SEQUÊNCIAMENTO GENÉTICO E CLONAGEM: MODELAGEM E APLICAÇÃO DA TEORIA DOS GRAFOS

Alex Júnio Maia de Oliveira, Artur Vidal Krause,
Lucas Coelho Silva, Lucas Dressler Sodré

Novembro 2024

Resumo

Conforme os estudos acerca da genética se desenvolvem e aumentam em complexidade, torna-se cada vez mais necessário o uso da computação para otimizar processos e a utilização de mecanismos que facilitem a visualização de determinadas informações, principalmente em se tratando de sequências genéticas, onde a quantidade de dados é exorbitante e, muitas vezes, é inviável analisar tais dados manualmente. O presente trabalho tem por objetivo explorar alguns do principais algoritmos no campo da bioinformática para o alinhamento de sequências genéticas, bem como visa explanar as características, mecanismos de funcionamento e as aplicações de cada um dos algoritmos abordados. Através de uma intensa pesquisa, fundamentada por meio de livros, artigos acadêmicos e vídeos sobre o assunto, foi identificada a importância do uso de tais ferramentas no estudo das sequências genéticas e a maneira pela qual o uso desses algoritmos impacta as diversas áreas da biologia, da medicina e, em especial, da genética.

Palavras-chave: Teoria dos grafos, Sequenciamento genético, Clonagem, Algoritmos, *Python*, *Numpy*, *Networkx*.

Sumário

1 Introdução

No campo da genética, observa-se uma ampla gama de aplicações no cotidiano humano. Nesse contexto, o estudo dos alinhamentos de sequências genéticas torna-se particularmente relevante, pois permite compreender diferenças e semelhanças entre essas sequências. Tal compreensão tem impacto direto na biologia molecular, no entendimento da evolução das espécies, na identificação de mutações, na montagem de genomas e em diversas outras aplicações.

Dado o tamanho das cadeias de bases nitrogenadas e os grandes conjuntos de proteínas ou nucleotídeos, é fundamental desenvolver métodos mais eficientes e rápidos para computar e analisar alinhamentos. Por exemplo, para duas sequências de tamanho 11, existem aproximadamente 706 mil alinhamentos possíveis, o que ilustra a complexidade envolvida[1].

Nesse cenário, a possível modelagem de problemas relacionados ao alinhamento genético com o uso de grafos surge como uma abordagem promissora. Algoritmos como Needleman-Wunsch, Smith-Waterman e Bellman-Ford, entre outros, oferecem ferramentas que facilitam tanto a compreensão quanto a resolução dessas problemáticas.

Assim, o alinhamento de sequências destaca-se como um campo essencial de estudo moderno, proporcionando assimilações que impactam diretamente a ciência e a sociedade. A utilização de grafos e algoritmos especializados não só otimiza a análise de grandes volumes de dados, mas também viabiliza soluções mais robustas para questões complexas, como a identificação de mutações. Com o avanço contínuo da tecnologia, espera-se que cada vez mais novas abordagens ampliem as possibilidades de tal segmento, promovendo descobertas e aplicações práticas que beneficiem o cotidiano humano.

2 Conceitos iniciais

Antes de iniciar um estudo aprofundado sobre as aplicações da teoria dos grafos no sequênciamento genético e na clonagem, além de descorrer sobre os algoritmos que tangem o assunto, deve-se ter em mente alguns dos conceitos mais fundamentais tanto da parte genética, quanto da parte de grafos.

2.1 Definições sobre Genética

As definições aqui apresentadas foram parafraseadas do livro *"Branching Processes in Biology"* dos autores *Marek Kimmel e David E. Axelrod.*

Definição 1. *NUCLEOTÍDEO Uma molécula composta por três componentes principais: uma base nitrogenada (adenina, guanina, citosina, timina ou uracila), um açúcar de cinco carbonos (ribose ou desoxirribose) e um ou mais grupos fosfato. Os nucleotídeos são as unidades estruturais básicas dos ácidos nucleicos (DNA e RNA) e participam de processos de armazenamento e transmissão de informação genética.*

Definição 2. *AMINOÁCIDO Uma molécula orgânica composta por um grupo amino (-NH_2), um grupo carboxila (-COOH) e uma cadeia lateral variável (R)*

[1] Disponível em: `https://www.bioinfoclass.com/blog/alinhamento`

ligada a um átomo de carbono central. Os aminoácidos são os blocos de construção das proteínas e desempenham papéis essenciais em processos metabólicos e na estrutura celular.

