Otimização de periodicidades fracas em genomas utilizando transformadas de Fourier e algoritmos genéticos

Dissertação de mestrado

MÍRIAM CELI DE SOUZA NUNES
GERALD WEBER (DF/UFMG, ORIENTADOR)

Núcleo de Pesquisas em Ciências Biológicas (NUPEB)
Pós-graduação em Biotecnologia
Área de concentração: Genômica e Proteômica
Universidade Federal de Ouro Preto

Ouro Preto, junho de 2013

Otimização de periodicidades fracas em genomas utilizando transformadas de Fourier e algoritmos genéticos

MÍRIAM CELI DE SOUZA NUNES
GERALD WEBER (DF/UFMG, ORIENTADOR)

DISSERTAÇÃO DE MESTRADO PELA UNIVERSIDADE FEDERAL DE OURO PRETO COMO PARTE DOS REQUISITOS BÁSICOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM BIOTECNOLOGIA. ÁREA DE CONCENTRAÇÃO: GENÔMICA E PROTEÔMICA.

INSTITUTO DE CIÊNCIAS EXATAS E BIOLÓGICAS - ICEB
UNIVERSIDADE FEDERAL DE OURO PRETO

Ouro Preto, junho de 2013

Dedicatória

Aos meus pais, Emanuel e Mercedes, pela educação primorosa recebida em casa, pelos valores essenciais ensinados desde o berço e pelo imenso amor e dedicação, guias fundamentais para meu sucesso. Ao meu irmão Daniel e à Tania, pelo cuidado, carinho e amizade.

Agradecimentos

Agradeço com carinho:

- Primeiramente ao Gerald, por todo o aprendizado, pela valiosa amizade e confiança, pela incrível paciência e tranquilidade, pela alegria contagiante e por todo o cuidado quase paternal que sempre me proporcionou.
- Aos colegas do GBC, especialmente ao Alcides, pelas inúmeras risadas!
- À Tauanne, pela amizade, alto astral e ótima companhia.
- Ao Rost Lab, em Munique, pela hospitalidade e trabalho enriquecedor.
- À Elizabeth Wanner, pelas importantes discussões nos estágios iniciais deste projeto.
- À UFOP e ao Programa de pós-graduação em Biotecnologia pela oportunidade.
- Ao CNPq, pela bolsa de mestrado, e ao INCT, CAPES e FAPEMIG pelo incentivo financeiro.

"Estude a si mesmo, observando que o autoconhecimento traz humildade e sem humildade é impossível ser feliz."

Allan Kardec

Sumário

Lista de Figuras **VI**

Lista de Tabelas **VIII**

Resumo **IX**

Abstract **X**

1 Introdução **1**

1.1 Bioinformática 1

1.2 Objeto de estudo: A molécula de DNA 2

1.2.1 Repetições do DNA 4

1.2.2 Métodos para análise de periodicidades 6

1.2.3 Métodos baseados na DFT 7

1.3 Problema da periodicidade nucleossomal 9

1.4 Detectando periodicidades fracas no DNA 11

1.4.1 Convertendo letras em números 11

1.4.2 Escolhendo o melhor mapeamento numérico 12

2 Motivação e objetivos **16**

3 Metodologia **17**

3.1 Dados utilizados nos testes 17

3.1.1 Sequências de DNA do *P. falciparum* e *D. melanogaster* 17

3.1.2 Sequências promotoras de *H. sapiens* 17

3.2 Aplicação da Transformada de Fourier Discreta (DFT) 18

3.3 Mapeamento indicador binário 18

3.4 Mapeamento binário de dinucleotídeos 20

3.5 Mapeamento de dinucleotídeos (*DM*) 21

3.6 Algoritmo genético (AG) 23

3.6.1 Implementação 23

3.6.2 Parâmetros de entrada e cálculo do *fitness* 23

3.6.3 Critério de parada 28

3.7 Passo-a-passo do AG 28

3.7.1 Construção do FPS 29

4 Resultados e Discussão **30**
4.1 Analisando a eficiência do método 30
4.1.1 Variação dos valores dos dinucleotídeos 30
4.1.2 Variação dos valores de *fitness* 31
4.2 Otimizando periodicidades fracas 33
4.2.1 Buscando pela periodicidade $p = 3$ 33
4.2.2 Revelando a periodicidade $p = 10$ 38
4.3 Teste da produção de falsos positivos 45
4.4 Resolvendo o problema da periodicidade nucleossomal 46

5 Conclusão **51**

6 Perspectivas Futuras **52**

7 Apêndice **53**
7.1 Algoritmos em Perl 53
7.1.1 Fourier binário dinucleotídeo 53
7.1.2 Algoritmo Genético + Fourier 54
7.1.3 Construção dos espectros de potência 57
7.2 Espectros completos 59

Referências **62**

Lista de Figuras

1 Esquema da composição estrutural dos nucleotídeos 2
2 Estrutura esquemática do DNA. 3
3 Pico na frequência $f = 1/3$, referente à região codificante no trecho de DNA do parasito *Plasmodium falciparum* 11
4 Representação hipotética de uma população dentro do algoritmo genético. 13
5 Seleção por torneio. 14
6 Processo de *crossover*. 14
7 Mecanismo de mutação. 15
8 Esquema demonstrando a seleção por torneio. 26
9 Flutuação dos valores dos dinucleotídeos em 10 repetições num trecho do DNA da *Drosophila melanogaster*. 30
10 Variação dos valores dos dinucleotídeos retornados pelo nosso AG numa repetição de 10 vezes, utilizando um trecho de DNA aleatorizado da *D. melanogaster*. 31
11 Flutuação dos valores de *fitness* ao para a maximização dos períodos 3 e 10 no *Plasmodium falciparum*, utilizando sequências originais e aleatorizadas. 32
12 Otimização da periodicidade $p = 3$ num trecho de 1000 nucleotídeos da *D. melanogaster*. 34
13 Valores dos dinucleotídeos para o mapeamento *DM*, responsáveis pela maximização da frequência $f = 1/3$ para a *D. melanogaster*. 37
14 Variação dos valores reais dos dinucleotídeos com o mapeamento DM^+ para a *D. melanogaster*. 37
15 Busca pela periodicidade de 3 nucleotídeos em um trecho de DNA do genoma do *P. falciparum* 38
16 Exemplo de periodicidade $p = 3$ inexistente em um trecho de DNA do *P. falciparum* 38
17 Valores dos dinucleotídeos para mapeamento *DM* para a maximização da periodicidade $p = 3$ no *P. falciparum*. 39
18 Valores dos dinucleotídeos do mapeamento DM^+ para a maximização da periodicidade $p = 3$ no *P. falciparum*. 39
19 Otimização do período $p = 10$ na *D. melanogaster*. 40
20 Variação dos valores dos dinucleotídeos do mapeamento *DM* para a maximização da periodicidade $p = 10$ em *D. melanogaster*. 41

21 Valores do mapeamento DM^+ retornados para a otimização do período $p = 10$ em *D. melanogaster*. 42
22 Maximização do período $p = 10$ num trecho do DNA do parasito *P. falciparum* 42
23 Variação dos valores dos dinucleotídeos retornados para o mapeamento *DM* no trecho de DNA do *P. falciparum*. 43
24 Flutuação dos valores dos dinucleotídeos do mapeamento DM^+ para a otimização do período $p = 10$ do *P. falciparum*. 44
25 Periodicidade inesperada encontrada num trecho de DNA do *P. falciparum*. 44
26 Periodicidade $p = 10$ inexistente no trecho de DNA da *D. melanogaster*. . 45
27 Histograma mostrando a distribuição dos valores de *fitness* para a periodicidade $p = 10$ nos promotores humanos. 48
28 Regressão linear mostrando a correlação entre os diferentes mapeamentos, DBI, *DM* e DM^+. 49
29 Espectro completo para a frequência f=1/3 da *Drosophila melanogaster*. . 59
30 Espectro completo para a frequência f=1/3 do *Plasmodium falciparum*. . . 59
31 Espectro completo para a frequência f=1/10 da *Drosophila melanogaster*. 60
32 Espectro completo para a frequência f=1/10 do *Plasmodium falciparum*. . 60
33 Espectro completo para a frequência f=1/3 onde a periodicidade é inexistente em *Plasmodium falciparum*. 61
34 Espectro completo para a frequência f=1/10 onde a periodicidade é inexistente em *Drosophila melanogaster* 61

Lista de Tabelas

1 Mapeamento indicador binário para uma sequência de DNA hipotética. . 19
2 Mapeamento indicador binário de dinucleotídeos para uma sequência de DNA hipotética. 21
3 Mapeamento de dinucleotídeos com valores positivos e negativos para uma sequência de DNA hipotética. 22
4 Mapeamento de dinucleotídeos com valores apenas positivos. 23
5 Distribuição dos valores de *fitness* dos promotores humanos para montagem do histograma. 48

Resumo

Transformadas de Fourier discretas e os seus espectros de potência associados são utilizados em biologia molecular e na bioinformática para a detecção de periodicidades e regiões codificantes no DNA. Tipicamente, o genoma primeiramente é mapeado para uma série numérica, então é feita sua transformada de Fourier, que será monitorada em busca de periodicidades biologicamente relevantes. A detecção de uma determinada periodicidade é criticamente dependente da forma como o genoma foi convertido numa sequência numérica. Uma vez que existem inúmeras maneiras de se mapear uma sequência de DNA, periodicidades biologicamente importantes podem passar despercebidas. Aqui apresentamos um método que utiliza um algoritmo genético para otimizar o mapeamento numérico que maximizará um pico em dada frequência do espectro de potências de Fourier. Nós mostramos que o método tem a capacidade de detectar periodicidades fracas que são normalmente perdidas com esquemas de mapeamento tradicionais, e dessa forma, aumenta em muito a sensibilidade de técnicas baseadas na transformada de Fourier discreta. Para exemplificar o uso deste novo método, nós o aplicamos para encontrar as periodicidades de 3 e 10 nucleotídeos em trechos dos genomas do *Plasmodium falciparum* e *Drosophila melanogaster*, além de aplicá-lo também em sequências promotoras de *Homo sapiens*, que foram recentemente analisadas para a detecção do período de 10 nucleotídeos. Trabalhos anteriores publicaram afirmações conflitantes sobre a presença desta periodicidade em torno dos sítios de iniciação de transcrição (TSS) de promotores humanos. Nós mostramos que esta periodicidade está realmente presente nessas sequências e também que o nosso método é robusto e eficiente para descobrir periodicidades fracas em genomas.

Abstract

Discrete Fourier transforms and their associated power spectra are used in molecular biology and bioinformatics for detecting periodicities and protein-coding genes. Typically, the genome is mapped into a numerical series which is Fourier-transformed and monitored for biologically relevant periodicities. The detection of a given periodicity is critically dependent on how the genome was converted into a numerical sequence. Since there are numerous ways of mapping the sequence, biologically important periodicities may go undetected. Here we present a method which employs a genetic algorithm to detect periodicities by optimising a given frequency peak of the Fourier power spectrum. We show that the method has the capability of detecting weak periodicities which are ordinarily missed with traditional mapping schemes, therefore greatly enhancing the sensitivity of discrete-Fourier-transform based techniques. To exemplify the use of this new method we apply it to find periodicities of 3 and 10 nucleotides on the *Plasmodium falciparum* and *Drosophila melanogaster* genomes and *Homo sapiens* promoter sequences, which were recently analysed for the detection of period 10. Previous works made confliting assertions about the presence of this periodicity around human TSS. We show that this periodicity is indeed present in these sequences and show that our method is robust and efficient to uncover weak periodicities in genomes.

1 Introdução

1.1 Bioinformática

A bioinformática é uma ciência relativamente nova que utiliza técnicas computacionais para analisar sistemas biológicos em busca da solução para um determinado problema. Muitas vezes, os problemas levados para análise pelo computador são aqueles difíceis de serem realizados na bancada dos laboratórios, envolvendo técnicas tradicionais de biologia molecular. Geralmente os trabalhos de bancada envolvendo manipulação do DNA despendem muito tempo, recursos financeiros e podem não gerar o resultado esperado. Simulações computacionais do DNA demandam poucos recursos, tanto financeiro quanto de pessoal, e se bem executadas, as análises retornam bons resultados. Portanto, em alguns casos é mais sensato fazer o uso da bioinformática como garantia em análises experimentais antes de levar os experimentos para a bancada, evitando assim gastos desnecessários. Entretanto, na maioria dos casos a bioinformática é utilizada na fase posterior à bancada. Por exemplo, nos experimentos de sequenciamento de material genético, os recursos computacionais da bioinformática são utilizados depois de obter os resultados, a fim de interpretar o material genético, descobrir o posicionamento dos genes e fazer as anotações do DNA.

A grande riqueza da bioinformática está na integração com diversas áreas do conhecimento, como matemática, física, química. Essa integração permite que os problemas em questão sejam abordados por uma ótica ampla, fazendo com que os resultados sejam mais confiáveis e robustos.

Ótimos resultados provenientes do uso da bioinformática garantem que a área se expanda cada dia mais. O surgimento de novas empresas prestadoras de serviços em bioinformática é cada vez mais frequente, notavelmente nas áreas da genética e da biologia molecular. A procura por esses serviços cresce e com isso vem a questão da qualidade dos serviços prestados. Este é um fator preocupante que pode fazer com que, muitas vezes, o produto final não esteja de acordo com o requerido inicialmente.

Técnicas consagradas no meio acadêmico podem eventualmente mostrar-se ineficientes e muitas vezes não são atualizadas pelos criadores originais. Os resultados, portanto, podem não ser confiáveis.

Para ilustrar, considere o exemplo do trabalho de Stewart e McLachlan *et al*, de 1975 [1], onde a sequência da proteína tropomiosina é estudada. Esta proteína, em conjunto com a troponina, está presente em filamentos de actina, um dos componentes responsáveis pela contração muscular. Para descobrir as repetições presentes na sequência protéica da tropomiosina, foi aplicada à sequência uma técnica chamada transformada

de Fourier discreta (DFT). Nos resultados é relatada a existência de periodicidades fracionárias, $f = 7/2$ e $f = 7/3$, que foram relacionadas com o enovelamento da alfa-hélice de proteínas.

Em um trabalho mais recente [2], o mesmo autor aplica uma técnica chamada Transformada de Fourier Multicanal, onde em uma sequência protéica, cada aminoácido corresponde a um canal. Foi utilizada sequência protéica da miosina, que trabalha em conjunto com a actina para produzir a contração muscular, referente ao organismo modelo *Dictyostelium discoideum*, eucarioto do filo Mycetozoa. Após a análise do cálculo de Fourier, o autor relata fortes periodicidades múltiplas de 28, que incluem as frequências $f = 7/2$ e $f = 7/3$, encontradas nos trabalhos anteriores na sequência da proteína tropomiosina.

Estes trabalhos foram publicados em revistas de grande renome, como Nature e *Journal of Molecular Biology*. Entretanto, recentemente nós refizemos estas análises [3] e chegamos à conclusão que as periodicidades encontradas pelo autor são subprodutos da frequência original $f = 1/28$, presente na sequência. Isso acontece devido a uma limitação da DFT, que causa harmônicos de frequência dependendo da periodicidade presente na sequência. Por isso, é importante que se conheça bem as limitações dos métodos que se pretende utilizar, a fim de evitar resultados duvidosos.

1.2 Objeto de estudo: A molécula de DNA

O DNA é a molécula responsável pela transmissão da informação genética entre todos os organismos. Em 1953, Watson e Crick descobriram a forma helicoidal da molécula de DNA [4], famoso trabalho que contribuiu para várias áreas da ciência desde então. As fitas de DNA são constituídas pelos **nucleotídeos**, que são moléculas formadas pela junção de uma pentose, um fosfato e uma base nitrogenada, como mostra o esquema da figura 1.

Existem quatro tipos de bases nitrogenadas: adenina (A), guanina (G), citosina (C) e timina (T). Através de ligações de hidrogênio, A se liga com T com duas ligações, C se

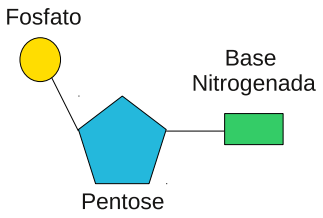

Figura 1
Esquema da composição dos nucleotídeos, moléculas que constituem o esqueleto da molécula de DNA. Eles são compostos por um fosfato, ligado a uma pentose, que se associa a uma base nitrogenada, que pode ser adenina (A), citosina (C), guanina (G) ou timina (T).

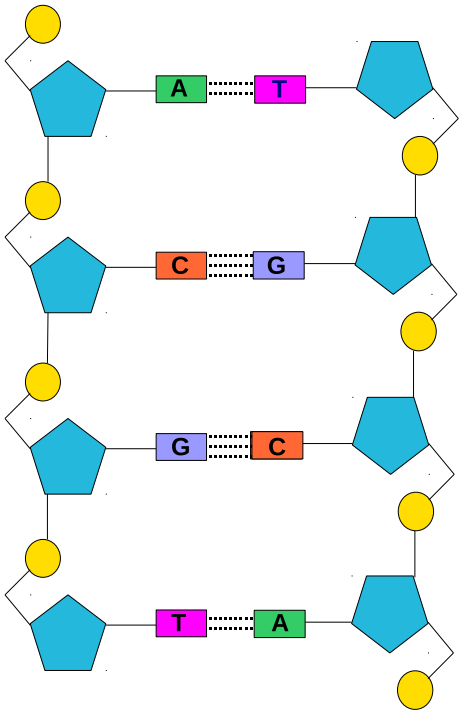

Figura 2
Estrutura esquemática do DNA. Através das ligações de hidrogênio, os nucleotídeos A se ligam com T fazendo 2 ligações, C se ligam com G fazendo 3 ligações. Com isso, as fitas de DNA se unem e formam a hélice dupla.

liga com G com três ligações [5]. Os nucleotídeos unem duas fitas de DNA, que toma a forma helicoidal. A figura 2 mostra um esquema bidimensional da estrutura do DNA.

O DNA é dividido em duas regiões principais: codificante e não-codificante. A região codificante de um organismo é composta pela soma de todas as sequências de DNA que codificam para proteínas. A região não-codificante de um organismo é a soma de todas as sequências do genoma que não codificam para proteínas. Nesta região encontram-se por exemplo os íntrons, regiões intergênicas, pseudogenes e elementos transponíveis (*transposons*). Entretanto, uma parte da região não-codificante é transcrita em moléculas de RNA não-codificantes funcionais, que exercem diversas funções regulatórias na célula. Pelo fato de uma grande parte da região não-codificante do DNA ainda não ter uma função biológica especificada, muitas vezes esta região foi referida como ”lixo genético”, conceito errôneo que hoje encontra-se ultrapassado. Recentemente, o projeto ENCODE (*Encyclopedia of DNA Elements*) [6] publicou um trabalho que afirma que mais de 76% do genoma humano é transcrito e sugere que 80% deste genoma pode ser funcional, num sentido bioquímico [7]. Os microRNAs são transcritos da região não-codificante do DNA e possuem diversas funções regulatórias, como por exemplo a regulação da tradução de

RNA mensageiro em proteínas [8], e por isso são um grande alvo das pesquisas em bioinformática na atualidade [9–13].

1.2.1 Repetições do DNA

Grande parte do genoma dos organismos dos diferentes reinos é composto por elementos repetitivos, ou periodicidades. Na nossa espécie, *Homo sapiens*, grande parte do genoma é constituído por repetições de vários tipos, sejam estas em regiões específicas dos cromossomos ou dispersas no DNA [14]. Existem periodicidades de vários tipos, e como exemplo, podemos citar uma repetição clássica, que são as chamadas repetições em *tandem*. Elas ocorrem em regiões específicas do DNA como telômeros e centrômeros, aparecendo lado a lado, no mesmo padrão e aproximadamente com a mesma composição nucleotídica [15]. Um outro tipo de repetições são aquelas que ocorrem intercaladas no DNA, e um tipo clássico são os *transposons*, ou elementos transponíveis.

Periodicidade 3 A periodicidade de 3 nucleotídeos foi descrita pela primeira vez por Shepherd *et al* em 1981 [16]. Neste trabalho, o autor utilizou primeiramente o DNA de cinco vírus que tinham sido recém-sequenciados. Ao analisar estas sequências de DNA, o autor relata a existência de padrões entre trios de nucleotídeos em regiões específicas dos diferentes genomas. Após testar quais tipos de trios compunham a região em questão, o autor nota que eles correspondiam exatamente à composição dos aminoácidos, que são também compostos por 3 nucleotídeos. Dessa forma, o autor sugere que a região onde são encontrados os padrões de 3 nucleotídeos corresponderiam à região codificante do genoma. Posteriormente, o autor testou sua descoberta em outros organismos, incluindo bactérias, leveduras e eucariotos, e descobriu que este padrão é uma característica geral para seres de todos os reinos. O autor também sugere que esta característica tem uma origem evolutiva comum a todos os organismos.

A partir do momento em que as regiões codificantes do DNA são constituídas na maior parte pelos *genes* do organismo, ela tornou-se alvo de incontáveis trabalhos no assunto. Várias ferramentas e métodos de detecção de regiões codificantes foram desenvolvidos, tanto para a detecção de regiões protéicas quanto de sítios funcionais de genes [17]. Disso surgiu o importante papel da bioinformática e biologia computacional para a área. Descobrir os genes, anotá-los e extrair informações da sequência genômica são processos fundamentais para entender e dar sentido ao DNA [18].

Um exemplo de aplicação interessante desenvolvida para separar regiões que codificam para proteínas é a apresentada pelo grupo de Anastassiou [5, 19]. Além de identificar as regiões codificantes, a ferramenta fornece um espectrograma de cores onde é possível

distinguir visualmente as regiões do DNA que codificam para proteínas. Inúmeras outras ferramentas estão constantemente sendo disponibilizadas para detectar regiões codificantes em genomas [20–29]. Ferramentas como estas ilustram a importância do estudo da região codificante, e até hoje é um assunto recorrente na literatura [30–33].

Periodicidade 10 A repetição de 10 nucleotídeos no DNA de organismos de todos os reinos já foi assunto para diversos trabalhos. Fukushima *et al* [22] analisa em seu trabalho diversos procariotos em busca de repetições. Uma das periodicidades encontradas é a de 10 nucleotídeos, que na literatura é conhecida por diversos motivos, um deles por ser o número de nucleotídeos responsáveis por formar a estrutura helicoidal do DNA [34] em procariotos. Neste mesmo trabalho, o autor relata a presença da periodicidade 10 em todos os eucariotos analisados. A repetição de 10 pares de base no genoma de arqueobactérias e 10.2 pares de base em eucariotos correspondem à uma volta na hélice dupla de DNA, enquanto que em eubactérias o pico ocorre em 11 pares de base [35]. Nos eucariotos, a curvatura da molécula de DNA causada pela periodicidade 10 ocorre basicamente devido à presença de dinucleotídeos de adenina (AA) [22].

Outro papel da periodicidade 10 largamente relatado na literatura é no posicionamento dos nucleossomos no DNA [36–38] no genoma de diversas espécies. Quando esta repetição é retirada de algumas porções do DNA, nota-se que os nucleossomos não encontram o seu local correto de posicionamento. Portanto, sugere-se que essa repetição exerça uma função essencial no DNA. Neste trabalho, realizamos vários testes em busca da periodicidade 10 nos eucariotos *Drosophila melanogaster*, mosca-das-frutas, e *Plasmodium falciparum*, parasito da malária, focando nas regiões do DNA onde é mais notável a função de posicionamento dos nucleossomos, como os sítios de iniciação da transcrição [39]. Na seção 1.3, página 9, explicamos detalhadamente sobre o papel dos nucleossomos e a importância da periodicidade 10 no DNA.

Outros exemplos Além das periodicidades de 3 e 10 nucleotídeos, existem outras periodicidades que são conhecidas por causarem doenças e distúrbios. Um exemplo é a Doença de Huntington [40], que é uma desordem autossômica neurodegenerativa resultante da expansão de várias repetições do trinucleotídeo CAG no gene HD. Geralmente, esta duplicação é resultado da duplicação do gene. Se o paciente apresentar 36 ou mais cópias do trinucleotídeo CAG no gene HD, ele será afetado pela doença [41]. Além disso, existem periodicidades no DNA que são relacionadas ao câncer [42], à síndrome do X-frágil [43] e a várias outras desordens genéticas [15, 44].

