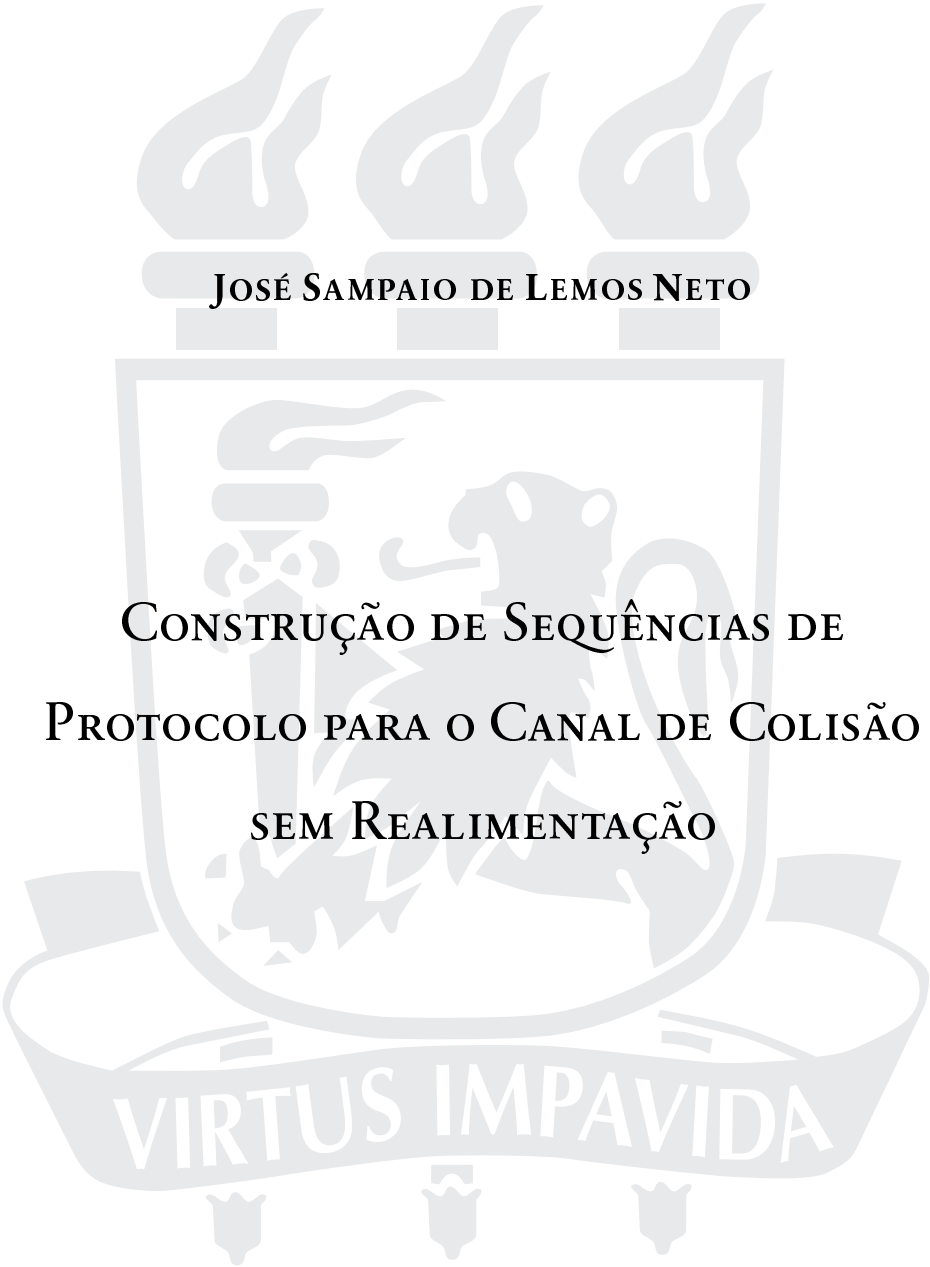
Universidade Federal de Pernambuco

Centro de Tecnologia e Geociências

Programa de Pós-graduação em Engenharia Elétrica



José Sampaio de Lemos Neto

Construção de Sequências de Protocolo para o Canal de Colisão sem Realimentação

Recife, Janeiro de 2011.

José Sampaio de Lemos Neto

Construção de Sequências de Protocolo para o Canal de Colisão sem Realimentação

Dissertação submetida ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Pernambuco como parte dos requisitos para obtenção do grau de **Mestre em Engenharia Elétrica**

Orientador: Prof. Valdemar Cardoso da Rocha Júnior, Ph.D.

Recife, Janeiro de 2011.

Catalogação na fonte
Rosineide Mesquita Gonçalves da Luz – CRB-4/1361 (BCTG/UFPE)

L557c Lemos Neto, José Sampaio de.
Construção de Sequências de Protocolo para o Canal de Colisão sem Realimentação / José Sampaio de Lemos Neto – Recife: O Autor, 2011.
96f., il., figs., gráfs., tabs.

Orientador: Prof. Valdemar Cardoso da Rocha Júnior Ph.D.

Dissertação (Mestrado) – Universidade Federal de Pernambuco. CTG. Programa de Pós-Graduação em Engenharia Elétrica, 2012.
Inclui Referências.

1. Engenharia Elétrica. 2. Canais de Comunicação. 3.Modelos de Canal. 4. Canais Multiusuários. 5. Códigos Corretores de Erro. 6. Códigos de Bloco. I. Rocha Júnior, Valdemar Cardoso. (Orientador). II. Título.

621.3 CDD (22.ed) UFPE/BCTG-2012 / 113

Universidade Federal de Pernambuco
Pós-Graduação em Engenharia Elétrica

PARECER DA COMISSÃO EXAMINADORA DE DEFESA DE DISSERTAÇÃO DO MESTRADO ACADÊMICO DE

JOSÉ SAMPAIO DE LEMOS NETO

TÍTULO

"CONSTRUÇÃO DE SEQUÊNCIAS DE PROTOCOLO PARA O CANAL DE COLISÃO SEM REALIMENTAÇÃO"

A comissão examinadora composta pelos professores: VALDEMAR CARDOSO DA ROCHA JÚNIOR, DES/UFPE, CECÍLIO JOSÉ LINS PIMENTEL, DES/UFPE e MARCELO SAMPAIO DE ALENCAR, DEE/UFCG sob a presidência do primeiro, consideram o candidato **JOSÉ SAMPAIO DE LEMOS NETO APROVADO.**

Recife, 28 de janeiro de 2011

RAFAEL DUEIRE LINS
Coordenador do PPGEE

VALDEMAR CARDOSO DA ROCHA JÚNIOR
Orientador e Membro Titular Interno

MARCELO SAMPAIO DE ALENCAR
Membro Titular Externo

CECÍLIO JOSÉ LINS PIMENTEL
Membro Titular Interno

Aos meus pais,

José Sampaio Filho e

Maria de Lourdes

Agradecimentos

Agradeço, primeiramente, a Deus por sempre iluminar meus pensamentos e, desta forma, permitir que eu sempre supere as dificuldades que encontro no decorrer da minha vida.

Agradeço as minhas maiores dádivas: José Sampaio Filho e Maria de Lourdes, pois o amor e o apoio incondicional deles me motivam a sempre lutar para realizar meus sonhos. Agradeço a Rosilda Silva, por sempre estar ao meu lado, principalmente, nos momentos mais difíceis. Agradeço também a meu irmão, José Michelângelo, pelo apoio no início do mestrado.

Em especial, agradeço ao meu orientador, Prof. Valdemar da Rocha, por sua dedicação, incentivo, amizade e, principalmente, por ter acreditado no meu potencial para realizar este trabalho. Além do mais, por ser um exemplo de professor e pesquisador, o qual pretendo seguir.

Agradeço, também, aos professores do Departamento de Eletrônica e Sistemas (DES), em especial aos professores do grupo de Comunicações: Ricardo Campello, Márcia Mahon, Cecílio Pimentel e Hélio Magalhães pelas disciplinas ministradas durante a graduação e pós-graduação, além dos exemplos de competência e dedicação.

Aos amigos de graduação e pós-graduação pela amizade, companheirismo e incentivo. Em especial a Frederico Basto, Raffaello Bruno, Júlio Jansen, Jairo Amaral, Carolina Bastos, Alinson Clementino, Daniel Façanha, Igor Gouveia, Eric Bouton, Marilú Gomes, Caio Marcelo, Paulo Freitas, Paulo Martins, Maurício Cordeiro e Daniel Simões.

Por fim, ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) e ao Programa de Pós-Graduação em Engenharia Elétrica pelo apoio financeiro.

José Sampaio de Lemos Neto

Universidade Federal de Pernambuco

28 de Janeiro de 2011

O único lugar onde o sucesso vem antes do trabalho é no dicionário.

— Albert Einstein

Resumo da Dissertação apresentada à UFPE como parte dos requisitos necessários para a obtenção do grau de Mestre em Engenharia Elétrica

Construção de Sequências de Protocolo para o Canal de Colisão sem Realimentação

José Sampaio de Lemos Neto

Janeiro/2011

Orientador: Prof. Valdemar Cardoso da Rocha Júnior, Ph.D.
Área de Concentração: Comunicações
Palavras-chaves: Canais de comunicação, Modelos de canal, Canais multiusuários, Códigos corretores de Erro, Códigos de bloco.
Número de páginas: 98

O canal de colisão sem realimentação foi introduzido em 1982 por J. L. Massey. É um modelo de canal proposto para situações em que um dado número de usuários compartilha o mesmo canal de comunicação, mas devido à falta de sincronização entre seus relógios e de um elo de realimentação, eles não podem enviar suas mensagens em um modo de transmissão por divisão de tempo (TDMA). Além do mais, a ausência deste elo de realimentação não permite que os usuários tenham informação sobre as mensagens enviadas. Neste modelo de canal, cada usuário possui uma sequência de protocolo que determina quando é permitido utilizar o canal e que é independente dos dados a serem enviados. Em 1992, Nguyen Q. A, L. Györfi e J. L. Massey mostraram que códigos ciclicamente permutáveis constituem uma solução natural para construção de sequências de protocolo para o canal de colisão sem realimentação quando um subconjunto de M usuários, de um total de U usuários, estão ativos num dado intervalo de tempo. As sequências de protocolo propostas nesta dissertação são construídas a partir de um subconjunto de palavras de um código constacíclico q-ário que possuem ordem constacíclica plena. Utilizando uma representação cíclica dos elementos de $GF(q)$ em N-uplas binárias, cada palavra-código do subconjunto selecionado é mapeada em um arranjo bidimensional o qual, quando convertido adequadamente em um vetor binário, produz o dicionário de um código ciclicamente permutável.

Abstract of Dissertation presented to UFPE as a partial fulfillment of the requirements for the degree of Master in Electrical Engineering

CONSTRUCTION OF PROTOCOL SEQUENCES FOR THE COLLISION CHANNEL WITHOUT FEEDBACK

José Sampaio de Lemos Neto

January/2011

Supervisor: Prof. Valdemar Cardoso da Rocha Júnior, Ph.D.
Area of Concentration: Communications
Keywords: Communication channels, Channel models, Multiuser channels, Error correction codes, Block codes.
Number of pages: 98

The collision channel without feedback was introduced in 1982 by J. L. Massey. It is a channel model for situations in which a given number of users share the same communication channel, but due to the lack of synchronization between their clocks and the lack of a feedback link, they cannot send their messages in a time-sharing mode (TDMA). Moreover, the lack of this feedback link does not allow users to have information about the messages sent. In this channel model, each user has a protocol sequence that determines when it is permitted use the channel and is independent of the data to be sent. In 1992, Nguyen Q. A, L. Gyorfi and J. L. Massey showed that cyclically permutable codes are a natural solution to the construction of protocol sequences for the collision channel without feedback when M users out of U are active at some time interval. The protocol sequences proposed in this dissertation are constructed from a subset of codewords of a q-ary constacyclic code, the codewords of which have full constacyclic order. Using a cyclic representation of the elements of $GF(q)$ as binary N-tuples, each codeword of the selected subset is mapped into a two-dimensional array which, when properly converted into a binary vector, produces the codewords of a cyclically permutable code.

Lista de Figuras

Lista de Tabelas

Sumário

CAPÍTULO I

INTRODUÇÃO

Não se pode ensinar tudo a alguém, pode-se,
apenas, ajudá-lo a encontrar por si mesmo.

— Galileu Galilei

ESTE capítulo tem por objetivo apresentar alguns conceitos que são discutidos ao longo desta dissertação. Inicialmente, são abordados sistemas de comunicação digital que envolvem um emissor e um destinatário. Depois, esta situação é generalizada para casos que envolvem vários emissores e um ou mais receptores. Algumas técnicas utilizadas para tornar a transmissão de dados possível nestes casos são mostradas. Posteriormente, apresenta-se a motivação e o objetivo do trabalho proposto nesta dissertação. Por fim, é dada uma rápida descrição do conteúdo dos capítulos.

1.1 SISTEMAS DE COMUNICAÇÃO DIGITAL

1.1.1 SISTEMAS DE COMUNICAÇÃO PONTO-A-PONTO

Um sistema de comunicação digital tem por objetivo transportar os dados de uma fonte de informação (usuário **A**) até um destinatário (usuário **B**). O sistema é denominado *digital* pelo fato de que a informação é representada por meio de um alfabeto finito[1] de símbolos,

[1] Ou, no máximo, infinito contável.

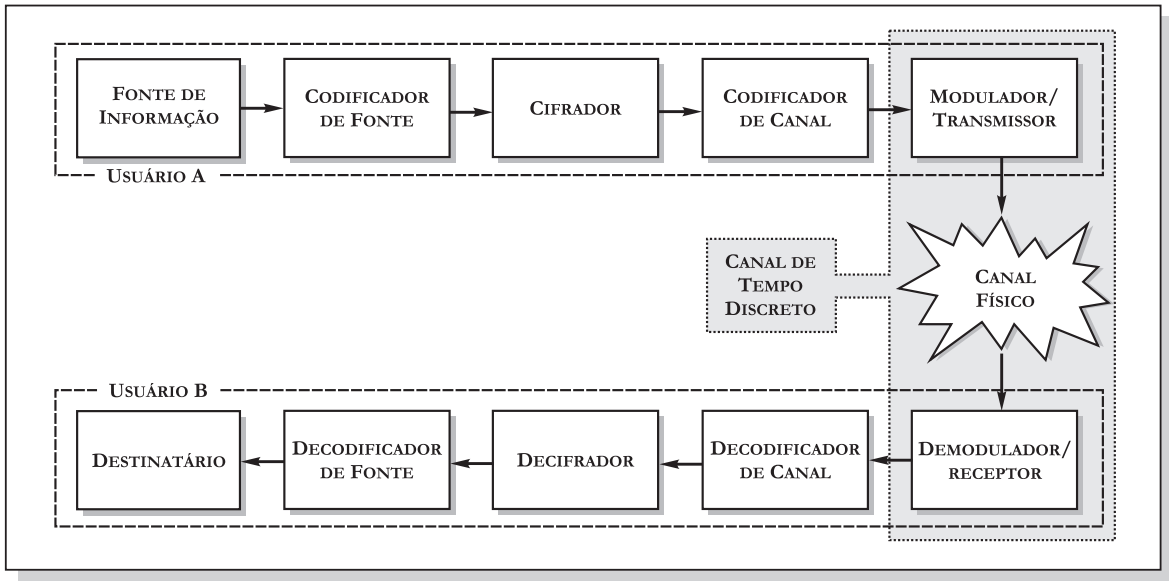


Figura 1.1: *Diagrama de blocos de um típico sistema de comunicação digital.*

sendo esta a diferença básica com relação a sistemas de comunicação analógicos, em que as mensagens são representadas por um alfabeto cujos símbolos variam continuamente em certo intervalo [1]. Desde a publicação dos trabalhos de Shannon [2], [3] e Hamming [4], e mais recentemente devido à evolução da tecnologia de circuitos integrados de larga escala, técnicas digitais têm substituído técnicas analógicas em sistemas de comunicação. Ao utilizar a informação em formato digital, habilita-se o uso de técnicas poderosas de processamento digital de sinais, incluindo o uso da codificação de fonte, da criptografia da informação transmitida e dos códigos corretores de erro. A Figura 1.1 mostra, por meio de um diagrama de blocos, um típico sistema de comunicação digital[2]. A seguir, é dada uma breve descrição das funções desempenhadas por cada um dos blocos da Figura 1.1.

A *fonte de informação* gera a informação a ser transmitida seja ela texto, voz ou imagens. Por exemplo, o sistema de telefonia móvel é um sistema de comunicação digital cujo principal objetivo é transmitir informação em formato de voz. O bloco que representa a fonte de informação, neste caso, é composto (1) pelo ser humano que gera a mensagem a ser transmitida, (2) pelo microfone que converte a voz em sinais elétricos (transdutor), e (3) por um conversor analógico-digital que converte a informação para o formato digital (os dois últimos, componentes de um aparelho móvel digital). Como as fontes de informação digital são representadas

[2] É comum o uso dos blocos Cifrador/Decifrador em aplicações que exigem uma segurança quanto ao conteúdo da informação transmitida. Em geral, eles são omitidos nos diagramas.

por um alfabeto finito, elas podem ser caracterizadas pela distribuição de probabilidade dos símbolos deste alfabeto e, portanto, a informação produzida pode ser medida por meio da *entropia da fonte* [5]. Por fim, a sequência de símbolos é emitida pela fonte a uma taxa média de R_s símbolos por segundo.

O *codificador de fonte* é utilizado para remover a redundância não-controlada que é naturalmente produzida pela fonte de informação. Em outras palavras, esse bloco minimiza o número médio de bits necessários para representar os símbolos emitidos pela fonte, de modo que toda a informação seja preservada. Além do mais, a codificação utilizada deve permitir que o destinatário seja capaz de recuperar a mensagem original sem ambiguidade, trabalho a ser realizado pelo *decodificador de fonte*. Para mais detalhes sobre este assunto, recomenda-se as referências [5]–[7].

O bloco *cifrador* da Figura 1.1 criptografa a informação transmitida de modo que só o destinatário seja capaz de entendê-la. Uma introdução a este assunto pode ser encontrada na referência [8].

O *codificador de canal* tem como objetivo inserir redundância controlada na sequência de informação, proveniente dos blocos anteriores, de tal forma que ao chegar no destino, o *decodificador de canal* seja capaz de detectar e possivelmente corrigir erros que surjam durante a transmissão. Em outras palavras, a informação transmitida torna-se mais imune aos efeitos do ruído oriundo do *canal físico* (ar, fibra ótica, fio metálico, etc.). Os códigos utilizados pelo codificador de canal são conhecidos como *códigos corretores de erro*. Eles são abordados no Capítulo 3 e podem ser consultados nas referências [9]–[14].

Ao inserir codificadores de canal em um sistema de comunicação digital, pode-se interpretar a trajetória percorrida pela informação a partir do bloco *codificador de canal* até chegar ao bloco *decodificador de canal* como um canal no qual é possível transmitir informação de forma suficientemente confiável. Tal canal é denominado *canal de tempo discreto* e está destacado na Figura 1.1. Uma vez que é possível utilizar modelos matemáticos para representar o *canal físico* [15], [16], a capacidade do canal de tempo discreto pode ser calculada por meio dos resultados estabelecidos pela teoria da informação [5]–[7].

O *modulador* tem a função de mapear os símbolos discretos emitidos pelo codificador de canal em formas de onda apropriadas para transmissão por um canal físico. Existem vários tipos de modulação digital que permitem diferentes desempenhos ao sistema [15], [16] e cuja escolha depende da aplicação. Na maioria dos casos, porém, a escolha da modulação para

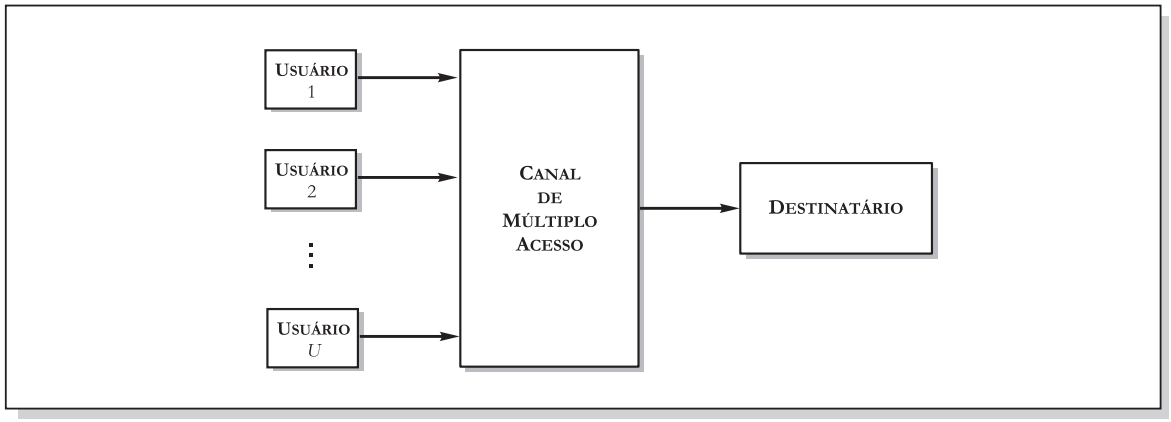


Figura 1.2: *Diagrama de blocos de um sistema de múltiplo acesso.*

um sistema de comunicação digital é limitada por questões de economia de energia ou da disponibilidade da largura de banda. Por exemplo, em um sistema de comunicação via satélite, a modulação escolhida visa minimizar a energia utilizada pelos receptores-transmissores localizados no satélite, ao passo que em um sistema de telefonia móvel, a modulação escolhida visa minimizar a largura de faixa utilizada por cada usuário [10].

Para finalizar a descrição do modelo do sistema digital da Figura 1.1, destaca-se que o projeto do *demodulador* e do *decodificador de canal* é, em geral, mais complexo que o projeto dos respectivos, *modulador* e *codificador de canal*. Projetos de demoduladores podem ser encontrados em [15], [16], enquanto decodificadores de canal podem ser encontrados em [9]–[14]. Existem também projetos que contemplam uma junção entre os blocos *codificador de canal* e *modulador*, e os respectivos *decodificador de canal* e *demodulador* (por exemplo, sistemas TCM), os quais podem ser encontrados em [16].

1.1.2 Sistemas de Múltiplo Acesso

O sistema de comunicação digital da Figura 1.1 é utilizado para modelar situações em que um único emissor (usuário **A**) transmite informação para um único destinatário (usuário B). Sendo assim, este modelo deve ser generalizado para representar sistemas de comunicação mais complexos que envolvem múltiplos emissores e múltiplos destinatários. Sistemas de comunicação envolvendo mais de um usuário e um ou mais destinatários são conhecidos como *sistemas de comunicação multiusuários* e foram inicialmente estudados por Shannon [17]. Há vários modelos de sistemas de comunicação multiusuário [7], [16]. Nesta dissertação, o interesse é no modelo específico de sistema de comunicação multiusuário em que vários

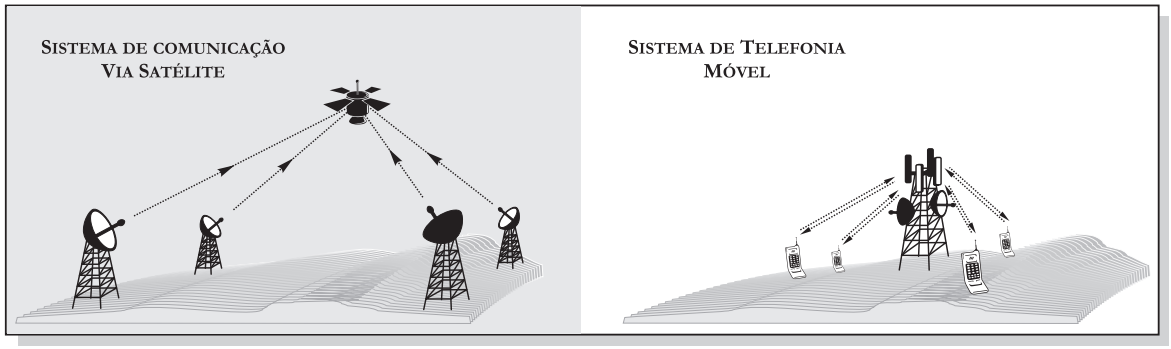


Figura 1.3: *Exemplos de canais de múltiplo acesso.*

emissores (usuários) compartilham o mesmo canal de comunicação e enviam mensagens para um único destinatário (receptor). Esses sistemas são conhecidos na literatura por *sistemas de múltiplo acesso* [16] e o canal de comunicação usado por eles é denominado *canal de múltiplo acesso* [7]. A Figura 1.2 ilustra esta situação. Exemplos de canais de múltiplo acesso, em situações práticas, aparecem em comunicação via satélite e em telefonia móvel, em que vários usuários (estações terrestres e telefones celulares), enviam sinais para o mesmo destinatário (satélites ou estações base). Esses exemplos são ilustrados na Figura 1.3. Nesses casos, além do ruído inserido pelo canal físico sobre os sinais transmitidos, cada usuário deve preocupar-se com a interferência causada pelos sinais emitidos pelos outros usuários que compartilham o canal. Alguns modelos teóricos de canais de múltiplo acesso podem ser encontrados em [18].

Técnicas de múltiplo acesso

Há várias técnicas que permitem dois ou mais usuários compartilharem o mesmo canal de comunicação [1], [7], [16], [18], [19]. Entretanto, as principais técnicas utilizadas em canais de múltiplo acesso são: múltiplo acesso por divisão de frequência (FDMA – *Frequency Division Multiple Access*), múltiplo acesso por divisão de tempo (TDMA – *Time Division Multiple Access*) e múltiplo acesso por divisão em códigos (CDMA – *Code-Division Multiple Access*).

Em sistemas FDMA, a banda disponível é dividida em subcanais. Cada usuário transmite seus dados em um subcanal exclusivo. Se um determinado usuário não está transmitindo, isto é, está inativo durante um certo intervalo de tempo, outros usuários não podem utilizar o subcanal deste usuário para transmitir, o que diminui a eficiência de transmissão do sistema.

Em sistemas TDMA, cada usuário só pode transmitir ou receber dados em intervalos de tempo bem definidos e exclusivos de cada um. Analogamente ao que ocorre em sistemas

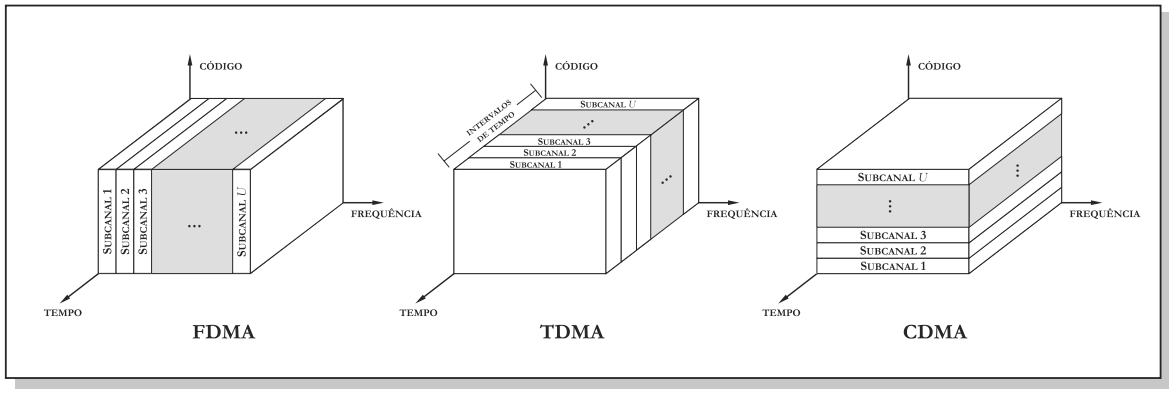


Figura 1.4: *Comparação entre as técnicas de múltiplo acesso FDMA, TDMA e CDMA.*

FDMA, em que os subcanais inativos não podem ser usados por outros usuários, em sistemas TDMA, intervalos de tempo inativos também não podem ser utilizados para aumentar a eficiência de transmissão do sistema. Sistemas TDMA necessitam que os usuários estejam com seus relógios de transmissão sincronizados para evitar que mais de um usuário transmita em um mesmo intervalo de tempo e, além disso, alguns intervalos de tempo são utilizados como proteção para separar os intervalos de transmissão de dois usuários. Essas especificações aumentam a complexidade de implementação de sistemas TDMA.

Por fim, em sistemas CDMA cada usuário recebe uma *sequência de assinatura*, a qual permite que o sinal utilizado por cada usuário para transmitir ocupe toda largura de banda disponível do canal. Nesse caso, o destinatário da Figura 1.2 distingue os sinais enviados pelos vários usuários realizando uma *correlação* [15], [16] entre o sinal recebido e a sequência de assinatura do usuário, identificando como emissor aquele cuja sequência de assinatura maximiza o valor da correlação.

As três técnicas de múltiplo acesso (FDMA, TDMA e CDMA) são ilustradas na Figura 1.4. Por meio da Figura 1.4, permite-se comparar como cada técnica divide o canal de múltiplo acesso. Por conveniência, denomina-se de subcanal cada um dos intervalos de tempo no sistema TDMA e cada uma das sequências de assinatura (códigos) no sistema CDMA.