Definição 3. *DNA Ácido desoxirribonuleico; o material genético. Uma longa dupla hélice com uma estrutura semelhante a uma escada torcida. A espinha dorsal da escada são fios compostos por açúcares alternados (desoxirribose) e grupos fosfato. Os degraus da escada são pares de subunidades de nucleotídeos. As subunidades de nucleotídeos são A (adenina), T (timina), G (guanina) e C (citosina). A é pareado com T e G é pareado com C.*

Definição 4. *RNA Ácido ribonucleico. Uma molécula semelhante ao DNA, mas com um açúcar diferente (ribose em vez de desoxirribose), um nucleotídeo diferente (U em vez de T) e principalmente de fita simples (em vez de fita dupla). Existem vários tipos de RNA. Um deles, o RNA mensageiro (mRNA), é transcrito como um componente cópia tária da sequência de nucleotídeos no DNA e funções para determinar o sequência de aminoácidos na proteína.*

Definição 5. *PROTEÍNA Uma molécula de polímero que consiste em subunidades de monômeros de aminoácidos. A sequência linear dos aminoácidos na proteína é determinada pela sequência correspondente sequência de nucleotídeos no DNA (gene). Algumas proteínas (enzimas) funcionam para incentivar reações químicas; Outras proteínas têm uma função estrutural.*

Definição 6. *GENE Uma sequência de bases no DNA que codifica uma proteína e influencia o características herdadas de uma célula ou organismo.*

Definição 7. *GENOMA Todo o DNA de um organismo, incluindo o DNA que codifica proteínas e o DNA que não codifica proteínas.*

Definição 8. *MUTAÇÃO Uma mudança na sequência de DNA. Geralmente detectada por uma súbita e inerente alteração de uma característica observada de uma célula ou de um organismo. No entanto, uma mutação pode ser detectada diretamente determinando uma mudança na sequência de DNA, mesmo que não haja alteração característica visível na célula ou organismo.*

Definição 9. *ALINHAMENTO DE SEQUÊNCIA O processo de disposição ordenada de sequências de DNA, RNA ou proteínas para identificar regiões de similaridade que possam indicar relações funcionais, estruturais ou evolutivas. O alinhamento pode ser realizado entre duas ou mais sequências e é fundamental em bioinformática para análises comparativas e predições biológicas.*

2.2 Definições sobre Grafos

As definições aqui apresentadas foram parafraseadas das referências *"Handbook of Graph Theory"* dos autores *Jonathan L. Gross, Jay Yellen e Ping Zhang*, *"Discrete Mathematics"* do autor *R. Johnsonbaugh*, *Teoria do Grafos* da *UNESP*, *Grafos: Conceitos, Algoritmos e Aplicações* dos autores *M. Goldbarg e E. Goldbarg*, *Teoria Computacional de Grafos: Os Algoritmos* do autor *J. L. Szwarcfiter*.

Definição 10. *GRAFO Um grafo $G = (V, E)$ consiste em dois conjuntos V e E. Os elementos de V são chamados de vértices (ou nós). Os elementos de E*

são chamados de arestas. Cada aresta tem um conjunto de um ou dois vértices associados a ela, que são chamados de seus pontos finais. Diz-se que uma aresta conecta seus pontos finais.

Definição 11. *GRAU O grau de um vértice em um grafo é o número de arestas que incidem nesse vértice. Em um grafo não direcionado, o grau é a contagem total de arestas conectadas ao vértice. Em um grafo direcionado, distingue-se entre grau de entrada (número de arestas que chegam ao vértice) e grau de saída (número de arestas que saem do vértice).*

Definição 12. *LAÇO Uma aresta em um grafo que conecta um vértice a si mesmo, ou seja, tem o mesmo vértice como extremidade inicial e final. Em termos formais, uma aresta $e = (v, v)$ é um laço se ambas as extremidades são o mesmo vértice v.*

Definição 13. *ARESTAS PARALELAS Duas ou mais arestas em um grafo que conectam o mesmo par de vértices. Em um grafo não direcionado, arestas paralelas possuem as mesmas extremidades; em um grafo direcionado, possuem os mesmos vértices inicial e final.*

Definição 14. *GRAFO DIRIGIDO Um grafo dirigido (ou digrafo) é composto por um conjunto de vértices conectados por arestas direcionadas, chamadas de arcos. Cada arco possui uma orientação específica, indo de um vértice inicial (cauda) para um vértice terminal (cabeça). Essa estrutura é usada para representar relações assimétricas entre os elementos.*

