



Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Ordenação por Transposições baseado no Formalismo Algébrico

Héderson Pereira dos Santos

Dissertação apresentada como requisito parcial
para conclusão do Mestrado em Informática

Orientadora
Prof.^a Dr.^a Maria Emília Machado Telles Walter

Brasília
2006

Universidade de Brasília – UnB
Instituto de Ciências Exatas
Departamento de Ciência da Computação
Mestrado em Informática

Coordenador: Prof.^a Dr.^a Alba Cristina M. A. de Melo

Banca examinadora composta por:

Prof.^a Dr.^a Maria Emília Machado Telles Walter (Orientadora) – CIC/UnB
Prof. Dr. João Meidanis – IC/Unicamp
Prof. Dr. Mauricio Ayala Rincón – Mat/UnB

CIP – Catalogação Internacional na Publicação

Héderson Pereira dos Santos.

Ordenação por Transposições baseado no Formalismo Algébrico/
Héderson Pereira dos Santos. Brasília : UnB, 2006.
81 p. : il. ; 29,5 cm.

Tese (Mestre) – Universidade de Brasília, Brasília, 2006.

1. Problema da Ordenação por Transposições, 2. Formalismo
Algébrico, 3. Rearranjo de Genomas, 4. Biologia Computacional

CDU 004

Endereço: Universidade de Brasília
Campus Universitário Darcy Ribeiro – Asa Norte
CEP 70910-900
Brasília – DF – Brasil

Agradecimentos

Para a elaboração desse trabalho contei com a ajuda e colaboração de várias pessoas, que de coração agradeço:

- Minha família, em especial aos meus pais, **Ivanildo** e **Autelina**, pelo carinho que sempre tiveram comigo e que mesmo de longe estiveram presentes dando apoio e incentivo. Também agradeço aos meus irmãos **Kezia**, **Jailson** e **Maria**.
- Professora **Maria Emília** por ser uma ótima orientadora, sempre presente nos momentos difíceis do trabalho, pela paciência, apoio, incentivo nos estudos e boa vontade para me explicar a teoria.
- Meus colegas de trabalho **Viviane**, **Cláudia**, **Luiz** e **Marcos** por terem permitido eu conciliar estudos e trabalho durante estes dois últimos anos. Também ao **Maciel** por ter me ajudado com as cargas dos arquivos de permutações no banco de dados.
- Meus colegas de mestrado **Eluzai**, **Jorge**, **Richardson**, **Emerson** e **Cae-tano** por terem compartilhado comigo dos seus conhecimentos nas disciplinas do mestrado. Esse grupo de estudos foi importante durante a etapa de curso das disciplinas.
- Ex-alunos da Professora Maria Emilia, em especial o **Mauro**, **Lorena** e **Daniela** que sempre me atenderam com presteza todas as vezes que eu necessitei.
- **Cleber Mira** por sempre estar pronto a ajudar, por ter esclarecido as minhas dúvidas e também por ter lido o meu trabalho corrigindo e apontado problemas.

Resumo

Biologia Computacional é uma área da Ciência da Computação que tem por objetivo o estudo e aplicação de técnicas e ferramentas computacionais aos problemas da Biologia Molecular. Dentre os problemas pesquisados, encontra-se o de evolução molecular, onde são estudados métodos para comparar seqüências de espécies distintas, baseados em eventos mutacionais. Estes métodos geram medidas de distância, que podem ser empregadas para verificar o relacionamento em termos evolutivos entre dois organismos. Uma técnica de computar distância é comparar blocos, formados por um ou mais genes, de genomas de dois organismos. O nosso trabalho pertence à área de **Rearranjo de Genomas** que, de forma genérica, visa resolver o problema combinatorial de encontrar uma seqüência mínima de eventos de rearranjo (mutações) que transformam um genoma em outro. Estudamos um evento de rearranjo específico – **transposição**, que move uma porção de genes de um local para outro dentro do mesmo cromossomo. Este evento gera o **problema da ordenação por transposições**, que consiste em computar e encontrar a menor seqüência de transposições que transformam um genoma em outro. Neste trabalho propusemos dois algoritmos de aproximação baseados no formalismo algébrico de Dias e Meidanis para o problema de ordenação por transposições. Implementamos estes algoritmos utilizando a linguagem Java e comparamos os resultados obtidos com outros encontrados na literatura. Este trabalho visa contribuir para encontrar a complexidade do problema de ordenação por transposições que ainda não é conhecida.

Palavras-chave: Problema da Ordenação por Transposições, Formalismo Algébrico, Rearranjo de Genomas, Biologia Computacional

Abstract

Computational Biology is an area that aims to study and to apply techniques and computational tools to problems of molecular biology. One of these problems is molecular evolution, in which methods are proposed for comparing sequences of distinct species, based on mutational events. These methods generate distance measures that could be employed to verify the evolutionary relationship between two organisms. A technique to compute distance is to compare blocks, composed by one or more genes, of the genomes of two organisms. This work belongs to the field of genome rearrangement, that has the objective to solve the combinatorial problem of finding a minimum sequence of rearrangement events that transform a genome into another. We studied a particular rearrangement event - transposition, that moves a portion of genes from a local to another inside one chromosome. This event generates the problem of sorting by transpositions, that consists in computing and finding the minimum sequence of transpositions that transform a genome into another. In this work, we proposed two approximation algorithms based on the algebraic formalism of Dias and Meidanis to solve the problem of sorting by transpositions. We implemented these algorithms using the Java language, and compared the results obtained with the results of other algorithms found in the literature. This work aims to contribute to find the complexity of this problem still unknown.

Keywords: Problem of Sorting by Transpositions, Algebraic Formalism, Genome Rearrangements, Computational Biology

Sumário

Lista de Figuras	8
Lista de Tabelas	10
Capítulo 1 Introdução	11
1.1 Conceitos básicos de Biologia Molecular	12
1.2 Rearranjo de Genomas	15
1.3 Levantamento bibliográfico do problema de ordenação por transposições	19
1.4 Objetivos	20
Capítulo 2 Algoritmos de aproximação de Bafna e Pevzner	21
2.1 Definições	21
2.2 Limite Inferior	23
2.3 O algoritmo de razão 2.0	26
2.4 O algoritmo de razão 1.5	28
Capítulo 3 Formalismo algébrico	38
3.1 Permutações	38
3.2 Problema de Ordenação por Transposições	49
Capítulo 4 Algoritmos de aproximação utilizando o formalismo algébrico	55
4.1 Algoritmo de razão 2.0	55
4.2 Algoritmo de razão 1.5	61
Capítulo 5 Experimentos e análise dos resultados	75
Capítulo 6 Conclusões e trabalhos futuros	78
Referências	79

Lista de Figuras

1.1	Genomas de duas espécies de plantas, onde cada número denota um bloco de genes, e as setas indicam as orientações relativas dos blocos de uma espécie em relação à outra [25].	12
1.2	(a) O açúcar desoxirribose. (b) O açúcar ribose. Os símbolos 1' a 5' representam os átomos de carbono. Nota-se que a diferença entre eles é o oxigênio ligado ao carbono 2' na ribose.	13
1.3	Cada fita é composta por nucleotídeos (fosfato ligado a uma base) ficando as duas fitas ligadas pelo pareamento A/T e C/G. O símbolo a denota o açúcar desoxirribose, o f denota o fosfato e A, T, G e C denotam as quatro bases.	13
1.4	A estrutura espacial do DNA é de dupla hélice, gerada pelas duas fitas contendo bases complementares.	13
1.5	O DNA pode ser representado por uma seqüência de letras, onde cada letra representa uma base. A orientação de cada fita é indicada por 5' e 3', e as duas fitas são orientadas de forma contrária, observando-se ainda o pareamento das bases.	14
1.6	Visão esquemática dos genes dentro dos cromossomos e dos cromossomos dentro do genoma. A região hachurada indica um gene.	15
1.7	Exemplos de eventos de rearranjo (a) <i>Reversão</i> - inverte a ordem e a orientação dos blocos de genes do cromossomo indicados pelo traço. (b) <i>Transposição</i> - move os blocos de genes do cromossomo indicados pelo traço para o local indicado pela seta. (c) <i>Transversão</i> - move os blocos de genes do cromossomo indicados pelo traço para o local indicado pela seta, mas invertendo a ordem e a orientação dos blocos de genes.	16
2.1	O grafo de ciclos, $G(\pi)$ para a permutação $\pi = [8\ 5\ 1\ 4\ 3\ 2\ 7\ 6]$. A ordem de leitura é fixada da direita para a esquerda.	22
2.2	Os grafos-de-ciclos mostram a seqüência de transposições transformando $\pi = [4\ 3\ 2\ 1\ 5]$ em $\iota = [1\ 2\ 3\ 4\ 5]$	23
2.3	$G(\pi)$ contendo três ciclos e $G(\tau\pi)$ contendo um ciclo. Temos então $\Delta c(\tau) = -2$	24
2.4	$G(\pi)$ contendo dois ciclos e $G(\tau\pi)$ contendo dois ciclos. Temos então $\Delta c(\rho) = 0$	24
2.5	(a) $G(\pi)$ contendo um ciclo e $G(\tau\pi)$ contendo um ciclo. Temos então $\Delta c(\tau) = 0$. (b) $G(\pi)$ contendo um ciclo e $G(\tau\pi)$ contendo três ciclos. Temos assim $\Delta c(\tau) = 2$	25

2.6	Um 2-movimento em um ciclo orientado.	27
2.7	Um 0-movimento criando um ciclo orientado.	27
2.8	A transposição $\tau(z, y, x) = \tau(y, x, z)$ transformando C em um ciclo não-orientado em $G(\tau\pi)$	29
2.9	A transposição $\tau(y, z, x) = \tau(z, x, y)$ é um 2-movimento.	29
2.10	Ciclos entrelaçados C e C'	29
2.11	Ciclos não-interferentes C e C'	30
2.12	Ciclo $C = (i_1 \dots i_k)$ cobrindo o ciclo $D = (j_1 \dots j_m)$	31
2.13	0-movimento levando a dois 2-movimentos.	33
2.14	Um 2-movimento válido e 0-2-2-movimentos em ciclos orientados.	35
2.15	Ciclos fortemente orientados: (a,b) primeiro tipo. (c) segundo tipo.	36
5.1	Comparação do tempo de execução dos algoritmos de razões 2.0 e 1.5.	77
5.2	Comparação do tempo de execução da nossa implementação (af15) com as implementações dos algoritmos BP-Bafna e Pevzner [4] sem heurísticas (implementado por Oliveira [7]) e BPh-Bafna e Pevzner [4] com heurísticas (implementado por Soares [24])	77

Lista de Tabelas

5.1	Comparação dos resultados do algoritmo de aproximação 2.0 (af2), 1.5, sem (af15) e com heurística (afh15) com os resultados do algoritmo exato.	75
5.2	Comparação dos resultados dos algoritmo de WDM-Walter, Dias e Meidanis [27], Ch-Christie [5] com heurísticas (implementado por Walter, Curado and Oliveira [26]), H-Hartman [10] (implementado por Honda [11]), BP-Bafna e Pevzner [4] sem heurísticas (implementado por Oliveira [7]) e BPh-Bafna e Pevzner [4] com heurísticas (implementado por Soares [24]) com os resultados do algoritmo exato.	76

Capítulo 1

Introdução

Desde a descoberta da estrutura do DNA em 1953 por Watson e Crick [29] até os projetos atuais de seqüenciamento de genomas, uma grande quantidade de informações sobre os genes de muitos organismos foram geradas, exigindo técnicas complexas de análise destes dados. Neste contexto, surgiu uma área de pesquisa relativamente recente, denominada **Biologia Computacional**. Esta é uma área onde são usados métodos e técnicas de Matemática e Ciência de Computação para resolver problemas da Biologia Molecular [22].

Uma das operações mais importantes em Biologia Computacional é a de comparar seqüências biológicas, pois inferências de funções dos genes identificados nos projetos de seqüenciamento de genomas ao redor de todo o mundo têm sido feitas utilizando esta operação. Medidas de comparação podem ser obtidas a partir de bases, aminoácidos ou genes de duas seqüências.

Algoritmos ou métodos de comparação utilizando bases ou aminoácidos são extensivamente utilizados, como os de Needleman-Wunsch [19], Smith-Waterman [23] e BLAST [1].

Porém, os biólogos descobriram que dois organismos podem evoluir um a partir do outro por mutações em porções grandes dos genomas e não apenas por mutações em bases ou aminoácidos. Nadeau e Taylor [18] e Palmer e Herbon [20] descobriram genomas de duas espécies quase idênticos em genes, mas muito diferentes na ordem destes genes (Figura 1.1). Estes trabalhos mostraram que mutações em porções grandes do genoma constituem uma forma de evolução molecular encontrada na natureza. Para estes tipos de mutações, os algoritmos mencionados anteriormente não produziam resultados adequados.

Neste contexto, algoritmos de comparação utilizando genes ou blocos de genes são utilizados para estudar o relacionamento evolutivo entre dois organismos de tal forma que árvores filogenéticas podem ser construídas a partir dos valores computados por estes algoritmos. Rearranjo de Genomas [22, 24] é uma subárea de Biologia Computacional que tem como objetivo realizar estudos para encontrar uma seqüência mínima de eventos de rearranjo (mutações) que transformam um genoma em outro. De forma genérica, o problema da distância de rearranjo visa encontrar uma seqüência mínima de eventos de rearranjo necessários para

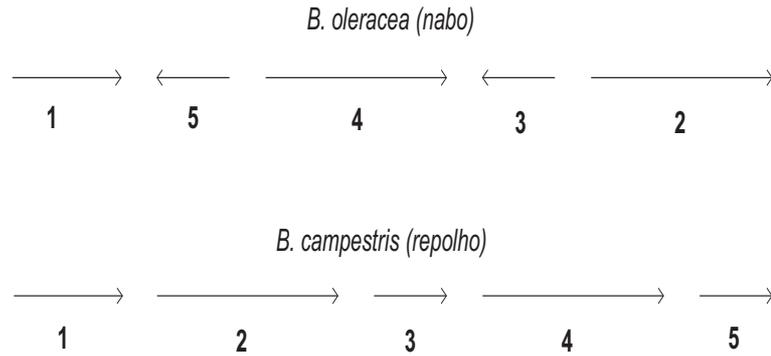


Figura 1.1: Genomas de duas espécies de plantas, onde cada número denota um bloco de genes, e as setas indicam as orientações relativas dos blocos de uma espécie em relação à outra [25].

transformar um genoma no outro.

1.1 Conceitos básicos de Biologia Molecular

Nesta seção apresentaremos conceitos básicos de Biologia Molecular necessários ao entendimento deste trabalho.

Biologia Molecular é uma ciência na qual são estudadas as funções de componentes importantes da estrutura dos organismos vivos, as proteínas e os ácidos nucléicos [22]. Os ácidos nucléicos são os responsáveis por codificar as informações necessárias para sintetizar as proteínas.

Existem dois tipos de ácidos nucléicos: o ácido desoxirribonucléico, DNA, e o ácido ribonucléico, RNA. Estas estruturas contêm, respectivamente, os açúcares desoxirribose e ribose em sua composição química, que contêm cinco átomos de carbono denotado por 1' a 5' (Figura 1.2).

DNA

O DNA é uma macromolécula constituída por duas cadeias que formam uma dupla hélice. Cada uma destas cadeias, chamada de fita, consiste de uma espinha dorsal com repetições das mesmas unidades básicas, os nucleotídeos. Um nucleotídeo é composto por açúcar (desoxirribose), fosfato e por uma das quatro bases: Adenina (A), Guanina (G), Citosina (C) e Timina (T) (Figura 1.3). As duas fitas ficam ligadas pelo pareamento das bases complementares, Adenina e Timina, Citosina e Guanina (Figura 1.4).

A ligação dos nucleotídeos em uma fita ocorre entre o carbono 3' de um nucleotídeo, o resíduo de fosfato e o carbono 5' do outro nucleotídeo. Por este motivo as moléculas de DNA têm uma orientação que, por convenção, inicia-se no carbono 5' e termina no 3', denominada direção canônica (Figura 1.5).

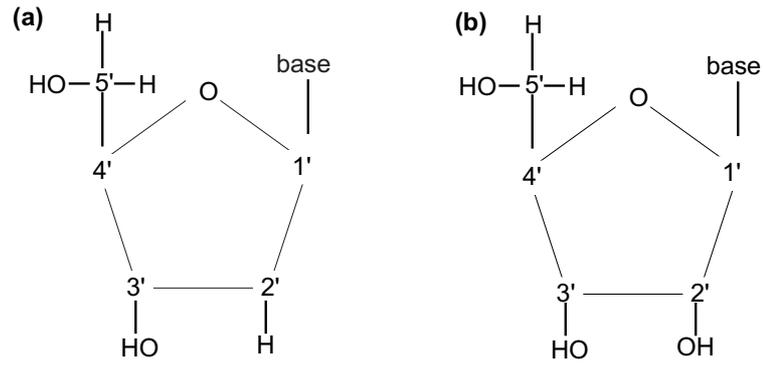


Figura 1.2: (a) O açúcar desoxirribose. (b) O açúcar ribose. Os símbolos 1' a 5' representam os átomos de carbono. Nota-se que a diferença entre eles é o oxigênio ligado ao carbono 2' na ribose.

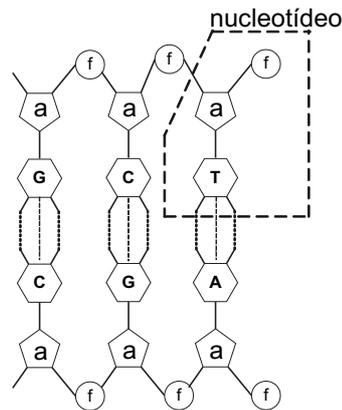


Figura 1.3: Cada fita é composta por nucleotídeos (fosfato ligado a uma base) ficando as duas fitas ligadas pelo pareamento A/T e C/G. O símbolo *a* denota o açúcar desoxirribose, o *f* denota o fosfato e A, T, G e C denotam as quatro bases.



Figura 1.4: A estrutura espacial do DNA é de dupla hélice, gerada pelas duas fitas contendo bases complementares.

5' ... ATCAAGT ... 3'
3' ... TAGTTCA ... 5'

Figura 1.5: O DNA pode ser representado por uma seqüência de letras, onde cada letra representa uma base. A orientação de cada fita é indicada por 5' e 3', e as duas fitas são orientadas de forma contrária, observando-se ainda o pareamento das bases.

É possível obter uma fita de DNA a partir da outra por uma operação denominada complementação reversa, definida da seguinte forma. Considerando-se o trecho da fita na direção canônica, primeiro invertemos o trecho, e depois substituímos as bases deste trecho por suas bases complementares. Por exemplo, tomando $s = ATCAGC$, invertemos s obtendo $s' = CGACTA$ e depois substituindo as bases complementares obtemos $\bar{s} = GCTGAT$. A barra sobre o s denota o complemento reverso \bar{s} de s .

A principal função do DNA é codificar todas as informações necessárias para construir cada proteína (ou RNA) encontrada no organismo, sendo estas informações passadas dos ancestrais para seus descendentes.

RNA

O RNA tem uma estrutura similar ao DNA, exceto que o açúcar é a ribose ao invés da desoxirribose (Figura 1.2).

O RNA não tem a base Timina (T), e sim a Uracila (U), que é a base complementar da Adenina (A). A estrutura espacial do RNA é bem mais variada que a estrutura do DNA, podendo formar por exemplo hélice híbrida RNA-DNA, hélice dupla ou fita simples.

Outra diferença fundamental entre o RNA e o DNA é que o RNA desempenha diferentes funções, como possibilitar a síntese de proteínas a partir das informações do DNA, enquanto que o DNA tem uma única função, que é a de guardar informações.

