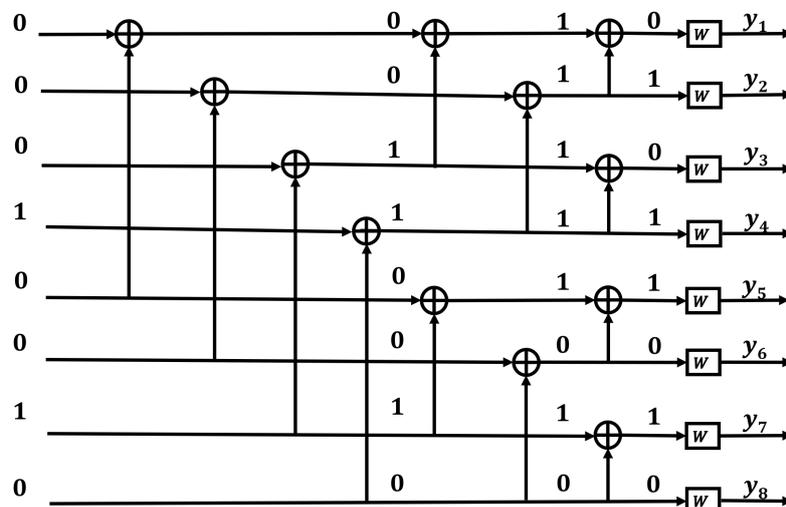


fonte: Própria autora.

Logo, a palavra-código é $x' = (1, 1, 0, 0, 0, 0, 1, 1)$.

- Para $u'' = (0, 0, 0, 1, 0, 0, 1, 0)$ temos a seguinte codificação:

Figura 24 – Codificação u'' .



fonte: Própria autora.

Logo, a palavra-código é $x'' = (0, 1, 0, 1, 1, 0, 1, 0)$.

Utilizando o circuito descrito na Figura 22, que utiliza da equação $x_1^8 = \tilde{u}_1^8 F^{\otimes 3}$ para a codificação do canal, faremos um exemplo para $\tilde{u}_1^8 = (0, 0, 0, 1, 0, 0, 1, 0)$. Tem-se que:

$$x_1^8 = (0, 0, 0, 1, 0, 0, 1, 0) \cdot \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

$$x_1^8 = (0, 1, 0, 1, 1, 0, 1, 0). \quad (4.14)$$

Portanto, a palavra-código x_1^8 é $(0, 1, 0, 1, 1, 0, 1, 0)$, que como pode-se observar é igual a palavra-código x'' .

4.2 DECODIFICAÇÃO DOS CÓDIGOS POLARES POR SC

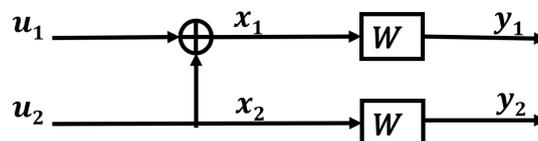
A decodificação de códigos polares é feita a partir de um decodificador de cancelamento sucessivo (*successive cancellation decoder*, SCD). Nessa técnica, o decodificador geralmente dispõe somente das informações sobre os valores e posições dos bits congelados, u_{Ac} e A respectivamente, e conforme é feita a decodificação do sinal estima-se \hat{u}_1^N referente aos vetores u_1^N . A razão de verossimilhança (*Likelihood Ratio*, LR) é obtida pela equação:

$$L_N^{(i)}(y_1^N, u_1^{i-1}) = \frac{W_N^i(y_1^N, u_1^{i-1} | u_i = 0)}{W_N^i(y_1^N, u_1^{i-1} | u_i = 1)}. \quad (4.15)$$

4.2.1 Decodificador de tamanho 2

Para um decodificador com tamanho $N = 2$. A Figura 25 demonstra como sua entrada é construída, o codificador mapeia u_1^N para x_1^N que serão transmitidos através do canal W e serão recebidos vetores y_1^N .

Figura 25 – Codificação com $N = 2$.



fonte: Própria autora.

Com essa arquitetura de canal, G_2 é seu próprio inverso, ou seja $x_1 = u_1 \oplus u_2$, $x_2 = u_2$ equivale a $u_1 = x_1 \oplus x_2$, $u_2 = x_2$

Para qualquer y recebido, as transições de probabilidades são $W(y|x=0)$ e $W(y|x=1)$. Fixando $a = W(y_1|x_1=0)$, $b = W(y_1|x_1=1)$, $c = W(y_2|x_2=0)$, $d = W(y_2|x_2=1)$. Somente os dados a, b, c, d são necessários, descartando a necessidade de y_1 e y_2

A decodificação de u_1 é feita a partir da computação e comparação das probabilidades $W(y_1, y_2|u_1=0)$ e $W(y_1, y_2|u_1=1)$. Os vetores y_1^N também dependem de u_2 , que possui uma probabilidade equivalente de ser tanto 0 quanto 1. Então

$$\begin{aligned} W(y_1, y_2|u_1=0) &= W(y_1, y_2|u_1=0, u_2=0)/2 + W(y_1, y_2|u_1=0, u_2=1)/2 \\ &= W(y_1, y_2|x_1=0, x_2=0)/2 + W(y_1, y_2|x_1=1, x_2=1)/2. \end{aligned}$$