Técnicas de resolução de colisão

Em alguns sistemas de múltiplo acesso, os usuários que compartilham o canal passam a maior parte do tempo inativos, ou seja, realizam transmissões eventualmente. Assim, utilizar as técnicas FDMA, TDMA ou CDMA acarreta numa baixa relação custo-benefício. Para essas situações, em geral, são utilizadas técnicas conhecidas como *Resolução de colisão* [20]. Essas

técnicas são inspiradas pelo sistema ALOHA proposto por Norman Abramson em 1970 [21]. O objetivo de Abramson era conectar vários terminais de computadores, espalhados pelas diversas ilhas do arquipélago do Havai, ao computador central da universidade. Como os terminais de computadores conectavam-se esporadicamente ao computador central e permaneciam conectados por um período de tempo variável, técnicas tradicionais, como FDMA e TDMA, demandariam mais recursos para implementação além de sobrecarregarem o computador central que frequentemente verificaria os terminais por eventuais transmissões.

A ideia básica proposta no sistema ALOHA é descrita a seguir. Os terminais dividem suas mensagens em blocos de comprimento fixo, denominados *pacotes*, e os transmitem quando disponíveis. Se durante o intervalo de tempo em que o terminal i transmite seus pacotes, nenhum outro terminal está transmitindo simultaneamente, então os pacotes do terminal i são recebidos corretamente pelo computador central e este informa ao terminal i que a operação foi realizada com sucesso. Porém, se outros usuários transmitem simultaneamente com o terminal i, então os pacotes colidem e são descartados pelo computador central. Nestas situações, o computador central informa a todos os terminais que houve colisão e os usuários que tiveram seus pacotes descartados devem esperar, cada um deles, um tempo aleatório para realizar nova transmissão. O tempo de retransmissão deve ser aleatório para evitar que os usuários envolvidos na colisão continuem a ter pacotes colidindo indefinidamente. Nessas condições, Abramson [21] mostra que a máxima vazão de pacotes é de $1/2e \approx 0,184$. Em outras palavras, o valor de 18% representa a porcentagem de pacotes que são recebidos com sucesso, sendo este o melhor desempenho obtido pelo sistema ALOHA.

O sistema proposto por Abramson passou a ser denominado ALOHA puro (*pure* ALOHA) depois que Roberts [22] propôs um modo de melhorar o desempenho do sistema criado por Abramson. A diferença fundamental, proposta por Roberts, é particionar o tempo em intervalos discretos de duração fixa de modo que cada intervalo de tempo corresponda ao tempo necessário para transmissão de um pacote, cujo comprimento também é fixo. O sistema pretendido por Roberts é denominado ALOHA particionado (*slotted* ALOHA) e a vazão máxima obtida por ele é o dobro do ALOHA puro, ou seja, $1/e \approx 0,368$.

Como é citado por Gallager em [20], a estratégia básica utilizada pelo sistema ALOHA, puro ou particionado, foi melhorada, generalizada e analisada de diferentes modos ao longo dos anos. Entre elas, uma em especial, o *canal de colisão sem realimentação* introduzido por Massey [23] e descrito detalhadamente por Massey e Mathys [24]. Uma característica impor-

tante desse modelo é a ausência de um elo de realimentação que, por exemplo, impossibilita os usuários de receber avisos do receptor sobre os pacotes perdidos em colisões. Outra característica importante é a estratégia pela qual os usuários compartilham o canal. Cada usuário recebe uma *sequência de protocolo* determinística que determina os intervalos de tempo em que eles podem transmitir seus pacotes. No Capítulo 2 desta dissertação, é apresentado o canal de colisão sem realimentação. São apresentados o modelo básico do canal e as restrições ao uso do canal que, segundo Massey e Mathys [24], são necessários para uma descrição completa do modelo.

1.2 MOTIVAÇÃO

Com os recentes desenvolvimentos em redes *ad hoc*, redes de sensores e sistemas de identificação por rádio frequência (RFID), são desejáveis protocolos simples que não exijam um monitoramento frequente do canal. Redes de sensores, por exemplo, representam um desafio interessante. Além de possuir um grande número de dispositivos distribuídos em uma topologia de rede que muda dinamicamente, os sensores, em geral, possuem restrições quanto ao tamanho e quanto ao consumo de energia [25].

O modelo do canal de colisão sem realimentação, proposto por Massey [23], [24], mostra-se adequado para ser utilizado nas aplicações citadas anteriormente. Em [24], Massey e Mathys propuseram o uso de *sequências de protocolo* para definir quando os usuários podem transmitir pacotes pelo canal de colisão sem realimentação. Posteriormente, A, Györfi e Massey [26] fizeram um contribuição importante mostrando que as palavras de um *código ciclicamente permutável* [27], de peso constante, podem ser usadas como sequências de protocolo. Um outro modo de construir sequências de protocolo, baseado no conceito de sequência de números primos, foi proposto por Wong [28].

Trabalhos relacionados à construção de sequências de protocolo para o canal de colisão sem realimentação podem ser encontrados nas referências [29]–[31]. Trabalhos recentes podem ser vistos nas referências [32]–[36].

1.3 OBJETIVOS

O principal objetivo desta dissertação é propor uma maneira original de construir sequências de protocolo para o canal de colisão sem realimentação. O trabalho é desenvolvido

baseado na ideia proposta por A, Györfi e Massey [26] de que códigos ciclicamente permutáveis [27] constituem uma solução natural para construção de sequências de protocolo para esse canal. A proposta desta dissertação, entretanto, utiliza uma classe de códigos corretores de erro diferente da que foi usada em [26]. É utilizada a classe de códigos constacíclicos [12] que ainda não haviam sido explorados neste contexto. Um método para associar arranjos bidimensionais e N-uplas, distinto daquele proposto em [26], é utilizado. No desenvolvimento para construção de sequências de protocolo, são propostos novos métodos para construção de códigos cíclicos binários não-lineares.

1.4 Organização da Dissertação

O conteúdo desta dissertação está dividido em seis capítulos. As referências encontram-se nas páginas finais e são ordenadas de acordo com a ordem em que foram citadas no texto. A seguir, um resumo dos capítulos seguintes da dissertação.

Capítulo 2. O objetivo principal deste capítulo é apresentar o canal de colisão sem realimentação introduzido por Massey [23] e descrito detalhadamente por Massey e Mathys [24]. O canal é apresentado por meio do seu modelo básico e das restrições ao seu uso. No que diz respeito às restrições ao uso do canal são abordadas as sequências de protocolo utilizadas pelos usuários do canal, as quais determinam quando eles estão aptos a utilizá-lo. A capacidade do canal é apresentada e um caso particular do uso do modelo é descrito.

Capítulo 3. Neste capítulo, os códigos utilizados pelo bloco *codificador de canal* da Figura 1.1 são apresentados. Embora o objetivo não seja utilizá-los no contexto de codificação de canal, são introduzidos os conceitos básicos destes códigos. São discutidos códigos de bloco e as propriedades que podem ser aproveitadas ao se utilizar uma estrutura algébrica para estes códigos. Após a introdução, é estudada a classe de códigos constacíclicos [12] que são explorados na construção de códigos cíclicos binários não-lineares e na construção das sequências de protocolo (Capítulo 2).

Capítulo 4. Após a apresentação dos códigos corretores de erro, é apresentada, neste capítulo, uma classe de códigos introduzida por Gilbert [27], os códigos ciclicamente permutáveis (códigos CP). Os códigos CP foram utilizados por A, Györfi e Massey [26] para construir as sequências de protocolo para o canal de colisão sem realimentação. Para construir

os códigos CP propostos nesta dissertação, são utilizados os códigos constacíclicos do Capítulo 3 juntamente com uma representação cíclica para os elementos de um corpo finito [9], [10] e uma relação adequada para converter arranjos bidimensionais em N-uplas binárias.

Capítulo 5. Uma vez que as construções de códigos CP foram propostas no Capítulo 4, este capítulo mostra tais construções aplicadas como sequências de protocolo. O desempenho das sequências de protocolo propostas nesta dissertação é analisado e comparado ao desempenho de outras sequências de protocolo [26], [29], também construídas por meio de códigos CP.

Capítulo 6. Este capítulo é dedicado à conclusão da dissertação juntamente com as contribuições e sugestões para futuras investigações.

Capítulo 2

O Canal de Colisão sem Realimentação

A teoria é o general; os experimentos são os soldados.

— Leonardo da Vinci

Conforme mencionado no Capítulo 1, modelos de canais de múltiplo acesso são utilizados para situações em que vários usuários compartilham o mesmo canal de transmissão para enviarem informação a um único receptor. Neste capítulo apresenta-se um modelo de canal de múltiplo acesso introduzido por Massey [23], descrito detalhadamente por Massey e Mathys [24] e denominado *canal de colisão sem realimentação*. Esse canal é apresentado por meio do seu modelo básico e das restrições ao seu uso. No que diz respeito às restrições ao uso do canal, são abordadas as sequências de protocolo que determinam quando os usuários do canal estão aptos a utilizá-lo. As regiões de capacidade desse canal são apresentadas e um caso particular do uso do modelo é descrito.

2.1 O Canal de Colisão sem Realimentação

O canal de colisão sem realimentação [24] é um modelo proposto para situações em que um dado número de usuários compartilha o mesmo canal de comunicação, mas devido à

defasagem entre seus relógios e de um elo de realimentação, eles não podem enviar suas mensagens em um modo de transmissão por divisão de tempo (TDMA). Além do mais, a ausência desse elo de realimentação impede que os usuários tenham alguma informação sobre as mensagens enviadas. Sendo assim, o receptor não é capaz de solicitar ao usuário o reenvio de uma determinada mensagem perdida em uma colisão, por exemplo. Neste modelo de canal, cada usuário possui uma sequência de protocolo que determina quando ele está apto a utilizar o canal e que é independente dos dados a serem enviados.

Nesta seção, é discutido o modelo teórico para o canal de colisão sem realimentação. De acordo com [24], modelos de canais, em geral, possuem duas características distintas e necessárias para especificar por completo o modelo:

i. O *modelo básico* do canal, que especifica as regras, probabilísticas ou determinísticas, que estabelecem a transição entre as entradas e as saídas do canal;

ii. As *restrições* ao uso do canal.

2.1.1 O Modelo Básico

Para dar início à discussão sobre o modelo teórico do canal de colisão sem realimentação [24], ilustra-se na Figura 2.1 o modelo básico do canal por meio de um diagrama de blocos. O objetivo é reproduzir uma situação em que um número M de *usuários* compartilha o mesmo canal de comunicação. Nesta situação, os pacotes emitidos por cada um dos M usuários podem assumir q valores distintos, sendo q um número inteiro tal que $q \geq 2$. Sendo assim, cada pacote pode conter até $\log_2 q$ bits de informação. Além do mais, define-se que cada um desses pacotes de informação possui uma duração fixa de T segundos.

Diz-se que um usuário está *ativo*, em um dado intervalo de tempo, quando ele está lançando seus pacotes no canal, caso contrário, o usuário está *em silêncio*. As formas de onda utilizadas por cada usuário, para transmitir seus pacotes pelo canal, são representadas na Figura 2.1 por $x_i(t), i = 1, 2, 3, \ldots, M$. Quando o usuário i está ativo, $x_i(t)$ é uma forma de onda apropriada com duração de T segundos (de acordo com a duração de tempo estabelecida para os pacotes de informação). Entretanto, quando o usuário está silencioso, $x_i(t)$ é um sinal nulo, ou seja, assume o valor nulo durante os T segundos estabelecidos.

Conforme é dito no início desta seção, neste modelo de canal, o relógio de cada usuário está, em princípio, fora de fase com o relógio dos outros usuários. Essa situação é reproduzida no modelo da Figura 2.1 pelos *deslocamentos de tempo*, δ_i, atribuídos aos sinais $x_i(t)$ emitidos

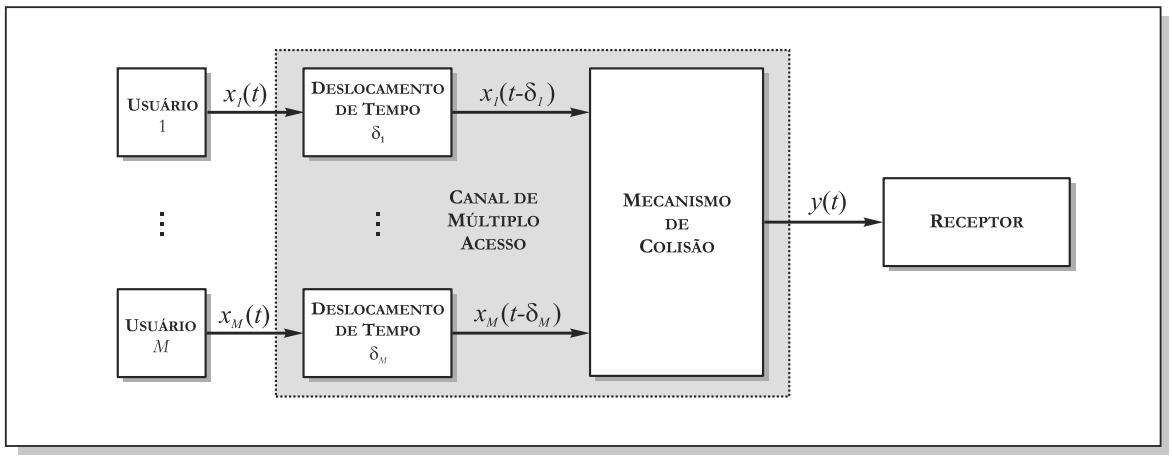


Figura 2.1: *Diagrama de blocos para o modelo básico do canal de colisão sem realimentação.*

pelos usuários. Em geral, cada usuário i possui um valor diferente para δ_i. O deslocamento de tempo δ_i pode ser interpretado como o tempo de propagação que o sinal de cada usuário leva para chegar ao receptor. Porém, o ponto chave no modelo é que o usuário i não tenha conhecimento prévio do valor do seu próprio deslocamento de tempo δ_i nem dos deslocamentos de tempo δ_j, $j \neq i$, dos outros $M-1$ usuários. Além do mais, esses valores não podem ser determinados, à medida que os usuários utilizam o canal, devido à falta de um elo de realimentação entre o canal e os usuários. O receptor também desconhece, previamente, o deslocamento de tempo dos sinais emitidos pelos usuários.

Continuando sobre a análise do modelo do canal da Figura 2.1, define-se em que situação ocorre *colisão* entre os pacotes emitidos por dois ou mais usuários. Seja P_i um pacote enviado pelo usuário i e P_j um pacote enviado pelo usuário j. Lembrando que os pacotes duram um tempo fixo de T segundos e admitindo que os pacotes P_i e P_j iniciaram, respectivamente, nos tempos t_i e t_j, então há colisão entre os pacotes P_i e P_j se e somente se

$$|(t_i - \delta_i) - (t_j - \delta_j)| < T, \tag{2.1}$$

ou seja, se a diferença de tempo, na recepção, entre suas bordas iniciais for menor que o tempo T de duração de um pacote. Na Desigualdade (2.1), o relógio do receptor é utilizado como referência. É importante ressaltar que, para a descrição do modelo ser precisa, o receptor tem de ser capaz de reconhecer pacotes que não colidiram e que estão precisamente adjacentes.

A saída $y(t)$ do *mecanismo de colisão* corresponde a uma das seguintes situações:

a. $y(t)$ é uma forma de onda que está associada a um pacote de informação que não sofreu

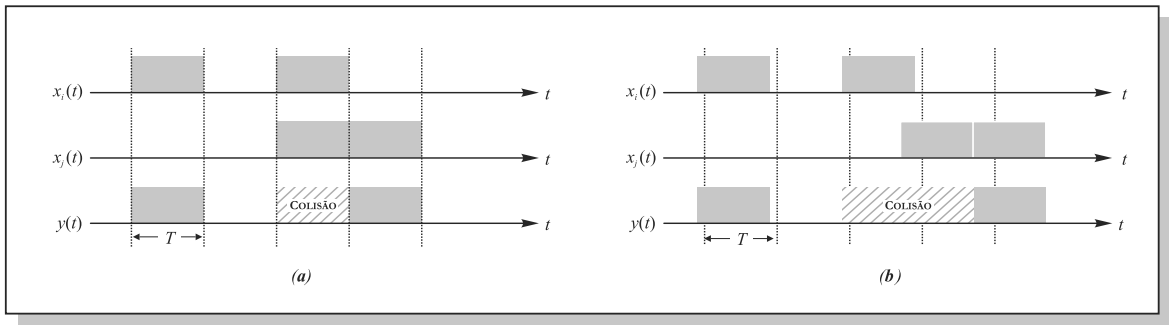


Figura 2.2: *(**a**) No modo de operação síncrono os pacotes sobrepõem-se completamente. (**b**) No modo de operação não-síncrono os pacotes sobrepõem-se parcialmente. Em ambos, o relógio do receptor é usado como referência.*

colisão;

b. $y(t)$ é uma forma de onda corrompida devido à colisão entre dois ou mais pacotes;

c. $y(t)$ é o sinal nulo, indicando um intervalo de silêncio em que pacotes, com colisão ou sem colisão, não foram recebidos.

Para finalizar a descrição do modelo básico do canal, são definidos os dois modos de operação utilizados no modelo. Em ambos os casos, os usuários particionam o tempo dos seus respectivos relógios em intervalos de tempo de duração T e os seus pacotes são transmitidos alinhados com estes intervalos. A diferença entres os dois modos de operação está, então, associada aos valores assumidos pelos deslocamentos de tempo δ_i. Na primeira situação, denominada *modo de operação com intervalos de tempo sincronizados*, δ_i só assume valores que são múltiplos inteiros de T (tempo de duração de cada pacote). Na outra situação, denominada *modo de operação com intervalos de tempo não-sincronizados*, δ_i assume valores arbitrários pertencentes ao conjunto dos números reais. Por simplicidade, faz-se referência a estes modos de operação, respectivamente, como *modo sincronizado* e *modo não-sincronizado*. Definindo o n-ésimo intervalo de tempo como sendo o intervalo semiaberto $nT \leq t < (n+1)T$, se o canal está operando no modo sincronizado e o usuário i emite um pacote no seu n-ésimo intervalo de tempo, então o pacote é recebido precisamente no intervalo de tempo $n+\delta_i/T$ do receptor, tomando como referência o relógio do usuário i. Assim, se todos os usuários transmitem seus pacotes alinhados com os intervalos de tempo de seus relógios, só há colisões quando os pacotes sobrepõem-se totalmente. Entretanto, no modo de operação não-sincronizado, colisões ocorrem mesmo que a sobreposição dos pacotes transmitidos seja parcial. A Figura 2.2 ilustra os dois modos de operação. Os pacotes enviados pelos usuários i e j são representados pelos

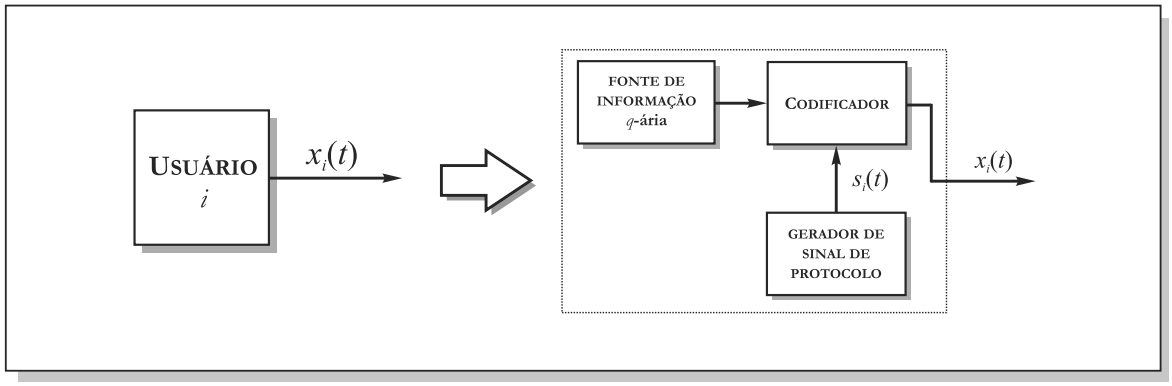


Figura 2.3: *Diagrama de blocos do modelo proposto para cada usuário.*

retângulos em cor cinza. Em ambos, o relógio do receptor é usado como referência. Percebe-se que no modo não-sincronizado os pacotes não estão alinhados com o relógio do receptor devido aos valores assumidos por δ_i não serem múltiplos inteiros de T.

2.1.2 As Restrições ao Uso do Canal

Para completar a descrição do modelo do canal de colisão sem realimentação, falta especificar as restrições ao seu uso. Em outras palavras, se um determinado usuário deseja transmitir no canal de múltiplo acesso ele tem de fazer isto de acordo com as regras estabelecidas para este fim. A Figura 2.3 mostra, por meio de um diagrama de blocos, a estrutura que é utilizada por cada usuário para ter acesso ao modelo básico da Figura 2.1.

Os processos de geração e transmissão de mensagens, executados pelos blocos da Figura 2.3, são descritos a seguir. Os pacotes que são transmitidos correspondem aos símbolos emitidos pela *fonte de informação* q-ária, quando demandada, no n-ésimo intervalo de tempo. Porém, esses pacotes só são codificados e convertidos nas formas de onda $x_i(t)$, trabalho executado pelo bloco *codificador*, dependendo dos sinais $s_i(t)$ emitidos pelo bloco *gerador de sinal de protocolo*. Os valores assumidos por $s_i(t)$ dependem da *sequência de protocolo* usada por cada usuário. As sequências de protocolo são sequências binárias de comprimento N, denotadas por $\mathbf{s}_i = \{s_{il}\}_{l=1}^{N}$, cuja relação com o sinal $s_i(t)$ é tal que $s_i(t) = s_{il}$ para $lT \leq t < (l+1)T$. Para $s_i(t) = 1$, o codificador emite os pacotes, caso contrário, quando $s_i(t) = 0$, o codificador emite o sinal nulo e o usuário fica em silêncio. Os usuários continuam usando periodicamente sua respectiva sequência de protocolo para transmitir até que não tenham mais pacotes para enviar. Após esse tempo de atividade, o usuário i deve permanecer inativo por, no mínimo, $N-1$ intervalos de tempo para poder voltar a emitir pacotes.

Embora o processo descrito anteriormente, de fato, resuma o que ocorre internamente nos blocos de cada usuário, algumas considerações precisam ser dadas. Inicialmente, destaca-se o fato de que a fonte de informação emite símbolos quando demandada, pois isto pode parecer estranho já que em sistemas de acesso aleatório os usuários não utilizam o canal frequentemente. Como é justificado em [24], pode-se interpretar o fato da fonte transmitir quando demandada como sendo o pior caso possível para os usuários utilizarem o canal. Isto é, pode-se presumir uma situação em que, num dado intervalo de tempo, todos os usuários estão ativos e possuem uma fila de pacotes esperando para ser transmitida. A capacidade do canal, calculada a partir dessa hipótese, pode ser vista, então, como o melhor desempenho atingido pelo sistema de múltiplo acesso em uma situação de sobrecarga. Vale destacar que o propósito inicial do cálculo da capacidade, para esse modelo de canal, é determinar o quanto de perda resulta quando M usuários compartilham o mesmo canal de comunicação uma vez que eles são impedidos, devido à ausência de um elo de realimentação, de realizar o compartilhamento por um modo de divisão de tempo.

Dando continuidade às considerações sobre as restrições ao uso do canal, destaca-se o codificador da Figura 2.3. Esse codificador é utilizado para enviar pacotes redundantes junto com os pacotes emitidos pela fonte q-ária de modo que o receptor seja capaz de recuperar os pacotes transmitidos, a partir do sinal recebido $y(t)$, desde que a probabilidade de erro, durante a transmissão, seja suficientemente pequena. Lembrando que o receptor deve realizar esta tarefa para cada um dos M usuários. Vale ressaltar que o bloco *codificador* da Figura 2.3 executa o papel dos blocos *codificador de canal* e *modulador* do sistema de comunicação digital ilustrado na Figura 1.1.

A última consideração a ser feita é sobre as sequências de protocolo. As sequências de protocolo são utilizadas neste modelo para evitar que o desempenho do sistema varie com as propriedades probabilísticas da fonte de informação. Ou seja, a incerteza da fonte de informação não interfere nos intervalos de tempo em que determinado usuário emite seus pacotes. Portanto, é tarefa exclusiva das sequências de protocolo determinar os intervalos de tempo em que os usuários transmitem. O sinal $s_i(t)$, gerado pelo bloco *gerador de sinal de protocolo*, é uma forma de onda predeterminada com período τ_i[1] que assume valores 0 ou 1, para todo t, somente em intervalos de tempo semiabertos com duração de T segundos. E, por definição, quando $s_i(t) = 1$ o codificador emite o pacote no n-ésimo intervalo de

[1] Embora, por conveniência analítica, sejam usados sinais $s_i(t)$ cujo período é finito, não há necessidade de limitar superiormente os possíveis valores para τ_i [24], pois o modelo de canal em questão não possui essa limitação.

tempo, caso contrário, $s_i(t) = 0$, o codificador não emite o pacote e o usuário permanece em silêncio neste intervalo de tempo. Além do mais, pode parecer estranho utilizar sequências de protocolo determinísticas para controlar o acesso de usuários em um modelo de canal cujo sistema de acesso é aleatório. Porém, nada impede que a escolha da sequência de protocolo, mais precisamente do primeiro período de $s_i(t)$, de cada usuário seja feita por um processo aleatório. Por fim, é assumido que cada usuário saiba qual a sequência de protocolo a ser utilizada pelos demais usuários e que, além disso, o receptor tenha conhecimento da escolha feita por cada um.

2.1.3 Regiões de Capacidade

A fim de derivar os resultados desejados para as regiões de capacidade deste canal, assume-se, daqui em diante, que a fonte de informação q-ária da Figura 2.3 é uma *fonte discreta simétrica sem memória* [5] e ela emite informação a uma taxa de R_i *pacotes/unidade de tempo*. Como o usuário i só está apto a utilizar o canal nos intervalos de tempo em que $s_i(t) = 1$, então a *taxa efetiva de transmissão de informação* do usuário i, denotada por r_i *pacotes/unidade de tempo*, difere da taxa R_i e só ocorre igualdade entre elas, $R_i = r_i$, caso o usuário i esteja apto a utilizar o canal em todos os intervalos de tempo, ou seja, $s_i(t) = 1\ \forall t$. Para estabelecer uma relação entres as taxas R_i e r_i, é conveniente definir o *fator de trabalho* p_i como um número racional no intervalo $0 \leq p_i \leq 1$ de modo que ele represente a fração de tempo em que a sequência de protocolo do usuário i assume o valor 1. Desta forma, a relação entre R_i, r_i e p_i é tal que $r_i = R_i/p_i$ e ocorre igualdade caso $p_i = 1$, pois, pela definição do fator de trabalho, isso indica que o usuário está apto a utilizar o canal em todos os intervalos de tempo.

Em geral, *regiões de capacidade* para canais de múltiplo acesso são definidas como o conjunto $\mathcal{C}$ de todos os vetores $\mathbf{R} = (R_1, R_2, \ldots, R_M)$ tal que é possível transmitir, pelo canal, com uma probabilidade de erro arbitrariamente pequena para qualquer vetor $\mathbf{R} \in \mathcal{C}$ e, consequentemente, impossível para qualquer outro vetor $\mathbf{R}' \notin \mathcal{C}$. O termo região é bem aplicado, pois o conjunto $\mathcal{C}$ corresponde a uma região $\mathcal{R}$ do espaço $\mathbb{R}^M$ que contém todos os pontos definidos pelo conjunto de vetores $\mathbf{R}$, ou seja, $\mathcal{R} = \{(R_1, R_2, \ldots, R_M) \in \mathbb{R}^M \mid \mathbf{R} = (R_1, R_2, \ldots, R_M) \in \mathcal{C}\}$. Como $\mathcal{R}$ é uma região no espaço $\mathbb{R}^M$, existe um conjunto de pontos em $\mathcal{R}$ que a delimitam do restante do espaço $\mathbb{R}^M$, ou seja, este conjunto de pontos é o *contorno* da região $\mathcal{R}$. Portanto, o *contorno* de uma região de capacidade $\mathcal{C}$ é definido como o conjunto de vetores $\mathbf{C} \in \mathcal{C}$ tal que não exista nenhum outro vetor $\mathbf{C}' \in \mathcal{C}$ tal que $\mathbf{C} \leq \mathbf{C}'$, em que a

desigualdade entre $\mathbf{C}$ e $\mathbf{C}'$ é avaliada entre as respectivas componentes destes vetores. Daqui em diante, os vetores $\mathbf{C} = (C_1, C_2, \ldots, C_M)$ denotam pontos no contorno de uma região de capacidade qualquer.

Para o modo de operação com intervalos de tempo não-sincronizados, mencionado anteriormente, a região de capacidade do canal de colisão sem realimentação, denotada por $\mathcal{C}_{ns}$, é definida [24] como o conjunto de todos os vetores $\mathbf{R} = (R_1, R_2, \ldots, R_M)$, com $R_i \geq 0$ e $1 \leq i \leq M$, acessíveis[2] no sentido que, dados quaisquer números positivos σ e ϵ, existe uma sequência de protocolo $s_i(t)$ e um código de bloco de comprimento n_i pacotes para cada usuário i tal que

1. blocos de, no mínimo, $(R_i/p_i - \sigma)n_i$ pacotes da fonte de informação do usuário i são codificados em blocos de comprimento n_i pacotes para serem transmitidos durante intervalos de tempo sucessivos nos quais o usuário i realmente usa o canal; e
2. o decodificador é capaz de reconstruir, por meio da saída do canal $y(t)$, a sequência emitida pela fonte de informação do usuário i com uma probabilidade de erro de pacote média de valor no máximo ϵ, desconsiderando os valores dos deslocamentos de tempo $\delta_1, \delta_2, \ldots, \delta_M$.