Proteínas

As proteínas são substâncias essenciais na estrutura dos seres vivos, existindo em cada célula. Exercem diversas funções importantes como o transporte de oxigênio no sangue, a defesa do organismo através dos anticorpos, catalisam diversas reações químicas realizadas pelas enzimas ou ainda formam fibras dos tecidos que sustentam o corpo.

Os aminoácidos são as unidades básicas que constituem as proteínas. Eles possuem um átomo de carbono central (chamado de α), ao qual se ligam um átomo de hidrogênio (H), um grupo amino (NH_2), um grupo carboxílico ($COOH$) e uma cadeia lateral, que pode ser tanto um único átomo de hidrogênio como um complexo conjunto de átomos. A cadeia lateral caracteriza um aminoácido, já

que o resto da molécula é idêntica em todos eles.

As proteínas podem diferir umas das outras pela quantidade de aminoácidos, pelos tipos de aminoácidos ou pela seqüência em que os aminoácidos estão ligados. Assim, mesmo que duas proteínas possuam o mesmo número e os mesmos tipos de aminoácidos, elas podem ser diferentes, dependendo da seqüência em que esses aminoácidos estiverem ligados.

Cromossomos, Genes e Genomas

Cada molécula de DNA é chamada de cromossomo, sendo que cada cromossomo possui trechos que codificam informações para construir proteínas. Cada um destes trechos, denominados de genes, tem a informação necessária para sintetizar um tipo de proteína.

Genoma é o conjunto completo de cromossomos dentro de uma célula (Figura 1.6). O número de cromossomos em um genoma é característica de uma espécie. Por exemplo, cada célula humana possui 46 cromossomos organizados em 23 pares de cromossomos.

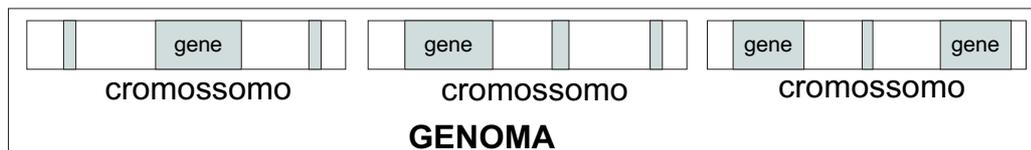


Figura 1.6: Visão esquemática dos genes dentro dos cromossomos e dos cromossomos dentro do genoma. A região hachurada indica um gene.

1.2 Rearranjo de Genomas

Todo o estudo realizado em Rearranjo de Genomas visa resolver um quebra-cabeça combinatorial para encontrar uma seqüência mínima de eventos de rearranjo (mutações), afetando porções grandes de genoma, que transformam um genoma no outro. Assim, os estudos de evolução baseados em rearranjos levam ao **problema de distância de rearranjo**, que de forma genérica, é encontrar a menor seqüência de eventos de rearranjo necessários para transformar um genoma em outro.

Encontrar a *menor* seqüência pode ser explicada pela Hipótese da Parcimônia. Esta hipótese considera que, provavelmente, mutações persistem na natureza através de um número mínimo de ocorrências.

Embora existam diversos tipos de evento, citaremos alguns eventos que ocorrem em um mesmo cromossomo: a **reversão**, que tem o efeito de inverter a ordem e a orientação dos genes na porção do cromossomo que sofreu mutação; **transposição**, que move uma porção de uma região para outra dentro do cromossomo; e a **transversão**, que move os blocos de genes de um local para outro dentro do

cromossomo, mas reverte a ordem e a orientação dos genes (Figura 1.7). Nessa dissertação estudaremos o evento de transposição.

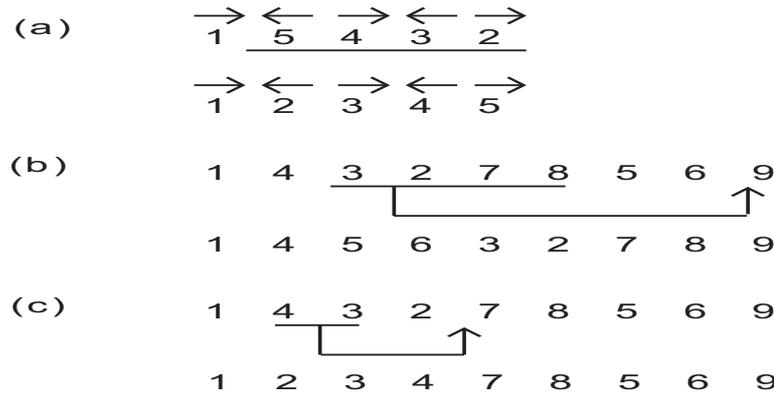


Figura 1.7: Exemplos de eventos de rearranjo (a) *Reversão* - inverte a ordem e a orientação dos blocos de genes do cromossomo indicados pelo traço. (b) *Transposição* - move os blocos de genes do cromossomo indicados pelo traço para o local indicado pela seta. (c) *Transversão* - move os blocos de genes do cromossomo indicados pelo traço para o local indicado pela seta, mas invertendo a ordem e a orientação dos blocos de genes.

Estes diferentes eventos de rearranjo levam a diversos problemas combinatórios, sendo que alguns deles ainda têm complexidades não conhecidas. A seguir formalizamos os três eventos mencionados acima.

Inicialmente modelaremos a seqüência de blocos de genes por uma permutação linear. Assim, a ordem dos blocos de genes é representada pela permutação $\pi = [\pi_1 \ \pi_2 \ \pi_3 \ \dots \ \pi_n]$, onde cada π_i é um inteiro com sinais, se a orientação dos blocos de genes for conhecida, ou um inteiro sem sinais, se estas orientações não são conhecidas. No primeiro caso a seta para a direita é representada pelo sinal $+$ e a seta para a esquerda pelo sinal $-$. Por exemplo, na Figura 1.7, $\overrightarrow{1} \ \overleftarrow{5} \ \overrightarrow{4} \ \overleftarrow{3} \ \overrightarrow{2}$ é representada por $\pi = [+1 \ -5 \ +4 \ -3 \ +2]$ e $1 \ 4 \ 3 \ 2 \ 7 \ 8 \ 5 \ 6 \ 9$ é representada por $\pi = [1 \ 4 \ 3 \ 2 \ 7 \ 8 \ 5 \ 6 \ 9]$.

Para uma permutação π , denotaremos o tamanho da permutação como $|\pi|$. Por exemplo, para $\pi = [1 \ 4 \ 3 \ 2 \ 7 \ 8 \ 5 \ 6 \ 9]$ temos $|\pi| = 9$.

Reversão

Definimos uma **reversão** r do intervalo $[i, j]$, denotado por $r(i, j)$, atuando em uma permutação linear com sinais $\pi = [\pi_1 \dots \pi_{i-1} \pi_i \pi_{i+1} \dots \pi_{j-1} \pi_j \pi_{j+1} \dots \pi_n]$, por

$$r(i, j) \cdot \pi = [\pi_1 \dots \pi_{i-1} \bar{\pi}_j \bar{\pi}_{j-1} \dots \bar{\pi}_{i+1} \bar{\pi}_i \pi_{j+1} \dots \pi_n]$$

onde $\bar{\pi}_k$ indica a inversão do sinal de π_k .

Exemplo 1.2.1 Seja $\pi = [+1 \ -5 \ +4 \ -3 \ +2]$ uma permutação linear com sinais. Os trechos sublinhados indicam as reversões aplicadas nas permutações.

$$\begin{aligned} \pi &= [+1 \ -5 \ +4 \ -3 \ +2] \\ \pi^1 &= r(5, 5) \cdot \pi = [+1 \ -5 \ +4 \ -3 \ -2] \end{aligned}$$

$$\begin{aligned}\pi^2 &= r(3,3) \cdot \pi^1 = [+1 \ -5 \ -4 \ -3 \ -2] \\ \pi^3 &= r(2,5) \cdot \pi^2 = [+1 \ +2 \ +3 \ +4 \ +5] = \sigma\end{aligned}$$

O problema da distância de reversão de cromossomos lineares, com orientações conhecidas, é formalizado como se segue. Dadas duas permutações lineares com sinais π e σ , modelando dois cromossomos lineares com orientações relativas conhecidas, o **problema da distância de reversão** de π e σ é encontrar uma seqüência de reversões r_1, r_2, \dots, r_ρ tais que $r_\rho \cdot r_{\rho-1} \cdot \dots \cdot r_2 \cdot r_1 \cdot \pi = \sigma$ e ρ é mínimo. Chamamos ρ de **distância de reversão entre π e σ** , denotada por $d_\rho(\pi, \sigma)$. No exemplo 1.2.1, temos $d_\rho(\pi, \sigma) = 3$.

Quando não temos informações sobre a orientação dos blocos de genes, isto é, quando a permutação π não tem sinais, a reversão r inverterá apenas a ordem dos blocos de genes. O exemplo seguinte mostra uma seqüência de reversão sem sinais.

Exemplo 1.2.2 *Tomemos as permutações $\pi = [7 \ 1 \ 2 \ 4 \ 5 \ 3 \ 6 \ 8]$ e $\sigma = [1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8]$. Temos que $d_\rho(\pi, \sigma) = 4$, como se segue:*

$$\begin{aligned}\pi &= [7 \ 1 \ 2 \ 4 \ 5 \ 3 \ 6 \ 8] \\ \pi^1 &= r(2,7) \cdot \pi = [7 \ 6 \ 3 \ 5 \ 4 \ 2 \ 1 \ 8] \\ \pi^2 &= r(1,7) \cdot \pi^1 = [1 \ 2 \ 4 \ 5 \ 3 \ 6 \ 7 \ 8] \\ \pi^3 &= r(3,5) \cdot \pi^2 = [1 \ 2 \ 3 \ 5 \ 4 \ 6 \ 7 \ 8] \\ \pi^4 &= r(4,5) \cdot \pi^3 = [1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8] = \sigma\end{aligned}$$

Sem perda de generalidade, podemos assumir σ como sendo a permutação **identidade** $\iota = (+1 \ +2 \ \dots \ +n)$ e considerar o problema equivalente de transformar π em ι . O problema da **ordenação π por reversões** é encontrar a distância de reversão, $d_\rho(\pi, \iota)$, que será denotada por $d_\rho(\pi)$.

A reversão já foi bastante estudada na área de rearranjo de genomas e existem programas disponíveis para analisar este evento, como em mitocôndrias de animais [21] e nas organelas de plantas [2].

Transposição

A **transposição** “corta” certos blocos de genes do cromossomo e “cola” estes blocos de genes em um outro local no mesmo cromossomo. Definimos uma **transposição** atuando em uma permutação π como se segue.

Para toda permutação π , a transposição $\tau(i, j, k)$, $1 \leq i < j \leq n + 1$, $1 \leq k \leq n + 1$ tal que $k \notin [i, j]$, aplicada em π , corta os elementos π_i a π_{j-1} e os “cola” entre π_{k-1} e π_k :

$$\begin{aligned}\tau(i, j, k) \cdot \pi &= \tau(i, j, k) \cdot [\pi_1 \cdots \pi_{i-1} \pi_i \cdots \cdots \cdots \pi_{j-1} \pi_j \cdots \pi_{k-1} \pi_k \cdots \pi_n] = \\ &[\pi_1 \cdots \pi_{i-1} \pi_j \cdots \pi_{k-1} \pi_i \cdots \cdots \cdots \pi_{j-1} \pi_k \cdots \pi_n]\end{aligned}$$

Exemplo 1.2.3 *Tomemos a permutação $\pi = [1 \ 5 \ 6 \ 2 \ 4 \ 3 \ 8 \ 7]$ e a transposição $\tau = (2, 4, 7)$, então:*

$$\tau(2, 4, 7) \cdot \pi = \tau(2, 4, 7) \cdot [1 \ 5 \ 6 \ 2 \ 4 \ 3 \ 8 \ 7] = [1 \ 2 \ 4 \ 3 \ 5 \ 6 \ 8 \ 7]$$

Observamos que a transposição $\tau(i, j, k) \cdot \pi$ tem o efeito de mover os genes $\pi_i, \pi_{i+1}, \dots, \pi_{j-1}$ para uma nova localização no genoma. Além disso, $\tau(i, j, k)$ tem o efeito de trocar os blocos $\pi_i \cdots \pi_{j-1}$ e $\pi_j \cdots \pi_{k-1}$, e $\tau(i, j, k) = \tau(j, k, i)$.

Dadas as permutações π e σ , o **problema de distância de transposição** consiste em achar o número mínimo de transposições que transforma o genoma π no genoma σ , isto é, queremos encontrar uma seqüência de transposições τ_1, \dots, τ_t tal que $\sigma = \tau_t \tau_{t-1} \cdots \tau_1 \pi$ e t é mínimo. O valor t é denominado a distância de transposição $d_t(\pi, \sigma)$ entre os dois genomas π e σ .

Um caso particular surge quando queremos computar a distância de transposição entre o genoma π e o genoma $\iota = [1 \ 2 \ \cdots \ n]$. Neste caso, tem-se o problema equivalente de **ordenação por transposições**, pois a distância de transposição entre π e σ e entre $\pi \sigma^{-1}$ e ι é a mesma, onde ι é a permutação identidade $[1 \ 2 \ \cdots \ n]$. Para simplificar denotaremos $d_t(\pi, \iota)$ por $d_t(\pi)$. O exemplo abaixo mostra uma seqüência de transposições ordenando uma permutação.

Exemplo 1.2.4 *A seguinte seqüência de transposições ordena a permutação $\pi = [4 \ 3 \ 2 \ 1 \ 5]$. Temos então que $d_t(\pi) = 3$.*

$$\tau(1, 4, 5) \cdot \pi = [1 \ 4 \ 3 \ 2 \ 5]$$

$$\tau(2, 4, 5) \cdot \tau(1, 4, 5) \cdot \pi = [1 \ 2 \ 4 \ 3 \ 5]$$

$$\tau(3, 4, 5) \cdot \tau(2, 4, 5) \cdot \tau(1, 4, 5) \cdot \pi = [1 \ 2 \ 3 \ 4 \ 5]$$

Transversão

Como na transposição, a **transversão** “corta” certos blocos de genes do cromossomo e “cola” estes blocos de genes em um outro local, no mesmo cromossomo, mas com a ordem e as orientações dos genes invertidas.

Definimos formalmente uma transversão $\bar{\tau}(i, j, k)$ por três inteiros i, j e k tais que $1 \leq i < j \leq n+1$, $1 \leq k \leq n+1$ e $k \notin [i, j]$ da seguinte forma. Ela “corta” a porção entre as posições i e $j-1$, incluindo os extremos, e “corta” esses elementos exatamente antes da posição k , invertendo a ordem e a orientação dos genes:

$$\begin{aligned} \bar{\tau}(i, j, k) \cdot \pi &= \bar{\tau}(i, j, k) \cdot [\pi_1 \pi_2 \cdots \pi_{i-1} \pi_i \cdots \pi_{j-1} \pi_j \cdots \pi_{k-1} \pi_k \cdots \pi_n] \\ &= [\pi_1 \pi_2 \cdots \pi_{i-1} \pi_j \cdots \pi_{k-1} \bar{\pi}_{j-1} \cdots \bar{\pi}_i \pi_k \cdots \pi_n] \end{aligned}$$

onde $\bar{\pi}_i$ indica a inversão do sinal de π_i .

O problema da distância de transversão de dois cromossomos lineares é formalizado como segue. Dadas duas permutações π e σ , modelando dois cromossomos, o **problema da distância de transversão** de π e σ é encontrar uma seqüência de transversões $\bar{\tau}_1, \bar{\tau}_2, \dots, \bar{\tau}_\delta$, tais que $\bar{\tau}_\delta \bar{\tau}_{\delta-1} \cdots \bar{\tau}_2 \bar{\tau}_1 \pi = \sigma$ e δ é mínimo. Chamamos δ de distância de transversão de π e σ , denotado por $d_\delta(\pi, \sigma)$.

Novamente, sem perda de generalidade, assumindo que $\iota = [1 \ 2 \ \cdots \ n]$, o **problema de ordenação por transversões** é o mesmo que encontrar a distância de transversão entre π e ι , $d_\delta(\pi, \iota)$ que será denotado apenas por $d_\delta(\pi)$.

Para permutações sem sinais, o enunciado do problema é inteiramente análogo, apenas lembrando que as transversões somente invertem a ordem de $\pi_i \cdots \pi_{j-1}$.

1.3 Levantamento bibliográfico do problema de ordenação por transposições

Descreveremos nesta seção os trabalhos que fundamentaram esta pesquisa.

Bafna e Pevzner [4] utilizaram uma estrutura denominada de grafo-de-ciclos, gerado a partir de permutações para resolver o problema de ordenação por transposições. Estudando propriedades de entrelaçamento de ciclos no grafo-de-ciclos, propuseram algoritmos com razões de aproximação 2.0, 1.75 e 1.5. A complexidade de tempo do algoritmo de razão 1.5 é $O(n^2)$, onde n é o tamanho da permutação.

Christie [5], também utilizando grafo-de-ciclos, propôs um algoritmo de aproximação diferente daquele proposto por Bafna e Pevzner, mas também com razão de aproximação 1.5. A complexidade de tempo deste algoritmo é $O(n^4)$ onde n é o tamanho da permutação. Christie comentou em sua tese que a complexidade de tempo teórica do algoritmo de Bafna e Pevzner não poderia ser alcançada em nenhuma implementação.

Walter, Dias e Meidanis [27] apresentaram um algoritmo baseado em uma estrutura denominada diagrama de pontos-de-quebra, com razão de aproximação 2.25. A complexidade de tempo deste algoritmo é $O(b(\pi)^2)$, onde $b(\pi)$ é o número de pontos-de-quebra no diagrama.

Walter, Curado e Oliveira [26] implementaram o algoritmo proposto por Christie [5]. Christie observou que triplas fortemente orientadas em um ciclo eram obtidas em tempo $O(n^3)$, enquanto que os outros passos do seu algoritmo poderiam ser executados em tempo $O(n^2)$. Portanto a complexidade de tempo do algoritmo de Christie era determinada pelo tempo de obter triplas fortemente orientadas. Walter, Curado e Oliveira reduziram a complexidade de tempo do algoritmo de Christie para $O(n^2)$ propondo um novo algoritmo para estas triplas.

Dias e Meidanis [8] propuseram um algoritmo com complexidade de tempo $O(n^2)$ para o problema de ordenação por transposições utilizando transposições por prefixos. Dias e Meidanis [13] relacionaram a teoria baseada no grafo de ciclos de Bafna e Pevzner com a teoria de grupos de permutações em Álgebra, que foi denominada de formalismo algébrico.

Hartman [10] propôs um algoritmo com razão de aproximação 1.5 e complexidade de tempo $O(n^2)$, $n = |\pi|$, para o problema de ordenação por transposições. Hartman simplificou a representação de ciclos no grafo-de-ciclos, gerando uma permutação denominada por ele de simples, que continha apenas ciclos de tamanho 2 e 3, o que simplificou bastante o algoritmo quando comparado com o de Bafna e Pevzner [4]. Posteriormente, Elias e Hartman [9] propuseram um algoritmo de razão de aproximação 1.375 com complexidade de tempo $O(n^2)$, utilizando permutações simples.

Soares, Walter e Dias [24, 28] implementaram o algoritmo de razão 1.5 de

Bafna e Pevzner [4] utilizando uma heurística baseada na técnica de *branch – and – bound*, o que levou aos melhores resultados, quando comparados com os outros algoritmos conhecidos.

Mira e Meidanis [13, 16, 15] utilizaram o formalismo algébrico para demonstrar resultados fundamentais para este problema, como os limites inferior e superior da distância de transposição.

