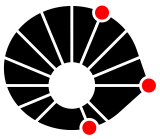

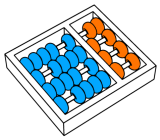

Ordenações Semi-Completas por Operações de Rearranjos de Genomas

J. P. Vianini *G. Siqueira* *Z. Dias*

Relatório Técnico - IC-PFG-24-43
Projeto Final de Graduação
2024 - Dezembro

UNIVERSIDADE ESTADUAL DE CAMPINAS
INSTITUTO DE COMPUTAÇÃO

Ordenações Semi-Completas por Operações de Rearranjos de Genomas

J. P. Vianini* Gabriel Siqueira* Zanoni Dias*

Dezembro de 2024

Resumo

Este trabalho apresenta uma variação do clássico problema da Distância de Rearranjos, cujo objetivo é identificar uma sequência de rearranjos que transforma um genoma em outro, respeitando critérios específicos de proximidade baseados em limiares. São exploradas variações baseadas em inversões, λ-permutações e entropia. Para cada uma das variações foi realizado um estudo do impacto do valor do limiar no problema, uma abordagem de aproximação e uma heurística que pode auxiliar na abordagem de aproximação.

Abstract

This work presents a variation of the classic Rearrangement Distance problem called the Semi-Complete Sorting by Rearrangement Events problem. This variation aims to identify a sequence of rearrangements that transforms one genome into another, adhering to specific proximity criteria based on thresholds. Variations based on inversions, λ-permutations, and entropy are explored. For each variation, a study was conducted on the impact of the threshold value on the problem, an approximation approach was proposed, and a heuristic that can support the approximation approach was developed.

1 Introdução

Desde a finalização do Projeto Genoma Humano, o que marcou o início da era pós-genômica na Biologia, o campo de Genômica Comparativa assumiu um papel central, não somente nas áreas de Bioinformática e Biologia Computacional, mas no âmbito geral das pesquisas biológicas. Um dos maiores desafios nessa disciplina é determinar o quão próximos dois organismos são a partir das similaridades e diferenças em seus materiais genéticos.

*Instituto de Computação, Universidade Estadual de Campinas, Campinas, SP

Partindo do *princípio da parcimônia*, o número mínimo de eventos de rearranjo, chamado de *distância de rearranjos* é um método amplamente adotado para estimar a distância evolutiva entre duas espécies a partir de genomas de dois indivíduos [6, 10]. Um *rearranjo* de um genoma é uma mutação em larga escala que altera a ordem e a orientação dos genes em um genoma. Dado um certo genoma, se algumas condições são satisfeitas, então ele pode ser representado por uma permutação.

As subseções seguintes são destinadas a uma introdução ao clássico Problema da Distância de Rearranjos e a conceitos básicos sobre esse problema. Na Seção 2 estão definidos conceitos adicionais necessários para a definição do Problema das Ordenações Semi-Completas por Operações de Rearranjos de Genomas, que acontece na Seção 3 e é o foco deste trabalho. Na Seção 4, uma heurística é apresentada que pode ser utilizada para ajudar a encontrar uma aproximação para esse problema. A Seção 5 apresenta uma conclusão para o trabalho.

1.1 Espaços Métricos e Permutações

Uma *métrica* d em um conjunto S é uma função $d : S \times S \to \mathbb{R}$ que satisfaz três propriedades para quaisquer $s, t, u \in S$: (i) $d(s,t) \geq 0$ com $d(s,t) = 0$ se, e somente se, $s = t$ (positividade); (ii) $d(s,t) = d(t,s)$ (simetria); e (iii) $d(s,u) \leq d(s,t) + d(t,u)$ (desigualdade triangular). Um conjunto S acompanhado de uma métrica d é chamado de *espaço métrico* e é denotado por (S,d). As distâncias de rearranjos, que são o interesse deste trabalho, são sempre métricas nos conjuntos de genomas. Quando os genomas sendo comparados (i) possuem apenas um cromossomo linear, (ii) compartilham o mesmo conjunto de n genes e (iii) não possuem genes repetidos, cada um dos genomas pode ser representado por uma permutação.

Seja $S = \{1, 2, \ldots, n\}$. Uma *permutação* de S é uma bijeção de S em S. O conjunto de todas as $n!$ permutações dos n elementos de S é denotado por S_n. Se $i_1, i_2, \ldots, i_n$ é um arranjo dos elementos de S e π é uma permutação que mapeia k em i_k, para todo $k \in S$, então uma notação clássica para denotar a permutação π é a notação de duas linhas

$$\pi = \begin{pmatrix} 1 & 2 & 3 & \cdots & n \\ i_1 & i_2 & i_3 & \cdots & i_n \end{pmatrix}$$

que é simplesmente uma variação da notação clássica de funções

$$\pi(1) = i_1,\ \pi(2) = i_2,\ \cdots,\ \pi(n) = i_n,$$

ou, ainda, da representação da função como o conjunto de pares ordenados

$$\pi = \{(1, i_1), (2, i_2), \ldots, (n, i_n)\},$$

de forma que cada coluna na notação de duas linhas é um par ordenado do conjunto. Por esse motivo, a ordem das colunas nesse tipo de notação é irrelevante. Assim, é sempre possível reordenar as colunas de forma que a primeira linha fique em ordem crescente. Dada uma permutação α, a *inversa* α^{-1} de α é a permutação $\{(j,i) \mid (i,j) \in \alpha\}$. Essa definição é equivalente à definição de função inversa de uma bijeção. Na notação de duas linhas, é

possível obter a permutação inversa trocando-se as linhas. Portanto,

$$\pi^{-1} = \begin{pmatrix} i_1 & i_2 & i_3 & \cdots & i_n \\ 1 & 2 & 3 & \cdots & n \end{pmatrix}.$$

Na literatura de rearranjo de genomas, é comum que a imagem de $k \in S$ em uma permutação π seja denotada por $\pi(k) = \pi_k$.

O *produto* (ou *composição*) de duas permutações π e σ de um conjunto S é uma operação em S_n e é denotado por $\pi \circ \sigma$. O significado de $\pi \circ \sigma$, oriundo da composição de funções, é que as permutações π e σ são aplicadas da direita para a esquerda, isto é, primeiro se aplica σ e ao resultado se aplica π. O conjunto S_n de todas as permutações de S junto à operação de produto $\circ$ define o *grupo simétrico*, denotado por $\mathcal{G}$.

A permutação *identidade* é

$$\iota = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & \cdots & n \\ 1 & 2 & 3 & 4 & 5 & \cdots & n \end{pmatrix}$$

uma vez que para toda permutação α, $\iota \circ \alpha = \alpha \circ \iota = \alpha$.

Uma notação tradicional utilizada na literatura de rearranjo de genomas é a *notação de uma linha* que mantém apenas a segunda linha, isto é,

$$\pi = \begin{pmatrix} 1 & 2 & \cdots & n \\ \pi_1 & \pi_2 & \cdots & \pi_n \end{pmatrix} = \begin{pmatrix} \pi_1 & \pi_2 & \cdots & \pi_n \end{pmatrix},$$

na qual supõe-se que a primeira linha (omitida) está ordenada de forma crescente.

A *extensão linear* de uma permutação de $S = \{1, 2, \ldots, n\}$ é a permutação π^ℓ de $\{0, 1, \ldots, n+1\}$, definida por $\pi^\ell = \begin{pmatrix} 0 & \pi_1 & \pi_2 & \cdots & \pi_n & n+1 \end{pmatrix}$.

1.2 Problema da Distância de Rearranjos

Uma das conveniências no uso de permutações para representação de genomas é que, como o produto é uma operação em $\mathcal{G}$, é também possível utilizar permutações para representar rearranjos. Isso é feito de forma com que o produto de uma permutação que representa o genoma original com a permutação que representa o rearranjo resulta na permutação que representa o genoma após o rearranjo, ou seja, um rearranjo ρ transforma um genoma π no genoma $\pi \circ \rho$. Assim, encontrar uma sequência de rearranjos que transformam uma permutação π em uma permutação σ é equivalente a encontrar uma sequência $\rho_1, \ldots, \rho_k$ tais que $\pi \circ \rho_1 \circ \cdots \circ \rho_k = \sigma$, ou, equivalentemente $\sigma^{-1} \circ \pi = \rho_k^{-1} \circ \cdots \rho_1^{-1}$.

Se o conjunto de rearranjos permitidos é tal que é sempre possível obter qualquer permutação em $\mathcal{G}$ pelo produto desses rearranjos, eles são ditos *geradores* de $\mathcal{G}$. Dado um conjunto G de geradores de $\mathcal{G}$, uma *sequência transformadora* β por geradores de G para permutações π e σ é uma sequência $\iota, \rho_1, \ldots, \rho_k$, tal que $\sigma^{-1} \circ \pi = \iota \circ \rho_k^{-1} \circ \cdots \circ \rho_1^{-1}$ em que $\rho_i \in G$, $1 \leq i \leq k$. Note que ι é o primeiro elemento de toda sequência transformadora. O *tamanho* de uma sequência transformadora $\beta = \iota, \rho_1, \ldots, \rho_k$ é o número de elementos de β diferentes de ι. Uma *distância* entre duas permutações π e σ pode ser definida como o tamanho de uma sequência transformadora mínima para π e σ. A Proposição 1.1 estabelece que essa distância é uma métrica.