Periodicidades fracas O foco do nosso trabalho são as periodicidades fracas que ocorrem no DNA. Em nosso estudo, definimos periodicidades fracas como sendo aquelas que se repetem em baixa quantidade na sequência genômica, e por este motivo se tornam difíceis de serem detectadas. Uma periodicidade fraca pode ser de qualquer tipo ou composição nucleotídica. O critério que utilizamos para classificá-la como fracas é a frequência que ela se repete no genoma. Mais adiante, nós mostraremos alguns métodos de detecção de periodicidades dos mais variados tipos, e detalharemos o método escolhido neste trabalho, que é a transformada de Fourier discreta.

1.2.2 Métodos para análise de periodicidades

Existem inúmeros métodos para detecção de periodicidades descritos na literatura. Eles se diferem basicamente na escolha da ferramenta utilizada na detecção das repetições ou no método de interpretação do DNA. Abaixo listamos alguns exemplos para ilustrar a importância desse assunto.

Comparação de *strings* O *Tandem Repeats Finder* (TRF) [45] é um programa desenvolvido para analisar regiões de interesse no DNA que causam padrões de repetição. É um programa simples, que usa basicamente o método de comparação de *strings* (texto) para encontrar periodicidades. Como explicado anteriormente neste mesmo capítulo, repetições em *tandem* são duas ou mais cópias de sequências de nucleotídeos, aparecendo lado a lado e contendo aproximadamente a mesma composição. Diz-se "aproximadamente" porque os processos de mutação ao longo da evolução podem ter mudado a composição daquela sequência. Isso não prejudica a detecção da periodicidade pelo programa, sendo que ele detectará as repetições que apresentam diferenças em relação ao padrão e irá mostrá-las ao usuário. A detecção de periodicidades é feita da seguinte maneira: uma determinada sequência de nucleotídeos retirada do próprio DNA será comparada com todo o restante da sequência de DNA, e caso apareça uma repetição daquele trecho em outras posições, será computada a distância que as separa. Então, calcula-se a probabilidade da repetição encontrada ser de fato uma periodicidade, cálculo que leva em conta o número de vezes que aquele trecho se repete em toda a sequência. O *software* retorna páginas *web* com os resultados dispostos em forma de tabelas contendo análises detalhadas das regiões que contêm as periodicidades, com informações como o tamanho da periodicidade, o número de vezes que ela se repete, os nucleotídeos que compõem a repetição, entre outros. Este programa está disponível livremente para *download* via *internet* [46]. Como exemplo, em nosso trabalho anterior [3] utilizamos este *software* para descobrir quais eram as repetições que causavam harmônicos de frequência no DNA do

Plasmodium falciparum e em quais posições elas ocorriam. O *Tandem Repeats Finder* nos ajudou a descobrir que a periodicidade que buscávamos era composta de 21 nucleotídeos.

Transformada de Fourier Discreta (DFT) A técnica que escolhemos para lidar com as periodicidades neste trabalho é a DFT, uma ferramenta matemática que pode ser implementada em algoritmos computacionais [22]. Ela interpreta a sequência de DNA como um código, ou um sinal digital, e isso ajuda na identificação de padrões de repetição ou características peculiares da sequência. Na bioinformática, a DFT é aplicada em várias áreas, e na atualidade é um assunto recorrente na literatura. Como exemplos de áreas de uso da DFT, podemos citar a predição de genes [23] ou éxons [47], predição a estrutura de conformação de RNA [48], detecção de íntrons em sequências genômicas [49, 50] e também para prever a estrutura secundária de moléculas de RNA [51]. Outra aplicação interessante da DFT é descobrir regiões conservadas em organismos durante a evolução. Em bactérias, regiões genômicas com alta porcentagem de nucleotídeos A e T são de grande interesse por fornecerem informações sobre transferência lateral de genes. Estas sequências são identificadas pela DFT e os picos onde existem as repetições são facilmente reconhecidos no espectro de potências do genoma [35].

Transformada de *wavelet* Uma das várias técnicas que podem ser utilizadas para analisar periodicidades em genomas é a transformada de *wavelet*. Esta técnica detecta singularidades nas sequências de DNA mesmo estando mascaradas [52–56]. A transformada de *wavelet* é uma transformada de Fourier com informações de localidade. Esta técnica é bastante avançada, e como não é o foco deste trabalho, será citada apenas a título de curiosidade. Existem também trabalhos que utilizam técnicas mistas, com o objetivo de conseguir melhores resultados. No trabalho de Epps *et al* [57] é apresentada uma nova técnica que utiliza transformada de *wavelet* e transformada de Fourier discreta para a caracterização de periodicidades em sequências genômicas.

1.2.3 Métodos baseados na DFT

Por ser uma ferramenta simples e eficiente, a DFT foi utilizada para detectar periodicidades em vários trabalhos. Abaixo listamos alguns exemplos para ilustrar a importância da ferramenta na bioinformática.

Spectral Repeat Finder O *Spectral Repeat Finder* (SRF) foi desenvolvido pelo grupo de Sharma *et al* [14]. Primeiramente, o programa faz um escaneamento de todo o DNA em busca de repetições, através da análise dos espectros produzidos pela DFT.

Os espectros que contêm sinais fortes são potenciais candidatos a serem trechos de DNA que possuem uma periodicidade. O segundo passo é repetir o rastreamento em todo o genoma, desta vez utilizando uma janela deslizante procurando especialmente por aquela repetição candidata. Com isso, é possível encontrar as áreas do DNA que contribuem mais para aquela repetição, ou seja, onde ela é mais forte. Identificada a área de interesse, o próximo passo é extrai-la e analisar quais nucleotídeos compõem a repetição.

A grande vantagem desse programa é que ele pode ser aplicado em sequências de DNA recém-sequenciadas. Por extrair informações do DNA através da DFT independentemente de se ter informações prévias de anotação do DNA, pode-se dizer que o SRF é um bom método *ab initio* para descobrir periodicidades.

SWIFT O *software* SWIFT, *Sequence Wide Investigation with Fourier Transform* [58], foi desenvolvido para analisar sequências protéicas e identificar sua classe a partir de sequências genômicas ainda não anotadas. A importância desse *software* é notável: com o crescente número de genomas sequenciados, a demanda pela identificação de seus genes, íntrons, regiões codificantes e outras regiões de interesse também cresce. O método faz um escaneamento completo do genoma em questão em busca de um padrão de aminoácidos definido pelo usuário. Geralmente este padrão caracteriza uma classe particular de uma ORF - *open reading frame*, que é uma parte do DNA que ao ser traduzida em aminoácidos não contém nenhum códon de parada. Ou seja, a sequência de DNA iniciada por uma ORF tem uma grande probabilidade de ser um gene, que quando traduzido, gera uma proteína. Através desse método de detecção, a localização das proteínas contidas nos novos genomas sequenciados poderão ser mapeadas para estudos posteriores.

Vantagens e Desvantagens da DFT Os métodos baseados na DFT possuem várias vantagens. A primeira delas é a rapidez com que os algoritmos são executados, permitindo assim que os resultados sejam retornados rapidamente. Em segundo lugar, a DFT é capaz de detectar repetições *exatas e inexatas* com uma sensibilidade notável. Esta é a diferença básica da DFT para os métodos baseados em comparação de *strings*, como o TRF, descrito acima. Programas como o TRF têm dificuldades em encontrar repetições inexatas (ou latentes [14]), pois precisam relaxar os critérios de busca e comparação de modo que os resultados podem se tornar muito genéricos e duvidosos. Outra vantagem é a possibilidade de adaptar a DFT para alguma das várias maneiras de se converter o DNA em um sinal numérico. Este assunto será abordado posteriormente com detalhes na Metodologia. Uma das grandes desvantagens da DFT é que ela não fornece informações de localização da periodicidade na sequência de DNA. Isso significa que ao detectar uma periodicidade no DNA, não há como saber em qual parte da sequência ela ocorre. Em

casos em que é interessante saber a localização da repetição, existem outros métodos alternativos à DFT que fornecerão estas informações. No nosso trabalho, é de nosso interesse saber se a periodicidade está ou não presente no trecho de DNA analisado, e sua localização não é importante. Por isso optamos pela DFT em nossos testes.

Existem vários outros métodos disponíveis para a detecção de periodicidades, entretanto não é o foco desse trabalho abordar tais métodos com detalhes.

1.3 Problema da periodicidade nucleossomal

Os nucleossomos são estruturas altamente conservadas entre os eucariotos. Sua função é empacotar o grande volume de material genético de modo que se encaixe totalmente no núcleo celular. Eles são constituídos por 147 pb de DNA, enovelados pelas histonas, que são proteínas especialmente fabricadas para compactar o material genético. No total, o trecho de DNA é enovelado por um octâmero de histonas, sendo duas de cada uma das classes: H2A, H2B, H3 e H4. Existem outros tipos de histonas, como a H1 e H5, que enovelam externamente o DNA e os nucleossomos, compactando-os mais ainda, para formar os cromossomos [59].

O enovelamento do DNA nas histonas dificulta o acesso de proteínas envolvidas no processamento e/ou regulação do material genético, e por isso, muitas vezes esse mecanismo é considerado uma proteção extra para o DNA [60]. Entretanto, são as histonas que controlam a entrada de elementos regulatórios para acessar o DNA em processos como transcrição, replicação e reparo [61].

As histonas não são estáticas: elas estão constantemente se acoplando/desacoplando ao DNA, facilitando a entrada e ligação de proteínas específicas e tornando acessível o trecho de DNA compactado [62]. Por isso, existe uma estreita relação entre o enovelamento dos nucleossomos e regulação gênica.

Estudos recentes têm mostrado que o posicionamento dos nucleossomos no DNA não é aleatório [62, 63]. Geralmente, os nucleossomos são melhor posicionados na região *downstream* [1] ao sítio de iniciação da transcrição (TSS), que corresponde à posição zero (0). Em especial, o primeiro nucleossomo é muito bem posicionado nesta região. A repetição de 10 pares de base tem sido relatada na literatura como sendo uma espécie de marcação para o posicionamento dos nucleossomos [39, 62, 64, 65]. De acordo com Chodavarapu *et al*, [59], existe uma relação estreita entre o posicionamento dos nucleossomos

[1] O termo *downstream* é consagrado no meio da genética e achamos mais pertinente não traduzí-lo. Para esclarecimento, esta região corresponde a todos os nucleotídeos que se encontram depois do sítio de iniciação da transcrição (TSS). Como padrão, a numeração dos nucleotídeos começa no TSS, que corresponde à posição zero (0). Por isso, os nucleotídeos *downstream* ao TSS são numerados com sinal positivo, ao passo que os *upstream* têm a numeração com sinal negativo.

e a metilação do DNA, comprovado com a existência de um pico no espectro de Fourier correspondente à periodicidade 10.

No entanto, outros grupos encontraram resultados diferentes. Em 2009, Tolstorukov *et al* [36] relataram em seu trabalho que o período de 10 pares de base, presente em sequências de *Saccharomyces cerevisiae*, não é uma característica geral para a cromatina de *H. sapiens*. No trabalho, foi utilizado o material genético resultante do sequenciamento do DNA de células T-CD4$^+$ e DNA de leveduras *S. cerevisiae*. Através do método da correlação de dinucleotídeos, os autores concluem que a periodicidade 10 não é pronunciada nas regiões de posicionamento dos nucleossomos H2A.Z em *H. sapiens*, ao contrário de *S. cerevisiae*, em que a periocidade aparece bem pronunciada. Entretanto, uma fraca expressão da periodicidade 10 foi encontrada nas sequências humanas para alguns dinucleotídeos, especialmente os do tipo GG/CC, AG/CT e GA/TC. Mesmo assim, os níveis de expressão estavam abaixo do nível estatístico de significância e foram desconsiderados.

Um ano depois, o grupo de Hebert *et al* [37] publicou um trabalho onde foram analisadas sequências promotoras humanas que coincidiam com o sítio de iniciação da transcrição (TSS), mais especificamente na região chamada *"nucleossomo +1"*, ou seja, o primeiro nucleossomo da região *downstream* ao TSS. Para este trabalho, o grupo utilizou um conjunto de mais de 13 mil sequências promotoras de *H. sapiens*, provenientes do sequenciamento de cDNA, baixadas da página do banco de dados DBTSS. Através da análise por DFT, os autores chegam à conclusão que as sequências apresentam um pico no espectro de Fourier correspondente à periodicidade 10 para a maioria dinucleotídeos na região do nucleossomo +1, principalmente aqueles formados por C/T e A/G. As conclusões do grupo de Tolstorukov são citadas nesse trabalho, que entram em conflito com os novos resultados.

Tendo em vista as conclusões desses dois trabalhos, ainda permanece a dúvida se a periodicidade 10 está presente, de fato, na região do nucleossomo +1 ao redor do TSS. Ambos trabalhos apresentam argumentações convincentes embasadas na literatura, o que contribui para a aparente veracidade dos seus resultados. Contudo, fica claro que algum deles está equivocado.

Para tentar elucidar qual trabalho apresenta os resultados mais razoáveis, nós pensamos em várias possíveis hipóteses. A que nos pareceu mais verdadeira é que a periodicidade 10 dos dinucleotídeos talvez esteja de fato presente nas sequências que ambos utilizaram, entretanto as *técnicas* escolhidas pelos autores não tenham sido eficientes para detectá-la.

Focando nesta hipótese, nós nos motivamos a construir uma técnica nova, capaz de detectar periodicidades fracas em sequências de DNA e resolver com simplicidade pro-

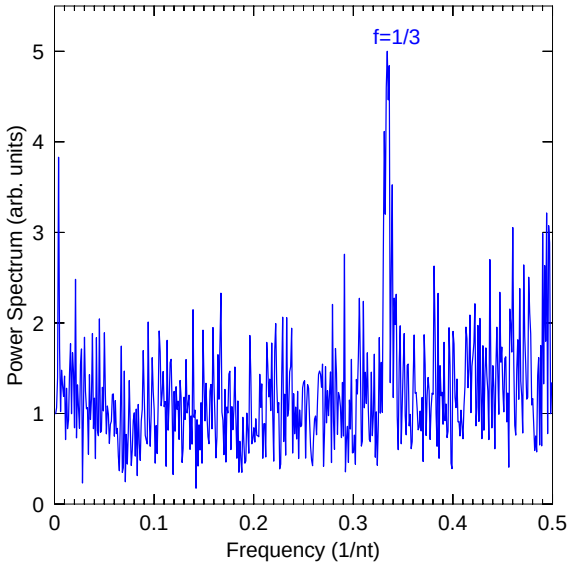

Figura 3
Espectro de potências para um trecho do DNA do cromossomo 5 do parasito *Plasmodium falciparum*. Na região da frequência $f = 1/3$ temos o característico pico referente à periodicidade de 3 nucleotídeos presente na região codificante do DNA, relacionado ao uso de códons. A repetição de 3 nucleotídeos é muito frequente, por isso produz o pico pronunciado. Neste teste foi utilizada a clássica tabela de conversão binária.

blemas como o apresentado acima.

1.4 Detectando periodicidades fracas no DNA

1.4.1 Convertendo letras em números

Com o problema do posicionamento dos nucleossomos, ficou claro que a bioinformática ainda carece de técnicas mais precisas para a detecção de periodicidades em genomas. Então, tendo um ótimo embasamento anterior [3] sobre a eficiência da DFT para tal finalidade, nós resolvemos estudá-la a fundo para compreender como poderíamos adaptar a técnica com o objetivo de melhorar a detecção de periodicidades.

O DNA é um código composto por 4 tipos de letras: A, C, G e T. Para ser trabalhado por uma técnica matemática, tal como a DFT, é preciso converter a sequência de *letras* em uma sequência de *números* [5]. Após a conversão do DNA para uma sequência numérica, aplica-se a DFT e assim teremos condição de visualizar as suas periodicidades. A figura 3 mostra o espectro de potências de Fourier para um corte de 10 mil nucleotídeos do cromossomo 5 do parasito *Plasmodium falciparum*. Na região correspondente à frequência $f = 0.3$, ou $f = 1/3$, encontra-se um pico pronunciado. Este pico é relativo à repetição dos códons para a síntese protéica na região codificante do trecho de DNA selecionado.

Classicamente, utiliza-se uma tabela de conversão chamada de *mapeamento indicador binário* para análises por DFT. Neste tipo de conversão, cada um dos 4 nucleotídeos, A, C, G e T, serão convertidos em uma matriz numérica individual composta por zeros (0) e uns (1). Por exemplo, na matriz relativa à Adenina, na posição do DNA onde houver a letra A será colocado um número 1, e onde houver qualquer outra letra, C, G ou T, será colocado o número 0. Este processo é repetido para os outros 3 tipos de nucleotídeos e posteriormente, é calculada a DFT para as 4 matrizes, e então é feita a soma delas para gerar o espectro de potências. Este processo será explicado com detalhes na seção 3.3 da Metodologia.

Uma extensão simples para o mapeamento indicador binário é utilizá-lo para converter os *dinucleotídeos* do DNA. A sua montagem é simples: para cada um dos 16 pares de dinucleotídeos possíveis, é construída uma matriz binária individual, da mesma forma que é feita para o indicador binário clássico. Posteriormente, é feito o cálculo da DFT, a soma das 16 matrizes, e então será possível visualizar suas periodicidades no espectro de potências de Fourier. Em nossos testes, nós utilizaremos apenas os indicadores binários dinucleotídeos.

A respeito dos mapeamentos indicadores binários, pelo fato de todas as posições da sequência receberem os mesmos valores - zero ou um - e cada matriz ser construída individualmente, pode-se dizer que o estes tipos de mapeamento são *neutros*, ou seja, todas as posições terão o mesmo *peso* na montagem dos espectros de potências.

Muitas vezes, periodicidades que temos certeza de estarem presentes em uma sequência não aparecem no espectro de potências utilizando o mapeamento indicador binário. Tendo em vista esse problema, nós resolvemos mudar o nosso tipo de tabela de conversão. Ao invés do clássico mapeamento indicador binário, nós utilizamos neste trabalho uma tabela de conversão com números reais. Essa mudança é uma contribuição importante do nosso trabalho, tendo em vista que não encontramos nenhum trabalho na literatura que tenha utilizado este tipo de tabela de conversão na DFT.

1.4.2 Escolhendo o melhor mapeamento numérico

Uma vez que escolhemos a estratégia da tabela de conversão com números reais, é hora de escolher *quais valores reais* serão atribuídos aos 16 dinucleotídeos com o objetivo de detectar uma determinada periodicidade. Testar cada combinação de números reais possíves com os algoritmos convencionais levaria um tempo de processamento impraticável. Por isso, nós recorremos a uma técnica matemática chamada de *algoritmo genético*.

Algoritmos genéticos são adaptações computacionais às idéias da seleção natural, pro-

posta por Charles Darwin, e herança genética, proposta por Gregor Mendel [66]. Utilizando os princípios da teoria da evolução, os algoritmos genéticos são métodos de inteligência artificial empregados na busca por soluções otimizadas para problemas de diversas áreas, como ecologia [67, 68], comunicação digital [69], engenharia [70] e até mesmo mercados financeiros [71, 72]. Problemas que admitem um grande número de possíveis soluções são adequados para serem resolvidos pelos algoritmos genéticos.

Numa visão simples, os algoritmos genéticos consistem em evoluir computacionalmente uma população de indivíduos sob uma pressão seletiva específica por um número definido de gerações [73]. A *pressão seletiva* representa o problema a ser resolvido, e os *indivíduos* são possíveis soluções ao problema, sendo os seus *genes* são constituídos por matrizes de números binários ou decimais [74]. Observe o exemplo de uma população simples, composta de apenas 3 indivíduos, ilustrada na figura 4. Cada indivíduo possui seu conjunto de genes, representados pelos quadrados coloridos. O conjunto de genes do indivíduo é denominado *genótipo*.

Para avaliar o quão adaptado é o indivíduo dentro da população, o algoritmo genético estabelecerá seu *fitness*, que é um valor calculado por uma função específica, chamada *função mérito*. Os indivíduos mais adaptados, ou seja, com maior valor de *fitness*, têm uma chance maior de passar seus genes para a próxima geração pelo princípio da seleção natural.

A estratégia de seleção escolhida para ser implementada neste algoritmo é o *torneio*. Nesta estratégia de seleção, dois indivíduos serão escolhidos aleatoriamente na população e seus valores de *fitness* serão comparados. O indivíduo que apresentar o maior valor de *fitness* será escolhido para compor um grupo anexo, chamado *grupo de reprodução*. Este processo será repetido com todos os indivíduos da população, até que o grupo de reprodução seja preenchido com os indivíduos de maior *fitness*. Observe a figura 5 que ilustra esse processo.

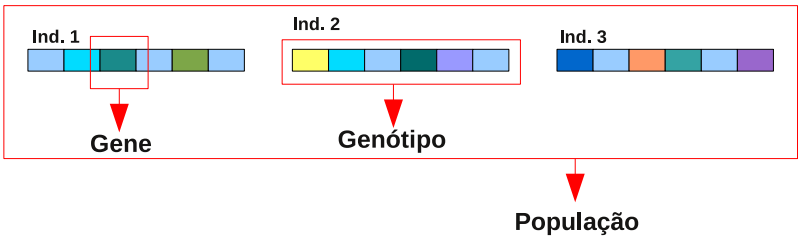

Figura 4
Cada indivíduo possui um conjunto de genes, representado pelos quadrados coloridos. O conjunto de genes de um indivíduo é chamado de genótipo. O conjunto de indivíduos é denominado população.

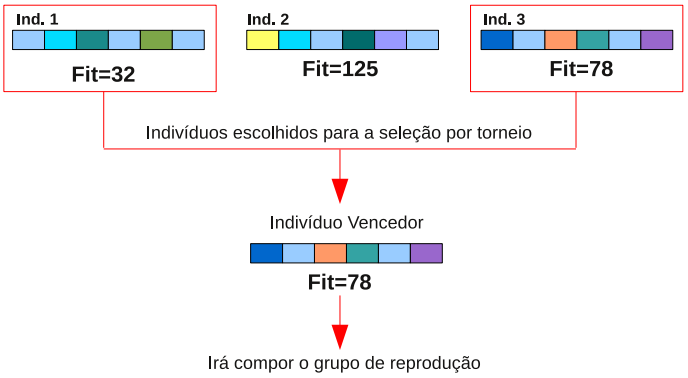

Figura 5
Dentro da população, dois indivíduos serão escolhidos aleatoriamente para compor o torneio. Os seus valores de *fitness* serão comparados e apenas o que tiver o maior valor será escolhido para compor o grupo de reprodução.

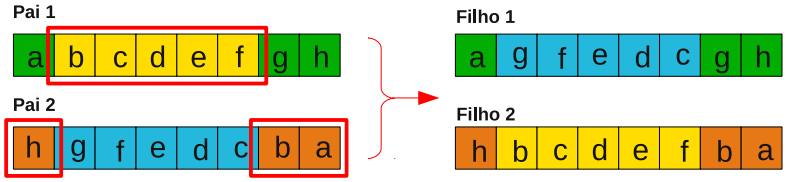

Figura 6
No processo de *crossover*, dois indivíduos são selecionados aleatoriamente dentro do grupo de reprodução, e então, irão produzir dois novos filhos. O processo se repetirá até atingir o número de indivíduos da população original.

Os indivíduos do grupo de reprodução irão compor a população da próxima geração. Mas antes, eles passarão por alguns processos para aumentar o número de indivíduos e a diversidade de seus genes.

O primeiro processo é o *crossover*, que, diferentemente do conceito biológico [2] , é um mecanismo em que dois indivíduos aleatórios do grupo de reprodução serão selecionados e produzirão dois novos filhos. Esse processo é repetido até que o número n de indivíduos no grupo de reprodução seja igual ao número n de indivíduos da população que iniciou o processo de seleção. Por exemplo: se a população é iniciada com $n = 300$ indivíduos, 150 deles serão selecionados pelo torneio, e então, novos indivíduos serão produzidos no grupo de reprodução até atingir $n = 300$. Veja a figura 6 que ilustra o processo de *crossover*.