1.4 Objetivos

Os objetivos desta dissertação são:

- Propor um algoritmo de razão de aproximação 1.5 para o problema da ordenação por transposições, utilizando o formalismo algébrico;
- Implementar o algoritmo proposto acima;
- Realizar experimentos com todas as permutações de tamanhos 2 a 11;
- Comparar os resultados obtidos do algoritmo proposto com outros encontrados na literatura.

Esta dissertação contribui para o estudo do problema da distância de transposições cuja complexidade ainda está em aberto.

No Capítulo 2 descrevemos os algoritmos de razão de aproximação 2.0 e 1.5 de Bafna e Pevzner. No Capítulo 3 abordamos conceitos básicos de grupo de permutações e do formalismo algébrico. No Capítulo 4 propomos dois algoritmos de aproximação de razão 2.0 e 1.5 respectivamente, baseados no formalismo algébrico. Mostramos ainda as estruturas de dados e algumas características da implementação desenvolvida para estes algoritmos. No Capítulo 5 mostramos os resultados dos experimentos feitos em todas as permutações de tamanho, 2 a 11, e comparações com outros algoritmos. Finalmente, no Capítulo 6 concluímos e sugerimos caminhos para trabalhos futuros.

Capítulo 2

Algoritmos de aproximação de Bafna e Pevzner

Neste capítulo apresentaremos dois algoritmos de razões 2.0 e 1.5 de Bafna e Pevzner [4, 24] baseados numa estrutura denominada grafo-de-ciclos para resolver o problema da ordenação por transposições. Inicialmente definiremos alguns conceitos básicos, e em seguida, estudaremos os limites inferiores e superiores, que provarão as razões de aproximação destes algoritmos.

Na Seção 2.1, apresentaremos definições que serão utilizadas nos algoritmos. Particularmente, descreveremos a estrutura denominada grafo-de-ciclos. Na Seção 2.2, um limite inferior baseado em propriedades do grafo-de-ciclos é proposto. Na Seção 2.3, um limite superior é computado, e o algoritmo de razão 2.0 é apresentado. Por fim, na Seção 2.4, apresentaremos um novo limite superior e o algoritmo de razão 1.5.

2.1 Definições

Um cromossomo possuindo n blocos de genes, sendo cada bloco representado por um inteiro em $\{1\ 3\ 4\ 2\}$ será formalizado por uma permutação $\pi = [1\ 3\ 4\ 2]$. Como dito anteriormente, um cromossomo será representado por uma permutação.

Definimos inicialmente o conceito de *ponto-de-quebra*. Adicionamos dois novos elementos à permutação π , $\pi_0 = 0$ e $\pi_{n+1} = n + 1$, obtendo uma **permutação estendida** que será chamada somente de permutação. A permutação estendida $\pi = [\pi_0\pi_1\pi_2\dots\pi_n\pi_{n+1}]$ será denominada de agora em diante por permutação.

Dada uma permutação π , o problema da ordenação por transposições é encontrar uma seqüência mínima de transposições que transformam π na permutação identidade $\iota = [0\ 1\ 2\ \dots\ n\ n + 1]$.

Para todo $0 \leq i \leq n$, o par (π_i, π_{i+1}) é um **ponto-de-quebra** se $\pi_{i+1} \neq \pi_i + 1$. Por exemplo, na permutação $\pi = [0\ 1\ 3\ 4\ 2\ 5]$ podemos representar os pontos-de-quebra por $(\pi_1\ \pi_2)$, $(\pi_3\ \pi_4)$ e $(\pi_4\ \pi_5)$. Utilizando o ponto \cdot para representar um ponto-de-quebra, temos os pontos-de-quebra:

$$[0 \ 1 \cdot 3 \ 4 \cdot 2 \cdot 5]$$

O número de pontos-de-quebra de uma permutação é denotado por $b(\pi)$. No exemplo anterior $b(\pi) = 3$. Um algoritmo para o problema de ordenação por transposição deve eliminar todos os pontos-de-quebra da permutação original, pois a permutação identidade $\iota = [0 \ 1 \ 2 \ \dots \ n]$ é a única que não tem pontos-de-quebra.

Em uma transposição aplicada a π é possível eliminar ou inserir, no máximo, três pontos-de-quebra. Então define-se o seguinte limite inferior a $d(\pi)$:

$$d(\pi) \geq \lceil b(\pi)/3 \rceil$$

que não é exato, pois existem permutações nas quais não é possível diminuir três pontos-de-quebra, de uma só vez, com uma transposição. Por exemplo, na permutação $\pi = [0 \cdot 4 \cdot 1 \cdot 3 \cdot 2 \cdot 5]$, $b(\pi) = 5$ e nenhuma transposição diminui três pontos-de-quebra. Aplicando a transposição $\tau(1, 2, 5)$ à permutação π , temos $\tau(1, 2, 5)\pi = [0 \ 1 \cdot 3 \cdot 2 \cdot 4 \ 5]$ e $b(\tau\pi) = 3$.

Bafna e Pevzner [4] propuseram uma estrutura para representar os pontos-de-quebra, denominada grafo-de-ciclos de π , denotado por $G(\pi)$, definido por um conjunto de vértices $\{0, -1, +1, -2, +2, \dots, -n, +n, -(n+1)\}$ e um conjunto de arestas direcionadas bi-coloridas, cinzas e pretas. Cada elemento π_i , $1 \leq i \leq n$, gera dois vértices, $-\pi_i$ e $+\pi_i$, $\pi_0 = 0$ gera o vértice $+0$ e $\pi_{n+1} = n+1$ gera o vértice $-(n+1)$. As arestas pretas são criadas entre os vértices $-\pi_i$ e $+\pi_{i+1}$, sendo direcionadas de $-\pi_i$ para $+\pi_{i+1}$, com $1 \leq i \leq n$, e as arestas cinzas são criadas entre $+i$ e $-(i+1)$, direcionadas de $+i$ para $-(i+1)$, com $0 \leq i \leq n$. A Figura 2.1 mostra um exemplo de $G(\pi)$, para $\pi = [8 \ 5 \ 1 \ 4 \ 3 \ 2 \ 7 \ 6]$. Note que os vértices gerados por cada elemento da permutação são escritos numa linha.

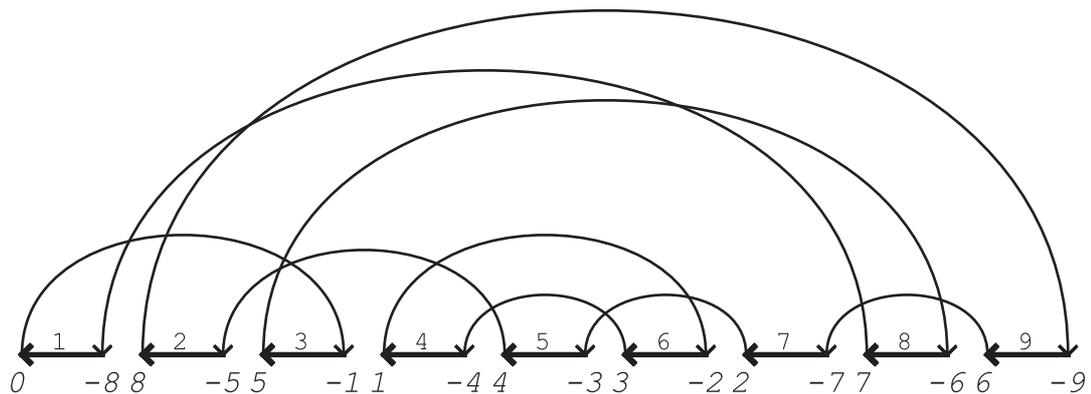


Figura 2.1: O grafo de ciclos, $G(\pi)$ para a permutação $\pi = [8 \ 5 \ 1 \ 4 \ 3 \ 2 \ 7 \ 6]$. A ordem de leitura é fixada da direita para a esquerda.

Intuitivamente, as arestas pretas indicam a situação atual e as arestas cinzas indicam a situação desejada. Quando as arestas pretas se tornam iguais às cinzas em todos os rótulos temos $\pi = \iota$. Então, ordenar uma permutação por transposições utilizando o grafo-de-ciclos significa quebrar os ciclos iniciais de $G(\pi)$ para criar $n + 1$ ciclos da permutação identidade ι . A Figura 2.2 mostra uma seqüência de transposições atuando em $G(\pi)$ para $\pi = [4\ 3\ 2\ 1\ 5]$ e criando $n + 1$ ciclos. Para simplificar, de agora em diante, tiraremos as setas dos grafos-de-ciclos e não representaremos explicitamente os vértices.

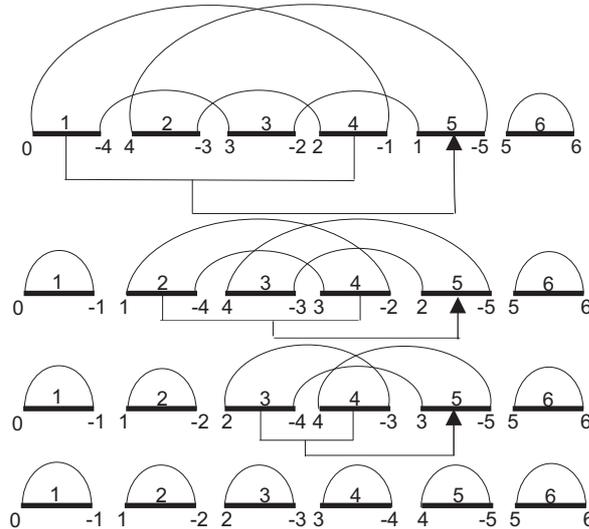


Figura 2.2: Os grafos-de-ciclos mostram a seqüência de transposições transformando $\pi = [4\ 3\ 2\ 1\ 5]$ em $\iota = [1\ 2\ 3\ 4\ 5]$.

Note que para todo vértice em $G(\pi)$, existe uma única aresta incidente e uma única aresta saindo do vértice, sendo que ambas as arestas têm cores opostas. Isso mostra que há uma única decomposição em ciclos para o conjunto de arestas do grafo-de-ciclos. Um ciclo alternado em $G(\pi)$ é um ciclo direcionado, no qual as arestas alternam suas cores.

As arestas pretas de um grafo-de-ciclos podem ser enumeradas, associando i à aresta preta $(-\pi_i, \pi_{i-1})$, $1 \leq i \leq n + 1$. Podemos representar os ciclos apenas por suas arestas pretas.

Um k -ciclo é um ciclo com k arestas pretas. Chamaremos um k -ciclo de **longo** se $k \geq 3$, e de **curto**, se $k \leq 2$.

2.2 Limite Inferior

Nesta seção, apresentaremos um limite inferior com base no número de ciclos do grafo-de-ciclos e no tamanho da permutação.

Existe um total de $2(n + 1)$ arestas, e no máximo $(n + 1)$ ciclos no grafo $G(\pi)$, sendo a permutação identidade a única com $(n + 1)$ ciclos. Denotamos o número de ciclos $G(\pi)$, como sendo $c(\pi)$. Na Figura 2.1 temos $c(\pi) = 3$. Então

a seqüência de transposições que ordenam π deve aumentar o número de ciclos de $c(\pi)$ para $n + 1$. Na Figura 2.2, observe que $c(\pi) = 2$ e $c(\iota) = 6$, sendo $|\pi| = 5$.

Vamos estudar agora como o número de ciclos em $G(\pi)$ varia quando uma transposição é aplicada a π , gerando um novo grafo-de-ciclos $G(\tau\pi)$. Para a permutação π e a transposição τ , define-se a variação no número de ciclos devido a transposição τ , como sendo:

$$\Delta c(\tau) = c(\tau\pi) - c(\pi)$$

Lema 2.2.1 $\Delta c(\tau) \in \{2, 0, -2\}$

Prova A transposição $\tau(i\ j\ k)$ envolve seis vértices do grafo $G(\pi)$,

$$\pi_{i-1}, \pi_i, \pi_{j-1}, \pi_j, \pi_{k-1}, \pi_k,$$

permitindo remover três arestas pretas, $(\pi_i, \pi_{i-1}), (\pi_j, \pi_{j-1}), (\pi_k, \pi_{k-1})$, e adicionar três novas arestas pretas, $(\pi_j, \pi_{i-1}), (\pi_i, \pi_{k-1}), (\pi_k, \pi_{j-1})$ a $G(\tau\pi)$. As três arestas removidas podem pertencer a três, dois ou um ciclo na decomposição de ciclos de $G(\pi)$. Temos três situações distintas, detalhadas abaixo. Nos grafos-de-ciclos desta prova, indicaremos π_i apenas por i .

Caso 1: As arestas removidas pertencem a três ciclos e o número de ciclos após aplicar a transposição τ varia segundo a fórmula $c(\tau\pi) = c(\pi) - 3 + 1$, já que estes três ciclos em $G(\pi)$ originam um único ciclo em $G(\tau\pi)$ (Figura 2.3).

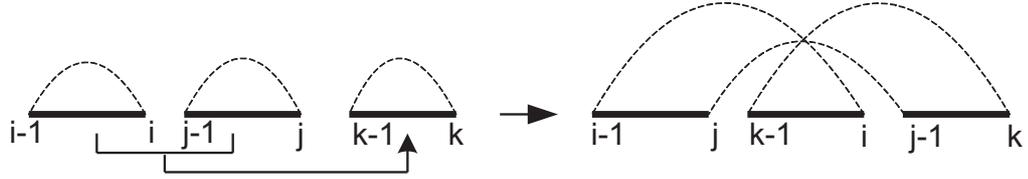


Figura 2.3: $G(\pi)$ contendo três ciclos e $G(\tau\pi)$ contendo um ciclo. Temos então $\Delta c(\tau) = -2$.

Caso 2: As arestas removidas pertencem a dois ciclos e o número de ciclos após aplicar a transposição τ varia segundo a fórmula $c(\tau\pi) = c(\pi) - 2 + 2$, já que os dois ciclos em $G(\pi)$ originam dois ciclos em $G(\tau\pi)$ (Figura 2.4).

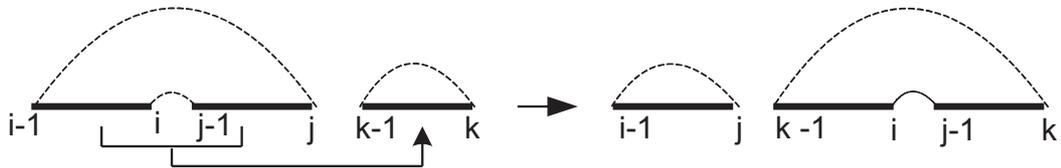


Figura 2.4: $G(\pi)$ contendo dois ciclos e $G(\tau\pi)$ contendo dois ciclos. Temos então $\Delta c(\rho) = 0$

Caso 3: As arestas removidas pertencem a um único ciclo C e existem dois casos. No primeiro, o número de ciclos, após aplicar a transposição τ , varia segundo a fórmula $c(\rho\pi) = c(\pi) - 1 + 1$, já que C em $G(\pi)$ origina um ciclo em $G(\tau\pi)$ (Figura 2.5a).

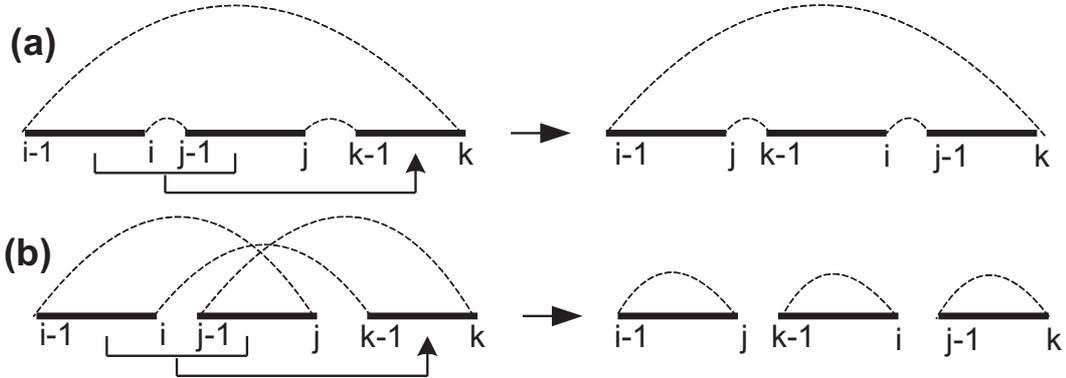


Figura 2.5: (a) $G(\pi)$ contendo um ciclo e $G(\tau\pi)$ contendo um ciclo. Temos então $\Delta c(\tau) = 0$. (b) $G(\pi)$ contendo um ciclo e $G(\tau\pi)$ contendo três ciclos. Temos assim $\Delta c(\tau) = 2$

No segundo caso, o número de ciclos, após aplicar a transposição τ , varia segundo a fórmula $c(\tau\pi) = c(\pi) - 1 + 3$, já que C em $G(\pi)$ origina três ciclos em $G(\tau\pi)$ (Figura 2.5b). \square

O lema anterior prova o seguinte teorema.

Teorema 2.2.1 $d(\pi) \geq \frac{(n+1)-c(\pi)}{2}$

Um **ciclo é ímpar** em $G(\pi)$ se ele possui um número ímpar de arestas pretas, caso contrário ele é **par**. Definimos $c_{imp}(\pi)$ ($c_{par}(\pi)$) como sendo o número de ciclos ímpares (pares) em $G(\pi)$. Para a permutação π e a transposição τ , denotamos $\Delta c_{imp}(\tau) = c_{imp}(\tau\pi) - c_{imp}(\pi)$ como a variação do número de ciclos ímpares devido à transposição τ . Bafna e Pevzner propuseram o lema a seguir, que estabelece a variação de $c_{imp}(\pi)$ ($c_{par}(\pi)$).

Lema 2.2.2 $\Delta c_{imp}(\tau) \in \{2, 0, -2\}$

Com base no lema acima, Bafna e Pevzner [4] propuseram um limite inferior mais aproximado para a distância de transposição.

Teorema 2.2.2 $d(\pi) \geq \frac{n+1-c_{imp}(\pi)}{2}$

Definimos $d(n) = \max_{\pi \in S_n} d(\pi)$ como o **diâmetro de transposição** do grupo simétrico S_n , que é o maior número de transposições necessárias para transformar

uma permutação qualquer na identidade, ou o máximo das distâncias mínimas entre quaisquer dois pares de permutações, π e $\sigma \in S_n$. Observamos que o diâmetro de transposição do grupo simétrico S_n é, no mínimo, $\lfloor \frac{n}{2} \rfloor + 1$, pois temos que $d(n \ n-1 \ \dots \ 1) = \lfloor \frac{n}{2} \rfloor + 1$, para todo n [5, 14].

2.3 O algoritmo de razão 2.0

Com base no limite inferior do Teorema 2.2.2, descreveremos nesta seção um algoritmo de razão 2.0 para o problema da ordenação por transposição.

Para $x \in \{2, 0, -2\}$, define-se um x -movimento em π , como uma transposição τ tal que $\Delta_c(\tau) = x$.

Lema 2.3.1 *Se a transposição τ age em um ciclo em $G(\pi)$ e cria mais de um novo ciclo em $G(\tau\pi)$, então τ é um 2-movimento.*

Um exemplo é dado na Figura 2.5b.

Lema 2.3.2 *Se a transposição τ age em arestas pertencentes a exatamente dois ciclos diferentes então τ é um 0-movimento.*

Um exemplo é dado na Figura 2.4.

Podemos classificar os ciclos como não-orientados, para os quais nenhum 2-movimento é possível, e orientados, para os quais pelo menos um 2-movimento é possível.

Para todo $k > 1$, o ciclo $C = (i_1 \ \dots \ i_k)$ é não orientado, se $i_1 \ \dots \ i_k$ é uma seqüência decrescente, caso contrário, C é um ciclo orientado. Temos ainda uma outra caracterização de ciclos não orientados em termos de direção de arestas. Uma aresta cinza, unindo $\pi_t = i - 1$ com $\pi_s = i$, em $G(\pi)$, é direcionada à esquerda, se $t > s$, e direcionada à direita, caso contrário. Podemos observar que um ciclo $C = (i_1 \ \dots \ i_k)$ é não orientado se, e somente se, $k > 1$ e C têm exatamente uma aresta direcionada à direita (uma aresta cinza de i_1 para i_k).