Proposição 1.1. *Seja $\mathcal{G} = (S_n, \circ)$, em que $S = \{1, 2 \ldots, n\}$, e G um conjunto de geradores de $\mathcal{G}$. Então, $d : S_n \times S_n \to \mathbb{N}$, em que para todas as permutações $\pi, \sigma \in S_n$, $d(\pi, \sigma)$ é o tamanho de uma sequência transformadora mínima para π e σ, é uma métrica em $\mathcal{G}$.*

Demonstração. Seja $\mathcal{G} = (S_n, \circ)$, em que $S = \{1, 2 \ldots, n\}$ e G um conjunto de geradores de $\mathcal{G}$. Para mostrar que d é uma métrica, é suficiente mostrar que as propriedades de positividade, simetria e a desigualdade triangular são satisfeitas. Sejam $\pi, \sigma, \tau \in S_n$.

O tamanho k de uma sequência transformadora é um número natural, logo $k \geq 0$. Suponha que $d(\pi, \sigma) = 0$. Então, todos os elementos na sequência transformadora são iguais a ι. Como $\iota^{-1} = \iota$ e $\iota \circ \iota = \iota$, tem-se que $\sigma^{-1} \circ \pi = \iota$. Portanto, $\pi = \sigma$. Suponha, agora, que $\pi = \sigma$. Então $\sigma^{-1} \circ \pi = \iota$. Segue que $d(\sigma, \pi) = 0$. Assim, $d(\pi, \sigma) \geq 0$ e $d(\pi, \sigma) = 0$ se, e somente se, $\pi = \sigma$.

Suponha que $d(\pi, \sigma) = k$. Então, existe uma sequência transformadora $\iota, \rho_1, \rho_2, \ldots, \rho_k$ tal que $\sigma^{-1} \circ \pi = \rho_k^{-1} \circ \cdots \circ \rho_1^{-1}$. Como, para quaisquer bijeções α e β em um conjunto S, $(\alpha \circ \beta)^{-1} = \beta^{-1} \circ \alpha^{-1}$, tem-se, tomando a inversa dos dois lados que

$$(\sigma^{-1} \circ \pi)^{-1} = (\rho_k^{-1} \circ \cdots \circ \rho_1^{-1})^{-1}$$
$$\pi^{-1} \circ \sigma = \rho_1 \circ \cdots \circ \rho_k.$$

Portanto, $\iota, \rho_k^{-1}, \ldots, \rho_1^{-1}$ é uma sequência transformadora por geradores de G para σ e π de tamanho k. Logo, $d(\sigma, \pi) = k$.

Suponha, agora, que $d(\pi, \sigma) = k_1$ e $d(\sigma, \tau) = k_2$. Suponha, para obter uma contradição, que $d(\pi, \tau) > k_1 + k_2$. Seja α uma sequência transformadora mínima de tamanho k_1 que transforma π em σ e β uma sequência transformadora mínima de tamanho k_2 que transforma σ em τ. Seja γ a concatenação de α e β. Então, γ é uma sequência transformadora que transforma π em τ com tamanho $k_1 + k_2$, o que é uma contradição. Logo, $d(\pi, \tau) \leq k_1 + k_2$. □

Para simplificar o enunciado do problema, é comum exigir que a distância tenha uma propriedade adicional. Uma distância d em $\mathcal{G}$ é *invariante à esquerda* se, para todo π, σ e τ em $\mathcal{G}$, $d(\pi, \sigma) = d(\tau \circ \pi, \tau \circ \sigma)$. Caso essa propriedade seja satisfeita, o problema fica reduzido a computar a distância entre as permutações $\sigma^{-1} \circ \pi$ e ι. Uma vez que, na maior parte do tempo, o foco é a distância entre uma permutação π e a permutação identidade ι, abrevia-se $d(\pi, \iota)$ para $d(\pi)$.

Dada uma permutação π, uma sequência de k rearranjos $\rho_1, \rho_2, \ldots, \rho_k$ é uma *sequência ordenadora* para π se $\pi \circ \rho_1 \circ \cdots \circ \rho_k = \iota$. O Problema 1 é o Problema da Distância de Rearranjos.

Problema 1 (Distância de Rearranjos por Ordenação de Genomas)**.**
ENTRADA: Permutação π.
OBJETIVO: Encontrar uma *sequência ordenadora* $\rho_1, \rho_2, \ldots, \rho_k$ com k mínimo para π.

O restante desta subseção é destinado à descrição de algumas variantes comuns do Problema 1.

1.2.1 Distância de Reversão

O primeiro problema a ser combinatoriamente estudado na área de rearranjos genômicos foi o problema da distância de reversão. A *reversão* do intervalo $[i, j]$ é a permutação

$$\rho = \begin{pmatrix} 1 & \cdots & i-1 & i & i+1 & \cdots & j-1 & j & j+1 & \cdots & n \\ 1 & \cdots & i-1 & j & j-1 & \cdots & i+1 & i & j+1 & \cdots & n \end{pmatrix}.$$

No problema da distância de reversão, o conjunto G de geradores é composto por todas as reversões em S_n. Dessa forma, o problema da *distância de reversão* é, dada uma permutação π, encontrar uma sequência ordenadora mínima $\rho_1, \rho_2, \ldots, \rho_k$ em que, para todo $i \in \{1, \ldots, k\}$, ρ_i é uma reversão. Esse problema foi introduzido por Watterson et al. [11] em 1982. Em 1999, Caprara [4] mostrou que o problema é $\mathcal{NP}$-difícil.

Ao descrever o problema, Watterson et al. [11] apresentaram um algoritmo aproximado polinomial que é uma 2-aproximação. A melhor aproximação conhecida é uma $\frac{11}{8}$-aproximação apresentada por Berman et al. [2] em 2002.

1.2.2 Distância de Transposição

Um dos problemas mais conhecidos de distância de rearranjo é o problema da distância de transposição. A *transposição* dos intervalos $[i, j-1]$ e $[j, k-1]$ é a permutação

$$\rho = \begin{pmatrix} 1 & \cdots & i-1 & i & i+1 & \cdots & j-2 & j-1 & j & j+1 & \cdots & k-1 & k & \cdots & n \\ 1 & \cdots & i-1 & j & j+1 & \cdots & k-1 & i & i+1 & \cdots & j-2 & j-1 & k & \cdots & n \end{pmatrix}.$$

No problema da distância de transposição, o conjunto G de geradores é composto por todas as transposições em S_n. Dessa forma, o problema da *distância de transposição* é, dada uma permutação π, encontrar uma sequência ordenadora mínima $\rho_1, \rho_2, \ldots, \rho_k$ em que, para todo $i \in \{1, \ldots, k\}$, ρ_i é uma transposição. Esse problema foi introduzido por Bafna e Pevzner [1] em 1998. Em 2012, Bulteau et al. [3] mostraram que o problema da distância de transposição é $\mathcal{NP}$-difícil.

A melhor aproximação conhecida para o problema da distância de transposição é uma $\frac{11}{8}$-aproximação devido a Elias e Hartman [5].

1.3 Breakpoints e Strips

Seja π uma permutação em um conjunto $S = \{1, 2, \ldots, n\}$ e π^ℓ sua extensão linear. Um *ponto* de π é um par ordenado $(\pi_i^\ell, \pi_{i+1}^\ell)$, para $0 \leq i \leq n$. Se $|\pi_{i+1}^\ell - \pi_i^\ell| = 1$, o ponto é uma *adjacência*. Se $|\pi_{i+1}^\ell - \pi_i^\ell| \neq 1$, então o ponto é chamado de *breakpoint*. Note que ι é a única permutação em S_n que não possui breakpoints. Como existem $n+1$ pontos na permutação, o número máximo de breakpoints é $n+1$. O *número de breakpoints* de π é denotado por $bp(\pi)$.

Ainda que não corresponda explicitamente a eventos de rearranjo, breakpoints podem ser utilizados como uma medida de similaridade entre genomas representados por permutações. A *distância de breakpoints* entre permutações π e σ, denotada $d_{bp}(\pi, \sigma)$, é definida como o número de breakpoints na permutação $\sigma^{-1} \circ \pi$, isto é $d_{bp}(\pi, \sigma) = bp(\sigma^{-1} \circ \pi)$. Essa distância corresponde ao número de adjacências em uma permutação que não são adjacências na outra.

Uma *strip* de π é um intervalo $[i, j]$ de π^ℓ tal que $(i-1, i)$ e $(j, j+1)$ são breakpoints e $(k, k+1)$ é uma adjacência para todo $i \leq k < j$. Uma strip $[i, j]$ de π é *decrescente* se $\pi^\ell_i > \pi^\ell_{i+1} > \cdots > \pi_j$. Uma strip de um elemento é sempre decrescente, a menos de π^ℓ_0 e π^ℓ_{n+1}, que são sempre crescentes.

Strips e breakpoints são comumente utilizados para obter limitantes e aproximações para distâncias de rearranjo. Diz-se que um rearranjo ρ remove k breakpoints de uma permutação π se $bp(\pi \circ \rho) = bp(\pi) - k$. O Lema 1.2 é devido a Kececioglu e Sankoff [7].

Lema 1.2. *Para toda permutação π que contém uma strip decrescente, existe uma reversão que remove ao menos um breakpoint de π.*

Demonstração. Seja $[i, j]$ a strip decrescente de π cujo último elemento, π_j é o menor. O elemento $\pi_j - 1$ deve estar em uma strip crescente (caso contrário, π_i não é o menor) que está ou à esquerda ou à direita da strip que contém π_i. Se a strip crescente está à esquerda da strip decrescente, a reversão do intervalo iniciado no primeiro elemento após o fim da strip crescente e terminado no último elemento da strip decrescente, quando aplicada em π, remove ao menos um breakpoint. Se a strip crescente está à direita da strip decrescente, a reversão do intervalo iniciado em $j+1$ e terminado na posição do elemento $\pi_j - 1$, quando aplicada em π, remove ao menos um breakpoint. □

O Lema 1.3 também é devido a Kececioglu e Sankoff [7].