Definição 15. *GRAFO COM PESOS Um grafo com pesos é um grafo no qual cada aresta está associada a um valor numérico, chamado de peso ou custo. Formalmente, um grafo com pesos é representado como $G = (V, E, w)$, onde V é o conjunto de vértices, E é o conjunto de arestas, e $w : E \to \mathbb{R}$ é uma função que atribui um peso a cada aresta. Grafos com pesos são amplamente usados em problemas de otimização, como caminhos mínimos e árvores geradoras mínimas.*

Definição 16. *CAMINHOS Um caminho em um grafo é um trajeto (sequência de vértices e arestas) tal que nenhum vértice interno é repetido.*

Definição 17. *CICLO Um ciclo em um grafo é uma sequência fechada de vértices $v_1, v_2, \ldots, v_k, v_1$, onde $k \geq 3$, tal que cada par consecutivo de vértices (v_i, v_{i+1}) (para $i = 1, \ldots, k-1$) e (v_k, v_1) são arestas no grafo, e todos os vértices (exceto v_1 no início e no fim) são distintos. Em grafos direcionados, a direção das arestas também deve ser respeitada ao formar o ciclo.*

Definição 18. *CONECTIVIDADE Um grafo é conexo se, entre cada par de vértices, existe um caminho.*

Definição 19. *REDE Uma rede é um grafo direcionado com uma fonte (source) e um sumidouro (sink), onde cada aresta possui uma capacidade que limita o fluxo. Um problema clássico seria é determinar o fluxo máximo entre a fonte e o sumidouro sem exceder as capacidades, por exemplo.*

Definição 20. *FLUXO Em um grafo direcionado com pesos nas arestas (representando capacidades), o fluxo é uma função $f : E \to \mathbb{R}$, onde E é o conjunto de arestas, que satisfaz as seguintes condições:*

- ***Restrição de Capacidade**: Para cada aresta $(u, v) \in E$, o fluxo $f(u, v)$ deve estar entre 0 e a capacidade da aresta $c(u, v)$, ou seja, $0 \leq f(u, v) \leq c(u, v)$.*

- ***Conservação de Fluxo**: Para cada vértice v, exceto a fonte s e o destino t, o fluxo total que entra em v deve ser igual ao fluxo total que sai de v:*

$$\sum_{u \in V} f(u, v) = \sum_{w \in V} f(v, w).$$

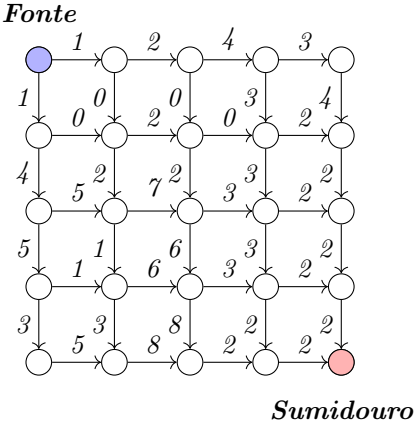
O fluxo máximo de um grafo é o maior fluxo possível da fonte s ao destino t que respeita as restrições de capacidade e conservação.

3 Algoritmos e parte computacional

Agora, vamos compreender os principais algoritmos utilizados no alinhamento de sequências e os implementar computacionalmente usando a linguagem *Python*. Além disso, tome as seguintes definições que serão necessárias para a total compreensão dos algoritmos:

Definição 21. *DISTÂNCIA DE LEVENSHTEIN Em teoria da informação, a distância Levenshtein é dada pelo número mínimo de operações necessárias para transformar um string em outra. Tais operações são inserção, deleção ou substituição de um carácter.*

Definição 22. *GRID Um grid é uma rede que que tem o seguinte formato:*



Definição 23. *MATCH, MISMATCH e GAP Um match, no contexto de alinhamento de sequências, ocorre quando duas bases nitrogendas (A, T, C ou G) são iguais no enfileiramento. Já um mismatch ocorre quando duas bases nitrogendas são distintas no alinhamento. Por fim, um gap ocorre quando, no alinhamento, há uma lacuna. Por exemplo, no enfileiramento ATC–GT e ACCT-GT ocorrem 4 matches, 1 mismatch e 2 gaps.*

Definição 24. *ALINHAMENTOS GLOBAL e LOCAL O alinhamento global é feito quando comparamos uma seqüência de aminoácidos ou nucleotídeos com outra, ao longo de toda sua extensão. Já o alinhamento local acontece quando*

a comparação entre duas seqüências não é feita ao longo de toda sua extensão, mas sim através de pequenas regiões[2].

