



Códigos BCH aplicados na análise mutacional da enzima mitocondrial ATP6 ¹

BCH codes related to mutational analysis of mitochondrial enzyme ATP6

Alice Noronha de Oliveira

Universidade Federal de Alfenas

alice_noronhaa@hotmail.com

ORCID: 0000-0002-4381-2477

Anderson José de Oliveira

Universidade Federal de Alfenas

anderson.oliveira@unifal-mg.edu.br

ORCID: 0000-0002-5467-9827

Diogo Guilherme Pereira

Universidade Federal de Alfenas

diogo.bioinformatica@gmail.com

ORCID: 0000-0002-5009-3423

Resumo. Neste trabalho foi utilizado o algoritmo de geração de proteínas, que realiza a construção de códigos BCH, para reproduzir uma sequência de DNA relacionada à enzima mitocondrial ATP6. Este algoritmo permite o estudo de mutações na sequência gerada, bem como a sua identificação, através de códigos corretores de erros. Na reprodução da sequência da enzima mitocondrial ATP6 foi possível observar que ocorre a troca de um nucleotídeo na posição da trinca 17, acarretando em uma mutação não silenciosa. Assim, será analisado como essa alteração pode modificar a arquitetura biológica da sequência gerada. Os resultados dessas análises poderão ser úteis no estudo de mutações genéticas, pois a troca de um aminoácido na enzima ATP6 pode acarretar em algumas doenças genéticas.

Palavras-chave. Álgebra. Biomatemática. Códigos BCH. Enzima ATP6.

Abstract. In this work, the protein generation algorithm was used, which performs the construction of BCH codes, in order to reproduce an enzyme-related DNA sequence mitochondrial ATP6. This algorithm allows the study of mutations in the generated sequence as well as its identification, through error-correcting codes. In sequence reproduction of the

¹À Pro-Reitoria de Pesquisa e Pós-Graduação, à UNIFAL-MG e à FAPEMIG.

mitochondrial enzyme ATP6 it was possible to observe that the exchange of a nucleotide occurs at the position of crack 17, resulting in a non-silent mutation. Thus, it will be analyzed how this alteration can modify the biological architecture of the generated sequence. The results of these analyzes may be useful in the study of genetic mutations, since the exchange of an amino acid in the ATP6 enzyme can cause some genetic diseases.

Keywords. Algebra. Biomathematics. BCH codes. ATP6 enzyme.

Mathematics Subject Classification (MSC): primary 03C05; secondary 92B05.

1 Introdução

A necessidade de se garantir que uma mensagem seja armazenada ou transmitida de forma confiável pelos diversos meios existentes, exige o uso da teoria dos códigos corretores de erros. Propostos inicialmente por R. W. Hamming, no final da década de 40, os códigos corretores de erros são atualmente utilizados desde o processo de comunicação via satélite até o processo de comunicação celular em um sistema biológico.

Embora o processo de transmissão de informação seja considerado, na comunicação digital, como um processo que realiza a passagem de sinais através dos meios físicos de comunicação, a transmissão de informação também pode ser considerada no contexto genético. Nesse contexto, a transmissão de informação é realizada por meio do processo de transcrição, em que a informação obtida no DNA é transformada em proteínas. Desse modo, é natural questionar se existe uma estrutura intrínseca de códigos corretores de erros na transmissão de informação genética, isto é, se sequências de DNA podem ser reproduzidas mediante a utilização de códigos corretores de erros. Os trabalhos de [8] e [3] apresentaram uma resposta afirmativa para esse questionamento. Eles apontaram a existência de um código matemático que transcreve a sequência de DNA, bem como formas de reproduzir e classificar matematicamente essas sequências, utilizando uma analogia entre um modelo de sistema de comunicação digital e um modelo de sistema de comunicação de informação genética. Essa correspondência possibilitou o desenvolvimento de um algoritmo de geração de proteínas sobre a estrutura algébrica de anéis e de corpos, que auxiliam no processo de análise de fenômenos mutacionais.

Diferentes sequências de DNA, com funcionalidades biológicas diversas, podem ser reproduzidas pelo algoritmo. No entanto, elas devem atender a duas restrições: 1) a sequência de DNA deve ter comprimento $n = 2^r - 1$ para a estrutura de anel e $n = 4^r - 1$ para a estrutura de corpo, onde r denota o grau da extensão e 2) é preciso obter uma sequência de DNA original postada no NCBI (National Center for Biotechnology

Information). Vale destacar, que para a obtenção dos resultados no presente artigo foi utilizada a estrutura de anel.

A enzima mitocondrial ATP sintase está localizada na parte interna da mitocôndria e a sua principal função é sintetizar a maior parte do ATP (adenosina trifosfato) que é a principal forma de energia química, viabilizando a maioria das reações metabólicas que ocorrem no interior das células. Essa enzima é composta por diferentes subunidades proteicas, dentre elas a subunidade *a* da membrana de ATP sintase, ATP6, que possui comprimento de $2^6 - 1 = 63$ nucleotídeos, e atende as restrições do algoritmo. A importância da análise dessa sequência, via códigos corretores de erros, reside no fato que alterações na sequência ATP6 acarretam em doenças mitocondriais graves como, por exemplo, a doença de NARP.

Por meio deste artigo, temos como propósito gerar a sequência de DNA associada ao ATP6, através do algoritmo de geração de proteínas, a fim de localizar onde ocorreu uma mutação, de efeito maléfico, nessa molécula de DNA e analisar como essa alteração pode interferir na produção de ATP, pela proteína ATP6. Por meio dessa análise é possível mostrar algebricamente a mutação que ocorre nessa sequência e as consequências bioquímicas acerca dessa mutação.

2 Preliminares

Nesta seção serão apresentados os conceitos teóricos fundamentais no processo de construção deste trabalho. As referências utilizadas foram [1], [3], [4] e [8].

2.1 Analogia entre o dogma central da biologia e um sistema de comunicação digital

A informação genética contida no DNA é conservada através da sua replicação e é organizada através de outros dois processos: a transcrição que converte a informação do DNA em uma fita de RNA e a tradução, que converte a informação contida no RNA em proteínas. A relação entre três bases nitrogenadas encontradas no RNA (adenina, uracila, guanina e citosina) e os aminoácidos encontrados em uma proteína é denominada de código genético.

Esse fluxo de informação do código genético é conhecido como dogma central da biologia, que lida com a transmissão e tradução de dados genéticos [8]. De acordo com esse dogma, os mecanismos de transmissão genética ocorre de um modo unidirecional em que o DNA produz o RNA por meio de transcrição e o RNA codifica proteínas por tradução. No entanto, ocasionalmente, podem ocorrer erros nesses processos provocando

alterações na sequência de nucleotídeos, essas alterações também são conhecidas como mutações [4].

Já um sistema de comunicação digital é compreendido como um conjunto de mecanismos que lida com a transmissão de informação de um emissor a um determinado receptor, por meio de um canal de comunicação. É desejável que essa transmissão seja efetuada com alta confiabilidade, permitindo que a mensagem recebida seja igual a original. Porém, nem sempre a troca de informações é bem sucedida, pois durante esse processo podem ocorrer perturbações no canal de comunicação que dificultam o entendimento da mensagem enviada pelo emissor. Para corrigir tais interferências é utilizada a teoria dos códigos corretores de erros.