Neste ponto, definimos mais precisamente ciclos orientados e não orientados. Considere um k -ciclo C visitando, na ordem imposta no ciclo, as arestas pretas $i_1 \ \dots \ i_k$. Um ciclo C pode ser escrito de k formas possíveis, dependendo da escolha da primeira aresta preta. Porém, vamos fixar uma representante canônica de um ciclo C , tomando como aresta preta inicial i_1 , a aresta mais a direita de C em π , isto é, $i_1 = \max_{1 \leq t \leq k} i_t$. No grafo de ciclos da Figura 2.1 temos três ciclos, com representantes canônicas $c_1 = [9, 7, 5, 2]$, $c_2 = [8, 1, 3]$ e $c_3 = [6, 4]$.

Lema 2.3.3 *Se C é um ciclo orientado, então existe um 2-movimento agindo em C . Se C é um ciclo não-orientado, não existe nenhum 2-movimento agindo em C .*

Prova Seja $C = (i_1 \cdots i_k)$ um ciclo orientado, e $3 \leq t \leq k$ um índice, tal que $i_t > i_{t-1}$. Considere a transposição $\tau(i_{t-1}, i_t, i_1)$ agindo em C (Figura 2.6). Esta transposição gera um 1-ciclo (nos vértices $\pi_{i_{t-1}-1}$ e π_{i_t}) e outros ciclos. Daí, pelo Lema 2.3 esta transposição é um 2-movimento. \square

Teorema 2.3.1 *Para uma permutação arbitrária π , existe um 2-movimento ou um 0-movimento, seguido de um 2-movimento.*

Prova Se $G(\pi)$ possui um ciclo orientado, então, pelo Lema 2.3.3, um 2-movimento é possível.

Por outro lado, considere um ciclo não-orientado $C = (i_1 \cdots i_k)$ e seja r a posição do elemento máximo de π no intervalo $[i_2, i_1 - 1]$. Seja s uma posição $\pi_r + 1$ em π .

Claramente $s \notin [\pi_{i_2}, \pi_{i_1}]$. Assumimos sem perda de generalidade que $s > i_1$ e consideramos uma transposição $\tau(r + 1, s, i_2)$. A transposição τ age nas arestas de dois ciclos diferentes e, dessa forma, pelo Lema 2.3, τ é um 0-movimento.

Como τ muda a direção da aresta esquerda (π_{i_1-1}, π_{i_2}) e não muda a direção da aresta direita $(\pi_{i_{k-1}}, \pi_{i_1})$, o ciclo C contendo estas arestas em $G(\tau\pi)$ tem pelo menos duas arestas direcionadas à direita (Figura 2.7). Dessa forma C é um ciclo orientado que permite um 2-movimento. \square

O Teorema 2.3.1 estabelece um crescimento de $c(\pi)$ de, no mínimo, dois em dois movimentos consecutivos, isto é, temos em média a criação de um ciclo em cada aplicação de transposição. Este fato permite estabelecer o seguinte limite superior para o algoritmo de ordenação por transposições.

Teorema 2.3.2 *Qualquer permutação π pode ser ordenada em $n + 1 - c(\pi)$ transposições.*

O Lema 2.3.3 e os Teoremas 2.3.1, 2.3.2 provam a correção do algoritmo com razão de aproximação 2.0 de Bafna e Pevzner [4].

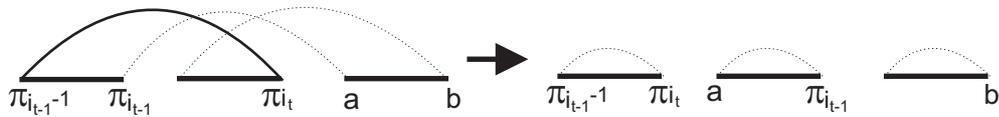


Figura 2.6: Um 2-movimento em um ciclo orientado.

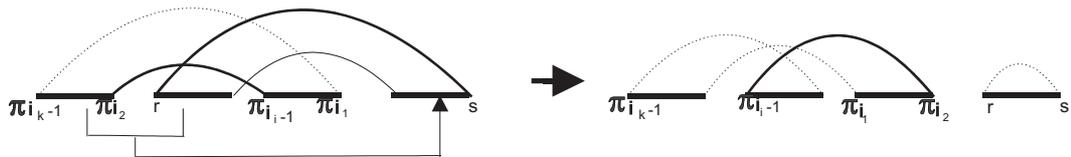


Figura 2.7: Um 0-movimento criando um ciclo orientado.

Algorithm TSort2()**Input:** $\pi, n = |\pi|$ **Output:** $\tau_1, \dots, \tau_k, d_t(\pi) \leq k \leq 2d_t(\pi)$

```

1:  $j = 0$ 
2:  $\pi_j = \pi$ 
3: while  $\pi_j \neq \iota$  do
4:   while há ciclos orientados em  $G(\pi)$  do
5:      $j++$ 
6:      $\tau_j \leftarrow$  2-movimento
7:      $\pi_j \leftarrow \tau_j \pi_{j-1}$ 
8:   end while
9:    $\{G(\pi)$  tem apenas ciclos não orientados  $\}$ 
10:  if há  $k$ -ciclos,  $k > 1$ , em  $G(\pi)$  then
11:     $j++$ 
12:     $\tau_j \leftarrow$  0-movimento
13:     $\pi_j \leftarrow \tau_j \pi_{j-1}$ 
14:     $j++$ 
15:     $\tau_j \leftarrow$  2-movimento
16:     $\pi_j \leftarrow \tau_j \pi_{j-1}$ 
17:  end if
18: end while

```

2.4 O algoritmo de razão 1.5

Considere a tripla (x, y, z) de arestas pretas pertencentes a um mesmo ciclo C em $G(\pi)$. O ciclo C força uma ordem cíclica em x, y, z e temos três representações possíveis desta ordem. Escolheremos como **representante canônica de uma tripla** (x, y, z) aquela iniciando na aresta preta mais à direita, isto é, a aresta preta rotulada por $\max\{x, y, z\}$. Uma tripla na ordem canônica é **não-orientada** se $x > y > z$, e **orientada** se $y < z < x$. No grafo de ciclos da Figura 2.1 temos as seguintes triplas não-orientadas: $(9, 7, 5)$, $(9, 7, 2)$ e $(7, 5, 2)$; e a tripla orientada $(8, 1, 3)$.

Duas seqüências ordenadas de inteiros $V = (v_1, \dots, v_k)$, onde $v_1 < \dots < v_k$, e $W = (w_1, \dots, w_k)$, onde $w_1 < \dots < w_k$, são ditas **seqüências entrelaçadas** se $v_1 < w_1 < v_2 < w_2 \dots < v_k < w_k$ ou $w_1 < v_1 < w_2 < v_2 < \dots < w_k < v_k$. Os conjuntos de inteiros V e W são **conjuntos entrelaçados** se possuírem seqüências entrelaçadas. Seja (x, y, z) uma tripla não orientada, isto é, $x > y > z$. Uma transposição $\tau(i, j, k)$ é uma **transposição embaralhada** com respeito à tripla (x, y, z) se os conjuntos (i, j, k) e (x, y, z) formarem seqüências entrelaçadas. Por exemplo, tomemos $V = \{1, 3, 5, 7\}$ e $W = \{0, 2, 4, 6\}$. A transposição $\tau(0, 2, 4)$ é embaralhada com relação a tripla $(1, 3, 5)$, pois $0 < 1 < 2 < 3 < 4 < 5$.

Lema 2.4.1 [4] *Seja (x, y, z) uma tripla no ciclo C , e $i, j, k \notin C$ três arestas pretas em $G(\pi)$. Então $\tau(i, j, k)$ muda a orientação da tripla (x, y, z) (isto é,*

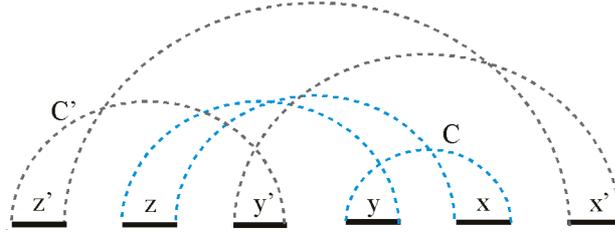


Figura 2.11: Ciclos não-interferentes C e C' .

Prova Se os ciclos C e C' em $G(\pi)$ são entrelaçados, existe uma tripla orientada $(x, z, y) \in C$ e uma tripla não-orientada $(x', y', z') \in C'$ que são entrelaçadas. Pelo Lema 2.4.3 a transposição $\tau(z, y, x)$ define um 2-movimento em C . Por outro lado, já que (x, y, z) e (x', y', z') são entrelaçadas, $\tau(z, y, x)$ é uma transposição embaralhada com respeito a (x', y', z') . Logo, pelo Lema 2.4.1, τ transforma C' em um ciclo orientado em $G(\tau\pi)$ e, pelo Lema 2.3.3, estabelece-se um segundo 2-movimento.

Alternativamente, se C e C' são não-interferentes então existe uma tripla orientada $(x, z, y) \in C$ e $(x', z', y') \in C'$ que são não-entrelaçadas (Figura 2.11). Pelo Lema 2.4.3, uma transposição $\tau(z, y, x)$ define um 2-movimento em C . Mas, (x', z', y') permanece uma tripla orientada (Lema 2.4.1) de C' em $G(\tau\pi)$, o que estabelece um segundo 2-movimento. □

Dizemos que uma **transposição age em dois ciclos** C e C' em $G(\pi)$ se ela agir nas arestas pretas de C e C' . Para provar o Teorema 2.4.1, apresentado à frente, precisaremos também das seguintes observações sobre transposições agindo em dois ciclos.

Lema 2.4.5 [4] *Seja C um ciclo contendo as arestas pretas x e y e D um ciclo contendo as arestas pretas x' e y' . Seja τ a transposição agindo em três das quatro arestas pretas x, y, x', y' .*

- *Se $\{x, y\}$ não entrelaça com $\{x', y'\}$ então τ cria um ciclo com uma tripla não-orientada.*
- *Se $\{x, y\}$ entrelaça com $\{x', y'\}$ então τ cria um ciclo com uma tripla orientada.*

Dizemos que o ciclo $C = (i_1 \dots i_k)$ **cobre** o ciclo $D = (j_1 \dots j_m)$, se $i_k < j_m < j_1 < i_1$. O lema a seguir ilustra uma propriedade importante dos ciclos não-orientados (Figura 2.12).

Lema 2.4.6 *Para todo ciclo não-orientado $C = (\dots a \dots b \dots)$, com arestas arbitrárias a, b , existe um ciclo $D = (\dots c \dots d \dots)$ tal que (a, b) e (c, d) fazem parte de duas triplas que se entrelaçam.*

Teorema 2.4.1 *Se existe um ciclo longo em $G(\pi)$, então um 2-movimento ou um 0-movimento seguido por dois 2-movimentos consecutivos são possíveis em π .*

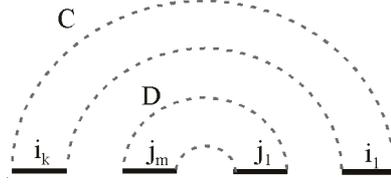


Figura 2.12: Ciclo $C = (i_1 \dots i_k)$ cobrindo o ciclo $D = (j_1 \dots j_m)$.

Prova Se $G(\pi)$ tem um ciclo orientado, então pelo Lema 2.3.3, um 2-movimento é possível. Da mesma forma, se existirem ciclos longos não-orientados C e D com triplas entrelaçadas $(r, s, t) \in C$ e $(x, y, z) \in D$, então um 0-movimento τ agindo nas arestas z, y, x é uma transposição embaralhada para C . Pelo Lema 2.4.1, τ transforma C em um ciclo orientado C' . Pelo Lema 2.4.2 τ transforma D em um ciclo não-orientado D' . É fácil ver que C' e D' são entrelaçados, por isso, pelo Lema 2.4.4 existem dois 2-movimentos consecutivos em $G(\tau\pi)$.

Desta forma, suponhamos que nenhum dos dois ciclos tenha triplas entrelaçadas. Vamos escolher um ciclo longo não-orientado $C = (i_1 \dots i_k)$, tal que C não seja coberto por nenhum ciclo longo. Encontre um ciclo $D = (x \dots c \dots d \dots y)$ tal que os pares (c, d) e (i_1, i_k) se entrelaçam (Lema 2.4.6). Note que se $y < i_k$, então $x < i_1$ caso contrário D cobriria C . Por outro lado, se $y > i_k$, então $x > i_1$ caso contrário (c, d) e (i_1, i_k) não se entrelaçariam. Desta forma, $y < i_k < x < i_1$ ou $i_k < y < i_1 < x$. Sem perda de generalidade, suponhamos o segundo caso. Seja s a aresta mais à direita de C à esquerda de y , isto é, $s = \max_{i \in C, i < y} i$. Podem surgir dois casos:

1. $s > i_k$:

Encontre o ciclo $E = (v \dots c \dots d \dots u)$, tal que os pares (c, d) e (i_k, s) entrelaçam (Lema 2.4.6). Se $u < i_k$ (Figura 2.13(a)) então $v < s$ porque caso contrário existiria E cobrindo C ($v > i_1$) ou teria uma tripla entrelaçada com $(i_k, s, i_1) \in C$ ($s < v < i_1$). Se $u > i_k$ (Figura 2.13(b-e)), então surgem quatro casos dependendo se v pertence a um dos intervalos $[s, y]$, $[y, i_1]$, $[i_1, x]$ ou $[x, n+1]$. As transposições $\tau(x, y, v)$ (Figura 2.13(a)) e $\tau(x, y, u)$ (Figura 2.13(b-e)) são embaralhadas, sem perda de generalidade, com respeito a tripla (i_1, s, i_k) de C , e pelo Lema 2.4.1, transforma C em um ciclo orientado C' em $G(\tau\pi)$. A transposição τ também transforma D e E em D' e em um 1-ciclo em $G(\tau\pi)$. A partir do Lema 2.4.5, D' é não orientado (Figura 2.13(a)). Nos demais casos, D' é orientado quando $v \in [y, x-1]$ e não-orientado caso contrário (Lema 2.4.5). Observe que no primeiro caso C' e D' são entrelaçados (Figura 2.13(b,e)), caso contrário seriam não-interferentes (Figura 2.13(c,d)). Em ambos os casos, dois 2-movimentos são possíveis em $G(\tau\pi)$ (Lema 2.4.4).

2. $s = i_k$:

Seja t a aresta preta mais à esquerda de C , à direita de y , isto é, $t = \min_{i \in C, i > y} i$. Como C é um ciclo longo, $t < i_1$. Encontre o ciclo $E = (v \dots c \dots d \dots u)$, tal que os pares (c, d) e (t, i_1) entrelaçam (Lema 2.4.6). O ciclo E é diferente de D porque caso contrário E e C deveriam ter triplas

entrelaçadas. Se $v > i_1$ então $u > t$ porque caso contrário E cobriria C ($u < i_1$) ou teria uma tripla entrelaçada com $(i_1, t, i_k) \in C$ ($i_1 < u < i_k$). Este caso é similar aos mostrados na Figura 2.13(d,e). Se $v < i_1$, então surgem três casos dependendo em quais dos intervalos $[0, i_k]$, $[i_k, y]$ ou $[y, t]$ u está contido. O primeiro desse três casos é mostrado na Figura 2.13(f), enquanto os outros dois casos são simétricos aos casos da Figura 2.13(c) e 2.5.1.6(e), respectivamente. A transposição $\tau(x, y, u)$ (Figura 2.13(f)) transforma C em um ciclo não-orientado C' (Lema 2.4.1), e os ciclos D e E em um ciclo orientado D' e um 1-ciclo em $G(\tau\pi)$ (Lema 2.4.5). Além disso, C' e D' são entrelaçados, e por isso dois 2-movimentos são possíveis em $G(\tau\pi)$. □

O Teorema 2.4.1 garante a criação de pelo menos quatro ciclos em três movimentos fazendo com que $\Delta c(\tau)$ seja $4/3$ em média, que é melhor que $\Delta c(\tau) = 1$, dado pelo Teorema 2.3.1. Contudo, isto só pode ser aplicado quando $G(\pi)$ possui ciclos longos. No caso de $G(\pi)$ só ter ciclos curtos, o melhor resultado que podemos garantir é um 0-movimento seguido por um 2-movimento criando quatro 1-ciclos a partir de dois 2-ciclos (Teorema 2.3.1). Os Teoremas 2.3.1 e 2.4.1 provam a correção do algoritmo **Tsort**.

Algorithm Tsort()

- 1: **while** $G(\pi)$ tiver um ciclo longo **do**
- 2: Execute um 2-movimento ou 0-2-2-movimento
- 3: **end while**
- 4: **if** $G(\pi)$ tem apenas ciclos curtos **then**
- 5: Execute um 0-movimento seguido de um 2-movimento
- 6: **end if**

O algoritmo acima tem razão de aproximação 1.75 [4]. Para melhorar a razão de aproximação, Bafna e Pevzner reviram o Teorema 2.4.1. Este teorema apenas garante um crescimento no número de ciclos, enquanto o Teorema 2.2.2 indica que precisamos aumentar o número de ciclos *ímpares*. Escolhendo o 2-movimento apropriado, se possível, devemos assegurar que o número de ciclos ímpares cresça de dois em cada 2-movimento.

Dizemos que uma transposição τ é **válida** se $\Delta c(\rho) = \Delta c_{\text{imp}}(\tau)$.

Para um ciclo C contendo arestas i e j , definimos $d(i, j)$ como o número de arestas pretas entre os vértices π_i e π_j em C . Em particular $d(i, j) = 1$ para arestas pretas consecutivas i e j).