Com transições $x_1 \rightarrow y_1$ e $x_2 \rightarrow y_2$ independentes, pode-se avaliar as probabilidades emparelhadas:

$$\begin{aligned} W(y_1, y_2|u_1=0) &= W(y_1|x_1=0)W(y_2|x_2=0)/2 + W(y_1|x_1=1)W(y_2|x_2=1)/2 \\ &= (ac + bd)/2. \end{aligned}$$

Da mesma forma,

$$\begin{aligned} W(y_1, y_2|u_1=1) &= W(y_1, y_2|u_1=1, u_2=0)/2 + W(y_1, y_2|u_1=1, u_2=1)/2 \\ &= W(y_1, y_2|x_1=1, x_2=0)/2 + W(y_1, y_2|x_1=0, x_2=1)/2 \\ &= W(y_1|x_1=1)W(y_2|x_2=0)/2 + W(y_1|x_1=0)W(y_2|x_2=1)/2 \\ &= (bc + ad)/2. \end{aligned}$$

A razão dessas probabilidades é $(ac + bd)/(bc + ad)$. Dividindo numerador e denominador por bd obtém-se:

$$\frac{\frac{a}{b} \frac{c}{d} + 1}{\frac{c}{d} + \frac{a}{b}}.$$

Essa é a razão de verossimilhança de u_1 com y_1 e y_2 dados, podendo ser expressada em termos de a/b e c/d que são as *LRs* de x_1 e x_2 .

Define-se a função:

$$f(p, q) = \frac{pq + 1}{p + q}. \quad (4.16)$$

O valor de \hat{u}_1 é escolhido de acordo com a seguinte relação:

$$\hat{u}_1 = \begin{cases} 0, & \text{se } f(a/b, c/d) \geq 1 \\ 1, & \text{caso contrário.} \end{cases}$$

A decodificação de u_2 é feita a partir da determinação de qual valor de u_2 possui maior probabilidade

de produzir y_1 e y_2 . É feito o cálculo de LR de $W(y_1, y_2|u_2 = 0)/W(y_1, y_2|u_2 = 1)$. De acordo com regra de cancelamento sucessivo, assumindo que a decodificação de u_1 foi feita corretamente, se $u_1 = 0$, LR será:

$$\begin{aligned} \frac{W(y_1, y_2|u_1 = 0, u_2 = 0)}{W(y_1, y_2|u_1 = 0, u_2 = 1)} &= \frac{W(y_1, y_2|x_1 = 0, x_2 = 0)}{W(y_1, y_2|x_1 = 1, x_2 = 1)} \\ &= ac/bd \\ &= (a/b)(c/d). \end{aligned}$$

E para $u_1 = 1$, LR será

$$\begin{aligned} \frac{W(y_1, y_2|u_1 = 1, u_2 = 0)}{W(y_1, y_2|u_1 = 1, u_2 = 1)} &= \frac{W(y_1, y_2|x_1 = 1, x_2 = 0)}{W(y_1, y_2|x_1 = 0, x_2 = 1)} \\ &= bc/ad \\ &= (a/b)^{-1}(c/d). \end{aligned}$$

Em ambos os casos, verifica-se novamente que a LR de u_2 pode ser expressada em termos de a/b e c/d , que são as LRs de x_1 e x_2 . Defini-se a função:

$$q(p, q, u_1) = p^{1-2u_1}q. \quad (4.17)$$

E então a escolha de \hat{u}_2 é feita a partir da relação:

$$\hat{u}_2 = \begin{cases} 0, & \text{se } g(a/b, c/d, \hat{u}_1) \geq 1 \\ 1, & \text{caso contrário.} \end{cases}$$

Portanto, o decodificador não necessita todos os valores a, b, c, ed . Tendo as LRs : a/b e c/d é suficiente.

4.2.2 Decodificador de tamanho N

A decodificação do canal é feita utilizando as saídas y_1^N do canal com probabilidade $W_N(y_1^N|u_1^N)$. Os valores de u_i^N nessa técnica de decodificação são estimados pelo conjunto de bits congelados A^c e pela equação (4.15), no qual se obtém:

$$\hat{u}_1 = \begin{cases} u_i, & \text{se } i \in A^c \\ 0, & \text{se } i \in A^c \text{ e } L_N^{(i)}(y_1^N, \hat{u}_1^{i-1}) \geq 1 \\ 1, & \text{se } i \in A^c \text{ e } L_N^{(i)}(y_1^N, \hat{u}_1^{i-1}) < 1 \end{cases} \quad (4.18)$$

A equação (4.15) pode ser calculada na forma recursiva, sendo assim:

$$L_N^{(i)}(y_1^N, u_1^{i-1}) = \begin{cases} f(a, b), & \text{para } i \text{ ímpar} \\ g(a, b, \hat{u}_{i-1}), & \text{para } i \text{ par} \end{cases} \quad (4.19)$$