A *região de capacidade com probabilidade de erro nula* [24] para o canal de colisão sem realimentação operando no modo não-sincronizado, denotada por $\mathcal{C}_{ns0}$, é definida de modo semelhante à definição de $\mathcal{C}_{ns}$ tal que $\epsilon = 0$. As regiões de capacidade para o canal de colisão sem realimentação operando no modo sincronizado podem, também, ser definidas de modo semelhante. Neste caso, denota-se estas regiões de capacidade por $\mathcal{C}_s$ e $\mathcal{C}_{s0}$.

O teorema a seguir estabelece uma importante relação para as regiões de capacidade do canal de colisão sem realimentação.

Teorema 2.1 – Teorema 1 em [24]

Para o canal de colisão sem realimentação com M usuários, $\mathcal{C} \triangleq \mathcal{C}_{ns0} = \mathcal{C}_{ns} = \mathcal{C}_{s0} = \mathcal{C}_s$. Além do mais, o contorno desta região de capacidade $\mathcal{C}$ é o conjunto de todos os vetores $\mathbf{C} = (C_1, C_2, \ldots, C_M)$ tal que

$$C_i = p_i \prod_{\substack{j=1 \\ j \neq i}}^{M} (1 - p_j), \tag{2.2}$$

em que $\mathbf{p} = (p_1, p_2, \ldots, p_M)$ é um vetor probabilidade que satisfaz $p_i \geq 0$ e $\sum_i p_i = 1$ para $1 \leq i \leq M$ e cada vetor $\mathbf{C}$ é determinado por um único vetor $\mathbf{p}$. □

[2] Segundo o conceito de Shannon [2] de taxa acessível em que as regiões de capacidade são sempre conjuntos fechados.

A prova do Teorema 2.1 é uma prova construtiva dada ao longo da referência [24]. Por se tratar de uma prova extensa, ela é omitida.

O Teorema 2.1 afirma que a capacidade $\mathcal{C}$ do canal de colisão sem realimentação independe se o canal está sendo operado no modo sincronizado ou no modo não-sincronizado. Além do mais, ele mostra que existe uma correspondência unívoca entre vetores probabilidade $\mathbf{p}$ e pontos no contorno da região de capacidade $\mathcal{C}$. A região de capacidade $\mathcal{C}$ não é convexa para $M \geq 2$, pois, de acordo com [24], a ausência de um elo de realimentação impede os usuários deste canal de usarem diferentes esquemas de codificação em um modo TDMA.

Para finalizar a discussão sobre as regiões de capacidade para o canal de colisão sem realimentação, trata-se do caso em que todos os usuários emitem informação com a mesma taxa. Nessa situação, define-se a *capacidade simétrica*, denotada por C_{sym}, para o canal de colisão sem realimentação como a taxa máxima r tal que $\mathbf{R} = (r/M, r/M, \ldots, r/M) \in \mathcal{C}$. Desta definição, segue um corolário para o Teorema 2.1.

Corolário 2.1 – Teorema 2.1

Para o modo sincronizado ou não-sincronizado e para uma probabilidade de erro arbitrariamente pequena ou nula, a capacidade simétrica para o canal de colisão sem realimentação é dada por

$$C_{\text{sym}} = \left(1 - \frac{1}{M}\right)^{M-1} \textit{ pacotes/unidade de tempo.} \tag{2.3}$$

Além do mais, o vetor $(C_{\text{sym}}/M, C_{\text{sym}}/M, \ldots, C_{\text{sym}}/M)$ *é realizável no modo de operação sincronizado.* □

2.1.4 Um Caso Particular

Um caso particular para o canal de colisão sem realimentação foi introduzido em [37], [38]. Neste caso, denotando por U o número total de usuários que compartilham o canal, no máximo, M usuários, $M \leq U$, estão ativos e cada *quadro* (do inglês, *frame*) que chega ao receptor é o conjunto de N intervalos de tempo de transmissão, em que cada intervalo possui duração de T segundos correspondendo ao tempo de duração dos pacotes emitidos pelos usuários. Ao considerar que M usuários estão ativos em um determinado quadro, admite-se que cada um deles emite, pelo menos, um pacote durante os N intervalos de transmissão de cada quadro recebido pelo receptor. Os M usuários ativos em um determinado quadro não são necessariamente os mesmos em todos os quadros.

Seja $\mathbf{s}_i$ uma sequência binária de comprimento N e $\{\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3, \ldots, \mathbf{s}_U\}$ seja o conjunto for-

mado por U sequências binárias. O conjunto $\{\mathbf{s}_i\}_{i=1}^{U}$ é denominado um *conjunto de sequências de protocolo*, representado por (U, M, N, σ), se cada uma das sequências binárias do conjunto for utilizada como sequência de protocolo pelos usuários do canal. Além do mais, para qualquer quadro que chegue ao receptor, assume-se que: (1) no máximo M usuários estão ativos, (2) o receptor é capaz de identificar cada um dos usuários ativos e que (3) σ pacotes emitidos por cada um dos usuários ativos chegam ao receptor sem sofrer colisão, no mínimo.

A, Györfi e Massey [26] mostraram que *códigos ciclicamente permutáveis* [27] constituem uma solução natural para o caso particular de acesso múltiplo em que M usuários, de um total de U, estão ativos em um dado quadro. No Capítulo 4 e no Capítulo 5, discute-se códigos ciclicamente permutáveis (códigos CP) e são dadas algumas construções de códigos CP que são aplicadas como sequências de protocolo.

Capítulo 3

Códigos Corretores de Erro

Ciência é fato; assim como casas são feitas de tijolos, ciência é constituída de fatos; mas uma pilha de tijolos não é uma casa assim como uma coleção de fatos não é necessariamente ciência.

— Jules Henri Poincaré

Shannon estabeleceu, por meio do teorema para codificação de canais ruidosos [2], que existem códigos corretores de erro capazes de permitir a transmissão da informação por um canal ruidoso com uma taxa de erro de *bit* arbitrariamente pequena, desde que a taxa de informação transmitida seja menor que uma grandeza definida como capacidade do canal. Embora o teorema em questão garanta a existência desses códigos, ele não mostra como obtê-los. Desde então, um novo ramo surgiu na área de comunicações, os *códigos corretores de erro*. Como mencionado no Capítulo 1, a finalidade dos códigos corretores de erro é inserir redundância de maneira controlada na informação a ser transmitida de modo que o receptor seja capaz de detectar e até mesmo corrigir eventuais erros introduzidos durante a transmissão. Entretanto, nesta dissertação códigos corretores de erro são usados em um contexto distinto. O intuito é utilizá-los como um meio de se construir códigos cíclicos binários não-lineares e códigos ciclicamente permutáveis para serem aplicados como sequências de protocolo para o

canal de colisão sem realimentação [26] apresentado no Capítulo 2.

Neste capítulo, são apresentados os conceitos básicos destes códigos. São discutidos códigos de bloco e as propriedades que podem ser usufruídas ao se utilizar uma estrutura algébrica para eles. Após esta introdução, aborda-se a classe de códigos constacíclicos [12] que são explorados na construção de códigos cíclicos binários não-lineares que, por sua vez, são usados para construir códigos ciclicamente permutáveis os quais são utilizados como sequências de protocolo para o canal de colisão sem realimentação do Capítulo 2. Ao leitor que desejar um aprofundamento sobre o tema, recomenda-se a leitura das referências bibliográficas [9]–[14]. Para um bom entendimento deste capítulo é necessário um conhecimento básico de corpos finitos e álgebra linear. Uma excelente introdução a esses conteúdos pode ser encontrada em [9, Cap. 2] e [10, Caps. 2 e 3].

3.1 Códigos de Bloco

Em princípio, é dada uma definição para códigos de bloco e em seguida são discutidas algumas de suas caraterísticas e propriedades. Nesta dissertação, só são abordados códigos de bloco, de modo que sempre que se utilizar o termo *código*, com relação a códigos corretores de erro, ele está referindo-se a códigos de bloco.

Definição 3.1 – Código de bloco

*Um código de bloco q-ário $\mathcal{C}$ é o conjunto $\{\mathbf{c}_0, \mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_{L-1}\}$ formado por L n-uplas q-árias de comprimento n denotadas por $\mathbf{c}_i = (c_0, c_1, c_2, \ldots, c_{n-1})$, $i = 0, 1, 2, \ldots, L-1$, e denominadas **palavras-código** ou **vetores-código**.* □

Ao código $\mathcal{C}$ da Definição 3.1, é associado um codificador que é responsável pelo mapeamento das mensagens, emitidas por uma fonte de informação q-ária, nas palavras-código $\mathbf{c}_i$. Vale ressaltar que um código $\mathcal{C}$ é único. Entretanto, o codificador não é único [14]. Isto é, o mapeamento entre mensagens e vetores-código pode mudar, mas todos os possíveis vetores obtidos na saída do codificador pertencem a $\mathcal{C}$.

O processo de codificação consiste, primeiramente, em dispor os dados emitidos pela fonte de informação em blocos de comprimento k, em seguida, mapeá-los em palavras-código. Esse mapeamento é um-a-um, para permitir que o destinatário seja capaz de recuperar a mensagem original enviada. Se a fonte de informação emite símbolos de um alfabeto q-ário, então as possíveis mensagens a serem codificadas correspondem a k-uplas, $\mathbf{m} = (m_0, m_1, \ldots, m_{k-1})$,

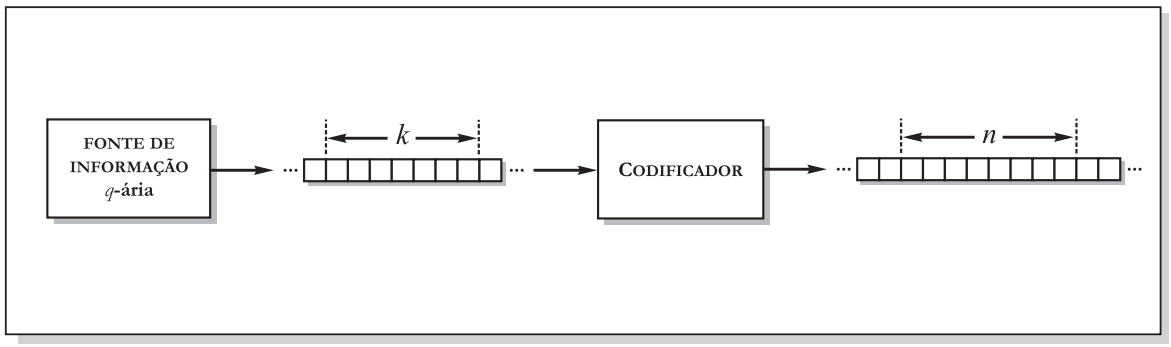


Figura 3.1: *Processo de codificação dos códigos de bloco.*

que formam um espaço vetorial sobre $GF(q)$. Visto que cada vetor $\mathbf{m}$ possui k símbolos e cada símbolo pode assumir q valores distintos, o total de possíveis *vetores-mensagem* é igual a q^k. Desta forma, $L = q^k$. O processo de codificação é ilustrado na Figura 3.1. Entretanto, há situações em que $L \neq q^k$. Nesses casos, a implementação, em geral, torna-se mais complexa. Em [10, pág. 69] mostra-se um exemplo em que o codificador tem de arranjar as mensagens em blocos com comprimento variável para tratar o caso em que $L \neq q^k$.

É analisado, neste ponto, como a utilização dos códigos de bloco permite inserir redundância, de modo controlado, nas mensagens codificadas. O conjunto de todas as n-uplas q-árias de comprimento n constituem o espaço vetorial $\mathbf{V}$ sobre $GF(q)$ contendo um total de q^n vetores. O conjunto de vetores que pertencem a $\mathcal{C}$ é um subconjunto de $\mathbf{V}$, portanto existem vetores de $\mathbf{V}$ que não estão em $\mathcal{C}$. Diz-se, então, que um código de bloco possui *redundância* quando o número de vetores que pertencem ao código é menor que o número total de vetores q-ários de comprimento n, ou seja, $L < q^n$. A redundância r pode ser expressa em forma logarítmica [10] por

$$r = n - \log_q L. \tag{3.1}$$

Em geral, utilizam-se códigos em que $L = q^k$. Portanto, a Fórmula (3.1) torna-se $r = n - k$. Ou seja, dos n símbolos transmitidos, k símbolos são de informação e o restante, $n-k$, são de redundância. Uma maneira mais usual para expressar a redundância de um código, é definir a taxa deste código.

Definição 3.2 – Taxa de um código

Seja $\mathcal{C}$ um código de bloco q-ário com L palavras-código, cada uma de comprimento n. A taxa R do código $\mathcal{C}$ é dada por

$$R = \frac{\log_q L}{n}. \tag{3.2}$$

Para os casos em que $L = q^k$, a Fórmula (3.2) reduz-se a

$$R = \frac{k}{n}. \tag{3.3}$$

□

A seguir, são apresentadas algumas definições e teoremas importantes na teoria dos códigos de bloco.

Definição 3.3 – Peso de Hamming de um vetor

*O **peso de Hamming**, ou simplesmente **peso**, de um vetor q-ário, em geral denotado por $w(\mathbf{v})$, é o número de coordenadas não nulas deste vetor.* □

Exemplo 3.1

Considere os seguintes vetores: $\mathbf{v}_1 = (1,0,0,1,1,0,1,0,1,1)$ um vetor binário, $\mathbf{v}_2 = (2,0,2,1,1,0,1,0,1,1)$ um vetor ternário e $\mathbf{v}_3 = (\alpha^3,0,0,0,0,0,\alpha^4,0,0,0)$ cujos elementos pertencem a $GF(2^3)$. Os respectivos pesos são: $w(\mathbf{v}_1) = 6$, $w(\mathbf{v}_2) = 7$ e $w(\mathbf{v}_3) = 2$. □

Definição 3.4 – Distância de Hamming

*A **distância de Hamming** entre dois vetores $\mathbf{v}$ e $\mathbf{w}$, de mesmo comprimento n, é o número de coordenadas em que eles diferem.*

$$d_{\text{Hamming}}(\mathbf{v},\mathbf{w}) = d(\mathbf{v},\mathbf{w}) \triangleq \#\{i \mid v_i \neq w_i,\ i = 0,1,\ldots,n-1\}, \tag{3.4}$$

em que $\#\{.\}$ denota a cardinalidade[1] do conjunto. □

Exemplo 3.2

No exemplo 3.1, os vetores $\mathbf{v}_1$, $\mathbf{v}_2$ e $\mathbf{v}_3$ são todos de mesmo comprimento, $n = 10$. Desta forma, a distância de Hamming entre eles é: $d(\mathbf{v}_1,\mathbf{v}_2) = 2$, $d(\mathbf{v}_1,\mathbf{v}_3) = 6$ e $d(\mathbf{v}_2,\mathbf{v}_3) = 7$. □

A Definição 3.4 tem uma importância significativa na teoria de códigos corretores de erro. A partir dela, pode-se definir uma importante propriedade dos códigos de bloco, um importante parâmetro para caracterizar a capacidade de detecção de erro, a capacidade de correção de erro e a capacidade de correção de apagamento. Tal propriedade é definida a seguir.

Definição 3.5 – Distância mínima de um código

*A **distância mínima**, em geral denotada por $d_{\min}$, de um código de bloco C é a menor distância de Hamming entre todos os pares distintos de palavras-código pertencentes a C.* □

[1] Termo utilizado para designar o número de elementos de um conjunto

Exemplo 3.3

Considere um código binário $\mathcal{C}$ formado pelo seguinte conjunto de palavras: $\{\mathbf{c}_0; \mathbf{c}_1; \mathbf{c}_2; \mathbf{c}_3\}$, *em que* $\mathbf{c}_0 = (0,0,0,1)$, $\mathbf{c}_1 = (1,1,0,1)$, $\mathbf{c}_2 = (0,1,0,1)$ *e* $\mathbf{c}_3 = (1,0,1,0)$. *Para determinar a distância mínima do código $\mathcal{C}$, calcula-se a distância de Hamming entre todos os pares de palavras-código. Como o código possui quatro palavras no total e* $d(\mathbf{v},\mathbf{w}) = d(\mathbf{w},\mathbf{v})$, *o total de distâncias calculadas que é dado por* $\binom{4}{2} = 6$. *Os resultados são:* $d(\mathbf{c}_0,\mathbf{c}_1) = 2$, $d(\mathbf{c}_0,\mathbf{c}_2) = 1$, $d(\mathbf{c}_0,\mathbf{c}_3) = 3$, $d(\mathbf{c}_1,\mathbf{c}_2) = 1$, $d(\mathbf{c}_1,\mathbf{c}_3) = 3$ *e* $d(\mathbf{c}_2,\mathbf{c}_3) = 4$. *Logo,* $d_{\min} = 1$. □

Por meio do Exemplo 3.3, percebe-se que encontrar a distância mínima de um código pode ser um processo exaustivo à medida que o número de palavras do código aumentar. Para ser mais específico, o número de comparações necessárias é dado por $\binom{L}{2} = L(L-1)/2$ ou $q^k(q^k-1)/2$, quando $L = q^k$, o que mostra que a complexidade do cálculo é exponencial e é cada vez maior à medida que k aumenta. Uma vez definida a distância mínima de um código, pode-se determinar a capacidade de detecção de erro, a capacidade de correção de erro e a capacidade de correção de apagamento para um código de bloco. Daqui em diante, denomina-se *padrão de erro* o vetor que representa os possíveis erros inseridos pelo canal na palavra-código transmitida.

Teorema 3.1 – Capacidade de detecção de erro [9]

Um código $\mathcal{C}$ com distância mínima $d_{\min}$ é capaz de detectar todos os padrões de erro com peso menor ou igual a $d_{\min} - 1$. □

Demonstração: Pela definição da distância mínima de um código, uma palavra-código difere das demais em, no mínimo, $d_{\min}$ coordenadas. Portanto, se um padrão de erro tem peso maior ou igual a $d_{\min}$, então a palavra-código transmitida pode tornar-se uma outra palavra-código, diferente daquela transmitida, e o receptor não será capaz de detectar que houve um erro na transmissão. Entretanto, se o peso do padrão de erro é menor ou igual a $d_{\min} - 1$, então a palavra transmitida se tornará um vetor que não pertence ao código, desta forma, sendo possível para o receptor detectar que houve erro na transmissão. ■

Teorema 3.2 – Capacidade de correção de erro [11]

Um código $\mathcal{C}$ com distância mínima $d_{\min}$ é capaz de corrigir todos os padrões de erro com peso menor ou igual a $t = \lfloor \frac{d_{\min}-1}{2} \rfloor$, em que $\lfloor x \rfloor$ denota o único número inteiro i tal que $i \leq x < i+1$. Se $d_{\min}$ é um número par, então o código corrige $(d_{\min}-2)/2$ erros ou detecta $d_{\min}/2$. □

Demonstração: Vide referência [11, pág. 10]. ■

Códigos corretores de erro também são utilizados em canais com *apagamento*. Nestas situações, o decodificador é projetado para declarar como apagamento um símbolo para o qual existe dúvida com relação ao real valor transmitido. Por exemplo, num *canal binário com apagamento* [5], os símbolos transmitidos assumem o valor 0 ou 1, mas nas saída do canal os símbolos podem ser 0, 1 ou $\star$, em que este último indica que o decodificador não tem certeza se o símbolo transmitido foi 0 ou 1. Em geral, um canal q-ário com apagamento possui q símbolos para o alfabeto de entrada e $q+1$ para o alfabeto de saída, tendo $\star$ como o símbolo adicional no alfabeto de saída para indicar apagamento. O número de apagamentos que um código de bloco com distância mínima $d_{\min}$ pode corrigir é dado no teorema a seguir.

Teorema 3.3 – Capacidade de correção de apagamento [13]

Um código $\mathcal{C}$ com distância mínima $d_{\min}$ é capaz de corrigir todos os padrões de apagamento com peso menor ou igual a ρ se $d_{\min} \geq \rho + 1$. Além do mais, quaisquer padrões de erro e apagamento em que ocorram t ou menos erros e ρ ou menos apagamentos simultaneamente, podem ser corrigidos se $d_{\min} \geq 2t + \rho + 1$. □

Demonstração: Vide referência [13, pág. 14] ■

Para o caso particular do Capítulo 2, afirma-se que os usuários do canal transmitem σ pacotes livres de colisão de um total de w pacotes enviados por quadro. Neste caso, até $w - \sigma$ pacotes podem ser considerados como apagamentos devido à colisão com pacotes de outros usuários. Logo, para garantir que os usuários transmitam σ pacotes livres de colisão, os codificadores destes usuários devem utilizar um código capaz de corrigir $w - \sigma$ apagamentos. Assim, os códigos usados devem possuir, de acordo com o Teorema 3.3, comprimento $n = w$, blocos de mensagem de comprimento $k = \sigma$ e distância mínima $d_{\min} \geq w - \sigma + 1$, ou seja, um código de bloco $(w, \sigma, w - \sigma + 1)$.

Um dos problemas com os códigos de bloco, em geral, é que é necessário armazenar todas as palavras do código para poder executar os processos de codificação e decodificação e quando o valor de k aumenta, o número de palavras-código aumenta exponencialmente. É visto na próxima seção que a complexidade desse problema pode ser reduzida se os códigos de bloco forem *lineares*.

3.2 CÓDIGOS DE BLOCO LINEARES

Nesta seção, mostra-se que a *linearidade* provê uma estrutura matemática aos códigos de bloco que permite fazer várias simplificações com relação às propriedades discutidas na seção anterior. Inicialmente, define-se um código de bloco linear.

Definição 3.6 – Códigos de bloco lineares

Um código de bloco q-ário $\mathcal{C}$ com q^k palavras-código é um ***código de bloco linear*** $(n, k, d_{\min})$ *se e somente se suas q^k palavras-código formam um subespaço de dimensão k do espaço vetorial de todas as n-uplas sobre $GF(q)$.* □

A Definição 3.6 permite que se utilize várias propriedades da teoria de espaços vetoriais amplamente conhecidas da álgebra linear [10, Cap. 2].

Propriedade 3.1 – [10]

A combinação linear de qualquer conjunto de palavras-código é uma palavra-código. Uma consequência disto é que um código linear sempre contém a palavra-código toda nula, daqui por diante, denotada por **0**. □

Demonstração: A prova é uma consequência direta da definição de espaço vetorial [10, Cap. 2]. ■

Propriedade 3.2 – [10]

A distância mínima de um código de bloco linear é igual ao peso da palavra-código de menor peso entre todas as palavras do código não-nulas. □

Demonstração: A distância mínima de um código $\mathcal{C}$ pode ser escrita matematicamente como $d_{\min} = \min_{\mathbf{c},\mathbf{c}' \in \mathcal{C},\ \mathbf{c} \neq \mathbf{c}'} d(\mathbf{c}, \mathbf{c}')$. Deste modo, pode-se reescrevê-la como $d_{\min} = \min_{\mathbf{c},\mathbf{c}' \in \mathcal{C},\ \mathbf{c} \neq \mathbf{c}'} w(\mathbf{c} - \mathbf{c}')$. Como o código é linear $\mathbf{c}'' = \mathbf{c} - \mathbf{c}'$ e $\mathbf{c}'' \in \mathcal{C}$. Logo, $d_{\min} = \min_{\mathbf{c}'' \in \mathcal{C},\ \mathbf{c}'' \neq \mathbf{0}} w(\mathbf{c}'')$. ■

Pela Propriedade 3.2 percebe-se como a complexidade para encontrar a distância mínima de um código é reduzida. Anteriormente, foi mencionado que a complexidade para encontrar a $d_{\min}$ de um código de bloco é dada por $q^k(q^k - 1)/2$. Para códigos lineares, este valor diminui para $q^k - 1$, pois só é preciso encontrar a palavra-código não nula com menor peso.

3.2.1 Matriz Geradora e Matriz de Verificação de Paridade

Dado um espaço vetorial $\mathbf{V}$, pode-se escolher um subconjunto finito de vetores, $\{\mathbf{v}_i\}$ tal que $\mathbf{v}_i \in \mathbf{V}$, de modo que qualquer outro vetor pertencente a $\mathbf{V}$ pode ser obtido como uma combinação linear dos vetores que estão no subconjunto $\{\mathbf{v}_i\}$. Entretanto, se os vetores que constituem o subconjunto $\{\mathbf{v}_i\}$ forem linearmente independentes, obtém-se uma base vetorial para o espaço $\mathbf{V}$. Por meio de uma base vetorial, o mapeamento entre as combinações lineares dos vetores-base e todos os vetores que pertencem a $\mathbf{V}$ é um-a-um. A definição de códigos lineares como subespaços vetoriais, permite, então, obter uma eficiente representação para esses códigos.

Seja $\{\mathbf{g}_0, \mathbf{g}_1, \ldots, \mathbf{g}_{k-1}\}$ uma base de vetores-código de um código linear q-ário $(n, k, d_{\min})$. Então, cada palavra-código $\mathbf{c}$ pode ser representada de modo único por $\mathbf{c} = m_0\mathbf{g}_0 + m_1\mathbf{g}_1 + \ldots + m_{k-1}\mathbf{g}_{k-1}$, em que $m_i \in GF(q)$, para $0 \leq i \leq k-1$, representam as coordenadas do vetor-mensagem $\mathbf{m} = (m_0, m_1, \ldots, m_{k-1})$. Esta situação pode ser representada matricialmente caso defina-se os vetores-base $\mathbf{g}_i$ como linhas de uma *matriz geradora* denotada por $\mathbf{G}$, do seguinte modo

$$\mathbf{G} = \begin{bmatrix} \mathbf{g}_0 \\ \mathbf{g}_1 \\ \vdots \\ \mathbf{g}_{k-1} \end{bmatrix} = \begin{bmatrix} g_{0,0} & g_{0,1} & \cdots & g_{0,n-1} \\ g_{1,0} & g_{1,1} & \cdots & g_{1,n-1} \\ \vdots & \vdots & \ddots & \vdots \\ g_{k-1,0} & g_{k-1,1} & \cdots & g_{k-1,n-1} \end{bmatrix}. \tag{3.5}$$

Pode-se usar diretamente a matriz $\mathbf{G}$ para codificar os vetores-mensagem $\mathbf{m} = (m_0, m_1, \ldots, m_{k-1})$ em vetores-código $\mathbf{c}$ procedendo da seguinte forma

$$\mathbf{c} = \mathbf{mG} = (m_0, m_1, \ldots, m_{k-1}) \begin{bmatrix} \mathbf{g}_0 \\ \mathbf{g}_1 \\ \vdots \\ \mathbf{g}_{k-1} \end{bmatrix} = m_0\mathbf{g}_0 + m_1\mathbf{g}_1 + \ldots + m_{k-1}\mathbf{g}_{k-1}. \tag{3.6}$$

A representação dos códigos lineares, por meio da matriz $\mathbf{G}$, soluciona o problema de armazenar todas as palavras de um código de bloco para executar o processo de codificação. Basta, neste caso, armazenar k palavras-código linearmente independentes. Vale ressaltar que o número de matrizes $\mathbf{G}$ distintas que podem representar um mesmo código linear q-ário

$(n, k, d_{\min})$ é dado por [8]

$$\prod_{i=0}^{k-1} \left(q^k - q^i\right). \tag{3.7}$$

É interessante destacar que o número de possíveis codificadores para um código de bloco não-linear com q^k palavras-código é igual $(q^k)!$ [8]. Esse número é bem maior que o número de matrizes geradoras distintas para um código linear dado por (3.7). A redução, deve-se ao fato da restrição de que as linhas de $\mathbf{G}$ são linearmente independentes. Dentre todos os possíveis codificadores previstos por (3.7), um deles destaca-se dentre os demais, é o *codificador sistemático*. Ao utilizar esse codificador, é possível distinguir na palavra-código quais são os símbolos de informação e quais são os símbolos de redundância. Em situações práticas, os códigos construídos na forma sistemática são preferidos. A matriz geradora de um código sistemático possui a seguinte forma

$$\mathbf{G} = [\mathbf{P}|\mathbf{I}_k] = \begin{bmatrix} p_{0,0} & p_{0,1} & \cdots & p_{0,n-k-1} & 1 & 0 & 0 & \ldots & 0 \\ p_{1,0} & p_{1,1} & \cdots & p_{1,n-k-1} & 0 & 1 & 0 & \ldots & 0 \\ p_{2,0} & p_{2,1} & \cdots & p_{2,n-k-1} & 0 & 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{k-1,0} & p_{k-1,1} & \cdots & p_{k-1,n-k-1} & 0 & 0 & 0 & \ldots & 1 \end{bmatrix}. \tag{3.8}$$

Exemplo 3.4

O código linear binário $(7, 4, 3)$ *possui a seguinte matriz geradora:*

$$\mathbf{G} = \begin{bmatrix} \mathbf{g}_0 \\ \mathbf{g}_1 \\ \mathbf{g}_2 \\ \mathbf{g}_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

As palavras-código $\mathbf{g}_0$, $\mathbf{g}_1$, $\mathbf{g}_2$ *e* $\mathbf{g}_3$ *são linearmente independentes e formam uma base vetorial para o subespaço de dimensão* k *que corresponde às palavras do código.* □

Outra importante matriz associada a um código de bloco linear é a *matriz de verificação de paridade* denotada por $\mathbf{H}$. As linhas desta matriz são elementos do conjunto de vetores $\{\mathbf{h}_0, \mathbf{h}_1, \ldots, \mathbf{h}_{n-k-1}\}$ que formam uma base vetorial do espaço $\mathcal{C}^{\perp}$ o qual denota o espaço

dual do código $\mathcal{C}$ gerado por $\mathbf{G}$.