Por meio das descrições feitas anteriormente em relação ao dogma central da biologia e um sistema de comunicação digital, pode-se realizar algumas associações. Em um sistema de comunicação digital a informação é transmitida por um emissor, biologicamente quem exerce este papel é o DNA; o processo de transcrição tem por objetivo a transmissão de informação genética, do ponto de vista da comunicação, o processo de transcrição é tido como sendo o canal de um sistema de comunicação; e o receptor é o destino final da mensagem enviada, no caso da transmissão de informação genética, o destino final da informação é a proteína. Com isso, os sistemas biológicos podem ser modelados e estudados como um sistema de comunicação no qual os blocos do transmissor, canal e receptor estão relacionados ao DNA, transcrição e proteína, respectivamente.

A aplicação da teoria da comunicação para análise de dados genéticos iniciou-se décadas atrás. No entanto devido ao pequeno número de dados genéticos, esses pesquisadores não obtiveram sucesso em suas pesquisas. Atualmente, apesar do número reduzido de pesquisadores, os trabalhos nessa linha de pesquisa seguem duas frentes principais: 1) aplicar os conceitos da teoria da informação com o objetivo de apresentar um método de determinação das regiões codantes e não-codantes na estrutura do DNA e 2) aplicar os conceitos da teoria da codificação para adquirir fundamentação para a caracterização dos códigos corretores de erros. Porém, nas duas linhas de pesquisa sempre foi questionado a existência de um código corretor de erros na estrutura do DNA.

Em 2010, [8] e [3] reponderam de maneira positiva a esse questionamento. Em seus trabalhos foi proposto um modelo para o sistema de comunicação de informação genética análogo ao modelo de um sistema de comunicação digital. Com o objetivo de que as bases nitrogenadas do DNA fossem identificadas como uma palavra-código de um código corretor de erros, realizou-se uma busca do código corretor de erro e da estrutura matemática que fosse mais capaz e apropriado para reproduzir sequências de DNA.

Os códigos corretores de erros são classificados em códigos lineares e não-lineares. Os códigos não-lineares Nordstrom-Robinson e Preparata apresentam uma alta capaci-

dade de correção de erros, no entanto como o seu processo de decodificação é mais complexo devido algumas propriedades estruturais no processo de geração, eles não seriam bons candidatos. Por outro lado, um dos principais códigos lineares, os códigos BCH se tornam bons candidatos a serem utilizados para a geração de sequências de DNA, devido à simplicidade dos processos de codificação e decodificação associados, além de possuir uma alta capacidade de correção de erros.

Os parâmetros associados a um código BCH são denotados por (n, k, d_{\min}) , onde n é o comprimento da palavra-código, k é a dimensão do código e d_{\min} é a distância mínima do código. Destaca-se que para obter o d_{\min} de um código BCH é utilizada a métrica de Hamming nos cálculos, uma vez que essa métrica é uma das mais utilizadas e que facilitam o processo de codificação e decodificação. No modelo foi assumido também que os códigos BCH utilizados correspondem a construção mais simples e conhecida, assim, algumas considerações foram assumidas, como por exemplo, apesar de ser verdade que um deslocamento cíclico de uma palavra-código de um código BCH será também uma palavra-código deste código, o mesmo não se pode afirmar biologicamente, ou seja, nem sempre um deslocamento cíclico em uma sequência de DNA funcional também acarretará uma sequência de DNA funcional.

Nos processos de codificação e decodificação dos códigos corretores de erros é necessária uma estrutura matemática bem definida. Com isso, no modelo proposto por [8] e [3], sabendo que os RNA são sequências constituídas de 4 elementos e que os únicos alfabetos matemáticos sobre os quais se conhecem técnicas para a construção de códigos corretores de erros de cardinalidade 4 são \mathbb{Z}_4 ou \mathbb{F}_4 , foi estabelecido um mapeamento entre o alfabeto do código genético e o alfabeto do código corretor de erros, em que ambos possuem cardinalidade 4. Desse modo, definiu-se o código corretor de erro e a estrutura matemática utilizada para identificação e geração de sequências de DNA.

2.2 Transformando sequências de DNA em palavras código de um código corretor de erro

O código genético é composto pela combinação das 4 bases nitrogenadas: adenina (A), citosina (C), guanina (G) e timina (T) ou uracila (U), assim, o alfabeto 4-ário do código genético pode ser denotado pelo conjunto $N = \{A, C, G, T/U\}$. Como a estrutura algébrica do alfabeto do código genético é desconhecida, o conjunto N pode ser relacionado ao alfabeto 4-ário dos códigos corretores de erros sobre uma estrutura de anel, indicado por $\mathbb{Z}_4 = \{0, 1, 2, 3\}$, que obedece as operações de adição e multiplicação módulo-4. A escolha dos elementos que constituem o conjunto \mathbb{Z}_4 é justificada por ser a estrutura de anel mais simples.

Logo após determinar a estrutura algébrica e o alfabeto utilizado pelo CCE, é necessário determinar o grau r da extensão de Galois e , conseqüentemente, os polinômios primitivos relacionados a r . Para cada valor de r , existem vários polinômios primitivos, que devem ser considerados como diferentes códigos a serem analisados. Além disso, tem-se que para cada polinômio primitivo existe um polinômio gerador $g(x)$ do código BCH de comprimento n .

O polinômio $g(x)$ é especificado em termos das suas raízes da extensão de Galois e ao determiná-lo, obtém-se a matriz geradora G e a matriz de verificação de paridade H , uma vez que a matriz G é obtida realizando o deslocamento dos termos do polinômio $g(x)$ e a matriz H é determinada pelo polinômio $h(x) = (x^n - 1) / g(x)$ [5].

Depois de realizar a construção do código BCH é verificado se este código é realmente capaz de reproduzir sequências de DNA. No entanto, o melhor mapeamento entre $N \rightarrow \mathbb{Z}_4$ é desconhecido, dessa maneira todas as possibilidades de associação dos elementos serão analisadas para verificar qual delas é a melhor. Assim, toda sequência de DNA será considerada como uma das 24 permutações entre esses dois conjuntos. Para determinar quais conversões são, de fato uma palavra-código a relação $v \cdot H^T$ é empregada, em que v é uma possível palavra-código e H^T é a transposta da matriz de verificação de paridade [1]. Se essa relação for igual a 0, essa permutação pertence ao código BCH de comprimento n e, desse modo, são geradas as sequências de DNA através de um CCE.

Devido a complexidade de gerar sequências de DNA mediante os códigos BCH, [8] e [3] desenvolveram o algoritmo de geração de proteínas.

3 Algoritmo de Geração de Proteínas

O algoritmo de geração de proteínas realiza a construção de códigos BCH sobre a estrutura de corpo e anel, capaz de reproduzir sequências de DNA de comprimento $n = 4^r - 1$ e $n = 2^r - 1$, respectivamente, com até dois nucleotídeos de diferença da sequência de DNA, disponível no NCBI. Além disso, possui a capacidade de correção de erros dada por meio da relação $d_{\min} = 2t + 1$, em que t indica a quantidade de erros. Os parâmetros do código BCH desse algoritmo são caracterizados da seguinte forma: n = comprimento das sequências de DNA; k = comprimento da sequência de informação responsável pela geração da sequência de DNA e d_{\min} = o menor número de posições em que quaisquer duas palavras código diferem.