Definição 25. *BACKTRACKING Backtracking é um algoritmo genérico que busca, por força bruta, soluções possíveis para problemas computacionais.*

3.1 Algoritmo de Neddleman-Wunsch

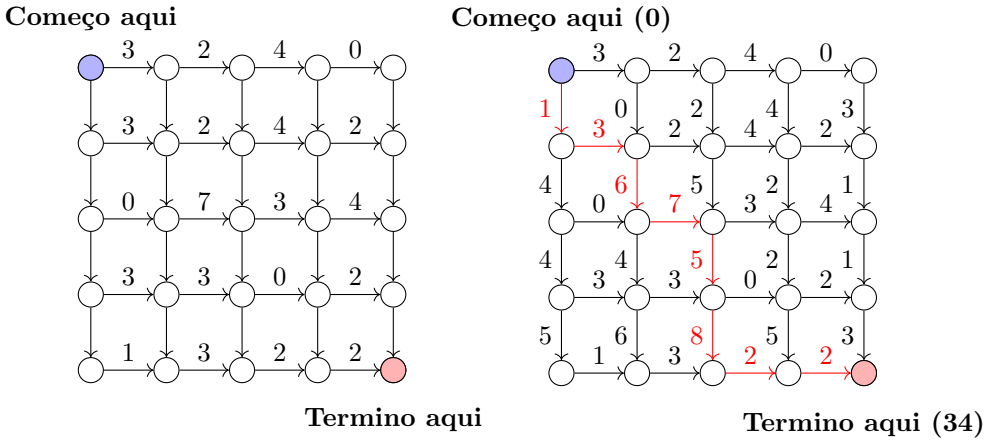
O algoritmo de Needleman-Wunsch, originalmente, não foi idealizado para ser utilizado no problema de alinhamento de sequências, contudo, com o surgimento do ramo da *Bioinformática*, o algoritmo foi sendo amplamente utilizado na modelagem do problema[3].

A entrada constitui de duas sequências (podendo ser de tamanhos diferentes) e o algoritmo retorna o alinhamento com o maior número de *matches*, no caso base, ou com a maior soma possível, no caso de pesos para um *match*, um *mismatch* e um *gap*. Assim, o retorno mostra a "configuração"mais parecida entre duas sequências (de maneira global), considerando os matches, os mismatches e os gaps[4].

3.1.1 Problema do Turista de Manhattan

Uma maneira de entender como o algoritmo funciona é estudando uma solução do Problema do Turista em Manhattan.

Imagine que um turista está na cidade de Manhanttan e decide visitar o maior número de atrações turísticas possíveis iniciando em um cruzamento de ruas e termine em outro. A ideia é transformar o problema em um problema de redes (ou seja, um grid) com os cruzamentos das ruas e calcular o caminho de peso máximo deste grafo (os pesos das arestas direcionadas seriam a quantidade de atrações entre um cruzamento e outro).



[2]Disponível em https://professor.pucgoias.edu.br/SiteDocente/admin/arquivosUpload/18497/material/Cap.203-20Alinhamento-20de-20sequncias.pdf

[3]Disponível em https://www.youtube.com/playlist?list=PLVSek57SJFtTdzGvBpmj5qvnqZ-3LL76u

[4]O algoritmo encontra-se no repositório https://github.com/alexoliveiraFGV24/seq-genetico.git

3.1.2 Modelagem do algoritmo

Para resolver o problema do alinhamento genético "mais parecido"podemos fazer um raciocínio semelhante.

Para efeitos de simplificação, considere os pesos (podem ser arbitrários, de modo que o peso para um match seja o maior) para *matches*, *mismatches* e *gaps* 1, 0 e 0, respectivamente (além disso, considere que *mismatches* e *gaps* sejam chamados de *indels*).

Para modelar o algoritmo, vamos fazer um grid parecido com o estruturado no problema do Turista de Manhattan, contudo, agora vamos adicionar arestas direcionadas na diagonal.

Considere que a primeira sequência tenha tamanho m e a segunda tenha tamanho n e que os valores dos *matches* e *indels* foram dados. Iremos criar uma matriz H, em que H(0,0) é a primeira entrada e H(m,n) a última.