Sendo,

$$f(a, b) = \frac{a \cdot b + 1}{a + b}. \quad (4.20)$$

E,

$$g(a, b, \hat{u}_{i-1}) = a^{1-2\hat{u}_{i-1}} \cdot b, \quad (4.21)$$

onde

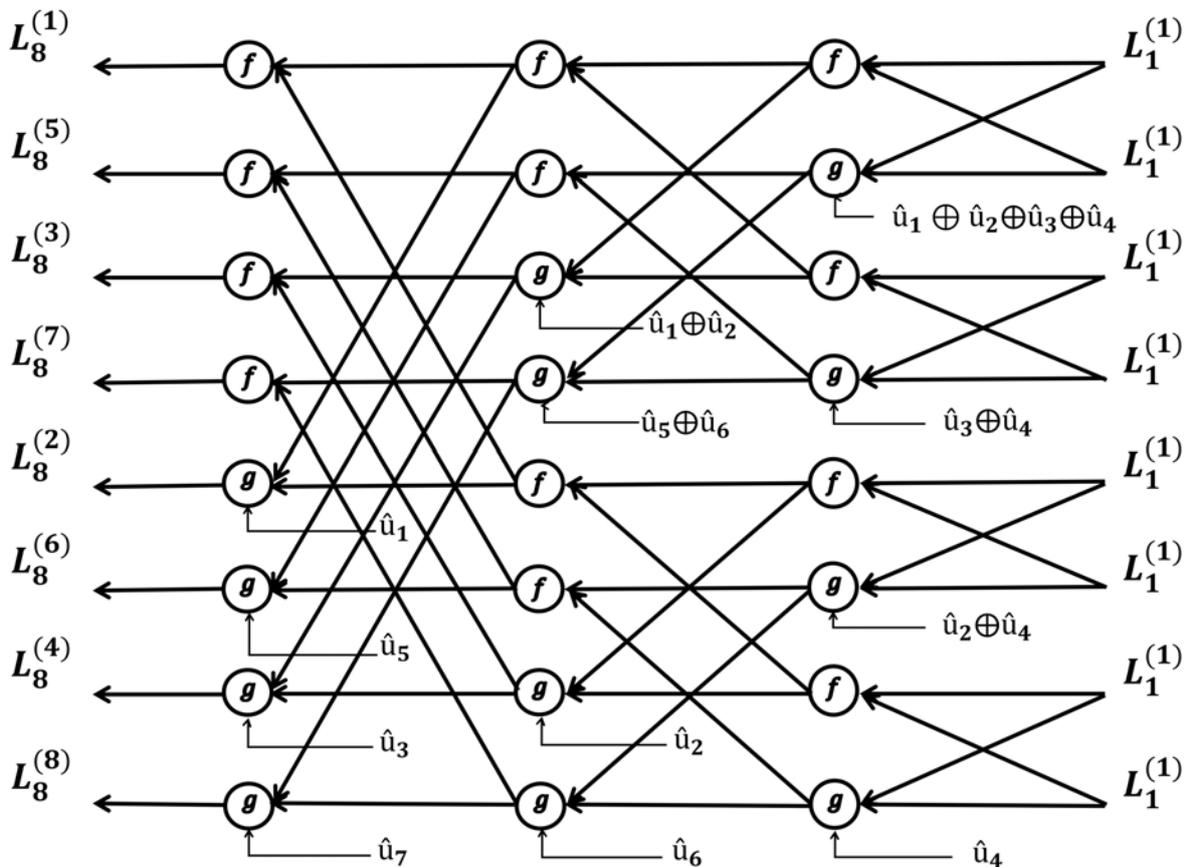
$$a = L_{N/2}^{i/2}(y_1^{N/2}), \hat{u}_{1,o}^{i-1} \oplus \hat{u}_{1,e}^{i-1} \text{ e } b = L_{N/2}^{i/2}(y_{N/2}^N), \hat{u}_{1,e}^{i-1}. \quad (4.22)$$

4.2.3 Passos para decodificação SC

Os passos a serem adotados pela codificação serão:

Passo 1: As variáveis que se localizam mais à direita da Figura 26 possuem inicialmente valores para $i = \{1, 2, \dots, N\}$, $L_1^{(1)}(y_i) = W(y_i|u_i = 0)W(y_i|u_i = 1)$.

Figura 26 – Arquitetura Decodificador SC $N = 8$.



fonte: Própria autora.

Passo 2: O primeiro bit u_1 é necessário para ser decodificado, caso for um bit congelado u_1 é definido como 0, caso contrário atualiza os índices de verossimilhança do lado mais à direita para o mais

à esquerda de acordo com (4.20) e (4.21) obtendo L_1^N pretendido. Se $L_1^N > 1$, u_1 é correspondente a 0. Caso $L_1^N < 1$ é igual a 1, caso contrario u_1 pode ser a 0 ou 1 com probabilidades iguais.