[2] *Crossover* é o processo de troca de material genético entre dois cromossomos homólogos. Genes correspondentes, contidos nos cromossomos homólogos das células diplóides (2N) são referidos como alelos. Durante o *crossover*, acontece a troca de alelos maternos e paternos entre os cromossomos homólogos. Esse processo é fundamental para o entendimento do mecanismo da herança genética, além de contribuir para a diversidade da população [75, 76].

Tendo formado o total de indivíduos que irão compor a próxima geração, o último processo é a *mutação*. Nela, os indivíduos terão a chance de substituir o valor de algum dos seus genes por outro valor. Esse processo é realizado para aumentar a diversidade da população. A figura 7 representa o processo de mutação.

Para adaptar o algoritmo genético ao nosso trabalho, imagine que os indivíduos da população sejam *mapeamentos candidatos* para otimizar a periodicidade desejada no DNA. Portanto, os *genes* de cada indivíduo serão constituídos de 16 valores para cada dinucleotídeo. A *pressão seletiva* que os indivíduos sofrerão é a capacidade do seu mapeamento otimizar determinada periodicidade. O valor de *fitness* é a medida do quão satisfatória foi a otimização. A partir disto, cada um sofrerá o processo de seleção como descrito acima e, quando o algoritmo finalizar seu trabalho, o mapeamento numérico mais adequado para a otimização da periodicidade para o trecho de DNA em questão será retornado.

Testamos nosso método em uma variedade de organismos, eucariotos e procariotos, para várias periodicidades, a fim de confirmar sua eficiência: *Saccharomyces cerevisiae*, *Anopheles gambiae*, *Caenorhabditis elegans*, *Escherichia coli*, *Methanococcus maripaludis*, e também *Plasmodium falciparum*, *Homo sapiens* e *Drosophila melanogaster*. Nos nossos resultados, mostraremos que os testes foram satisfatórios, com as otimizações sendo bem-sucedidas. Entretanto, selecionamos apenas os resultados para os três últimos organismos citados, pois resumem com sucesso todos os testes que fizemos com a nova técnica.

Figura 7
Os indivíduos submetidos ao processo de mutação têm a chance de trocar os valores de seus genes, com o objetivo de aumentar a diversidade da população.

2 Motivação e objetivos

Muitas dúvidas que surgem no ramo de detecção de periodicidades colocam em xeque a eficiência dos métodos desenvolvidos para este fim. Em nosso trabalho anterior [3], descrevemos como algumas periodicidades aparentemente presentes em um espectro de potências de Fourier podem ser falsas. Na Introdução, descrevemos o problema da periodicidade nucleossomal, de 10 nucleotídeos, que dois autores relatam resultados conflitantes ao afirmar sobre sua presença em sequências promotoras de *Homo sapiens*.

Tendo isto em vista, nosso objetivo geral é:

1. Construir um algoritmo na linguagem Perl que seja eficiente e confiável e que tenha como foco encontrar um mapeamento numérico ideal para otimizar uma periodicidade específica em uma sequência de DNA, utilizando para isto a combinação da transformada de Fourier discreta com um algoritmo genético.

A partir disso, os objetivos específicos são:

1. No *Plasmodium falciparum* e na *Drosophila melanogaster*, otimizar as periodicidades de 3 e 10 nucleotídeos, correspondentes às frequências $f = 1/3$ e $f = 1/10$. A primeira é relativa à síntese protéica nas regiões codificantes, e a segunda, ao número de nucleotídeos responsáveis por uma rotação na hélice dupla da molécula de DNA e também ao posicionamento dos nucleossomos no genoma.

2. Especificamente para *Homo sapiens*, separar a região promotora descrita como correspondente ao posicionamento do *nucleossomo +1* e detectar a presença da periodicidade de 10 nucleotídeos nesta região, com o objetivo de esclarecer o problema da periodicidade nucleossomal.

3. Analisar e comparar qual seria a diferença em todos estes testes utilizando vários mapeamentos numéricos para transformar o DNA em um sinal digital:

   (a) Mapeamento binário di-nucleotídeo;

   (b) Mapeamento di-nucleotídeo usando números reais, sendo eles:

      i. Com valores variando numa escala entre -1 e 1;

      ii. Com valores apenas positivos, numa escala entre 0.01 e 1.

3 Metodologia

3.1 Dados utilizados nos testes

3.1.1 Sequências de DNA do *P. falciparum* e *D. melanogaster*

Os testes de otimização para estes organismos foram feitos utilizando os arquivos FASTA dos seguintes cromossomos:

- Cromossomo 8 do *Plasmodium falciparum* - número de acesso *NC*_004329.1.
- Cromossomo 2L da *Drosophila melanogaster* - número de acesso *NT*_033779.4.

O *download* dos arquivos foi feito a partir do site do NCBI [77]. Após analisar o arquivo FASTA referente ao cromossomo 8 do *P. falciparum*, pudemos perceber que o mesmo era composto de muitas posições com caracteres "N", ou seja, nucleotídeos que ainda não foram identificados. As primeiras cem mil posições do cromossomo continham menos caracteres deste tipo, por isso optamos por fazer um corte da posição 0 até 100000 do arquivo, e fazer nossos testes utilizando este trecho. Por motivos de padronização, optamos por fazer um corte nas mesmas posições do arquivo FASTA referente à *D. melanogaster*. Devido ao tempo impraticável gasto pelo algoritmo para percorrer um cromossomo inteiro, nós optamos por usar os cortes de 100000 nucleotídeos, que foram suficientes para nos fornecer os resultados necessários para testar o método.

3.1.2 Sequências promotoras de *H. sapiens*

Para confirmar a presença da periodicidade de 10 nucleotídeos nas sequências nucleossomais de *Homo sapiens*, nós fizemos o *download* das 9714 sequências promotoras humanas disponíveis em formato FASTA na página da *internet* do banco de dados EPD-*Eukaryotic Promoter Database* [78]. Pelo fato da localização dos sítios de iniciação da transcrição (TSS) [3] dessas sequências terem sido previamente definidas e publicadas na literatura, o *site* do EPD permite que o usuário escolha as posições em que gostaria de iniciar a extração das sequências, bem como a quantidade de nucleotídeos. A posição zero (0) corresponde ao início do TSS. A página permite que o usuário escolha quantas

[3] TSS é a sigla para o termo em inglês *Transcription Start Site*. Em português, a tradução mais utilizada na literatura é "sítio de iniciação da transcrição". Os TSS são sequências de DNA que ficam imediatamente após as regiões promotoras e antes de um provável gene. A enzima RNA polimerase reconhece os nucleotídeos da região promotora (um exemplo clássico é o *TATA Box*) e logo em seguida reconhece o TSS, onde ela se acopla inicia a transcrição do trecho de DNA subsequente, que geralmente corresponde a um gene.

posições ao redor do TSS serão incluídas no arquivo FASTA final. A posição do nucleossomo +1 coincide geralmente com as posições $+40$ até $+190$ no sentido *downstream*, ou seja, abaixo do TSS [62]. Nesta seção de DNA a periodicidade de 10 nucleotídeos [39] deverá ser mais pronunciada [37].

3.2 Aplicação da Transformada de Fourier Discreta (DFT)

A transformada de Fourier discreta (DFT) é uma técnica matemática que mostra as frequências (periodicidades) presentes em um sinal digital através do espectro de potências de Fourier (FPS). Para que a DFT seja aplicada no DNA, é necessário convertê-lo em uma sequência numérica, o que é feito com facilidade utilizando tabelas de conversão. Uma periodicidade que eventualmente esteja presente no DNA apresenta-se na forma de um pico na sua frequência específica no FPS. Quanto mais alto e forte for o pico, maior é a quantidade de vezes que a periodicidade aparece na série. A DFT é muito utilizada em análises na genômica e proteômica, devido à sua facilidade de implementação, eficiência e praticidade [3, 5, 19, 47, 79–82].

A implementação computacional da DFT nos nossos algoritmos foi feita através da *fast Fourier transform* (FFT), ferramenta que realiza a interface entre o algoritmo e a sequência de DNA. Em nossos algoritmos, utilizamos o pacote FFTW (*fastest Fourier transform in the west*) para os cálculos da DFT nas sequências de DNA. A página para *download* do pacote encontra-se no endereço presente na referência 83.

3.3 Mapeamento indicador binário

O DNA é composto por símbolos que correspondem aos nucleotídeos A, C, G e T, que matematicamente resultam numa sequência de caracteres que pode ser analisada através de ferramentas computacionais. Um sinal binário, por sua vez, é uma sequência em que cada posição é 0 (zero) ou 1 (um). A seguinte equação exemplifica como construir um indicador binário para uma sequência de DNA qualquer:

$$u_{\alpha,n} \begin{cases} 1 & \text{se for } \alpha \text{ na posição } n \\ 0 & \text{para outro nucleotídeo.} \end{cases} \tag{1}$$

onde $\alpha = \mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}$.

A partir das sequências de DNA, construímos então 4 sequências binárias, uma para cada nucleotídeo [84]. Para ilustrar o método, considere a seguinte sequência de DNA hipotética: *ATCGATCG*. Transformando-a em indicador binário para cada tipo de nucleotídeo, teremos as seguintes sequências:

Para o espectro da adenina (A), por exemplo, toda vez que o algoritmo encontrar o símbolo A na sequência, será assinalado o valor 1 (um) àquela posição, e para qualquer um dos outros símbolos, C, G ou T, será atribuído o valor 0 (zero). O mesmo é feito para os outros tipos de nucleotídeos, criando assim 4 matrizes binárias diferentes, que então podem ser tratadas numericamente. Uma vez montados os indicadores binários das sequências de DNA, será possível realizar a DFT da sequência e procurar por periodicidades.

Para uma sequência de DNA de tamanho N contendo os nucleotídeos A, C, G e T, a aplicação da DFT retornará uma sequência de mesmo tamanho N, sendo que cada posição representa os coeficientes de Fourier correspondentes a cada nucleotídeo. Então, considere que uma *periodicidade* p do DNA é interpretada pela DFT como uma *frequência* $f = 1/p$. Para uma determinada posição k no FPS de tamanho N, uma frequência f será considerada:

$$f = \frac{k}{N} \tag{2}$$

Consequentemente, a frequência f pode ser usada para determinar em qual posição k na sequência N ela ocorrerá:

$$k = fN \tag{3}$$

Para exemplificar, considere uma sequência de DNA de tamanho $N = 999$ que gostaríamos de encontrar a posição em que a periodicidade $p = 3$ ocorre. Aplicando nestes dados a equação 2, descobrimos que a frequência correspondente a esta periodicidade é $f = 1/3$ [4] . Calculado o f, fica possível descobrir a posição k aplicando-se a equação 3:

[4] A rigor, a notação correta para a frequência correspondente à periodicidade $p = 3$ seria $f = 1/3$ nt^{-}1, entretanto por simplicidade optamos por omitir a unidade $nt^{-}1$ no restante desta dissertação.

	A	T	C	G	A	T	C	G
n	0	1	2	3	4	5	6	7
$u_{A,n}$	1	0	0	0	1	0	0	0
$u_{C,n}$	0	0	1	0	0	0	1	0
$u_{G,n}$	0	0	0	1	0	0	0	1
$u_{T,n}$	0	1	0	0	0	1	0	0

Tabela 1
Mapeamento indicador binário para uma sequência de DNA hipotética. A posição 0 (zero) corresponde ao primeiro nucleotídeo da sequência. O número 1 (um), em azul, significa que o nucleotídeo em pauta foi encontrado na sequência. Caso contrário, será atribuído o valor 0 (zero) àquela posição.

$k = N/3$. Uma vez que $N = 999$, logo a posição em que a periodicidade 3 está localizada no FPS é $k = 333$.

A aplicação da DFT nos indicadores binários obedece à seguinte equação:

$$U_{\alpha,k} = \frac{1}{N} \sum_{n=0}^{N-1} u_{\alpha,n} e^{-i\frac{2\pi}{N}kn}, \quad k = 0,1,2,\ldots,N-1, \tag{4}$$

onde $\alpha = \mathrm{A,C,G,T}$.

Calculada a DFT para cada um dos 4 elementos das matrizes numéricas, é necessário somar os coeficientes de Fourier para produzir apenas uma matriz, e a partir de então será possível montar o FPS. Este procedimento é ilustrado pela seguinte equação:

$$S_k = \sum_{\alpha=\mathrm{A,C,G,T}} |U_{\alpha,k}|^2, \quad k = 0,1,2,\ldots,N-1 \tag{5}$$

Uma vez montado o FPS referente à sequência de DNA desejada, será possível então observar todas as frequências que ocorrem em um determinado sinal digital.

3.4 Mapeamento binário de dinucleotídeos

A característica principal dos mapeamentos binários é a neutralidade. Isso significa que todas as posições possuem pesos idênticos, zero (0) ou um (1). Com isso, as informações e a estrutura contida na sequência de DNA são preservadas, mesmo quando ela é traduzida para uma sequência numérica [85].

Existem 16 pares possíveis de dinucleotídeos nas sequências de DNA:

Adenina	AA	AC	AG	AT
Citosina	CA	CC	CG	CT
Guanina	GA	GC	GG	GT
Timina	TA	TC	TG	TT

Nesse caso, dois nucleotídeos consecutivos são convertidos numa sequência binária. Em termos matemáticos, a montagem destas sequências é feita facilmente reescrevendo a Eq. (1) dos indicadores binários para um nucleotídeo:

$$v_{n,n+1}^{\alpha,\beta} = \begin{cases} 1 & \text{se for } \alpha \text{ na posição } n \text{ e } \beta \text{ na posição } n+1 \\ 0 & \text{para outro dinucleotídeo} \end{cases} \tag{6}$$

onde $\alpha = \mathrm{A,C,G,T}$.

A partir desta equação, serão formadas 16 sequências binárias para cada um dos dinucleotídeos. O próximo passo é calcular a DFT para cada um deles:

	A	C	G	T	A	A	G	C	C	T	C	A	T	G	G	A	T
n	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
$u_{AA,n}$	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	
$u_{AC,n}$	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
$u_{AG,n}$	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	
$u_{AT,n}$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
$u_{CA,n}$	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	
$u_{CC,n}$	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	
$u_{CG,n}$	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
$u_{CT,n}$	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	
$u_{GA,n}$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	
$u_{GC,n}$	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	
$u_{GG,n}$	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	
$u_{GT,n}$	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
$u_{TA,n}$	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	
$u_{TC,n}$	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	
$u_{TG,n}$	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
$u_{TT,n}$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

Tabela 2
Mapeamento indicador binário de dinucleotídeos para uma sequência de DNA hipotética. A posição 0 (zero) corresponde ao primeiro dinucleotídeo da sequência. O número 1 (um), em azul, significa que o dinucleotídeo em pauta foi encontrado na sequência. Caso contrário, será atribuído o valor 0 (zero) àquela posição.

$$V_k^{\alpha,\beta} = \frac{1}{N}\sum_{n=0}^{N-1} v_n^{\alpha,\beta} e^{-i\frac{2\pi}{N}kn}, \quad k = 0,1,2,\ldots,N-1, \tag{7}$$

onde $\alpha = \mathrm{A},\mathrm{C},\mathrm{G},\mathrm{T}$.

O último passo é somar os 16 FPS, como mostrado na Eq. (5), para produzir apenas um FPS com o total:

$$T_k = \sum_{\alpha,\beta=\mathrm{A},\mathrm{C},\mathrm{G},\mathrm{T}} |V_k^{\alpha,\beta}|^2. \tag{8}$$

Na tabela 2 temos um exemplo de como ficaria o mapeamento binário de dinucleotídeos para uma sequência de DNA hipotética. Uma metodologia similar a esta foi utilizada nos trabalhos de Reynolds *et al.* e Widom *et al.* [64, 86].

3.5 Mapeamento de dinucleotídeos (*DM*)

Neste tipo de mapeamento, nós utilizamos números reais nos lugares dos dinucleotídeos, ao invés dos tradicionais zeros (0) e uns (1) dos mapeamentos binários. Uma característica do mapeamento com números reais é que ele não possui a neutralidade,

característica marcante dos mapeamentos binários. Isto acontece porque este tipo de mapeamento é construído atribuindo-se um número real diferente a cada um dos 16 pares possíveis de dinucleotídeos. Com isso, os dinucleotídeos terão um peso diferente, ao contrário dos mapeamentos indicadores binários, em que todas as posições têm o mesmo peso.

Representando em forma de equação, a DFT para o mapeamento *DM* tem a seguinte forma:

$$D_k = \frac{1}{N}\sum_{n=0}^{N-2} d_{n,n+1}^{\alpha,\beta} e^{-i\frac{2\pi}{N}kn}, \quad k = 0,1,2,\ldots,N-2 \tag{9}$$

onde α e β são os dinucleotídeos nas posições n e $n+1$, respectivamente.

Nesta série, os dinucleotídeos receberão valores numa escala entre -1 e 1.

Com isso, um determinado dinucleotídeo pode naturalmente receber um valor maior do que outro. Imagine que gostaríamos de maximizar uma frequência qualquer em uma sequência de DNA, e que o dinucleotídeo $d^{\mathrm{C,A}}$ receba o valor 0,1 e $d^{\mathrm{T,G}}$ receba 0,6. Numa análise simples, fica claro que o dinucleotídeo $d^{\mathrm{T,G}}$ terá uma *contribuição maior* no espectro do que o dinucleotídeo $d^{\mathrm{C,A}}$. A explicação mais plausível para isso é que a periodicidade que estamos procurando seja constituída em uma maior parte pelos dinucleotídeos contendo timina e guanina, fazendo com que eles tenham maior relevância comparando-se aos que contenham citosina e adenina.

Para enriquecer a interpretação dos resultados, nós incluímos outra escala de valores para atribuir aos dinucleotídeos. Variando entre 0.01 e 1, nesta nova escala os dinucleotídeos receberão apenas valores positivos. Nós nomeamos este tipo de mapeamento como DM^{+}.

Na tabela 3, temos um exemplo do mapeamento *DM*, e na tabela 4, um exemplo do mapeamento DM^{+}. Em ambos exemplos, utilizamos a mesma sequência de DNA hipotética da tabela 2.

	A	C	G	T	A	A	G	C	C	T	C	A	T	G	G	A	T
n	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
Indivíduo 1	0.44	0.98	-0.77	0.52	-0.41	0.05	-0.99	0.4	-0.82	0.69	0.91	0.26	-0.92	-1	-0.52	0.12	
Indivíduo 2	0.06	-0.36	-0.16	0.02	0.29	-0.43	1	0.13	0.37	0.97	0.26	-0.05	-0.02	-0.19	-0.11	-0.06	
indivíduo 3	0	0.21	-0.37	-0.11	0.98	0.76	0.9	0.38	-0.13	0.37	0.48	-0.9	-0.08	-0.89	0.99	0.28	

Tabela 3

Mapeamento de dinucleotídeos (*DM*) para uma sequência de DNA hipotética. A posição 0 (zero) corresponde ao primeiro dinucleotídeo da sequência. Vários números reais diferentes são atribuídos aos diferentes dinucleotídeos, variando numa escala de -1 a 1.

	A	C	G	T	A	A	G	C	C	T	C	A	T	G	G	A	T
n	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
Indivíduo 1	0.83	0.9	0.21	0.63	0.09	0.59	0.71	0.39	0.42	0.89	0.82	0.49	0.08	0.06	0.27	0.92	
Indivíduo 2	0.01	0.01	0.38	0.01	0.01	0.02	0.01	0.01	0.42	0.03	0.01	0.01	0.01	1	0.12	0.01	
indivíduo 3	0.03	0.98	0.97	0.16	0.01	0.72	0.07	0.05	0.72	0.19	0.07	0.01	0.14	0.95	0.56	0.02	

Tabela 4
Mapeamento de dinucleotídeos com valores apenas positivos (DM^{+}) para uma sequência de DNA hipotética. A posição 0 (zero) corresponde ao primeiro dinucleotídeo da sequência. Vários números reais diferentes são atribuídos aos diferentes dinucleotídeos, variando numa escala de 0.01 a 1.

3.6 Algoritmo genético (AG)

3.6.1 Implementação

Os algoritmos genéticos (AG), também chamados de algoritmos evolucionários, são técnicas de inteligência artificial, muito utilizados nas áreas exatas em problemas de otimização [73, 74]. Esta área está sendo cada vez mais utilizada, também pelo fato dos AG's poderem ser aplicados em problemas de otimização em diversas áreas, como medicina [87,88], engenharia [89,90], biologia [91,92] e na física acústica [93,94]. O objetivo dos AG's é avaliar e otimizar as possíveis soluções para um determinado problema complexo, e retornar uma ou mais soluções ótimas [95]. Neste trabalho, nós utilizamos o pacote AI::Genetic, que contém uma implementação em Perl de um AG, livremente disponível *online* [96].

3.6.2 Parâmetros de entrada e cálculo do *fitness*

População Para fornecer ao AG um grande número de possíveis soluções, nós configuramos o tamanho da *população* em $p = 300$ indivíduos por geração. Para entender a idéia, imagine que cada indivíduo dentro da população seja uma alternativa de mapeamento para achar determinada frequência no DNA. Os 300 indivíduos serão avaliados com relação ao seu conteúdo, que são os *genes* e comparados entre si para ver qual maximiza melhor a periodicidade de interesse.

Número de genes De acordo com o funcionamento dos AG's [74], cada indivíduo da população tem um conjunto de genes que pode variar em tipo (texto, numeral, binário) e quantidade. Esse é o chamado *genótipo*. Como o nosso interesse é encontrar o melhor mapeamento de dinucleotídeos para otimizar uma determinada frequência, nosso conjunto de genes será constituído de 16 valores, que correspondem aos 16 possíveis pares de dinucleotídeos encontrados no DNA.

Cada gene de um indivíduo receberá um valor dentro de uma das escalas de números

reais que nós definimos para este trabalho - DM e DM^+. Pelo fato do pacote AI::Genetic trabalhar apenas com valores inteiros, nós tivemos que fazer uma adaptação para gerar os valores das escalas. Ao gerar a população, a cada um dos 16 genes será atribuído inicialmente um valor inteiro entre -100 e 100 (DM) ou entre 1 e 100 (DM^+). O algoritmo trabalhará com esses valores durante toda a execução, e ao encontrar o melhor conjunto que maximize a frequência de interesse, ao final todos os 16 valores serão divididos por 100, e daí teremos a escala entre -1 e 1 e entre 0.01 e 1. O tipo de mapeamento (DM ou DM^+) é escolhido ao iniciar a execução do programa, de acordo com o interesse do usuário.

Determinação do *fitness* de cada indivíduo Uma vez que a população inicial p com 300 indivíduos foi gerada, significa que cada indivíduo será constituído de um mapeamento de números reais. Para calcular o *fitness*, o primeiro passo é calcular a DFT para cada indivíduo. O algoritmo então produzirá o seu respectivo FPS. O próximo passo é avaliar a intensidade do pico na frequência desejada. Esta etapa é importante para descobrir se a otimização foi efetiva. Para isso, criamos uma *função mérito*, que foi baseada na função chamada de *relação sinal-ruído*, utilizada em alguns trabalhos [47, 97, 98]. Esta função leva em conta a altura do pico no FPS na frequência desejada e a compara com a altura dos demais picos do espectro. Disso, tem-se uma taxa que diz o quão forte é uma periodicidade.

Para descobrir periodicidades p de dinucleotídeos que estejam fracas em determinada sequência, nós maximizamos o FPS S_k em uma determinada frequência $f = 1/p$ em relação à intensidade total do espectro $S = \sum_k S_k$, o que consiste então na função mérito $g(p)$, dada por:

$$g(p) = S_k/S \tag{10}$$

Dependendo do tamanho da sequência, há um problema técnico: o pico na frequência desejada pode não coincidir com a posição calculada pelo algoritmo. Por exemplo, a posição k que corresponde à frequência $f = 1/3$ em uma sequência de 1000 nucleotídeos é k =333.3. Este valor não é um número inteiro, portanto não existe esta posição no espectro por problemas de arredondamento. A posição mais próxima seria k =333, mas a frequência representada neste ponto não é necessariamente $f = 1/3$. A consequência disso é que algumas frequências podem simplesmente não aparecer no espectro.