Lema 1.3. *Seja π uma permutação com uma strip decrescente. Se toda reversão que remove um breakpoint de π resulta em uma permutação sem strips decrescentes, então π possui uma reversão que remove dois breakpoints.* □

Os Lemas 1.2 e 1.3 são a base de um algoritmo guloso proposto por Kececioglu e Sankoff [7] que consegue ordenar uma permutação π com no máximo $bp(\pi) - 1$ reversões. Esse fato, aliado à observação, utilizada pelo algoritmo, de que uma reversão pode reduzir o número de breakpoints em no máximo dois, tem-se os seguintes limitantes para a distância d de reversões:

$$\frac{bp(\pi)}{2} \leq d(\pi) \leq bp(\pi) - 1. \tag{1}$$

As observações da Equação (1) são também o motivo pelo qual o algoritmo guloso é uma 2-aproximação para a distância de reversão.

Dada uma strip $[i, j]$ de uma permutação π, denota-se por $adj\_pos(i)$ a posição $\pi^{-1}_{\pi_i - 1}$, se $\pi_i - 1$ não pertence a strip $[i, j]$, ou a posição $\pi^{-1}_{\pi_i + 1}$, se $\pi_i + 1$ não pertence a strip $[i, j]$. A posição $adj\_pos(j)$ é definida de forma similar.

O Lema 1.4 mostra uma relação entre quebrar e combinar strips e a remoção de breakpoints e seu corolário, Corolário 1.5 é utilizado na Seção 3.

Lema 1.4. *Seja π uma permutação e $[i, j]$ uma strip de π. Se uma quebra da strip $[i, j]$ em $k \geq 3$ substrips seguida do rearranjo dessas substrips (com reversões e transposições) resulta em uma permutação com menos breakpoints que π, então $k-1$ quebras foram desfeitas pelo rearranjo.*

Demonstração. Seja π uma permutação e $[i, j]$ uma strip de π. Considere que a strip $[i, j]$ foi quebrada em $k \geq 3$ substrips $[i, t_1]$, $[t_1 + 1, t_2]$, ..., $[t_k, j]$. Note que cada uma dessas quebras introduz um breakpoint, portanto, para que a permutação resultante tenha menos breakpoints que π, o rearranjo deve remover $k + 1$ breakpoints. É possível remover um breakpoint movendo a strip $[i, t_1]$ para que ela fique adjacente à posição $adj\_pos(i)$ e é possível remover um breakpoint movendo a strip $[t_k, j]$ para que ela fique adjacente à posição $adj\_pos(j)$. Dessa forma, ainda são necessárias as remoções de $k - 1$ breakpoints. Seja $[t_\ell, t_{\ell+1}]$ uma das substrips resultantes da quebra de $[i, j]$ diferente de $[i, t_1]$ e $[t_k, j]$. Para remover breakpoints, a substrip $[t_\ell, t_{\ell+1}]$ deve ser movida para ser concatenada com elementos adjacentes, isto é, ela deve ser concatenada com a strip do elemento $\pi_{t_\ell} - 1$ ou com a strip do elemento $\pi_{t_\ell+1} + 1$. Contudo, como se trata de uma substrip interna de $[i, j]$, a strip do elemento $\pi_{t_\ell} - 1$ e a strip do elemento $\pi_{t_\ell+1} + 1$ também são substrips de $[i, j]$, então essa concatenação desfaz uma das quebras realizadas. Como há $k - 1$ breakpoints restantes a serem removidos, há $k - 1$ concatenações a serem feitas que desfazem quebras realizadas e o resultado segue.

□

Corolário 1.5. *Seja π uma permutação e $[i, j]$ uma strip de π. e uma quebra da strip $[i, j]$ em substrips seguida do rearranjo dessas substrips (com reversões e transposições) resulta em uma permutação com menos breakpoints que π, então $[i, j]$ foi quebrada em duas substrips.*

Demonstração. Seja π uma permutação e $[i, j]$ uma strip de π. Suponha, para obter uma contradição, que a strip $[i, j]$ foi quebrada em $k \geq 3$ substrips. Então, pelo Lema 1.4, $k - 1$ quebras devem ser desfeitas, o que é equivalente a quebrar a strip em duas substrips. □

2 Preliminares

Nesta seção, são definidos conceitos que são fundamentais para definir as variantes do problema central deste projeto. A Subseção 2.1 trata do conceito de inversões, a Subseção 2.2 trata do conceito de λ-permutações e a Subseção 2.3 trata do conceito de entropia.

2.1 Inversões

Sejam π e σ permutações. Um par de elementos (π_i, π_j), $i < j$, de π é chamado de *par invertido* ou *inversão* em relação a σ se $\sigma^{-1}_{\pi_i} > \sigma^{-1}_{\pi_j}$. O *número de inversões* entre π e σ, denotado por $\text{inv}(\pi, \sigma)$ é o número de pares invertidos de π em relação a σ. Se $\sigma = \iota$, então $\text{inv}(\pi, \iota) = \text{inv}(\pi)$ é o número de pares de elementos (π_i, π_j) , $i < j$, tais que $\pi_i > \pi_j$.

O Lema 2.1 estabelece uma importante característica de permutações com pares invertidos.

Lema 2.1. *Sejam π e σ permutações de $S = \{1, \ldots, n\}$. Se existe um par invertido em π em relação a σ, então existe um par invertido em π em relação a σ cujos elementos são consecutivos.*

Demonstração. Sejam π e σ permutações e suponha que (π_i, π_j), $i < j$ é um par invertido em relação a σ. Então, $\sigma_{\pi_i}^{-1} > \sigma_{\pi_j}^{-1}$. Se $j = i + 1$, o resultado segue. Suponha, então, que $j \neq i + 1$. Então, existem $(j - i) \geq 1$ elementos entre π_i e π_j. Suponha, para obter uma contradição, que não existam elementos π_k, π_{k+1}, $i \leq k < j$ tais que $\sigma_{\pi_k}^{-1} > \sigma_{\pi_{k+1}}^{-1}$. Portanto, para todo π_k, $i \leq k < j$, $\sigma_{\pi_k}^{-1} < \sigma_{\pi_{k+1}}^{-1}$. Como não há elementos repetidos na sequência $\pi_i, \ldots, \pi_j$, isso implica que $\sigma_{\pi_i}^{-1} < \sigma_{\pi_{i+1}}^{-1} < \cdots < \sigma_{\pi_j}^{-1}$, o que contradiz o dado de que $\sigma_{\pi_i}^{-1} > \sigma_{\pi_j}^{-1}$. Assim, existem elementos π_k, π_{k+1}, $i \leq k < j$ tais que $\sigma_{\pi_k}^{-1} > \sigma_{\pi_{k+1}}^{-1}$ e o resultado segue. □

Se (i, j) é um par invertido de π em relação a σ e $j = i + 1$, então o par é chamado de *inversão adjacente*.

Sejam $\pi = \begin{pmatrix} 3 & 5 & 2 & 1 & 4 \end{pmatrix}$ e $\sigma = \begin{pmatrix} 4 & 5 & 2 & 3 & 1 \end{pmatrix}$. Então, $(3,4)$, $(5,4)$, $(2,4)$, $(1,4)$, $(3,5)$ e $(3,2)$ são inversões de π em relação a σ. Logo, $\text{inv}(\pi, \sigma) = 6$. Caso o objetivo seja comparar o número de inversões entre π e $\iota = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \end{pmatrix}$, tem-se que $(3,1)$, $(3,2)$, $(5,1)$, $(5,2)$, $(5,4)$ e $(2,1)$ são as inversões de π em relação à ι. Logo, $\text{inv}(\pi, \iota) = \text{inv}(\pi) = 6$.

Uma consequência do Lema 2.1 é que $\text{inv}(\pi, \sigma) = x$ se, e somente se, é possível transformar π em σ trocando de posição x inversões adjacentes.

2.2 λ-Permutações

Sejam π e λ permutações de um conjunto S. O *limite de deslocamento* entre π e σ, denotado por $\text{ldes}(\pi, \sigma)$, é definido como $\text{ldes}(\pi, \sigma) = \max\{|\sigma_i^{-1} - \pi_i^{-1}| : i \in \{1, \ldots, n\}\}$. Seja λ um inteiro maior ou igual a dois. A permutação σ é uma *λ-permutação* de uma permutação π se $\text{ldes}(\pi, \iota) < \lambda$. De certa forma, as λ-permutações permitem que se restrinja a distância entre a posição de um elemento na permutação e sua posição na permutação original, ou seja, essas permutações permitem uma flexibilidade limitada na posição dos elementos. Esse fator é útil para analisar configurações em que cada elemento não desvia muito de sua posição na permutação de referência. Se a permutação original for ι, é possível determinar que uma permutação σ é uma λ-permutação de ι verificando que $|\sigma_i - i| < \lambda$.

Se $n = 5$ e $\pi = \begin{pmatrix} 1 & 2 & 3 & 5 & 4 \end{pmatrix}$, então $\sigma = \begin{pmatrix} 2 & 1 & 3 & 4 & 5 \end{pmatrix}$ é uma 2-permutação de π, porque cada elemento em σ está a uma distância de no máximo uma posição de sua posição original em π. Da mesma forma, $\tau = \begin{pmatrix} 3 & 1 & 4 & 2 & 5 \end{pmatrix}$ é uma 3-permutação de π, pois cada elemento em τ está a uma distância de no máximo duas posições de sua posição original em π.