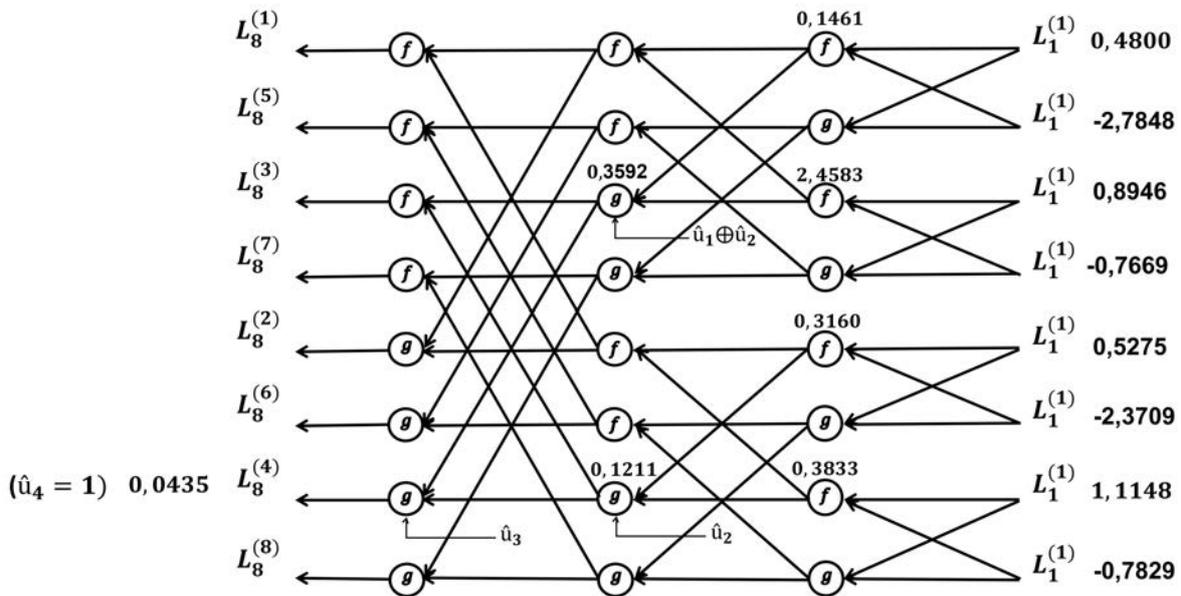
Passo 3: Nesse passo u_2 é decodificado, com um processo similar ao Passo 2, porém quando é calculado L_2^N utiliza-se o valor u_1 nas funções de g . Determina-se u_2 no momento em que se obtém L_2^N .

Passo 4: A partir do bit u_3 até u_N será feita a decodificação de acordo com as informações dos bits anteriores.

4.2.4 Exemplo decodificação SC para N=8

Para a decodificação de um código com tamanho $N = 8$ e taxa de código $R = 1/3$ que é transmitido por um canal AWGN com $E_b/N_0 = 1$ dB. Dispõe-se das seguintes informações: $A = \{4, 6, 8\}$, sendo os valores dos bits de informação todos 1, os bits congelados: (u_1, u_2, u_5, u_7) correspondem à 1. Os cálculos das LR para os bits de informação serão demonstrados na Figura 27. Os cálculos feitos são baseados em (4.15), designando u_i como 0 se L_N for maior que 1 e vice-versa. A Figura 27 demonstra os cálculos feitos para u_4 .

Figura 27 – Decodificador SC, $N = 8$.



fonte: Própria autora.

Passo 1: No início as variáveis que estão na extrema direita da Figura 26 são iniciadas como: $\{0,4800; -2,7848; 0,8946; -0,7669; 0,5275; -2,3709; 1,1148; -0,7829\}$, sendo esses valores observações dos canais.

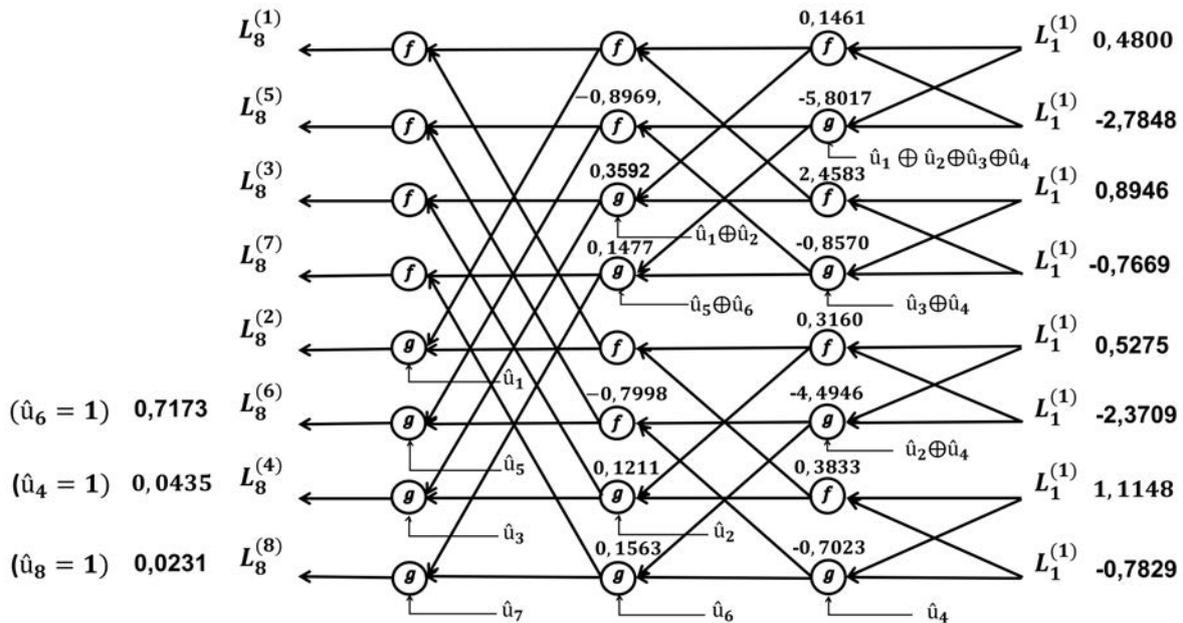
Passo 2: Descobre-se quando u_1 é um bit de informação antes de fazer sua decodificação, desde que os valores congelados são $\{1, 2, 3, 5\}$ então é possível saber que u_1 é um bit congelado e vale 0.