No algoritmo, os códigos BCH sobre anéis serão construídos em todas as distâncias mínimas e em todos os polinômios primitivos e geradores de cada extensão de Galois de grau r , com o objetivo de encontrar um código capaz de gerar as sequências de DNA.

Esse algoritmo pode ser descrito, passo a passo, da seguinte forma:

- **Passo 1:** especificar a estrutura matemática e o alfabeto do código.
- **Passo 2:** determinar a extensão de Galois.
- **Passo 3:** determinar todos os polinômios primitivos com grau r .
- **Passo 4:** selecionar um polinômio primitivo de cada vez e determinar a extensão de Galois, e com esse mesmo polinômio primitivo determinar a extensão do anel \mathbb{Z}_4 .
- **Passo 5:** encontrar o polinômio gerador $g(x)$ e em seguida o polinômio de verificação de paridade $h(x)$ do código BCH. Ao determinar esses polinômios é possível obter a matriz geradora e a matriz de verificação de paridade.
- **Passo 6:** converter a sequência original composta dos elementos do grupo N na sequência correspondente com elementos em \mathbb{Z}_4 .
- **Passo 7:** verificar pela relação $v \cdot H^T$ se cada uma das sequências convertidas é uma palavra do código:
 - se $v \cdot H^T = 0$, então armazene a sequência;
 - se $v \cdot H^T \neq 0$, implica que pode haver até t diferenças de nucleotídeos, então precisa-se considerar todas as combinações possíveis, ou seja, se um nucleotídeo é rotulado com o valor 2, então trocaremos esse valor para as outras possibilidades de valores disponíveis em \mathbb{Z}_4 . Ao alterar o valor da rotulagem, devemos verificar novamente se a nova sequência de DNA produzida é uma palavra-código do código BCH. Se for, o algoritmo o armazena; caso contrário, desconsidera.

É importante destacar nesse passo, a complexidade computacional utilizada pelo algoritmo para verificar se as sequências convertidas são palavras-código. O parâmetro de entrada mais importante que impactará diretamente na execução do programa é o grau r dos polinômios primitivos. Pois este parâmetro indica quantos polinômios primitivos deverão ser usados nos passos do programa e armazenados na memória do computador.

O Quadro 1 apresenta o número de polinômios primitivos armazenados separados por grau.

Quadro 1: número de polinômios armazenados no banco de dados separados por grau.

Grau r	Polinômios	Grau r	Polinômios
2	1	3	2
4	2	5	6
6	6	7	18
8	16	9	48
10	60	11	176
12	144	13	630
14	756	15	1.800
16	2.048	17	7.710
18	7.776	19	27.594
20	24.000		

Note que o crescimento do número de polinômios primitivos é exponencial à medida que o grau do polinômio primitivo aumenta.

O Quadro 2 apresenta alguns tempos de execução, conforme o valor dos parâmetros de entrada, isto é, comprimento da sequência de DNA, n , a utilização dos 24 rótulos possíveis.

Quadro 2: tempos de execução do programa conforme os dados de entrada.

n	3 rótulos	24 rótulos
63	0,017 s	0,033 s
127	0,059 s	0,093 s
255	0,361 s	0,235 s
511	0,325 s	2,003 s
1023	1,155 s	8,731 s
2047	12,158 s	96,031 s

- **Passo 8:** converter cada sequência armazenada no **Passo 7** com elementos pertencentes ao conjunto \mathbb{Z}_4 , na sequência correspondente com elementos em N . Compare cada uma dessas sequências com a sequência original e mostre a posição em que os nucleotídeos diferem.
- **Passo 9:** volte ao **Passo 4**. Selecione outro polinômio primitivo e verifique se todos os outros já foram usados: caso contrário, repita os **Passos 5** ao **7** para cada um;

caso contrário, vá para o **Passo 10**.

- **Passo 10:** fim.

4 Resultados e Discussões

De acordo com [8], diferentes sequências de DNA com comprimentos diversos podem ser reproduzidas pelo algoritmo de geração de proteínas, desde que atendam as restrições impostas pelo algoritmo. No entanto, neste trabalho, considerou-se apenas sequências de DNA de comprimento de $n = 2^r - 1 = 63$ nucleotídeos, em que $r = 6$, sendo este o comprimento mínimo de uma sequência de DNA que atende a restrição do algoritmo. Ao realizar um levantamento das possíveis sequências de DNA com 63 nucleotídeos foram obtidas várias sequências de DNA relacionadas a vírus e bactérias e uma sequência referente à proteína mitocondrial ATP6.

A enzima mitocondrial ATP6 é uma proteína essencial nas mitocôndrias. As mitocôndrias são organelas celulares e a sua principal função é a de transformar energia química encontrada no citoplasma em energia acessível à célula. Essa energia é acumulada principalmente em forma de ATP (adenina trifosfato), que será utilizada para atividades metabólicas que ocorrem dentro da célula. A ATP sintase mitocondrial está localizada na membrana interna das mitocôndrias e é responsável por sintetizar a maior parte do ATP [9]. Essa enzima é um complexo proteico de grandes dimensões, sendo composta por duas regiões rotativas F_o e F_1 , em que cada uma dessas regiões é constituída de várias subunidades proteicas.

A enzima ATP6 é uma subunidade proteica pertencente ao ATP sintase. Essa subunidade desempenha um importante papel no transporte dos prótons que serão sintetizados em ATP. Assim, a ocorrência de uma mutação na subunidade ATP6 pode alterar o funcionamento normal de uma mitocôndria e tem o potencial de gerar, nos seres humanos, distúrbios complexos com expressão e gravidade diferenciadas, afetando da infância até a vida adulta, com grau de manifestação diferenciadas.

Apesar de alguns distúrbios neurodegenerativos e cardiovasculares estarem relacionados com mutações no ATP6, os mecanismos pelos quais as alterações nessa proteína interferem na síntese do ATP ainda precisam ser elucidados [2].

Nesta seção foi executado o algoritmo de geração de proteínas, a fim de identificar se a enzima mitocondrial ATP6 poderia ser identificada e reproduzida por meio dos códigos corretores de erros. Uma vez observada e determinada a estrutura algébrica nessa sequência, pode-se realizar análises mutacionais identificando possíveis mutações que podem ocorrer na sequência ATP6 e estudar se essa alteração seria de natureza maléfica ou be-

néfica. Os passos do algoritmo são determinados de maneira análoga ao procedimento adotado na Seção 3.

• **Passo 1: especificar a estrutura matemática e o alfabeto do código**

O alfabeto 4-ário do código genético está relacionado ao conjunto formado pelos nucleotídeos denotados por $N = \{A, C, G, T/U\}$ que corresponde, respectivamente, à adenina, citosina, guanina e timina/uracila. Adotaremos o alfabeto 4-ário denotado por $\mathbb{Z}_4 = \{0, 1, 2, 3\}$. Todas as operações algébricas obedecerão às operações de adição e multiplicação módulo 4, Quadros 3 e 4, respectivamente.