Agora, vamos criar um grid em que os nós sejam cada entrada da matriz H e as arestas sejam sejam da forma (considere que e((i,j),(k,p)) é uma aresta direcionada da entrada H(i,j) para a entrada H(k,p) e s1(i) seja base da posição i da primeira sequência):

$$e = \begin{cases} e((i-1,j-1),(i,j)) \text{ com peso match se } s1(i-1) = s2(j-1), \\ e((i-1,j-1),(i,j)) \text{ com peso indel se } s1(i-1) \neq s2(j-1), \\ e((i-1,j),(i,j)) \text{ com peso indel}, \\ e((i,j-1),(i,j)) + \text{ com peso indel}, \\ i,j > 0 \end{cases}$$

Se percorremos a diagonal, significa um match, se percorremos a vertical, significa um gap na primeira sequência e se percorremos a horizontal, significa um gap na segunda sequência.

Dessa forma, para achar o melhor alinhamento possível, basta observar a configuração do caminho de tamanho máximo do grafo construído.

3.2 Algoritmo de Smith-Waterman

O algoritmo de Smith-Waterman foi elaborado para encontrar o melhor alinhamento local de duas sequências de nucleotídeos ou proteínas, sendo assim, ele otimiza o comprimento da similaridade comparando todos os possíveis comprimentos, determinando as regiões mais similares entre elas[5].

O algoritmo de Smith-Waterman assim como o Algoritmo de Needleman-Wunsch são algoritmos de programação dinâmica, ou seja, além de encontrar uma solução ótima(a melhor solução, com o score mais alto) em seu interior estão contidas soluções ótimas para todos os sub-comprimentos. A principal diferença deste algoritmo para o Needleman-Wunsch é que os valores negativos deste algoritmo são definidos como zero, o que torna a pontuação do algoritmo pontuada positivamente.

O processo de Backtraking revela o melhor alinhamento local (com maior score), ele começa do ponto da matriz com maior pontuação e retorna até um ponto com pontuação zero (menor pontuação neste algoritmo)[6].

[5]Disponível em: `https://en.wikipedia.org/wiki/Smith-Waterman_algorithm`

[6]O algoritmo encontra-se em `https://github.com/alexoliveiraFGV24/seq-genetico.git`

3.2.1 Modelagem do algoritmo para Grafo

Definiremos uma matriz H para comparar duas sequências da seguinte forma:

Cada sequencia da matriz $H(i, j)$ é inicializada com um "gap"no início, ou seja, a primeira linha e coluna são puladas pelo início da sequência e preenchidas com zeros, a partir daí o sistema de pontuação e definido da seguinte forma,

- score_match: **+2 para cada par de caracteres correspondentes**.
- score_mismatch: **-1 para cada par de caracteres não correspondentes**.
- gap_penalty: **-1 para inserções ou deleções**.

$$H(i,j) = \max \begin{cases} 0, \\ H(i-1, j-1) + (\text{score\_match ou score\_mismatch}), \\ H(i-1, j) + \text{gap\_penalty}, \\ H(i, j-1) + \text{gap\_penalty} \end{cases}$$

A matriz $H(i, j)$ é preenchida utilizando a fórmula acima para cada posição (i, j), comparando os caracteres das duas sequências:

- Diagonal ($H(i-1, j-1)$): Representa o alinhamento entre dois caracteres.
- Para cima ($H(i-1, j)$): Representa a inserção de um gap na sequência horizontal.
- Para a esquerda ($H(i, j-1)$): Representa a inserção de um gap na sequência vertical.

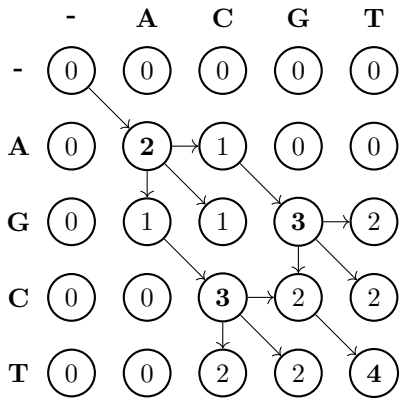
Contudo algoritmo de Smith-Waterman também pode ser interpretado como a construção de um **grafo direcionado ponderado**, onde:

- Cada célula da matriz $H(i, j)$ representa um **nó** do grafo.
- As transições entre as células da matriz ($H(i-1, j-1)$, $H(i-1, j)$, $H(i, j-1)$) representam as **arestas direcionadas**.
- Os pesos das arestas são definidos pelo mesmo sistema de pontuação citado acima.