2.3 Entropia

Sejam π e σ permutações. A *entropia* entre π e σ, denotada por $\text{entr}(\pi, \sigma)$, é dada por $\sum_{i=1}^{n} |\sigma_i^{-1} - \pi_i^{-1}|$. De certa forma, a entropia mede a *desordem* ou o quanto a permutação σ se desvia da permutação π. Ao se restringir que permutações não excedam uma determinada entropia de uma permutação referência, a soma dos deslocamentos dos elementos deve ser fixa, ou seja, ainda que um elemento se desloque muito, é possível que isso seja "compensado" por deslocamentos menores de outros elementos.

Considere as permutações $\pi = \begin{pmatrix}5 & 3 & 2 & 4 & 1\end{pmatrix}$ e $\sigma = \begin{pmatrix}1 & 5 & 4 & 3 & 2\end{pmatrix}$. Então, $|\sigma_1^{-1} - \pi_1^{-1}| = |1-5| = 4$, $|\sigma_2^{-1}-\pi_2^{-1}| = |5-3| = 2$, $|\sigma_3^{-1}-\pi_3^{-1}| = |4-2| = 2$, $|\sigma_4^{-1}-\pi_4^{-1}| = |3-4| = 1$ e $|\sigma_5^{-1} - \pi_5^{-1}| = |2-1| = 1$. Portanto, $\text{entr}(\pi, \sigma) = 10$.

3 Ordenações Semi-Completas

Neste trabalho, o foco central está em um tipo de problema de rearranjos de genomas em que o objetivo é encontrar uma sequência de rearranjos que transforma π em uma permutação σ que é *próxima o suficiente* de ι por algum parâmetro. Para formalizar esse problema, algumas definições se mostram necessárias.

Métricas são utilizadas para determinar distâncias entre dois elementos em um espaço métrico. Quando o interesse está em determinar a distância de um elemento a um conjunto de elementos, novos conceitos são necessários. Denota-se por $\mathscr{P}(S)$ o conjunto de todos os subconjuntos de S. Uma *distância* (elemento-conjunto) em um conjunto S é uma função $d : S \to \mathscr{P}(S)$ que satisfaz as seguintes propriedades para todo $x \in S$, $S_1, S_2 \in \mathscr{P}(S)$: $(i.)$ $d(x, \{x\}) = 0$; $(ii.)$ $d(x, \varnothing) = \infty$; $(iii.)$ $d(x, S_1 \cup S_2) = \min\{d(x, S_1), d(x, S_2)\}$; e $(iv.)$ se $S_1^k = \{x \in S \mid d(x, S_1) \leq k\}$, então $d(x, S_1) \leq d(x, S_1^k) + k$, para todo $k \in \mathbb{R}_{\geq 0}$. Um *espaço de aproximação* é um par (S, d) em que S é um conjunto e d é uma distância elemento-conjunto em S. Mais informações sobre espaços de aproximação podem ser encontradas no livro de R. Lowen [8].

Para os propósitos deste trabalho, é suficiente definir uma *distância* d de uma permutação π de um conjunto S a um conjunto $A \subseteq S_n$ a partir de uma métrica d', isto é, para todo $\pi \in S_n$ e todo conjunto $A \in \mathscr{P}(S_n)$ $d(s, A) = \min\{d'(\pi, \sigma) \mid \sigma \in A\}$. Vale notar que, devido à simetria da métrica, a menor distância de π até A é a menor distância de um elemento de A até π. Dessa forma, embora o primeiro parâmetro da métrica seja um elemento e o segundo um subconjunto, essa distância vale nas duas direções. Além disso, a distância pode ser calculada através de geradores, considerando-se a menor sequência transformadora por geradores de π a um elemento de A. Pode-se, então, falar de uma *sequência transformadora mínima* entre um conjunto A e uma permutação π como sendo uma sequência transformadora entre π e uma permutação $\sigma \in A$ tal que $d(\pi, \sigma)$ é mínima. O problema de encontrar a menor distância de uma permutação a um conjunto A é chamado de *problema das ordenações semi-completas por operações de rearranjos de genomas*. Seja ψ uma função que associa a um par de permutações uma medida de similaridade, então esse problema está definido no Problema 2.

Problema 2 (Ordenações Semi-Completas por Operações de Rearranjos de Genomas)**.**
Entrada: Permutação π, função de comparação ψ e limiar k.
Objetivo: Encontrar uma *sequência transformadora* entre π e o conjunto $\{\sigma \mid \psi(\sigma, \iota) \leq k\}$.

Neste trabalho, as funções ψ que são consideradas são inv, ldes ou entr. Todas essas funções são invariantes à esquerda e simétricas, portanto pode-se supor que essas propriedades são válidas para ψ no restante deste trabalho.

3.1 Problema Equivalente

Considere o Teorema 3.1, apresentado abaixo.

Teorema 3.1. *Seja π uma permutação. Seja ψ a função de comparação e A' o conjunto de todas as permutações σ tais que $\psi(\sigma, \iota) \leq k$. Então, o conjunto $A = \{\tau \mid \psi(\pi^{-1}, \tau) \leq k\}$ é tal que para todo $\sigma \in A'$, existe um $\tau \in A$ de forma que $d(\pi, \sigma) = d(\tau, \iota)$.*

Demonstração. Seja π uma permutação, ψ uma função de comparação e k um limiar. Dada uma permutação $\sigma \in A'$, seja $\tau = \pi^{-1} \circ \sigma$. Como ψ é invariante à esquerda e simétrica, $\psi(\sigma, \iota) = \psi(\iota, \sigma) = \psi(\pi^{-1}, \pi^{-1} \circ \sigma) = \psi(\pi^{-1}, \tau)$. Logo, $\tau \in A$. Além disso, como a distância é simétrica e invariante à esquerda, $d(\pi, \sigma) = d(\sigma, \pi) = d(\pi^{-1} \circ \sigma, \pi^{-1} \circ \pi) = d(\tau, \iota)$ e o resultado segue.

□

Considere, agora, o Problema 3 apresentado abaixo.

Problema 3 (Ordenações Semi-Completas por Operações de Rearranjos de Genomas - Versão 2)**.**
ENTRADA: Conjunto A de permutações.
OBJETIVO: Encontrar uma sequência transformadora mínima entre um conjunto A e ι.

O Teorema 3.2 apresenta uma relação entre os Problemas 2 e 3.

Teorema 3.2. *O Problema 2 se reduz ao Problema 3.*

Demonstração. Considere uma instância do Problema 2, ou seja, suponha que π é uma permutação, ψ uma função de comparação e k um limiar. Considere o conjunto $A' = \{\sigma \mid \psi(\sigma, \iota) \leq k\}$. Então, pelo Teorema 3.1, existe um conjunto A tal que para todo $\sigma \in A'$, existe uma permutação $\tau \in A$ de forma que $d(\pi, \sigma) = d(\tau, \iota)$. Seja A, formado como $\{\pi^{-1} \circ \sigma \mid \sigma \in A'\}$ - conforme o Teorema 3.1-, o conjunto de entrada de uma instância do Problema 3.

Mostra-se que uma distância entre π um $\sigma \in A'$ é ótima se, e somente se, a distância entre $\pi^{-1} \circ \sigma \in A$ e ι é ótima.

($\rightarrow$) Seja $\sigma \in A'$ tal que $d(\pi, \sigma) = t$ é mínima. Seja $\tau \in A$, tal que $\tau = \pi^{-1} \circ \sigma$. Pela forma que τ foi definido, $d(\tau, \iota) = t$. Suponha, para obter uma contradição, que exista uma permutação $\alpha \in A$ tal que $d(\alpha, \iota) < t$. Então, $\beta = \pi \circ \alpha \in A'$ e, como $d(\alpha, \iota) = d(\pi, \beta)$, $d(\pi, \beta) < t$, uma contradição. Portanto, $d(\tau, \iota)$ é mínimo dentre todos os elementos de A.

($\leftarrow$) Seja $\tau \in A$ tal que $d(\tau, \iota) = t$ é mínima. Seja $\sigma \in A'$, tal que $\sigma = \pi \circ \tau$. Então, $d(\pi, \sigma) = t$. Suponha, para obter uma contradição, que exista uma permutação $\alpha \in A'$ tal que $d(\pi, \alpha) < t$. Então, $\beta = \pi^{-1} \circ \alpha \in A$ e, como $d(\pi, \beta) = d(\alpha, \iota)$, $d(\alpha, \iota) < t$, uma contradição. Portanto, $d(\pi, \sigma)$ é mínimo dentre todos os elementos de A'. □

Devido a essa equivalência, o foco deste trabalho está no Problema 3.

3.2 Abordagem de Aproximação

Esta seção apresenta uma redução que pode ser utilizada para prover algoritmos que garantem um fator de aproximação para os problemas tratados neste trabalho.