Quadro 3: adição módulo 4

+	0	1	2	3
0	0	1	2	3
1	1	2	3	0
2	2	3	0	1
3	3	0	1	2

Quadro 4: multiplicação módulo 4

·	0	1	2	3
0	0	0	0	0
1	0	1	2	3
2	0	2	0	2
3	0	3	2	1

• **Passo 2: determinar a extensão de Galois**

Aqui, será utilizada uma sequência de DNA formada por 63 nucleotídeos. Logo, o grau dos polinômios primitivos a serem usados na extensão de Galois do corpo $\text{GF}(2^r)$ é $r = 6$, pois $n = 2^r - 1 = 2^6 - 1 = 63$.

• **Passo 3: determinar todos os polinômios primitivos $p(x)$, relacionados à extensão de Galois**

Neste passo, são informados todos os polinômios primitivos $p(x)$ relacionados ao grau de extensão do corpo de Galois $r = 6$, como mostra o Quadro 5. Estes polinômios primitivos podem ser encontrados em [1].

• **Passo 4: selecionar um polinômio primitivo de cada vez e determinar a extensão de Galois, e com esse mesmo polinômio primitivo determinar a extensão do anel \mathbb{Z}_4**

Quadro 5: polinômios primitivos da extensão de Galois de grau $r = 6$.

Polinômios primitivos $p(x)$	
$p_1(x) = x^6 + x + 1$	$p_4(x) = x^6 + x^5 + x^2 + x + 1$
$p_2(x) = x^6 + x^4 + x^3 + x + 1$	$p_5(x) = x^6 + x^5 + x^3 + x^2 + 1$
$p_3(x) = x^6 + x^5 + 1$	$p_6(x) = x^6 + x^5 + x^4 + x + 1$

Primeiramente, deve-se obter a extensão do corpo $GF(2)$ por meio de um polinômio primitivo de grau $r = 6$. Em seguida, utilizando o mesmo polinômio primitivo de grau $r = 6$, constrói-se a extensão do anel de Galois $GF(4, 6) = GF(p^k, r) = \mathbb{Z}_4$. Feito isto, determina-se a ordem do grupo cíclico pertencente ao grupo das unidades e assim, pode-se determinar todos os elementos pertencentes ao grupo das unidades de ordem n . O grupo das unidades de ordem n dá condições de especificar um código BCH de comprimento $n = 63$.

A extensão do corpo $GF(2)$ é obtida por um ideal gerado por qualquer um dos polinômios primitivos de grau $r = 6$. Porém, para que este exemplo não fique muito extenso serão apresentados apenas os passos para o polinômio primitivo $x^6 + x^4 + x^3 + x^1 + 1$. Assim, considere o corpo de Galois $GF(2^r) = GF(2^6) = GF(64) = \mathbb{F}_{64}$ e assumamos que o polinômio primitivo $p(x)$ utilizado é o $x^6 + x^4 + x^3 + x^1 + 1$.

Seja α uma raiz de $x^6 + x^4 + x^3 + x^1 + 1 = 0$, então temos $\alpha^6 + \alpha^4 + \alpha^3 + \alpha^1 + 1 = 0$ ou $\alpha^6 = -\alpha^4 - \alpha^3 - \alpha^1 - 1$. Porém, como os coeficientes dos polinômios que formam o conjunto dos elementos de \mathbb{F}_{64} pertencem a \mathbb{F}_2 devemos fazer a redução módulo 2. Portanto, $\alpha^6 = \alpha^4 + \alpha^3 + \alpha^1 + 1$. Os elementos de \mathbb{F}_{64} são mostrados no Quadro 6.

Quadro 6: elementos de \mathbb{F}_{64}

\mathbb{F}_{64}	$(\alpha^0 \alpha^1 \alpha^2 \alpha^3 \alpha^4 \alpha^5)$	\mathbb{F}_{64}	$(\alpha^0 \alpha^1 \alpha^2 \alpha^3 \alpha^4 \alpha^5)$	\mathbb{F}_{64}	$(\alpha^0 \alpha^1 \alpha^2 \alpha^3 \alpha^4 \alpha^5)$
0	(000000)	1	(100000)	α^1	(010000)
α^2	(001000)	α^3	(000100)	α^4	(000010)
α^5	(000001)	α^6	(110110)	α^7	(011011)
α^8	(111011)	α^9	(101011)	α^{10}	(100011)
α^{11}	(100111)	α^{12}	(100101)	α^{13}	(100100)
α^{14}	(010010)	α^{15}	(001001)	α^{16}	(110010)
α^{17}	(011001)	α^{18}	(111010)	α^{19}	(011101)
α^{20}	(111000)	α^{21}	(011100)	α^{22}	(001110)
α^{23}	(000111)	α^{24}	(110101)	α^{25}	(101100)
α^{26}	(010110)	α^{27}	(001011)	α^{28}	(110011)
α^{29}	(101111)	α^{30}	(100001)	α^{31}	(100110)
α^{32}	(010011)	α^{33}	(111111)	α^{34}	(101001)
α^{35}	(100010)	α^{36}	(010001)	α^{37}	(111110)
α^{38}	(011111)	α^{39}	(111001)	α^{40}	(101010)
α^{41}	(010101)	α^{42}	(111100)	α^{43}	(011110)
α^{44}	(001111)	α^{45}	(110001)	α^{46}	(101110)
α^{47}	(010111)	α^{48}	(111101)	α^{49}	(101000)
α^{50}	(010100)	α^{51}	(001010)	α^{52}	(000101)

α^{53}	(110100)	α^{54}	(011010)	α^{55}	(001101)
α^{56}	(110000)	α^{57}	(011000)	α^{58}	(001100)
α^{59}	(000110)	α^{60}	(000011)	α^{61}	(110111)
α^{62}	(101101)	α^{63}	(100000)		

Agora é possível determinar a extensão do anel \mathbb{Z}_4 . A extensão do anel \mathbb{Z}_4 é dada pelo quociente do conjunto de todos os polinômios com coeficiente em \mathbb{Z}_4 pelo ideal gerado pelo $x^6 + x^4 + x^3 + x^1 + 1$.

Seja β uma raiz de $p(x)$. Então, $\beta^6 + \beta^4 + \beta^3 + \beta^1 + 1 = 0$ levando a $\beta^6 = -\beta^4 - \beta^3 - \beta^1 - 1$. Então, $\beta^6 = 3\beta^4 + 3\beta^3 + 3\beta^1 + 3$. Considerando $f = (010000) = \beta$, o Quadro 7 apresenta todos os elementos não nulos e inversíveis do grupo cíclico do grupo $GR^*(4, 6)$:

Quadro 7: elementos do grupo cíclico $GR^*(4, 6)$

$GR^*(4, 6)$	$(\beta^0 \beta^1 \beta^2 \beta^3 \beta^4 \beta^5)$	$GR^*(4, 6)$	$(\beta^0 \beta^1 \beta^2 \beta^3 \beta^4 \beta^5)$	$GR^*(4, 6)$	$(\beta^0 \beta^1 \beta^2 \beta^3 \beta^4 \beta^5)$
1	(100000)	$f = \beta^1$	(010000)	$f^2 = \beta^2$	(001000)
$f^3 = \beta^3$	(000100)	$f^4 = \beta^4$	(000010)	$f^5 = \beta^5$	(000001)
$f^6 = \beta^6$	(330330)	$f^7 = \beta^7$	(033033)	$f^8 = \beta^8$	(113013)
$f^9 = \beta^9$	(121011)	$f^{10} = \beta^{10}$	(302031)	$f^{11} = \beta^{11}$	(320133)
$f^{12} = \beta^{12}$	(102123)	$f^{13} = \beta^{13}$	(120322)	$f^{14} = \beta^{14}$	(232212)
$f^{15} = \beta^{15}$	(203001)	$f^{16} = \beta^{16}$	(310230)	$f^{17} = \beta^{17}$	(031023)
$f^{18} = \beta^{18}$	(113212)	$f^{19} = \beta^{19}$	(231101)	$f^{20} = \beta^{20}$	(313000)
$f^{21} = \beta^{21}$	(031300)	$f^{22} = \beta^{22}$	(003130)	$f^{23} = \beta^{23}$	(000313)
$f^{24} = \beta^{24}$	(110101)	$f^{25} = \beta^{25}$	(301300)	$f^{26} = \beta^{26}$	(030130)
$f^{27} = \beta^{27}$	(003013)	$f^{28} = \beta^{28}$	(110011)	$f^{29} = \beta^{29}$	(301331)
$f^{30} = \beta^{30}$	(320023)	$f^{31} = \beta^{31}$	(102112)	$f^{32} = \beta^{32}$	(230031)
$f^{33} = \beta^{33}$	(313333)	$f^{34} = \beta^{34}$	(101003)	$f^{35} = \beta^{35}$	(120210)
$f^{36} = \beta^{36}$	(012021)	$f^{37} = \beta^{37}$	(331132)	$f^{38} = \beta^{38}$	(213333)
$f^{39} = \beta^{39}$	(131003)	$f^{40} = \beta^{40}$	(123210)	$f^{41} = \beta^{41}$	(012321)
$f^{42} = \beta^{42}$	(331122)	$f^{43} = \beta^{43}$	(213332)	$f^{44} = \beta^{44}$	(201113)
$f^{45} = \beta^{45}$	(130221)	$f^{46} = \beta^{46}$	(303312)	$f^{47} = \beta^{47}$	(210111)
$f^{48} = \beta^{48}$	(311301)	$f^{49} = \beta^{49}$	(321020)	$f^{50} = \beta^{50}$	(032102)
$f^{51} = \beta^{51}$	(223030)	$f^{52} = \beta^{52}$	(022303)	$f^{53} = \beta^{53}$	(112300)
$f^{54} = \beta^{54}$	(011230)	$f^{55} = \beta^{55}$	(001123)	$f^{56} = \beta^{56}$	(110222)
$f^{57} = \beta^{57}$	(231202)	$f^{58} = \beta^{58}$	(203300)	$f^{59} = \beta^{59}$	(020330)
$f^{60} = \beta^{60}$	(002033)	$f^{61} = \beta^{61}$	(110313)	$f^{62} = \beta^{62}$	(121101)
$f^{63} = \beta^{63}$	(302000)	$f^{64} = \beta^{64}$	(030200)	$f^{65} = \beta^{65}$	(003020)
$f^{66} = \beta^{66}$	(000302)	$f^{67} = \beta^{67}$	(220210)	$f^{68} = \beta^{68}$	(022021)
$f^{69} = \beta^{69}$	(332132)	$f^{70} = \beta^{70}$	(213033)	$f^{71} = \beta^{71}$	(131013)
$f^{72} = \beta^{72}$	(123211)	$f^{73} = \beta^{73}$	(302211)	$f^{74} = \beta^{74}$	(320111)
$f^{75} = \beta^{75}$	(322301)	$f^{76} = \beta^{76}$	(322120)	$f^{77} = \beta^{77}$	(032212)

$f^{78} = \beta^{78}$	(223001)	$f^{79} = \beta^{79}$	(312230)	$f^{80} = \beta^{80}$	(031223)
$f^{81} = \beta^{81}$	(113232)	$f^{82} = \beta^{82}$	(231103)	$f^{83} = \beta^{83}$	(133220)
$f^{84} = \beta^{84}$	(013322)	$f^{85} = \beta^{85}$	(221112)	$f^{86} = \beta^{86}$	(202331)
$f^{87} = \beta^{87}$	(310123)	$f^{88} = \beta^{88}$	(101122)	$f^{89} = \beta^{89}$	(230332)
$f^{90} = \beta^{90}$	(203213)	$f^{91} = \beta^{91}$	(130031)	$f^{92} = \beta^{92}$	(303333)
$f^{93} = \beta^{93}$	(100003)	$f^{94} = \beta^{94}$	(120110)	$f^{95} = \beta^{95}$	(012011)
$f^{96} = \beta^{96}$	(331131)	$f^{97} = \beta^{97}$	(323003)	$f^{98} = \beta^{98}$	(102010)
$f^{99} = \beta^{99}$	(010201)	$f^{100} = \beta^{100}$	(331310)	$f^{101} = \beta^{101}$	(033131)
$f^{102} = \beta^{102}$	(333203)	$f^{103} = \beta^{103}$	(103030)	$f^{104} = \beta^{104}$	(010303)
$f^{105} = \beta^{105}$	(111100)	$f^{106} = \beta^{106}$	(011110)	$f^{107} = \beta^{107}$	(001111)
$f^{108} = \beta^{108}$	(330001)	$f^{109} = \beta^{109}$	(323330)	$f^{110} = \beta^{110}$	(032333)
$f^{111} = \beta^{111}$	(113303)	$f^{112} = \beta^{112}$	(121000)	$f^{113} = \beta^{113}$	(012100)
$f^{114} = \beta^{114}$	(001210)	$f^{115} = \beta^{115}$	(000121)	$f^{116} = \beta^{116}$	(330302)
$f^{117} = \beta^{117}$	(213210)	$f^{118} = \beta^{118}$	(021321)	$f^{119} = \beta^{119}$	(332022)
$f^{120} = \beta^{120}$	(213022)	$f^{121} = \beta^{121}$	(201122)	$f^{122} = \beta^{122}$	(200332)
$f^{123} = \beta^{123}$	(200213)	$f^{124} = \beta^{124}$	(130131)	$f^{125} = \beta^{125}$	(303303)
$f^{126} = \beta^{126}$	(100000)				

Ao determinar a extensão do anel \mathbb{Z}_4 , o elemento f^{126} é igual ao elemento f^1 . Logo, f gera um grupo cíclico de ordem 126. Então f gera um grupo cíclico de ordem $n \cdot d$ em $GR^*(4, 6)$, com $d \geq 1 \in \mathbb{Z}$ e f^d gera um subgrupo cíclico cuja ordem é 63 em $GR^*(4, 6)$. Como f gera um grupo de ordem $n \cdot d$, onde $n = 63$ (comprimento da sequência), temos que $n \cdot d = 63 \cdot d = 126$, implicando em $d = 2$. Maiores informações podem ser encontradas em [5].

Como $d = 2$, tem-se um subgrupo formado por 63 elementos com $f^2 \rightarrow (001000)$, sendo este considerado como o elemento primitivo que gera o subgrupo cíclico G_{63} . O Quadro 8 apresenta os elementos constituintes do subgrupo.