Para resolver este problema, realizamos uma amostragem de δ posições do vetor Fourier em torno do pico. Neste trabalho, nós utilizamos em geral δ=1, ou seja, uma posição de cada lado da frequência desejada. Matematicamente, considerando que o FPS S será

representado por um vetor $\mathscr{S} = s_1, s_2, \ldots, s_{N/2}$, para uma frequência f:

$$S_k^{\delta}(N) = \sum_{n=i-\delta}^{i+\delta} s_n, \quad i = \text{round}(k) \tag{11}$$

Então, voltando ao nosso exemplo, para a frequência $f = 1/3$ num trecho de $N = 1000$ nucleotídeos, as posições 332, 333 e 334 do vetor serão utilizadas para fazer o cálculo. Isso garante que a frequência em questão seja contemplada no FPS.

Estratégia de seleção Depois de gerada a população e calculados seus valores de *fitness*, é preciso decidir quais são os indivíduos mais adaptados para serem inseridos na população da próxima geração. Para isso, nós utilizamos uma técnica de seleção chamada *torneio*. Este método foi escolhido por sua simplicidade e porque geralmente é considerado bastante eficiente [99]. A melhor característica desta técnica é que ela preserva a diversidade da população, pois uma vez que todos os indivíduos participam do torneio, todos têm chance de serem selecionados [100].

No processo de seleção, uma amostragem de N_T indivíduos será tirada a partir de toda a população. Este número é chamado de tamanho do torneio, ou N_T. Os indivíduos escolhidos serão comparados entre si quanto ao seu valor de *fitness*, e apenas o que possuir o maior valor será escolhido. Este processo é repetido até que toda a população seja testada.

Neste trabalho, escolhemos $N_T = 2$, ou seja, apenas 2 indivíduos serão comparados de cada vez. Os indivíduos selecionados em cada torneio vão compor o grupo de reprodução. Neste grupo, os indivíduos irão gerar uma prole, que será a base para a população da próxima geração [101].

O número de indivíduos que vão compor o tamanho do torneio é decisivo para manter a diversidade da população. Quanto mais indivíduos compuserem o torneio, significa que uma pequena parte da população contribuirá efetivamente para a diversidade genética [102, 103]. Isso acontece porque vários deles não serão selecionados para compor o grupo de reprodução e não terão a chance de participar novamente de um torneio, pois são descartados [100, 101]. Este é mais um motivo pelo qual escolhemos fazer o torneio com apenas dois indivíduos, $N_T = 2$, pois comparando aos pares haverá uma maior chance de escolhermos os melhores indivíduos.

Ao fim da seleção por torneio, toda a população já terá sido testada aos pares, e todos os indivíduos vencedores estarão no grupo de reprodução. Naturalmente, os indivíduos desse grupo terão um *fitness* médio maior que o resto da população. Essa, então, será a *pressão seletiva* que os indivíduos sofrerão, pois a cada geração, só aqueles com maior *fitness* serão escolhidos. No grupo de reprodução, os indivíduos sofrerão dois processos:

crossover e mutação, para gerar uma prole, que será a base da próxima população.

Crossover O conceito de *crossover* nos AG's é diferente do utilizado na recombinação gênica. Na biologia, o *crossover* ocorre durante a meiose, processo que resulta na formação de gametas nas células germinativas. Os cromossomos homólogos (parentais) trocam partes entre si, o que resulta em cromossomos com maior diversidade genética, que irão para os gametas [75]. Nos AG's, os cromossomos são equivalentes aos indivíduos da população. No grupo de reprodução, dois indivíduos (pais) serão selecionados aleatoriamente para trocar partes de seu conteúdo entre si, e formar **dois** novos indivíduos (prole), que farão parte da população da próxima geração. O *crossover* nos AG's, portanto, combina informações de dois indivíduos parentais que são supostamente diferentes.

No nosso AG, a taxa de *crossover* é uniforme, ou seja, a característica desse tipo de recombinação é que, a partir dos pais, ele constrói um novo filho selecionando aleatoriamente, para cada *locus*, um alelo de um ou de outro pai [104, 105]. Para entender melhor a construção da prole, observe o simples exemplo abaixo. Imagine que, para os genes de cada pai, o AG construirá uma máscara binária aleatoriamente, que corresponderá aos dois genótipos:

Pai 1	A	B	C	D	E
Pai 2	a	b	c	d	e
Máscara binária	0	1	0	0	1

O primeiro filho herdará os genes do Pai 1 que correspondam ao número 1 (um) na máscara, e os genes do Pai 2 que correspondam ao número 0 (zero). O segundo filho é

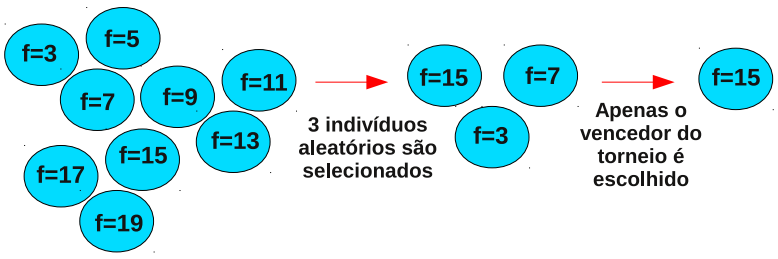

Figura 8
Esquema exemplificando como funciona o torneio como estratégia de seleção nos AG's. À esquerda, os círculos representam os indivíduos de uma população. Cada um possui um valor de *fitness*, representado pela letra f. Aleatoriamente, três indivíduos serão escolhidos para formar o torneio. Neste caso, $N_T = 3$. Os três serão comparados em relação ao seus valores de *fitness*, e apenas o que possuir o maior valor será selecionado para compor o grupo de reprodução. Este processo é repetido para todos os indivíduos da população, até que o grupo de reprodução seja preenchido.

construído obedecendo à regra contrária: Então:

Filho 1	a	B	c	d	E
Filho 2	A	b	C	D	e

Assim, o AG construirá para cada par de pais do grupo de reprodução uma nova prole [104].

Após testarmos diferentes valores para se atribuir à taxa de *crossover*, decidimos usar no nosso AG uma taxa definida em 0.7. Isso significa que, a cada par de pais escolhidos do grupo de reprodução, o AG escolherá um número aleatório entre 0 e 1. Se o número for menor que 0.7, o *crossover* ocorrerá e dois novos indivíduos serão gerados. Caso contrário, o par de pais será deixado de lado e um novo será escolhido. Este processo se repetirá até que se atinjam 300 indivíduos, que corresponde ao número de população para começar uma nova geração.

Mutação Após atingido o número de 300 indivíduos, a nova população passará pelo processo de mutação. A mutação permite que um indivíduo tenha a chance de trocar o valor de algum dos seus genes de acordo com a *taxa de mutação*. Este processo tem por objetivo aumentar a variabilidade da população. O mecanismo é simples: imagine que um indivíduo não selecionado pelo torneio tivesse um gene importante. Pelo fato do indivíduo ter sido descartado, aquele gene foi perdido. Durante o processo de mutação, ao trocar o valor de um gene de um indivíduo, existem chances do novo valor ser exatamente aquele perdido pelo indivíduo não selecionado anteriormente [105].

A taxa de mutação tem um papel importante, uma vez que valores muito altos aumentam muito a variabilidade da população de uma maneira que prejudica a eficiência da seleção, que se torna praticamente aleatória [106]. Por isso, após testar várias taxas de mutação diferentes, nós fixamos seu valor em 0.01 ou 1% no nosso AG. Ao aplicar a mutação nos indivíduos, o AG aplica um método parecido com o *crossover*. Ele irá gerar um valor aleatório entre 0 e 1. Caso este valor esteja *abaixo* de 1%, um gene do indivíduo em questão irá substituir seu valor por um novo, dentro da escala normal de construção do genótipo, definida pelo usuário. Este processo é repetido com todos os indivíduos do grupo.

Finalizada a seleção, o *crossover* e a mutação, teremos pronta a nova população, com valores de *fitness* melhores e que tenderão a melhorar ainda mais a cada geração, contribuindo para que o AG encontre a solução ideal no menor tempo possível.

3.6.3 Critério de parada

Com o objetivo de melhorar o tempo de execução do AG, nós implementamos um critério de parada para testar se os últimos valores de *fitness* retornados pelo AG já são os maiores que ele poderia alcançar. Para isso, nós armazenamos em um vetor os valores de *fitness* retornados pelas últimas 30 gerações: $\{g_1, g_2, \ldots, g_{30}\}$. Em seguida, fazemos uma subtração entre o trigésimo e o primeiro valor do vetor:

$$|g_{30} - g_1| < t \tag{12}$$

Se o valor de t estiver abaixo de $t = 0.01$, significa que o mapeamento numérico já se encontra otimizado no seu limite, uma vez que não houve praticamente nenhuma mudança no seu valor nas últimas 30 gerações. Então o AG retornará o mapeamento e fecha a geração corrente, passando para o próximo trecho de DNA, repetindo o mesmo procedimento. Caso o resultado da subtração esteja acima de $t = 0.01$, ele continuará realizando as otimizações até que o critério seja alcançado, ou então parará a execução quando chegar no número limite de 500 gerações, retornando então o último mapeamento.

3.7 Passo-a-passo do AG

Nesta seção mostraremos como o algoritmo funciona tecnicamente. No Apêndice, seção 7.1.2, está listado o algoritmo completo para conferência. Abaixo listamos os principais passos do seu funcionamento.

1. Os parâmetros de entrada são:
   (a) arquivo com a sequência de DNA;
   (b) nome que deseja dar ao arquivo de saída;
   (c) frequência que deseja otimizar no formato $f = 1/p$;
   (d) tipo de mapeamento que deseja usar, DM^+ ou DM.
2. Extrair os dados do arquivo contendo a sequência de DNA.
3. Percorrer toda a sequência utilizando uma janela deslizante, sendo esta:
   (a) de 1000 nucleotídeos para o *P. falciparum* e *D. melanogaster*.
   (b) de 150 nucleotídeos para os promotores de *H. sapiens*, sendo que neste caso o tamanho da janela é igual ao tamanho da sequência.

4. Em cada uma das janelas, procurar pelo melhor mapeamento que maximize com eficiência a frequência desejada:

   (a) frequências $f = 1/3$ e $f = 1/10$, correspondentes aos períodos $p = 3$ e $p = 10$ no caso de *P. falciparum* e *D. melanogaster*;

   (b) apenas $f = 1/10$ no caso dos promotores de *H. sapiens*.

5. Aleatoriamente, atribuir aos 16 pares de dinucleotídeos valores reais positivos ou negativos dependendo do mapeamento escolhido: DM^{+} ou DM.

6. Em seguida, testar através da DFT se o mapeamento foi satisfatório para maximizar a frequência.

7. Repetir o processo até obedecer ao critério de parada ou atingir o limite de 500 gerações.

8. Terminado o processo, salvar no arquivo de saída o melhor valor de *fitness* para a sequência e os valores reais ótimos para cada um dos dinucleotídeos.

9. Passar para a próxima janela de DNA ou para o próximo arquivo de sequências promotoras e repetir todo o processo.

3.7.1 Construção do FPS

Com os valores reais retornados pelo AG, será possível construir os espectros de Fourier para cada sequência de DNA e assim comparar os resultados. O algoritmo que faz este trabalho se encontra na seção 7.1.3 do Apêndice.

Este algoritmo basicamente irá extrair os valores reais do arquivo de saída retornado pelo AG, aplicará os valores aos dinucleotídeos correspondentes no DNA e calculará a DFT. Em seguida irá montar o FPS onde será possível observar se a periodicidade em questão foi maximizada satisfatoriamente.

4 Resultados e Discussão

4.1 Analisando a eficiência do método

4.1.1 Variação dos valores dos dinucleotídeos

O algoritmo genético (AG) inicia uma nova população a partir de valores completamente aleatórios. Uma dúvida inicial que nos surgiu é: se repetirmos o AG na mesma sequência várias vezes, os valores reais retornados para os dinucleotídeos seriam completamente diferentes uns dos outros? Caso afirmativo, os valores não fariam sentido, pois seriam totalmente aleatórios.

Para resolver esta questão, nós aplicamos nosso AG dez vezes nos 100000 nucleotídeos selecionados do cromossomo 2L da *Drosophila melanogaster* buscando pelos melhores valores dos dinucleotídeos que maximizem a frequência $f = 1/3$. Ao final de cada rodada, o AG retorna os valores dos dinucleotídeos. O tempo médio gasto pelo AG para terminar uma rodada completa no trecho de 100000 nucleotídeos é de aproximadamente 8 horas. Terminada a décima repetição, montamos um gráfico que mostra se os valores para cada dinucleotídeo sofreram muita variação. O resultado desse teste está mostrado na figura 9. Pode-se perceber que as flutuações dos valores foram mínimas. Existe portanto uma tendência clara para a manutenção dos valores dentro de uma faixa, raramente apresentando diferenças muito importantes.

A pergunta que surge a seguir é: os valores otimizados convergem para os mesmos va-

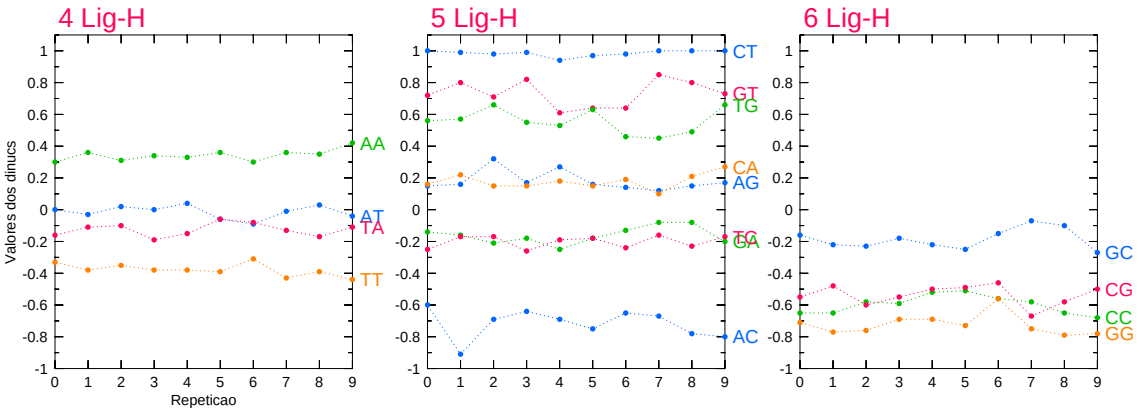

Figura 9
Flutuação dos valores dos dinucleotídeos quando repetidos 10 vezes na mesma sequência. Para facilitar a análise, separamos os dinucleotídeos pelo número de ligações de hidrogênio que fazem. No primeiro painel, os que fazem 4 ligações, no painel do meio 5 ligações e no último, 6 ligações. Percebe-se uma tendência à manutenção dos valores em uma faixa, com poucas variações importantes.

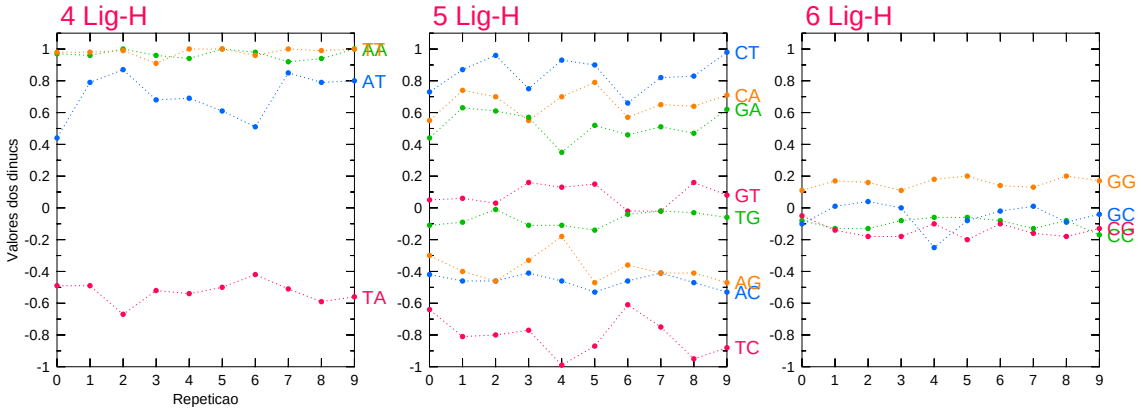

Figura 10
Os valores dos dinucleotídeos retornados para a sequência aleatorizada da *Drosophila melanogaster* também tendem a manter seus valores dentro de uma faixa quando repetimos o AG dez vezes na mesma sequência.

lores por se tratar de uma sequência de DNA estruturada, com significado biológico bem definido? Uma forma de responder a esta pergunta é aleatorizar totalmente a sequência de DNA utilizada no teste anterior. Com isso, destruímos o significado biológico, organização das repetições e toda a estrutura natural do DNA. Depois, novamente aplicamos o AG 10 vezes nessa sequência, buscando maximizar a mesma frequência $f = 1/3$ do teste anterior. Em seguida, anotamos os valores dos dinucleotídeos retornados pelo AG a cada rodada. A figura 10 mostra os resultados desse teste. Em comparação com o teste anterior, percebe-se um leve aumento na variação dos valores, para alguns dinucleotídeos. Entretanto, mais uma vez, a técnica se mostrou eficiente em manter dentro de uma faixa os valores atribuídos aos dinucleotídeos quando se testa a maximização de uma frequência numa mesma sequência de DNA várias vezes. Isso significa que os valores tendem a convergir especificamente para otimizar a frequência desejada. Com estes dois testes simples, concluímos que os resultados independem tanto da aleatoridade dos valores atribuídos aos genes dos indivíduos na população inicial quanto da aleatoridade da sequência de DNA utilizada. Portanto, nosso AG possui o direcionamento correto para retornar os melhores valores para os dinucleotídeos. Ficou claro que o método retorna valores consistentes e robustos.

O próximo passo, então, é avaliar os valores de *fitness* ao longo do DNA.

4.1.2 Variação dos valores de *fitness*

Utilizando a mesma idéia do teste anterior, o objetivo agora é analisar se em uma sequência de DNA normal e na mesma sequência aleatorizada os valores de *fitness* irão

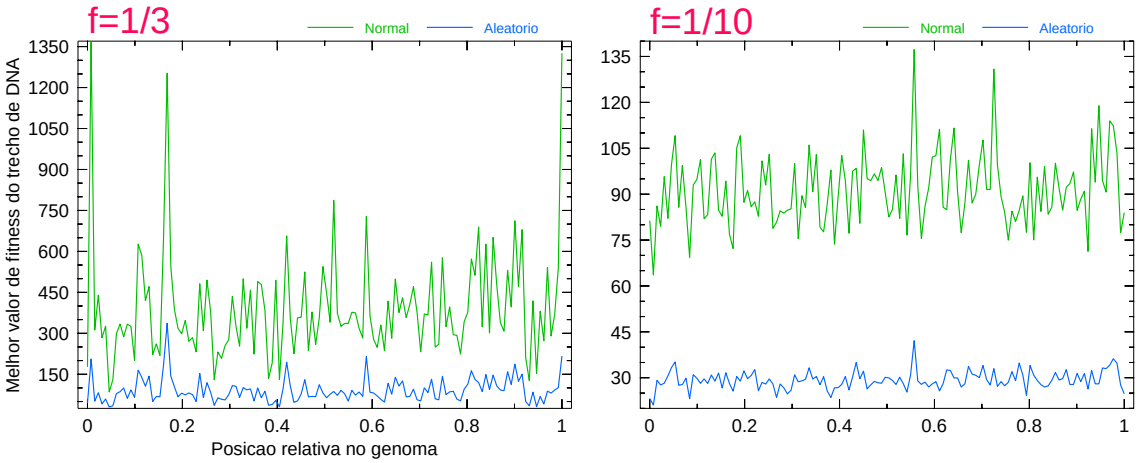

Figura 11
Para descobrir se os valores de *fitness* retornados pelo nosso AG possuem um significado lógico, nós buscamos otimizar as frequências $f = 1/10$ e $f = 1/3$, correspondentes aos períodos 10 e 3 respectivamente, num corte de 100 mil nucleotídeos do cromossomo 8 do *Plasmodium falciparum*. Nós aleatorizamos o corte de DNA original e também aplicamos nosso AG nestas sequências para mostrar que o valor de *fitness* retornado pelo AG possui um significado lógico. Ao longo do genoma, os valores de *fitness* do corte de DNA original variam muito para ambas as frequências procuradas, diferentemente da sequência aleatorizada.

sofrer os mesmos graus de variação. Neste teste, aplicamos nosso AG nos 100 mil nucleotídeos separados do cromossomo 8 do *Plasmodium falciparum* em busca da maximização das frequências $f = 1/3$ e $f = 1/10$. Para comparar, repetimos este mesmo teste com o trecho de DNA completamente aleatorizado. Uma vez que o valor de *fitness* é o resultado da função mérito, que traduz o quão satisfatória foi a maximização para a frequência desejada naquele trecho de DNA, percebe-se claramente que o DNA não-aleatório apresenta valores muito superiores e mais variáveis que a sequência aleatória. Isso acontece para ambas frequências.

Nas sequências aleatórias, percebe-se que os valores de *fitness* são muito menores quando se procura pela frequência $f = 1/10$. Isso acontece, provavelmente, porque ao se aleatorizar uma sequência de DNA, corre-se o risco, mesmo que mínimo, de produzir uma periodicidade. Entretanto, é mais fácil produzir acidentalmente uma periodicidade de 3 nucleotídeos que uma de 10 nucleotídeos. Com isso, em algumas partes, pode-se notar picos onde o valor de *fitness* é maior para $f = 1/3$ na sequência aleatorizada.

Estes testes nos deram confiança para começar as maximizações no *P. falciparum* e *D. melanogaster*, pois mostraram que o método é robusto e direciona os resultados exatamente para o objetivo do usuário. Os valores dos dinucleotídeos e os valores de *fitness* mostraram-se identificados com o objetivo da técnica, que é encontrar os melhores valores para maximizar uma frequência no DNA.

4.2 Otimizando periodicidades fracas

Uma vez que implementamos nosso AG com sucesso, como mostrado nas seções anteriores, o próximo passo é aplicar a técnica em sequências de DNA buscando por periodicidades fracas, buscando maximizá-las. Para isso, selecionamos os cortes de 100 mil nucleotídeos dos cromossomos 8 do *P. falciparum* e do cromossomo 2L da *D. melanogaster*, usados nos testes anteriores.

Para enriquecer as análises, utilizaremos três tipos de tabelas de conversão do DNA para fazer a transformada de Fourier discreta (DFT). Todas elas já foram explicadas com detalhes na Metodologia, seções 3.4 e 3.5, páginas 20 e 21. A primeira tabela de conversão é a que utiliza o mapeamento binário de dinucleotídeos (DBI). Este tipo de mapeamento é o mais simples de todos, e é usado na sequência de DNA sem otimizações, para servir de comparação para os mapeamentos otimizados. Através dele, podemos ver se a frequência desejada encontra-se na sequência. Caso negativo, procuraremos otimizá-la utilizando dois tipos de tabelas de conversão: a que utiliza o mapeamento de dinucleotídeos com números reais assumindo apenas valores positivos, DM^+, e com valores positivos e negativos, DM.

4.2.1 Buscando pela periodicidade $p = 3$

Escolhemos testar primeiramente a maximização da frequência $f = 1/3$, correspondente à periodicidade $p = 3$, presente em regiões codificantes. Esta frequência aparece em praticamente todas as sequências de DNA, independentemente da posição cromossômica, exceto em algumas regiões não-codificantes, que obviamente não apresentarão este padrão. Por isso, nosso objetivo com este teste é procurar sequências de DNA em que esta frequência *não é encontrada* no mapeamento *binário de dinucleotídeos* e então, através da otimização pelos mapeamentos DM e DM^+, passe a ser observada no espectro de potências de Fourier (FPS).