Teorema 3.3. *Considere duas versões do Problema 3:*

- *versão* 1*: considera-se uma distancia* $d_{bp}(\tau, \iota) = bp(\tau)$*;*
- *versão* 2*: considera-se uma distancia* $d(\tau, \iota)$*, que pode ser aproximada por* d_{bp}*, ou seja,* $\frac{d_{bp}(\tau,\iota)}{\alpha} \leq d(\tau, \iota) \leq \beta d_{bp}(\tau, \iota)$*, para dois inteiros* α *e* β*.*

Então, uma solução para a versão 1 *do problema é uma* $\alpha\beta$*-aproximação para a versão* 2*.*

Demonstração. Para cada permutação $\tau \in A$ tem-se $\frac{d_{bp}(\tau,\iota)}{\alpha} \leq d(\tau, \iota) \leq \beta d_{bp}(\tau, \iota)$. Seja τ' a permutação obtida na versão 1 do problema, que minimiza d_{bp} e τ'' a permutação obtida na versão 2 do problema, que minimiza d. Portanto, $\frac{d_{bp}(\tau',\iota)}{\alpha} \leq \frac{d_{bp}(\tau'',\iota)}{\alpha} \leq d(\tau'', \iota) \leq d(\tau', \iota) \leq \beta d_{bp}(\tau', \iota)$. Assim, $d(\tau', \iota)$ está a um fator $\alpha\beta$ de $d(\tau'', \iota)$. □

Ainda que as variações de ψ a serem exploradas neste trabalho sejam inv, ldes e entr, uma outra variação de ψ pode ser utilizada para clarificar a abordagem de aproximação. Seja π uma permutação de um conjunto S de n elementos. Seja $\psi = d_{bp}$. Então, $A = \{\tau \in S_n \mid d_{bp}(\pi, \tau) \leq k\}$, para algum limiar k. Assim, para todo $\tau \in A$, $bp(\tau^{-1} \circ \pi) \leq k$. Note que se $bp(\pi) \leq k$, então $\iota \in A$ e a distância se reduz trivialmente a zero. Se $bp(\pi) > k$, então a permutação em A com menos breakpoints possui $bp(\pi) - k$ breakpoints, pois k breakpoints em π são adjacências em A. Logo, seguindo o resultado do Teorema 3.3, $bp(\pi) - k$ é uma $\alpha\beta$-aproximação quando $\psi = d_{bp}$. De fato, se a distância considerada é a distância de reversões, pela Equação (1), $bp(\pi) - k$ é uma 2-aproximação para a distância (elemento-conjunto) de reversões entre π e $A = \{\tau \in S_n \mid d_{bp}(\pi, \tau) \leq k\}$.

3.3 Ordenações Semi-Completas Limitadas por Inversões

A primeira variação apresentada do Problema 3 utiliza inversões. Nesse problema, a função de comparação utilizada para definir o conjunto A é inv. O problema está definido no Problema 4.

Problema 4 (Ordenações Semi-Completas por Operações de Rearranjos de Genomas Limitadas por Inversões)**.**
Entrada: Permutação π de um conjunto S com n elementos e limiar k.
Objetivo: Encontrar uma sequência transformadora mínima entre um conjunto $A = \{\sigma \in S_n \mid \text{inv}(\pi, \sigma) \leq k\}$ e ι.

Utilizando-se da abordagem de aproximação detalhada na seção anterior, um caminho possível para obter uma aproximação para o problema é listar todos os elementos do conjunto A e, então, determinar o elemento com menor número de breakpoints em relação a ι. Contudo, a cardinalidade do conjunto A pode explodir combinatoriamente com facilidade, como mostrado nas seções 3.3.1 e 3.6. Dessa forma, é conveniente empregar métodos alternativos

para determinar $\min\{d_{bp}(\sigma, \iota) \mid \sigma \in A\}$. Caso exista um método polinomial no tamanho da permutação π para determinar essa distância de breakpoints mínima, o Teorema 3.3 permite encontrar uma aproximação para a distância em tempo polinomial.

Em alguns casos, o elemento do conjunto A que possui menos breakpoints é a própria permutação π. Nesse caso, a abordagem aproximada é entre a própria permutação π e ι, o que torna o problema equivalente ao Problema 1. O Teorema 3.4 estabelece situações em que a permutação π é a permutação de A com menos breakpoints.

Teorema 3.4. *Seja π uma permutação de um conjunto S com n elementos. Seja $[i, j]$ uma strip de π de tamanho b de forma que $|i - adj\_pos(i)| = \ell$ e $|j - adj\_pos(j)| = p$. Então, não existe uma sequência de k inversões que resulta em uma permutação com menos breakpoints que π se*

$$k < \min\left\{\max\left\{\min\left\{\ell, p\right\}, b\right\}, \max\left\{l + p, \frac{b}{2}\right\}\right\}.$$

Demonstração. Seja π uma permutação de um conjunto S com n elementos. Seja $[i, j]$ uma strip de π de tamanho b de forma que $|i - adj\_pos(i)| = \ell$ e $|j - adj\_pos(j)| = p$. Seja $k \in \mathbb{Z}$ tal que $k < \min\left\{\max\left\{\min\left\{\ell, p\right\}, b\right\}, \max\left\{l + p, \frac{b}{2}\right\}\right\}$. Considera-se dois casos.

Caso 1: *a strip $[i, j]$ é movida sem ser quebrada.* Suponha, para obter uma contradição, que existe uma sequência de k inversões que move a strip $[i, j]$ sem quebrá-la e resulta em uma permutação com menos breakpoints que π. Para mover a strip e formar uma adjacência são necessárias ℓ inversões, se a strip for concatenada com a strip contendo o elemento em $adj\_pos(i)$ ou p inversões, se a strip for concatenada com a strip contendo o elemento em $adj\_pos(j)$. Se a strip for concatenada com a strip contendo o elemento em $adj\_pos(i)$, um elemento entre as posições $adj\_pos(i)$ e i de π deve participar de ao menos b inversões para ser movido para depois da strip $[i, j]$. Da mesma forma, se $[i, j]$ for concatenada com a strip contenod o elemento em $adj\_pos(j)$, um elemento entre as posições j e $adj\_pos(j)$ deve participar de ao menos b inversões. Portanto, ao menos $\max\left\{\min\left\{\ell, p\right\}, b\right\}$ inversões são necessárias para resultar em uma permutação com menos breakpoints que π. Como $k < \max\left\{\min\left\{\ell, p\right\}, b\right\}$, há uma contradição e não existe uma sequência de k inversões que move a strip $[i, j]$ sem quebrá-la e resulta em uma permutação com menos breakpoints que π.

Caso 2: *a strip $[i, j]$ é quebrada.* Suponha, para obter uma contradição, que existe uma sequência de k inversões que quebra a strip $[i, j]$ e resulta em uma permutação com menos breakpoints que π. Pelo Corolário 1.5 é suficiente considerar que a strip foi quebrada uma única vez. Note que uma quebra de uma strip introduz um breakpoint e, para que o número de breakpoints da permutação diminua, é necessário que as duas partes resultantes da quebra se tornem partes de strips maiores, removendo, assim, dois dos breakpoints de π e reduzindo o número total de breakpoints em um. Como a strip é quebrada uma única vez, ela dá origem a duas strips. Cada uma dessas strips deve ser movida para os elementos adjacentes aos extremos π_i e π_j e, portanto, ao menos $\ell + p$ inversões devem ser realizadas. Além disso, como a strip tem tamanho b e foi dividida em duas partes, ao menos uma das partes deve ter tamanho maior ou igual a $\frac{b}{2}$. Se a nova strip com tamanho ao menos $\frac{b}{2}$ for a strip iniciada em π_i, então um elemento entre as posições $adj\_pos(i)$ e i de π deve

participar de ao menos $\frac{b}{2}$ inversões. Da mesma forma, se a nova strip com tamanho ao menos $\frac{b}{2}$ for a strip terminada em π_j, então um elemento entre as posições j e $adj\_pos(j)$ deve participar de ao menos $\frac{b}{2}$ inversões. Assim, ao menos $\max\{\ell + p, \frac{b}{2}\}$ são necessárias para resultar em uma permutação com menos breakpoints que π. Como $k < \max\left\{\ell + p, \frac{b}{2}\right\}$, há uma contradição e não existe uma sequência de k inversões que separa a strip $[i, j]$ e resulta em uma permutação com menos breakpoints que π. □

3.3.1 Cálculo da Cardinalidade do Conjunto A

Embora, dos problemas apresentados, a inversão seja o mais simples, não se conhece uma fórmula fechada para determinar o número de permutações σ cujo número de inversões em relação a uma permutação π é menor ou igual a k. Esse problema é um problema clássico em Combinatória e está associado aos números mahonianos. Os *números mahonianos* $M(n, s)$ representam a quantidade de permutações de n elementos que possuem exatamente s inversões. Esse nome foi dado em homenagem a Percy Alexander MacMahon. Em 1913, MacMahon [9] demonstrou que o número de permutações de n elementos com exatamente k inversões é igual ao número de permutações π de n elementos com $\sum_{\pi_i > \pi_{i+1}} i = k$. Algumas formas conhecidas de se calcular os números mahonianos utilizam funções geradoras. Abaixo, um método que utiliza programação dinâmica para calcular a cardinalidade de $A = \{\sigma \in S \mid \text{inv}(\pi, \sigma) \leq k\}$ está apresentado.

Seja $\tau = \sigma^{-1} \circ \pi$. Então, uma inversão entre π e σ é um par (π_i, π_j), $i < j$ em que $\sigma^{-1}_{\pi_i} > \sigma^{-1}_{\pi_j}$, ou seja, em que $\tau_i > \tau_j$. Dessa forma, $\text{inv}(\pi, \sigma) = \text{inv}(\tau, \iota) = \text{inv}(\tau)$. Para todo $i \in S$, define-se $c_i = |\{j \mid j > i \text{ e } \tau_i > \tau_j\}|$ como o número de inversões que o elemento na posição i participa. Por consequência imediata da definição, $0 \leq c_i \leq n - i$. O número total de inversões de τ é dado por $\text{inv}(\tau) = \sum_{i=1}^{n} c_i$. O objetivo torna-se, então, contar o número de sequências $(c_1, c_2, \ldots, c_n)$ tais que $0 \leq c_i \leq n - i$ e $\sum_{i=1}^{n} c_i \leq k$.