Quadro 8: elementos de G_{63}

$GR^*(4, 6)$	$(\alpha^0\alpha^1\alpha^2\alpha^3\alpha^4\alpha^5)$	$GR^*(4, 6)$	$(\alpha^0\alpha^1\alpha^2\alpha^3\alpha^4\alpha^5)$
$(f^2)^1 = \beta^2$	(001000)	$(f^2)^2 = \beta^4$	(000010)
$(f^2)^3 = \beta^6$	(330330)	$(f^2)^4 = \beta^8$	(113013)
$(f^2)^5 = \beta^{10}$	(302031)	$(f^2)^6 = \beta^{12}$	(102123)
$(f^2)^7 = \beta^{14}$	(232212)	$(f^2)^8 = \beta^{16}$	(310230)
$(f^2)^9 = \beta^{18}$	(113212)	$(f^2)^{10} = \beta^{20}$	(313000)
$(f^2)^{11} = \beta^{22}$	(003130)	$(f^2)^{12} = \beta^{24}$	(110101)
$(f^2)^{13} = \beta^{26}$	(030130)	$(f^2)^{14} = \beta^{28}$	(110011)
$(f^2)^{15} = \beta^{30}$	(320023)	$(f^2)^{16} = \beta^{32}$	(230031)
$(f^2)^{17} = \beta^{34}$	(101003)	$(f^2)^{18} = \beta^{36}$	(012021)

$(f^2)^{19} = \beta^{38}$	(213333)	$(f^2)^{20} = \beta^{40}$	(123210)
$(f^2)^{21} = \beta^{42}$	(331122)	$(f^2)^{22} = \beta^{44}$	(201113)
$(f^2)^{23} = \beta^{46}$	(303312)	$(f^2)^{24} = \beta^{48}$	(311301)
$(f^2)^{25} = \beta^{50}$	(032102)	$(f^2)^{26} = \beta^{52}$	(022303)
$(f^2)^{27} = \beta^{54}$	(011230)	$(f^2)^{28} = \beta^{56}$	(110222)
$(f^2)^{29} = \beta^{58}$	(203300)	$(f^2)^{30} = \beta^{60}$	(002033)
$(f^2)^{31} = \beta^{62}$	(121101)	$(f^2)^{32} = \beta^{64}$	(030200)
$(f^2)^{33} = \beta^{66}$	(000302)	$(f^2)^{34} = \beta^{68}$	(022021)
$(f^2)^{35} = \beta^{70}$	(213033)	$(f^2)^{36} = \beta^{72}$	(123211)
$(f^2)^{37} = \beta^{74}$	(320111)	$(f^2)^{38} = \beta^{76}$	(322120)
$(f^2)^{39} = \beta^{78}$	(223001)	$(f^2)^{40} = \beta^{80}$	(031223)
$(f^2)^{41} = \beta^{82}$	(231103)	$(f^2)^{42} = \beta^{84}$	(013322)
$(f^2)^{43} = \beta^{86}$	(202331)	$(f^2)^{44} = \beta^{88}$	(101122)
$(f^2)^{45} = \beta^{90}$	(203213)	$(f^2)^{46} = \beta^{92}$	(303333)
$(f^2)^{47} = \beta^{94}$	(120110)	$(f^2)^{48} = \beta^{96}$	(331131)
$(f^2)^{49} = \beta^{98}$	(102010)	$(f^2)^{50} = \beta^{100}$	(331310)
$(f^2)^{51} = \beta^{102}$	(333203)	$(f^2)^{52} = \beta^{104}$	(010303)
$(f^2)^{53} = \beta^{106}$	(011110)	$(f^2)^{54} = \beta^{108}$	(330001)
$(f^2)^{55} = \beta^{110}$	(032333)	$(f^2)^{56} = \beta^{112}$	(121000)
$(f^2)^{57} = \beta^{114}$	(001210)	$(f^2)^{58} = \beta^{116}$	(330302)
$(f^2)^{59} = \beta^{118}$	(021321)	$(f^2)^{60} = \beta^{120}$	(213022)
$(f^2)^{61} = \beta^{122}$	(200332)	$(f^2)^{62} = \beta^{124}$	(130131)
$(f^2)^{63} = \beta^{126}$	(100000)		

- **Passo 5: encontrar o polinômio gerador $g(x)$ e em seguida o polinômio de verificação de paridade $h(x)$ do código BCH. Ao determinar esses polinômios é possível obter a matriz geradora e a matriz de verificação de paridade**

O polinômio gerador do código BCH de comprimento n tem como raízes os elementos $\{(\beta^i), (\beta^i)^p, \dots, (\beta^i)^{p^{r-1}(\bmod n)}\}$, e é representado por:

$$g(x) = mmc(M_1(x), M_2(x), \dots, M_{2t}(x)),$$

em que $M_i(x)$ é o polinômio minimal associado ao elemento primitivo β^i , $i = 1, 2, \dots, 2t$, e mmc denota o mínimo múltiplo comum.

O polinômio $g(x)$ é calculado por meio de três etapas:

1º) Cálculo das raízes dos polinômios minimais:

Para cada polinômio minimal $M_i(x) = M_i$, com $i = 1, 2, \dots, 62$, temos:

$$\begin{aligned} M_1(x) &= \{(\beta^1)(\beta^1)^2 \dots, (\beta^1)^{2^{6-1(\bmod 63)}}\} \rightarrow M_1(x) = \{(\beta), (\beta^2), (\beta^4), (\beta^8), (\beta^{16}), (\beta^{32})\}, \\ M_2(x) &= \{(\beta^2), (\beta^2)^2, \dots, (\beta^2)^{2^{6-1(\bmod 63)}}\} \rightarrow M_2(x) = \{(\beta^2), (\beta^4), (\beta^8), (\beta^{16}), (\beta^{32}), (\beta)\}, \\ &\vdots = \vdots \\ M_{62}(x) &= \{(\beta^{62}), (\beta^{62})^2, \dots, (\beta^{62})^{2^{6-1(\bmod 63)}}\} \rightarrow \\ &M_{62}(x) = \{(\beta^{62}), (\beta^{61}), (\beta^{59}), (\beta^{55}), (\beta^{47}), (\beta^{31})\}. \end{aligned}$$

2º) Cálculo dos polinômios minimais $M_i(x)$, para todo $i = 1, 2, \dots, 62$:

Os polinômios minimais $M_i(x)$, para todo $i = 1, 2, \dots, 62$, são calculados realizando-se o produtório das suas respectivas raízes minimais. Desse modo, o polinômio minimal $M_1(x)$ é obtido da seguinte maneira:

$$\begin{aligned} M_1(x) &= \{(x - \beta)(x - \beta^2)(x - \beta^4)(x - \beta^8)(x - \beta^{16})(x - \beta^{32})\} \\ M_1(x) &= x^6 + 2x^3 + 3x + 1. \end{aligned}$$

De maneira análoga, os outros polinômios minimais são calculados.

3º) Cálculo dos polinômios geradores para $1 \leq t \leq 31$:

O polinômio gerador $g(x)$ para cada valor de t é dado pelo mínimo múltiplo comum formado pelos polinômios minimais diferentes entre si. Neste trabalho, a título de exemplo, considera-se que a distância mínima do código seja $d_{\min} = 3$, então o polinômio gerador do código é dado por $g(x) = x^6 + 2x^5 + x^4 + x^3 + 3x^1 + 1$.