Nossas análises foram feitas utilizando janelas de 1000 nucleotídeos. Uma vez que nossos trechos de DNA possuem 100000 nucleotídeos, ao final da execução o arquivo de saída conterá 100 valores otimizados dos nucleotídeos e os seus respectivos valores de *fitness* correspondentes a cada janela. A escolha pela partição da sequência de DNA em janelas foi para aumentar o número de resultados, e assim, ter uma melhor base para comparar as otimizações ao longo do genoma.

Um exemplo de uma maximização realizada com sucesso está mostrado na figura 12. Esse trecho corresponde às posições 72000-73000 do trecho de 100 mil nucleotídeos extraídos do cromossomo 8 do *P. falciparum*. Para facilitar a visualização da frequência $f = 1/3$, que corresponde a $f = 0.33$, nós aplicamos um *zoom* na região correspondente

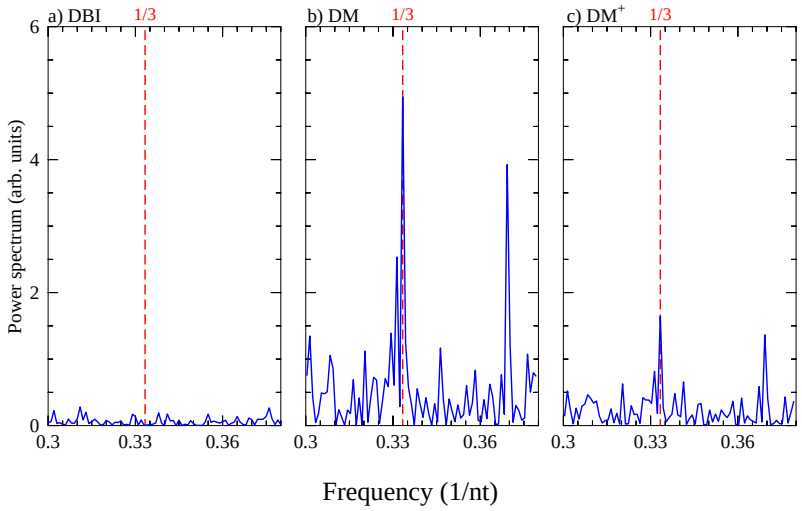

Figura 12
Otimização da frequência $f = 1/3$, correspondente ao período $p = 3$, encontrado em regiões de DNA que codificam para proteínas. Este trecho corresponde às posições 72000-73000 do corte de 100000 nucleotídeos do cromossomo 2L da *D. melanogaster*. Observamos a inexistência da periodicidade no primeiro painel, (a), relativo ao clássico mapeamento binário de dinucleotídeos (DBI). Nos painéis seguintes, (b) e (c), observa-se a clara maximização da periodicidade. No painel (b), referente ao mapeamento *DM*, que utiliza números reais positivos e negativos em sua escala, a maximização é mais pronunciada que no painel (c), que utiliza o mapeamento DM^+, apenas com números reais positivos. O FPS completo está mostrado na figura 29 do Apêndice, página 59.

a esta frequência no FPS. Por isso, os gráficos para esta frequência terão a escala do eixo x configurada de $f = 0.30$ até $f = 0.39$. No painel (a) da figura, encontra-se o FPS para o mapeamento binário de dinucleotídeos (DBI). Percebe-se claramente, com o desenho da linha tracejada vertical, que não existe nenhum sinal detectável na região referente à frequência $f = 1/3$. No FPS completo, percebe-se que não existe nenhuma outra frequência mais pronunciada, que poderia ter suprimido de certo modo o aparecimento de um pico em $f = 1/3$. Assim, a periodicidade $p = 3$ aparentemente não existe neste trecho do genoma. Após aplicar nosso AG nesta mesma sequência de DNA utilizando os mapeamentos *DM* e DM^+, cujos resultados estão mostrados nos painéis (b) e (c) da figura, percebe-se a clara maximização da frequência $f = 1/3$. A situação desses painéis, em que há uma aparência de desordem no FPS, com vários picos aproximadamente na mesma altura e mesma intensidade é chamada *ruido*. O painel (b) claramente possui uma maximização mais pronunciada que o painel (c). Nós observamos essa tendência em todos os nossos resultados.

Apesar de maximizar com sucesso a periodicidade $p = 3$, o mapeamento DM^+ possui picos com intensidade menor no FPS em comparação com o mapeamento *DM*.

Outra análise que fizemos é ver a flutuação dos valores dos dinucleotídeos ao longo das janelas do genoma. Como demonstrado na primeira seção dos Resultados, os valores dos dinucleotídeos são *direcionados* para maximizar uma frequência específica. Portanto, é esperado que os valores mudem de acordo com o trecho de DNA analisado. Para a maximização da frequência $f = 1/3$ nos 100 mil nucleotídeos da *D. melanogaster*, observe a variação dos valores para os dois mapeamentos, *DM* e DM^+, mostrados respectivamente nas figuras 13 e 14.

O eixo horizontal corresponde à posição relativa no corte de 100 mil nucleotídeos, e no eixo vertical temos os valores atribuídos aos dinucleotídeos. Com o objetivo de facilitar a visualização dos dados, nós selecionamos uma janela a cada 10000 nucleotídeos, totalizando assim 10 janelas. Nós dividimos os dinucleotídeos em 3 categorias, de acordo com o número de ligações de hidrogênio que fazem. Inicialmente, a divisão foi feita dessa forma porque estávamos interessados em descobrir se a contribuição de certos tipos de dinucleotídeos na otimização da frequência era diferente que outros tipos. Nós pensamos nesta possibilidade porque o genoma do *P. falciparum* possui um alto conteúdo AT [107], e investigar o papel dos dinucleotídeos envolvendo A e T poderia render uma análise interessante. Entretanto, esta análise inicialmente não nos retornou um resultado significativo. Por isso, continuamos dividindo os dinucleotídeos no gráfico de acordo com seu número de ligações de hidrogênio apenas por simplicidade. Os dinucleotídeos com 5 ligações são o grupo de maior número, por isso decidimos separá-los em dois painéis também com o objetivo de facilitar a visualização. Observe que para os dois tipos de mapeamentos, as variações dos valores são intensas ao longo do trecho completo de DNA. As variações são mais intensas para o mapeamento *DM* em comparação ao DM^+, como já pontuado anteriormente. Percebe-se claramente que o mapeamento *DM* possui os resultados mais expressivos, como mostrado na figura 12.

Então por quê utilizamos ambos mapeamentos, *DM* e DM^+ nas nossas análises? A figura 14 fornece uma explicação. Observe que a variação dos valores para o mapeamento DM^+ também é intensa. Entretanto, nota-se que muitos dinucleotídeos tendem a manter seus valores próximos a zero por vários trechos. Quando um dinucleotídeo tem seu valor em torno de 0 (zero), significa que o seu *peso*, ou seja, sua *contribuição* para a maximização da frequência em questão naquele trecho de DNA é *pouco relevante*. Levando em consideração essa afirmação, as análises do que significam os valores atribuídos aos dinucleotídeos se tornam mais simples. Quando um dinucleotídeo assume um valor em torno de 1 (um), sua contribuição para a maximização da frequência é grande. O AG atribui um peso maior exatamente por notar que sua relevância é maior. Por isso a interpretação do mapeamento DM^+ é considerada mais intuitiva. É importante considerar que esta análise é válida apenas para o mapeamento DM^+. No mapeamento DM, o fato

de aceitar valores positivos e negativos torna a interpretação dos valores um pouco menos intuitiva. Ao receber um valor negativo, não significa que aquele dinucleotídeo seja totalmente irrelevante. A interpretação dos valores reais retornados pelo mapeamento DM é um tópico que ainda necessita ser estudado com mais cuidado, que é uma das perspectivas futuras do nosso trabalho.

A fim de aplicarmos nossos testes com o AG em um organismo diferente da *D. melanogaster*, nós escolhemos o genoma do parasito da malária, *Plasmodium falciparum*, organismo cujo genoma é bem conhecido no meio acadêmico. O motivo da nossa escolha foi a complexidade de seu DNA e a diferença notável de estrutura em comparação ao da *D. melanogaster*, como por exemplo seu alto conteúdo AT e a ocorrência de harmônicos de frequência em seu FPS [3, 14]. Por se tratar de genomas tão diferentes, havia uma dúvida se as maximizações do *P. falciparum* com nosso AG traria resultados com a mesma qualidade que os gerados para a *D. melanogaster*. Nós aplicamos este mesmo teste para o corte de 100000 nucleotídeos do cromossomo 8 do *P. falciparum*. Separamos o conteúdo em janelas de 1000 nucleotídeos, aplicamos o nosso AG e uma lista contendo os valores dos dinucleotídeos e os valores de *fitness* foi retornada para cada uma das 100 janelas. O trecho que corresponde às posições 82000-83000 mostrou uma maximização interessante para $f = 1/3$ e encontra-se na figura 15. O mapeamento DBI não mostra qualquer pico na região referente à $f = 1/3$. Percebe-se que não há qualquer periodicidade visível neste FPS. Entretanto, após aplicar o nosso AG, a frequência aparece otimizada nos picos em $f = 1/3$ dos painéis (b) e (c). Por mais que a maximização do DM^+ seja menos pronunciada, quando comparada ao DBI, a maximização é extremamente eficiente.

Por outro lado, existem alguns trechos de DNA em que a maximização simplesmente não ocorre. É o caso do trecho correspondente às posições 64000-65000 do genoma do *P. falciparum*. Observe na figura 16 que nenhum dos três painéis exibem um pico considerável em $f = 1/3$. Considerando que existem regiões do DNA que não codificam para proteínas, este trecho de 1000 nucleotídeos provavelmente é um exemplo de sequência desse tipo. A periodicidade $p = 3$, apesar de ser de fácil detecção, não consegue ser otimizada em qualquer dos mapeamentos que construímos para este trecho. Note que mesmo no FPS completo nenhuma frequência diferente é encontrada.

A análise dos valores dos dinucleotídeos para a frequência $f = 1/3$ no *P. falciparum* também mostra resultados interessantes. As figuras 17 e 18 mostram as variações dos valores para a maximização da frequência $f = 1/3$ em toda a extensão dos 100 mil nucleotídeos. Assim como foi feito para a *D. melanogaster*, nas figuras está uma amostragem de 10 pontos dentre os 100 retornados pelo AG, que correspondem a posições a cada 10 mil nucleotídeos.

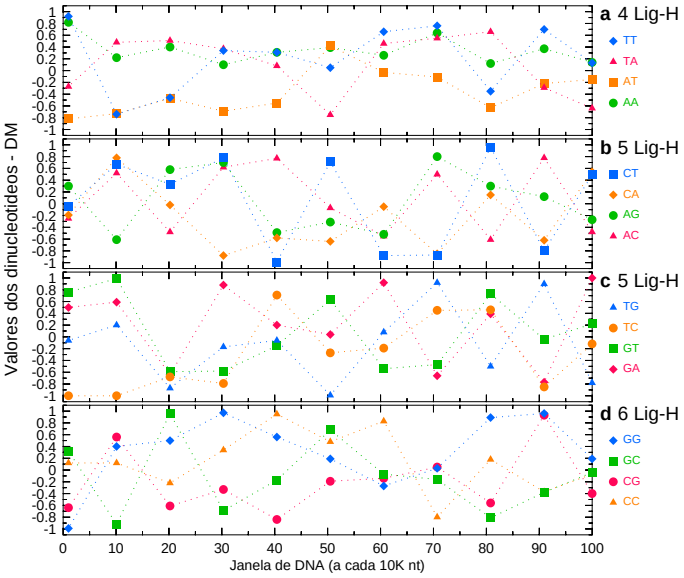

Figura 13
Variação dos valores dos dinucleotídeos retornados para o mapeamento *DM* buscando a maximização da frequência $f = 1/3$ na *D. melanogaster*. O primeiro painel (a) contém os dinucleotídeos com 4 ligações, os dois próximos painéis (b) e (c) os dinucleotídeos com 5 ligações e o último, (d), com 6 ligações.

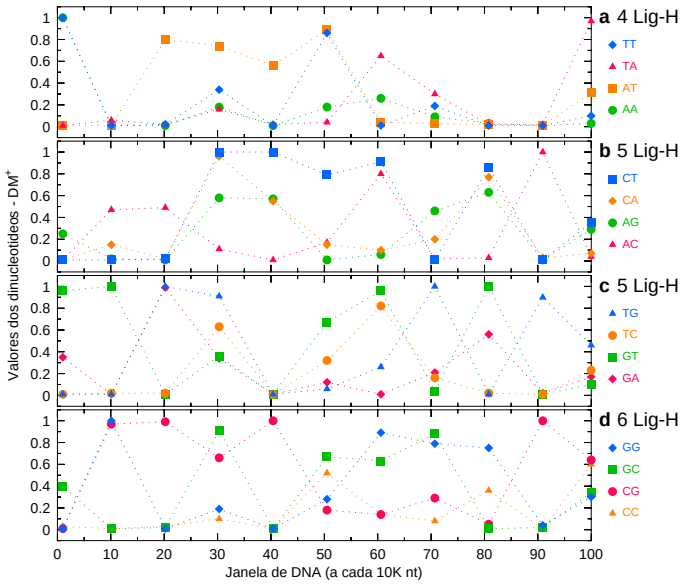

Figura 14
Flutuação dos valores reais atribuídos aos dinucleotídeos utilizando o mapeamento DM^+ para os 100 mil nucleotídeos do cromossomo 2L da *D. melanogaster*. O primeiro painel (a) contém os dinucleotídeos com 4 ligações, os dois próximos painéis (b) e (c) os dinucleotídeos com 5 ligações e o último, (d), com 6 ligações.

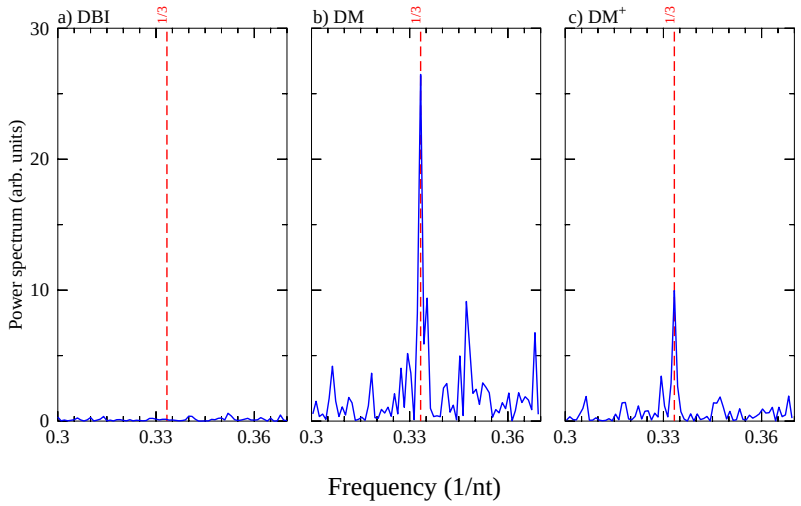

Figura 15
Trecho correspondente às posições 82000-83000 do corte de 100000 nucleotídeos do cromossomo 8 do *P. falciparum*, mostrando a maximização da periodicidade $p = 3$. O FPS completo desse teste encontra-se na figura 30 do Apêndice, página 59.

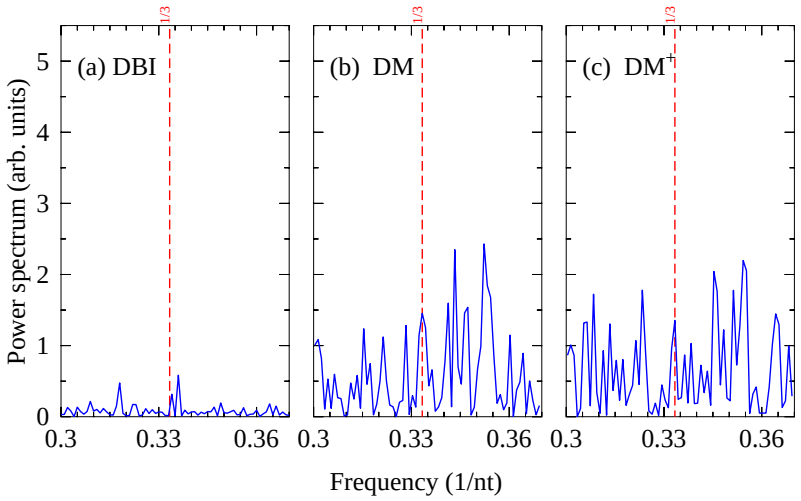

Figura 16
Exemplo de maximização em que não se encontra nenhum pico em $f = 1/3$ no trecho de 64000-65000 nucleotídeos do corte do cromossomo 8 do *P. falciparum*. Observe que nenhum dos painéis exibe qualquer pico na região de $f = 1/3$. Os mapeamentos otimizados *DM* e DM^+ exibem picos em meio a muito ruido no FPS. O FPS completo para este trecho encontra-se no Apêndice, na figura 33, página 61.

4.2.2 Revelando a periodicidade $p = 10$

Como descrito na Introdução, a periodicidade de 10 nucleotídeos possui vários significados biológicos, sendo o mais clássico a correspondência com a quantidade de nu-

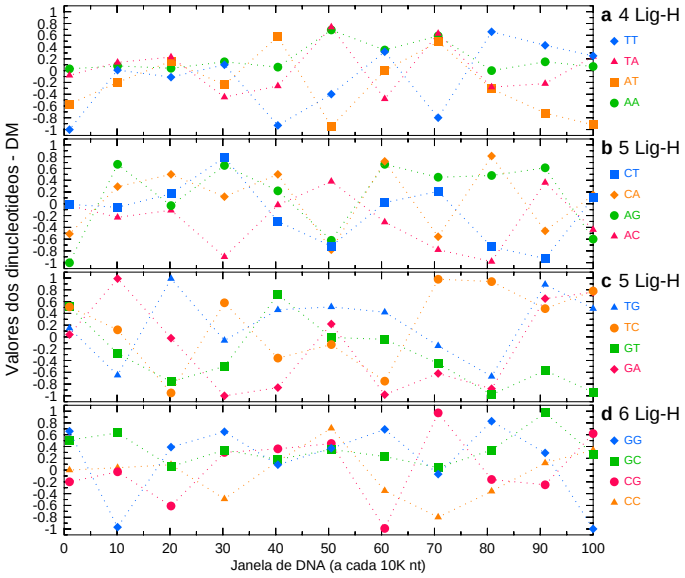

Figura 17
Variação dos valores dos dinucleotídeos para a maximização da periodicidade $p = 3$ nos 100 mil nucleotídeos do *P. falciparum*. O primeiro painel (a) contém os dinucleotídeos com 4 ligações, os dois próximos painéis (b) e (c) os dinucleotídeos com 5 ligações e o último, (d), com 6 ligações. Observe que a variação é grande para todo o trecho de DNA.

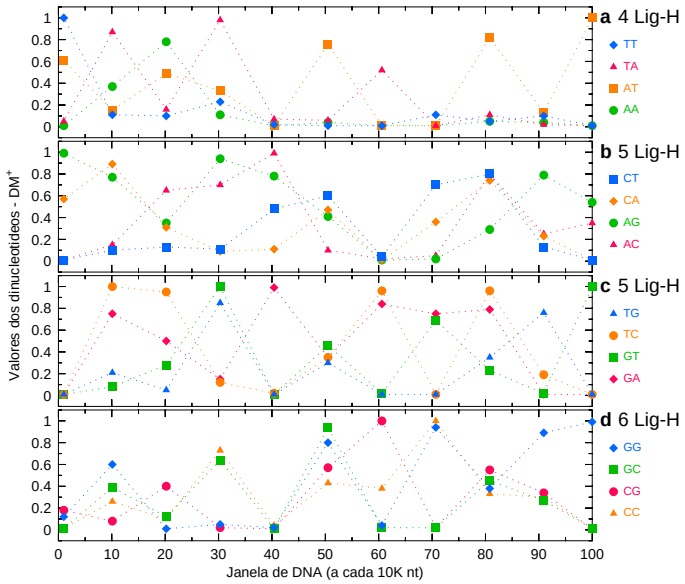

Figura 18
Flutuação dos valores dos dinucleotídeos para a maximização do período 3 no *P. falciparum*, utilizando o mapeamento DM^{+}. O primeiro painel (a) contém os dinucleotídeos com 4 ligações, os dois próximos painéis (b) e (c) os dinucleotídeos com 5 ligações e o último, (d), com 6 ligações.

cleotídeos necessários para conferir uma volta da hélice dupla do DNA. Além disso, esta periocidade é relacionada ao posicionamento dos nucleossomos nas regiões ao redor do sítio de inciação da transcrição (TSS). Entretanto, é a periodicidade $p = 10$ é mais fraca que $p = 3$, e portanto, mais difícil de ser detectada.

Para maximizar a periodicidade $p = 10$, procedemos da mesma forma que para a periodicidade $p = 3$. Nós aplicamos nosso AG nos trechos de 100 mil nucleotídeos dos organismos *P. falciparum* e *D. melanogaster*, utilizando janelas de 1000 nucleotídeos. Após o AG finalizar sua otimização, montamos os 100 FPS referentes às janelas e analisamos as otimizações da frequência $f = 1/10$, correspondente ao período $p = 10$.

Assim como foi feito para as maximizações da periodicidade $p = 3$, nós aplicamos um *zoom* no eixo x dos gráficos da otimização de $p = 10$. A frequência correspondente à periodicidade em questão é $f = 1/10$, que é igual a $f = 0.10$. Por isso, ajustamos a escala do eixo x para apresentar valores entre 0.08 e 0.13.

A figura 19 mostra um bom exemplo de maximização, correspondente às posições 44000-45000 do trecho de DNA da *D. melanogaster*. No painel (a), nota-se que o mapeamento DBI não detectou nenhum pico na frequência $f = 1/10$. Entretanto, após aplicado nosso AG, a periodicidade passa a ser visível, como mostrado nos painéis (b) e (c).

Veja as figuras 20 e 21, que mostram respectivamente os painéis de variação para os mapeamentos *DM* e DM^+. A variação dos valores dos dinucleotídeos mostrou variações importantes e seguiu o mesmo padrão dos apresentados até aqui. O mapeamento DM^+

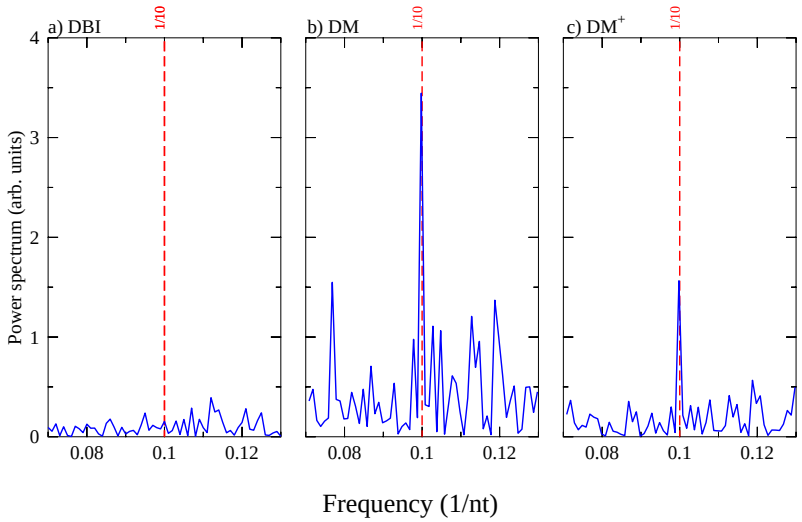

Figura 19
Trecho de DNA correspondente às posições 44000-45000 da *D. melanogaster* mostrando a otimização da periodicidade $p = 10$. O FPS completo encontra-se no Apêndice, figura 31, página 60.