Define-se a função $f(i, s)$ como o número de sequências $(c_1, c_2, \ldots, c_i)$ que satisfazem $0 \leq c_j \leq n - j$ para $1 \leq j \leq i$ e $\sum_{j=1}^{i} c_j = s$. O caso base da programação dinâmica é dado por $f(0, s) = 1$, para $0 \leq s \leq k$. Para $i \geq 1$ e $s \geq 0$,

$$f(i, s) = \sum_{\substack{c_i = 0 \\ s - c_i \geq 0}}^{n-i} f(i - 1, s - c_i).$$

O número total de permutações com número de inversões menor ou igual a k é dado por

$$\sum_{s=0}^{k} f(n, s).$$

3.4 Ordenações Semi-Completas Limitadas por λ-Permutações

Uma outra opção para a função de comparação é ldes. Nesse caso, o conjunto A é o conjunto de todas as λ-permutações quando $\lambda = k$. O problema está definido no Problema 5.

Problema 5 (Ordenações Semi-Completas por Operações de Rearranjos de Genomas Limitadas por λ-Permutações)**.**
ENTRADA: Permutação π de um conjunto S com n elementos e limiar k.
OBJETIVO: Encontrar uma sequência transformadora mínima entre um conjunto $A = \{\sigma \in S_n \mid \text{lides}(\pi, \sigma) < k\}$ e ι.

Da mesma forma que no caso das inversões, é possível obter uma aproximação para o problema listando todos os elementos do conjunto A e determinando o elemento com menor número de breakpoints em relação a ι. Caso exista um método polinomial no tamanho da permutação π para determinar a distância de breakpoints mínima entre um elemento de A e ι, o Teorema 3.3 permite encontrar uma aproximação para o problema em tempo polinomial. Também da mesma forma que no caso das inversões, em alguns casos π pode ser a permutação que minimiza a distância de breakpoints entre um elemento de A e ι e, nesse caso, o problema se reduz ao Problema 1. O Teorema 3.5 estabelece situações em que a permutação π é a permutação de A com menos breakpoints.

Teorema 3.5. *Seja π uma permutação de um conjunto S com n elementos. Seja $[i, j]$ uma strip de π de tamanho b de forma que $|i - adj\_pos(i)| = \ell$ e $|j - adj\_pos(j)| = p$. Então, não existe uma λ-permutação de π com menos breakpoints que π se*

$$\lambda \leq \min\left\{\max\left\{\min\left\{\frac{\ell}{2}, \frac{p}{2}\right\}, \frac{b}{2}\right\}, \max\left\{\frac{l}{2}, \frac{p}{2}, \frac{b}{4}\right\}\right\}.$$

Demonstração. Seja π uma permutação de um conjunto S com n elementos. Seja $[i, j]$ uma strip de π de tamanho b de forma que $|i - adj\_pos(i)| = \ell$ e $|j - adj\_pos(j)| = p$. Seja $\lambda \in \mathbb{Z}$ tal que $\lambda \leq \min\left\{\max\left\{\min\left\{\frac{\ell}{2}, \frac{p}{2}\right\}, \frac{b}{2}\right\}, \max\left\{\frac{l}{2}, \frac{p}{2}, \frac{b}{4}\right\}\right\}$. Considera-se dois casos.

Caso 1: *a strip $[i, j]$ é movida sem ser quebrada.* Suponha, para obter uma contradição, que existe uma sequência de deslocamentos que move a strip $[i, j]$ sem quebrá-la e resulta em uma λ-permutação de π com menos breakpoints que π. Para remover um breakpoint é necessário concatenar a strip $[i, j]$ com a strip do elemento em $adj\_pos(i)$ ou com a strip do elemento $adj\_pos(j)$. Para concatenar a strip com a strip do elemento em $adj\_pos(i)$ é necessário que a strip $[i, j]$ ou a strip do elemento em $adj\_pos(i)$ se desloque pelo menos $\ell/2$ posições. Para concatenar a strip com a strip do elemento em $adj\_pos(j)$ é necessário que a strip $[i, j]$ ou a strip do elemento em $adj\_pos(j)$ se desloque pelo menos $p/2$ posições. Além disso, caso a strip seja concatenada com a strip contendo $adj\_pos(i)$, um elemento entre as posições adj_pos e i deve se deslocar pelo menos $b/2$ posições. Algo similar ocorre na concatenação entre $[i, j]$ e a strip com o elemento em $adj\_pos(j)$. Portanto, deslocamentos de no mínimo $\max\left\{\min\left\{\frac{\ell}{2}, \frac{p}{2}\right\}, \frac{b}{2}\right\}$ posições devem ser pertmitidos, o que contradiz o fato de que $\lambda \leq \max\left\{\min\left\{\frac{\ell}{2}, \frac{p}{2}\right\}, \frac{b}{2}\right\}$. Portanto, não existe uma uma sequência de deslocamentos que move a strip $[i, j]$ sem quebrá-la e resulta em uma λ-permutação de π com menos breakpoints que π.

Caso 2: *a strip $[i, j]$ é quebrada.* Suponha, para obter uma contradição, que existe uma sequência de deslocamentos que quebra a strip $[i, j]$ e resulta em uma λ-permutação de π com menos breakpoints que π. Pelo Corolário 1.5 é suficiente considerar que a strip foi quebrada uma única vez. Note que uma quebra de uma strip introduz um breakpoint e, para que o

número de breakpoints da permutação diminua, é necessário que as duas partes resultantes da quebra se tornem partes de strips maiores, removendo assim dois dos breakpoints de π e reduzindo o número total de breakpoints em um. Como a strip é quebrada uma única vez, ela dá origem a duas strips. Cada uma dessas strips deve ser movida para os elementos adjacentes aos extremos π_i e π_j. Para mover a substrip que contém o elemento π_i para concatená-la com a strip do elemento em $adj\_pos(i)$ é necessário que a substrip de π_i ou a strip do elemento em $adj\_pos(i)$ se desloque pelo menos $\ell/2$ posições. Para mover a substrip que contém o elemento π_j para concatená-la com a strip do elemento $adj\_pos(j)$ é necessário que a substrip de π_j ou a strip do elemento em $adj\_pos(j)$ se desloque pelo menos $p/2$ posições. Além disso, como a strip tem tamanho b e foi dividida em duas partes, ao menos uma das partes deve ter tamanho maior ou igual a $\frac{b}{2}$. Se a nova strip com tamanho ao menos $\frac{b}{2}$ for a strip iniciada em π_i, então um elemento entre as posições $adj\_pos(i)$ e i de π deve se deslocar ao menos $\frac{b}{4}$ posições para que ele apareça na λ-permutação após a strip concatenada. Da mesma forma, se a nova strip com tamanho ao menos $\frac{b}{2}$ for a strip terminada em π_j, então um elemento entre as posições j e $adj\_pos(j)$ deve se deslocar $b/2$ posições para que ele apareça na λ-permutação antes da strip concatenada. Assim, deslocamentos de tamanho ao menos $\max\left\{\frac{l}{2}, \frac{p}{2}, \frac{b}{4}\right\}$ devem ser permitidos, o que contradiz o fato de que $\lambda \leq \max\left\{\frac{l}{2}, \frac{p}{2}, \frac{b}{4}\right\}$. Portanto, não existe uma uma sequência de deslocamentos que quebra a strip $[i, j]$ e resulta em uma λ-permutação de π com menos breakpoints que π. □

Além disso, nos casos em que $\iota \in A$, a distância se reduz trivialmente a zero.

3.4.1 Cálculo da Cardinalidade do Conjunto *A*

Dada uma permutação π de um conjunto S de n elementos, a cardinalidade do conjunto $A = \{\sigma \mid \sigma \text{ é uma } \lambda\text{-permutação de } \pi\}$ varia de forma complexa com fatores como n e λ. Apresenta-se um método que, utilizando-se de programação dinâmica, determina a cardinalidade desse conjunto. Para cada posição $p \in S$, define-se o conjunto E_p de *elementos permitidos* como

$$E_p = \{e \in S \mid |\pi_i^{-1} - p| < \lambda\}.$$

Define-se uma função $f(p, U)$ que representa o número de formas de atribuir elementos às posições de p a n, dado que os elementos já utilizados nas primeiras $p-1$ posições estão em $U \subseteq S$. O caso base da programação dinâmica é $f(n+1, S) = 1$. Para $p \in \{n, n-1, \ldots, 1\}$, calcula-se

$$f(p, U) = \sum_{e \in E_p \setminus U} f(p+1, U \cup \{e\}),$$

em que $E_p \setminus U$ é o conjunto de elementos permitidos na posição p que ainda não foram utilizados. Então, o número total de λ-permutações de π é dado por $f(1, \varnothing)$, ou seja, as formas de atribuir elementos às posições de 1 a n de forma que nenhum elemento já tenha sido utilizado.

3.5 Ordenações Semi-Completas Limitadas por Entropia

A terceira variação apresentada do Problema 3 utiliza a entropia como critério de proximidade; isto é, a função de comparação utilizada para definir o conjunto A é entr. O problema está definido no Problema 6.

Problema 6 (Ordenações Semi-Completas por Operações de Rearranjos de Genomas Limitadas por Entropia)**.**
ENTRADA: Permutação π de um conjunto S com n elementos e limiar k.
OBJETIVO: Encontrar uma sequência transformadora mínima entre um conjunto $A = \{\sigma \in S_n \mid \text{ldes}(\pi, \sigma) \leq k\}$ e ι.