Neste caso o **backtracking** para encontrar o alinhamento ótimo local corresponde a encontrar o **caminho com o maior peso** no grafo, partindo do nó com o maior valor (score máximo) até um nó com valor zero (score mínimo).

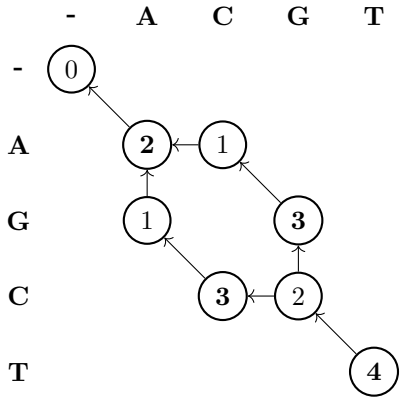
Exemplo para as sequências

seq1: **ACGT**, seq2: **AGCT**



Para encontrar a(s) sequência(s) com o melhor alinhamento local basta realizar o backtracking.

Para realizar o **backtracking** deve-se inverter a orientação das arestas do grafo, e registrar a volta (como descrito abaixo) para cada possível caminho no digrafo, ao iniciar pelo nó com o maior valor, neste caso $H(4,4) = 4$, e finaliza-lo em um nó com valor zero, neste caso $H(0,0) = 0$.



Os passos para registrar os alinhamentos são:

- **Aresta na diagonal** indica que foi realizado um **match** entre as duas sequências (o carácter e igual). Registre o carácter.
- **Aresta na horizontal** indica que não foi realizado um match. Registre o carácter da sequência **horizontal** e insira um gap na sequência **vertical**.
- **Aresta na vertical** indica que não foi realizado um match. Registre o carácter da sequência vertical e insira um gap na sequência **horizontal**.

Os alinhamentos correspondentes são os seguintes (um para cada possível caminho):

Seq1(local): **AGC-T**.

Seq2(local): **A-GCT**.

Portanto, o **alinhamento ótimo local** foi encontrado utilizando o maior valor de um nó do digrafo como ponto de partida e realizando o backtracking para

recuperar as passagens anteriores. Esse processo evidencia as regiões de maior similaridade entre as duas sequências dadas.

3.3 Algoritmo de Ford-Fulkerson

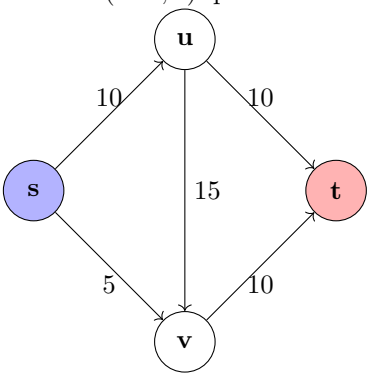
O algoritmo de Ford-Fulkerson é uma abordagem clássica para resolver problemas de fluxo máximo em redes de grafos direcionados. Ele foi desenvolvido para determinar a quantidade máxima de fluxo que pode ser transportada de um nó fonte (*source*) para um nó destino (*sink*) em um grafo, respeitando as capacidades das arestas.

A entrada do algoritmo consiste em um grafo direcionado onde cada aresta possui uma capacidade associada, e o objetivo é maximizar o fluxo entre dois vértices específicos, considerando as restrições impostas pelas capacidades. O método utiliza o conceito de **caminhos aumentantes** e o teorema do corte mínimo para garantir a otimalidade do fluxo máximo[7].

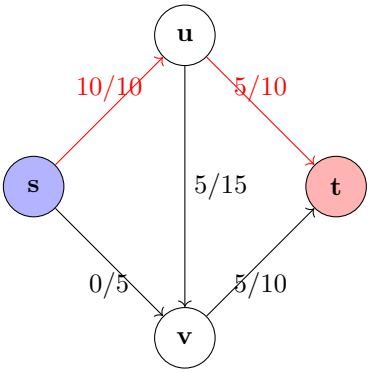
3.3.1 Modelo de Rede de Fluxo

Para entender a aplicação do algoritmo, é importante compreender o modelo de rede de fluxo. Um grafo direcionado $G = (V, E)$ é definido como: - V: o conjunto de vértices (nós). - E: o conjunto de arestas direcionadas. Cada aresta $(u, v) \in E$ possui uma capacidade $c(u, v) \geq 0$, indicando a quantidade máxima de fluxo que pode ser enviada de u para v.