Lema 2.4.7 *Se existe um ciclo orientado em $G(\pi)$, então temos um 2-movimento válido ou um 0-movimento válido seguido de dois 2-movimentos válidos consecutivos.*

Prova Suponha que não exista nenhum 2-movimento válido em π . Para um ciclo orientado arbitrário C em $G(\pi)$ considere o conjunto S de triplas orientadas de C tal que a distância entre o primeiro e o segundo elemento da tripla seja ímpar:

$$S = \{(x, y, z) : x, y, z \in C \text{ e } d(x, y) \text{ é ímpar}\}$$

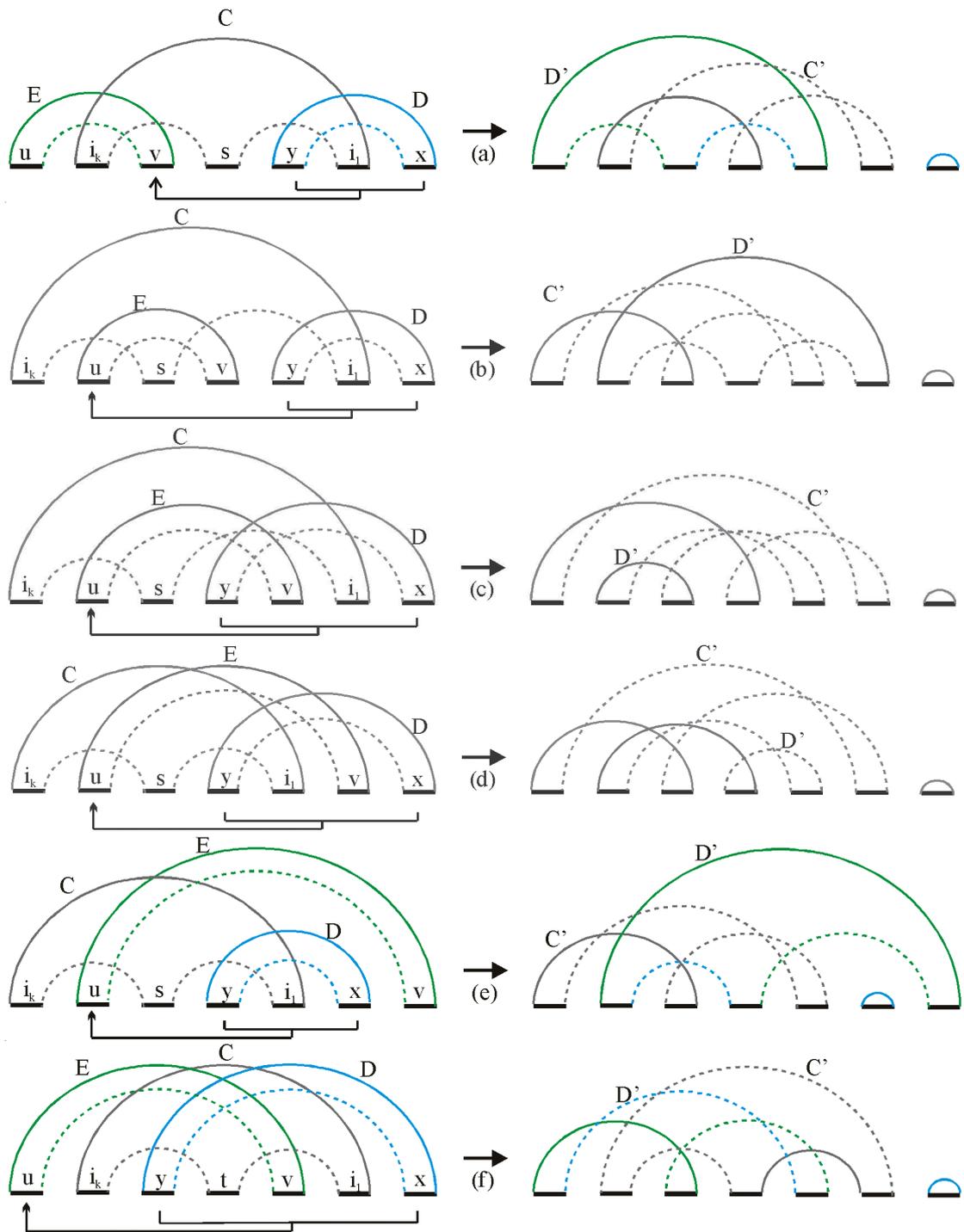


Figura 2.13: 0-movimento levando a dois 2-movimentos.

A observação que todo ciclo orientado C tem uma tripla orientada (x, y, z) tal que x e y são arestas pretas consecutivas em C implica que S é não-vazio. Seja (x, y, z) uma tripla em S com máximo x .

A transposição τ agindo nas arestas y, z e x transforma C em 3 ciclos C_1, C_2 e C_3 consistindo das arestas pretas $d(x, y), d(y, z)$ e $d(z, x)$. Como $(x, y, z) \in S$, o ciclo C_1 é ímpar. Se C_2 ou C_3 forem ímpares então $\Delta_{c_{\text{imp}}}(\tau) = 2$ e τ é um 2-movimento válido contradizendo a suposição que não existe nenhum 2-movimento válido em π . Desta forma, tanto $d(y, z)$ como $d(z, x)$ são pares. Como ambas $d(y, z)$ e $d(z, x)$ são pares, os trechos de C , de y para z e de z para x contém, no mínimo, duas arestas. Seja a a aresta preta que precede z em C e b a aresta preta posterior a z em C (Figura 2.14(a)).

Se $y < a < x$ então a transposição τ agindo nas arestas y, a e x cria ciclos de tamanho $d(y, z) - 1$ e $d(x, y)$. Ambos os números são ímpares e, deste modo, τ é um 2-movimento válido contradizendo assim a suposição. Desta forma $a \notin [y, x]$. Argumentos simétricos demonstram que $b \notin [y, x]$.

Se $a > x$ então (a, z, x) é uma tripla orientada com $d(a, z) = 1$ ímpar, contradizendo assim a escolha de (x, y, z) . Desta forma $a < y$. Se $b > x$ então (b, a, z) é uma tripla orientada com $d(b, a) = d(b, x) + d(x, y) + d(y, a) = (d(z, x) - 1) + d(x, y) + (d(y, z) - 1)$ ímpares, contradizendo assim a escolha de (x, y, z) . Logo $a, b < y$.

As situações descritas pela condição $b < a$ e $a < b$ são apresentadas nas (Figura 2.14(b) e (c)). Se $b < a$, então $\tau(b, a, z)$ é um 2-movimento válido (Figura 2.14(b)). Se $a < b$, então existe um 2-movimento mas nenhum 2-movimento válido em π . Contudo, existe um 0-movimento válido seguido de dois 2-movimentos válidos consecutivos (Figura 2.14(c)).

□

A Figura 2.14(c) apresenta um exemplo de um ciclo orientado que não permite um 2-movimento válido. Este ciclo tem uma estrutura de “auto-entrelaçamento” complexa e no que se segue, tentaremos evitar a construção de tais ciclos. Para atingir esse objetivo, definimos **ciclos fortemente orientados**, que tem uma certa estrutura de “auto-entrelaçamento” ao longo de todos os ciclos orientados. Esta idéia levará ao Teorema 2.4.2.

Seja $C = (i_1, \dots, i_k)$ um ciclo em $G(\pi)$ e $C^* = (j_1 = i_1 > \dots > j_k)$ a seqüência das arestas pretas de C em ordem decrescente. As seqüências C e C^* coincidem para um ciclo não-orientado e são diferentes caso contrário. Definimos **ciclos fortemente orientados** como os ciclos para os quais C^* pode ser transformado em C por uma única transposição, isto é, C pode ser particionado nos trechos $C_1 = (i_1, \dots, i_a), C_2 = (i_{a+1}, \dots, i_b), C_3 = (i_{b+1}, \dots, i_c)$ e $C_4 = (i_{c+1}, \dots, i_k)$ tais que $C = C_1 C_2 C_3 C_4$ e $C^* = C_1 C_3 C_2 C_4$ (C_4 pode estar vazio). Por exemplo, a Figura 2.14(b) mostra um exemplo de um ciclo fortemente orientado $C = x y a z b$ transformado em $C^* = x z y a b$ por uma única transposição. Claramente, cada ciclo fortemente orientado tem exatamente duas arestas direcionadas à direita. Por outro lado, nem todos os ciclos orientados com duas arestas direcionadas à direita são fortemente orientados (Figura 2.14(c)).

Lema 2.4.8 [4] *Um ciclo fortemente orientado permite um 2-movimento válido.*

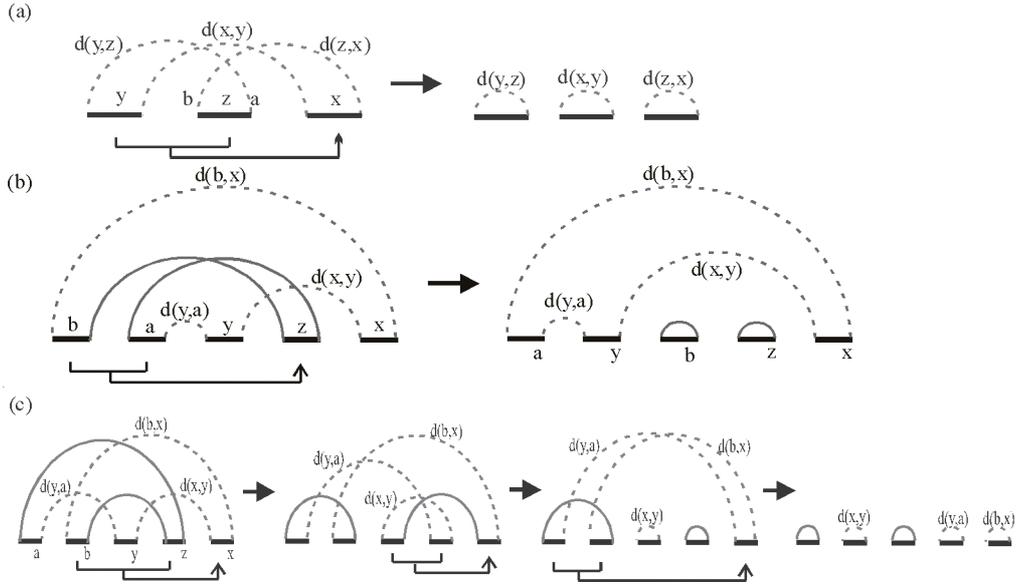


Figura 2.14: Um 2-movimento válido e 0-2-2-movimentos em ciclos orientados.

A seguir apresentamos dois lemas que mostram como os ciclos fortemente orientados surgem de ciclos não-orientados.

Lema 2.4.9 [4] *Se τ é uma transposição embaralhada com relação a um ciclo não-orientado C , então τ transforma C em um ciclo fortemente orientado em $G(\tau\pi)$.*

Lema 2.4.10 [4] *Sejam $D(x, \dots, y)$ e $E(x', \dots, y')$ dois ciclos não-orientados em $G(\pi)$ sem nenhuma tripla entrelaçada e seja τ a transposição agindo em três das quatro arestas pretas x, y, x', y' . Então τ cria um ciclo fortemente orientado e somente se D e E tiverem pares de arestas entrelaçadas.*

Cada ciclo fortemente orientado tem exatamente duas arestas direcionadas à direita, uma das quais é da forma (r, i_1) . Denotamos a outra como (s, t) . Para os ciclos fortemente orientados do primeiro tipo (Figuras 2.15(a) e (b)) definimos:

- $r' = \max i$ ($i \in$ esquerda)
- $t' = \min i$ ($i \in$ direita)

e consideramos os três intervalos $I_1(C) = [r', r]$, $I_2(C) = [t, t']$ e $I_3 = [0, s] \cup [i_1, n + 1]$.

Para os ciclos fortemente orientados de segundo tipo (Figura 2.15(c)) definimos

- $s' = \max i$ ($i \in$ esquerda)
- $t' = \min i$ ($i \in$ direita)
- $a = \max i$ ($i \in$ centro)
- $a' = \min i$ ($i \in$ centro)

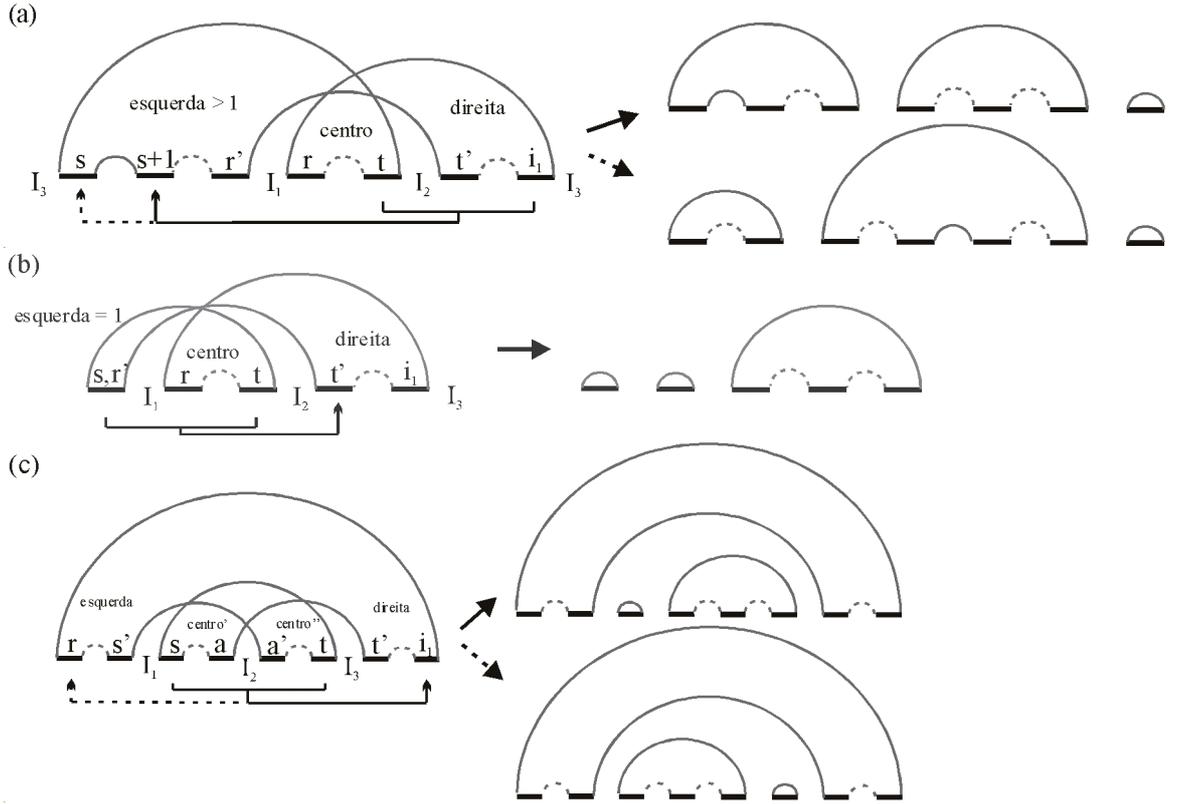


Figura 2.15: Ciclos fortemente orientados: (a,b) primeiro tipo. (c) segundo tipo.

e consideramos os três intervalos $I_1(C) = [s', s]$, $I_2(C) = [t, t']$ e $I_3 = [a, a']$.

Um ciclo fortemente orientado C e um ciclo não-orientado $C' = (i_1, \dots, i_k)$ são **fortemente cruzados** se existir uma aresta preta x em C' tal que cada um dos conjuntos $I_1(C)$, $I_2(C)$ e $I_3(C)$ contém exatamente um elemento da tripla (i_1, x, i_k) . Note que o 2-movimento para C descrito na prova do Lema 2.4.7 forma uma transposição embaralhada, sem perda de generalidade, (i_1, x, i_k) . Esta observação e o Lema 2.4.8 permitem estabelecer o seguinte lema.

Lema 2.4.11 [4] *Se $G(\pi)$ tem ciclos fortemente cruzados então existem dois 2-movimentos válidos consecutivos em $G(\pi)$.*

Finalmente, são usados os Lemas 2.4.7 a 2.4.11 para provar o Teorema 2.4.2.

Teorema 2.4.2 [4] *Se existe um ciclo longo em $G(\pi)$, então um 2-movimento válido ou um 0-movimento válido seguido de dois 2-movimentos válidos consecutivos são possíveis em π .*

Teorema 2.4.3 [4] *Se $G(\pi)$ tem somente ciclos curtos, então é possível um 0-movimento seguido por um 2-movimento válido.*

Algorithm TransSort()

- 1: **while** $G(\pi)$ tem um ciclo longo **do**
- 2: Execute um 2-movimento válido ou 0-2-2-movimentos válidos

3: **end while**
4: **if** $G(\pi)$ tem apenas ciclos curtos **then**
5: Execute um 0-movimento válido seguido de 2-movimento válido
6: **end if**

Bafna e Pevzner [4] apresentam uma complexidade de $O(n^2)$ para o algoritmo *TransSort*. Os Teoremas 2.3.1, 2.4.2 e 2.4.3 permitem estabelecer os seguintes corolários.

Corolário 2.4.1 *O Algoritmo **TransSort** ordena π em no máximo $^{3/4} \cdot (n + 1 - c_{\text{imp}}(\pi))$ transposições, assegurando assim uma razão de 1.5.*

Corolário 2.4.2 *O diâmetro da transposição do grupo simétrico S_n é no máximo $^{3/4} n$.*

Capítulo 3

Formalismo algébrico

Bafna e Pevzner [4, 3] utilizaram grafo de ciclos para modelar o problema de ordenação por transposição. Neste capítulo, descreveremos o formalismo algébrico que Dias e Meidanis [13, 16, 15] propuseram para tentar resolver este problema.

Na Seção 3.1, apresentamos conceitos básicos de permutações usados no formalismo algébrico. Na Seção 3.2, utilizando o formalismo algébrico [13, 16, 15], definimos os conceitos de Bafna e Pevzner [4, 3] e um limite inferior, conforme provado por Mira e Meidanis [16, 15].

3.1 Permutações

Seja E um conjunto arbitrário. Uma permutação π sobre E é uma bijeção $\pi : E \rightarrow E$. A permutação identidade é a permutação ι tal que $\iota x = x \forall x \in E$ [12, 17].

Dadas as permutações π e σ sobre E , a composição destas permutações é a permutação $\pi\sigma : E \rightarrow E$ tal que $\pi\sigma(x) = \pi(\sigma(x)) \forall x \in E$.

A composição é aplicada a partir “da direita”, isto é, em $\pi\sigma(x)$ primeiro localizamos a imagem de x por σ e então aplicamos π em σx . O produto $\theta = \pi\sigma$, onde π e σ são permutações sobre E é simplesmente a composição de π com σ .

O operador produto é associativo, isto é, dadas as permutações π , σ e θ então $\pi(\sigma\theta) = (\pi\sigma)\theta$. A permutação inversa de π é a permutação π^{-1} , tal que $\pi\pi^{-1} = \pi^{-1}\pi = \iota$.

O conjunto de todas permutações sobre o conjunto E e o operador produto definem o grupo simétrico $S(E)$. Sempre que o domínio de permutações é $\{1, 2, \dots, n\}$ o grupo simétrico é denotado por S_n .

O produto entre duas permutações π e σ sobre um conjunto E é executado da seguinte forma. Para cada elemento $x \in E$, usamos a fórmula $(\pi\sigma)x = \pi(\sigma(x))$. Por exemplo se definirmos $E = \{a, b, c, d, e, f, g\}$ e considerarmos as permutações:

$$\pi a = b, \pi b = d, \pi c = f, \pi d = a, \pi e = c, \pi f = e, \pi g = g \text{ e}$$

$$\sigma a = g, \sigma b = e, \sigma c = c, \sigma d = a, \sigma e = f, \sigma f = d, \sigma g = b$$

O produto $\theta = \pi\sigma$ é dado por:

$$\theta a = g \quad \theta b = c \quad \theta c = f \quad \theta d = b \quad \theta e = e \quad \theta f = a \quad \theta g = d$$

Neste caso, por exemplo, $\theta a = \pi\sigma a = \pi g = g$

Em um produto $\pi\sigma$, sempre que $\pi = \sigma$, escrevemos $\pi\pi$ como π^2 . Em geral, escrevemos a composição $\pi \cdots \pi$ (k vezes) como π^k para $k \geq 1$ e por convenção $\pi^0 = \iota$.

Para toda permutação π sobre um conjunto finito E existe um número inteiro $N \geq 0$, tal que $\pi^N = \iota$, isto é, a aplicação sucessiva da composição de uma permutação nela mesma depois de um certo número de vezes resulta na identidade. A **ordem** de uma permutação π , denotado por $ord(\pi)$, é o menor inteiro não negativo em que $\pi^{ord(\pi)} = \iota$.

Por exemplo: considere as permutações π e σ sobre $E = \{a, b, c\}$:

$$\pi a = b, \quad \pi b = a, \quad \pi c = c$$

e

$$\sigma a = c, \quad \sigma b = a, \quad \sigma c = b.$$

Então temos:

$$\pi\sigma a = c \quad \pi\sigma b = b \quad \pi\sigma c = a.$$

Como $\pi^2 = \iota$ e $\pi \neq \iota$, então $ord(\pi) = 2$ e $ord(\sigma) = 3$.

Seja π uma permutação sobre E e n um inteiro positivo. Definimos

$$\pi^{-n} = (\pi^{-1})^n$$

. Além disso temos:

$$\pi^{-n} = (\pi^{-1})^n = (\pi^n)^{-1}.$$

Os resultados a seguir são bem conhecidos [12, 17] e algumas provas não serão apresentadas.