Passo 3: Semelhante ao passo 2, sabe-se que u_2 e u_3 são bits congelados, portanto eles correspondem à 0, decodificando u_4 , no momento em que é feita a decodificação a informação de LR é atualizada da direita para a esquerda, utilizando somente valores numéricos como exemplo. O valor

0,1461 é obtido calculando uma função $f = 1 + 0,4800 \cdot (-2,7848)0,4800 - 2,7848 = 0,1461$.
 Obtém-se o valor 0,3592 por uma função $g = 0,14611 - 2^{(u_1 \oplus u_2)} \cdot 2,4583 = 0,3592$.

Com valores de u_2 e u_1 conhecidos antes de fazer a decodificação, os valores restantes são obtidos de maneira similar, após o cálculo de $L_4^8 = 0,0435$, define-se que u_4 é 1, pois L_4^8 é menor que 1. Os bits restantes são calculados da mesma maneira na Figura 28 sendo possível observar o resultado geral da decodificação.

Figura 28 – Decodificador SC $N = 8$ após todos os passos.



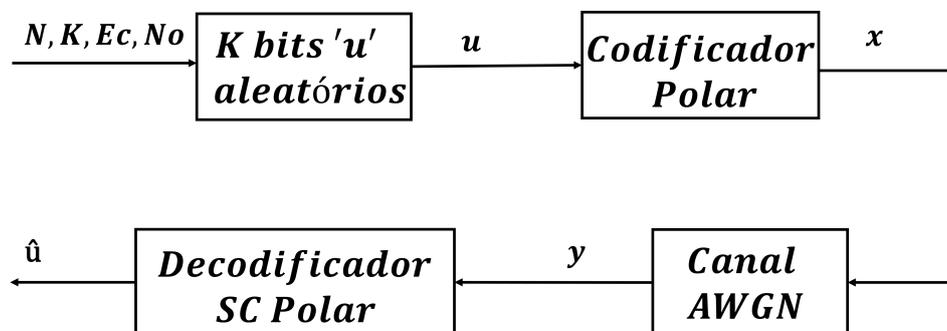
fonte: Própria autora.

4.3 IMPLEMENTAÇÃO COMPUTACIONAL

Nesta seção, apresentamos uma implementação computacional feita por meio de um algoritmo utilizando o software MATLAB, que foi baseado no algoritmo proposto em (VANGALA, 2018).

O diagrama de blocos a seguir ilustra o funcionamento do Algoritmo proposto.

Figura 29 – Diagrama de Blocos Codificação/Decodificação.



fonte: Própria autora.

O Algoritmo é inicializado propondo-se valores para N, K, Ec e N_0 , onde Ec corresponde à energia de símbolo em escala linear da modulação $BPSK$, N_0 é a densidade espectral de potência do

ruído, N é o comprimento do bloco e K o comprimento da mensagem. Com os valores de K e N se obtém a taxa de bits do canal.

A partir dos dados de entrada, o Algoritmo gera K **bits aleatórios** em um vetor u de comprimento N que servirão como mensagem de entrada no codificador, as demais entradas $N - K$ de u serão fixadas (congeladas) pelo processo de polarização de canal. O **Codificador Polar** transformará esse vetor u na palavra-código x da seguinte forma

$$x = d.F(x, n), \quad (4.23)$$

onde d é um vetor de N bits incluindo os bits de informação e os bits congelados, e $F(x, n)$ é o n -ésimo produto de Kronecker da matriz F dada em (4.5). A Equação 4.23 desenvolvida pelo algoritmo, utiliza (4.1) e (4.12).

Os bits que saem do codificador passam por um **Modulador BPSK** e por um **Canal** no qual será introduzido um ruído AWGN.

Posteriormente, o sinal entrará no **Decodificador SC Polar** sendo primeiramente demodulado para então ser decodificado. A decodificação será feita pela técnica de cancelamento sucessivo que foi apresentada na Seção 4.2, essa decodificação obterá um valor \hat{u} estimado. Por fim, é feita a comparação entre os valores u de entrada e \hat{u} de saída e então será gerado um gráfico com os valores obtidos de BER em função da razão sinal ruído, E_b/N_o .