Além disso, o grafo inclui: - Um nó fonte (*source*, s) que inicia o fluxo. - Um nó destino (*sink*, t) que recebe o fluxo.



Exemplo de rede de fluxo.



Estado final do fluxo.

3.3.2 Caminhos Aumentantes e Fluxo Residual

O algoritmo de Ford-Fulkerson utiliza a ideia de caminhos aumentantes em um grafo residual. O grafo residual representa as capacidades restantes das arestas, considerando o fluxo já alocado. Para encontrar o fluxo máximo, o algoritmo segue os seguintes passos: 1. Inicialize o fluxo $f(u, v) = 0$ para todas as arestas $(u, v) \in E$. 2. Enquanto houver um caminho p de s a t no grafo residual com capacidade positiva: - Encontre a capacidade mínima $c_{\min}$ ao longo de p. - Atualize o fluxo f ao longo do caminho p. 3. Quando não houver mais caminhos aumentantes, o fluxo f é máximo.

[7] O algoritmo encontra-se no repositório `https://github.com/alexoliveiraFGV24/seq-genetico.git`

3.3.3 Teorema do Corte Mínimo

Um resultado fundamental relacionado ao algoritmo é o **Teorema do Corte Mínimo**, que estabelece que o valor do fluxo máximo é igual à capacidade do corte mínimo que separa o nó fonte s do nó destino t. Esse teorema garante a otimalidade do algoritmo e sua aplicação prática em problemas reais, como redes de transporte, alocação de recursos e sistemas de distribuição.

Portanto, o algoritmo de Ford-Fulkerson oferece uma solução eficiente para problemas de fluxo máximo, com diversas aplicações práticas e teóricas no campo da otimização em redes.

4 Aplicações

4.1 Aplicações do Algoritmo de Needleman-Wunsch na Genética

O algoritmo de Needleman-Wunsch, amplamente utilizado no campo da bioinformática, apresenta diversas aplicações relevantes no estudo de sequências biológicas e no entendimento de processos genéticos e moleculares. Dentre suas principais aplicações, destacam-se:

O algoritmo permite alinhar sequências biológicas de maneira ótima, identificando similaridades e diferenças entre nucleotídeos ou aminoácidos. Essa aplicação é essencial para a análise de genes e a identificação de mutações.

Ao determinar o alinhamento de sequências de organismos diferentes, é possível inferir relações evolutivas, identificar genes homólogos e traçar árvores filogenéticas.

O alinhamento ajuda a identificar regiões conservadas em genes e proteínas, auxiliando no entendimento de funções biológicas e na identificação de famílias proteicas.

A partir do alinhamento de sequências, o algoritmo auxilia na predição de estruturas terciárias de proteínas, baseando-se em similaridades com proteínas de estruturas conhecidas.

4.2 Aplicações do Algoritmo de Smith-Waterman na Genética

O algoritmo de Smith-Waterman é amplamente reconhecido por sua eficiência no alinhamento local de sequências biológicas, sendo uma ferramenta fundamental na bioinformática. Suas principais aplicações incluem:

O algoritmo é utilizado para encontrar as regiões de maior similaridade entre sequências de DNA, RNA ou proteínas, mesmo quando essas apresentam diferenças significativas em outras regiões. Isso é particularmente útil para identificar domínios conservados em genes ou proteínas.

O algoritmo Smith-Waterman é empregado para buscar sequências semelhantes em grandes bases de dados, ajudando a identificar genes relacionados, proteínas com funções semelhantes ou variantes genéticas importantes.

4.3 Aplicações do Algoritmo de Ford-Fulkerson na Genética

O algoritmo de Ford-Fulkerson, originalmente desenvolvido para resolver problemas de fluxo máximo em redes, também encontra aplicações em diversas áreas da genética e da biologia computacional. Entre suas principais utilizações estão:

É possível usar o algoritmo para traçar a transmissão de características genéticas, como no caso da hemofilia. Redes de fluxo podem representar genes e suas probabilidades de transmissão entre gerações, ajudando na análise de padrões de herança.

Redes de fluxo podem ser usadas para modelar interações entre genes, avaliando como diferentes combinações genéticas influenciam características fenotípicas ou suscetibilidade a doenças.

Em estudos evolutivos, o algoritmo pode ajudar a inferir relações entre espécies, modelando fluxos de similaridade genética para determinar os caminhos evolutivos mais prováveis.