De forma similar às outras variações apresentadas, é possível obter uma aproximação para o problema desde que se saiba a menor distância de breakpoints entre um elemento de A e ι. Caso exista um método polinomial no tamanho da permutação π para determinar a distância de breakpoints mínima entre um elemento de A e ι, essa aproximação pode, portanto, ser feita em tempo polinomial. Em alguns casos, a permutação π pode ser uma permutação que, dentro dos elementos de A, tem a menor distância de breakpoints para ι. Nesse caso, o Problema 6 se reduz ao Problema 1. O Teorema 3.6 estabelece situações em que a permutação π é a permutação de A com menos breakpoints.

Teorema 3.6. *Seja π uma permutação de um conjunto S com n elementos. Seja $[i, j]$ uma strip de π de tamanho b de forma que $|i - adj\_pos(i)| = \ell$ e $|j - adj\_pos(j)| = p$. Então, não existe uma sequência de deslocamentos de elementos de π que resulta em uma permutação σ com menos breakpoints que π de forma que $\sum_{i=1}^{n} |\sigma_i^{-1} - \pi_i^{-1}| = k$ se*

$$k < \min\left\{\max\left\{\min\left\{\ell, p\right\}, b\right\}, \max\left\{l + p, \frac{b}{2}\right\}\right\}.$$

Demonstração. Seja π uma permutação de um conjunto S com n elementos. Seja $[i, j]$ uma strip de π de tamanho b de forma que $|i - adj\_pos(i)| = \ell$ e $|j - adj\_pos(j)| = p$. Seja $k \in \mathbb{Z}$ tal que $k < \min\left\{\max\left\{\min\left\{\ell, p\right\}, b\right\}, \max\left\{l + p, \frac{b}{2}\right\}\right\}$. Considera-se dois casos.

Caso 1: *a strip $[i, j]$ é movida sem ser quebrada.* Suponha, para obter uma contradição, que existe uma sequência de deslocamentos de elementos da permutação π que move a strip $[i, j]$ sem quebrá-la e resulta em uma permutação σ tal que $\sum_{i=1}^{n} |\sigma_i^{-1} - \pi_i^{-1}| = k$ com menos breakpoints que π. Para mover a strip e formar uma adjacência é necessário que cada elemento da strip se desloque ℓ posições, se a strip for concatenada com a strip contendo o elemento em $adj\_pos(i)$ ou p posições, se a strip for concatenada com a strip contendo o elemento em $adj\_pos(j)$. Se a strip for concatenada com a strip contendo o elemento em $adj\_pos(i)$, um elemento entre as posições $adj\_pos(i)$ e i de π deve se deslocar ao menos b posições para ser movido para depois da strip $[i, j]$. Da mesma forma, se $[i, j]$ for concatenada com a strip contendo o elemento em $adj\_pos(j)$, um elemento entre as posições j e $adj\_pos(j)$ deve se deslocar ao menos b posições. Portanto, a soma das posições deslocadas é maior ou igual a $\max\{\min\{\ell, p\}, b\}$ para que o resultado seja uma permutação com menos breakpoints que π. Como $k < \max\{\min\{\ell, p\}, b\}$, há uma contradição e não existe uma sequência de deslocamentos de elementos da permutação π que move a strip $[i, j]$

sem quebrá-la e resulta em uma permutação σ tal que $\sum_{i=1}^{n} |\sigma_i^{-1} - \pi_i^{-1}| = k$ com menos breakpoints que π

Caso 2: *a strip* $[i, j]$ *é quebrada.* Suponha, para obter uma contradição, que existe uma sequência de deslocamentos de elementos da permutação π que quebra a strip $[i, j]$ e resulta em uma permutação σ tal que $\sum_{i=1}^{n} |\sigma_i^{-1} - \pi_i^{-1}| = k$ com menos breakpoints que π. Pelo Corolário 1.5 é suficiente considerar que a strip foi quebrada uma única vez. Note que uma quebra de uma strip introduz um breakpoint e, para que o número de breakpoints da permutação diminua, é necessário que as duas partes resultantes da quebra se tornem partes de strips maiores, removendo assim dois dos breakpoints de π e reduzindo em um o número total de breakpoints. Como a strip é quebrada uma única vez, ela dá origem a duas strips. Cada uma dessas strips deve ser movida para os elementos adjacentes aos extremos π_i e π_j e, portanto, deslocamentos de $\ell + p$ posições devem ser realizados. Além disso, como a strip tem tamanho b e foi dividida em duas partes, ao menos uma das partes deve ter tamanho maior ou igual a $\frac{b}{2}$. Se a nova strip com tamanho ao menos $\frac{b}{2}$ for a strip iniciada em π_i, então um elemento entre as posições $adj\_pos(i)$ e i de π deve se deslocar ao menos $\frac{b}{2}$ posições. Da mesma forma, se a nova strip com tamanho ao menos $\frac{b}{2}$ for a strip terminada em π_j, então um elemento entre as posições j e $adj\_pos(j)$ deve se deslocar ao menos $\frac{b}{2}$ posições. Assim, deslocamentos de ao menos $\max\{\ell + p, \frac{b}{2}\}$ posições são necessários para resultar em uma permutação com menos breakpoints que π. Como $k < \max\left\{l + p, \frac{b}{2}\right\}$, há uma contradição e não existe uma sequência de deslocamentos de elementos da permutação π que quebra a strip $[i, j]$ e resulta em uma permutação σ tal que $\sum_{i=1}^{n} |\sigma_i^{-1} - \pi_i^{-1}| = k$ com menos breakpoints que π. □

Notavelmente, a distância se torna zero se $\text{entr}(\pi, \iota) \leq k$. Uma forma de encontrar a permutação com a menor distância de breakpoints em relação a ι dentre os elementos de A é listar todos esses elementos, o que pode ser possível para valores pequenos de n e k, mas torna-se impraticável conforme esses valores crescem.

3.5.1 Cálculo da Cardinalidade do Conjunto *A*

Dada uma permutação π de um conjunto S de n elementos, a cardinalidade do conjunto $A = \{\sigma \mid \sigma \text{ é uma permutação de } S \text{ com entropia em relação a } \pi \text{ menor ou igual a } k\}$ varia de forma complexa com fatores como n e k. Um método que calcula essa cardinalidade, baseado em programação dinâmica, está apresentado abaixo.

Define-se, para cada elemento $e \in S$ e cada posição $p \in S$, o custo de atribuir e a uma posição p como $c(e, p) = |\pi_e^{-1} - p|$. Define-se, também, uma função $f(p, U, d)$ que representa o número de formas de atribuir elementos às posições de p a n, dado que o conjunto de elementos já utilizados nas primeiras $p - 1$ posições é $U \subseteq S$ e a entropia total acumulada é d. O caso base, quando todas as posições já foram preenchidas, é

$$f(n + 1, S, d) = \begin{cases} 1, & \text{se } d \leq k; \\ 0, & \text{se } d > k. \end{cases}$$

Para $p \in \{1, 2, \ldots, n\}$ calcula-se

$$f(p, U, d) = \sum_{\substack{e \in S \setminus U \\ d + c(e,p) \leq k}} f(p+1, U \cup \{e\}, d + c(e, p)).$$

O número total de permutações de π com entropia de no máximo k em relação a π é dado por $f(1, \varnothing, 0)$, ou seja, as formas de atribuir elementos às posições de 1 a n de forma que nenhum elemento já tenha sido utilizado e a entropia acumulada no início é zero.

3.6 Comparação das Cardinalidades em Função dos Limiares

Embora o algoritmo de programação dinâmica para determinar a cardinalidade do conjunto A no caso das inversões seja polinomial em n (considerando $k < n^2$), as demais formulações não o são. Além disso, nota-se que mesmo que nos casos da entropia e das λ-permutações, em que a permutação π é utilizada para definir os conjuntos utilizados para o cálculo, os valores dependem apenas de n e k. A Tabela 1 apresenta os valores para a cardinalidade de A em cada um dos problemas para valores pequenos de n e de k. Essa tabela foi construída utilizando-se uma implementação em Python[1] dos algoritmos de programação dinâmica apresentados.

[1] Implementação disponível em `https://github.com/jpvianini/cardinality_count`.

n	k	**Inversões**	**λ-permutações**	**Entropia**
5	1	5	1	1
	2	14	8	5
	4	49	78	17
	6	91	120	41
	8	115	120	76
	10	120	120	100
	20	120	120	120
	45	120	120	120
10	1	10	1	1
	2	54	89	10
	4	649	19708	62
	6	4015	329462	286
	8	16599	1865520	1076
	10	51909	3628800	3426
	20	1319957	3628800	184968
	45	3628800	3628800	3485952
15	1	15	1	1
	2	119	987	15
	4	2924	5284109	132
	6	34900	615260976	856
	8	266338	12764590275	4501
	10	1487262	119892387720	20127
	20	546874905	1307674368000	6842736
	45	323682655417	1307674368000	14750025280

Tabela 1: Comparação das cardinalidades do conjunto A em função dos limiares, do tamanho da permutação e da variante do problema.

Analisando-se a Tabela 1 é possível perceber que a quantidade de λ-permutações cresce mais rapidamente que as outras variações, atingindo a totalidade das permutações quando $k = n$. As inversões crescem rapidamente também, atingindo a totalidade das permutações quando $k = \binom{n}{2}$. A entropia cresce mais lentamente com os valores de k.

4 Heurística para a Abordagem de Aproximação

Combinar strips é uma forma de se remover breakpoints. Considere as seguintes permutações do conjunto $S = \{1, 2, 3, 4, 5, 6, 7, 8\}$:

i. $\begin{pmatrix} 1 & 2 & 7 & 8 & 3 & 4 & 5 & 6 \end{pmatrix}$;

ii. $\begin{pmatrix} 1 & 2 & 5 & 3 & 4 & 8 & 7 & 6 \end{pmatrix}$; e

iii. $\begin{pmatrix} 1 & 2 & 5 & 6 & 3 & 4 & 7 & 8 \end{pmatrix}$.