Duas propriedades fundamentais relacionadas ao produto de permutações são mostradas na proposição a seguir.

Proposição 3.1.1 *Seja π uma permutação sobre E , para quaisquer inteiros r, s temos:*

- $\pi^r \pi^s = \pi^{r+s}$
- $(\pi^r)^s = \pi^{rs}$

Observe que $\pi^m \pi^n = \pi^n \pi^m$ para quaisquer inteiros m, n é uma consequência direta da Proposição 3.1.1.

Considere as sucessivas aplicações de uma permutação π sobre E :

$$\pi, \pi x, \pi \pi x = \pi^2 x, \pi \pi \pi x = \pi^3 x, \dots, \pi^m x, \dots$$

onde $x \in E$ é um elemento arbitrário. O conjunto de todas as imagens de um dado elemento $x \in E$ sob potências de π é $\pi^m x$. Um elemento y é acessível a partir de x , denotado por $x \equiv_\pi y$, se $y = \pi^m x$.

Proposição 3.1.2 *Dada uma permutação π sobre E , a relação $x \equiv_\pi y$ é uma relação de equivalência onde $x, y \in E$.*

Considere a seguinte permutação σ sobre o conjunto $E = \{a, b, c, d, e, f, g, h, i, j\}$:

$$\sigma = \begin{pmatrix} a & b & c & d & e & f & g & h & i & j \\ b & d & i & f & g & a & c & h & e & j \end{pmatrix}$$

A permutação é representada na forma de matriz, onde cada elemento na primeira linha é mapeado para o elemento na mesma coluna na segunda linha. Por exemplo, o elemento c é mapeado para o elemento i . O conjunto de elementos acessíveis a partir do elemento a é $A(a) = \{a, b, d, f\}$. Denotamos o conjunto de elementos acessíveis a partir de um elemento x por $A(x)$. Por exemplo, para a permutação σ temos $A(a) = A(b) = A(d) = A(f)$. Também temos $A(c) = \{c, i, e, g\}$, $A(h) = \{h\}$ e $A(j) = \{j\}$. Então os conjuntos $A(a)$, $A(c)$, $A(h)$ e $A(j)$ são partes de E . Geralmente, como a relação de acessibilidade é uma relação de equivalência em um conjunto arbitrário E , existe uma divisão de E em classes de equivalências disjuntas. Como as classes de equivalência são definidas a partir de uma permutação, a divisão de E em classes de equivalência sugere uma notação alternativa para representar uma permutação. Representamos uma permutação como um produto de ciclos. Para uma permutação π sobre E , tomemos um elemento x sobre E . Achamos os sucessivos elementos πx , $\pi \pi x$, $\pi \pi \pi x$, etc. até que para um finito k , $\pi^k x = x$. Assim termina o primeiro ciclo. Escolha um novo elemento em E tal que este elemento não pertence a nenhum ciclo anterior. Repita o procedimento até que todos elementos em E tenham sido usados. Por exemplo, se $E = \{1, 2, 3, 4, 5, 6\}$ e

$$\pi = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 3 & 4 & 1 & 6 & 5 \end{pmatrix}$$

um possível produto de ciclos é $(1\ 2\ 3\ 4)(5\ 6)$. Um produto de ciclos para uma permutação π é também chamado de decomposição de ciclos de π . Então, podemos representar uma permutação pelo produto de ciclos, onde cada ciclo é da forma: $(a_1\ a_2\ \dots\ a_m)$ e é definido como $\pi a_k = a_{k+1}$ para $k = 1, \dots, m-1$ e $\pi a_m = a_1$.

Uma permutação π sobre $E = \{x_1, x_2, \dots, x_n\}$ será representada da seguinte forma: $\pi = [x_1\ x_2\ \dots\ x_n]$.

O **suporte**, $Supp(\pi)$, de uma permutação π é o subconjunto de elementos não fixados em π . Por exemplo, na permutação $\pi = [0\ 3\ 1\ 5\ 4\ 2]$ sobre $E = \{0, 1, 2, 3, 4, 5\}$, os elementos 0 e 4 são fixados e $Supp(\pi) = \{1, 2, 3, 5\}$.

Dada uma permutação π sobre E , a **órbita** de um elemento $x \in E$ em π é o conjunto $orb(x, \pi) = \{y | y = \pi^k x \text{ onde } k \text{ é um inteiro}\}$. O **número de órbitas** de π sobre E é denotado por $o(\pi, E)$. Uma **órbita** é chamada de **trivial** quando contém somente um elemento, caso contrário é chamada de **não trivial**.

Um **ciclo** é uma permutação com no máximo uma órbita não trivial. Um **k -ciclo**, $k > 1$, é um ciclo com órbita não trivial de tamanho k . Representa-se um ciclo por uma seqüência de números fechados entre parênteses. Por exemplo, a permutação $\pi = [3\ 1\ 5\ 4\ 2]$ sobre $E = \{1, 2, 3, 4, 5\}$ é um 4-ciclo que pode ser representado por $(1\ 3\ 5\ 2)$. A identidade é 1-ciclo denotada por $\iota = (1)(2) \cdots (n)$.

Dois **ciclos** são **disjuntos** quando ambos têm órbitas não-triviais disjuntas ou quando um dos ciclos é a identidade. Para uma permutação π , a decomposição de um *ciclo* é um produto de k -ciclos, não necessariamente disjuntos, que é igual a π . Por exemplo, o produto $(1\ 3)(3\ 5)(5\ 2)$ é uma decomposição em 2-ciclos do ciclo $(1\ 3\ 5\ 2)$.

Teorema 3.1.1 *Qualquer permutação sobre E pode ser representada como um único produto de ciclos disjuntos, isto é, há uma única decomposição de E em órbitas disjuntas.*

Por exemplo, a representação em ciclos da permutação $\pi = [1\ 3\ 0\ 5\ 4\ 2]$ sobre $E = \{0, 1, 2, 3, 4, 5\}$ é $(0\ 1\ 3\ 5\ 2)(4)$, que é a sua decomposição em ciclos disjuntos. Uma permutação pode ter infinitas decomposições em ciclos ou nenhuma decomposição em ciclos disjuntos. A permutação π anterior pode também ser obtida por dois produtos:

$$\pi = (0\ 1)(1\ 3\ 5)(0\ 4)(5\ 2)(0\ 4)$$

ou

$$\pi = (0\ 1\ 3)(3\ 5)(0\ 4\ 5)(4\ 5\ 2)(0\ 4)$$

Elevando uma permutação π à potência positiva $p \geq 1$, obtemos a seguinte propriedade:

$$\pi^p = (\pi_1 \pi_2 \cdots \pi_r)^p = \pi_1^p \pi_2^p \cdots \pi_r^p$$

onde $\pi_1, \pi_2, \dots, \pi_r$ são os ciclos pertencentes a decomposição de ciclos disjuntos de π .

Como o operador produto não é comutativo em geral, a regra de distribuição de exponenciação, que é $(\pi\sigma)^r = \pi^r \sigma^r$, é válida somente para certas permutações, como ciclos disjuntos.

A permutação inversa de um ciclo $\pi = (a_1\ a_2\ \cdots\ a_n)$ é dada por $\pi^{-1} = (a_n\ a_{n-1}\ \cdots\ a_1)$. Para uma permutação $\alpha = \alpha_1 \cdots \alpha_r$, com $\alpha_1, \alpha_2, \dots, \alpha_r$ ciclos sobre E , a inversa é $\alpha_r^{-1} \cdots \alpha_1^{-1}$. Observe que se $\alpha_r^{-1} \cdots \alpha_1^{-1} = \alpha_1^{-1} \cdots \alpha_r^{-1}$ então os ciclos disjuntos comutam.

Dadas as permutações α e β sobre E , a **conjugação** de β por α é a permutação $\alpha\beta\alpha^{-1}$. A operação de conjugação é representada por $\alpha \cdot \beta = \alpha\beta\alpha^{-1}$. Dizemos que β é um **conjugado** de α quando há uma permutação γ tal que $\beta = \gamma \cdot \alpha$.

A operação de conjugação preserva a estrutura da permutação que está sendo conjugada. Por exemplo, para $\beta = (a\ b\ c\ d\ e)$:

$$\alpha(a\ b\ c\ d\ e)\alpha^{-1} = (\alpha a\ \alpha b\ \alpha c\ \alpha d\ \alpha e)$$

Lema 3.1.1 *Dadas as permutações $\alpha, \beta, \gamma, \tau \in S(E)$, então:*

1. $\tau \cdot (\alpha\beta) = (\tau \cdot \alpha)(\tau \cdot \beta)$
2. $\iota \cdot \alpha = \alpha$
3. $\tau \cdot \alpha^{-1} = (\tau \cdot \alpha)^{-1}$
4. $\gamma \cdot (\tau \cdot \alpha) = \gamma\tau \cdot \alpha$
5. $(\tau \cdot \alpha)^k = \tau \cdot \alpha^k$ onde $k \geq 0$

Prova Considere as permutações $\alpha, \beta, \gamma, \tau \in S(E)$ sobre E então:

1. $\tau \cdot (\alpha\beta) = \tau(\alpha\beta)\tau^{-1} = \tau\alpha\tau^{-1}\tau\beta\tau^{-1} = (\tau \cdot \alpha)(\tau \cdot \beta)$
2. $\iota \cdot \alpha = \iota\alpha\iota^{-1} = \iota\alpha\iota = \alpha$
3. $\tau \cdot \alpha^{-1} = \tau\alpha^{-1}\tau^{-1} = \tau(\tau\alpha)^{-1} = (\tau\alpha\tau^{-1})^{-1} = (\tau \cdot \alpha)^{-1}$
4. $\gamma \cdot (\tau \cdot \alpha) = \gamma(\tau\alpha\tau^{-1})\gamma^{-1} = \gamma\tau\alpha(\gamma\tau)^{-1} = (\gamma\tau) \cdot \alpha$
5. Provaremos utilizando indução em k . Para $k = 0$,

$$(\tau \cdot \alpha)^0 = \iota = \tau \cdot \iota = \tau \cdot \alpha^0$$

Suponha $(\tau \cdot \alpha)^l = \tau \cdot \alpha^l$, para $l < k$. Então temos $(\tau \cdot \alpha)^k = (\tau \cdot \alpha)(\tau \cdot \alpha)^{k-1}$. Por hipótese de indução $(\tau \cdot \alpha)^{k-1} = \tau \cdot \alpha^{k-1}$, então

$$(\tau \cdot \alpha)^k = (\tau \cdot \alpha)(\tau \cdot \alpha)^{k-1} = \tau\alpha\tau^{-1}\tau\alpha^{k-1}\tau^{-1} = \tau\alpha^k\tau^{-1} = \tau \cdot \alpha^k$$

□

Lema 3.1.2 *Dadas as permutações α, β e γ sobre E , temos:*

1. Se $\beta = \gamma \cdot \alpha$ então α e β têm a mesma estrutura de ciclo
2. Se α e β têm a mesma estrutura de ciclo então β é um conjugado de α
3. A relação “é um conjugado de” é uma relação de equivalência

Prova

1. Vamos mostrar que para cada ciclo em α há um único ciclo correspondente de mesmo tamanho em $\gamma \cdot \alpha$. Para todo ciclo $(a_1 \cdots a_k)$ na decomposição de ciclos disjuntos de α há um ciclo correspondente $(\gamma a_1 \cdots \gamma a_k)$ na decomposição de ciclos disjuntos de $\gamma \cdot \alpha$. Logo, as órbitas das permutações α e $\beta = \gamma \cdot \alpha$ têm o mesmo tamanho. Portanto $\beta = \gamma \cdot \alpha$ e α têm a mesma estrutura de ciclo.
2. Sejam α e β duas permutações com a mesma estrutura de ciclo. Tomemos uma permutação γ que, para cada par de ciclos (um ciclo em α e outro em β) com o mesmo tamanho, mapeia cada elemento de um ciclo $(a_1 \cdots a_k)$ de α em um elemento fixado em outro par de ciclo $(b_1 \cdots b_k)$ de β tal que $\gamma(\alpha^l a_1) = \beta^l b_1$ para qualquer inteiro l . Então para $\alpha = \alpha_1 \cdots \alpha_r$ e $\beta = \beta_1 \cdots \beta_r$, onde α_i e β_i são ciclos de mesmo tamanho, para todo i , temos:

$$\beta_i = (\gamma a_1 \cdots \gamma a_k),$$

onde $k = |\text{orb}(\alpha_i, E)|$ e $\alpha_i = (a_1 \cdots a_k)$ para $1 \leq i \leq r$.

Para $l = 0$ temos $\gamma a_1 = b_1$, então $\gamma \alpha^l a_1 = \beta^l \gamma a_1$. Qualquer $x \in E$ tal que x está no mesmo ciclo de a_1 , pode ser escrito como $x = \alpha^l a_1$, então

$$\beta \gamma x = \beta \gamma \alpha^l a_1 = \beta \beta^l \gamma a_1 = \beta^{l+1} \gamma a_1 = \gamma \alpha^{l+1} a_1 = \gamma \alpha \alpha^l a_1 = \gamma \alpha x.$$

Então $\beta = \gamma \cdot \alpha$ porque

$$\beta \gamma x = \gamma \cdot \alpha(\gamma x) = \gamma \alpha \gamma^{-1} \gamma x = \gamma \alpha \cdot \gamma x = \gamma \cdot \alpha \gamma x$$

para qualquer $x \in E$. Portanto β é conjugado de α por γ .

3. Como $\alpha = \iota \cdot \alpha$ então α é conjugado de α e a relação é reflexiva. Além disso, se α é um conjugado de β então $\alpha = \gamma \cdot \beta$. Logo $\beta = \gamma^{-1} \cdot \alpha$ e então β é um conjugado de α . Considere as permutações α , β e τ tal que α é um conjugado de τ e τ é um conjugado de β . Logo $\alpha = \rho \cdot \tau$ e $\tau = \mu \cdot \beta$ e por esta razão α é um conjugado de β , pois:

$$\alpha = \rho \cdot (\mu \cdot \beta) = (\rho \mu) \cdot \beta.$$

Portanto a relação “é um conjugado de” é uma relação de equivalência. □

Lema 3.1.3 *Toda permutação em $S(E)$ tem uma decomposição em 2-ciclos.*

Prova Como qualquer permutação pode ser representada como um produto de ciclos disjuntos, podemos encontrar uma decomposição em 2-ciclos para cada um dos ciclos. Assim, uma permutação arbitrária pode ser representada por uma decomposição em 2-ciclos. Então, o ciclo $(a_1 a_2 \cdots a_r)$ pode ser escrito como:

$$(a_1 a_2 \cdots a_r) = (a_1 a_r)(a_1 a_{r-1}) \cdots (a_1 a_3)(a_1 a_2)$$

□

A **norma** de π , denotada por $\|\pi\|$ é o número mínimo de 2-ciclos em uma decomposição em 2-ciclos de π . Por exemplo, tomando $\pi = (1\ 6\ 4)(3\ 5\ 2)(0\ 7)$ e $\sigma = (1\ 5\ 3)(0\ 2\ 6\ 4\ 7)$. O número mínimo de decomposição em 2-ciclos de π e σ é:

$$\pi = (1\ 6\ 4)(3\ 5\ 2)(0\ 7) = (1\ 4)(1\ 6)(3\ 2)(3\ 5)(0\ 7), \|\pi\| = 5$$

$$\sigma = (1\ 5\ 3)(0\ 2\ 6\ 4\ 7) = (1\ 3)(1\ 5)(0\ 7)(0\ 4)(0\ 6)(0\ 2), \|\sigma\| = 6$$

Lema 3.1.4 *Para todas permutações α e β sobre um conjunto E , temos:*

1. $\|\alpha\| = 0$ se e somente se $\alpha = \iota$
2. $\|\alpha^{-1}\| = \|\alpha\|$
3. $\|\beta \cdot \alpha\| = \|\alpha\|$
4. $\|\alpha\beta\| \leq \|\alpha\| + \|\beta\|$
5. $\|\alpha\beta\| = \|\beta\alpha\|$

Prova

1. Como a forma mínima de expressar uma permutação α através do produto de 2-ciclos não envolve qualquer 2-ciclos então a permutação deve ser a identidade.
2. Seja $\alpha_1 \cdots \alpha_k$ o produto mínimo de dois ciclos que representa α então

$$\alpha^{-1} = (\alpha_1 \cdots \alpha_k)^{-1} = \alpha_k^{-1} \cdots \alpha_1^{-1} = \alpha_k \cdots \alpha_1.$$

Consequentemente, o produto $\alpha_k \cdots \alpha_1$ é uma decomposição em 2-ciclos de α^{-1} e $\|\alpha^{-1}\| \leq \|\alpha\|$. De forma simétrica temos $\|\alpha^{-1}\| \geq \|\alpha\|$. Portanto, $\|\alpha^{-1}\| = \|\alpha\|$.

3. Dado $\alpha = \alpha_1 \cdots \alpha_k$, onde $\alpha_1 \cdots \alpha_k$ é a decomposição em 2-ciclos de α , então

$$\beta \cdot \alpha = (\beta \cdot \alpha_1) \cdots (\beta \cdot \alpha_k)$$

e $\beta \cdot \alpha_i$, onde $1 \leq i \leq k$, é um 2-ciclo, pois a conjugação preserva a estrutura de ciclos. Daí, $\|\beta \cdot \alpha\| \leq \|\alpha\|$. Como esse resultado é válido para permutações arbitrárias, temos $\|\beta^{-1} \cdot (\beta \cdot \alpha)\| \leq \|\beta \cdot \alpha\|$. Porém $\beta^{-1} \cdot (\beta \cdot \alpha) = \alpha$, então $\|\alpha\| \leq \|\beta \cdot \alpha\|$. Finalmente, $\|\alpha\| = \|\beta \cdot \alpha\|$.

4. Sejam $\alpha = \alpha_1 \cdots \alpha_k$ e $\beta = \beta_1 \cdots \beta_l$ as decomposições mínimas em 2-ciclos das permutações α e β . Então

$$\alpha\beta = \alpha_1 \cdots \alpha_k \beta_1 \cdots \beta_l$$

$$\text{e } \|\alpha\beta\| \leq k + l = \|\alpha\| + \|\beta\|.$$

5. Pelo item 3, temos $\|\beta\alpha\| = \|\alpha \cdot (\beta\alpha)\|$, e $\alpha \cdot (\beta\alpha) = \alpha\beta$. Daí, $\|\beta\alpha\| = \|\alpha\beta\|$

□

Lema 3.1.5 *Dada uma permutação π sobre E tal que $\pi_1 \cdots \pi_k$ é uma decomposição em 2-ciclos de π , então $k \geq |E| - o(\pi, E)$.*

Prova Provaremos usando indução em k . Para $k = 0$, a permutação π é ι (identidade), então $o(\pi, E) = |E|$ e $|E| - o(\pi, E) = 0$. Suponhamos então que a hipótese seja verdadeira para qualquer k' tal que $k' < k$. Seja $\pi' = \pi_1 \cdots \pi_{k-1}$ para $k > 0$. Por hipótese de indução $k - 1 \geq |E| - o(\pi', E)$. Mas, $o(\pi, E) = o(\pi' \pi_k, E) \geq o(\pi', E) - 1$. Então

$$k = (k - 1) + 1 \geq |E| - o(\pi', E) + 1 = |E| - (o(\pi', E) - 1) \geq |E| - o(\pi, E).$$

□

Corolário 3.1.1 *Dada uma permutação π sobre E então $\|\pi\| \geq |E| - o(\pi, E)$.*

Lema 3.1.6 *Dada uma permutação π sobre E , então $\|\pi\| \leq |E| - o(\pi, E)$*

Prova Seja $\pi_1 \cdots \pi_k$ uma decomposição em 2-ciclos de π . Utilizando o Lema 3.1.3 podemos escrever um ciclo π_i como um produto de $l_i - 1$ ciclos na sua decomposição em 2-ciclos, onde l_i é o tamanho do ciclo π_i . Então α pode ser escrito como um produto de $\sum_{i=1}^k (l_i - 1)$ de 2-ciclos. Daí, temos:

$$\sum_{i=1}^k (l_i - 1) = \sum_{i=1}^k l_i - k = |E| - o(\pi, E).$$

Portanto temos $\|\pi\| \leq |E| - o(\pi, E)$.