Considerando o polinômio gerador $g_1(x) = x^6 + 2x^5 + x^4 + x^3 + 3x^1 + 1$, realizando os deslocamentos dos coeficientes do polinômio $g(x)$ da esquerda para a direita, obtém-se a matriz geradora G , com dimensão 57×63 .

Já para a obtenção do polinômio $h(x)$ tem-se:

$$h(x) = \frac{x^n - 1}{g(x)} = \frac{x^{63} - 1}{x^6 + 2x^5 + x^4 + x^3 + 3x^1 + 1}$$

$$h(x) = 1x^{57} + 2x^{56} + 3x^{55} + 3x^{54} + 1x^{53} + 1x^{52} + 3x^{51} + 1x^{50} + 2x^{49} + 2x^{48} + 1x^{47} + 3x^{45} + 2x^{44} + 1x^{43} + 2x^{40} + 3x^{39} + 3x^{38} + 2x^{36} + 3x^{35} + 1x^{34} + 1x^{33} + 3x^{32} + 2x^{31} + 1x^{30} + 3x^{29} + 3x^{28} + 1x^{26} + 2x^{25} + 1x^{24} + 3x^{23} + 1x^{21} + 2x^{19} + 1x^{18} + 1x^{17} + 1x^{15} + 3x^{14} + 1x^{10} + 3x^7 + 2x^6 + 2x^5 + 2x^4 + 3x^2 + 3x^1 + 3.$$

Pode-se agora construir a matriz verificação de paridade H por meio do polinômio $h(x)$. A primeira linha da matriz será constituída pelos coeficientes do referido polinômio. Para a composição das linhas subsequentes deve-se fazer um deslocamento da direita para esquerda dos elementos formadores da primeira linha.

- **Passo 6: converter a sequência original composta dos elementos do grupo N na sequência correspondente com elementos em \mathbb{Z}_4**

Neste passo o objetivo é verificar se o código BCH sobre anel é capaz de reproduzir a sequência de DNA Homo sapiens [P30049 - 12587], ATP synthase subunit delta (ATP5F1D), mitochondrial precursor, inner membrane 6 com 63 nucleotídeos - GI número 513.

No Passo 1 especificou-se o alfabeto utilizado no código, em que foi determinada uma analogia entre o alfabeto 4-ário do conjunto de nucleotídeos denotado por $N = \{A, C, G, T/U\}$ e o alfabeto 4-ário matemático, denotado por $\mathbb{Z}_4 = \{0, 1, 2, 3\}$. Essa analogia é necessária para realizar o mapeamento entre o conjunto $N = \{A, C, G, T/U\}$ e o conjunto $\mathbb{Z}_4 = \{0, 1, 2, 3\}$ ou vice-versa. No entanto, o melhor mapeamento entre esses dois conjuntos ainda é desconhecido, dessa maneira todas as possibilidades de associação dos elementos serão analisadas para verificar qual delas é a melhor.

Assim, a sequência de DNA Homo sapiens [P30049 - 12587], ATP synthase subunit delta (ATP5F1D), mitochondrial precursor, inner membrane 6 - GI número 513 será considerada como uma das 24 permutações entre $N \rightarrow \mathbb{Z}_4$.

Seja a sequência do *NCBI* igual a:

CTGCCCCGCCGCGCTGCTCCGCCGCCCGGACTTGGCCGCCTCGTCCGCCACGCCCGTGCCTAT

então obtemos a matriz P . As 24 linhas da matriz P correspondem às 24 permutações da sequência de DNA:



$$P = \begin{bmatrix} 132111211212132131121121112220133221121131231121101211123211303 \\ 123111311313123121131131113330122331131121321131101311132311202 \\ 231222122121231232212212221110233112212232132212202122213122303 \\ 213222322323213212232232223330211332232212312232202322231322101 \\ 312333233232312313323323332220311223323313213323303233321233101 \\ 321333133131321323313313331110322113313323123313303133312133202 \\ 032000200202032030020020002221033220020030230020010200023200313 \\ 023000300303023020030030003331022330030020320030010300032300212 \\ 23022202202030232202202220001233002202232032202212022203022313 \\ 203222322323203202232232223331200332232202302232212322230322010 \\ 320333033030320323303303330001322003303323023303313033302033212 \\ 302333233232302303323323332221300223323303203323313233320233010 \\ 031000100101031030010010001112033110010030130010020100013100323 \\ 013000300303013010030030003332011330030010310030020300031300121 \\ 130111011010130131101101110002133001101131031101121011103011323 \\ 103111311313103101131131113332100331131101301131121311130311020 \\ 310333033030310313303303330002311003303313013303323033301033121 \\ 301333133131301303313313331112300113313303103313323133310133020 \\ 021000100101021020010010001113022110010020120010030100012100232 \\ 012000200202012010020020002223011220020010210020030200021200131 \\ 120111011010120121101101110003122001101121021101131011102011232 \\ 102111211212102101121121112223100221121101201121131211120211030 \\ 210222022020210212202202220003211002202212012202232022201022131 \\ 201222122121201202212212221113200112212202102212232122210122030 \end{bmatrix}$$

- **Passo 7: verificar pela relação $v \cdot H^T$ se cada uma das seqüências convertidas é uma palavra-código do código**

No Passo 6, a seqüência de DNA do NCBI foi convertida nas 24 permutações possíveis. No entanto, para verificar se cada uma dessas permutações são de fato uma palavra-código utiliza-se a relação $v \cdot H^T = 0$, em que v é a possível palavra-código e H^T é a transposta da matriz de verificação de paridade obtida no Passo 5. Desse modo, analisa-se as palavras-código cujo nucleotídeos de diferença da seqüência de DNA do NCBI é nula. Além disso, nessa etapa são analisadas quais seqüências de DNA apresentando os padrões de erros de 1 e 2 nucleotídeos de diferença da seqüência de DNA do NCBI são palavras-código dos códigos (n, k, d_H) , da seguinte maneira:

a) para analisarmos as seqüências de DNA com até 1 nucleotídeo de diferença da seqüência de DNA do *NCBI*, cuja notação é $D(a, b) = 1$, são consideradas as 4356 possíveis palavras-código para cada seqüência de DNA analisada e então usamos a relação $v \cdot H^T = 0$. As palavras-código encontradas são armazenadas. Essa quantidade de

possíveis palavras-código foi obtida considerando as sequências de DNA diferindo em um nucleotídeo, analisando as 3 outras possibilidades de nucleotídeos em cada posição na sequência para as 24 permutações.

b) já para analisarmos as sequências de DNA com até 2 nucleotídeos de diferença da sequência de DNA do *NCBI*, cuja notação é $D(a, b) = 2$, são consideradas todas as combinações 2 a 2 dos n nucleotídeos de comprimento da sequência para as 24 permutações, resultando em 17577 possíveis palavras-código para cada sequência de DNA analisada e, então usamos a relação $v \cdot H^T = 0$. As palavras-código encontradas são armazenadas.

- **Passo 8: converter cada sequência armazenada no Passo 5**

Neste passo, todas as palavras-código armazenadas no passo anterior estão rotuladas na forma alfabeto do código $\mathbb{Z}_4 = \{0, 1, 2, 3\}$, e serão convertidas em nucleotídeos usando o alfabeto do código genético $N = \{A, C, G, T\}$. Em seguida, as palavras-código são comparadas, uma a uma, com a sequência de DNA original, mostrando onde os nucleotídeos diferem.