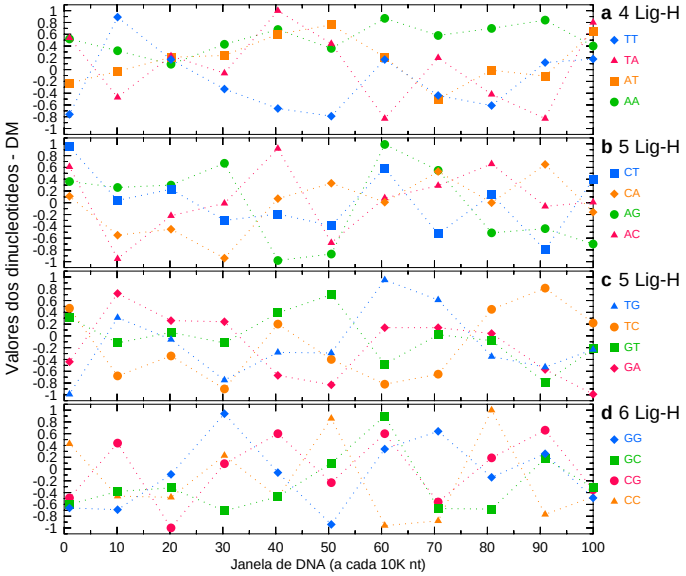

Figura 20
Valores dos dinucleotídeos do mapeamento *DM* retornados pelo AG para a maximização da periodicidade $p = 10$ em *D. melanogaster*. O primeiro painel (a) contém os dinucleotídeos com 4 ligações, os dois próximos painéis (b) e (c) os dinucleotídeos com 5 ligações e o último, (d), com 6 ligações.

mostra um padrão de variação um pouco menor em comparação com o mapeamento *DM*.

O resultado da maximização da periodicidade $p = 10$ para o *P. falciparum* também foi satisfatória, como mostra a figura 22. Este trecho é correspondente às posições 30000-31000 do trecho de DNA. Nesta figura, o pico no FPS do binário, no painel (a), é ainda menor que nas outras figuras. A maximização é claramente percebida nos painéis (b) e (c) quando aplicado o nosso AG com os mapeamentos otimizados.

A flutuação dos valores dos dinucleotídeos para esta maximização encontra-se nas figuras 23 e 24, que mostram respectivamente os valores para os mapeamentos *DM* e DM^+. Uma observação importante a se fazer com relação aos valores dos dinucleotídeos é que não é possível dizer se uma periodicidade foi maximizada ou não apenas tendo-os como base. Apenas com a montagem do FPS e a visualização do pico na frequência referente à periodicidade em questão que podemos ter a certeza da sua maximização.

Muitas vezes, ao mesmo tempo que o método é capaz de otimizar o mapeamento para uma frequência em particular, outros picos em outras frequências podem aparecer.

É importante considerar que apesar do mapeamento numérico retornado pelo AG ser específico para otimizar determinada frequência, outras periodicidades podem ser também maximizadas por acaso, através do mesmo mapeamento retornado. Isso acontece porque a combinação numérica para os 16 nucleotídeos, que serve para otimizar uma

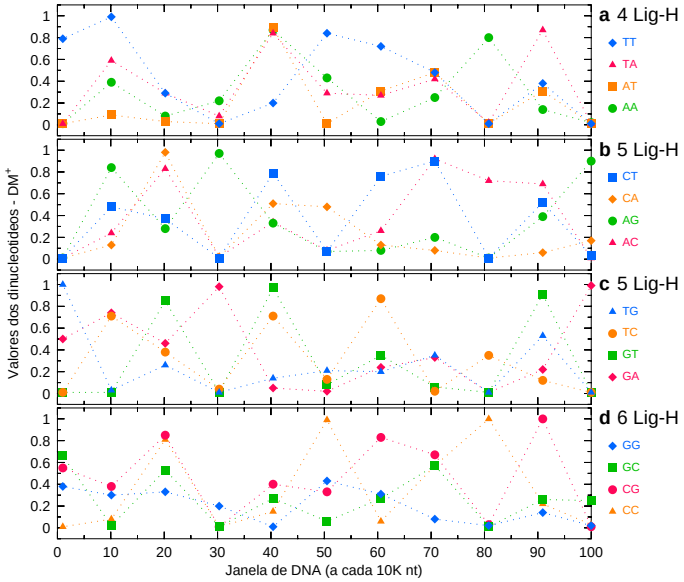

Figura 21
Variação dos valores do mapeamento DM^{+} retornados para a otimização do período $p = 10$ em *D. melanogaster*.

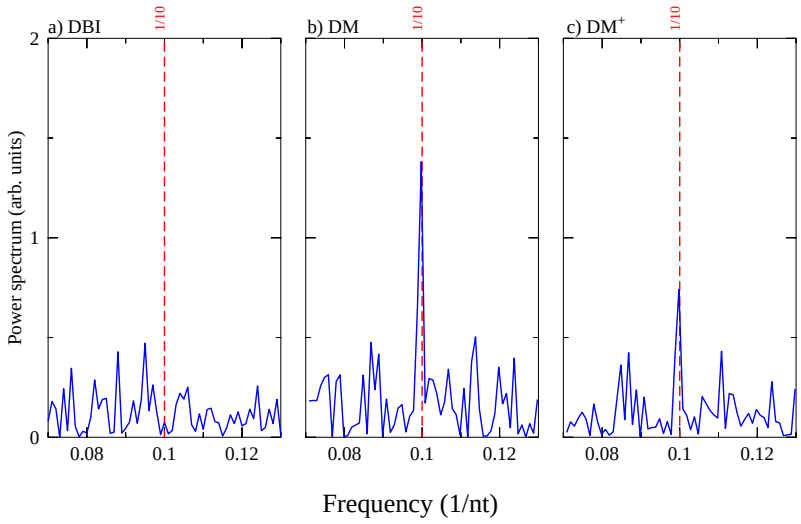

Figura 22
Otimização da frequência $f = 1/10$, relativa ao período $p = 10$ num trecho de DNA correspondente às posições 30000-31000 no DNA do parasito *P. falciparum*.

periodicidade específica de interesse, ocasionalmente pode também servir para otimizar outra periodicidade que não estávamos esperando. Apesar de parecer uma situação negativa à primeira vista, ela não é indesejada, uma vez que estes picos que aparecem inesperadamente no FPS podem indicar outras periodicidades relevantes que não são detectadas normalmente com o FPS binário clássico.

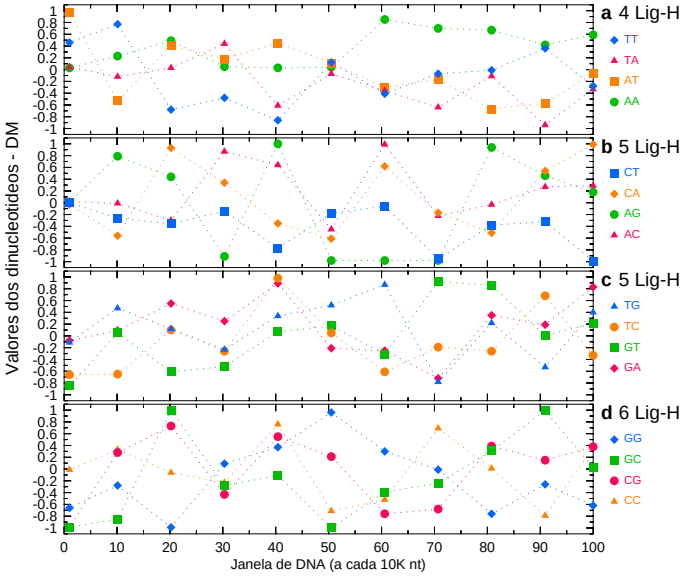

Figura 23
Flutuação dos valores dos dinucleotídeos para o mapeamento *DM* para a maximização do período $p = 10$ ao longo do trecho de 100000 nucleotídeos separados do cromossomo 8 do *P. falciparum*. O primeiro painel (a) contém os dinucleotídeos com 4 ligações, os dois próximos painéis (b) e (c) os dinucleotídeos com 5 ligações e o último, (d), com 6 ligações.

Um exemplo para essa situação encontra-se na figura 25, onde analisamos o trecho de 26000-27000 nucleotídeos, onde originalmente rastreamos pela frequência $f = 1/10$. O objetivo original deste teste é otimizar a periodicidade $p = 10$ e este foi alcançado, como mostra a linha pontilhada na cor cinza. Veja que o painel do mapeamento *DM* tem uma otimização satisfatória para essa frequência. Contudo, também encontramos uma periodicidade forte em $f = 1/16$, como mostra a linha pontilhada vermelha que encontra a periodicidade de modo exato nos painéis referentes aos mapeamentos *DM* e DM^{+}. Veja que o indicador binário, mostrado no painel (a), mostra um pico nesta área, entretanto é um pico fraco que se mistura aos outros picos do FPS, o que faz parecer ser apenas ruido. É possível que esta periodicidade esteja relacionada com algum significado biológico para o *P. falciparum*, entretanto nós não investigamos se esta periodicidade é específica para o *P. falciparum* ou se aparece nos outros organismos estudados neste trabalho. A investigação desta periodicidade ficou como perspectiva para trabalhos futuros.

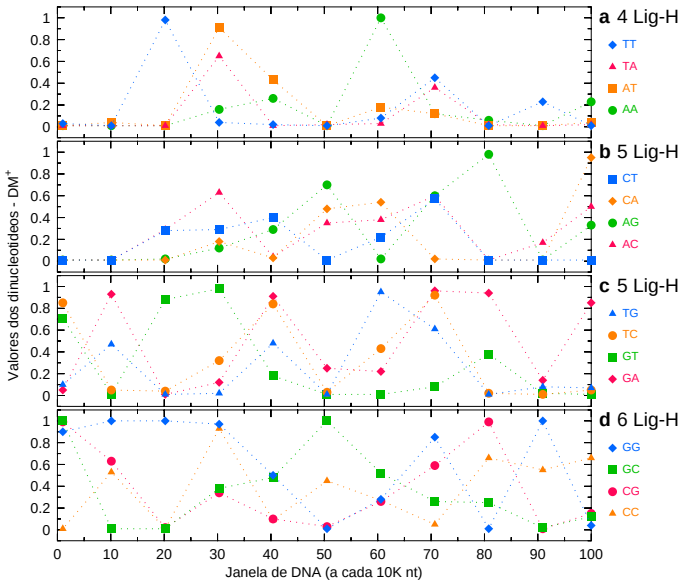

Figura 24
Flutuação dos valores dos dinucleotídeos do mapeamento DM^+ retornados para a maximização do período $p = 10$ ao longo dos 100 mil nucleotídeos do *P. falciparum*. O primeiro painel (a) contém os dinucleotídeos com 4 ligações, os dois próximos painéis (b) e (c) os dinucleotídeos com 5 ligações e o último, (d), com 6 ligações.

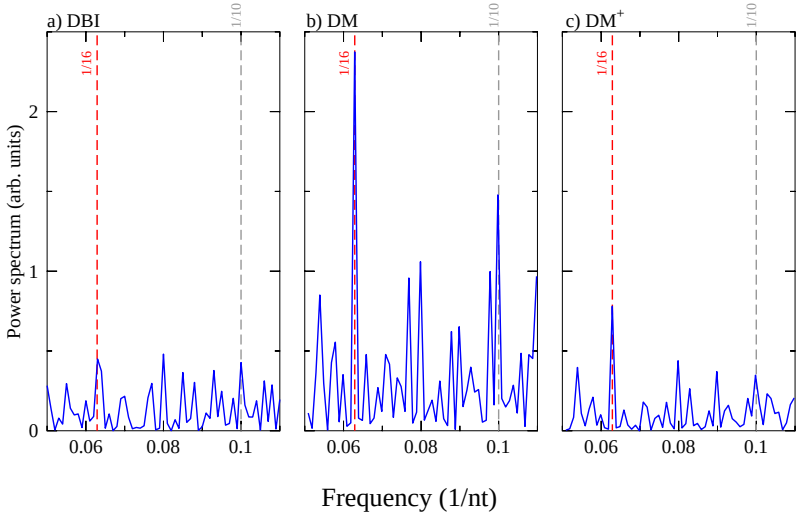

Figura 25
Ao realizar a maximização de $f = 1/10$, encontramos um trecho de DNA correspondente às posições 26000-27000 do corte de 100000 nucleotídeos do *P. falciparum* que contém uma periodicidade muito forte de 16 nucleotídeos. Ela aparece como um pico fraco no indicador binário, e quando a sequência é otimizada, o pico em $f = 1/16$ aparece bastante pronunciado no mapeamento *DM*.

4.3 Teste da produção de falsos positivos

Uma questão que ainda necessita ser respondida é se nosso AG pode maximizar qualquer periodicidade, estando esta presente ou não na sequência. Em outras palavras: é possível que nosso AG produza resultados falso-positivos, maximizando uma periodicidade que não existe de fato no DNA? O resultado prático de um falso-positivo seria um pico forte no FPS que não se relaciona com nenhuma periodicidade. Uma vez que resultados falsos positivos constituem um dos maiores problemas das técnicas presentes na literatura atualmente, será uma grande vantagem do método caso o nosso AG seja apto a produzir poucos ou até mesmo nenhum falso positivo.

Observe a figura 26, onde separamos um trecho de DNA correspondente às posições de 16000-17000 da *D. melanogaster* em que o pico na frequência $f = 1/10$ não é detectável, apesar das várias rodadas de otimização feitas naquele trecho de DNA. O AG faz várias tentativas de achar a periodicidade $p = 10$, mas não a encontra. Com isto, maximiza quase tudo ao redor, na tentativa de pegar a periodicidade desejada. Perceba que o pico em $f = 1/10$ é praticamente imperceptível nos três mapeamentos, DBI, *DM* e DM^+, ficando escondido entre o ruido. Para esse tipo de situação, podemos dizer com segurança que não há periodicidade $p = 10$ significativa no trecho de DNA selecionado.

Os valores dos dinucleotídeos retornados pelo AG ao terminar a análise de otimização para esse trecho de DNA são os melhores que o AG encontrou para maximizar $f = 1/10$,

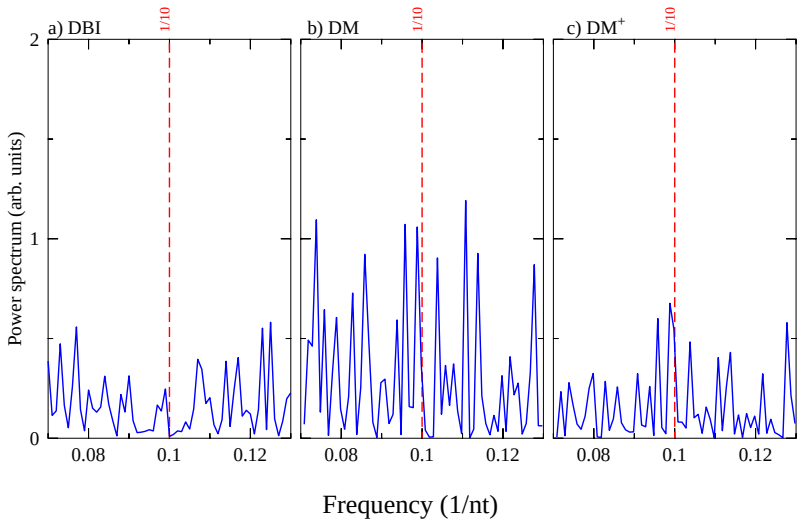

Figura 26
Tentativa de otimização da periodicidade $p = 10$ no trecho correspondente às posições 16000-17000 do trecho de DNA da *D. melanogaster*. Observe que em nenhum dos painéis o pico na frequência $f = 1/10$ aparece, mostrando que a periodicidade encontra-se inexistente nesse trecho de DNA.

mas não necessariamente significa que a periodicidade seria de fato otimizada. Se, porventura, em uma das rodadas, o AG tivesse encontrado um conjunto de valores aleatórios que maximizasse esta frequência, provavelmente ele teria sido selecionado e, ao final da execução, teria sido retornado ao usuário. Esse seria o caso se a periodicidade estivesse de fato *presente* na sequência. No caso mostrado na figura 26, mesmo após inúmeras tentativas de otimização, a periodicidade continua inexistente. Isso significa que quando uma periodicidade está *ausente* num trecho de DNA, o AG *não* achará um conjunto de valores ótimos para maximizar a periodicidade.

É importante pontuar que utilizamos sequências de DNA de apenas 1000 nucleotídeos para estas maximizações, tamanho considerado médio. Para sequências de DNA de tamanho maior, é quase impossível que pequenas periodicidades como a de 3 nucleotídeos estejam completamente ausentes. Isso ocorre porque pela própria natureza do DNA, alguns nucleotídeos irão se repetir periodicamente. Veja o exemplo mostrado na figura 16, que ilustra esta situação.

4.4 Resolvendo o problema da periodicidade nucleossomal

Como explicado com detalhes na seção 1.3 da Introdução, dois trabalhos recentes [36, 37] chegaram a resultados conflitantes a respeito da presença da periodicidade nucleossomal $p = 10$ na região próxima ao TSS em sequências promotoras de *Homo sapiens* e *Saccharomyces cerevisiae*. Primeiramente, o grupo de Tolstorukov *et al* não encontrou a periodicidade $p = 10$ nas regiões promotoras humanas. Entretanto, em promotores *S. cerevisiae* o autor encontra a periodicidade, que inclusive é bem pronunciada. Para estes testes foi utilizada a técnica de correlação. Um ano depois, o grupo de Hebert *et al* refez estas análises, desta vez utilizando DFT, e chegou a uma conclusão diferente, afirmando que a periodicidade encontra-se presente tanto em promotores humanos quanto de *S. cerevisiae*. Ambos trabalhos possuem argumentações sólidas e convincentes acerca de seus pontos de vista, o que dificulta ao leitor formar uma opinião conclusiva.

Após aplicar com sucesso nosso AG em vários organismos e mostrar aqui resultados surpreendentes na descoberta de periodicidades fracas, nós decidimos aplicá-lo nas sequências promotoras humanas em busca da periodicidade $p = 10$. Para isso, nós selecionamos as 9714 sequências promotoras de *H. sapiens* disponíveis no banco de dados EPD - *Eukaryotic Promoter Database* [78] contendo 150 nucleotídeos cada. Esses 150 nucleotídeos correspondem à região onde se posiciona o primeiro nucleossomo *downstream* ao TSS, e que engloba os nucleotídeos da posição +40 até +190, região chamada de "nucleossomo +1". Nesta região, o sinal da periodicidade $p = 10$ é mais forte, e é relacionado ao posicionamento dos nucleossomos no DNA [39].

Devido à impossibilidade de plotar os FPS dos 9714 promotores, nós baseamos nossas análises no valor de *fitness* retornado pelo AG para a otimização da periodicidade $p = 10$. Como relatado nas seções anteriores, o valor de *fitness* significa o quão satisfatória foi a otimização em relação aos outros picos do FPS. Grandes valores de *fitness* provavelmente significam uma boa maximização.

Para facilitar a análise de cada valor de *fitness*, nós montamos um histograma, mostrado na figura 27. Os 9714 valores de *fitness* para cada promotor foram separados em 10 faixas distintas. Então, contamos quantos promotores se encaixam em cada faixa. A tabela 5 mostra o valor de *fitness* que foi utilizado para separar cada uma das faixas e a quantidade de promotores que se encontram nelas.

Observe que na curva relativa ao DBI, a maioria absoluta das sequências promotoras pertence aos 3 primeiros grupos, onde o *fitness* não ultrapassa o valor de 1.8. O restante das outras faixas apresentam sequências com valores de *fitness* também considerados baixos. Isso significa que o pico na frequência $f = 1/10$ não foi encontrado para estas sequências, justificando os valores baixos. Apenas uma única sequência promotora do mapeamento DBI teve um valor de *fitness* considerado alto, de aproximadamente 11.38.

Os valores de *fitness* das curvas correspondentes aos mapeamentos *DM* e DM^+ são muito maiores em comparação ao mapeamento DBI, principalmente os valores do mapeamento *DM*, como já era esperado. As curvas correspondentes aos mapeamentos otimizados possuem um padrão muito similar. Os valores de *fitness* do grupo *DM* são aproximadamente o dobro do valor do DM^+, e por isso pode-se dizer que a maximização é mais satisfatória. Algumas faixas de valores desses mapeamentos ultrapassam bem mais que o dobro do maior valor de *fitness* encontrado pelo DBI, com faixas superando o valor de 35 no mapeamento *DM*. O FPS dessas sequências mostram picos consideráveis na frequência $f = 1/10$. Estes resultados mostram que a absoluta maioria dos promotores humanos analisados pelo nosso AG possui de fato a periodicidade $p = 10$, e assim dá suporte ao trabalho publicado pelo grupo de Hebert *et al*.

A última pergunta que nos surgiu ao observar nossos resultados é: as otimizações pelos mapeamentos *DM* e DM^+ apenas realçam os resultados do DBI ou produzem novidades? Realçar é quando um valor de *fitness* alto retornado pelo DBI continua alto depois da maximização pelo *DM* ou DM^+. Caracterizamos como novidade quando um valor de *fitness* baixo no DBI torna-se alto nos mapeamentos *DM* ou DM^+ depois da otimização.

Para fazer este teste, nós aplicamos o método da regressão linear para comparar cada um dos 3 mapeamentos. A regressão linear compara dois grupos de dados e mostra o quanto eles são correlacionados através do coeficiente de regressão linear, chamado de R^2. Este coeficiente varia numa escala de 0 a 1. Quanto mais próximo de 1 for o coeficiente, significa que os dados são mais correlacionados. Adaptando a idéia ao nosso projeto,

Faixa	DBI		DM^+		DM	
	fitness até	Quant.	*fitness* até	Quant.	*fitness* até	Quant.
1	0	2877	1.88	341	3.85	126
2	1.26	3876	3.56	2838	7.03	1864
3	2.52	1871	5.23	3805	10.21	3803
4	3.79	747	6.90	1972	13.40	2669
5	5.05	242	8.57	619	16.58	946
6	6.32	75	10.25	119	19.76	241
7	7.58	21	11.92	17	22.95	57
8	8.85	3	13.59	1	26.13	5
9	10.11	1	15.26	1	29.32	2
10	11.38	1	20.28	1	35.68	1

Tabela 5
Nós dividimos os valores de *fitness* retornados para os 9714 promotores humanos em faixas para facilitar a análise dos resultados. Cada um dos três mapeamentos foi dividido em 10 faixas.

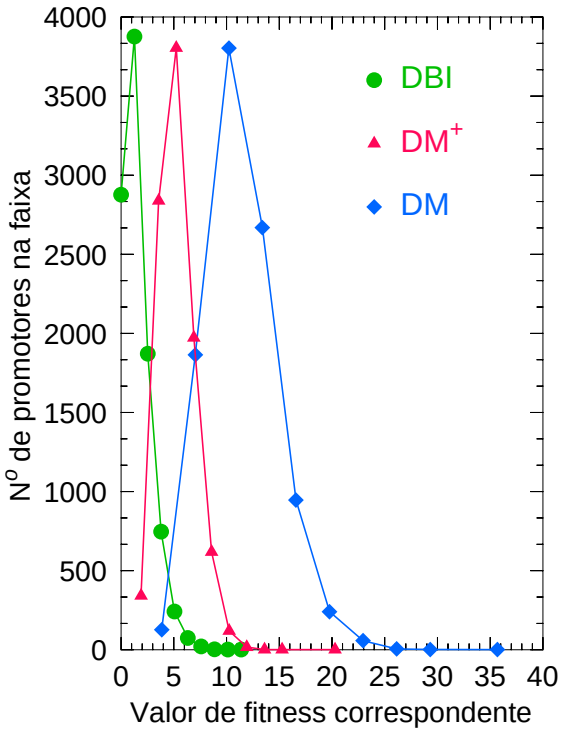

Figura 27
Histograma mostrando a distribuição dos 9714 promotores humanos em relação ao seus valores de *fitness*. O mapeamento DBI possui os piores valores de *fitness*, enquanto que os mapeamentos otimizados DM e DM^+ apresentam valores maiores na maximização da periodicidade $p = 10$.

quanto mais próximo de 1 for o coeficiente, mais os mapeamentos serão correspondentes.

O resultado encontra-se na figura 28. No primeiro painel onde temos a comparação entre os valores de *fitness* do mapeamento DBI com o *DM*, o coeficiente de correlação foi $R^2 = 0.53$. A linha pontilhada cinza desenhada no gráfico indica a tendência que os valores se distribuem. Todos os valores acima da linha pontilhada correspondem a promotores em que os valores de *fitness* foram realçados. Abaixo da linha pontilhada, temos as novidades: todos estes promotores tinham valores de *fitness* baixos, e depois da maximização se tornaram altos.