□

Teorema 3.1.2 *Dada uma permutação π sobre E , então $\|\pi\| = |E| - o(\pi, E)$.*

Prova A demonstração segue do Corolário 3.1.1 e do Lema 3.1.6.

□

Uma **permutação α divide uma permutação β** , ambas permutações sobre E , que é denotado por $\alpha|\beta$, quando $\|\beta\alpha^{-1}\| = \|\beta\| - \|\alpha\|$. Por exemplo, considere as seguintes permutações $\alpha = (0 \ 2 \ 1)$, $\beta = (0 \ 4 \ 2 \ 1 \ 3)$. Então $\alpha|\beta$, pois $\|\beta\alpha^{-1}\| = 2$, $\|\beta\| = 4$ e $\|\alpha\| = 2$.

Teorema 3.1.3 *A relação de divisibilidade é uma relação de ordem, isto é:*

1. $\alpha|\alpha$
2. Se $\alpha|\beta$ e $\beta|\alpha$ então $\alpha = \beta$
3. Se $\alpha|\beta$ e $\beta|\gamma$ então $\alpha|\gamma$

Prova

1. A propriedade reflexiva segue diretamente de $\|\alpha\alpha^{-1}\| = \|\iota\| = 0 = \|\alpha\| - \|\alpha\|$.

2. Sejam α e β duas permutações sobre E tal que $\alpha|\beta$ e $\beta|\alpha$. Então

$$\|\beta\alpha^{-1}\| = \|\beta\| - \|\alpha\| \text{ e } \|\alpha\beta^{-1}\| = \|\alpha\| - \|\beta\|.$$

Mas, como $\alpha\beta^{-1} = (\beta\alpha^{-1})^{-1}$, então $\|\beta\alpha^{-1}\| = \|\alpha\beta^{-1}\|$. Daí, $\|\alpha\| = \|\beta\|$ e $\|\alpha\beta^{-1}\| = 0$, isto é, $\alpha\beta^{-1} = \iota$. Então $\alpha = \beta$.

3. Sejam α , β e γ permutações sobre E tal que $\alpha|\beta$ e $\beta|\gamma$. Então

$$\|\beta\alpha^{-1}\| = \|\beta\| - \|\alpha\| \text{ e } \|\gamma\beta^{-1}\| = \|\gamma\| - \|\beta\|$$

e somando estas equações temos:

$$\|\beta\alpha^{-1}\| + \|\gamma\beta^{-1}\| = \|\gamma\| - \|\alpha\|.$$

Verificamos o relacionamento entre $\|\gamma\alpha^{-1}\|$ e $\|\gamma\| - \|\alpha\|$. Então,

$$\|\gamma\alpha^{-1}\| = \|\gamma\beta^{-1}\beta\alpha^{-1}\| \leq \|\gamma\beta^{-1}\| + \|\beta\alpha^{-1}\| \leq \|\gamma\| - \|\alpha\|,$$

e

$$\|\gamma\| = \|\gamma\alpha^{-1}\alpha\| \leq \|\gamma\alpha^{-1}\| + \|\alpha\|$$

implica em $\|\gamma\alpha^{-1}\| \geq \|\gamma\| - \|\alpha\|$. Daí, $\|\gamma\alpha^{-1}\| = \|\gamma\| - \|\alpha\|$ e $\alpha|\gamma$.

□

Lema 3.1.7 *Dadas as permutações α e β sobre E , se $\alpha|\beta$ então $o(\beta\alpha^{-1}, E) = o(\beta, E) + \|\alpha\|$.*

Prova

$$\begin{aligned} o(\beta\alpha^{-1}, E) &= |E| - \|\beta\alpha^{-1}\| = |E| - (\|\beta\| - \|\alpha\|) = \\ &= |E| - (|E| - o(\beta, E)) + \|\alpha\| = o(\beta, E) + \|\alpha\| \end{aligned}$$

Daí, $o(\beta\alpha^{-1}, E) = o(\beta, E) + \|\alpha\|$. □

Lema 3.1.8 *Dados o 2-ciclo $\alpha = (a b)$ e um ciclo β ambos sobre E onde $\beta a = b$, temos $\alpha\beta a = a$ e $\alpha\beta\beta^{-1}a = b$, enquanto $\alpha\beta x = \beta x$ para cada $x \in E$ distinto de a e $\beta^{-1}a$.*

Prova Se $\beta a = b$ então $\alpha\beta a = \alpha b = a$, e $\alpha\beta\beta^{-1}a = \alpha a = b$. Além disso, para qualquer $x \in E$, se $x \neq \beta^{-1}a$ então $\beta x \neq a$, enquanto se $x \neq a$ então $\beta x \neq b$. Daí, como $\beta x \neq a$ e $\beta x \neq b$ então $\alpha\beta x = \beta x$. □

Proposição 3.1.3 *Dados um 2-ciclo $\alpha = (a b)$ e uma permutação β ambos sobre E , temos:*

1. *Os elementos a e b aparecem no mesmo ciclo γ de β se e somente se γ é dividido nos ciclos γ_a e γ_b , com $a \in \text{Supp}(\gamma_a)$ a menos que $\gamma_a = \iota$, e $b \in \text{Supp}(\gamma_b)$ a menos que $\gamma_b = \iota$, enquanto todos os outros ciclos de β são preservados em $\alpha\beta$.*

2. Os elementos a e b aparecem em ciclos distintos γ_a e γ_b de β se e somente se γ_a e γ_b são unidos no mesmo ciclo γ , enquanto todos os outros ciclos de β são preservados em $\alpha\beta$.

Proposição 3.1.4 Dado um 3-ciclo $\alpha = (a\ b\ c)$ e uma permutação β ambos sobre E , então:

1. Os elementos a, b e c aparecem em três ciclos distintos de β , $(a\cdots)(b\cdots)(c\cdots)$, se e somente se esses três ciclos são unidos em um único ciclo $(a\cdots b\cdots c\cdots)$ de $\alpha\beta$.
2. Dois elementos de a, b pertencem ao mesmo ciclo de β e c pertence a um ciclo distinto de β , isto é, $(a\cdots c\cdots)(c\cdots)$ são ciclos de β , se e somente se estes ciclos tornam-se $(a\cdots c\cdots)(b\cdots)$ e $o(\beta, E) = o(\alpha\beta, E)$.
3. Os elementos a, b e c aparecem no mesmo ciclo na ordem $(a\cdots b\cdots c\cdots)$ se e somente se o ciclo torna-se $(a\cdots c\cdots b\cdots)$ em $\alpha\beta$. Caso contrário, o ciclo $(a\cdots c\cdots b\cdots)$ é um ciclo de β , se e somente se este ciclo é dividido em três ciclos $(a\cdots)(b\cdots)(c\cdots)$.

Lema 3.1.9 Sejam $\alpha = (a\ b)$ um 2-ciclo, $\gamma = (a\ b\ c)$ um três ciclo e β uma permutação. Os três definidos sobre E . Então:

1. $a, b \in \text{Supp}(\beta_i)$ para um ciclo β_i de β se e somente se $(a\ b)|\beta$.
2. $a, b, c \in \text{Supp}(\beta_i)$, para um ciclo β_i de β , aparecem nesta ordem em β_i se e somente se $(a\ b\ c)|\beta$.

Prova

1. Se $a, b \in \text{Supp}(\beta_i)$ para um ciclo β_i de β então o ciclo β_i de β é dividido em dois ciclos com Suporte $\{a, \beta a, \dots, \beta^{-1}b\}$ e $b, \beta b, \beta^2 b, \dots, \beta^{-1}a$ em $\alpha\beta$ pela Proposição 3.1.3. Daí, $o(\alpha\beta, E) = o(\beta, E) + 1$. Mas $\|\alpha\beta\| = \|\beta\alpha^{-1}\|$ pelo Lema 3.1.4, então

$$\|\beta\alpha^{-1}\| = |E| - o(\alpha\beta, E) = |E| - (o(\beta, E) + 1) = (|E| - o(\beta, E)) - 1 = \|\beta\| - \|\alpha\|,$$

isto é $\alpha|\beta$.

Por outro lado, suponha que $\alpha|\beta$, onde $\alpha = (a\ b)$. Então, pelo Lema 3.1.7, temos $o(\alpha\beta, E) = o(\beta, E) + 1$, isto é, o número de órbitas é aumentada em mais uma em $\alpha\beta$ em relação a β . Então pela Proposição 3.1.3, os elementos a e b pertencem ao mesmo ciclo de β . Daí, $a, b \in \text{Supp}(\beta_i)$ para um ciclo β_i de β .

2. Se os elementos $a, b, c \in \text{Supp}(\beta_i)$, para um ciclo β_i de β , aparecem nesta ordem em β_i então pela Proposição 3.1.4, temos $o(\beta\alpha^{-1}, E) = o(\beta) + 2$. Além disso, temos $\|\alpha\| = 2$, pois α é um 3-ciclo. Então

$$\|\beta\alpha^{-1}\| = |E| - o(\beta\alpha^{-1}) = |E| - (o(\beta) + 2) = |E| - (|E| - \|\beta\|) - 2 = \|\beta\| - \|\alpha\|.$$

Daí, $\|\beta\alpha^{-1}\| = \|\beta\| - \|\alpha\|$ e $\alpha|\beta$.

Por outro lado, se $\alpha|\beta$ então $o(\beta\alpha^{-1}) = o(\beta) + \|\alpha\|$ (Lema 3.1.7, isto é, α^{-1} é um 3-ciclo tal que $o(\beta\alpha^{-1}) = o(\beta) + 2$). Daí, pela Proposição 3.1.4, os elementos $a, b, c \in \text{Supp}(\beta_i)$ para um ciclo β_i de β aparecem nesta ordem em β_i

□

Lema 3.1.10 *Seja $\alpha = (a\ b)$ um 2-ciclo e β uma permutação, ambos sobre E . Então*

$$\|\beta\alpha\| = \begin{cases} \|\beta\| - 1 & \text{se } \alpha|\beta, \\ \|\beta\| + 1 & \text{se } \alpha \nmid \beta. \end{cases}$$

Prova Se $\alpha|\beta$ então $\|\beta\alpha^{-1}\| = \|\beta\| - \|\alpha\|$, mas $\alpha^{-1} = \alpha$ e $\|\alpha\| = 1$ porque α é um 2-ciclo, então

$$\|\beta\alpha\| = \|\beta\| - \|\alpha\| = \|\beta\| - 1.$$

Por outro lado, se $\alpha \nmid \beta$ então os elementos a e b pertencem a ciclos distintos de β pelo Lema 3.1.9. Daí, pela Proposição 3.1.3, os dois ciclos distintos são unidos em $\beta\alpha$ e temos $o(\beta\alpha) = o(\beta) - 1$. Então

$$\|\beta\alpha\| = |E| - o(\beta\alpha) = |E| - o(\beta) + 1 = \|\beta\| + 1.$$

□

O **tamanho** de um **ciclo** α é o tamanho da sua órbita não trivial ou 1 se α for a identidade. Um k -*ciclo* $\alpha, k > 1$, é chamado um ciclo de uma permutação β , onde α e β são ambas permutações sobre E , quando $\alpha|\beta$ e a órbita não trivial de α pertence ao conjunto de órbitas de β . Por exemplo, a permutação $(1\ 5\ 3)$ é um ciclo de $(1\ 5\ 3)(0\ 2\ 6\ 4\ 7)$. A identidade é um ciclo de qualquer permutação.

Um **ciclo** é **ímpar** se seu tamanho é ímpar, caso contrário é um ciclo par. Uma **permutação** é **ímpar**, se sua norma é ímpar, caso contrário é uma permutação par. Denotamos por $o_{\text{ímpar}}(\pi, E)$ ($o_{\text{par}}(\pi, E)$) o **número de órbitas com tamanhos ímpares** (pares).

Lema 3.1.11 *Seja $\alpha = (a\ b)$ um 2-ciclo e β uma permutação, ambos sobre E . Então:*

1. se $\alpha|\beta$ então $o(\alpha\beta, E) = o(\beta, E) + 1$
2. se $\alpha \nmid \beta$ então $o(\alpha\beta, E) = o(\beta, E) - 1$

Prova

1. Suponha que $\alpha \nmid \beta$ então pelo Lema 3.1.7 temos $o(\beta\alpha^{-1}, E) = o(\beta, E) + \|\alpha\|$. Mas $\alpha\beta = \alpha^{-1} \cdot (\beta\alpha^{-1})$, então $o(\alpha\beta, E) = o(\beta\alpha^{-1}, E)$ pelo Lema 3.1.2. Daí, $o(\alpha\beta, E) = o(\beta, E) + 1$.
2. Como $o(\alpha\beta, E) = o(\beta\alpha^{-1}, E)$ e usando o Lema 3.1.9 então a, b pertencem a ciclos diferentes de β . Estes ciclos diferentes são transformados em um ciclo em $\alpha\beta$ pela Proposição 3.1.3. Daí, $o(\alpha\beta, E) = o(\beta) - 1, E$.

□

Lema 3.1.12 *Sejam $\alpha = (a b c)$ um 3-ciclo, e β uma permutação, ambos sobre E . Então $\alpha|\beta$ se e somente se $o(\beta\alpha^{-1}, E) = o(\beta, E) + 2$*

Prova Se $\alpha|\beta$ então $\|\beta\alpha^{-1}\| = \|\beta\| - \|\alpha\|$. Temos $\|\alpha\| = 2$, $\|\beta\alpha^{-1}\| = |E| - o(\beta\alpha^{-1}, E)$ e $\|\beta\| = |E| - o(\beta, E)$, então substituindo na fórmula anterior temos $|E| - o(\beta\alpha^{-1}, E) = |E| - o(\beta, E) + 2$ então $\|\beta\alpha^{-1}\| = \|\beta\| - 2$, mas como $\|\alpha\| = 2$ então $\|\beta\alpha^{-1}\| = \|\beta\| - \|\alpha\|$. Daí, $\alpha|\beta$. □

3.2 Problema de Ordenação por Transposições

Nesta seção mostraremos como utilizar o formalismo algébrico para modelar conceitos de Bafna e Pevzner [4, 3] para o problema de distância de ordenação por transposições. Este formalismo algébrico foi apresentado por Dias e Meidanis [8], como alternativa ao grafo-de-ciclos, para solucionar problemas de rearranjo de genomas.

Seja $\pi = [x_1 \ x_2 \ \dots \ x_n]$ uma permutação sobre $E = \{x_1, \dots, x_n\}$ modelando um cromossomo.

Tomando $E = \{0, x_1, \dots, x_n\}$, no formalismo algébrico, um genoma é representado como $\pi = (0 \ x_1 \ x_2 \ \dots \ x_n)$. O zero é incluído para representar as seqüências de elementos x_1, x_2, \dots, x_n de π que podem ser mapeadas para as $n!$ permutações na representação tradicional.

Dado um genoma π sobre E , uma **transposição** no formalismo algébrico é a permutação $\tau = (u \ v \ w)$, onde $u, v, w \in E$, tal que $\tau|\pi$. Neste caso dizemos que τ é **aplicável** a π .

Uma transposição no formalismo algébrico é relacionada aos elementos da permutação π e não às posições destes elementos em π . Esta definição é diferente da definição de transposição feita por Bafna e Pevzner no grafo-de-ciclos, onde as posições de três elementos são usadas como parâmetros para a transposição. No formalismo algébrico, para aplicar uma transposição no genoma π , temos que executar o produto $\tau\pi$. Por exemplo:

$$\tau\pi = (4 \ 2 \ 5)(0 \ 1 \ 4 \ 3 \ 2 \ 5) = (0 \ 1 \ 2 \ 4 \ 3 \ 5)$$

Observe que dado um genoma π , seja τ_1, \dots, τ_k a seqüência mínima de transposições tal que

$$\tau_k \cdots \tau_1\pi = \sigma$$

então

$$\tau_k \cdots \tau_1 = \sigma\pi^{-1}$$

e $\tau_k \cdots \tau_1$ é uma decomposição em 3-ciclos de $\sigma\pi^{-1}$.

No restante desta seção, transcrevemos a prova de Mira e Meidanis [16, 15] de um limite inferior para o problema da ordenação por transposições.

Dada uma permutação α sobre E , a **norma-3** de α , denotada por $\|\alpha\|_3$, é o número mínimo de 3-ciclos $\tau_1, \tau_2, \dots, \tau_k$, $k = \|\alpha\|_3$, não necessariamente disjuntos, tal que $\alpha = \tau_1\tau_2 \dots \tau_k$. Temos que $\tau_1, \tau_2, \dots, \tau_k$ é a decomposição mínima em 3-ciclos de α e $\|\alpha\|_3$ é definido somente para permutações α tal que $\|\alpha\|$ é par, isto é, a norma-3 é definida somente para permutações pares.

A razão para esta restrição é a impossibilidade de decompor uma permutação ímpar em 3-ciclos somente, isto é, permutações ímpares podem ser decompostas por um produto de 3-ciclos e um único 2-ciclo. Por exemplo, se tomarmos $E = \{0, 1, 2, 3, 4, 5\}$ temos:

$$\theta = (0\ 5\ 3\ 1\ 2\ 4) = (0\ 5)(5\ 3)(3\ 1)(1\ 2)(2\ 4) = (0\ 5\ 3)(3\ 1\ 2)(2\ 4).$$

Lema 3.2.1 *Dadas as permutações α e β sobre E , temos $\|\alpha\beta\| \equiv \|\alpha\| + \|\beta\| \pmod{2}$.*

Prova Utilizaremos indução em $\|\beta\|$.

Para $\|\beta\| = 0$ temos, $\beta = \iota$ e Conseqüentemente $\|\alpha\beta\| \equiv \|\alpha\| \equiv \|\alpha\| + \|\beta\| \pmod{2}$.

Se $\|\beta\| = 1$, então $\beta = (a\ b)$ e teremos dois sub-casos:

- se $(a\ b) \nmid \alpha$, então $\|\alpha\beta\| = \|\alpha\| + 1$ pela Proposição 3.1.10. Conseqüentemente $\|\alpha\beta\| \equiv \|\alpha\| + 1 \equiv \|\alpha\| + \|\beta\| \pmod{2}$.
- se $(a\ b) \mid \alpha$, então $\|\alpha\beta\| = \|\alpha\| - 1$ pela Proposição 3.1.10. Daí,

$$\begin{aligned} \|\alpha\beta\| &= \|\alpha\| - 1 \equiv (\|\alpha\| \pmod{2} + (-1) \pmod{2}) \\ &\pmod{2} \equiv (\|\alpha\| + 1) \pmod{2} \equiv \|\alpha\| + \|\beta\| \pmod{2}. \end{aligned}$$

Por hipótese de indução, o resultado vale para $\|\beta\| < k$.