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}_0 \\ \mathbf{h}_1 \\ \vdots \\ \mathbf{h}_{n-k-1} \end{bmatrix} = \begin{bmatrix} h_{0,0} & h_{0,1} & \dots & h_{0,n-1} \\ h_{1,0} & h_{1,1} & \dots & h_{1,n-1} \\ \vdots & \vdots & \ddots & \vdots \\ h_{n-k-1,0} & h_{n-k-1,1} & \dots & h_{n-k-1,n-1} \end{bmatrix}. \tag{3.9}$$

Na forma sistemática, a matriz $\mathbf{H}$ pode ser obtida diretamente da matriz $\mathbf{G}$ sistemática (3.8).

$$\mathbf{H} = [\mathbf{I}_{n-k} | -\mathbf{P}^T] = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 & -p_{0,0} & -p_{1,0} & \dots & p_{k-1,0} \\ 0 & 1 & 0 & \dots & 0 & -p_{0,1} & -p_{1,1} & \dots & p_{k-1,1} \\ 0 & 0 & 1 & \dots & 0 & -p_{0,2} & -p_{1,2} & \dots & p_{k-1,2} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 & -p_{0,n-k-1} & -p_{1,n-k-1} & \dots & p_{k-1,n-k-1} \end{bmatrix}. \tag{3.10}$$

Teorema 3.4 – [10]

Um vetor $\mathbf{c}$ *é uma palavra do código* $\mathcal{C}$ *se e somente se* $\mathbf{c}\mathbf{H}^T = \mathbf{0}$*, em que* $\mathbf{H}^T$ *denota a matriz transposta da matriz de verificação de paridade* $\mathbf{H}$*.* □

Demonstração: Vide [10, pág. 84]. ■

Teorema 3.5 – [10]

Seja $\mathcal{C}$ *um código linear com matriz de paridade* $\mathbf{H}$*. A distância mínima de* $\mathcal{C}$ *é igual ao menor número, diferente de zero, de colunas da matriz* $\mathbf{H}$ *cuja combinação linear resulta em* $\mathbf{0}^T$*.* □

Demonstração: Vide [10, pág. 84]. ■

Teorema 3.6 – Cota de Singleton [10]

A distância mínima de um código $(n, k, d_{\min})$ *é limitada superiormente por*

$$d_{\min} \leq n - k + 1. \tag{3.11}$$

□

Demonstração: Vide [10, pág. 84]. ■

Códigos que satisfazem a cota de Singleton com igualdade são conhecidos como *códigos MDS* (*Maximum Distance Separable*), ou seja, códigos cuja distância mínima é a máxima possível.

Para evitar dúvidas quando se fizer referência a códigos de bloco lineares e códigos de bloco não-lineares, daqui por diante, usa-se a terminologia *códigos não-lineares* para designar códigos de bloco que não são lineares.

3.3 Códigos Cíclicos

Assim como a linearidade agregou mais propriedades aos códigos de bloco, mais propriedades podem ser acrescentadas se, além de lineares, estes códigos forem *cíclicos*. Os códigos cíclicos obtiveram grande destaque em aplicações práticas como, por exemplo, no *compact disc* (CD) e no *NASA Deep Space Standard* para comunicações via satélite [10].

Definição 3.7 – Códigos cíclicos

Um código de bloco $\mathcal{C}$ é denominado ***cíclico*** *se qualquer deslocamento cíclico de uma palavra-código resulta em uma palavra-código.* □

Embora seja bastante comum empregar o termo *código cíclico* a um código que é simultaneamente linear e cíclico [9], [10], [14], a Definição 3.7 abrange os códigos lineares e os códigos não-lineares. De modo que, para evitar ambiguidade, usa-se a expressão *código cíclico linear* para designar os casos em que o código em questão é linear e cíclico.

Ao trabalhar com códigos cíclicos, é comum representar as palavras-código por meio de polinômios. Isto é, a palavra-código $\mathbf{c} = (c_0, c_1, c_2, \ldots, c_{n-1})$ pode ser representada pelo *polinômio-código* $c(x) = c_0 + c_1x + c_2x^2 + \ldots + c_{n-1}x^{n-1}$. Considere, neste ponto, como os deslocamentos cíclicos realizados nas palavras-código podem ser reproduzidos na representação polinomial. Se $\mathbf{c} = (c_0, c_1, c_2, \ldots, c_{n-1}) \in \mathcal{C}$, então a palavra-código obtida ao deslocar $\mathbf{c}$ uma posição para direita é $\mathbf{c}' = (c_{n-1}, c_0, c_1, \ldots, c_{n-2}) \in \mathcal{C}$. Este resultado é obtido na representação polinomial ao se efetuar a seguinte operação

$$\begin{aligned} xc(x) &= (c_0x + c_1x^2 + c_2x^3 + \ldots + c_{n-1}x^n) \bmod (x^n - 1) \\ &= (c_{n-1} + c_0x + c_1x^2 + \ldots + c_{n-2}x^{n-1}) \bmod (x^n - 1) \\ &= c'(x) \bmod (x^n - 1). \end{aligned}$$

Sendo assim, efetuar dois deslocamentos para a direita em $\mathbf{c}$ seria equivalente a multiplicar $c(x)$ por $x^2 \bmod (x^n - 1)$ e assim sucessivamente até $n - 1$ deslocamentos, os quais seriam obtidos, na forma polinomial, multiplicando $c(x)$ por x^{n-1}. Obviamente, multiplicar por x^n seria o mesmo que multiplicar por 1, pois $x^n = 1 \bmod (x^n - 1)$. Na forma vetorial, significa

dizer que deslocar ciclicamente n vezes obtém-se a palavra-código original. A partir desta constatação, pode-se definir a *ordem cíclica* de uma palavra-código.

Definição 3.8 – Ordem cíclica

A ***ordem cíclica*** *de uma palavra-código é o menor inteiro positivo i tal que $x^i c(x) = c(x) \bmod (x^n - 1)$.* □

Embora a Definição 3.8 refira-se a vetores que pertencem a um código cíclico, o conceito de ordem cíclica pode ser expandido a qualquer n-upla q-ária de comprimento n. É importante ressaltar que os possíveis valores para a ordem cíclica de uma palavra-código são os divisores de n. Diz-se, então, que uma palavra-código tem ***ordem cíclica plena*** quando a ordem cíclica desta palavra é igual a n.

Teorema 3.7 – [10]

Seja $\mathcal{C}$ um código cíclico linear q-ário $(n, k, d_{\min})$.

a. *Existe um único polinômio mônico $g(x)$ de grau $n-k$, o qual representa uma palavra-código* **g**, *que é o polinômio de menor grau entre todos os polinômios que representam as palavras-código de $\mathcal{C}$. $g(x)$ é denominado* ***polinômio gerador*** *do código $\mathcal{C}$.*

b. *O polinômio gerador $g(x)$ de um código cíclico linear $(n, k, d_{\min})$ é fator de $x^n - 1$.*

c. *Cada polinômio-código $c(x) \in \mathcal{C}$ é expresso unicamente como $c(x) = m(x)g(x)$, em que $m(x)$ é o* ***polinômio-mensagem*** *de grau menor que k e $g(x)$ é o polinômio gerador de $\mathcal{C}$.* □

Demonstração:

a. Seja $g(x) = g_0 + g_1 x + \ldots + g_{n-k-1}x^{n-k-1} + x^{n-k}$ um polinômio-código mônico não nulo. Supondo que $g(x)$ não é único, então existe outro polinômio-código mônico de grau $n-k$, digamos $f(x) = f_0 + f_1 x + \ldots + f_{n-k-1}x^{n-k-1} + x^{n-k}$. Uma vez que $\mathcal{C}$ é linear, $g(x) - f(x) = (g_0 - f_0) + (g_1 - f_1)x + \ldots + (g_{n-k-1} - f_{n-k-1})x^{n-k-1} + \overbrace{(1-1)}^{=0} x^{n-k}$ é um outro polinômio-código cujo grau é $n-k-1$. Mas, isto é impossível, pois $n-k-1 < n-k$. Logo, $g(x) - f(x) = 0 \Rightarrow g(x) = f(x)$. Consequentemente, $g(x)$ é único.

b. Vide [11, pág. 191].

c. Vide [11, pág. 191]. ■

Uma observação importante é que para garantir a Propriedade ***a*** no Teorema 3.7, $g_0 \neq 0$. Pois, deslocar ciclicamente $g(x)$ uma posição para a esquerda produz $g'(x) = g_1 + g_2x + \ldots + x^{n-k-1} + g_0x^{n-k}$. Logo, $g_0 \neq 0$, senão um polinômio de grau menor que $n-k$ seria um polinômio-código, o que é impossível.

A forma geral da matriz $\mathbf{G}$ de um código cíclico linear pode ser obtida por meio da Propriedade ***c*** no Teorema 3.7. Sendo o polinômio mensagem $m(x) = m_0 + m_1x + \ldots + m_{k-1}x^{k-1}$, a multiplicação $c(x) = m(x)g(x)$ pode ser escrita da seguinte forma

$$\begin{aligned} c(x) &= (m_0 + m_1x + \ldots + m_{k-1}x^{k-1})g(x) \\ &= m_0g(x) + m_1xg(x) + \ldots + m_{k-1}x^{k-1}g(x). \end{aligned} \tag{3.12}$$

A Expressão (3.12) pode ser reescrita na forma de multiplicação de matrizes como pode ser observado em (3.13).

$$c(x) = [m_0\ m_1\ \ldots\ m_{k-1}] \begin{bmatrix} g(x) \\ xg(x) \\ \vdots \\ x^{k-1}g(x) \end{bmatrix}. \tag{3.13}$$

Lembrando que multiplicar $g(x)$ por $x^i \bmod (x^n - 1)$, sendo i um número inteiro positivo, é equivalente a deslocar ciclicamente $\mathbf{g}$ de i posições para a direita, então a matriz $\mathbf{G}$ pode ser escrita conforme (3.14). Também é possível construir a matriz $\mathbf{G}$ na forma sistemática para um código cíclico linear. Um algoritmo mostrando como realizar este procedimento pode ser visto em [10, pág. 107].

$$\mathbf{G} = \begin{bmatrix} g(x) \\ xg(x) \\ \vdots \\ x^{k-1}g(x) \end{bmatrix} \equiv \begin{bmatrix} g_0 & g_1 & \cdots & g_{n-k} & & & & \mathbf{0} \\ & g_0 & g_1 & \cdots & g_{n-k} & & & \\ & & \ddots & \ddots & \ddots & \ddots & & \\ & & & g_0 & g_1 & \cdots & g_{n-k} & \\ \mathbf{0} & & & & g_0 & g_1 & \cdots & g_{n-k} \end{bmatrix}. \tag{3.14}$$

Segue da Propriedade ***b*** que para cada polinômio gerador $g(x)$, existe um *polinômio-paridade* $h(x)$ mônico, com $h_0 \neq 0$ e de grau k, tal que $g(x)h(x) = x^n - 1$. Sabe-se que $c(x)$ é um polinômio-código se e somente se ele é um múltiplo de $g(x)$, alternativamente, $c(x)$ é um polinômio-código se e somente se $c(x)h(x) = 0 \bmod (x^n - 1)$. Esta equação pode ser manipulada [10] de modo a ser escrita na forma matricial como $\mathbf{cH}^T = \mathbf{0}$. A matriz $\mathbf{H}_{(n-k)\times n}$

em questão é a matriz de verificação de paridade para o código gerado por $g(x)$.

$$\mathbf{H} = \begin{bmatrix} h_k & \dots & h_1 & h_0 & & & \mathbf{0} \\ & h_k & \dots & h_1 & h_0 & & \\ & & \ddots & \ddots & \ddots & \ddots & \\ & & & h_k & \dots & h_1 & h_0 \\ \mathbf{0} & & & & h_k & \dots & h_1 & h_0 \end{bmatrix}^T . \tag{3.15}$$

Como pode ser visto em (3.15), as linhas da matriz $\mathbf{H}$ são formadas pelo deslocamento cíclico de um polinômio que não é o polinômio-paridade $h(x) = h_0 + h_1 x + \ldots + h_k x^k$, mas é um polinômio cujos coeficientes são os mesmos de $h(x)$ só que em ordem reversa. Tal polinômio, $h^*(x) = h_k + h_{k-1}x + \ldots + h_1 x^{k-1} + h_0 x^k$, é denominado *polinômio-recíproco*[2] de $h(x)$. O teorema a seguir mostra uma importante relação entre $h^*(x)$ e o código dual $\mathcal{C}^\perp$ de um código cíclico linear $\mathcal{C}$ gerado por $g(x)$.

Teorema 3.8 – [10]

Seja $\mathcal{C}$ um código linear cíclico q-ário (n, k) com polinômio gerador $g(x)$. O código dual $\mathcal{C}^\perp$, do código $\mathcal{C}$, é um código cíclico linear q-ário $(n, n-k)$ cujo polinômio gerador é $h^*(x)$. □

Demonstração: A matriz $\mathbf{H}$ tem a mesma estrutura da matriz $\mathbf{G}$ em (3.14), portanto esta matriz $\mathbf{H}$ gera um subespaço de dimensão $(n-k)$ que é o código $\mathcal{C}^\perp$. ■

3.3.1 Códigos BCH

Os códigos BCH são assim denominados em homenagem aos pesquisadores que os investigaram inicialmente: A. Hocquenghem em 1959 [39], Bose e Ray-Chaudhuri em 1960 [40], [41]. Entretanto, coube a outro pesquisador, W. Peterson, mostrar que esses códigos eram cíclicos e construir o primeiro algoritmo de decodificação algébrica para códigos BCH [42].

Em geral, quando se escolhe um polinômio gerador $g(x)$ e gera-se um código cíclico linear não se tem ideia do valor da distância mínima deste código. Por isto, os códigos BCH destacam-se em relação a códigos cíclicos arbitrários, por garantirem um limite inferior para a distância mínima do código. Tal resultado é obtido impondo algumas restrições à escolha do polinômio gerador $g(x)$. O limite inferior para a distância mínima do código é denominado *distância projetada* do código e é denotado por δ. O teorema a seguir especifica como escolher as raízes do polinômio gerador a fim de garantir a distância projetada do código.

[2] Em geral, o polinômio-recíproco de $a(x) = a_0 + a_1 x + \ldots + a_{n-1}x^{n-1} + a_n x^n$ é o polinômio $a^*(x) = x^n a(x^{-1}) = a_n + a_{n-1}x + \ldots + a_1 x^{n-1} + a_0 x^n$.

Teorema 3.9 – cota BCH [10]

Seja C um código linear cíclico q-ário $(n, k, d_{\min})$ *cujo polinômio gerador é* $g(x)$*. Além do mais, seja* m *a ordem multiplicativa de* $q \bmod n$ *tal que* $GF(q^m)$ *é o menor corpo de extensão de* $GF(q)$ *que contém a* n*-ésima raiz da unidade. Considere* α *a* n*-ésima raiz primitiva da unidade. Assegure que* $g(x)$ *foi escolhido como o polinômio de menor grau tal que* $g(\alpha^b) = g(\alpha^{b+1}) = g(\alpha^{b+2}) = \ldots = g(\alpha^{b+\delta-2}) = 0$*, em que* b *é um número inteiro* $b \geq 0$ *e* $\delta \geq 1$*. Desta forma,* $g(x)$ *tem* $(\delta - 1)$ *potências consecutivas de* α *como raízes. Portanto, o código C gerado por* $g(x)$ *tem distância mínima* $d_{\min} \geq \delta$*.* □

Demonstração: Vide [10, pág. 176–180] ■

Vale destacar que em muitas situações a distância mínima, $d_{\min}$, do código BCH coincide com a distância projetada, δ. Entretanto, há também várias situações em que $d_{\min}$ é maior que δ [9].

Uma vez que a distância mínima do código BCH está garantida pela cota BCH do Teorema 3.9, um procedimento para construir um código BCH q-ário de comprimento n e corretor de t erros pode ser o seguinte [10]:

i. Encontre uma n-ésima raiz primitiva α de $x^n - 1$ em um corpo $GF(q^m)$ para o menor valor possível de m;

ii. Selecione $(\delta - 1) = 2t$ potências consecutivas de α: α^b, α^{b+1}, α^{b+2},..., α^{b+2t}, sendo b um número inteiro não negativo;

iii. Seja $M_i(x)$ o polinômio mínimo de α^i e seus conjugados. Então faça $g(x)$ o mínimo múltiplo comum entre os polinômios mínimos de todas as potências consecutivas de α.

$$g(x) = MMC\{M_b(x), M_{b+1}(x), M_{b+2}(x), \ldots, M_{b+2t}(x)\}. \tag{3.16}$$

Para $b = 1$ os códigos BCH são denominados de *sentido restrito*. Para $n = q^m - 1$, m sendo um número inteiro positivo, o código BCH é denominado *primitivo*. Quando $n \neq q^m - 1$, os códigos BCH são denominados *não-primitivos* e os possíveis valores de n são os divisores de $q^m - 1$.

3.3.2 Códigos Reed-Solomon

Códigos Reed-Solomon (RS), assim como ocorre com os códigos BCH, recebem esta denominação em homenagem aos primeiros pesquisadores a investigá-los, I. S. Reed e G. Solo-

mon [43]. Entretanto, há várias formas de definir um código RS [10]. A definição usada nesta dissertação segue aquela em que esses códigos são uma extensão natural dos códigos BCH.

Definição 3.9 – Códigos Reed-Solomon

*Um **código Reed-Solomon** é um código BCH de comprimento $n = q^m - 1$ cujos símbolos pertencem a $GF(q^m)$.* □

A partir da Definição 3.9, pode-se destacar que, diferentemente dos códigos BCH mencionados na subseção anterior, a n-ésima raiz da unidade não precisa ser procurada num corpo de extensão $GF(q^m)$, pois os códigos RS são definidos sobre este corpo e, portanto, $\alpha \in GF(q^m)$. Além do mais, os polinômios mínimos $M_s(x)$ são da forma $(x - \alpha^s)$, pois os elementos não nulos de $GF(q^m)$ são as raízes de $x^{q^m-1} - 1$ e, desta forma, não há raízes conjugadas.

Códigos RS destacam-se com relação a outros códigos BCH por possuírem várias propriedades que estes outros códigos não compartilham. Não é por acaso que códigos RS constituem a subclasse dos códigos BCH não-binários mais conhecida, haja vista as várias aplicações em que eles são utilizados [44]. Entre essas propriedades, a mais importante está relacionada à distância mínima dos códigos RS como mostra o teorema a seguir.

Teorema 3.10 – Distância mínima de um código RS [10]

Um código RS (n, k, d_{min}) possui distância mínima igual a $d_{\min} = n - k + 1$. □

Demonstração: Inicialmente, lembre-se de que $grau[g(x)] = n - k$ (Teorema 3.7). Porém, o polinômio gerador $g(x)$ de um código RS possui $\delta - 1$ potências consecutivas de α como raízes, sendo α a n-ésima raiz primitiva da unidade. Logo, $grau[g(x)] = \delta - 1$ e assim $\delta - 1 = n - k$. Pela cota BCH (Teorema 3.9), $d_{\min} \geq \delta = n - k + 1$. Entretanto, pela cota de Singleton (Teorema 3.6) $d_{\min} \leq n - k + 1$. Portanto, a única maneira de satisfazer a cota BCH e a cota de Singleton simultaneamente ocorre para $d_{\min} = n - k + 1$. ■

Desta forma, códigos RS são códigos MDS. Códigos MDS possuem propriedades interessantes que podem ser vistas em [10]. Uma destas propriedades tem um papel importante no trabalho apresentado nesta dissertação, portanto ela é enunciada no teorema a seguir.

Teorema 3.11 – [10]

O número de palavras-código com peso de Hamming j em um código q-ário (n, k) e MDS é dado

por

$$A_j = \binom{n}{j}(q-1)\sum_{i=0}^{j-d_{\min}}(-1)^i\binom{j-1}{i}q^{j-i-d_{\min}}. \tag{3.17}$$

□

Demonstração: Vide [12, pág. 429–431]. ■

3.4 Códigos Constacíclicos

Códigos constacíclicos [12], também conhecidos como *códigos pseudocíclicos* [45]–[47], podem ser definidos de acordo com a Definição 3.10.

Definição 3.10 – Códigos constacíclicos

Um código $\mathcal{C}$ é dito ser um ***código constacíclico*** *se qualquer* ***deslocamento constacíclico*** *de uma palavra-código resulta em uma palavra-código.* □

Na Definição 3.10, deslocar constaciclicamente uma posição para a direita a palavra-código $\mathbf{c} = (c_0, c_1, c_2, \ldots, c_{n-1})$ produz a palavra-código $\mathbf{c}' = (ac_{n-1}, c_0, c_1, \ldots, c_{n-2})$, em que a é um elemento não nulo de $GF(q)$. Observando com atenção a Definição 3.10, é possível perceber que a classe de códigos constacíclicos contém a classe de códigos cíclicos, para isto basta fazer $a = 1$ na Definição 3.10. Além do mais, os *códigos negacíclicos* [12], obtidos fazendo $a = -1$ na Definição 3.10, também são um caso particular dos códigos constacíclicos.

3.4.1 Códigos Constacíclicos de Comprimento $q+1$

A construção de códigos constacíclicos pode ser obtida como uma generalização da construção de códigos cíclicos. Sendo assim, as palavras-código de um código constacíclico linear q-ário (n, k) são reduzidas $\mathrm{mod}(x^n - a)$, $a \in GF(q)$, e o polinômio gerador $g(x)$, de grau $(n-k)$, é fator de $x^n - a$. Os polinômios-código são da forma $c(x) = m(x)g(x)$, em que $m(x)$ é o polinômio-mensagem de grau menor ou igual a $k-1$.

Nesta dissertação, estamos interessados na construção de códigos constacíclicos de comprimento $n = q+1$, sendo q um número primo tal que $q > 3$, e com o elemento a de $x^n - a$ sendo um elemento primitivo do grupo multiplicativo de $GF(q)$. Esta escolha deve-se ao fato de que algumas propriedades interessantes são conhecidas para códigos constacíclicos com esses parâmetros [45]. Uma descrição da existência de códigos constacíclicos para outros valores de n e a pode ser encontrada em [48]. Daqui por diante e até que se informe o contrário, o elemento a do polinômio $x^n - a$ é um elemento primitivo do grupo multiplicativo de $GF(q)$.

Para $n = q + 1$ e a um elemento primitivo do grupo multiplicativo de $GF(q)$, é bem conhecido [45] que as raízes de $x^{q+1} - a$ pertencem a $GF(q^2)$ e podem ser escritas na forma $\alpha^{1+(q-1)i}$, para $0 \leq i \leq q$, ou ainda na forma $\alpha^{q+(q-1)i}$, para $-(q-1)/2 \leq i \leq (q+1)/2$. De acordo com [45], o polinômio $x^{q+1} - a$ é fatorado em $(q+1)/2$ polinômios de grau dois. Logo, as raízes de $x^{q+1} - a$ pertencem a classes conjugadas de cardinalidade dois. Sendo assim, também é possível representá-las por meio de seus expoentes como $\{1-(q-1)i,\ q+(q-1)i\}$, para $0 \leq i \leq (q-1)/2$. Uma consequência deste fato é que o grau dos possíveis polinômios geradores $g(x)$ é um número par, isto é, $n - k$ sempre é par. Como $n = q + 1$ também é par, a dimensão do código, k, sempre será um número par entre $2 \leq k \leq q - 1$.

Exemplo 3.5

Considere $q = 5$. *Os elementos de* $GF(5)$ *são* $\{0, 1, 2, 3, 4\}$ *sendo* 2 *e* 3 *elementos geradores do grupo multiplicativo de* $GF(5)$, *ou seja, elementos cuja ordem multiplicativa é igual a* $q - 1 = 4$. *Desta forma, podemos construir códigos constacíclicos* 5*-ários* $(n = q + 1, k)$ *cujos polinômios-código são reduzidos* $\bmod(x^6 - 2)$ *ou* $\bmod(x^6 - 3)$. *Como a dimensão do código,* k, *é um número par entre* $2 \leq k \leq 4$, *os possíveis códigos constacíclicos* 5*-ários possuem parâmetros* $(6, 2, d_{\min})$ *e* $(6, 4, d_{\min})$. □

Analisa-se, neste ponto, os resultados discutidos no parágrafo anterior para o polinômio $x^6 - 3$ do Exemplo 3.5. Como $q = 5$, considere o corpo de extensão $GF(25)$ gerado por $p(x) = 3 + 2x + x^2$ que é um polinômio primitivo sobre $GF(5)$. Além do mais, considere α um elemento primitivo do grupo multiplicativo de $GF(25)$, tal que $p(\alpha) = 3 + 2\alpha + \alpha^2 = 0$. Portanto, as raízes de $x^6 - 3$ em $GF(25)$ escritas na forma α^{1+4i}, para $0 \leq i \leq 5$, são $\{\alpha, \alpha^5, \alpha^9, \alpha^{13}, \alpha^{17}, \alpha^{21}\}$. Alternativamente, escrevendo as raízes pela forma α^{5+4i}, para $-2 \leq i \leq 3$, obtemos $\{\alpha^{17}, \alpha^{21}, \alpha, \alpha^5, \alpha^9, \alpha^{13}\}$. As classes conjugadas para as raízes de $x^6 - 3$ são $\{1, 5\}$, $\{9, 21\}$ e $\{13, 17\}$ e todas as classes possuem cardinalidade dois, como era esperado. Os polinômios mínimos associados a cada uma das classes conjugadas podem ser vistos na Tabela 3.1. Desta forma, o polinômio $x^6 - 3$ é fatorado em três, $(5+1)/2 = 3$, polinômios de grau dois, ou seja, $x^6 - 3 = M_1(x)M_9(x)M_{13}(x)$.

Observando o conjunto de raízes $\{\alpha, \alpha^5, \alpha^9, \alpha^{13}, \alpha^{17}, \alpha^{21}\}$ percebe-se que os expoentes das raízes formam uma progressão aritmética (PA) de razão 4, no caso geral $q-1$, e com primeiro termo 1 ou, ainda, pode-se dizer que estas raízes formam um progressão geométrica (PG) de razão α^4, no caso geral α^{q-1}, cujo primeiro termo é α. Como é provado em [45], ocorre que, em geral, pode-se escolher um polinômio gerador $g(x)$ com $2t$ raízes consecutivas, pois

Tabela 3.1: *Classes conjugadas e polinômios mínimos sobre* $GF(5)$ *para* $x^6 - 3$

Classe conjugada	Polinômio mínimo
$\{1, 5\}$	$M_1(x) = 3 + 2x + x^2$
$\{9, 21\}$	$M_9(x) = 3 + x^2$
$\{13, 17\}$	$M_{13}(x) = 3 + 3x + x^2$

elas são termos consecutivos de uma PG de razão α^{q-1}, e construir um código constacíclico corretor de t erros cuja distância mínima é limitada inferiormente por $d_{\min} \geq 2t + 1$. É válido salientar que os expoentes de α, nestes casos, são reduzidos $\bmod(q^2 - 1)$. Isto implica, por exemplo, que as raízes α^{21} e α são consideradas consecutivas neste contexto. Esta cota inferior para a distância mínima dos códigos constacíclicos é semelhante à cota BCH do Teorema 3.9. Como o grau do polinômio gerador $g(x)$ é igual a $n - k$ e este polinômio possui $2t$ raízes, $n - k = 2t$ e, assim, $d_{\min} \geq n - k + 1$. Porém, pela cota de Singleton, $d_{\min} \leq 2t + 1$ ou $d_{\min} \leq n - k + 1$. Portanto, desde que o polinômio gerador $g(x)$ possua $2t$ raízes consecutivas de $x^{q+1} - a$, o código gerado por $g(x)$ tem $d_{\min} = n - k + 1$ (MDS) para satisfazer a cota BCH e a cota de Singleton simultaneamente. A vantagem de construir códigos constacíclicos MDS é que a distribuição dos pesos das palavras-código é conhecida, Fórmula (3.17). Para códigos constacíclicos que não são MDS não há uma fórmula fechada para obter a distribuição de pesos das palavras-código, sendo necessária a construção do dicionário do código para conseguir esta informação.

Exemplo 3.6

Para gerar os códigos $(6, 4, d_{\min})$ *do Exemplo 3.5 o grau de* $g(x)$ *é igual a* $n - k = 2$*, logo tem-se três opções para o polinômio gerador, são elas:* $g(x) = M_i(x)$ *para* $i = \{1, 9, 13\}$ *(Tabela 3.1). Para* $g_1(x) = M_1(x)$ *e* $g_2(x) = M_{13}(x)$ *os polinômios geradores possuem duas raízes que são termos consecutivos da PG* $\{\alpha, \alpha^5, \alpha^9, \alpha^{13}, \alpha^{17}, \alpha^{21}\}$*, logo* $d_{\min} = 3$ *e estes códigos são MDS com parâmetros* $(n, k, d_{\min}) = (6, 4, 3)$*. Entretanto, para* $g_3(x) = M_9(x)$ *as raízes são* α^9 *e* α^{21} *e estas não são termos consecutivos da PG, logo o código não é MDS. Neste caso,* $d_{\min} = 2$ *e o código possui parâmetros* $(n, k, d_{\min}) = (6, 4, 2)$*.* □

Baseando-se no Exemplo 3.6, para construir os códigos constacíclicos 5-ários de parâmetros $(6, 2, d_{\min})$, previstos no Exemplo 3.5, o grau do polinômio gerador $g(x)$ tem de ser igual a $n - k = 4$. De acordo com a Tabela 3.1, polinômios geradores de grau 4 são obtidos quando $g(x) = M_i(x)M_j(x)$, $i \neq j$ e $i, j = \{1, 9, 13\}$. Logo, os possíveis polinômios geradores são

Tabela 3.2: *Deslocamentos constacíclicos de* $g(x) = 4 + 2x^2 + x^4 \leftrightarrow \mathbf{g} = (4, 0, 2, 0, 1, 0)$.

i	$x^i g(x) \bmod (x^6 - 3)$
0	$(4, 0, 2, 0, 1, 0)$
1	$(0, 4, 0, 2, 0, 1)$
2	$(3, 0, 4, 0, 2, 0)$
3	$(0, 3, 0, 4, 0, 2)$
4	$(1, 0, 3, 0, 4, 0)$
5	$(0, 1, 0, 3, 0, 4)$
6	$(2, 0, 1, 0, 3, 0)$
7	$(0, 2, 0, 1, 0, 3)$
8	$(4, 0, 2, 0, 1, 0)$

$g_1(x) = M_1(x)M_9(x)$, $g_2(x) = M_1(x)M_{13}(x)$ e $g_3(x) = M_9(x)M_{13}(x)$. Os polinômios $g_1(x)$ e $g_3(x)$ possuem quatro raízes que são termos consecutivos da PG, logo os códigos gerados por eles são MDS de parâmetros $(6, 2, 5)$. O código gerado por $g_2(x)$ não possui as quatro raízes como termos consecutivos da PG e, neste caso, há duas raízes consecutivas que são as raízes de $M_1(x)$ ou, equivalentemente, as de $M_{13}(x)$, logo este código possui parâmetros $(6, 2, 3)$.