Para obtermos os resultados, foram feitas cinco simulações para os seguintes valores de N e K :

$$(N, K) = (128, 64; 256, 128; 512, 256; 1024, 512; 2048, 1024)bits,$$

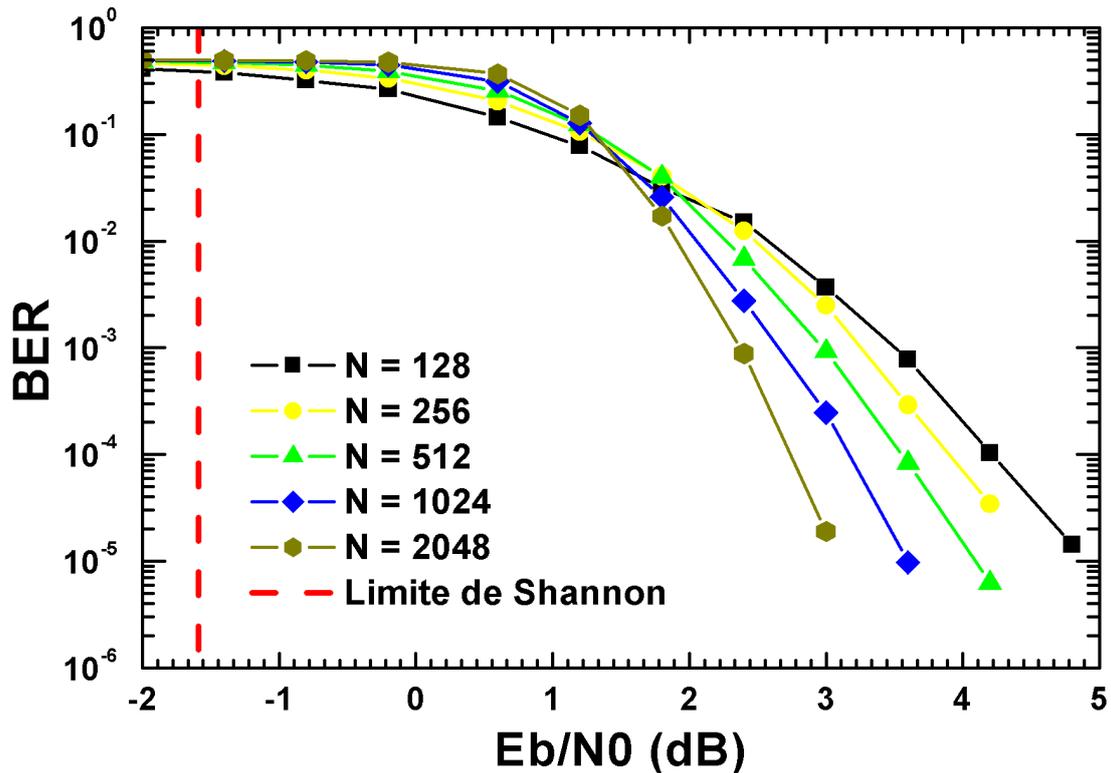
para gerar um gráfico comparativo entre as curvas obtidas para diferentes valores. Os demais dados utilizados para inicializar o Algoritmo foram

$$E_c = 1 ; N_o = 2$$

e E_b/N_o foi variado de 0 a 5 dB.

O gráfico a seguir ilustra as curvas obtidas para valores de N e K variantes.

Figura 30 – Gráfico BER por Eb/No.