Combinar as strips $\begin{pmatrix} 1 & 2 \end{pmatrix}$ e $\begin{pmatrix} 3 & 4 \end{pmatrix}$ reduz uma quantidade diferente de breakpoints em cada uma das permutações acima. Na permutação i, um breakpoint pode ser removido, resultando na permutação $\begin{pmatrix} 1 & 2 & 3 & 4 & 7 & 8 & 5 & 6 \end{pmatrix}$. Na permutação ii, dois breakpoints podem ser removidos, o que resulta na permutação $\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 8 & 7 & 6 \end{pmatrix}$. Já na permutação iii, até três breakpoints podem ser removidos, o que resulta na permutação identidade $\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{pmatrix}$.

De fato, ao se combinar strips de uma permutação, 1, 2 ou 3 dos breakpoints da permutação podem ser removidos. Embora não necessariamente encontre a permutação com menos breakpoints do conjunto A dos Problemas 4, 5 e 6, uma ideia para encontrar uma strip com menos breakpoints que π está descrita na Heurística 4.1.

Heurística 4.1 (Heurística Gulosa para Concatenação de Strips)**.** Para encontrar uma permutação com menos breakpoints que a permutação π, combinar strips de π de acordo com o seguinte critério. A cada passo, para gerar uma permutação π', deve ser escolhida a combinação de strips que maximize o valor $\frac{\Delta bp}{\Delta \psi}$, onde Δbp é número de breakpoints removidos e $\Delta\psi$ é 1, se o valor de $\psi(\pi, \pi')$ diminuiu ou a variação do valor de $\psi(\pi, \pi')$, se o valor de $\psi(\pi, \pi')$ aumentou.

Como exemplo, segue o funcionamento da heurística para o caso do Problema 4 em que a entrada consiste da permutação $\pi = \begin{pmatrix} 6 & 4 & 1 & 3 & 7 & 2 & 5 \end{pmatrix}$ e do limiar $k = 3$. As strips são todos os elementos individuais, já que todos os pontos da permutação são breakpoints. A Tabela 2 mostra quantos breakpoints podem ser removidos a partir de todas as concatenações das strips de π para a primeira inversão.

Strip	Concatenar com	Inversões Necessárias	Breakpoints removidos	$\Delta bp/\Delta\psi$
$\begin{pmatrix} 0 \end{pmatrix}$	$\begin{pmatrix} 1 \end{pmatrix}$	2	1	0.5
$\begin{pmatrix} 1 \end{pmatrix}$	$\begin{pmatrix} 2 \end{pmatrix}$	2	2	1
$\begin{pmatrix} 2 \end{pmatrix}$	$\begin{pmatrix} 3 \end{pmatrix}$	2	2	1
$\begin{pmatrix} 3 \end{pmatrix}$	$\begin{pmatrix} 4 \end{pmatrix}$	2	1	0.5
$\begin{pmatrix} 4 \end{pmatrix}$	$\begin{pmatrix} 5 \end{pmatrix}$	4	1	0.25
$\begin{pmatrix} 5 \end{pmatrix}$	$\begin{pmatrix} 6 \end{pmatrix}$	6	1	0.17
$\begin{pmatrix} 6 \end{pmatrix}$	$\begin{pmatrix} 7 \end{pmatrix}$	3	1	0.33
$\begin{pmatrix} 7 \end{pmatrix}$	$\begin{pmatrix} 8 \end{pmatrix}$	2	1	0.5

Tabela 2: Concatenações possíveis de strip para a primeira inversão

Nota-se que as concatenações que maximizam $\Delta bp/\Delta\psi$ são:

- concatenação da strip (1) com a strip (2), invertendo-se os elementos 7 e 2 e depois os elementos 3 e 2, o que resulta na permutação $(6 \quad 4 \quad 1 \quad 2 \quad 3 \quad 7 \quad 5)$ (dois breakpoints a menos); e
- concatenação da strip (2) com a strip (3), invertendo-se os elementos 7 e 2 e depois os elementos 3 e 2, o que resulta na permutação $(6 \quad 4 \quad 1 \quad 2 \quad 3 \quad 7 \quad 5)$ (dois breakpoints a menos).

Ou seja, o melhor caso guloso é o que junta as strips (1), (2) e (3), usa duas inversões e remove dois breakpoints e resulta na permutação $(6 \quad 4 \quad 1 \quad 2 \quad 3 \quad 7 \quad 5)$. A única concatenação de strips possível com exatamente uma inversão é a concatenação da strip (7) com a strip (8), o que reduz mais um breakpoint. Logo, a heurística devolve a permutação $(6 \quad 4 \quad 1 \quad 2 \quad 3 \quad 5 \quad 7)$, que tem três breakpoints a menos que π.

Não há garantias que a permutação devolvida pela heurística tem a menor distância de breakpoints para ι dentre todas as permutações do conjunto A, uma vez que ela preserva as strips existentes em π e existem casos em que quebrar uma strip pode fornecer a permutação que minimiza a distância de breakpoints para ι dentro do conjunto A. Ainda assim, a permutação devolvida pela heurística pode dar uma aproximação mais justa que, por exemplo, utilizar o número de breakpoints de π.

5 Considerações Finais

Neste trabalho, o problema das ordenações semi-completas por operações de rearranjos de genomas, uma extensão do problema clássico da distância de rearranjos, foi abordado. Três variantes do problema, baseadas em diferentes métricas, foram apresentadas: a variante das inversões, a variante das λ-permutações e a variante da entropia.

A Heurística 4.1, proposta para reduzir o número de breakpoints em permutações dentro de conjuntos delimitados por restrições específicas, pode mostrar-se uma ferramenta eficaz na aproximação de soluções para os problemas considerados, uma vez que não foi encontrado um método polinomial para determinar a permutação que minimiza o número de breakpoints dentre todas em algum desses conjuntos.

A análise detalhada das cardinalidades dos conjuntos associados às três variantes revelou diferenças significativas em suas complexidades, sendo as λ-permutações as mais restritivas para valores baixos de k, enquanto a métrica baseada em entropia apresenta crescimento mais lento.

Apesar das contribuições deste trabalho, desafios permanecem. A busca por algoritmos mais eficientes para calcular a distância de rearranjos em espaços de aproximação pode se mostrar um campo promissor e ter aplicações na biologia, como, por exemplo, calcular distâncias evolutivas entre um indivíduo e conjuntos de indivíduos.

O problema apresentado neste trabalho pode ajudar a modelar diversos problemas em genômica comparativa, com aplicações potenciais em áreas como filogenética e evolução. Trabalhos futuros podem explorar heurísticas mais eficientes ou generalizações para genomas com múltiplos cromossomos.

Referências

[1] V. Bafna e P. A. Pevzner. "Sorting by Transpositions". Em: *SIAM Journal on Discrete Mathematics* 11.2 (1998), pp. 224–240. DOI: 10.1137/S089548019528280X.

[2] P. Berman, S. Hannenhalli e M. Karpinski. "1.375-Approximation Algorithm for Sorting by Reversals". Em: *Proceedings of the 10th Annual European Symposium on Algorithms (ESA'2002)*. Berlin, Heidelberg: Springer-Verlag, 2002, pp. 200–210. ISBN: 3-540-44180-8.

[3] L. Bulteau, G. Fertin e I. Rusu. "Sorting by Transpositions Is Difficult". Em: *SIAM Journal on Discrete Mathematics* 26.3 (2012), pp. 1148–1180. DOI: 10.1137/110851390.

[4] A. Caprara. "Sorting Permutations by Reversals and Eulerian Cycle Decompositions". Em: *SIAM Journal on Discrete Mathematics* 12.1 (1999), pp. 91–110. DOI: 10.1137/S089548019731994X.

[5] I. Elias e T. Hartman. "A 1.375-Approximation Algorithm for Sorting by Transpositions". Em: *Algorithms in Bioinformatics*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 204–215.

[6] G. Fertin, A. Labarre, I. Rusu, E. Tannier e S. Vialette. *Combinatorics of Genome Rearrangements*. 1ª ed. Computational Molecular Biology. London, England: The MIT Press, 2009. ISBN: 9780262258753. DOI: 10.7551/mitpress/9780262062824.001.0001.

[7] J. Kececioglu e D. Sankoff. "Exact and approximation algorithms for the inversion distance between two chromosomes". Em: *Combinatorial Pattern Matching*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1993, pp. 87–105. ISBN: 978-3-540-47732-7.

[8] R. Lowen. *Approach Spaces: The Missing Link in the Topology—Uniformity—Metric Triad*. Oxford University Press, 1997.

[9] P. A. MacMahon. "The Indices of Permutations and the Derivation Therefrom of Functions of a Single Variable Associated with the Permutations of any Assemblage of Objects". Em: *American Journal of Mathematics* 35.3 (1913), pp. 281–322.

[10] A. R. Oliveira, K. L. Brito, A. O. Alexandrino, G. Siqueira, U. Dias e Z. Dias. "Rearrangement Distance Problems: An updated survey". Em: *ACM Computing Surveys* 56.8 (2024). ISSN: 0360-0300. DOI: 10.1145/3653295.

[11] G.A. Watterson, W.J. Ewens, T.E. Hall e A. Morgan. "The chromosome inversion problem". Em: *Journal of Theoretical Biology* 99.1 (1982), pp. 1–7. ISSN: 0022-5193. DOI: 10.1016/0022-5193(82)90384-8.