3.4.2 Ordem Constacíclica das Palavras-Código

Definição 3.11 – Ordem constacíclica

Considere um código constacíclico linear q-ário $(n, k, d_{\min})$ *em que as palavras-código são reduzidas* $\bmod(x^n - a)$, *em que* a *é um elemento não nulo de* $GF(q)$. *A* ***ordem constacíclica*** *de uma palavra-código é o menor inteiro positivo* i *tal que* $x^i c(x) = c(x) \bmod (x^n - a)$. □

Analogamente ao que ocorre com os códigos cíclicos, multiplicar um polinômio-código $c(x)$ por x^i equivale a deslocar constaciclicamente (Definição 3.10) em i posições a palavra-código $\mathbf{c}$.

Exemplo 3.7

Considere um código constacíclico 5*-ário* $(6, 2, 3)$ *gerado por* $g(x) = M_1(x)M_{13}(x) = 4+2x^2+x^4$ *(Vide Tabela 3.1) e que os polinômios-código são reduzidos* $\bmod (x^6 - 3)$. *Como* $g(x)$ *é um polinômio-código, ele pode ser representado pela palavra-código* $\mathbf{g} = (4, 0, 2, 0, 1, 0)$. *A Tabela 3.2 mostra os deslocamentos constacíclicos de* $\mathbf{g}$. *Logo, a ordem constacíclica de* $g(x)$ *é 8.*

Exemplo 3.8

De maneira semelhante ao Exemplo 3.7, pode-se mostrar que o polinômio gerador $g(x) = M_1(x)M_9(x) = 4 + x + x^2 + 2x^3 + x^4 \leftrightarrow \mathbf{g} = (4, 1, 1, 2, 1, 0)$*, o qual gera um código constacíclico* 5*-ário* $(6, 2, 5)$ *com os polinômios-código reduzidos* $\bmod(x^6 - 3)$*, possui ordem constacíclica* 24*.* □

A partir dos Exemplos 3.7 e 3.8 é possível definir um conceito importante sobre a ordem constacíclica das palavras de um código constacíclico e este conceito é bastante mencionado deste ponto em diante.

Definição 3.12 – Ordem constacíclica plena

Uma palavra-código, pertencente a um código constacíclico, com $n = q+1$*, tem* ***ordem constacíclica plena*** *quando o menor valor de* i *tal que* $x^i c(x) = c(x) \bmod (x^n - a)$ *é* $i = q^2 - 1$*.* □

Neste ponto, pode-se notar que, diferentemente do que ocorre com os códigos cíclicos cuja ordem cíclica é um divisor de n (comprimento do código), a ordem constacíclica pode ser maior que o comprimento, $n = q + 1$, do código constacíclico. De fato, no caso dos códigos cíclicos BCH primitivos, por exemplo, as raízes do polinômio $x^{q^m-1} - 1$ estão em $GF(q^m)$, logo a ordem multiplicativa das raízes do polinômio gerador $g(x)$ de um código BCH primitivo é igual ao comprimento do código $n = q^m - 1$ ou a um dos seus divisores. Portanto, a ordem cíclica não será maior que o comprimento do código. Mas, no caso dos códigos constacíclicos, as raízes do polinômio $x^{q+1} - a$ estão em $GF(q^2)$, portanto a ordem multiplicativa das raízes do polinômio gerador $g(x)$ de um código constacíclico é igual a $q^2 - 1$ ou a um dos seus divisores. Desta forma, a ordem constacíclica pode ser maior que o comprimento do código. Para $q = 5$, por exemplo, temos $q^2 - 1 = 24$, logo os possíveis valores para a ordem constacíclica das palavras-código dos códigos $(6, 2, 3)$ e $(6, 2, 5)$ são $1, 2, 3, 4, 6, 8, 12, 24$. Observando os Exemplos 3.7 e 3.8, percebe-se que este resultado é satisfeito.

A relação entre a ordem multiplicativa das raízes do polinômio $x^{q+1} - a$, consequentemente das raízes de $g(x)$, e a ordem constacíclica das palavras de um código constacíclico é explorada na construção destes códigos tal que o maior número possível de palavras-código tenha ordem constacíclica plena. Essa característica é interessante, pois maximiza o número de possíveis palavras de um código constacíclico que podem ser usadas na construção de códigos cíclicos binários como é feito no Capítulo 4.

Voltando ao Exemplo 3.8, o código constacíclico $(6, 2, 5)$ tem dimensão $k = 2$, logo ele possui $q^k = 5^2 = 25$ palavras-código. Como o código é linear, a palavra-código $\mathbf{0}$ pertence

Tabela 3.3: *Palavras não nulas do código* $(6, 2, 3)$ *gerado por* $g(x) = 4+2x^2+x^4$. *A primeira coluna corresponde à quantidade de deslocamentos constacíclicos para direita.*

i	$\mathbf{c}_1 = (4,0,2,0,1,0)$	$\mathbf{c}_2 = (2,1,1,3,3,4)$	$\mathbf{c}_3 = (3,1,4,3,2,4)$
1	$(0,4,0,2,0,1)$	$(2,2,1,1,3,3)$	$(2,3,1,4,3,2)$
2	$(3,0,4,0,2,0)$	$(4,2,2,1,1,3)$	$(1,2,3,1,4,3)$
3	$(0,3,0,4,0,2)$	$(4,4,2,2,1,1)$	$(4,1,2,3,1,4)$
4	$(1,0,3,0,4,0)$	$(3,4,4,2,2,1)$	$(2,4,1,2,3,1)$
5	$(0,1,0,3,0,4)$	$(3,3,4,4,2,2)$	$(3,2,4,1,2,3)$
6	$(2,0,1,0,3,0)$	$(1,3,3,4,4,2)$	$(4,3,2,4,1,2)$
7	$(0,2,0,1,0,3)$	$(1,1,3,3,4,4)$	$(1,4,3,2,4,1)$

ao código e, pela Definição 3.11, possui ordem constacíclica igual a 1. Sendo assim, restam 24 palavras-código. Conforme o Exemplo 3.8, a palavra-código $\mathbf{g} = (4,1,1,2,1,0)$ possui ordem constacíclica 24. Logo, as 24 palavras-código restantes do código $(6,2,5)$ gerado por $g(x) = 4 + x + x^2 + 2x^3 + x^4$ correspondem à palavra-código $\mathbf{g}$ e seus deslocamentos constacíclicos. Com esse exemplo, é possível deduzir que o maior número de palavras-código com ordem constacíclica plena que é possível obter para um código constacíclico é igual a $q^k - 1$, pois como estes códigos são lineares, a palavra toda nula sempre pertence ao código. Em contrapartida, o código constacíclico do Exemplo 3.7, que também possui 25 palavras-código, é composto pela palavra-código $\mathbf{0}$ e por outras 24 palavras-código de ordem constacíclica 8. A Tabela 3.3 mostra as palavras não-nulas do código $(6,2,3)$ do Exemplo 3.7. Nela pode-se observar como as palavras-código não-nulas podem ser obtidas por intermédio do deslocamento constacíclico das palavras-código $\mathbf{c}_1$, $\mathbf{c}_2$ e $\mathbf{c}_3$.

Conforme mencionado, a relação entre a ordem multiplicativa das raízes do polinômio $x^{q+1} - a$ e a ordem constacíclica das palavras-código é de fundamental importância na construção de códigos constacíclicos com o maior número possível de palavras-código com ordem constacíclica plena. O teorema a seguir mostra como escolher um polinômio gerador $g(x)$ tal que o código constacíclico obtido tenha $q^k - 1$ palavras-código com ordem constacíclica plena.

Teorema 3.12

Seja o polinômio $x^{q+1} - a$, *em que* q *é um número primo,* $q > 3$, *e* $a \neq 0$ *é um elemento primitivo do grupo multiplicativo de* $GF(q)$. *As raízes de* $x^{q+1} - a$ *pertencem a* $GF(q^2)$ *e podem ser escritas na forma* $\alpha^{1+(q-1)i}$, *para* $0 \leq i \leq q$. *Assuma que pelo menos um par de raízes conjugadas de* $x^{q+1} - a$ *tenha ordem multiplicativa* $q^2 - 1$. *Se todas as raízes de* $x^{q+1} - a$ *que não têm ordem*

multiplicativa igual a q^2-1 são escolhidas como raízes do polinômio gerador $g(x)$ de um código $\mathcal{C}$ constacíclico linear q-ário $(q+1, k, d_{\min})$, então todas as palavras-código não-nulas de $\mathcal{C}$ possuem ordem constacíclica plena. □

Demonstração: Para que qualquer $c(x) \in \mathcal{C}$ tenha ordem cíclica plena, o menor valor de i tal que

$$x^i c(x) \quad = \quad c(x) \bmod (x^{q+1} - a)$$

ou, equivalentemente,

$$(x^i - 1)c(x) \quad = \quad 0 \bmod (x^{q+1} - a) \tag{3.18}$$

tem de ser $i = q^2 - 1$. Entretanto, pode-se escrever $c(x) = m(x)g(x)$ e substituir $c(x)$ por $m(x)g(x)$ na Equação (3.18). Logo,

$$(x^i - 1)m(x)g(x) \quad = \quad 0 \bmod (x^{q+1} - a). \tag{3.19}$$

A Equação (3.19) implica que todas as raízes do polinômio $x^{q+1} - a$ estão presentes em $(x^i - 1)m(x)g(x)$. Como grau$[m(x)g(x)] \leq q$, no mínimo uma raiz de $x^{q+1} - a$ é comum a $x^i - 1$. Por hipótese, todas as raízes de $x^{q+1} - a$ que possuem ordem multiplicativa diferente de $q^2 - 1$ estão em $g(x)$. Desta forma, as raízes de $x^{q+1} - a$ comuns a $x^i - 1$ possuem ordem multiplicativa $q^2 - 1$. Portanto, $i = q^2 - 1$ é o valor mínimo para que a Equação (3.19) seja satisfeita e, sendo assim, todas as palavras-código não-nulas de $\mathcal{C}$ possuem ordem constacíclica plena. ■

É interessante utilizar os Exemplos 3.7 e 3.8 para ilustrar os resultados enunciados no Teorema 3.12. Os códigos daqueles exemplos, possuem polinômios geradores que são fatores de $x^6 - 3$ o qual possui quatro raízes com ordem multiplicativa 24, $\{\alpha, \alpha^5, \alpha^{13}, \alpha^{17}\}$ e duas raízes com ordem multiplicativa 8, $\{\alpha^9, \alpha^{21}\}$. No Exemplo 3.7, o código constacíclico $(6, 2, 3)$ é gerado por $g(x) = M_1(x)M_{13}(x)$ cujas raízes são $\{\alpha, \alpha^5, \alpha^{13}, \alpha^{17}\}$. Observe que as raízes de $x^6 - 3$ que não têm ordem multiplicativa 24 não fazem parte de $g(x)$. Logo, este código não satisfaz a construção do Teorema 3.12. Além do mais, este código possui todas as palavras-código não-nulas com ordem constacíclica 8 (Vide Tabela 3.3). Entretanto, no Exemplo 3.8, o código constacíclico $(6, 2, 5)$ é gerado por $g'(x) = M_1(x)M_9(x)$ e, neste caso, as raízes de $x^6 - 3$ que não têm ordem multiplicativa 24 estão em $g'(x)$. Logo, todas as palavras deste código possuem ordem constacíclica plena (24). Este resultado já era esperado, pois as palavras-código não nulas do código $(6, 2, 5)$ gerado por $g'(x) = M_1(x)M_9(x)$ são formadas pelos

deslocamentos constacíclicos de $g'(x)$ cuja ordem constacíclica é plena. Ainda pelo Teorema 3.12, não há dúvidas que o código constacíclico gerado por $g''(x) = M_9(x)M_{13}(x)$ (Tabela 3.1) possui todas as palavras-código não-nulas com ordem constacíclica plena. Finalmente, caso a escolha do polinômio gerador do código constacíclico não satisfaça as condições do Teorema 3.12, ainda é possível prever qual a ordem constacíclica das palavras-código. No código $(6, 2, 3)$ do Exemplo 3.7, o polinômio $M_9(x)$, que possui raízes com ordem multiplicativa 8, não é fator do polinômio gerador, logo o código possui palavras com ordem constacíclica 8.

Embora o Teorema 3.12 garanta uma maneira de construir códigos constacíclicos cujas palavras-código não-nulas tenham ordem constacíclica plena, não se tem a garantia de que estes códigos também serão MDS, já que a escolha do polinômio gerador para garantir que ele seja MDS está relacionada com a escolha das raízes consecutivas de $x^{q+1} - a$. Para construir códigos constacíclicos que sejam MDS e que possuam todas as palavras-código com ordem constacíclica plena limita-se as possíveis escolhas para o polinômio gerador desses códigos, visto que é necessário se preocupar em selecionar raízes de $x^{q+1} - a$ que sejam consecutivas e que possuam ordem multiplicativa diferente de $q^2 - 1$. Nem sempre é possível escolher polinômios geradores com essas características. Porém, há um caso particular do Teorema 3.12 [36] que garante a construção de códigos constacíclicos MDS cujas palavras-código não-nulas tenham ordem constacíclica plena.

Lema 3.1

Se $q = 2^m - 1$ e m é um número inteiro ímpar tal que $2^m - 1$ seja um primo de Mersenne [49], *então todas as raízes de $x^{q+1} - a$ possuem ordem multiplicativa igual a $q^2 - 1$.* □

Demonstração: Como já é conhecido, as raízes de $x^{q+1} - a$ pertencem a $GF(q^2)$ e podem ser escritas na forma $\alpha^{1+(q-1)i}$, para $0 \leq i \leq q$, em que α é um elemento primitivo do grupo multiplicativo de $GF(q^2)$, ou seja, a ordem multiplicativa de α é igual a $\text{ord}(\alpha) = q^2 - 1$. Fazendo $\beta = \alpha^{1+(q-1)i}$, a ordem multiplicativa de β é dada por [10, pág. 35]

$$\begin{aligned} \text{ord}(\beta) &= \frac{\text{ord}(\alpha)}{\text{mdc}\,[\text{ord}(\alpha), 1 + (q-1)i]} \\ &= \frac{q^2 - 1}{\text{mdc}\,[q^2 - 1, 1 + (q-1)i]}. \end{aligned} \tag{3.20}$$

Substituindo $q = 2^m - 1$ no denominador da Equação (3.20)

$$\begin{aligned} \text{mdc}\,[2^{2m} - 2^{m+1}, 1 + (2^m - 2)i] &= \text{mdc}\,[2^m(2^m - 2), 1 + (2^m - 2)i] \\ &= 1. \end{aligned}$$

Pois, $1+(2^m-2)i$ é um número ímpar e não possui fatores comuns com 2^m ou 2^m-2 que são números pares. Portanto, todas as raízes de $x^{q+1}-a$ possuem ordem multiplicativa $\text{ord}(\beta)=q^2-1$ quando q é um primo de Mersenne. ■

Corolário 3.1 – Teorema 3.12

Se $q=2^m-1$ é um primo de Mersenne, então todas as palavras-código não-nulas de um código constacíclico q-ário $(q+1,k,d_{\min})$, cujo polinômio gerador $g(x)$ é fator de $x^{q+1}-a$, possuem ordem constacíclica plena. □

O Corolário 3.1 garante que se $q=2^m-1$ é um primo de Mersenne, não é necessário escolher para o polinômio gerador $g(x)$ de um código constacíclico as raízes de $x^{q+1}-a$ que possuam ordem multiplicativa diferente de q^2-1, pois todas as raízes, neste caso, são elementos primitivos do grupo multiplicativo de $GF(q^2)$. Desta forma, se são selecionadas $2t$ raízes consecutivas de $x^{q+1}-a$ para o polinômio gerador de um código constacíclico, então é obtido um código constacíclico MDS com todas as palavras-código não-nulas com ordem cíclica plena.

3.4.3 Classes de Equivalência Constacíclica

Definição 3.13 – Classe de equivalência constacíclica

Considere dois polinômios-código $c_1(x)$ e $c_2(x)$ pertencentes a um código constacíclico q-ário C cujos polinômios-código são reduzidos $\text{mod}(x^{q+1}-a)$. *Diz-se que $c_1(x)$ e $c_2(x)$ pertencem à mesma* ***classe de equivalência constacíclica*** *se $x^i c_1(x)=c_2(x) \bmod (x^{q+1}-a)$ para $1\leq i<q^2-1$. Se $c_1(x)$ tem ordem constacíclica igual a j, então a classe de equivalência constacíclica que contém $c_1(x)$ possui j polinômios-código, que correspondem aos deslocamentos constacíclicos de $c_1(x)$, e a classe de equivalência constacíclica, da qual $c_1(x)$ agora é denominado* ***líder****, também tem ordem constacíclica igual a j.* □

Decorre da Definição 3.13 que a palavra-código **0** constitui uma classe de equivalência constacíclica de ordem igual a 1. Além do mais, qualquer palavra-código pertencente a uma mesma classe de equivalência pode ser definida como líder de sua classe.

Pode-se exemplificar o uso da Definição 3.13 utilizando a Tabela 3.3 na qual estão todas as palavras não-nulas do código do Exemplo 3.7. As palavras-código $\mathbf{c}_1$, $\mathbf{c}_2$ e $\mathbf{c}_3$ são líderes de suas respectivas classes de equivalência constacíclica. Essas palavras-código possuem todas

ordem constacíclica igual a 8, logo cada uma dá origem, por meio de seus deslocamentos constacíclicos, a uma classe de equivalência com ordem constacíclica igual a 8 e com oito elementos cada. Portanto, o código do Exemplo 3.7 possui uma classe de equivalência constacíclica com ordem constacíclica igual a 1 e três classes de equivalência com ordem constacíclica 8. Para finalizar, o código dado no Exemplo 3.8 possui duas classes de equivalência constacíclica, uma que tem a palavra-código **0** como líder e, portanto, tem ordem constacíclica igual a 1 e outra com ordem constacíclica igual a 24 a qual tem como líder o polinômio gerador do código.

Capítulo 4

Construção de Códigos Ciclicamente Permutáveis

Eu ouço, eu esqueço. Eu vejo, eu lembro. Eu faço, eu aprendo.

— Confúcio

Neste capítulo, apresenta-se uma classe de códigos introduzida por Gilbert [27], os *códigos ciclicamente permutáveis* (códigos CP). Essa classe de códigos foi utilizada por A, Györfi e Massey [26] para construir sequências de protocolo para o canal de colisão sem realimentação (Capítulo 2) no caso de M usuários ativos de um total de U. Para construir sequências de protocolo por meio dessa classe de códigos, são utilizados os códigos constacíclicos discutidos no Capítulo 3.

Este capítulo inicia abordando uma relação adequada para converter arranjos bidimensionais em N-uplas binárias. Depois, define-se uma representação cíclica para os elementos de um corpo finito, $GF(q)$. Utilizando estes dois resultados iniciais, juntamente com os códigos constacíclicos (Capítulo 3), mostra-se como construir códigos cíclicos binários não-lineares. Finalmente, são apresentadas as construções para códigos CP.

4.1 Construção de Códigos Cíclicos Binários

4.1.1 Arranjos Bidimensionais e N-uplas

O objetivo é estabelecer uma correspondência um-a-um entre arranjos bidimensionais e N-uplas. Os arranjos bidimensionais considerados são semelhantes ao arranjo $A_{m\times n}$ mostrado em (4.1), cujos elementos $a(i,j)$, $i=\{0,1,\ldots,m-1\}$ e $j=\{0,1,\ldots,n-1\}$, pertencem a um alfabeto arbitrário.

$$A=\begin{bmatrix} a(0,0) & a(0,1) & \ldots & a(0,n-1) \\ a(1,0) & a(1,1) & \ldots & a(1,n-1) \\ \vdots & \vdots & \ddots & \vdots \\ a(m-1,0) & a(m-1,1) & \ldots & a(m-1,n-1) \end{bmatrix}_{m\times n}. \tag{4.1}$$

Em [26] foi utilizada uma correspondência um-a-um entre arranjos bidimensionais e N-uplas, com $N=mn$, que é garantida pelo *teorema chinês do resto* [49]. Nesta situação, é necessário que m e n sejam primos entre si, isto é, $\text{mdc}(m,n)=1$. Nesta dissertação, utiliza-se uma correspondência entre arranjos bidimensionais e N-uplas que foi proposta em [50] e apresentada de uma forma mais simples em [36]. Tal correspondência é distinta da proposta utilizada em [26] e, além disso, não necessita que m e n sejam primos entre si.

De acordo com [36], [50], para m e n números inteiros a relação que estabelece uma correspondência um-a-um entre o arranjo $A_{m\times n}$ da Equação (4.1) e N-uplas da forma $\mathbf{b}=(b_0,b_1,\ldots,b_{mn-1})$, ambos com elementos pertencentes a um mesmo alfabeto, é dada por

$$b_{in+j}=a(i,j),\ 0\leq i\leq m-1 \text{ e } 0\leq j\leq n-1. \tag{4.2}$$

Exemplo 4.1

Considere o arranjo $A_{3\times 3}$ a seguir

$$A=\begin{bmatrix} a(0,0) & a(0,1) & a(0,2) \\ a(1,0) & a(1,1) & a(1,2) \\ a(2,0) & a(2,1) & a(2,2) \end{bmatrix}_{3\times 3} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}_{3\times 3}. \tag{4.3}$$

Pela relação dada em (4.2), com $n=3$, a 9-upla correspondente ao arranjo dado em (4.3) é $\mathbf{b}=(b_0,b_1,b_2,b_3,b_4,b_5,b_6,b_7,b_8)=(1,2,3,4,5,6,7,8,9)$. *Veja os cálculos na Tabela 4.1.* □

Tabela 4.1: *Correspondência entre os elementos do arranjo $A_{3\times3}$ e os elementos da 9-upla* **b**.

(i,j)	$in+j$	$b_{in+j}=a(i,j)$
$(0,0)$	0	$b_0=a(0,0)=1$
$(0,1)$	1	$b_1=a(0,1)=2$
$(0,2)$	2	$b_2=a(0,2)=3$
$(1,0)$	3	$b_3=a(1,0)=4$
$(1,1)$	4	$b_4=a(1,1)=5$
$(1,2)$	5	$b_5=a(1,2)=6$
$(2,0)$	6	$b_6=a(2,0)=7$
$(2,1)$	7	$b_7=a(2,1)=8$
$(2,2)$	8	$b_8=a(2,2)=9$

Definição 4.1 – Operador **DB(.)**

O operador $\mathbf{DB}(.)$ *atua sobre um arranjo bidimensional* $A_{m\times n}$*, produzindo o arranjo* $A''_{m\times n}$*, da seguinte forma:*

1. *O operador* $\mathbf{DB}(.)$*, inicialmente, desloca ciclicamente todas as colunas do arranjo* $A_{m\times n}$ *uma posição para a direita produzindo um novo arranjo* $A'_{m\times n}$*;*
2. *depois, o operador* $\mathbf{DB}(.)$ *desloca ciclicamente uma posição para baixo a coluna mais à esquerda do arranjo* $A'_{m\times n}$ *produzindo o arranjo* $A''_{m\times n}$*.* □

Exemplo 4.2

Aplicando o operador $\mathbf{DB}(.)$ *da Definição 4.1 ao arranjo* $A_{3\times3}$ *do Exemplo 4.1 obtém-se*

$$A=\begin{bmatrix}1&2&3\\4&5&6\\7&8&9\end{bmatrix}_{3\times3}\xrightarrow{\mathbf{DB}(A)}A''=\begin{bmatrix}9&1&2\\3&4&5\\6&7&8\end{bmatrix}_{3\times3}. \tag{4.4}$$

□

A 9-upla $\mathbf{b}''=(9,1,2,3,4,5,6,7,8)$ é obtida aplicando a relação dada em (4.2) ao arranjo A'' do Exemplo 4.2. Nota-se que $\mathbf{b}''$ corresponde a um deslocamento cíclico para direita da 9-upla $\mathbf{b}=(1,2,3,4,5,6,7,8,9)$ do Exemplo 4.1. Desta forma, definindo um operador $\mathbf{S}^i(.)$, o qual aplicado a uma N-upla a desloca ciclicamente de i posições para a direita, a relação entre as 9-uplas $\mathbf{b}$ e $\mathbf{b}''$ pode ser expressa por intermédio deste operador como $\mathbf{S}(\mathbf{b})=\mathbf{b}''$. Vale lembar que a ordem cíclica, definida em termos do operador $\mathbf{S}^i(.)$, é o menor valor de i

para o qual $\mathbf{S}^i(\mathbf{b}) = \mathbf{b}$; e os possíveis valores de i são os divisores de N (comprimento da N-upla). O teorema a seguir faz uso dos operadores $\mathbf{DB}(.)$ e $\mathbf{S}^i(.)$ para estabelecer um resultado importante relacionando um conjunto de arranjos bidimensionais $m \times n$ e o conjunto de mn-uplas derivado deste conjunto de arranjos por meio da relação dada em (4.2).

Teorema 4.1

Considere um conjunto $\mathcal{X}$ formado por arranjos bidimensionais $m \times n$ cujos elementos pertencem a um alfabeto arbitrário. O conjunto $\mathcal{X}$ será fechado em relação à operação realizada por $\mathbf{DB}(.)$ se e somente se o conjunto correspondente de mn-uplas for fechado em relação à operação realizada por $\mathbf{S}^i(.)$. □

Demonstração: Seja $A_{m\times n}$ um arranjo bidimensional pertencente ao conjunto $\mathcal{X}$ e seja $\mathbf{b}$ a mn-upla binária correspondente ao arranjo $A_{m\times n}$ de acordo com a Relação (4.2). Seja $A''_{m\times n}$ um arranjo bidimensional tal que $\mathbf{DB}(A_{m\times n}) = A''_{m\times n}$. A relação entre os elementos dos arranjos $A_{m\times n}$ e $A''_{m\times n}$ é dada por

$$a''(i,j) = a(i \bmod m, j-1 \bmod n), \text{ para } 0 \le i \le m-1,\ 1 \le j \le n-1, \text{ e} \quad (4.5)$$

$$a''(i,0) = a(i-1 \bmod m, n-1), \text{ para } 0 \le i \le m-1 \text{ e } j = 0, \quad (4.6)$$

em que $l \bmod y$ denota o resto da divisão quando l é dividido por y. Sendo $\mathbf{S}(\mathbf{b}) = \mathbf{b}'$, a relação entre os elementos das mn-uplas $\mathbf{b}$ e $\mathbf{b}'$ é tal que

$$b'_{in+j \bmod mn} = b_{in+j-1 \bmod mn}, \text{ para } 0 \le i \le m-1 \text{ e } 0 \le j \le n-1. \quad (4.7)$$

A mn-upla $\mathbf{b}''$ é obtida aplicando-se a Relação (4.2) ao arranjo $A''_{m\times n}$. Logo, usando as Relações (4.5) e (4.6) e para $i = \{0, 1, 2, \ldots, m-1\}$ obtém-se

$$b''_{in+j \bmod mn} = b_{in+j-1 \bmod mn},\ 1 \le j \le n-1, \text{ e} \quad (4.8)$$

$$b''_{in+j \bmod mn} = b_{in-1 \bmod mn},\ j = 0. \quad (4.9)$$

Comparando as Relações (4.8) e (4.9) com a Relação (4.7), conclui-se que $\mathbf{b}' = \mathbf{b}''$ e, assim, $\mathbf{S}(\mathbf{b}) = \mathbf{b}''$. Portanto, uma condição suficiente para o conjunto $\mathcal{X}$ ser fechado com relação à operação realizada por $\mathbf{DB}(.)$ é o conjunto de mn-uplas ser fechado com relação à operação realizada por $\mathbf{S}^i(.)$. De maneira análoga, pode-se mostrar que uma condição necessária para o conjunto $\mathcal{X}$ ser fechado com relação à operação realizada por $\mathbf{DB}(.)$ é o conjunto de mn-uplas ser fechado em relação à operação realizada por $\mathbf{S}^i(.)$. ■

4.1.2 Uma Representação Cíclica para os Elementos de $GF(q)$

Nesta subseção, o interesse é representar os elementos $\{0, 1, 2, \ldots, q-1\}$ de $GF(q)$ por meio de N-uplas binárias. Sendo q um número primo ímpar, $q-1$ sempre será um número par. Conforme mencionado anteriormente, a ordem cíclica de uma N-upla é igual a N ou a um de seus divisores. Logo, para $N = q-1$, sempre existe uma $(q-1)$-upla binária de ordem cíclica igual a 2 que corresponde a $(q-1)$-upla de peso igual a $(q-1)/2$ cujas coordenadas assumem valores alternados 0 ou 1, isto é, $(1, 0, 1, 0, \ldots, 1, 0)$ ou $(0, 1, 0, 1, \ldots, 0, 1)$. Além do mais, sempre existe pelo menos uma $(q-1)$-upla binária de ordem cíclica $q-1$ que corresponde à $(q-1)$-upla de peso unitário. Porém, pode haver outras $(q-1)$-uplas com ordem cíclica igual a $q-1$, além da que foi citada, e que não tenham peso igual a 1.