No segundo painel temos a comparação dos valores de *fitness* do DBI com o DM^+. O coeficiente de correlação é $R^2 = 0.3$, sugerindo que os dois mapeamentos tem uma correlação menos pronunciada em relação ao mapeamento *DM*. Neste painel, os pontos do gráfico tendem a tomar uma distribuição mais vertical em comparação ao painel anterior. Isso significa que o mapeamento DM^+ tem uma tendência maior a realçar os valores de *fitness* dos promotores. Entretanto há também alguns pontos na faixa que correspondem às novidades, apesar de bem menos frequentes em relação ao mapeamento *DM*.

O terceiro painel é o que compara os mapeamentos *DM* e DM^+ e teve o maior coeficiente de correlação: $R^2 = 0.64$. Este resultado nos mostra que os resultados alcançados por ambos mapeamentos são correspondentes. A grande maioria dos valores de *fitness* é realçado ou tem valores proporcionais. Entretanto, mesmo sendo correspondentes, há diferenças importantes entre os mapeamentos. Caso o coeficiente de correlação retornado

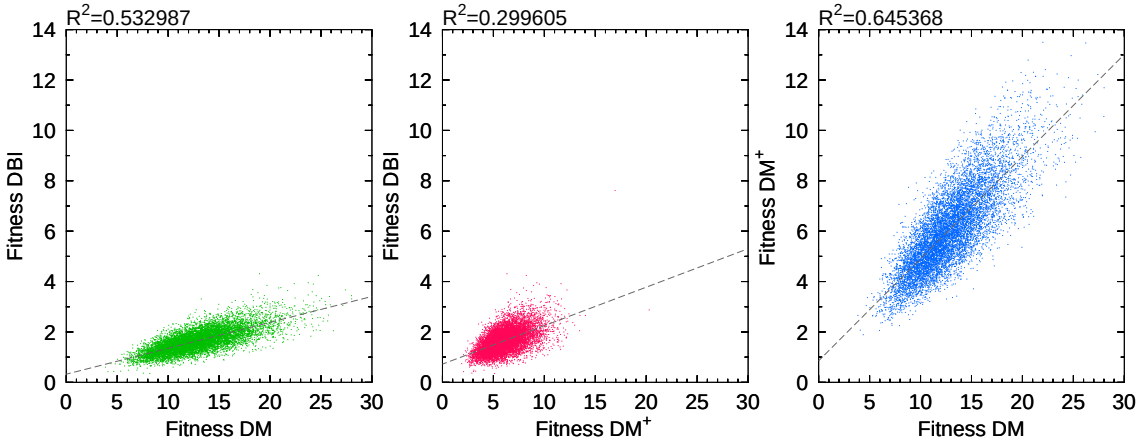

Figura 28
Correlação existente entre os valores de *fitness* retornados pelos três diferentes mapeamentos utilizados neste trabalho: DBI, *DM* e DM^+. O painel (a) mostra a correlação dos valores de *fitness* entre os mapeamentos DBI e *DM*. O painel (b) mostra a correlação entre DBI e DM^+. No painel (c), a correlação entre DM^+ e *DM*.

fosse igual a 1, significaria que os mapeamentos seriam totalmente correspondentes e não faria sentido utilizar ambos em nossas análises. A justificativa é que alguns pontos abaixo da linha pontilhada indicam que há maximizações novas que não foram captadas pelo mapeamento DM^+. Por isso, esta diferença faz com que o mapeamento DM continue sendo considerado mais eficiente nas maximizações.

Caso o coeficiente de correlação tivesse sido $R^2 = 1$ para todos os testes, significaria que o nosso AG apenas realçou os resultados do DBI, e consequentemente, não haveria nenhuma novidade. Nossos resultados mostram o contrário: os mapeamentos otimizados DM e DM^+ apresentaram coeficientes variados, que nos renderam análises positivas quanto ao nosso AG.

Este último teste nos mostrou mais uma vez a importância de cada um dos mapeamentos desenvolvidos para o nosso AG e também mostrou a eficiência em otimizar periodicidades fracas, confirmando a robustez do método.

5 Conclusão

Neste trabalho nós desenvolvemos uma nova técnica para detectar periodicidades fracas em genomas. Utilizando a transformada de Fourier discreta, técnica clássica para detecção de periodicidades, e um algoritmo genético, método de inteligência artificial apropriado para problemas de otimização, nós conseguimos construir um algoritmo robusto e eficiente para este fim.

Em nossos testes, utilizamos sequências de DNA de vários organismos, mas neste trabalho apresentamos resultados para 3 organismos em especial: *Plasmodium falciparum*, parasito da malária, escolhido devido à peculiaridade do seu genoma com alto teor de repetições, *Drosophila melanogaster* ou mosca-das-frutas, organismo-modelo largamente utilizado na ciência, e o *Homo sapiens*. As sequências de DNA utilizadas nestes testes foram de 1000 pares de base para os dois primeiros organismos, e 150 pares de base para as sequências humanas. Todos os resultados foram satisfatórios na otimização de periodicidades que antes não eram detectadas pela transformada de Fourier clássica.

Os grupos de Tolstorukov *et al* [36] e Hebert *et al* [37] recentemente publicaram informações conflitantes acerca da presença da periodicidade de 10 nucleotídeos em sequências promotoras de *H. sapiens*. Segundo o grupo de Tolstorukov, a periodicidade não está presente em promotores humanos, informação que contradiz vários outros trabalhos e coloca em dúvida o papel da periodicidade que antes acreditava ser um sinal para o posicionamento dos nucleossomos naquela região. O grupo de Hebert veio mais tarde questionar os resultados do primeiro grupo, dizendo que a periodicidade está de fato presente nos promotores humanos.

Tendo em vista este problema e buscando resolver a contradição instalada, nós aplicamos nosso novo método em sequências promotoras de *H. sapiens*. Nós constatamos que a periodicidade 10 está de fato presente nestas sequências.

Ao contrário do postulado por diversos trabalhos, a transformada de Fourier discreta pode ser feita com sucesso em sequências menores de DNA sem prejuízo da qualidade das detecções. Nós provamos isto ao detectar com sucesso a periodicidade 10 nas sequências promotoras humanas, que tinham um tamanho considerado muito pequeno, de apenas 150 pares de base. As otimizações da *Drosophila melanogaster* e do *Plasmodium falciparum* foram feitas em trechos de DNA de 1000 pares de base, e também apresentaram resultados surpreendentes.

Com isso, nós concluímos que o novo método proposto nesse trabalho é robusto, eficiente e confiável na busca por repetições fracas em genomas, contribuindo para a bioinformática no sentido de aumentar a confiabilidade dos resultados na busca de repetições.

6 Perspectivas Futuras

- Lançar um *website* que tornará a ferramenta desenvolvida neste projeto disponível para uso livre pela comunidade acadêmica, com dois propósitos principais:

    1. Detectar uma determinada periodicidade em uma sequência de DNA escolhida pelo usuário.
    2. Otimizar uma periodicidade específica, de interesse do usuário, através do nosso AG.

- Entender melhor o significado dos valores numéricos retornados para cada dinucleotídeo pelos dois mapeamentos otimizados desenvolvidos: DM e DM^{+}.

7 Apêndice

7.1 Algoritmos em Perl

Nesta seção estão os algoritmos que construí em linguagem Perl, para este trabalho.

7.1.1 Fourier binário dinucleotídeo

Calcula o Fourier binário dinucleotídeo de uma sequência de DNA.

```
#!/usr/bin/perl -w

use lib '/users/miriam/usr/local/lib/perl5/site_perl/5.12.1';
use lib '/users/miriam/Documents/mestrado/projeto/programas/modelos';
use procedures;
use Math::FFTW;
use Math::Trig;
use strict;

my $arq;
my $maxp;
my $res;

foreach my $arg (@ARGV)
{
  if ($arg =~ /-fasta=(.*)/)  {$arq=$1;}
  if ($arg =~ /-maxp=(.*)/)   {$maxp=$1;}
  if ($arg =~ /-result=(.*)/) {$res=$1;}
}

my @dinucs=("AA","AC","AG","AT","CA","CC","CG",
    "CT","GA","GC","GG","GT","TA","TC","TG","TT");
my @espectro_total;
our $dna=extrair_dados($arq);
$dna =~ tr/actg/ACTG/;

foreach my $dinuc (@dinucs)
{
  my @seq;
  for(my $a=0; $a<(length($dna)-1); $a++)
  {
    my $par=substr($dna,$a,2);
    if ($par eq $dinuc) { push(@seq,1); }
    else                { push(@seq,0); }
  }
  my @f=fourier(\@seq);
  for (my $i=0; $i < scalar(@f); $i++)
  { $espectro_total[$i] += $f[$i]; }
}
media(\@espectro_total,scalar(@espectro_total)/$maxp,$res);
exit;
```

7.1.2 Algoritmo Genético + Fourier

A partir de uma sequência de DNA, encontra os melhores valores reais para otimizar uma certa frequência através da transformada de Fourier. Retorna estes valores reais na forma de arquivo de texto.

```
#!/usr/bin/perl -w

use lib '/users/miriam/usr/local/lib/perl5/site_perl/5.12.1';
use lib '/users/miriam/Documents/mestrado/projeto/programas/modelos';
use procedures;
use Math::FFTW;
use Math::Trig;
use AI::Genetic;
use IO::Handle;
use strict;

my $input="";
my $output="";
my $start=0;
my $window=10000; # $window=150 para sequencias promotoras de H sapiens
my $step=10000;
my $freq=0;
my $dev=1;
my $scale=100;
my $positive=0;
my $generations=500;
my $pop=300;
my $dna="";
my $parada=0.1;
my $header=1;
our $len=0;
our $sseq="";
my @evol=();
our @pairlist=();
my %dinucs=('AA'=>0,'AC'=>1,'AG'=>2,'AT'=>3,'CA'=>4,'CC'=>5,'CG'=>6,'CT'=>7,
'GA'=>8,'GC'=>9,'GG'=>10,'GT'=>11,'TA'=>12,'TC'=>13,'TG'=>14,'TT'=>15);

foreach my $arg (@ARGV)
{
  if ($arg =~ /-input=(.*)/)            {$input=$1;}
  if ($arg =~ /-output=(.*)/)           {$output=$1;}
  if ($arg =~ /-start=(.*)/)            {$start=$1;}
  if ($arg =~ /-frequency=(.*)/)        {$freq=eval($1);}
  if ($arg =~ /-positive=(.*)/)         {$positive=$1;}
}

open(RES,">>$output"); RES->autoflush(1);
if ($header)
{ print RES "Begin End Fitness AA AC AG A CA CC CG CT GA GC GG GT TA TC TG TT\n"; }

$dna = extrair_dados($input) or die "Couldn't open \"$input\".\n";
$dna =~ tr/actg/ACTG/;
$dna =~ tr/nN/N/;
my $N=length($dna);
```

```
  die "Window $window bigger than genome size $N.\n" if ($window > $N);

  for (my $x=$start; $x<=($N-$window); $x+=$step)
  {
    $sseq = substr($dna,$x,$window);
    $len=length($sseq);

    if ($sseq !~ /[^N]/i)
    { print "Window at position $x contains only characters N and will be skipped.\n\n"; next; }

    @pairlist=();
    for (my $i=0; $i <= ($len-2); $i++)
    {
      my $pair=substr($sseq,$i,2);
      push(@pairlist,$pair);
    }
    if ($len < $window) {print "Substring $len smaller than window $window in position $x\n"; last;}

    my $ga = new AI::Genetic(
    '-fitness'    => \&fitnessFunc,
    '-type'       => 'rangevector',
    '-population' => $pop,
    '-crossover'  => 0.7,
    '-mutation'   => 0.01,
    '-terminate'  => \&terminateFunc,
    );

    if ($positive)
    {
      $ga->init([[1,$scale],[1,$scale],[1,$scale],[1,$scale],[1,$scale],[1,$scale],
      [1,$scale],[1,$scale],[1,$scale],[1,$scale],[1,$scale],[1,$scale],[1,$scale],
      [1,$scale],[1,$scale],[1,$scale]]);
    }
    else
    {
      $ga->init([[-$scale,$scale],[-$scale,$scale],[-$scale,$scale],[-$scale,$scale],
      [-$scale,$scale],[-$scale,$scale],[-$scale,$scale],[-$scale,$scale],[-$scale,$scale],
      [-$scale,$scale],[-$scale,$scale],[-$scale,$scale],[-$scale,$scale],[-$scale,$scale],
      [-$scale,$scale],[-$scale,$scale]]);
    }

    $ga->evolve('tournamentUniform', $generations);
    print RES $x, " ", $x+$window, " ", $ga->getFittest->score, " ";
    my $mult=1;
    $mult=-1 if ($ga->getFittest->genes->[$dinucs{'AA'}] < 0);
    for my $di (sort keys %dinucs)
    {
      print RES $mult*($ga->getFittest->genes->[$dinucs{$di}])/$scale," ";
      if ($di eq 'TT') {print RES "\n";}
    }
    undef $ga;
  }
  exit;

sub fitnessFunc
```

```
{
  my @seq;
  my $genes = shift;
  my @gen=@{$genes};
  for (my $i=0; $i < scalar(@gen); $i++)
    {
    $gen[$i] /= $scale;
    }
  for (my $i=0; $i < ($len-2); $i++)
  {
    my $pair=$pairlist[$i];
    if (exists $dinucs{$pair})
    { $seq[$i]=$gen[$dinucs{$pair}]; }
    else { $seq[$i]=0; }
  }
  my @transf=fourier(\@seq);
  my $sum=0;
  my $N=scalar(@transf);
  foreach my $t (@transf) {$sum += $t;}
  my $media=$sum/$N;
  my $res=0;
  my $fi=int($N*$freq*2);
  for (my $l=$fi-$dev; $l < $fi+$dev; $l++)
  { $res += $transf[$l];}
  my $fitness=$res/$media;
  return $fitness;
}

sub terminateFunc
{
  my $ga = shift;
  my $len=30;
  shift(@evol) if (scalar(@evol)>$len);
  push(@evol,$ga->getFittest->score);

  open(FIT,">>$output.fit");
  print FIT $ga->{GENERATION}," ", $ga->getFittest->score,"\n";
  close(FIT);

  if (scalar(@evol)>=$len)
  { return 1 if (abs($evol[$len-1]-$evol[0]) < $parada); }

  return 0;
}
```

7.1.3 Construção dos espectros de potência

Coleta os dados retornados pelo Algoritmo Genético + Fourier, descrito na seção 7.1.2 acima. Com os valores reais dos dinucleotídeos, aplica a transformada de Fourier e constrói os espectros de potência.

```
#!/usr/bin/perl -w

use lib '/users/miriam/usr/local/lib/perl5/site_perl/5.12.1';
use lib '/users/miriam/Documents/mestrado/projeto/programas/modelos';
use procedures;
use Math::FFTW;
use Math::Trig;
use File::Basename;
use strict;

my $gen="";
my $fasta="";
my $maxpontos=0;
our $res="";
my @seq=();
my $sseq="";
my $len=0;
my $count=0;
our $nome="";
our %genes_pos=('AA'=>0,'AC'=>1,'AG'=>2,'AT'=>3,'CA'=>4,'CC'=>5,'CG'=>6,'CT'=>7,
    'GA'=>8,'GC'=>9,'GG'=>10,'GT'=>11,'TA'=>12,'TC'=>13,'TG'=>14,'TT'=>15);

foreach my $arg (@ARGV)
{
  if ($arg =~ /-gen=(.*)/)      {$gen=$1};
  if ($arg =~ /-fasta=(.*)/)    {$fasta=$1};
  if ($arg =~ /-maxp=(.*)/)     {$maxpontos=$1};
  if ($arg =~ /-result=(.*)/)   {$res=$1};
}

open(GEN,$gen) or die "Nao foi possivel abrir seu arquivo gen $gen.\n";
our $seq = extrair_dados($fasta) or die "Nao foi possivel abrir o arquivo FASTA $fasta.\n";
$seq =~ tr/actg/ACTG/;
our $N=length($seq);

while (my $linha=<GEN>)
{
  if ($linha =~ /^Begin/) {next;}
  my @info=split(" ",$linha);
  my $x=$info[0];
  my $y=$info[1];
  my $janela=$y-$x;
  my @genes_linha=@info[3..19];
  my %genes;

  foreach my $x (sort keys %genes_pos)
  {$genes{$x}=$genes_linha[$genes_pos{$x}];}
  $nome=$res;
  $nome =~ (s/\.dat/-$x.dat/);
```

```
    $sseq = substr($seq,$x,$janela);
    @seq=0;
52  $len=length($sseq);
    if ($len < $janela) {print "Substring $len menor que janela $janela na posicao $x\n"; last;}
    my $naotrad=0;
    for (my $i=0; $i < $len-1; $i++)
    {
57    my $par=substr($sseq,$i,2);
      if (exists $genes{$par}) {push(@seq,$genes{$par});}
      else                     {push(@seq,0);$naotrad++;}
    }
    if ($naotrad) {print "Nao traduzidos=$naotrad, mas nao afeta a periodicidade.\n";}
62  my @transf = fourier(\@seq);
    media(\@transf,scalar(@transf)/$maxpontos,$nome);
  }
  exit;
```

7.2 Espectros completos

Na seção Resultados e Discussão, apresentamos os espectros de Fourier com *zoom* na região da frequência de interesse, com o objetivo de facilitar a visualização. Aqui, listamos as figuras com os espectros completos. Para a discussão dos resultados, por favor veja a sessão Resultados e Discussão, iniciada na página 30.

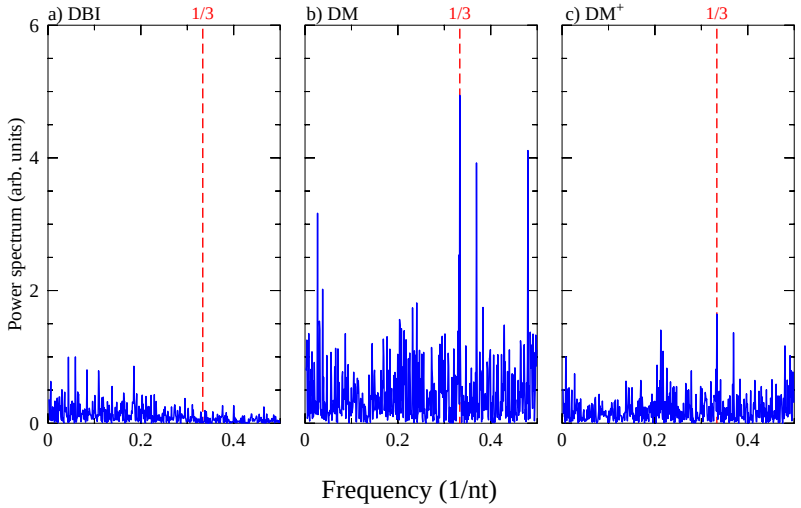

Figura 29
Espectro completo para a maximização da frequência f=1/3 da *Drosophila melanogaster*.

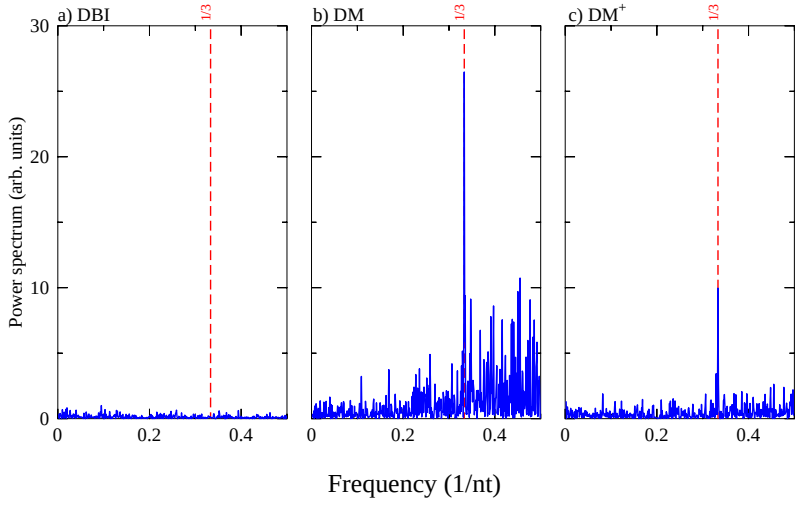

Figura 30
Espectro completo para a maximização frequência f=1/3 do *Plasmodium falciparum*.

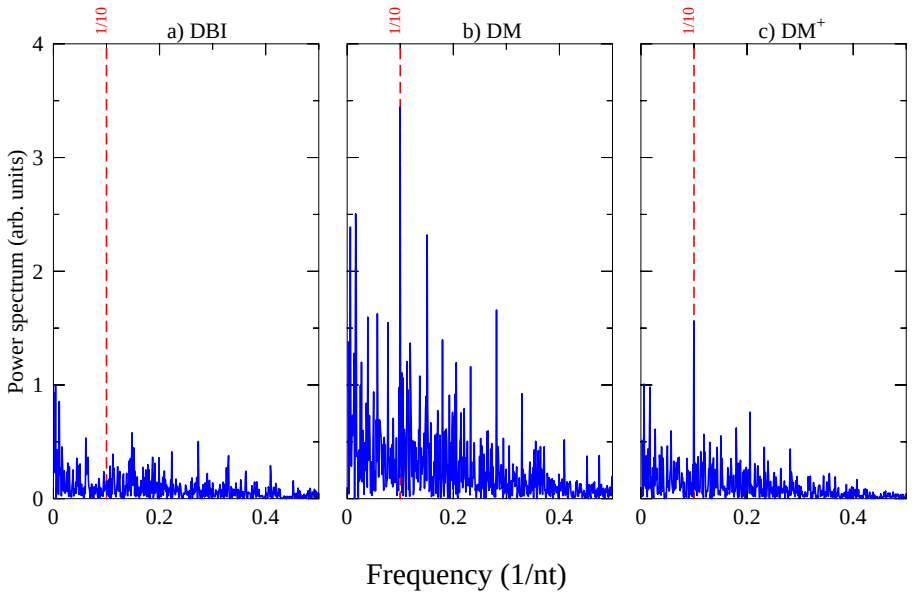

Figura 31
Espectro completo para a maximização da frequência f=1/10 da *Drosophila melanogaster*.

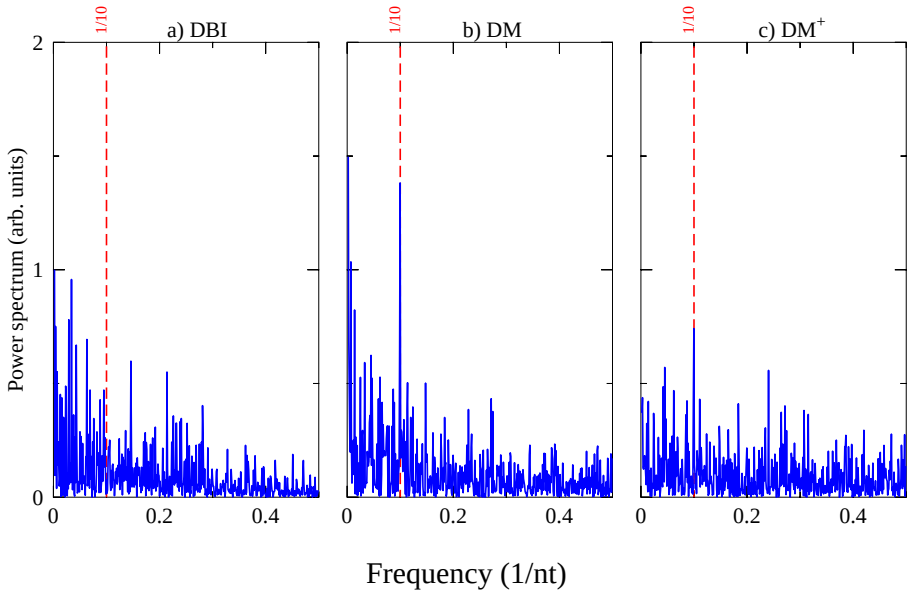

Figura 32
Espectro completo para a maximização frequência f=1/10 do *Plasmodium falciparum*.