- **Passo 9: voltar para o Passo 4 e determinar outro $g(x)$. Repetir os Passos 5 ao 7**

Neste passo, determina-se outro valor da distância mínima, e usa-se o mesmo procedimento, apresentado no Passo 5, para calcular o polinômio gerador relativo a esta distância. E depois, repete-se os Passos 5 ao 7 para o novo $g(x)$, até que se esgote todas as possibilidades de $g(x)$.

- **Passo 10: fim**

Como resultado das simulações da enzima ATP6 foram obtidas 8 palavras-código. Essas palavras-código são diferentes em termo do alfabeto do código $\mathbb{Z}_4 = \{0, 1, 2, 3\}$, no entanto quando as rotulamos para o alfabeto do código genético $N = \{A, C, G, T/U\}$ elas são iguais, resultando em única sequência de DNA. Como as sequências de DNA reproduzidas são iguais, é suficiente analisar apenas um dos casos (Figura 1). Vale destacar que a diferença do dígito nessas palavras-código sempre ocorre na mesma posição.

A Figura 1, mostra a sequência de direcionamento da enzima ATP6 reproduzida pelo código BCH sobre $GR(4, 6)$, através do polinômio primitivo $p(x) = x^6 + x^5 + 1$ e do polinômio gerador $g(x) = x^6 + 3x^5 + 2x^3 + 1$.

Abreviações: Oaa = aminoácido original; Ont = nucleotídeos originais; OLb = rotulagem original; GLb = rotulagem gerada; Gnt = nucleotídeos gerados; Goo = aminoácido gerado.

Figura 1: Sequência gerada com $n = 63$ nucleotídeos

Palavra-código: 3 de rotulamento C :: (T = 0, C = 1, G = 2, A = 3)

Oaa:

.L..P..A..A..L..L..R..R..P..G..L..G..R..L..V..R..H..A..R..A..Y.

Ont:

CTGCCCGCCGCGCTGCTCCGCCGCCCGGGACTTGGCCGCCTCGTCCGCCACGCCCGTGCCTAT

OLb:

102111211212102101121121112223100221121101201121131211120211030

GLb:

10211121121210210112112111222310022112110120112113211120211030

Gnt:

CTGCCCGCCGCGCTGCTCCGCCGCCCGGGACTTGGCCGCCTCGTCCGCCAGGCCCGTGCCTAT

Goa:

.L..P..A..A..L..L..R..R..P..G..L..G..R..L..V..R..Q..A..R..A..Y.

Observe que, na posição da trinca 17 houve uma troca do nucleotídeo citosina (CAC) por um nucleotídeo guanina (CAG), ocasionando a troca de aminoácido nesta posição, sendo a histidina (H) substituída pela glutamina (Q).

Ao realizar um estudo mutacional do resultado obtido pela simulação foi possível verificar que a substituição do nucleotídeo gerada pelo algoritmo está relacionada com uma mutação atípica de característica maléfica encontrada na enzima mitocondrial ATP6, que acarreta a doença de NARP. Esta mutação foi verificada e comprovada por meio de análises laboratoriais a partir de 2013, em [2] e [6], até então era conhecido que apenas as mutações que levam a substituição do nucleotídeo timina pelo nucleotídeo guanina ou o nucleotídeo timina pelo nucleotídeo citosina que poderiam estar relacionados com patologias clínicas ocasionadas por mutações no ATP6.

5 Considerações Finais

Neste estudo foi analisado um caso particular, mas que amparado por estudos biológicos de [2] e [6], indicam possíveis implicações do algoritmo de geração de proteínas no estudo de mutações genéticas. Resultados similares foram obtidos nos trabalhos de [8] e [3] em que foram analisadas 92 sequências de DNA relacionadas a vírus e bactérias. Dentre elas, foram estudadas uma variação da proteína H1N1 e três sequências referentes às proteínas p53 e BCRA1 relacionadas ao câncer, em que nessas sequências foi possível verificar trocas de nucleotídeos, no entanto, a interpretação mutacional dessas sequências foi deixada para trabalhos futuros. No estudo de [7], com o objetivo de verificar a apli-

cabilidade do algoritmo de geração de proteínas para a biologia, foi gerada e analisada a sequência de DNA relacionada ao éxon 14 do gene BRCA1, em que mutações nesse gene estão conectados com o câncer de mama. Ao realizar o estudo mutacional foi possível identificar todas as mutações nonsense e missense do éxon 14 do gene BRCA1.

Assim, os resultados obtidos neste estudo e nos trabalhos de [8], [3] e [7], evidenciam que os códigos corretores de erros são eficientes para localizar onde ocorreu troca de nucleotídeos em uma sequência de DNA. Portanto, futuramente o algoritmo de geração de proteínas poderá proporcionar ganhos no estudo de patologias clínicas, pois uma vez observada e determinada a estrutura algébricas nas sequências de DNA é possível realizar estudos mutacionais e filogenéticos dessas sequências.

Agradecimentos

À Pro-Reitoria de Pesquisa e Pós-Graduação, à UNIFAL-MG e à FAPEMIG.

Referências

- [1] COSTELO JR, D. J.; LIN, S. *et al.* **Error control coding: fundamentals and applications.**, Prentice Hall, Englewood Cliffs, 1983.
- [2] DUNO, M. *et al.* A novel mitochondrial mutation m.8989 G>C associated with neuropathy, ataxia, retinitis pigmentosa - The NARP syndrome, **Gene**, v. 515, p. 372-375, 2013.
- [3] FARIA, L. C. B. **Existências de Códigos Corretores de Erros e Protocolos de Comunicação em Sequências de DNA.** 2011. Tese (Doutorado em Engenharia Elétrica) - Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, Campinas, 2011.
- [4] GRIFFITHS, A. J. F. *et al.* **Introdução à genética**, Guanabara Koogan, Rio de Janeiro, 2006.
- [5] INTERLANDO, J. C. **Uma contribuição à construção e decodificação de códigos lineares sobre grupos abelianos via concatenação de códigos sobre anéis de inteiros residuais.** 1994. Tese (Doutorado em Engenharia Elétrica) - Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, Campinas, 1994.



- [6] KOMULAINEN, T. *et al.* A novel mutation m.8561C>G in MT-ATP6/8 causing a mitochondrial syndrome with ataxia, peripheral neuropathy, diabetes mellitus, and hypergonadotropic hypogonadism, **J Neurol**, v. 263, p. 2188-2195, 2016.
- [7] PEREIRA, D. G. **Uma abordagem computacional para a análise de sequências de DNA por meio dos códigos corretores de Erros.** 2014. Dissertação (Mestrado em Engenharia Elétrica) - Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, Campinas, 2014.
- [8] ROCHA, A. S. L. **Modelo de sistema de Comunicações digital para o mecanismo de importação de proteínas mitocondriais através de códigos corretores de erros.** 2010. Tese (Doutorado em Engenharia Elétrica) - Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, Campinas, 2010.
- [9] SENIOR, A. E.; WEBER, J. ATP synthesis driven by proton transport in F1F0-ATP synthase, **FEB SLetters**, v. 545, p. 61-70, 2003.

Submetido em set. 2021
Aceito em fev. 2022