O algoritmo pode ser aplicado para modelar o transporte de moléculas, como nutrientes ou proteínas, em redes metabólicas ou celulares, otimizando o entendimento de processos biológicos.

Em genética de populações, o algoritmo auxilia na modelagem de como genes se movem entre diferentes subpopulações, permitindo a análise de conectividade genética e fluxo gênico.

Redes de fluxo podem ser usadas para identificar rotas metabólicas críticas em organismos, modelando o fluxo de substratos ou energia dentro de vias metabólicas.

5 Conclusão

Conclui-se, portanto, que o estudo dos alinhamentos de sequências genéticas desempenha um papel central na genética, possibilitando avanços significativos em áreas como biologia molecular (clonagem), evolução e medicina (mutações do câncer). A aplicação de modelos baseados em grafos e algoritmos eficientes não apenas simplifica a análise de sequências complexas, mas também abre caminhos para novas descobertas e moldes práticos.

Dessa maneira, a integração entre biologia e computação continua a ser uma ferramenta indispensável para lidar com os desafios e oportunidades apresentados por esse campo em constante evolução.

Por fim, a crescente demanda por soluções rápidas e precisas nesse domínio impulsiona o desenvolvimento de tecnologias e metodologias inovadoras. Essas abordagens ampliam as possibilidades de análise, permitindo lidar com volumes de dados cada vez maiores e aumentando a precisão na identificação de padrões e mutações. Assim, o alinhamento de sequências genéticas não apenas contribui para o avanço científico, mas também possui aplicações práticas que impactam diretamente a saúde e pleno bem-estar humano.

Referências

[1] Jonathan L. Gross, Jay Yellen, Ping Zhang. *Handbook of Graph Theory*. 2ª edição. Páginas 1173–1454, 2013.

[2] Marek Kimmel, David E. Axelrod. *Branching Process in Biology*. Volume 19. Páginas 19–31, 2002.

[3] Lee A. Segel, Leah Edelstein-Keshet. *A Primer on Mathematical Models in Biology*. Páginas 251–310, volume 129, 2014.

[4] Bisognin, G., Franco, F. B., Bisognin, V. *Estudo de Grafos e Aplicações*. Páginas 78–79, 2001.

[5] Johnsonbaugh, R. *Discrete Mathematics*. 7ª edição, 2009.

[6] Rangel, S. *Teoria dos Grafos, Notas de aula*, IBILCE, Unesp. 2002-2013.

[7] Goldbarg, M., Goldbarg, E. *Grafos: Conceitos, Algoritmos e Aplicações*, Elsevier, 2012.

[8] Szwarcfiter, J. L. *Teoria Computacional de Grafos: Os Algoritmos*, Ed. Elsevier, 2018. (Inclui algoritmos implementados na linguagem Python por Fabiano S. Oliveira e Paulo E. D. Pinto)

[9] Bioinfo Class. *Alinhamento de Sequências - Blog Bioinfo Class*. Disponível em: `https://www.bioinfoclass.com/blog/alinhamento`. Acessado em: 18 nov. 2024.

[10] PUC Goiás - Professor. *Capítulo 3: Alinhamento de Sequências*. Disponível em: `https://professor.pucgoias.edu.br/SiteDocente/admin/arquivosUpload/18497/material/Cap.%203%20Alinhamento%20de%20sequ%C3%AAncias.pdf`. Acessado em: 18 nov. 2024.

[11] OnlineBioInfo Bioinformática. *Playlist - Alinhamento de sequências*. Disponível em: `https://www.youtube.com/playlist?list=PLVSek57SJFtTdzGvBpmj5qvnqZ-3LL76u`. Acesso em: 19 nov. 2024.

[12] Wikipédia. Algoritmo de Smith-Waterman. Disponível em: `https://en.wikipedia.org/wiki/Smith-Waterman_algorithm`. Acesso em: 20 de novembro de 2024.

[13] Bio Scholar. Vídeo: Smith Waterman Algorithm. Disponível em: `https://www.youtube.com/watch?v=bFDRny7T3_s`. Acesso em: 20 de novembro de 2024.

[14] Alex Oliveira. Repositório *seq-genetico*. Disponível em: `https://github.com/alexoliveiraFGV24/seq-genetico.git`. Acesso em: 21 nov. 2024.

[15] Kamil Slowikowski. Arquivo *needleman-wunsch*. Disponível em: `https://gist.github.com/slowkow/06c6dba9180d013dfd82bec217d22eb5`. Acesso em: 20 nov. 2024.