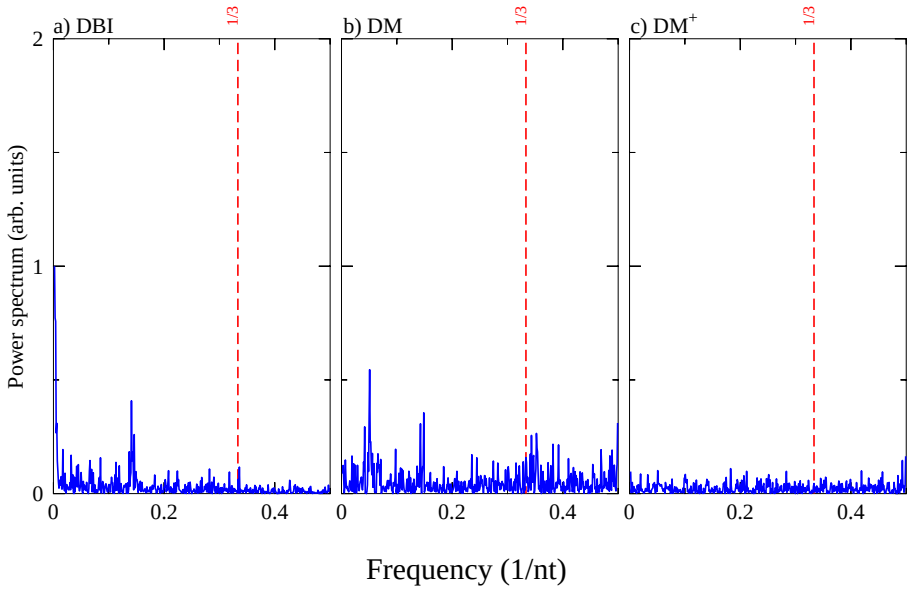

Figura 33
Espectro completo para a frequência f=1/3 onde a periodicidade é inexistente em *Plasmodium falciparum*.

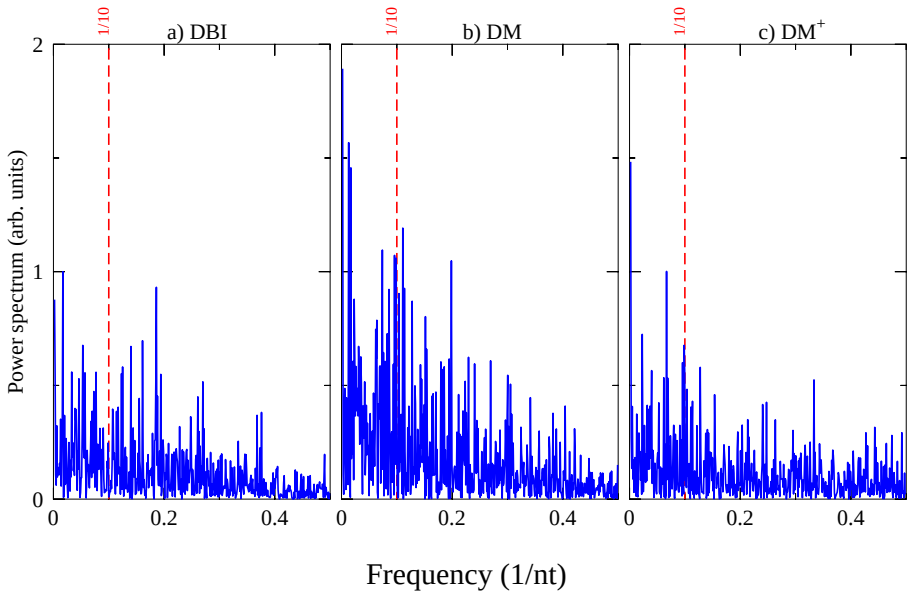

Figura 34
Espectro completo para a frequência f=1/10 onde a periodicidade é inexistente em *Drosophila melanogaster*

Referências

[1] Stewart, M. and McLachlan, A. D. Fourteen actin-binding sites on tropomyosin? *Nature* **257**, 331–333 (1975).

[2] McLachlan, A. D. Multichannel Fourier analysis of patterns in protein sequences. *J. Phys. Chem.* **97**(12), 3000–3006 (1993).

[3] Nunes, M., Wanner, E., and Weber, G. Origin of multiple periodicities in the Fourier power spectra of the *Plasmodium falciparum* genome. *BMC Genomics* **12**(Suppl 4), S4 (2011).

[4] Watson, J. D. and Crick, F. H. C. Molecular structure of nucleic acids: A structure for deoxyribose nucleic acid. *Nature* **171**, 737–738 (1953).

[5] Anastassiou, D. Genomic signal processing. *IEEE Signal Processing Mag.* **July**, 8–20 (2001).

[6] Bernstein, B., et al. An integrated encyclopedia of DNA elements in the human genome. *Nature* **489**(7414), 57 (2012).

[7] Pennisi, E. ENCODE project writes eulogy for junk DNA. *Science* **337**, 1159–1161 (2012).

[8] Mattick, J. Challenging the dogma: the hidden layer of non-protein-coding RNAs in complex organisms. *Bioessays* **25**(10), 930–939 (2003).

[9] Kiriakidou, M., et al. A combined computational-experimental approach predicts human microRNA targets. *Genes Dev.* **18**(10), 1165–1178 (2004).

[10] Jacobsen, A., Krogh, A., Kauppinen, S., and Lindow, M. miRMaid: a unified programming interface for microRNA data resources. *BMC bioinformatics* **11**(1), 29 (2010).

[11] Najafi-Shoushtari, S. H., et al. MicroRNA-33 and the SREBP host genes cooperate to control cholesterol homeostasis. *Science* **328**, 1566–1569 (2010).

[12] Rayner, K. J., et al. miR-33 contributes to the regulation of cholesterol homeostasis. *Science* **328**, 1570–1573 (2010).

[13] Fagundes-Lima, D. and Weber, G. CG-content log-ratio distributions of *Caenorhabditis elegans* and *Drosophila melanogaster* mirtrons. *arXiv preprint arXiv:1301.6099* (2013).

[14] Sharma, D., Issac, B., Raghava, G. P. S., and Ramaswamy, R. Spectral repeat finder (SRF): identification of repetitive sequences using Fourier transformation. *Bioinformatics* **20**(9), 1405–1412 (2004).

[15] Gupta, R., Sarthi, D., Mittal, A., and Singh, K. A novel signal processing measure to identify exact and inexact tandem repeat patterns in DNA sequences. *EURASIP Journal on Bioinformatics and Systems Biology* **2007**, 3–3 (2007).

[16] Shepherd, J. C. Periodic correlations in DNA sequences and evidence suggesting their evolutionary origin in a comma-less genetic code. *Journal of Molecular Evolution* **17**(2), 94–102 (1981).

[17] Wang, Z., Chen, Y., Li, Y., et al. A brief review of computational gene prediction methods. *Genomics Proteomics Bioinformatics* **2**(4), 216–221 (2004).

[18] Yamagishi, M. E. B. and Shimabukuro, A. I. Nucleotide frequencies in human genome and Fibonacci numbers. *Bulletin of Mathematical Biology* **70**(3), 643–653 (2008).

[19] Anastassiou, D. Frequency-domain analysis of biomolecular sequences. *Bioinformatics* **16**(12), 1073 (2000).

[20] Yan, M., Lin, Z. S., and Zhang, C. T. A new Fourier transform approach for protein coding measure based on the format of the Z curve. *Bioinformatics* **14**(8), 685–690 (1998).

[21] Zhang, C.-T. and Wang, J. Recognition of protein coding genes in the yeast genome at better than 95% accuracy based on the Z curve. *Nucleic acids research* **28**(14), 2804–2814 (2000).

[22] Fukushima, A., et al. Periodicity in prokaryotic and eukaryotic genomes identified by power spectrum analysis. *Gene* **300**, 203–211 (2002).

[23] Issac, B., Singh, H., Kaur, H., and Raghava, G. Locating probable genes using Fourier transform approach. *Bioinformatics* **18**(1), 196–197 (2002).

[24] Shepherd, A. J., Gorse, D., and Thornton, J. M. A novel approach to the recognition of protein architecture from sequence using Fourier analysis and neural networks. *Proteins: Structure, Function, and Bioinformatics* **50**(2), 290–302 (2003).

[25] Gao, F. and Zhang, C.-T. Comparison of various algorithms for recognizing short coding sequences of human genes. *Bioinformatics* **20**(5), 673–681 (2004).

[26] Yin, C. and Yau, S. S.-T. Prediction of protein coding regions by the 3-base periodicity analysis of a DNA sequence. *Journal of theoretical biology* **247**(4), 687–694 (2007).

[27] Akhtar, M., Epps, J., and Ambikairajah, E. Signal processing in sequence analysis: advances in eukaryotic gene prediction. *IEEE Journal on Selected Topics in Signal Processing* **2**(3), 310–321 (2008).

[28] Schweikert, G., et al. mGene.web: a web service for accurate computational gene finding. *Nucleic acids research* **37**(suppl 2), W312–W316 (2009).

[29] Hyatt, D., et al. Prodigal: prokaryotic gene recognition and translation initiation site identification. *BMC Bioinformatics* **11**(1), 119 (2010).

[30] Zhang, L., Tian, F., and Wang, S. A modified statistically optimal null filter method for recognizing protein-coding regions. *Genomics, Proteomics & Bioinformatics* (2012).

[31] Chen, B. and Ji, P. Numericalization of the self adaptive spectral rotation method for coding region prediction. *Journal of Theoretical Biology* **296**, 95–102 (2012).

[32] Wijaya, E., Frith, M. C., Horton, P., and Asai, K. Finding protein-coding genes through human polymorphisms. *PloS one* **8**(1), e54210 (2013).

[33] Fang, Y. and Li, J. Genomic law guided gene prediction in fungi and metazoans. *International Journal of Computational Biology and Drug Design* **6**(1), 157–169 (2013).

[34] Tomita, M., Wada, M., and Kawashima, Y. ApA dinucleotide periodicity in prokaryote, eukaryote, and organelle genomes. *Journal of molecular evolution* **49**(2), 182–192 (1999).

[35] Worning, P., Jensen, L. J., Nelson, K. E., Brunak, S., and Ussery, D. W. Structural analysis of DNA sequence: evidence for lateral gene transfer in *Thermotoga maritima*. *Nucleic Acids Research* **28**(3), 706–709 (2000).

[36] Tolstorukov, M., Kharchenko, P., Goldman, J., Kingston, R., and Park, P. Comparative analysis of H2A.Z nucleosome organization in the human and yeast genomes. *Genome research* **19**(6), 967–977 (2009).

[37] Hebert, C. and Crollius, H. Nucleosome rotational setting is associated with transcriptional regulation in promoters of tissue-specific human genes. *Genome Biology* **11**(5), R51 (2010).

[38] Tanaka, Y., Yamashita, R., Suzuki, Y., and Nakai, K. Effects of Alu elements on global nucleosome positioning in the human genome. *BMC Genomics* **11**(1), 309 (2010).

[39] Segal, E., et al. A genomic code for nucleosome positioning. *Nature* **442**(7104), 772–778 (2006).

[40] Duyao, M., et al. Trinucleotide repeat length instability and age of onset in Huntington's disease. *Nature genetics* **4**(4), 387–392 (1993).

[41] Warby, S. C., et al. CAG expansion in the Huntington disease gene is associated with a specific and targetable predisposing haplogroup. *The American Journal of Human Genetics* **84**(3), 351–366 (2009).

[42] Nishiyama, R., Qi, L., Lacey, M., and Ehrlich, M. Both hypomethylation and hypermethylation in a 0.2-kb region of a DNA repeat in cancer. *Molecular cancer research* **3**(11), 617–626 (2005).

[43] Sutherland, G. R. and Richards, R. I. Simple tandem DNA repeats and human genetic disease. *Proceedings of the National Academy of Sciences* **92**(9), 3636–3641 (1995).

[44] Sinden, R. R., et al. Triplet repeat DNA structures and human genetic disease: dynamic mutations from dynamic DNA. *Journal of biosciences* **27**(1), 53–65 (2002).

[45] Benson, G. Tandem repeats finder: a program to analyze DNA sequences. *Nucl. Acids. Res.* **27**(2), 573–580 (1999).

[46] `http://tandem.bu.edu/trf/trf.download.html`.

[47] Zhang, W.-F. and Yan, H. Exon prediction using empirical mode decomposition and Fourier transform of structural profiles of DNA sequences. *Pattern Recognition* **45**(3), 947–955 (2012).

[48] Senter, E., Sheikh, S., Dotu, I., Ponty, Y., and Clote, P. Using the fast Fourier transform to accelerate the computational search for RNA conformational switches. *PloS one* **7**(12), e50506 (2012).

[49] Epps, J., Ambikairajah, E., and Akhtar, M. An integer period DFT for biological sequence processing. In *Genomic Signal Processing and Statistics, 2008. GENSiPS 2008. IEEE International Workshop on*, 1–4. IEEE, (2008).

[50] Liew, A. W.-C., Yan, H., and Yang, M. Pattern recognition techniques for the emerging field of bioinformatics: A review. *Pattern Recogn.* **38**(11), 2055–2073 (2005).

[51] Berryman, M. J., Allison, A., Wilkinson, C. R., and Abbott, D. Review of signal processing in genetics. *Fluctuation and Noise Letters* **5**(4), R13–R35 (2005).

[52] Arneodo, A., Bacry, E., Graves, P., and Muzy, J. Characterizing long-range correlations in DNA sequences from Wavelet analysis. *Physical Review Letters* **74**(16), 3293–3296 (1995).

[53] Arneodo, A., et al. Wavelet based fractal analysis of DNA sequences. *Physica D: Nonlinear Phenomena* **96**(1), 291–320 (1996).

[54] Audit, B., et al. Long-range correlations in genomic DNA: a signature of the nucleosomal structure. *Physical Review Letters* **86**(11), 2471–2474 (2001).

[55] Audit, B., Vaillant, C., Arneodo, A., d'Aubenton Carafa, Y., and Thermes, C. Long-range correlations between DNA bending sites: relation to the structure and dynamics of nucleosomes. *Journal of Molecular Biology* **316**(4), 903–918 (2002).

[56] Su, J. and Bao, J. A Wavelet transform based protein sequence similarity model. *Appl. Math* **7**(3), 1103–1110 (2013).

[57] Epps, J. A hybrid technique for the periodicity characterization of genomic sequence data. *EURASIP Journal on Bioinformatics and Systems Biology* **2009** (2009).

[58] D'Avenio, G., Grigioni, M., Orefici, G., and Creti, R. SWIFT (sequence-wide investigation with Fourier transform): a software tool for identifying proteins of a given class from the unannotated genome sequence. *Bioinformatics* **21**(13), 2943–2949 (2005).

[59] Chodavarapu, R. K., et al. Relationship between nucleosome positioning and DNA methylation. *Nature* **466**(7304), 388–392 (2010).

[60] Xu, F., Colasanti, A. V., Li, Y., and Olson, W. K. Long-range effects of histone point mutations on DNA remodeling revealed from computational analyses of sin-mutant nucleosome structures. *Nucleic acids research* **38**(20), 6872–6882 (2010).

[61] Zentner, G. E. and Henikoff, S. Regulation of nucleosome dynamics by histone modifications. *Nature Structural & Molecular Biology* **20**(3), 259–266 (2013).

[62] Bai, L. and Morozov, A. Gene regulation by nucleosome positioning. *Trends in Genetics* **26**(11), 476–483 (2010).

[63] Struhl, K. and Segal, E. Determinants of nucleosome positioning. *Nature Structural & Molecular Biology* **20**(3), 267–273 (2013).

[64] Reynolds, S., Bilmes, J., and Noble, W. On the relationship between DNA periodicity and local chromatin structure. In *Research in Computational Molecular Biology*, 434–450. Springer, (2009).

[65] Collings, C. K., Fernandez, A. G., Pitschka, C. G., Hawkins, T. B., and Anderson, J. N. Oligonucleotide sequence motifs as nucleosome positioning signals. *PloS one* **5**(6), e10933 (2010).

[66] Haupt, R., Haupt, S., and Wiley, J. *Practical genetic algorithms*. Wiley Online Library, (2004).

[67] Cropper, W. and Anderson, P. Population dynamics of a tropical palm: use of a genetic algorithm for inverse parameter estimation. *Ecological modelling* **177**(1), 119–127 (2004).

[68] Cropper, W. and Comerford, N. Optimizing simulated fertilizer additions using a genetic algorithm with a nutrient uptake model. *Ecological modelling* **185**(2), 271–281 (2005).

[69] Hoffmann, J., Tenorio, M., Wille, E., and Godoy, W. Applying genetic algorithms to the information sets search problem. In *Communications (LATINCOM), 2011 IEEE Latin-American Conference on*, 1–5. IEEE, (2011).

[70] Keane, A. Genetic algorithm optimization of multi-peak problems: studies in convergence and robustness. *Artificial Intelligence in Engineering* **9**(2), 75–83 (1995).

[71] Tsai, T., Yang, C., and Peng, Y. Genetic algorithms for the investment of the mutual fund with global trend indicator. *Expert Systems with Applications* **38**(3), 1697–1701 (2011).

[72] Molla-Alizadeh-Zavardehi, S., Hajiaghaei-Keshteli, M., and Tavakkoli-Moghaddam, R. Solving a capacitated fixed-charge transportation problem by artificial immune and genetic algorithms with a Prüfer number representation. *Expert Systems with Applications: An International Journal* **38**(8), 10462–10474 (2011).

[73] Mitchell, M. and Taylor, C. Evolutionary computation: an overview. *Annual Review of Ecology and Systematics* **30**, 593–616 (1999).

[74] Forrest, S. Genetic algorithms: Principles of natural selection applied to computation. *Science* **261**(5123), 872–878 (1993).

[75] Koszul, R., Meselson, M., Van Doninck, K., Vandenhaute, J., and Zickler, D. The centenary of Janssens's chiasmatype theory. *Genetics* **191**(2), 309–317 (2012).

[76] Creighton, H. B. and McClintock, B. A correlation of cytological and genetical crossing-over in *Zea mays*. *Proceedings of the National Academy of Sciences of the United States of America* **17**(8), 492 (1931).

[77] `http://www.ncbi.nlm.nih.gov/`.

[78] Schmid, C., Perier, R., Praz, V., and Bucher, P. EPD in its twentieth year: towards complete promoter coverage of selected model organisms. *Nucleic Acids Research* **34**(suppl 1), D82–D85 (2006).

[79] Coward, E. Equivalence of two Fourier methods for biological sequences. *J. Math. Biol.* **36**, 64–70 (1997).

[80] Barman, S., Saha, S., Mandal, A., and Roy, M. Prediction of protein coding regions of a DNA sequence through spectral analysis. In *Informatics, Electronics & Vision (ICIEV), 2012 International Conference on*, 12–16. IEEE, (2012).

[81] Pique-Regi, R., Ortega, A., Tewfik, A., and Asgharzadeh, S. Detecting changes in DNA copy number: Reviewing signal processing techniques. *Signal Processing Magazine, IEEE* **29**(1), 98–107 (2012).

[82] Audit, B., et al. Multiscale analysis of genome-wide replication timing profiles using a Wavelet-based signal-processing algorithm. *Nature protocols* **8**(1), 98–110 (2012).

[83] Frigo, M. and Johnson, S. G. FFTW: An adaptive software architecture for the FFT. In *Proc. 1998 IEEE Intl. Conf. Acoustics Speech and Signal Processing*, volume 3, 1381–1384. IEEE, (1998).

[84] Gao, J., Qi, Y., Cao, Y., and Tung, W. Protein coding sequence identification by simultaneously characterizing the periodic and random features of DNA sequences. *Journal of Biomedicine and Biotechnology* **2**, 139 (2005).

[85] Brodzik, A. Quaternionic periodicity transform: an algebraic solution to the tandem repeat detection problem. *Bioinformatics* **23**(6), 694 (2007).

[86] Widom, J. et al. Short-range order in two eukaryotic genomes: relation to chromosome structure. *Journal of molecular biology* **259**(4), 579–588 (1996).

[87] Ocak, H. A medical decision support system based on support vector machines and the genetic algorithm for the evaluation of fetal well-being. *Journal of medical systems* **37**(2), 1–9 (2013).

[88] Sharma, M. and Mukharjee, S. Brain tumor segmentation using genetic algorithm and artificial neural network fuzzy inference system (ANFIS). In *Advances in Computing and Information Technology*, 329–339. Springer, (2013).

[89] Osman, N., Ng, A., and McManus, K. Selection of important input parameters using neural network trained with genetic algorithm for damage to light structures. In *Proceedings of the fifth International Conference on Engineering Computational Technology: Las Palmas de Gran Canaria, Spain 12-15 September, 2006*, 123–124. Civil-Comp, (2012).

[90] Zhao, J.-q., Wang, L., Zeng, P., and Fan, W.-h. An effective hybrid genetic algorithm with flexible allowance technique for constrained engineering design optimization. *Expert Systems with Applications* **39**(5), 6041–6051 (2012).

[91] Cropper, W. P., Holm, J. A., and Miller, C. J. An inverse analysis of a matrix population model using a genetic algorithm. *Ecological Informatics* **7**(1), 41–45 (2012).

[92] Volk, M., Lautenbach, S., Strauch, M., and Whittaker, G. Quantifying tradeoffs between water availability, water quality, food production and bioenergy production in a Central German Catchment. In *EGU General Assembly Conference Abstracts*, volume 14, 3364, (2012).

[93] Said, Y. *On Genetic Algorithms and their Applications - Handbook of Statistics*. Elsevier, (2005).

[94] Oh, J., Nam, H., Choi, J., and Lee, S. Prediction of atomic arrangement of Pt-Cu nanoalloy by genetic algorithm. In *Journal of Physics: Conference Series*, volume 410, 012084. IOP Publishing, (2013).

[95] Singh, D. K., Srinivas, K., and Das, D. B. A dynamic channel assignment in GSM telecommunication network using modified genetic algorithm. In *Proceedings of the 6th Euro American Conference on Telematics and Information Systems*, 425–429. ACM, (2012).

[96] Qumsieh, A. AI::Genetic v. 0.05: A pure Perl genetic algorithm implementation, (2003-2005).

[97] Tiwari, S., Ramachandran, S., Bhattacharya, A., Bhattacharya, S., and Ramaswamy, R. Prediction of probable genes by Fourier analysis of genomic sequences. *Comput. Appl. Biosci.* **13**(3), 263–270 (1997).

[98] Shao, J., Yan, X., and Shao, S. SNR of DNA sequences mapped by general affine transformations of the indicator sequences. *Journal of Mathematical Biology* , 1–19 (2012).

[99] Zhong, J., Hu, X., Gu, M., and Zhang, J. Comparison of performance between different selection strategies on simple genetic algorithms. In *Computational Intelligence for Modelling, Control and Automation, 2005 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, International Conference on*, volume 2, 1115–1121. IEEE, (2005).

[100] Razali, N. M. and Geraghty, J. Genetic algorithm performance with different selection strategies in solving TSP. In *Proceedings of the World Congress on Engineering*, volume 2, (2011).

[101] Miller, B. L. and Goldberg, D. E. Genetic algorithms, tournament selection, and the effects of noise. *Urbana* **51**, 61801 (1995).

[102] Goldberg, D. E. and Deb, K. A comparative analysis of selection schemes used in genetic algorithms. *Urbana* **51**, 61801–2996 (1991).

[103] Blickle, T. and Thiele, L. A comparison of selection schemes used in evolutionary algorithms. *Evolutionary Computation* **4**(4), 361–394 (1996).

[104] Falkenauer, E. The worth of the uniform. In *Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on*, volume 1. IEEE, (1999).

[105] Srinivas, M. and Patnaik, L. M. Adaptive probabilities of crossover and mutation in genetic algorithms. *Systems, Man and Cybernetics, IEEE Transactions on* **24**(4), 656–667 (1994).

[106] Grefenstette, J. J. Optimization of control parameters for genetic algorithms. *Systems, Man and Cybernetics, IEEE Transactions on* **16**(1), 122–128 (1986).

[107] Gardner, M. J., et al. Genome sequence of the human malaria parasite *Plasmodium falciparum*. *Nature* **419**(6906), 498–511 (2002).