Exemplo 4.3

Para $q = 7$, as 6-uplas binárias $\mathbf{v}_1 = (1, 0, 0, 0, 0, 0)$, $\mathbf{v}_2 = (1, 1, 1, 0, 0, 0)$ e $\mathbf{v}_3 = (1, 1, 0, 1, 0, 0)$ possuem ordem cíclica igual a 6, enquanto que a 6-upla binária $\mathbf{v}_4 = (1, 0, 1, 0, 1, 0)$ possui ordem cíclica igual a 2. □

A definição a seguir estabelece uma representação para os elementos de $GF(q)$ por meio de $(q-1)$-uplas binárias.

Definição 4.2 – Representação-**V**

Seja $\mathbf{v}$ uma $(q-1)$-upla binária cuja ordem cíclica é igual a $q-1$. Define-se a ***representação-V****, como sendo uma representação para os elementos de $GF(q)$ por intermédio de $(q-1)$-uplas binárias tal que os elementos não-nulos a^i, $i = 0, 1, 2, \ldots, q-2$, são representados pelas $(q-1)$-uplas binárias $\mathbf{S}^i(\mathbf{v})$ em que a é um elemento gerador do grupo multiplicativo de $GF(q)$ e $\mathbf{S}^i(.)$ é o operador que desloca ciclicamente de i posições para a direita a $(q-1)$-upla $\mathbf{v}$. Além disso, o elemento 0 pode ser representado por uma $(q-1)$-upla binária não-nula $\mathbf{v}'$ e seus deslocamentos cíclicos tal que $\mathbf{v}' \neq \mathbf{S}^i(\mathbf{v})$ para $0 \leq i \leq q-2$. Em particular, $\mathbf{v}'$ pode ser escolhida como a $(q-1)$-upla toda nula $\mathbf{0}$.* □

Exemplo 4.4 – Continuação do Exemplo 4.3

*Considere $q = 7$, $a = 3$, $\mathbf{v}' = \mathbf{v}_4 = (1, 0, 1, 0, 1, 0)$ e $\mathbf{v} = \mathbf{v}_3 = (1, 1, 0, 1, 0, 0)$. A representação-**V** resultante para $GF(7)$ é dada na Tabela 4.2.* □

A representação-**V** dada na Definição 4.2 nada mais é que um conjunto de $(q-1)$-uplas binárias, logo é possível interpretá-la como um código de bloco cíclico não-linear. Assim,

Tabela 4.2: *representação-***V** *para o elementos de* $GF(7)$.

a^i	6-upla
0	$(1,0,1,0,1,0)$
0	$(0,1,0,1,0,1)$
3^0	$(1,1,0,1,0,0)$
3^1	$(0,1,1,0,1,0)$
3^2	$(0,0,1,1,0,1)$
3^3	$(1,0,0,1,1,0)$
3^4	$(0,1,0,0,1,1)$
3^5	$(1,0,1,0,0,1)$

pode-se associar uma distância mínima, denotada por $d(\mathbf{v})$, cuja definição é a mesma dada na Definição 3.5. Baseando-se nesta afirmação, se existir uma representação-**V** na qual a distância de Hamming entre qualquer par de $(q-1)$-uplas deste conjunto for igual a $d(\mathbf{v})$, então esta representação é dita ser *equidistante*. Claramente, a representação-**V** dada na Tabela 4.2 não é uma representação equidistante. Se a representação-**V** for limitada aos elementos do grupo multiplicativo de $GF(q)$, e a denotando por representação-$\mathbf{V}^*$, então, por exemplo, para $\mathbf{v}^* = (1,0,0,\ldots,0)$ ou $\mathbf{v}^* = (0,1,1,\ldots,1)$ a representação-$\mathbf{V}^*$ é equidistante com $d(\mathbf{v}^*) = 2$.

4.1.3 CONSTRUÇÕES

Neste ponto, o objetivo é construir códigos cíclicos binários não-lineares. Para isto, são utilizados os códigos constacíclicos lineares q-ários (n,k,d), definidos no Capítulo 3, em que n, k e d representam, respectivamente, o comprimento das palavras, a dimensão e a distância mínima do código. Juntamente com esses códigos, utiliza-se a representação cíclica definida para $GF(q)$ e a relação entre arranjos bidimensionais e N-uplas. O procedimento para construir os códigos cíclicos binários é descrito a seguir. Primeiro, cada palavra q-ária $\mathbf{c} = (c_0, c_1, \ldots, c_{n-1})$, pertencente ao código constacíclico, é mapeada em um arranjo bidimensional cujas colunas são as transpostas das $(q-1)$-uplas binárias, dadas pela representação-**V**, para cada coordenada c_i, $0 \leq i \leq n-1$, da palavra-código $\mathbf{c}$. Depois, os arranjos bidimensionais são convertidos em N-uplas binárias por meio da relação dada em (4.2).

Antes de enunciar o Teorema 4.2, que estabelece o principal resultado para gerar códigos cíclicos binários, vale ressaltar que na representação-**V**, Definição 4.2, o elemento 0 é representado por uma $(q-1)$-upla $\mathbf{v}'$ e seus deslocamentos cíclicos, logo uma palavra-código q-ária

$\mathbf{c} = (c_0, c_1, \ldots, c_{n-1})$ que tenha uma ou mais coordenadas nulas, $c_i = 0$ para $0 \leq i \leq n-1$, pode ser associada a mais de um arranjo bidimensional e, consequentemente, a mais de uma N-upla binária. Sendo assim, deve-se ter um cuidado especial para que as q^k palavras do código constacíclico representem exatamente q^k N-uplas binárias correspondendo as palavras do código cíclico binário. Para isto, as palavras do código constacíclico devem ser separadas em classes de equivalência constacíclica conforme a Definição 3.13. Depois disso, seleciona-se, arbitrariamente, uma palavra-código $\mathbf{c}$ para ser líder de sua respectiva classe de equivalência constacíclica e a ela associa-se um arranjo bidimensional $A_{(q-1)\times n}$. Se $\mathbf{c}$ possui todas as coordenadas não-nulas, o mapeamento de $\mathbf{c}$ para $A_{(q-1)\times n}$ é um-a-um e, portanto, não há problemas. Entretanto, se $\mathbf{c}$ possui uma ou mais coordenadas nulas, o mapeamento de $\mathbf{c}$ para $A_{(q-1)\times n}$ é feito escolhendo-se, inicialmente, uma $(q-1)$-upla $\mathbf{v}'$ arbitrária para representar o elemento 0 e mantendo fixa esta escolha. Os arranjos associados às palavras-código que pertencem à mesma classe de equivalência constacíclica de $\mathbf{c}$, são obtidos aplicando-se o operador $\mathbf{DB}(.)$ ao arranjo $A_{(q-1)\times n}$ de modo que a palavra-código $\mathbf{c}'$, correspondente ao i-ésimo deslocamento constacíclico de $\mathbf{c}$, é representada pelo arranjo bidimensional $Z_{(q-1)\times n}$ obtido ao aplicar o operador $\mathbf{DB}(.)$ i vezes ao arranjo $A_{(q-1)\times n}$. Daqui em diante, faz-se referência a esse processo como *geração biunívoca de arranjos*.

Teorema 4.2 – Códigos cíclicos binários

Seja q um número primo, $q > 3$, e $\mathcal{C}$ um código constacíclico linear q-ário de parâmetros (n, k, d). Considere que cada palavra-código, $\mathbf{c} = (c_0, c_1, \ldots, c_{n-1})$, líder de classe de equivalência constacíclica determine um arranjo bidimensional $A_{(q-1)\times n}$ de modo que a i-ésima coluna de $A_{(q-1)\times n}$ seja a transposta de uma $(q-1)$-upla binária que corresponde à representação-$\mathbf{V}$ da i-ésima componente de $\mathbf{c}$ e defina $\mathbf{b}$ como sendo a N-upla, com $N = (q-1)n$, que corresponde ao arranjo $A_{(q-1)\times n}$ por meio da Relação (4.2). Além do mais, considere que as demais palavras-código são mapeadas em N-uplas binárias com o auxílio do processo de ***geração biunívoca de arranjos****. Então, o conjunto de q^k N-uplas binárias correspondentes às q^k palavras-código de $\mathcal{C}$ formam um código cíclico binário de distância mínima $d_{\min} \geq dd(\mathbf{v})$ com igualdade se a representação-$\mathbf{V}$ de $GF(q)$ for equidistante.* □

Demonstração: Seja $\mathbf{c} \in \mathcal{C}$ uma palavra-código líder de classe de equivalência constacíclica e seja $A_{(q-1)\times n}$ o arranjo bidimensional correspondente a $\mathbf{c}$. Uma vez que $\mathcal{C}$ é um código linear constacíclico, deslocar constaciclicamente para a direita a palavra-código $\mathbf{c}$ produz uma palavra-código $\mathbf{c}' \in \mathcal{C}$ cujo arranjo bidimensional, denotado por $A'_{(q-1)\times n}$, é

tal que $\mathbf{DB}(A_{(q-1)\times n}) = A'_{(q-1)\times n}$. Sendo assim, os q^k arranjos bidimensionais, correspondentes às palavras-código de $\mathcal{C}$, formam um conjunto $\mathcal{Y}$ fechado em relação à operação realizada pelo operador $\mathbf{DB}(.)$. Segue do Teorema 4.1 que o conjunto de q^k N-uplas binárias $\mathbf{b}$, com $N = (q-1)n$, obtidas ao se aplicar a relação dada em (4.2) aos arranjos bidimensionais do conjunto $\mathcal{Y}$, é um conjunto fechado em relação à operação realizada pelo operador $\mathbf{S}^i(.)$ e, portanto, é um código cíclico binário.

Para concluir a demonstração, resta deduzir o limitante inferior dado para $d_{\min}$. Como o código $\mathcal{C}$ tem distância mínima d, duas palavras-código distintas $\mathbf{c}_1$ e $\mathbf{c}_2$ diferem em d coordenadas no mínimo, isto é, a distância de Hamming entre elas satisfaz $d(\mathbf{c}_1, \mathbf{c}_2) \geq d$. Sendo assim, as N-uplas binárias $\mathbf{b}_1$ e $\mathbf{b}_2$, correspondendo a $\mathbf{c}_1$ e $\mathbf{c}_2$, respectivamente, diferem em $dd(\mathbf{v})$ coordenadas no mínimo, em que $d(\mathbf{v})$ é a distância mínima da representação-$\mathbf{V}$. Uma vez que $d(\mathbf{c}_1, \mathbf{c}_2) \geq d$ é satisfeita com igualdade para algumas escolhas de $\mathbf{c}_1$ e $\mathbf{c}_2$, conclui-se que $d_{\min} \geq dd(\mathbf{v})$ e ela é satisfeita com igualdade caso a representação-$\mathbf{V}$ seja equidistante. ■

Exemplo 4.5

Considere as 6*-uplas* 5*-árias* $(4,0,2,0,1,0)$ *e* $(2,1,1,3,3,4)$*, palavras do código constacíclico do Exemplo 3.7, e a representação-*$\mathbf{V}$ *em que* $\mathbf{v} = (1,1,0,0)$ *e* $\mathbf{v}' = (1,0,1,0)$*. Sendo assim, a representação-*$\mathbf{V}$ *para os elementos* $\{0,1,2,3,4\}$ *de* $GF(5)$ *é:* $0 \leftrightarrow (1,0,1,0)$ *ou* $0 \leftrightarrow (0,1,0,1)$, $3^0 = 1 \leftrightarrow (1,1,0,0)$, $3^1 = 3 \leftrightarrow (0,1,1,0)$, $3^2 = 4 \leftrightarrow (0,0,1,1)$ *e* $3^3 = 2 \leftrightarrow (1,0,0,1)$*. Logo, os arranjos bidimensionais* $A_{4\times 6}$ *correspondentes às duas palavras-código dadas são:*

$$(4,0,2,0,1,0) \Leftrightarrow \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}_{4\times 6} \quad \text{e} \quad (2,1,1,3,3,4) \Leftrightarrow \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{4\times 6}.$$

As respectivas 24*-uplas são*

$$(4,0,2,0,1,0) \quad \Leftrightarrow \quad (0,1,1,1,1,1,0,0,0,0,1,0,1,1,0,1,0,1,1,0,1,0,0,0) \ \textit{e}$$
$$(2,1,1,3,3,4) \quad \Leftrightarrow \quad (1,1,1,0,0,0,0,1,1,1,1,0,0,0,0,1,1,1,1,0,0,0,0,1). \qquad \square$$

Os códigos cíclicos binários construídos por meio do Teorema 4.2 podem ser de peso constante ou não, tal característica depende da escolha feita para as $(q-1)$-uplas $\mathbf{v}$ e $\mathbf{v}'$ da representação-$\mathbf{V}$. Se $w(\mathbf{v}) = w(\mathbf{v}')$, então o código cíclico binário é de peso constante com

$w = nw(\mathbf{v})$. Caso contrário, se $w(\mathbf{v}) \neq w(\mathbf{v}')$, então o código cíclico binário não é de peso constante.

A seguir, são apresentadas algumas construções de códigos cíclicos binários baseadas no Teorema 4.2.

Construção 4.1

Seja q um número primo, $q > 3$, $n = q+1$ e k um número par tal que $2 \leq k \leq q-1$. Escolha uma representação-$\mathbf{V}$ com $w(\mathbf{v}) = w(\mathbf{v}')$, e um código constacíclico linear q-ário $(q+1, k, q-k+2)$ (MDS). Pelo Teorema 4.2, obtém-se um código cíclico binário de peso constante com q^k palavras-código de comprimento $N = q^2 - 1$ e com distância mínima $d_{\min} \geq (q-k+2)d(\mathbf{v})$. □

Construção 4.2

Seja q um número primo, $q > 3$, $n = q+1$ e k um número par tal que $2 \leq k \leq q-1$. Escolha uma representação-$\mathbf{V}$ com $w(\mathbf{v}) = w(\mathbf{v}')$, e um código constacíclico linear q-ário $(q+1, k, d)$ construído de acordo com o Teorema 3.12. Pelo Teorema 4.2, obtém-se um código cíclico binário de peso constante com q^k palavras-código de comprimento $N = q^2 - 1$, com distância mínima $d_{\min} \geq dd(\mathbf{v})$ e tal que todas as palavras-código têm ordem cíclica plena. □

Construção 4.3

Seja $q = 2^m - 1$ um primo de Mersenne, $n = q + 1$ e k um número par tal que $2 \leq k \leq q - 1$. Escolha uma representação-$\mathbf{V}$ com $w(\mathbf{v}) = w(\mathbf{v}')$, e um código constacíclico linear q-ário $(q+1, k, q-k+2)$ (MDS). Pelo Teorema 4.2, obtém-se um código cíclico binário de peso constante com q^k palavras-código de comprimento $N = q^2-1$, com distância mínima $d_{\min} \geq (q-k+2)d(\mathbf{v})$ e tal que todas as palavras-código têm ordem cíclica plena. □

Na construção de códigos CP é interessante escolher uma representação-$\mathbf{V}$ com $\mathbf{v} = (1, 0, 0, \ldots, 0)$ e $\mathbf{v}' = (0, 0, 0, \ldots, 0)$. Portanto, substituindo esta representação-$\mathbf{V}$ nas Construções 4.1, 4.2 e 4.3 obtém-se as construções a seguir.

Construção 4.4

Seja q um número primo, $q > 3$, $n = q + 1$ e k um número par tal que $2 \leq k \leq q - 1$. Escolha uma representação-$\mathbf{V}$ com $\mathbf{v} = (1, 0, 0, \ldots, 0)$ e $\mathbf{v}' = (0, 0, 0, \ldots, 0)$, e um código constacíclico linear q-ário $(q+1, k, q-k+2)$ (MDS). Pelo Teorema 4.2, obtém-se um código cíclico binário com q^k palavras-código de comprimento $N = q^2 - 1$ e com distância mínima $d_{\min} = q - k + 2$. □

Construção 4.5

Seja q um número primo, $q > 3$, $n = q + 1$ e k um número par tal que $2 \leq k \leq q - 1$. Escolha uma representação-$\mathbf{V}$ com $\mathbf{v} = (1, 0, 0, \ldots, 0)$ e $\mathbf{v}' = (0, 0, 0, \ldots, 0)$, e um código constacíclico linear q-ário $(q + 1, k, d)$ construído de acordo com o Teorema 3.12. Pelo Teorema 4.2, obtém-se um código cíclico binário com q^k palavras-código de comprimento $N = q^2 - 1$, com distância mínima $d_{\min} = d$ e tal que todas as palavras-código não nulas têm ordem cíclica plena. □

Construção 4.6

Seja $q = 2^m - 1$ um primo de Mersenne, $n = q + 1$ e k um número par tal que $2 \leq k \leq q - 1$. Escolha uma representação-$\mathbf{V}$ com $\mathbf{v} = (1, 0, 0, \ldots, 0)$ e $\mathbf{v}' = (0, 0, 0, \ldots, 0)$, e um código constacíclico linear q-ário $(q+1, k, q-k+2)$ (MDS). Pelo Teorema 4.2, obtém-se um código cíclico binário com q^k palavras-código de comprimento $N = q^2-1$, com distância mínima $d_{\min} = q-k+2$ e tal que todas as palavras-código não-nulas têm ordem cíclica plena. □

Vale destacar que para as Construções 4.4 a 4.6, se no lugar de $\mathbf{v} = (1, 0, 0, \ldots, 0)$ for utilizada $\mathbf{v} = (0, 1, 1, \ldots, 1)$ os resultados obtidos seriam semelhantes, sendo o peso das palavras dos códigos cíclicos binários obtidos a única diferença.

4.2 Construção de Códigos Ciclicamente Permutáveis

Códigos ciclicamente permutáveis (códigos CP) foram introduzidos por Gilbert [27] em 1963. A definição a seguir é semelhante àquela introduzida em [27].

Definição 4.3

*Um **código ciclicamente permutável** é um código de bloco binário de comprimento N em que cada palavra-código tem ordem cíclica plena e tal que as palavras-código são ciclicamente distintas.* □

A Definição 4.3 garante que dada uma palavra do código, digamos $\mathbf{c}$, nenhuma outra palavra deste código pode ser obtida deslocando ciclicamente a palavra-código $\mathbf{c}$ uma ou mais vezes.

O conceito de classe de equivalência constacíclica (Definição 3.13), definido para as palavras de um código constacíclico, pode ser aplicado de maneira semelhante às palavras de um código cíclico ou para uma N-upla, em geral. Desta forma, sendo $\mathbf{b}$ uma N-upla l-ária, l um número inteiro, pode-se definir *classe de equivalência cíclica* como sendo o conjunto de N-uplas cujos elementos correspondem a todos os deslocamentos cíclicos distintos de $\mathbf{b}$. Em

termos do operador $\mathbf{S}^i(.)$, duas N-uplas, $\mathbf{b}$ e $\mathbf{b}'$, pertencem à mesma classe de equivalência cíclica se $\mathbf{S}^i(\mathbf{b}) = \mathbf{b}'$ para algum valor de i, $1 \leq i \leq N-1$. Se $\mathbf{b}$ tem ordem cíclica j, então a classe de equivalência a qual $\mathbf{b}$ pertence tem j N-uplas e, portanto, esta classe tem ordem cíclica j. Desta forma, um código CP pode ser definido, alternativamente, como um código de bloco binário tal que suas palavras-código pertencem a diferentes classes de equivalência cíclica cada uma com ordem cíclica igual ao comprimento das palavras-código.

Exemplo 4.6

Um código de bloco constituído pelas palavras-código $\{(0,0,0,1);(0,0,1,1);(0,1,1,1)\}$ *é um código CP. Observe que todas as palavras-código tem ordem cíclica igual a 4, ou seja, igual ao comprimento das palavras-código e nenhum deslocamento cíclico de uma palavra-código gera outra palavra-código.* □

A *distância mínima cíclica* de um código CP de comprimento N, denotada por d_c, é definida como a menor distância de Hamming entre uma palavra-código, digamos $\mathbf{c}$, e seus deslocamentos cíclicos $\mathbf{S}^i(\mathbf{c})$, $1 \leq i \leq N-1$, ou os deslocamentos cíclicos de uma outra palavra-código $\mathbf{S}^i(\mathbf{c}')$. Por exemplo, para determinar a distância mínima cíclica do código CP do Exemplo 4.6, deve-se construir as classes de equivalência cíclica, geradas por meio de cada uma das palavras deste código CP, e encontrar a menor distância de Hamming entre um par de 4-uplas de um total de doze 4-uplas binárias. Na Tabela 4.3 mostra-se as classes de equivalência cíclica para as palavras do código CP do Exemplo 4.6. Neste caso, $d_c = d_{\text{Hamming}}\{(0,0,1,1);(0,1,1,1)\} = 1$. Observando atentamente a Tabela 4.3, percebe-se que o conjunto de doze 4-uplas constitui um código cíclico binário, pois, de acordo com a Definição 3.7, os deslocamentos cíclicos de qualquer uma das doze 4-uplas pertence ao conjunto. Isto implica que o procedimento para calcular a distância mínima cíclica d_c do código CP do Exemplo 4.6, realizado anteriormente, é equivalente a calcular a distância mínima $d_{\min}$ do código cíclico binário composto pelas doze 4-uplas da Tabela 4.3. Portanto, em geral, a distância mínima cíclica d_c de um código CP é igual a distância mínima $d_{\min}$ do código cíclico binário obtido ao gerar as classes de equivalência cíclica para cada uma das palavras do código CP. Daqui em diante, denota-se um código CP por $CCP(N, M_c, d_c)$ em que N é o comprimento das palavras-código, M_c é o número de palavras-código e d_c é a distância mínima cíclica deste código.

Em geral, dado um código cíclico binário $(N, k, d_{\min})$, se seu dicionário for dividido em classes de equivalência cíclica e, no máximo, uma palavra-código de cada uma das distintas

Tabela 4.3: *Classes de equivalência cíclica para as palavras do código CP do Exemplo 4.6.*

i	$(0,0,0,1)$	$(0,0,1,1)$	$(0,1,1,1)$
1	$(1,0,0,0)$	$(1,0,0,1)$	$(1,0,1,1)$
2	$(0,1,0,0)$	$(1,1,0,0)$	$(1,1,0,1)$
3	$(0,0,1,0)$	$(0,1,1,0)$	$(1,1,1,0)$

classes de equivalência cíclica, de ordem N, for selecionada, então obtém-se um $CCP(N, M_c, d_c)$ com $d_c \geq d_{\min}$ e M_c igual ao número de palavras do código cíclico binário selecionadas. Isto implica que se pode obter códigos CP utilizando os códigos cíclicos binários das Construções 4.1 a 4.6 da subseção anterior. Um procedimento direto para realizar essa tarefa é denominado de *força-bruta* e é descrito na sequência. Inicialmente, escolhe-se, arbitrariamente, uma palavra $\mathbf{c}$ do código cíclico binário para ser uma palavra do código CP. Depois, outra palavra do código cíclico binário $\mathbf{c}'$ é escolhida e compara-se todos os deslocamentos cíclicos de $\mathbf{c}'$, $\mathbf{S}^i(\mathbf{c}')$, com a palavra-código $\mathbf{c}$. Se $\mathbf{S}^i(\mathbf{c}') = \mathbf{c}$, $1 \leq i \leq N-1$, então a palavra é descartada, caso contrário, $\mathbf{c}'$ é escolhida como uma palavra do código CP. Este processo continua até que todas as q^k palavras do código cíclico binário sejam testadas.

Se o número de palavras do código cíclico binário é elevado, então o processo de seleção por força-bruta é ineficiente. Além do mais, segundo A, Györfi e Massey [26], para o procedimento de geração de um código CP ser qualificado como *construção*, tal procedimento deve ser facilmente implementável. Construções de códigos CP, no sentido definido em [26], são apresentadas em [26], [29], [51], [52]. O teorema a seguir baseia-se no procedimento para geração de códigos CP usado em [51] e estabelece um resultado importante para o procedimento de geração de códigos CP proposto nesta dissertação.

Teorema 4.3

Seja $\mathcal{C}$ um código constacíclico linear q-ário $(q+1, k, d)$ gerado por $g(x)$ e seja $s(x)$ um polinômio de grau 2 que pertence ao expoente q^2-1 e é fator do polinômio de verificação de paridade $h(x)$. Além do mais, considere $m(x)$ um polinômio-mensagem cujo grau é menor ou igual a $k-3$. Então, as palavras-código $c(x)$ selecionadas tal que $c(x) = g(x)[1 + s(x)m(x)]$ têm ordem constacíclica plena e são constaciclicamente distintas. □

Demonstração: Se $c(x) \in \mathcal{C}$ tem ordem constacíclica plena, então o menor valor para o

qual i satisfaz

$$\begin{aligned} x^i c(x) &= c(x) \bmod x^{q+1} - a \text{ ou, equivalentemente,} \\ (x^i - 1)c(x) &= 0 \bmod x^{q+1} - a, \end{aligned} \tag{4.10}$$

é $i = q^2 - 1$. Visto que $s(x)$ é um fator de $h(x)$, pode-se escrever $h(x) = a(x)s(x)$, em que $a(x)$ é fator de $h(x)$. Substituindo $c(x)$ por $g(x)[1 + s(x)m(x)]$ em (4.10) obtém-se

$$\begin{aligned} (x^i - 1)g(x)[1 + s(x)m(x)] &= 0 \bmod g(x)h(x), \\ (x^i - 1)[1 + s(x)m(x)] &= 0 \bmod h(x) \\ &= 0 \bmod a(x)s(x). \end{aligned} \tag{4.11}$$

Como mdc$[1 + s(x)m(x), s(x)] = 1$, a Equação (4.11) é satisfeita se e somente se $s(x)$ tem fator comum com $x^i - 1$. Porém, por definição, $s(x)$ pertence ao expoente $q^2 - 1$. Assim, o menor valor de i para o qual a Equação (4.11) é satisfeita é $i = q^2 - 1$. Portanto, as palavras-código $c(x)$ selecionadas de modo que $c(x) = g(x)[1 + s(x)m(x)]$ têm ordem constacíclica plena.

Para provar que as palavras-código são constaciclicamente distintas, considere duas palavras-código, distintas, $c_1(x) = g(x)[1 + s(x)m_1(x)]$ e $c_2(x) = g(x)[1 + s(x)m_2(x)]$. Suponha que $c_1(x)$ e $c_2(x)$ pertençam à mesma classe de equivalência constacíclica. Ou seja,

$$x^i c_2(x) = c_1(x) \bmod x^{q+1} - a,\ 0 < i < q^2 - 1. \tag{4.12}$$

Manipulando algebricamente a Equação (4.12) obtém-se

$$\begin{aligned} x^i c_2(x) - c_1(x) &= 0 \bmod x^{q+1} - a, \\ x^i[1 + s(x)m_2(x)]g(x) - [1 + s(x)m_1(x)]g(x) &= 0 \bmod g(x)h(x), \\ x^i - 1 + s(x)[x^i m_2(x) - m_1(x)] &= 0 \bmod a(x)s(x). \end{aligned} \tag{4.13}$$

Para que a Equação (4.13) seja satisfeita, $s(x)$ deve ser fator de $x^i - 1$. Entretanto, esta condição é impossível, pois $s(x)$ pertence ao expoente $q^2 - 1$ e $0 < i < q^2 - 1$. Assim, a hipótese de que as palavras-código $c_1(x)$ e $c_2(x)$ pertencem à mesma classe de equivalência constacíclica é falsa e, portanto, elas são constaciclicamente distintas. ■

Construção 4.7

Seja q um número primo, $q > 3$, $n = q + 1$ e k um número par tal que $4 \leq k \leq q - 1$. Escolha uma representação-$\mathbf{V}$ com $\mathbf{v} = (1, 0, 0, \ldots, 0)$, $\mathbf{v}'$ com ordem cíclica $q - 1$ e $w(\mathbf{v}') \geq 3$, e um código C constacíclico linear q-ário $(q + 1, k, q - k + 2)$ (MDS). Aplicando o Teorema 4.2 a cada palavra-código $c(x)$ selecionada de acordo com o Teorema 4.3, obtém-se um $CCP(N, M_c, d_c)$ com $N = q^2 - 1$, $M_c = q^{k-2}$ e com distância mínima cíclica $d_c \geq (q - k + 2)d(\mathbf{v})$. □

De maneira análoga ao que é feito em [26], a eficiência do procedimento utilizado para gerar os códigos CP da Construção 4.7 é analisada. As palavras do código CP da Construção 4.7 pertencem a um código cíclico binário com $N = q^2 - 1$ e $M_p = q^k$ palavras-código. Desta forma, a razão $M_p/N = q^k/q^2 - 1$ é um limitante superior para o número de palavras que podem ser selecionadas para o código CP. O procedimento utilizado na Construção 4.7 seleciona $M_c = q^{k-2}$ e este valor é menor que o limitante superior por um fator de $(q^2 - 1)/q^2$ que tende ao valor 1 à medida que o valor de q aumenta.