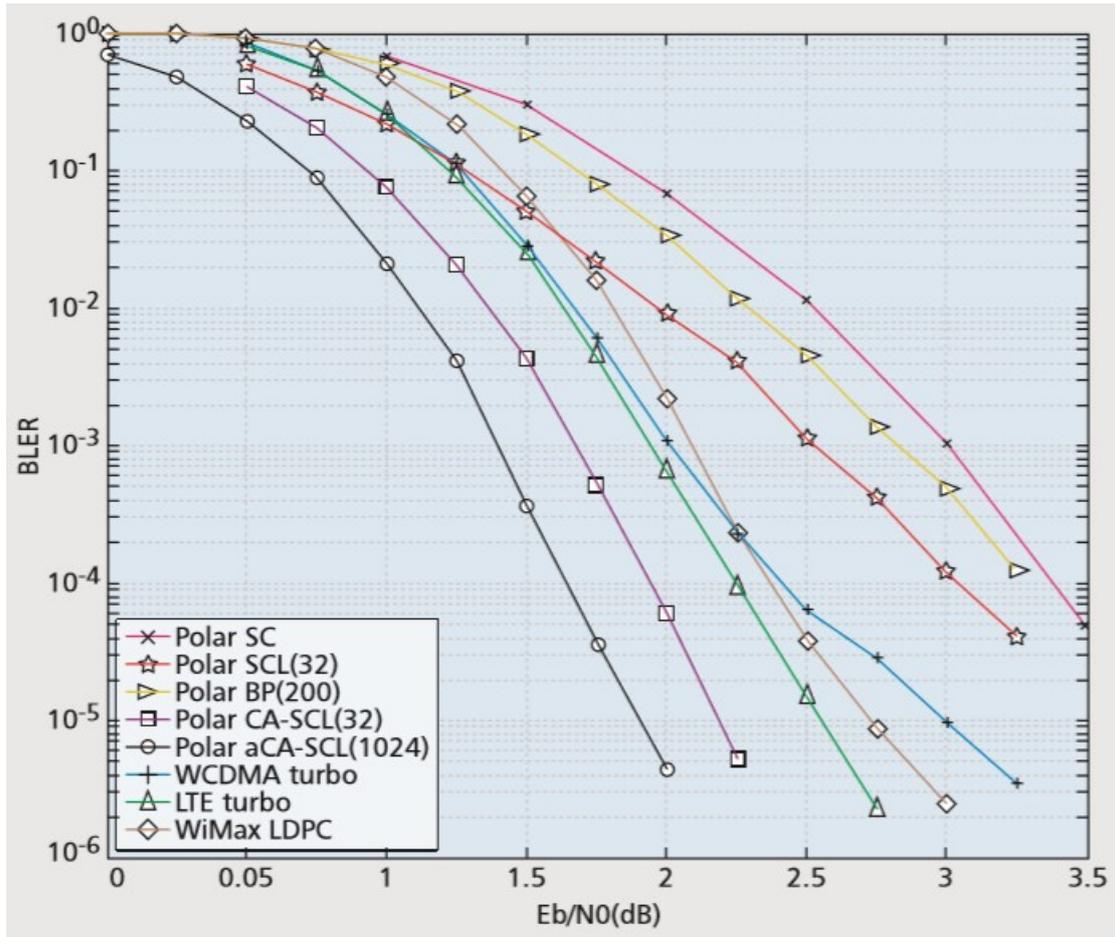


fonte: Própria autora.

A partir do gráfico da Figura 30 observa-se que ao variar o valor do tamanho N da palavra-código para diferentes valores de relação sinal-ruído, obtém-se curvas distintas umas das outras. Para valores de N crescentes, as curvas geradas ficam cada vez mais íngremes e vão se aproximando do Limite de Shannon, como ilustrado na Figura 30. É possível observar o aumento na eficiência desse código, pois ao se aproximar do Limite de Shannon, essas curvas possuem uma relação sinal ruído diminuída e também relação sinal-ruído cada vez menor, para menores valores de E_b/N_0 aumenta-se a eficiência espectral do canal. A partir da polarização do canal utilizada na codificação, de acordo com a teoria descrita no Capítulo 3, para valores de N tendendo ao infinito os canais polarizados tenderão para os extremos correspondentes a 0 ou 1, ou seja, para maiores valores de N haverá um aumento na confiabilidade do canal, dado que os canais ruins utilizados para enviar os bits congelados serão bem diferentes dos canais bons utilizados para enviar informações.

O limite de Shannon ilustrado na Figura 30 considera uma largura de banda infinita com E_b/N_0 tendo um valor aproximado por $\ln(2) = -1,59$.

Em (NIU et al., 2014) foram feitas simulações para diferentes esquemas de codificação, variando os parâmetros de cada codificação e a relação sinal-ruído para se obter as taxas de erro de blocos (*Block Error Ratio*, BLER). É possível observar essas simulações na Figura 31.

Figura 31 – Gráfico BLER por E_b/N_0 para diferentes codificações.

fonte: (NIU et al., 2014).

Para obter este resultado, foram utilizados códigos de tamanho $N = 1024$ para todos os esquemas, exceto para o código LDPC que utilizou $N = 1056$, com taxas de código igual a $1/2$, sendo transmitidos por um canal AWGN com entradas binárias. Para os códigos turbo foram utilizados dois esquemas de codificação: Acesso Múltiplo por Divisão de Código de Banda Larga (*Wide-Band Code-Division Multiple Access*, WCDMA) e LTE, com um máximo de oito iterações. Para os códigos LDPC utilizou-se o padrão WiMAX e um algoritmo de propagação de crenças (*Belief Propagation*, BP) com um número máximo de 200 iterações. Já para os códigos polares foram utilizados tamanhos de lista para verificação de redundância cíclica na lista de cancelamento sucessivo (*Cyclic Redundancy Check Successive Cancellation List*, CA-SCL) de 32 e um valor máximo de lista para verificação de redundância cíclica auxiliar na lista de cancelamento sucessivo (*Cyclic Redundancy Check aided Successive Cancellation List*, aCA-SCL) de 1024, foi utilizado número máximo de 200 iterações no decodificador BP.

É possível observar nesse gráfico que a codificação polar com decodificação SC apesar de possuir uma menor complexidade, tem uma performance mais fraca em comparação com outras técnicas. Esse gráfico demonstra que para um mesmo tamanho de palavra código em diferentes técnicas os códigos polares possuem vantagem em relação aos códigos turbo e LDPC, porém os códigos polares que possuem essa vantagem são os CA-SCL e aCA-SCL. Nesse trabalho não foi abordado a construção dos códigos polares descritos no gráfico, porém elas podem ser encontradas em (NIU et al., 2014).

5 CONCLUSÃO

Neste trabalho apresentamos um estudo sobre a codificação e a decodificação de códigos polares utilizando a técnica de polarização de canal. Como vimos, com a polarização de canal é possível separar canais considerados bons ou ruins através do cálculo de suas capacidades, após a separação desses canais serão enviados bits de informação nos canais bons para que não sofram degradação e bits fixados (congelados) nos canais ruins, pois serão enviados os valores desses bits para os decodificadores, portanto esses bits podem sofrer degradações pelos canais.