Uma das construções de códigos CP propostas em [26] utiliza códigos Reed-Solomon. As palavras do código CP pertencem a um código binário cíclico de comprimento $N = q(q - 1)$ com $M_p = q^k$ palavras-código. O procedimento utilizado em [26] seleciona $M_c = q^{k-2}$ palavras para o código CP. Logo, o limitante superior é dado por $M_p/N = q^{k-1}/(q - 1)$ e o total de palavras $M_c = q^{k-2}$ difere desse limitante por um fator de $(q - 1)/q$. Comparando-se este fator com o fator $(q^2 - 1)/q^2$, percebe-se que para qualquer valor de q, $(q^2 - 1)/q^2 > (q-1)/q$. Portanto, a Construção 4.7 é mais eficiente do que a Construção baseada em códigos RS proposta em [26]. De modo semelhante, é facilmente demonstrável que a Construção 4.7 é mais eficiente do que a outra construção de códigos CP, baseada em códigos Berlekamp-Justesen, também proposta em [26]. Portanto, esta é uma contribuição significativa desta dissertação.

Os códigos CP da Construção 4.7 não são de peso constante. Embora seja suficiente escolher uma representação-$\mathbf{V}$ com $w(\mathbf{v}) = w(\mathbf{v}')$ para obter códigos CP de peso constante por meio da Construção 4.7, essa escolha não é boa para aplicar os códigos CP como sequências de protocolo. Para construir códigos CP de peso constante adequados para aplicação como sequência de protocolo, utiliza-se um subconjunto de palavras de um código constacíclico com peso constante tal que elas não possuam nenhuma coordenada nula. As duas construções propostas a seguir contemplam as condições propostas.

Construção 4.8

Seja q um número primo, $q > 3$, $n = q + 1$ e k um número par tal que $4 \leq k \leq q - 1$. Escolha a representação-$\mathbf{V}^*$ com $\mathbf{v}^* = (1, 0, 0, \ldots, 0)$, e um código $\mathcal{C}$ constacíclico linear q-ário $(q + 1, k, q - k + 2)$ (MDS) construído de acordo com o Teorema 3.12. Divida o subconjunto de palavras de $\mathcal{C}$ com peso $w = q + 1$ em classes de equivalência constacíclica e defina uma palavra-código para ser a líder de cada uma destas classes. Aplicando o Teorema 4.2 a cada palavra líder de uma classe de equivalência constacíclica, obtém-se um $CCP(N, M_c, d_c)$ com $N = q^2 - 1$, $M_c = A_{q+1}/N$, em que A_{q+1} é dado pela Fórmula (3.17), e com distância mínima cíclica igual a $d_c = 2(q - k + 2)$. □

Observe que na Construção 4.8, o código constacíclico deve ser MDS e todas suas palavras-código devem ter ordem constacíclica plena. Tal construção é possível escolhendo adequadamente o polinômio gerador $g(x)$, mesmo não tendo a garantia de que todas as raízes de $x^{q+1} - a$ sejam primitivas como ocorre quando q é um primo de Mersenne. O código constacíclico do Exemplo 3.8 mostra que é possível esse tipo de construção. Com relação aos primos de Mersenne, a construção a seguir, proposta em [36], é um caso particular da Construção 4.8 quando q é um primo de Mersenne.

Construção 4.9

Seja $q = 2^m - 1$ um primo de Mersenne, $n = q + 1$ e k um número par tal que $4 \leq k \leq q - 1$. Escolha a representação-$\mathbf{V}^*$ com $\mathbf{v}^* = (1, 0, 0, \ldots, 0)$, e um código $\mathcal{C}$ constacíclico linear q-ário $(q+1, k, q-k+2)$ (MDS). Divida o subconjunto de palavras de $\mathcal{C}$ com peso $w = q+1$ em classes de equivalência constacíclica e defina uma palavra-código para ser a líder de cada uma destas classes. Aplicando o Teorema 4.2 a cada palavra líder de uma classe de equivalência constacíclica, obtém-se um $CCP(N, M_c, d_c)$ com $N = q^2 - 1$, $M_c = A_{q+1}/N$, em que A_{q+1} é dado pela Fórmula (3.17), e com distância mínima cíclica igual a $d_c = 2(q - k + 2)$. □

Capítulo 5

Sequências de Protocolo

Ninguém é tão sábio que nada tenha para aprender, nem tão tolo que nada tenha para ensinar.

— Blaise Pascal

Neste capítulo, o objetivo é mostrar a aplicação de códigos ciclicamente permutáveis (códigos CP) como sequências de protocolo para o canal de colisão sem realimentação (Capítulo 2). O desempenho das sequências de protocolo propostas é analisado e é feita uma comparação com as sequências de protocolo, também construídas por meio de códigos CP, propostas em [26], [29].

5.1 Sequências de Protocolo para o Canal de Colisão sem Realimentação

A, Györfi e Massey [26] mostraram que *códigos ciclicamente permutáveis* [27] constituem uma solução natural para o caso particular de acesso múltiplo em que M usuários, de um total de U, estão ativos em um dado quadro. Nesta situação, cada usuário recebe uma palavra do código CP e a utiliza como sequência de protocolo para controlar suas transmissões. Desta forma, as palavras do código CP constituem um conjunto (U, M, N, σ) de sequências de pro-

tocolo, em que U denota o total de usuários que compartilham o canal, M denota o número de usuários ativos por quadro, cujo comprimento é denotado por N, e σ denota o número total de pacotes que os usuários podem emitir, por quadro, com a garantia de que são recebidos livres de colisão.

Uma outra abordagem, complementar ao trabalho de A, Györfi e Massey [26], é utilizar códigos ciclicamente permutáveis de peso não-constante como sequências de protocolo. Nesta situação, usuários distintos podem ter sequências de protocolo com diferentes fatores de trabalho. Lembrando que no Capítulo 2, o fator de trabalho p_i do usuário i, $1 \leq i \leq U$, é definido como a fração de tempo em que a sua sequência de protocolo assume o valor 1. Então para sequências de protocolo provenientes das palavras de um código CP de comprimento N, pode-se, alternativamente, definir o fator de trabalho p_i como sendo a razão entre o peso w_i da palavra-código, correspondente à sequência de protocolo, e o comprimento N das palavras-código, logo $p_i = w_i/N$. Obviamente, se o código CP é de peso constante, então todos os usuários possuem fator de trabalho dado por $p = w/N$.

O Teorema 5.1, enunciado na sequência, estabelece o principal resultado para o cálculo de M e σ quando as palavras de um código CP, de peso constante ou não, são usadas como sequências de protocolo. Na prova do Teorema 5.1, faz-se uso da propriedade de *correlação* que é definida como o número de coordenadas em que duas N-uplas binárias coincidentemente possuem valor 1.

Lema 5.1

Em um código ciclicamente permutável de peso não-constante, $CCP(N, M_c, d_c)$, a ***correlação****, denotada por ρ, entre qualquer palavra-código e seus deslocamentos cíclicos ou entre quaisquer deslocamentos cíclicos de duas palavras-código distintas satisfaz*

$$\rho \leq w_{\max} - d_c/2, \tag{5.1}$$

em que $w_{\max}$ denota o maior peso de Hamming dentre as palavra do código. □

Demonstração: Para duas N-uplas binárias quaisquer que possuam pesos w_i e w_j e cuja distância de Hamming entre elas seja d, o número de 1's em que elas coincidem é exatamente $(w_i + w_j)/2 - d/2$. Sendo as N-uplas binárias palavras de um código CP, sendo $w_{\max}$ o maior peso de Hamming dentre as palavras-código e sendo d_c a distância mínima cíclica, então o valor máximo para correlação entre duas palavras do código é obtido quando ambas possuem peso $w_{\max}$. Assim, $\rho = w_{\max} - d_c/2$. Portanto, a correlação entre

quaisquer deslocamentos cíclicos de duas palavras-código distintas satisfaz $\rho \leq w_{\max} - d_c/2$. ■

Teorema 5.1

Seja $CCP(N, M_c = U, d_c)$ um código ciclicamente permutável de peso não-constante cujas palavras-código possuem peso mínimo e peso máximo denotados, respectivamente, por $w_{\min}$ e $w_{\max}$. Para um número inteiro σ, $1 \leq \sigma \leq w_{\max}$, um código ciclicamente permutável $CCP(N, M_c = U, d_c)$ é um conjunto de sequências de protocolo, representadas por (U, M, N, σ), para

$$M \geq \min\left\{U, \left\lfloor \frac{w_{\min} - 1}{w_{\max} - d_c/2} \right\rfloor, \left\lfloor \frac{w_{\min} - \sigma}{w_{\max} - d_c/2} \right\rfloor + 1 \right\}. \tag{5.2}$$

□

Demonstração: Inicialmente, considere a estratégia pela qual o receptor é capaz de identificar os usuários cujos pacotes foram recebidos com sucesso. Para isto, considere o conjunto $\mathcal{W}$ cujos elementos são os pesos das sequências de protocolo dos U usuários do canal e considere um quadro arbitrário de comprimento N que é processado pelo receptor em um instante de tempo também arbitrário. Seja $\tau = [\tau_1, \tau_2, \ldots, \tau_N]$ a N-upla binária que representa o *vetor atividade de transmissão*, em que τ_j, $1 \leq j \leq N$, assume valores 0 ou 1 correspondendo, respectivamente, ao símbolo de silêncio ou outro símbolo qualquer (pacote ou colisão) recebido no j-ésimo intervalo de tempo desse quadro. O receptor decide se o usuário i está ativo no quadro recebido se e somente se os valores de j para os quais $\tau_j = 1$, $1 \leq j \leq N$, coincidem com os valores de l para os quais $s_{il} = 1$, $1 \leq l \leq N$, em que $\mathbf{s}_i = \{s_{il}\}_{l=1}^{N}$, $1 \geq i \geq U$, denota a sequência de protocolo do usuário i. Se o usuário i, de fato, está ativo no quadro, então a regra de decisão descrita está sempre correta. No entanto, se o usuário i não está ativo no quadro, então a regra de decisão utilizada falha. Para deduzir uma condição suficiente assegurando que o usuário i não está ativo em um quadro arbitrário, considere um número M de usuários ativos cujas sequências de protocolo possuem peso qualquer, não necessariamente iguais, pertencente a $\mathcal{W}$ e considere, ainda, que o usuário i possui sequência de protocolo cujo peso é $w_i \in \mathcal{W}$. Portanto, se o usuário i não está ativo no quadro, se M usuários estão ativos, no máximo, e se ρ denota o número máximo de 1's em que as sequências de protocolo dos M usuários cobrem, uma por vez, os 1's em $\mathbf{s}_i$, então $M\rho < w_{\min}$ é uma condição suficiente para identificar corretamente que o usuário i não está ativo, qualquer que seja o $w_i \in \mathcal{W}$. Mas, $\rho \leq w_{\max} - d_c/2$ de acordo com o Lema 5.1 e pelo fato de que a sequência de protocolo

de cada usuário pode estar deslocada ciclicamente. Assim, $M(w_{\max} - d_c/2) < w_{\min}$ ou, equivalentemente, $M(w_{\max} - d_c/2) \leq w_{\min} - 1$ é uma condição suficiente para identificar corretamente os usuários ativos por quadro. Logo, o número de usuários ativos M que satisfaz essa condição é dado por

$$M = \left\lfloor \frac{w_{\min} - 1}{w_{\max} - d_c/2} \right\rfloor . \tag{5.3}$$

Dando continuidade a demonstração, o objetivo é mostrar uma condição suficiente para que cada um dos M usuários ativos, por quadro, possa enviar, no mínimo, σ pacotes que são recebidos livres de colisão. Para isto, suponha que o usuário i está ativo. Como os pacotes dos demais $M-1$ usuários ativos podem colidir com, no máximo, $w_{\max} - d_c/2$ pacotes enviados pelo usuário i, então o usuário i tem a garantia de que $w_{\min} - (M-1)(w_{\max} - d_c/2)$ dos seus pacotes chegam ao receptor sem sofrer colisão qualquer que seja o peso $w_i \in \mathcal{W}$ da sequência de protocolo do usuário i. Logo, $\sigma \geq w_{\min} - (M-1)(w_{\max} - d_c/2)$ ou, equivalentemente,

$$M = \left\lfloor \frac{w_{\min} - \sigma}{w_{\max} - d_c/2} \right\rfloor + 1. \tag{5.4}$$

Por fim, é trivial que a condição $M \leq U$ seja satisfeita e, portanto, se o valor de M é o mínimo entre U e os valores inteiros dados pelas Fórmulas (5.3) e (5.4), então o receptor é capaz de identificar corretamente os usuários ativos por quadro e cada um deles tem a garantia de poder enviar σ pacotes que são recebidos livres de colisão. Porém, as Fórmulas (5.3) e (5.4) foram deduzidas considerando o pior caso, pois é possível situações em que só há usuários ativos que possuem sequências de protocolo com peso $w_{\min}$ e, então, o número de usuários ativos é maior que o valor calculado em (5.3) e (5.4). Logo, justifica-se a desigualdade em (5.2) e a condição de igualdade ocorre quando o código CP é de peso constante. ■

Obviamente, se o código CP for de peso constante, então $w_{\min} = w_{\max} = w$ e o Teorema 5.1 reduz-se ao resultado proposto pelo Teorema 4 em [26] para códigos CP de peso constante. Portanto, o Teorema 5.1 é uma generalização do Teorema 4 em [26], sendo esta uma contribuição importante desta dissertação.

Pelo Teorema 5.1, as construções de códigos CP propostas no Capítulo 4 podem ser aplicadas como sequências de protocolo para o canal de colisão sem realimentação apresentado no Capítulo 2. Os códigos CP da Construção 4.7 são adequados para situações em que os

usuários possuem diferentes fatores de trabalho, pois os códigos não são de peso constante. Já os códigos CP das Construções 4.8 e 4.9 são de peso constante e são adequados para situações em que os usuários possuem o mesmo fator de trabalho.

5.1.1 Usuários com Diferentes Fatores de Trabalho

Conforme dito anteriormente, os códigos CP da Construção 4.7 não são de peso-constante, logo as suas palavras-código são adequadas para o uso como sequências de protocolo numa situação em que os usuários possuem diferentes fatores de trabalho. Lembrando que os fatores de trabalho são calculados por $p_i = w_i/N,\ 1 \leq i \leq U$, em que w_i é o peso da palavra-código escolhida como sequência de protocolo e N é o comprimento do código CP, então os fatores de trabalho dependem do peso $w(\mathbf{v}')$ da $(q-1)$-upla binária $\mathbf{v}'$. Assim, variando o valor de $w(\mathbf{v}')$ pode-se ajustar o valor dos fatores de trabalho p_i. No entanto, deve-se perceber que à medida que o valor de $w(\mathbf{v}')$ aumenta, o valor do peso máximo $w_{\max}$ na Desigualdade (5.2) também aumenta, mas a distância mínima cíclica d_c permanece constante. Logo, o limitante inferior para o número de usuários ativos por quadro diminui e, portanto, deve-se ter cuidado ao definir o valor para $w(\mathbf{v}')$.

Do ponto de vista prático, aumentar o valor de $w(\mathbf{v}')$ implica aumentar o número de intervalos de tempo em que os usuários transmitem pacotes num quadro. Como se trata de um canal de múltiplo acesso, se um número considerável de usuários possuem altas taxas de transmissão, o número de colisões aumenta e o número de usuários ativos por quadro deve diminuir, concordando com o resultado previsto pela Desigualdade (5.2).

Exemplo 5.1

Para $q = 41$ *e* $k = 4$, *o código constacíclico* $(42, 4, 39)$ *satisfaz as condições da Construção 4.7. Sendo* $\mathbf{v} = (1, 0, 0, \ldots, 0)$ *e escolhendo* $\mathbf{v}' = (1, 1, 1, 0, 0, \ldots, 0)$, *ambos com* 40 *coordenadas,* $w(\mathbf{v}') = 3$, *obtém-se um* $CCP(1680, 1681, 78)$. *Como o código constacíclico possui palavras-código de peso* 42, 41, 40 *e* 39, *os possíveis pesos das palavras do código CP são* 42, 44, 46 *e* 48. *Logo, as sequências de protocolo obtidas possuem fatores de trabalho* $\frac{42}{1680}$, $\frac{44}{1680}$, $\frac{46}{1680}$ *e* $\frac{48}{1680}$. *Pela Desigualdade (5.2), com* $w_{\min} = 42$, $w_{\max} = 48$ *e* $\sigma = 24$, *obtém-se* $M \geq 3$. *Isto é, o conjunto de sequências de protocolo* $(1681, 3, 1680, 24)$ *permite que, no mínimo,* 3 *usuários, com quaisquer dos quatro fatores de trabalho permitidos, possam estar ativos num mesmo quadro emitindo* 24 *pacotes que são recebidos livres de colisão. É valido observar que para* $\sigma = 15$ *é possível atingir* $M \geq 4$, *produzindo um conjunto de sequências de protocolo* $(1681, 4, 1680, 15)$. *A escolha de*

utilizar um dos dois conjuntos de sequências de protocolo depende da aplicação, pois em algumas o número de usuários ativos pode ser prioridade, enquanto que em outras um número maior de pacotes transmitidos pode ser mais importante. □

Conforme mencionado no final da demonstração do Teorema 5.1, é possível calcular o número máximo de usuários por quadro, considerando o caso quando só usuários com fatores de trabalho igual $p = w_{\min}/N$ estão ativos. Assim, o número máximo de usuários ativos, M, pode ser calculado por meio da Desigualdade (5.2) substituindo $w_{\max}$ por $w_{\min}$. Para o Exemplo 5.1, o valor máximo é $M = 13$ com $\sigma = 6$, portanto o número de usuários ativos por quadro varia no intervalo $3 \leq M \leq 13$, enquanto que o número de pacotes recebidos livres de colisão varia entre $6 \leq \sigma \leq 24$.

5.1.2 Usuários com o mesmo Fator de Trabalho

Os exemplos a seguir ilustram o uso dos códigos CP das Construções 4.8 e 4.9 quando aplicados como sequências de protocolo para o canal de colisão sem realimentação.

Exemplo 5.2

Para $q = 13$ e $k = 4$, o código constacíclico $(14, 4, 11)$ com polinômio gerador $g(x) = M_1(x)M_{25}(x)M_{37}(x)M_{49}(x)M_{61}(x)$ (Tabela 5.1) satisfaz as condições da Construção 4.8. Pela Fórmula (3.17), $A_{14} = 8736$, logo $M_c = 8736/168 = 52$ e, portanto, obtém-se um $CCP(168, 52, 22)$ de peso constante $w = 14$. Utilizando a Desigualdade (5.2) do Teorema 5.1 para $\sigma = 8$, o valor obtido para M é dado por

$$\begin{aligned} M &= \lfloor (w-1)/(w-d_c/2) \rfloor = \lfloor (14-1)/(14-11) \rfloor \\ &= \lfloor 13/3 \rfloor = 4 \quad \text{ou,} \\ M &= \lfloor (w-\sigma)/(w-d_c/2) \rfloor + 1 = \lfloor (14-8)/(14-11) \rfloor + 1 \\ &= \lfloor 6/3 \rfloor + 1 = 3. \end{aligned}$$

Finalmente o conjunto de sequências de protocolo possui parâmetros $(U, M, N, \sigma) = (52, 3, 168, 8)$. Em outras palavras, desde que, no máximo, $M = 3$ usuários, de um total de $U = 52$ estejam ativos em cada quadro recebido, é garantido para estes usuários ativos enviarem 8 pacotes livres de colisão de um total de 14 pacotes enviados em um quadro com $N = 168$ intervalos de tempo. □

Tabela 5.1: *Classes conjugadas e polinômios mínimos sobre* $GF(13)$ *para* $x^{14}-2$. *Considere* $GF(13^2)$ *gerado por* $p(x)=2+x+x^2$.

Classe conjugada	Ordem multiplicativa	Polinômio mínimo
$\{1,13\}$	168	$M_1(x)=2+x+x^2$
$\{25,157\}$	168	$M_{25}(x)=2+4x+x^2$
$\{37,145\}$	168	$M_{37}(x)=2+6x+x^2$
$\{49,133\}$	24	$M_{49}(x)=2+x^2$
$\{61,121\}$	168	$M_{61}(x)=2+7x+x^2$
$\{73,109\}$	168	$M_{73}(x)=2+9x+x^2$
$\{85,97\}$	168	$M_{85}(x)=2+12x+x^2$

Exemplo 5.3

Para $q=31$ *e* $k=4$, *o código constacíclico* $(32,4,29)$ *satisfaz as condições da Construção 4.9. Pela Fórmula (3.17),* $A_{32}=297600$, *logo* $M_c=297600/960=310$ *e, portanto, obtém-se um* $CCP(960,310,58)$ *de peso constante* $w=32$. *Utilizando a Desigualdade (5.2) do Teorema 5.1 para* $\sigma=17$, *o valor obtido para* M *é dado por*

$$\begin{aligned} M &= \lfloor (w-1)/(w-d_c/2)\rfloor = \lfloor (32-1)/(32-29)\rfloor \\ &= \lfloor 31/3\rfloor = 10 \quad \text{ou,} \\ M &= \lfloor (w-\sigma)/(w-d_c/2)\rfloor + 1 = \lfloor (32-17)/(32-29)\rfloor + 1 \\ &= \lfloor 15/3\rfloor + 1 = 6. \end{aligned}$$

Finalmente o conjunto de sequências de protocolo possui parâmetros $(U,M,N,\sigma)=(310,6,960,17)$. *Em outras palavras, desde que, no máximo,* $M=6$ *usuários, de um total de* $U=310$ *estejam ativos em cada quadro recebido, é garantido para estes usuários ativos enviarem* 17 *pacotes livres de colisão de um total de* 32 *pacotes enviados em um quadro com* $N=960$ *intervalos de tempo.* □

5.2 Desempenho das Sequências de Protocolo

De acordo com [28], avaliar o desempenho de sequências de protocolo não é simples e o resultado depende, em geral, da natureza da aplicação pretendida. No entanto, os seguintes critérios são comumente considerados.

a. O número de usuários, M, que podem estar ativos por quadro;

b. A taxa total de transmissão (R_{sum}) dada por

$$R_{\text{sum}} = \frac{M\sigma}{N}, \tag{5.5}$$

em que M é o número de usuários ativos por quadro, σ é o número de pacotes transmitidos que são recebidos livres de colisão e N é o comprimento do quadro;

c. O número máximo de sequências distintas que podem ser geradas;

d. O comprimento do quadro, N, utilizado pelos usuários (o qual coincide com o comprimento das sequências de protocolo);

e. Suporte a usuários com diferentes fatores de trabalho. Isso é desejável, pois diferentes sensores (redes de sensores) ou estações de trabalho (redes *ad hoc*), podem necessitar de diferentes taxas de transmissão;

f. Uso de cabeçalhos para resolver, por exemplo, o problema de identificar os usuários cujos pacotes são recebidos corretamente.

O desempenho das sequências de protocolo propostas nesta dissertação, obtidas por meio das Construções 4.7, 4.8 e 4.9, é analisado com base nos critérios citados anteriormente. O desempenho também é analisado fazendo uma comparação com as sequências de protocolo propostas em [26], [29], que também são obtidas por meio da construção de códigos CP. Um dos conjuntos de sequências de protocolo propostos por [26] são baseados na construção de códigos CP por meio de códigos Reed-Solomon (RS), portanto, daqui em diante, refere-se a essas sequências como *Sequências-RS*. As sequências de protocolo propostas por [29] são baseadas na construção de códigos CP por meio de códigos BCH e, assim, refere-se a elas como *Sequências-BCH*. Vale destacar que as Sequências-RS são um caso particular das Sequências-BCH, conforme foi demonstrado em [29].

A seguir, são apresentados os parâmetros (U, M, N, σ) para as Sequências-BCH, Sequências-RS e para as sequências de protocolo baseadas nas Construções 4.7, 4.8 e 4.9. Também são mostrados, para cada uma das sequências mencionadas, um limitante superior para o valor de M e um limitante inferior para o valor de R_{sum}. Depois, é feita uma comparação entre as sequências utilizando os critérios apresentados no início desta seção.

5.2.1 Sequências-BCH e Sequências-RS

De acordo com [29], os parâmetros (U, M, N, σ) das sequências-BCH são

$$U = q^{(k-2)r}, \tag{5.6}$$

$$N = q(q^r - 1), \tag{5.7}$$

$$M = \min\left\{U, \left\lfloor \frac{w-1}{w-d_c/2} \right\rfloor, \left\lfloor \frac{w-\sigma}{w-d_c/2} \right\rfloor + 1\right\}, \tag{5.8}$$

em que $q \geq 5$, $1 \leq r$ e $3 \leq k \leq q-1$. As sequências de protocolo são palavras de um código CP de peso constante com $w = q^r - 1$ e $d_c \geq 2(q^r - 1 - (k-1)q^{r-1})$. Para $r = 1$, as Sequências-BCH equivalem às Sequências-RS, considerando o comprimento máximo, $q-1$, para os códigos Reed-Solomon utilizados em [26].

De acordo com [28], o limitante superior para o valor de M pode ser deduzido considerando o número máximo de usuários ativos que podem transmitir, no mínimo, um pacote que seja recebido livre de colisão ($\sigma = 1$) em um quadro de comprimento N. Assim, para $\sigma = 1$, o segundo termo, do lado direito, na Fórmula (5.8) é o menor, logo

$$\begin{aligned} M &= \left\lfloor \frac{w-1}{w-d_c/2} \right\rfloor \\ &\geq \left\lfloor \frac{(q^r-1)-1}{(q^r-1)-[q^r-1-(k-1)q^{r-1}]} \right\rfloor \\ &= \left\lfloor \frac{q^r-2}{(k-1)q^{r-1}} \right\rfloor. \end{aligned}$$

Para valores elevados de q, esse limitante inferior para o valor de M é aproximadamente igual a $\lceil q/(k-1) \rceil$. Se este resultado é utilizado para avaliar o número de usuários ativos, então, no máximo, $\lceil q/2 \rceil$ podem estar ativos, uma vez que $3 \leq k \leq q-1$.

De acordo com o desenvolvimento realizado em [29], para deduzir o limitante inferior para o valor de R_{sum}, o primeiro passo é avaliar para quais valores de σ, no lado direito da Fórmula (5.8), o terceiro termo é o menor. Para isto, o segundo e o terceiro termos do lado direito de (5.8) podem ser reescritos, respectivamente, do seguinte modo

$$\begin{aligned} m_1 &\leq \frac{w-1}{w-d_c/2} \quad \text{e} \\ m_2 &\leq \frac{w-\sigma}{w-d_c/2} + 1 \\ &= \frac{2w-\sigma-d_c/2}{w-d_c/2}, \end{aligned}$$

em que m_1 e m_2 denotam o segundo e o terceiro termos em questão. Para que $m_2 \leq m_1$, é suficiente que $(2w - \sigma - d_c/2) \leq (w-1)$, o que implica em $\sigma \geq w + 1 - d_c/2$. Logo, para

$\sigma \geq (k-1)q^{r-1}+1$, o terceiro termo é o menor, assim

$$M \geq \left\lfloor \frac{w-\sigma}{(k-1)q^{r-1}} \right\rfloor + 1 > \frac{w-\sigma}{(k-1)q^{r-1}},$$

o qual, quando substituído na Fórmula (5.5), produz

$$R_{\text{sum}} \geq \frac{\sigma(w-\sigma)}{N(k-1)q^{r-1}}, \tag{5.9}$$

cujo valor é máximo para $\sigma = w/2$, desde que $(w/2) \geq (k-1)q^{r-1}+1$. Logo, sendo $N = q(q^r-1) = qw$,

$$\begin{aligned} R_{\text{sum}} &\geq \frac{(w/2)(w-w/2)}{qw(k-1)q^{r-1}} \\ &= \frac{w^2}{4qw(k-1)q^{r-1}} \\ &= \frac{w}{4(k-1)q^r} \\ &= \frac{q^r-1}{4(k-1)q^r}, \end{aligned}$$

que para valores elevados de q pode ser aproximado para

$$R_{\text{sum}} \approx \frac{1}{4(k-1)}, \tag{5.10}$$

sendo este um limitante inferior para R_{sum}.

5.2.2 Sequências Baseadas na Construção 4.7

De acordo com a Construção 4.7, os parâmetros (U, M, N, σ) das sequências baseadas nessa construção são

$$U = q^{k-2}, \tag{5.11}$$

$$N = q^2-1, \tag{5.12}$$

$$M \geq \min\left\{U, \left\lfloor \frac{w_{\min}-1}{w_{\max}-d_c/2} \right\rfloor, \left\lfloor \frac{w_{\min}-\sigma}{w_{\max}-d_c/2} \right\rfloor + 1\right\}, \tag{5.13}$$

em que $q \geq 5$ e $4 \leq k \leq q-1$. As sequências de protocolo são palavras de um código CP, de peso não constante, com $w_{\min} = q+1$, $w_{\max} = (q-k+2)+(k-1)w(\mathbf{v}')$ e $d_c \geq (q-k+2)d(\mathbf{v})$, em que $w(\mathbf{v}')$, $w(\mathbf{v}') \geq 3$, denota o peso da $(q-1)$-upla que representa o elemento 0 na representação-**V** cuja distância mínima é denotada por $d(\mathbf{v})$.

Para se obter o limitante superior para o valor de M, segue-se o mesmo raciocínio aplicado anteriormente para as Sequências-BCH com a hipótese adicional de que todos os usuários

ativos, num determinado quadro, possuam sequências de protocolo que correspondem a palavras do código CP com peso mínimo ($w_{\min}$). Tal hipótese corresponde a substituir $w_{\max}$ por $w_{\min}$ na Desigualdade (5.13) que, nesse caso, é satisfeita com igualdade. Além do mais, para palavras do código CP com peso igual a $w_{\min}$, $d(\mathbf{v}) = 2$. Assim, para $\sigma = 1$, $w_{\max} = w_{\min}$ e $d(\mathbf{v}) = 2$, obtém-se

$$\begin{aligned} M &= \left\lfloor \frac{w_{\min} - 1}{w_{\min} - d_c/2} \right\rfloor \\ &\geq \left\lfloor \frac{(q+1) - 1}{(q+1) - (q-k+2)} \right\rfloor \\ &= \left\lfloor \frac{q}{(k-1)} \right\rfloor . \end{aligned}$$

Assim, se este resultado é utilizado para avaliar o número de usuários ativos, então, no máximo, $\lfloor q/3 \rfloor$ podem estar ativos, uma vez que $4 \leq k \leq q-1$.