Além disso, foi discutido e exemplificado como foram feitas a codificação, que pode ser feita de duas maneiras: utilizando um circuito de codificação para canais W_N e também através de expressões algébricas utilizando o produto de Kronecker de matrizes, e decodificação, através da técnica de cancelamento sucessivo de um sinal por meio dos códigos polares, essa técnica consiste em fazer o caminho reverso da codificação para sucessivamente ir descobrindo os valores de capacidade referentes aos canais, após finalizada faz a decisão se o valor da saída é 0 ou 1 por meio de uma relação estabelecida. Foi possível observar que esses códigos possuem uma baixa complexidade de codificação, tornando-o assim um código de fácil implementação e com uma decodificação com grande eficiência.

Observamos também através da implementação computacional realizada para diferentes comprimentos do código, a eficiência dessa codificação e decodificação. Comparando sua relação sinal ruído com a taxa de erro de bits obtida, vimos que quanto maior o valor de N , mais o sinal se aproxima do limite da capacidade do canal de Shannon aumentando sua eficiência espectral com menores valores de taxa de erro de bit. Em uma comparação com diferentes codificações foi demonstrado que códigos polares são mais eficientes, porém com técnicas de decodificação diferentes do que a utilizada nesse trabalho.

Diversas pesquisas já foram feitas com o intuito de melhorar cada vez mais as técnicas de codificação e de decodificação por cancelamento sucessivo. Pesquisas futuras poderão focar em estudar mais profundamente variações destas técnicas de codificação e decodificação como por exemplo: codificação na forma sistemática (ARIKAN, 2011), (VANGALA; HONG; VITERBO, 2016) ; uma permutação na técnica de decodificação SC (PSCD), (VANGALA; VITERBO; HONG, 2014) e punctionamento (LAMARE, 2017) .

REFERÊNCIAS

- ARIKAN, E. Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels. **IEEE Transactions on Information Theory**, IEEE, v. 55, n. 7, p. 3051–3073, 2009.
- ARIKAN, E. Systematic polar coding. **IEEE Communications Letters**, IEEE, v. 15, n. 8, p. 860–862, 2011.
- COVER, J. A. T. T. M. **Elements of Information Theory**. [S.l.]: Wiley, 2012.
- HAYKIN, S. S. **Communication systems**. [S.l.]: Wiley, 2001.
- HEFEZ, M. L. T. V. A. **Códigos Corretores de Erros**. [S.l.]: IMPA, 2008.
- HUAWEI. **Huawei 5G: New Breakthrough on Channel Coding Technology with Polar Code**. 2016. Disponível em: <<http://www.huawei.com/en/press-events/news/2016/10/Huawei-5G-channel-coding-breakthrough>>.
- HUILGOL, S. **Channel Coding Techniques for 5G Using Polar Codes**. 2017.
- ISCAN, O.; LENTNER, D.; XU, W. A comparison of channel coding schemes for 5g short message transmission. In: **2016 IEEE Globecom Workshops (GC Wkshps)**. [S.l.: s.n.], 2016. p. 1–6.
- KUMAR, B. A.; RAO, P. T. Overview of advances in communication technologies. In: **2015 13th International Conference on Electromagnetic Interference and Compatibility (INCEMIC)**. [S.l.]: IEEE, 2015. p. 102–106.
- LAMARE, R. M. O. e Rodrigo C. de. Codigos polares e funcionamento baseado em polarizacao para sistemas 5g. **XXXV SIMPOSIO BRASILEIRO DE TELECOMUNICACOES E PROCESSAMENTO DE SINAIS**, p. 629 – 633, 2017.
- NIU, K. et al. Polar codes: Primary concepts and practical decoding algorithms. **IEEE Communications Magazine**, v. 52, n. 7, p. 192–203, 2014.
- RYAN, W.; LIN, S. **Channel Codes: Classical and Modern**. [S.l.]: Cambridge University Press, 2009.
- SHAFI, M. et al. 5g: A tutorial overview of standards, trials, challenges, deployment, and practice. **IEEE Journal on Selected Areas in Communications**, IEEE, v. 35, n. 6, p. 1201–1221, 2017.
- SHANNON, C. E. A mathematical theory of communication. **The Bell System Technical Journal**, Nokia Bell Labs, v. 27, n. 3, p. 379–423, 1948.
- SHARMA, A.; SALIM, M. Polar code: The channel code contender for 5g scenarios. **2017 International Conference on Computer, Communications and Electronics (Comptelix)**, IEEE, p. 676–682, 2017.
- VANGALA, H. **Polar Codes**. 2018. Disponível em: <<http://www.polarcodes.com/>>.
- VANGALA, H.; HONG, Y.; VITERBO, E. Efficient algorithms for systematic polar encoding. **IEEE Communications Letters**, IEEE, v. 20, n. 1, p. 17–20, 2016.

VANGALA, H.; VITERBO, E.; HONG, Y. Permuted successive cancellation decoder for polar codes. In: **2014 International Symposium on Information Theory and its Applications**. [S.l.]: IEEE, 2014. p. 438–442.

WANG, R.; LIU, R. A novel puncturing scheme for polar codes. **IEEE Communications Letters**, IEEE, p. 2081–2084, 2014.

WASSERMAN, D. **Technical Report 2054: Polar Codes**. [S.l.]: Spawar Systems Center Pacific, San Diego, CA, USA, 2014.