Para deduzir o limitante inferior para o valor de R_{sum}, o primeiro passo é avaliar para quais valores de σ, no lado direito da Desigualdade (5.13), o terceiro termo é o menor. Seguindo o mesmo raciocínio utilizado para as Sequências-BCH, obtém-se $\sigma \geq w_{\max} + 1 - d_c/2$. Como $d(\mathbf{v}) \geq 2$ para $w(\mathbf{v}') \geq 3$, $\sigma \geq (k-1)w(\mathbf{v}') + 1$. Logo, para $\sigma \geq (k-1)w(\mathbf{v}') + 1$ o terceiro termo é o menor, assim

$$M \geq \left\lfloor \frac{w_{\min} - \sigma}{(k-1)w(\mathbf{v}')} \right\rfloor + 1 > \frac{w_{\min} - \sigma}{(k-1)w(\mathbf{v}')},$$

o qual, quando substituído na Fórmula (5.5), produz

$$R_{\text{sum}} \geq \frac{\sigma(w_{\min} - \sigma)}{N(k-1)w(\mathbf{v}')}, \tag{5.14}$$

cujo valor é máximo para $\sigma = w_{\min}/2$, desde que $(w_{\min}/2) \geq (k-1)w(\mathbf{v}') + 1$. Logo, sendo $N = q^2 - 1 = (q+1)(q-1) = w_{\min}(q-1)$,

$$\begin{aligned} R_{\text{sum}} &\geq \frac{(w_{\min}/2)(w_{\min} - w_{\min}/2)}{w_{\min}(q-1)(k-1)w(\mathbf{v}')} \\ &= \frac{w_{\min}^2}{4w_{\min}(q-1)(k-1)w(\mathbf{v}')} \\ &= \frac{w_{\min}}{4(q-1)(k-1)w(\mathbf{v}')} \\ &= \frac{(q+1)}{4(q-1)(k-1)w(\mathbf{v}')}, \end{aligned}$$

que para valores elevados de q pode ser aproximado para

$$R_{\text{sum}} \approx \frac{1}{4(k-1)w(\mathbf{v}')}, \tag{5.15}$$

sendo este um limitante inferior para R_{sum}.

5.2.3 Sequências Baseadas nas Construções 4.8 e 4.9

De acordo com as Construções 4.8 e 4.9, os parâmetros (U, M, N, σ) das sequências baseadas nessas construções são

$$U = A_{q+1}/N, \tag{5.16}$$

$$N = q^2 - 1, \tag{5.17}$$

$$M = \min\left\{U, \left\lfloor \frac{w-1}{w-d_c/2} \right\rfloor, \left\lfloor \frac{w-\sigma}{w-d_c/2} \right\rfloor + 1 \right\}, \tag{5.18}$$

em que $q \geq 5$, $4 \leq k \leq q-1$ e o valor de A_{q+1} é dado pela Fórmula (3.17). As sequências de protocolo são palavras de um código CP de peso constante com $w = q+1$ e $d_c = 2(q-k+2)$.

O limitante superior para o valor de M é o mesmo deduzido para as sequências baseadas na Construção 4.7, pois a hipótese assumida, naquele ponto, de que todos os usuários ativos possuem sequências de protocolo que são as palavras do código CP com peso mínimo ($w_{\min}$), corresponde às sequências baseadas nas Construções 4.8 e 4.9. Logo, $M \leq \lfloor q/3 \rfloor$.

A dedução para o limitante inferior para o valor de R_{sum} segue o raciocínio já discutido anteriormente para as outras sequências. Dessa forma, $\sigma \geq w + 1 - d_c/2$. Logo, para $\sigma \geq k$, o terceiro termo, do lado direito, na Fórmula (5.18) é o menor, assim

$$M \geq \left\lfloor \frac{w-\sigma}{(k-1)} \right\rfloor + 1 > \frac{w-\sigma}{k-1},$$

o qual, quando substituído na Fórmula (5.5), produz

$$R_{\text{sum}} \geq \frac{\sigma(w-\sigma)}{N(k-1)}, \tag{5.19}$$

cujo valor é máximo para $\sigma = w/2$, desde que $(w/2) \geq k$. Logo, sendo $N = q^2 - 1 = (q+1)(q-1) = w(q-1)$,

$$\begin{aligned} R_{\text{sum}} &\geq \frac{(w/2)(w-w/2)}{w(q-1)(k-1)} \\ &= \frac{w^2}{4w(q-1)(k-1)} \\ &= \frac{w}{4(q-1)(k-1)} \\ &= \frac{(q+1)}{4(q-1)(k-1)}, \end{aligned}$$

que para valores elevados de q pode ser aproximado para

$$R_{\text{sum}} \approx \frac{1}{4(k-1)}, \tag{5.20}$$

sendo este um limitante inferior para R_{sum}.

5.2.4 Análise do Desempenho das Sequências

Neste ponto, realiza-se uma comparação entre as sequências de protocolo apresentadas segundo os critérios vistos no início desta seção. O objetivo é comparar o desempenho das sequências propostas nesta dissertação, sequências baseadas nas Construções 4.7, 4.8 e 4.9, com as Sequências-RS e Sequências-BCH, propostas em [26], [29], respectivamente. A Tabela 5.2 apresenta os critérios mencionados no início desta seção e que são utilizados para avaliar o desempenho das sequências.

Inicialmente, como todas as sequências de protocolo mencionadas são obtidas por meio de palavras de um código CP, isto implica que quando os pacotes chegam ao receptor num dado quadro de transmissão, ele é capaz de distinguir os usuários ativos sem a necessidade de cabeçalhos de identificação [28], [51]. Pois, a sequência de protocolo de cada usuário pode ser identificada mesmo que ela seja recebida com algum deslocamento cíclico, uma vez que a sequência resultante do deslocamento cíclico dela é diferente da sequência de protocolo dos outros usuários e também dos deslocamentos cíclicos de tais sequências. Este fato é consequência direta da definição de códigos CP (Definição 4.3).

Outra característica interessante é que as sequências de protocolo deem suporte a usuários com diferentes fatores de trabalho [28]. Como dito anteriormente, isso é desejável, pois diferentes sensores (redes de sensores) ou estações de trabalho (redes *ad hoc*), podem necessitar de usuários com diferentes taxas de transmissão. Dentre as sequências discutidas, só as sequências baseadas na Construção 4.7 apresentam essa característica, pois o código CP utilizado como base não é de peso constante, como ocorre com as demais sequências. Portanto, as sequências baseadas na Construção 4.7 são uma contribuição relevante desta dissertação.

Tabela 5.2: *Resumo dos critérios de desempenho para as sequências de protocolo. Para as sequências propostas nesta dissertação,* $q \geq 5$, $4 \leq k \leq q-1$ *e* $w(\mathbf{v}') \geq 3$. *Para as Sequências-RS e Sequências-BCH,* $q \geq 5$, $3 \leq k \leq q-1$ *e* $r > 1$.

Critérios	**Sequências**			
	Construção 4.7	Construções 4.8 e 4.9	RS	BCH
Limitante inferior para R_{sum}	$\frac{1}{4(k-1)w(\mathbf{v}')}$	$\frac{1}{4(k-1)}$	$\frac{1}{4(k-1)}$	$\frac{1}{4(k-1)}$
Limitante superior para M	$\lfloor q/3 \rfloor$	$\lfloor q/3 \rfloor$	$\lceil q/2 \rceil$	$\lceil q/2 \rceil$
Nº de sequências geradas (U)	q^{k-2}	$\frac{A_{q+1}}{N}$	q^{k-2}	$q^{(k-2)r}$
Comprimento do quadro (N)	q^2-1	q^2-1	q^2-q	$q(q^r-1)$
Diferentes fatores de trabalho	*sim*	*não*	*não*	*não*
Cabeçalhos de Identificação	*não*	*não*	*não*	*não*

Outro parâmetro importante, ao se analisar o desempenho das sequências de protocolo, é o comprimento N do quadro utilizado nas transmissões (o qual coincide com o comprimento das sequências). Segundo [29], quanto maior o comprimento do quadro, maior a complexidade de decodificação por intervalo de tempo, pois os algoritmos para decodificação de apagamento possuem complexidade proporcional a n^2 (lembrando que, conforme foi discutido após o Teorema 3.3 sobre correção de apagamentos, Capítulo 3, os pacotes emitidos pelos usuários são codificados utilizando um código corretor de erro de parâmetros $n = w$, $k = \sigma$ e $d_{\min} \geq w - \sigma + 1$). Portanto, com relação ao comprimento dos quadros, as sequências de protocolo propostas nesta dissertação possuem comprimento $N = q^2 - 1$, em que $q \geq 5$, que é aproximadamente o mesmo valor do comprimento das Sequências-RS, $N = q^2 - q$ para $q \geq 5$. Comparando com as Sequencias-BCH, cujo comprimento é $N = q(q^r - 1)$, em que $q \geq 5$ e $r > 1$, as sequências propostas possuem comprimento bem inferior, principalmente, à medida que o valor de r aumenta. Por exemplo, para $q = 13$, $k = 4$ e $r = 2$, o comprimento das Sequências-BCH é $N = 2184$, enquanto que para as sequências propostas nesta dissertação e para as Sequências-RS, considerando os mesmos valores de q e k, os comprimento são $N = 168$ e $N = 156$, respectivamente.

Com relação ao número máximo de sequências distintas que podem ser geradas, o desempenho das sequências propostas nesta dissertação é analisado. O número de sequências distintas que podem ser geradas é igual ao número total de usuários que compartilham o canal e que é denotado por U. Assim, as sequências baseadas na Construção 4.7 geram $U = q^{k-2}$, $q \geq 5$ e $4 \leq k \leq q - 1$, sequências distintas, enquanto que as sequências baseadas nas Construções 4.8 e 4.9 geram $U = A_{q+1}/N$ sequências distintas, em que N denota o comprimento do quadro e A_{q+1}, dado pela Fórmula (3.17), denota o número de palavras do código constacíclico com peso $w = q + 1$. Primeiramente, comparando o valor de U das sequências baseadas na Construção 4.7 com o valor de U das Sequências-RS e Sequências-BCH, conclui-se que ele é igual ao valor da primeira e inferior ao valor da segunda, $U = q^{(k-2)r}$ ($3 \leq k \leq q - 1$, $q \geq 5$ e $r > 1$), principalmente para valores elevados de r. Já o valor de U das sequências baseadas nas Construções 4.8 e 4.9, quando comparado com os valore de U das sequências baseadas na Construção 4.7, Sequências-RS e Sequências-BCH, é sempre inferior e deve-se ao fato de restringir-se a construção das sequências às palavras do código constacíclico de peso igual $w = q + 1$.

A respeito do limitante superior para o valor de M, as sequências de protocolo propostas

possuem o mesmo valor, $M \leq \lfloor q/3 \rfloor$, que é menor que o limitante inferior para as Sequências-RS e Sequências-BCH dado por $M \leq \lceil q/2 \rceil$. Porém, a diferença entre os valores dos limitantes é cada vez menor à medida que o valor de q aumenta.

Por fim, são comparados os valores encontrados para os limitantes inferiores de R_{sum}. De acordo com a Tabela 5.2, as sequências baseadas nas Construções 4.8 e 4.9 possuem o mesmo limitante inferior das Sequências-RS e das Sequências-BCH. Já as sequências baseadas na Construção 4.7 possuem um limitante inferior que é menor que o limitante inferior das demais sequências por um fator de $\frac{1}{w(\mathbf{v}')}$, $w(\mathbf{v}') \geq 3$. Esta diminuição é devida ao fato dos códigos CP da Construção 4.7 não serem de peso constante e o valor de $w(\mathbf{v}')$ influenciar diretamente no peso das palavras-código. Assim, como há usuários com fatores de trabalho maiores do que outros, o número de usuários ativos por quadro pode diminuir. Porém, como já foi analisado anteriormente, as sequências baseadas na Construção 4.7 são as únicas, entre as apresentadas, que oferecem suporte a usuários com diferentes fatores de trabalho.

Capítulo 6

Conclusões, Comentários e Sugestões

A mecânica quântica está a impor-se. Porém, uma voz interior diz-me que ainda não é a teoria certa... estou convencido de que Ele não joga dados.

— Albert Einstein

Neste capítulo, é sumarizado o trabalho descrito nesta dissertação. São feitos alguns comentários, é apresentada uma lista das contribuições e, finalmente, são propostos alguns tópicos para futuras investigações.

6.1 Resumo do conteúdo e Comentários Finais

O principal objetivo desta dissertação é propor uma maneira original de construir sequências de protocolo para o canal de colisão sem realimentação. O trabalho é desenvolvido baseado na idéia proposta por Györfi e Massey [26] de que códigos ciclicamente permutáveis constituem uma solução natural para construção de sequências de protocolo para este canal. Entretanto, a proposta apresentada nesta dissertação utiliza uma classe de códigos diferente da que foi utilizada em [26]. Utiliza-se a classe de códigos constacíclicos os quais ainda não haviam sido explorados neste contexto. Além do mais, um método generalizado para associar arranjos bidimensionais e N-uplas, proposto em [50], foi utilizado.

Inicialmente, apresenta-se um modelo de canal de acesso múltiplo introduzido por Massey e Mathys [24], o canal de colisão sem realimentação. São apresentados o *modelo básico*

do canal e as *restrições* ao uso deste canal, as quais, segundo [24], são necessárias para que o modelo do canal seja completamente especificado. Mostra-se que as sequências de protocolo são utilizadas para determinar quando o usuário é permitido, ou não, emitir seus pacotes pelo canal. As sequências de protocolo podem ser obtidas diretamente por meio de códigos ciclicamente permutáveis para serem utilizadas no caso particular em que o canal é compartilhado por um total de U usuários, mas só M destes usuários podem estar ativos em um determinado quadro.

Posteriormente, estuda-se os conceitos básicos da teoria de códigos corretores de erros. É feita uma revisão dos conceitos de código de bloco e depois, em sequência, códigos lineares e códigos cíclicos lineares. O objetivo é definir os conceitos básicos para abordar, enfim, a classe de códigos constacíclicos que são o alicerce para desenvolver o restante do trabalho. O estudo de códigos constacíclicos concentrou-se, basicamente, nos casos em que o comprimento das palavras-código é igual a $q+1$. Um resultado inédito, obtido nesta dissertação, permite escolher o polinômio gerador para os códigos constacíclicos de forma que todas as palavras-código não-nulas tenham ordem constacíclica plena (Teorema 3.12).

Definidos os fundamentos teóricos dos códigos corretores de erros, apresenta-se os códigos ciclicamente permutáveis (códigos CP) que passam a ser o objeto de estudo. Antes disso, porém, define-se uma representação cíclica para os elementos de $GF(q)$ por intermédio de $(q-1)$-uplas binárias e apresenta-se uma relação que permite um mapeamento um-a-um entre arranjos bidimensionais e N-uplas binárias. A ideia é usar os códigos constacíclicos q-ários, já bem estudados, juntamente com a representação cíclica para $GF(q)$ e a relação entre arranjos bidimensionais e N-uplas binárias, a fim de se construir códigos cíclicos binários não-lineares, que é o último passo até chegar aos códigos CP.

Após definir os códigos CP, são discutidos alguns conceitos definidos por A, Györfi e Massey [26] e apresentam-se as construções de códigos CP propostas nesta dissertação. Em seguida, mostra-se a utilização dos códigos CP propostos como sequências de protocolo para o caso especial do canal de colisão sem realimentação em que M usuários, de um total U usuários, estão ativos num determinado quadro. O desempenho das sequências de protocolo propostas é analisado e comparado com o desempenho de outras sequências de protocolo existentes. A Tabela 5.2 resume os critérios de desempenho avaliados e pode-se concluir que quando comparado ao desempenho das sequências de protocolo já existentes na literatura, o desempenho das sequências propostas é satisfatório.

6.2 Contribuições da Dissertação

As contribuições desta dissertação são listadas a seguir.

- Na Subseção 3.4.2, o Teorema 3.12 é inédito e mostra como se escolher o polinômio gerador de um código constacíclico de modo que todas as palavras-código possuam ordem constacíclica plena. Nesse mesmo contexto, o Lema 3.1 e o Corolário 3.1 são contribuições;
- As construções de códigos cíclicos binários não-lineares e códigos CP propostas em [26] utilizam códigos cíclicos e uma correspondência um-a-um entre arranjos bidimensionais e N-uplas que é garantida pelo teorema chinês do resto. No Capítulo 4, as construções de códigos cíclicos binários não-lineares e códigos CP utilizam a mesma ideia utilizada em [26]. Porém, o procedimento é estendido à classe de códigos constacíclicos e utiliza-se uma correspondência um-a-um entre arranjos bidimensionais e N-uplas, diferente daquela usada em [26], que foi proposta inicialmente em [50] e é apresentada de uma forma mais simples em [36];
- Na Seção 4.2, o Teorema 4.3 é original no que diz respeito ao seu uso com códigos constacíclicos. Até então, o seu uso havia sido restrito a códigos cíclicos;
- Conforme é demonstrado na Seção 4.2, a Construção 4.7 para códigos CP é mais eficiente do que as construções de códigos CP baseadas em códigos Reed-Solomon e Berlekamp-Justesen propostas em [26];
- Na Seção 5.1, o Lema 5.1 e o Teorema 5.1 complementam os resultados apresentados em [26] para o caso de códigos CP de peso não necessariamente constante;
- Na Subseção 5.1.1, as sequências de protocolo apresentadas, baseadas na Construção 4.7, dão suporte a usuários com diferente fatores de trabalho. Tal característica não é contemplada pelas sequências de protocolo existentes que são construídas por meio de códigos CP;
- Na Subseção 5.2.4, o desempenho satisfatório das sequências de protocolo propostas neste trabalho reforça a contribuição da nova gama de opções de sequências de protocolo para o canal de colisão sem realimentação.

6.3 Contribuições Futuras

A seguir, apresenta-se algumas sugestões para trabalhos futuros que possam ser realizados a partir dos resultados expostos nesta dissertação.

- Nesta disssertação, os resultados são demonstrados para códigos constacíclicos com comprimento de bloco igual a $q+1$. Em [48], mostra-se a existência de códigos constacíclicos para outros comprimentos além de $q+1$. Portanto, uma continuação natural e imediata é generalizar os resultados para quaisquer parâmetros possíveis de códigos constacíclicos;
- Investigar um modo facilmente implementável de selecionar todas as palavras de um código constacíclico que possuam ordem constacíclica plena. Pois, embora o método utilizado pelo Teorema 4.3 (Seção 4.2) seja facilmente implementável, para alguns códigos constacíclicos, ele não é capaz de selecionar todas as palavras-código que possuem ordem constacíclica plena;
- Comparar o desempenho das sequências de protocolo propostas com o desempenho de sequências de protocolo que não sejam necessariamente construídas por meio de códigos CP. Por exemplo, as sequências apresentadas em [28];
- Realizar simulações computacionais e experimentos para o modelo do canal de colisão sem realimentação a fim de comparar o desempenho teórico das sequências de protocolo com o desempenho simulado e com o desempenho prático.

REFERÊNCIAS

[1] B. P. Lathi, **Modern Digital and Analog Communication Systems**, 3ª ed. Oxford University Press, 1998.

[2] C. E. Shannon, A mathematical theory of communication, *Bell System Technical Journal*, v. 27, n. 3 and 4, p. 379–423 and 623–656, July and October 1948.

[3] ——, Communication theory of secrecy systems, *Bell System Technical Journal*, v. 28, n. 4, p. 656–715, October 1949.

[4] R. W. Hamming, Error detecting and error correcting codes, *Bell System Technical Journal*, v. 29, n. 2, p. 147–160, Abril 1950.

[5] J. L. Massey, Applied Digital Information Theory – part I, Swiss Federal Institute of Technology-Zurich, Notas de Aula, 1980, *Information and Signal Processing Laboratory*.

[6] N. Abramson, **Information Theory and Coding**. New York: McGraw-Hill, 1963.

[7] T. M. Cover & J. A. Thomas, **Elements of Information Theory**, 2ª ed. John Wiley and Sons, 2006.

[8] J. L. Massey, Applied Digital Information Theory – part II, Swiss Federal Institute of Technology-Zurich, Notas de Aula, 1981, *Information and Signal Processing Laboratory*.

[9] S. Lin & D. J. Costello, **Error Control Coding**, 2ª ed. Prentice Hall, 2004.

[10] S. B. Wicker, **Error Control Systems for Digital Communication and Storage**. Prentice Hall, 1995.

[11] F. J. MacWilliams & N. J. A. Sloane, **The Theory of Error-Correcting Codes**. North-Holland, 1977.

[12] E. R. Berlekamp, **Algebraic Coding Theory**. McGraw-Hill, 1968.

[13] R. E. Blahut, **Algebraic Codes for Data Transmission**. Cambridge University Press, 2003.

[14] T. K. Moon, **Error correction coding : mathematical methods and algorithms**. John Wiley and Sons, 2005.

[15] C. Pimentel, **Comunicação Digital**. Brasport, Rio de Janeiro, 2007.

[16] J. G. Proakis & M. Salehi, **Digital Communications**, 5ª ed. McGraw-Hill Science/Engineering/Math, New York, 2007.

[17] C. E. Shannon, Two-way communication channels, In: **Proceedings of 4th Berkeley Symposium on Mathematical Statistics and Probability**, v. 1, n. 4, 1961, p. 611–644.

[18] E. Biglieri & L. Györfi, Eds., **Multiple Access Channels - Theory and Practice**, ser. NATO Security through Science. Amsterdan: IOS Press, 2007.

[19] K. S. Zigangirov, **Theory of Code Division Multiple Access Communication**. John Wiley and Sons, 2004.

[20] R. G. Gallager, A perspective on multiaccess channels, *IEEE Transactions on Information Theory*, v. IT-31, n. 2, p. 124–142, March 1985.

[21] N. Abramson, The ALOHA system – another alternative for computer communications, In: **Proceedings of Fall Joint Computer Conference**, v. 37. New York, USA: AFIPS Conference, November 1970, p. 281–285.

[22] L. G. Roberts, ALOHA packet system with and without slots and capture, In: **ACM SIGCOMM Computer Communication Review**, v. 5, n. 2, New York, USA, April 1975, p. 28–42.

[23] J. L. Massey, The capacity of the collision channel without feedback, *Abstracts of Papers, IEEE Int. Symp. Inform. Theory*, p. 101, June 1982.

[24] J. L. Massey & P. Mathys, The collision channel without feedback, *IEEE Transactions on Information Theory*, v. IT-31, n. 2, p. 192–204, March 1985.

[25] F. Zhao & L. J. Guibas, **Wireless Sensor Networks: An Information Processing Approach**. Elsevier, 2004.

[26] N. Q. A, L. Györfi, & J. L. Massey, Constructions of binary constant-weight cyclic codes and cyclically permutable codes, *IEEE Transactions on Information Theory*, v. IT-38, n. 3, p. 940–948, May 1985.

[27] E. N. Gilbert, Cyclically permutable error-correcting codes, *IEEE Transactions on Information Theory*, v. IT-9, n. 3, p. 175–182, July 1963.

[28] W. S. Wong, New protocol sequences for random-access channels without feedback, *IEEE Transactions on Information Theory*, v. IT-53, n. 6, p. 2060–2071, June 2007.

[29] L. Györfi & I. Vajda, Constructions of protocol sequences for multiple access collision channel without feedback, *IEEE Transactions on Information Theory*, v. IT-39, n. 5, p. 1762–1765, September 1993.

[30] G. Thomas, Capacity of the wireless packet collision channel without feedback, *IEEE Transactions on Information Theory*, v. IT-46, n. 3, p. 1141–1144, May 2000.

[31] V. C. Da Rocha Jr., Protocol sequences for collision channel without feedback, *IEE Electronics Letters*, v. 36, n. 24, p. 2010–2012, November 2000.

[32] L. Györfi & S. Győri, Coding for multiple-access collision channel without feedback, In: **Multiple Access Channels - Theory and Practice**, ser. NATO Security through Science Series, E. Biglieri & L. Györfi, Eds. Amsterdan: The Netherlands – IOS Press, 2007, p. 299–326.

[33] K. W. Shum, W. S. Wong, C. W. Sung, & C. S. Chen, Design and construction of protocol sequences: Shift invariance and user irrepressibility, In: **Proceedings of the International Symposium on Information Theory**. Seoul, Korea: ISIT, June 2009, p. 1368–1372.

[34] K. W. Shum, C. S. Chen, C. W. Sung, & W. S. Wong, Shift-invariant protocol sequences for the collision channel without feedback, *IEEE Transactions on Information Theory*, v. IT-55, n. 7, p. 3312–3322, July 2009.

[35] Y. Zhang, K. W. Shum, & W. S. Wong, User-detectable sequences for the collision channel without feedback, In: **Proceedings of the 7th International Symposium on Wireless Communication Systems**. York, United Kingdom: ISWCS, September 2010, p. 770–774.

[36] V. C. Da Rocha Jr. & J. S. de Lemos Neto, Nonlinear binary codes derived from constacyclic codes, In: **Proceedings of the 7th International Telecommunications Symposium**. Manaus, Brazil: ITS, September 2010.

[37] B. S. Tsybakov & N. B. Likhanov, Packet communication on a channel without feedback, *Problems of Information Transmission*, v. 19, n. 2, p. 69–84, May 1983.

[38] L. A. Bassalygo & M. S. Pinsker, Limited multiple-access of a nonsynchronous channel, *Problems of Information Transmission*, v. 19, n. 4, p. 92–96, May 1983, (in Russian).

[39] A. Hocquenghem, Codes correcteurs d'erreurs, *Chiffres*, v. 2, p. 147–156, 1959.

[40] R. C. Bose & D. K. Ray-Chaudhuri, On a class of error correcting binary group codes, *Information and Control*, v. 3, p. 68–79, March 1960.

[41] ——, Further results on error correcting group codes, *Information and Control*, v. 3, p. 279–290, September 1960.

[42] W. W. Peterson, Encoding and error-correction procedures for de Bose-Chaudhuri codes, *IRE Transactions on Information Theory*, v. IT-6, p. 459–470, September 1960.

[43] I. S. Reed & G. Solomon, Polynomial codes over certain finite fields, *Journal of the Society for Industrial and Applied Mathematics*, v. 8, n. 2, p. 300–304, June 1960.

[44] S. B. Wicker & V. K. Bhargava, **Reed-Solomon Codes and Their Applications**. IEEE Press, 1994.

[45] V. C. Da Rocha Jr., Maximum distance separable multilevel codes, *IEEE Transactions on Information Theory*, v. IT-30, n. 3, p. 547–548, May 1984.

[46] ——, Algebraic decoding of a class of multilevel pseudocyclic codes, *IEE Electronics Letters*, v. 25, n. 5, p. 341–342, March 1989.

[47] V. C. Da Rocha Jr., R. M. C. de Souza, & P. G. Farrell, Multilevel pseudocyclic codes, *Journal of Information and Optimization Sciences*, v. 11, n. 1, p. 101–106, January 1990.

[48] A. Krishna & D. Sarwate, Pseudocyclic maximum-distance-separable codes, *IEEE Transactions on Information Theory*, v. IT-36, n. 4, p. 880–884, May 1990.

[49] D. M. Burton, **Elementary Number Theory**, 7ª ed. McGraw-Hill, 2010.

[50] J. M. Jensen, Cyclic concatenated codes with constacyclic outer codes, *IEEE Transactions on Information Theory*, v. IT-38, n. 3, p. 950–959, May 1992.

[51] S. Sriram & S. Hosur, Cyclically permutable codes for rapid acquisition in DS-CDMA systems with asynchronous base stations, *IEEE Journal on Selected Areas in Communications*, v. 19, n. 1, p. 83–94, January 2001.

[52] M. Kuribayashi & H. Tanaka, How to generate cyclically permutable codes from cyclic codes, *IEEE Transactions on Information Theory*, v. IT-52, n. 10, p. 4660–4663, October 2006.

SOBRE O AUTOR

O autor nasceu em Bezerros, Pernambuco, no dia 27 de Novembro de 1980. Formou-se em Engenharia Elétrica, modalidade Eletrônica, pela Universidade Federal de Pernambuco em 2004. É membro da *Sociedade Brasileira de Telecomunicações*. Seus interesses de pesquisa incluem Teoria da Informação, Códigos Corretores de Erro, Sistemas de Comunicação Digital, Processamento Digital de Sinais e Matemática Aplicada.

Endereço: Av. Benedita de Andrade, 92
55660 – 000 São Sebastião
Bezerros – PE
Brasil

e-mail: netosam@msn.com

Esta dissertação foi diagramada usando LaTeX 2ε[1] pelo autor.

[1] LaTeX 2ε é uma extensão do LaTeX. LaTeX é uma coleção de macros criadas por Leslie Lamport para o sistema TeX, que foi desenvolvido por Donald E. Knuth. TeX é uma marca registrada da Sociedade Americana de Matemática (AMS). O estilo usado na formatação desta dissertação foi escrito por Dinesh Das, Universidade do Texas. Modificado por Renato José de Sobral Cintra (2001) e por André Leite Wanderley (2005), ambos da Universidade Federal de Pernambuco. Sua última modificação ocorreu em 2010 realizada por José Sampaio de Lemos Neto, também da Universidade Federal de Pernambuco.