Seja $\beta = \beta_1 \beta_2 \dots \beta_k$, onde β_i é um 2-ciclo e k é mínimo. Considere β_1 e $\beta' = \beta_2 \dots \beta_k$. Pela hipótese de indução e $\|\beta'\| < \|\beta\|$ (pela minimalidade da decomposição de 2-ciclos) temos $\|\alpha\beta_1\beta'\| \equiv \|\alpha\beta_1\| + \|\beta'\| \pmod{2}$. Mas $\|\alpha\beta_1\| \equiv \|\alpha\| + 1 \pmod{2}$ como mostramos no caso base. Conseqüentemente $\|\alpha\beta\| \equiv (\|\alpha\| + 1 \pmod{2}) + \|\beta'\| \pmod{2} \equiv \|\alpha\| + (1 + \|\beta'\|) \pmod{2}$. Como $\|\beta\| = 1 + \|\beta'\|$, então temos $\|\alpha\beta\| \equiv \|\alpha\| + \|\beta\| \pmod{2}$. \square

Lema 3.2.2 *Ciclos ímpares são permutações pares e ciclos pares são permutações ímpares.*

Prova Pelo Lema 3.1.3 um r -ciclo $\pi = (a_1 a_2 \cdots a_r)$ pode ser escrito como:

$$\pi = (a_1 a_r)(a_1 a_{r-1}) \cdots (a_1 a_3)(a_1 a_2)$$

que é um produto de $(r - 1)$ 2-ciclos. Através da aplicação do produto de 2-ciclos podemos verificar que o produto anterior de 2-ciclos é certamente uma decomposição em 2-ciclos de π . Temos que mostrar que esta é uma decomposição mínima em 2-ciclos. Pelo Teorema 3.1.2 temos $\|\pi\| = |E| - o(\pi)$. Seja f o número de elementos fixos em π . Então:

$$|E| - o(\pi, E) = |Supp(\pi)| + f - (no(\pi) + f) = |Supp(\pi)| - no(\pi) = r - 1,$$

onde $no(\pi)$ é o número de órbitas não triviais em $\sigma\pi^{-1}$. Como π é um r -ciclo, então o produto de 2-ciclos envolve $\|\pi\|$ e sua decomposição em 2-ciclos de π . Daí, $\|\pi\|$ é ímpar (par) se e somente se r é par (ímpar). Portanto, ciclos ímpares são permutações pares e ciclos pares são permutações ímpares. \square

O lema seguinte mostra que a norma-3 é definida somente para permutações pares.

Lema 3.2.3 *Dada uma permutação π , existe uma decomposição em 3-ciclos de π se e somente se $\|\pi\|$ é par.*

Prova Seja π uma permutação tal que $\|\pi\|$ é par, isto é, $0 \equiv \|\pi\| \pmod{2}$. Para cada par consecutivo de 2-ciclos, α e β , em uma decomposição de 2-ciclos de π , podemos ter 3-ciclo $(a b c)$, de acordo com os três casos:

1. Se $Supp(\alpha) = Supp(\beta)$, isto é, $\alpha = \beta = (a b)$, então $\alpha\beta = \beta\alpha = \iota$. Como um resultado, podemos eliminar o par repetido dos ciclos na permutação π e analisamos o conjunto de pares de 2-ciclos restantes.
2. Se $|Supp(\alpha) \cap Supp(\beta)| = 1$, isto é, $\alpha = (a b)$ e $\beta = (b c)$ então construímos um 3-ciclo $(a b c)$.
3. Se $|Supp(\alpha) \cap Supp(\beta)| = \emptyset$, isto é, $\alpha = (a b)$, $\beta = (c d)$ e $a \neq b \neq c \neq d$, então construímos dois 3-ciclos $(a b c)(b c d)$.

Observe que o número de 3-ciclos é $\|\pi\|/2$ se existe somente pares de 2-ciclos do caso 2. Claramente, se aplicar as transformações de pares de 2-ciclos em 3-ciclos como acima, o resultado será a decomposição da permutação original em um produto de 3-ciclos. Agora, considere a permutação π tal que há uma decomposição mínima em 3-ciclos de π : $\pi = \alpha_1\alpha_2 \cdots \alpha_k$. Daí, $\|\pi\| = \|\alpha_1\alpha_2 \cdots \alpha_k\|$ e pela generalização do Lema 3.2.1, temos:

$$\|\alpha_1\alpha_2 \cdots \alpha_k\| \equiv \|\alpha_1\| + \|\alpha_2\| + \cdots + \|\alpha_k\| \pmod{2}$$

mas a norma de um 3-ciclo é 2, então $\|\alpha_1\alpha_2 \cdots \alpha_k\| \equiv 0 \pmod{2}$, isto é, a norma de π é par. Portanto $\|\pi\|_3$ é definida somente para permutações pares. \square

Proposição 3.2.1 *Dada a permutação par α e a permutação β , ambas sobre E , temos:*

1. $\|\alpha^{-1}\|_3 = \|\alpha\|_3$
2. $\|\beta \cdot \alpha\|_3 = \|\alpha\|_3$
3. $\|\alpha\beta\|_3 \leq \|\alpha\|_3 + \|\beta\|_3$ onde β é uma permutação par
4. $\|\alpha\beta\|_3 = \|\beta\alpha\|_3$ onde β é uma permutação par

Teorema 3.2.1 (Limite Inferior) *Dados os genomas π e σ sobre E , temos*

$$d_t(\pi, \sigma) \geq \|\sigma\pi^{-1}\|_3.$$

Prova Tomamos uma seqüência de transposições $\tau_1, \tau_2, \dots, \tau_k$ transformando uma permutação inicial π em uma permutação σ , isto é, $\tau_k \dots \tau_1 \pi = \sigma$, e τ_i é aplicável a $\tau_{i-1} \dots \tau_1 \pi$ para $1 \leq i \leq k$. Reescrevendo a equação, temos $\tau_k \tau_{k-1} \dots \tau_1 = \sigma\pi^{-1}$. Conseqüentemente, o produto $\tau_k \dots \tau_1$ é uma decomposição em 3-ciclos de $\sigma\pi^{-1}$. Como π e σ têm a mesma paridade pela definição de genomas então $\sigma\pi^{-1}$ é uma permutação par pelo Lema 3.2.1. Então há uma decomposição mínima em 3-ciclos de $\sigma\pi^{-1}$ tal que $\|\sigma\pi^{-1}\|_3 \leq k$. Conseqüentemente $d_t(\pi, \sigma) \geq \|\sigma\pi^{-1}\|_3$ é válido para uma seqüência mínima de transposições que transforma π em σ . \square

O limite inferior do Teorema 3.2 é equivalente ao limite inferior de Bafna e Pevzner [4], $d_t(\pi, \sigma) \geq \frac{n - o_{\text{impar}}(\sigma\pi^{-1})}{2}$.

Lema 3.2.4 *Se π é uma permutação par sobre E , tal que $\pi = \pi_1 \dots \pi_k$ é uma decomposição em 3-ciclos de π , onde π_i , $1 \leq i \leq k$, é um 3-ciclo, então $k \geq \frac{n - o_{\text{impar}}(\pi)}{2}$ para $n = |E|$.*

Prova Utilizaremos indução em k . Se $k = 0$, então $\pi = \iota$ e $o_{\text{impar}}(\pi) = n$. Conseqüentemente temos $0 \geq \frac{n-n}{2} = 0$.

Por hipótese de indução, k' é tal que $0 \leq k' < k$ e $\pi = \pi_1 \dots \pi_{k'}$, então $k' \geq \frac{n - o_{\text{impar}}(\pi)}{2}$.

Por hipótese de indução, $k-1 \geq \frac{n - o_{\text{impar}}(\pi')}{2}$ é verdade para $\pi' = \pi_1 \dots \pi_{k-1}$. Mas $o_{\text{impar}}(\pi) = o_{\text{impar}}(\pi' \pi_k) \geq o_{\text{impar}}(\pi') - 2$. Daí,

$$\begin{aligned} k = (k-1) + 1 &\geq \frac{n - o_{\text{impar}}(\pi')}{2} + 1 \\ &= \frac{n - (o_{\text{impar}}(\pi') - 2)}{2} \\ &\geq \frac{n - o_{\text{impar}}(\pi)}{2} \end{aligned}$$

Conseqüentemente $k \geq \frac{n - o_{\text{impar}}(\pi)}{2}$ para $n = |E|$. \square

Lema 3.2.5 *Se π é uma permutação sobre E tal que $\|\pi\|$ é par, então para $n = |E|$,*

$$\|\pi\|_3 \leq \frac{n - o_{\text{impar}}(\pi)}{2}.$$

Prova Considere um reordenamento dos ciclos da decomposição em ciclos disjuntos de π , tal que o tamanho de ciclos ímpares são agrupados no início da permutação, isto é, temos $\pi = \alpha' \alpha''$, onde α' é o produto de ciclos ímpares de π e α'' é o produto de ciclos pares de π . Observe que este reordenamento é possível devido a propriedade comutativa dos ciclos disjuntos de uma permutação:

$$\pi = \alpha_1 \alpha_2 \cdots \alpha_l \alpha_{l+1} \alpha_{l+2} \cdots \alpha_m$$

Mostraremos que $\|\pi\|_3 \leq \frac{n - o_{\text{ímpar}}(\pi)}{2}$. Para isto, construímos uma decomposição em 3-ciclos de π através da decomposição em 2-ciclos de α' e α'' .

Seja $\alpha' = \alpha_1 \alpha_2 \cdots \alpha_l$, onde α_i , para $i = 1, \dots, l$, é um ciclo ímpar de π . Denotamos $n' = |\text{Supp}(\alpha')|$ e $n'' = |\text{Supp}(\alpha'')|$. Observe que $\|\alpha'\| = n' - o_{\text{ímpar}}(\pi)$ e que $\|\alpha'\|$ é par, porque a norma de um ciclo ímpar é par. Considere uma decomposição mínima em 2-ciclos de α' . Então é possível agrupar os 2-ciclos de α' em pares $\alpha_i \alpha_j$, tal que $|\text{Supp}(\alpha_i) \cap \text{Supp}(\alpha_j)| = 1$, isto é, os 2-ciclos são da forma $(a \ b)(b \ c)$ e para cada par $(a \ b)(b \ c)$ há um 3-ciclo correspondente $(a \ b \ c)$. Daí, $\|\alpha'\|_3 \leq \frac{n' - o_{\text{ímpar}}(\pi)}{2}$.

Considere agora uma decomposição mínima em 2-ciclos de α'' . Como a norma de um ciclo par é ímpar e $\|\pi\|$ é par, então o número de ciclos de α'' $o_{\text{par}}(\pi)$ é par. Seja $L(\alpha_i)$ o tamanho do ciclo α_i , para $i = l+1, \dots, m$. Então a decomposição em 2-ciclos de α é tal que:

$$\|\alpha''\| = \sum_{i=l+1}^m (L(\alpha_i) - 1) = \left(\sum_{i=l+1}^m L(\alpha_i) \right) - (m - l) = n'' - o_{\text{par}}(\pi)$$

Observe que há dois pares de 2-ciclos $\alpha_i \alpha_j$, tal que $|\text{Supp}(\alpha_i) \cap \text{Supp}(\alpha_j)| = 0$ e pares $\alpha_r \alpha_s$ tal que $|\text{Supp}(\alpha_r) \cap \text{Supp}(\alpha_s)| = 1$, porque a norma de um ciclo par é ímpar. Chamamos o primeiro tipo de par de 2-ciclos de pares disjuntos, enquanto o segundo tipo é chamando de pares não disjuntos. Podemos construir um 3-ciclo a partir de um par não disjunto da mesma forma que fizemos com pares de 2-ciclos $(a \ b)(b \ c)$ em α' . Por outro lado, um par disjunto $(a \ b)(c \ d)$ gera dois 3-ciclos $(a \ b \ c)(b \ c \ d)$. Daí, cada par de 2-ciclos contribui com um 3-ciclo para a decomposição em 3-ciclos de α'' mais um número x de 3-ciclos, onde x é o número de pares disjuntos na decomposição em 2-ciclos de π . Então:

$$\|\alpha''\|_3 = \frac{n'' - o_{\text{par}}(\pi)}{2} + x$$

Como os dois ciclos disjuntos são tomados em pares, verificamos que:

$$\left\lceil \frac{o_{\text{par}}(\alpha'') - 1}{2} \right\rceil \leq x \leq o_{\text{par}}(\alpha'') - 1$$

De fato, temos $x = \left\lceil \frac{o_{\text{par}}(\alpha'') - 1}{2} \right\rceil$, porque para cada par disjunto combinado é gerado dois 3-ciclos, uma possibilidade distinta, pares disjuntos não serão combinadas pela paridade (ímpar) do número de 2-ciclos de α'' . Mas:

$$\|\alpha''\|_3 = \frac{n'' - o_{\text{par}}(\pi)}{2} + \left\lceil \frac{o_{\text{par}}(\alpha'') - 1}{2} \right\rceil = \frac{n'' - o_{\text{par}}(\pi)}{2} + \left(\frac{o_{\text{par}}(\pi) - 2}{2} + 1 \right) = \frac{n''}{2}$$

Como α' e α'' são permutações pares e pela Proposição 3.2.1, então:

$$\|\pi\|_3 \leq \|\alpha'\|_3 + \|\alpha''\|_3 \leq \frac{n' - o_{\text{impar}}(\pi)}{2} + \frac{n''}{2} \leq \frac{n - o_{\text{impar}}(\pi)}{2}$$

□

Os dois últimos lemas são usados para provar as seguintes fórmulas para a norma-3 de uma permutação π .

Teorema 3.2.2 *Dada uma permutação π sobre E , temos $\|\pi\|_3 = \frac{(n - o_{\text{impar}}(\pi))}{2}$.*

Um corolário imediato do teorema acima é uma apresentação equivalente do limite inferior de Bafna e Pevzner [4] para o problema da ordenação por transposições.

Teorema 3.2.3 *Dados os genoma π e σ sobre E , temos*

$$d_t(\pi, \sigma) \geq \frac{(n - o_{\text{impar}}(\sigma\pi^{-1}))}{2}.$$

Capítulo 4

Algoritmos de aproximação utilizando o formalismo algébrico

Neste capítulo, utilizando o formalismo algébrico de Mira, Dias e Meidanis [13, 15], propomos dois algoritmos de aproximação. O algoritmo de razão 2.0 foi elaborado em conjunto com Mira. O algoritmo de razão 1.5 foi inspirado no algoritmo de razão 1.5 de Bafna e Pevzner [3, 4].

Na Seção 4.1, descrevemos o algoritmo de razão 2.0. Na Seção 4.2, propomos o algoritmo de razão 1.5. Em ambas as seções, inicialmente definimos conceitos básicos e resultados que provam a correção dos algoritmos propostos.

4.1 Algoritmo de razão 2.0

Para uma permutação α sobre E e dois elementos $v, w \in E$, dizemos que w **segue** v para o par (u, α) , $u \in E$, denotado por $v \rightarrow_{u, \alpha} w$, quando $v = \alpha^{k_1} u$, $w = \alpha^{k_2} u$, e $0 < k_1 < k_2 < |\text{orb}(u, \alpha)|$.

Proposição 4.1.1 *Dados um 3-ciclo $\alpha = (u v w)$ e uma permutação β sobre E , então temos:*

1. $o(\alpha\beta, E) = o(\beta, E)$ e $w \rightarrow_{u, \alpha\beta} v$ se e somente se $v \rightarrow_{u, \beta} w$.
2. $o(\alpha\beta, E) = o(\beta, E)$, $u, v \notin \text{orb}(w, \alpha\beta)$ e $u \in \text{orb}(v, \alpha\beta)$ ou $o(\beta\alpha, E) = o(\beta, E)$, $u, w \notin \text{orb}(v, \beta\alpha)$, e $u \in \text{orb}(w, \beta\alpha)$ se e somente se $v, w \notin \text{orb}(u, \beta)$ e $v \in \text{orb}(w, \beta)$.
3. $o(\alpha\beta, E) = o(\beta, E) + 2$ e u, v, w pertencem a órbitas distintas em $\alpha\beta$ se e somente se $w \rightarrow_{u, \beta} v$.

Prova

1. Temos $\alpha\beta = (u \cdots w \cdots v \cdots)$, $u, v \in \text{orb}(w, \alpha\beta)$. Então $\alpha^{-1}(\alpha\beta) = \iota\beta = \beta$, ou seja,

$$\alpha^{-1}(\alpha\beta) = (w v u)(u \cdots w \cdots v \cdots) = (u \cdots v \cdots w \cdots) = \beta.$$

Daí, $v \rightarrow_{u,\beta} w$. Por outro lado, se $v \rightarrow_{u,\beta} w$, então $\beta = (u \cdots v \cdots w \cdots)$. Portanto,

$$\alpha\beta = (u v w)(u \cdots v \cdots w \cdots) = (u \cdots w \cdots v \cdots).$$

Daí, $u, w \in orb(v, \alpha\beta)$, $w \rightarrow_{u,\alpha\beta} v$ e $o(\alpha\beta, E) = o(\beta, E) = 1$.

2. Temos $\alpha\beta = (u \cdots v \cdots)(w \cdots)$, pois $u, v \notin orb(w, \alpha\beta)$ e $u \in orb(v, \alpha\beta)$, então $\alpha^{-1}(\alpha\beta) = \iota\beta = \beta$, ou seja,

$$\alpha^{-1}(\alpha\beta) = (w v u)(u \cdots v \cdots)(w \cdots) = (w \cdots v \cdots)(u \cdots) = \beta.$$

Daí, $v, w \notin orb(u, \beta)$ e $v \in orb(w, \beta)$. Por outro lado temos $\beta = (v \cdots w \cdots)(u \cdots)$, pois $v, w \notin orb(u, \beta)$ e $v \in orb(w, \beta)$. Portanto,

$$\alpha\beta = (u v w)(v \cdots w \cdots)(u \cdots) = (u \cdots v \cdots)(w \cdots).$$

Daí, $u \in orb(v, \alpha\beta)$ e $u, v \notin orb(w, \alpha\beta)$. Além disso, $o(\alpha\beta, E) = o(\beta, E) = 2$. A outra prova é análoga.

3. Temos $\alpha\beta = (u \cdots)(v \cdots)(w \cdots)$, pois u, v, w pertencem a órbitas distintas em $\alpha\beta$. Então, $\alpha^{-1}(\alpha\beta) = \iota\beta = \beta$, ou seja,

$$\alpha^{-1}(\alpha\beta) = (w v u)(u \cdots)(v \cdots)(w \cdots) = (u \cdots w \cdots v \cdots) = \beta.$$

Temos que $o(\alpha\beta, E) = o(\beta, E) + 2$. Daí, $u, w \in orb(v, \beta)$ e $w \rightarrow_{u,\beta} v$.

Por outro lado, temos $\beta = (u \cdots w \cdots v \cdots)$, pois $w \rightarrow_{u,\beta} v$. Portanto,

$$\alpha\beta = (u v w)(u \cdots w \cdots v \cdots) = (u \cdots)(v \cdots)(w \cdots).$$

Daí, temos que $o(\alpha\beta, E) = o(\beta, E) + 2$ e u, v, w pertencem a órbitas distintas em $\alpha\beta$.

□

Lema 4.1.1 *Dado um 3-ciclo $\alpha = (u v w)$ e uma permutação β sobre E , temos $\alpha|\beta$ se e somente se $v \rightarrow_{u,\beta} w$.*

Prova Se $\alpha|\beta$ então $\|\beta\alpha^{-1}\| = \|\beta\| - \|\alpha\|$. Usando a fórmula $\|\alpha\| = |E| - o(\alpha, E)$ e tomando $\|\alpha\| = 2$ temos que $\|\beta\alpha^{-1}\| = |E| - o(\beta\alpha^{-1}, E)$ então $|E| - o(\beta\alpha^{-1}, E) = |E| - o(\beta, E) - 2$. Então, $o(\beta\alpha^{-1}, E) = o(\beta, E) + 2$. Pela Proposição 4.1.1, se u, v, w pertencem a órbitas distintas em $\beta\alpha^{-1}$, $v \rightarrow_{u,\beta} w$.

Por outro lado temos $v \rightarrow_{u,\beta} w$ então $\beta = (u \cdots v \cdots w \cdots)$. Daí,

$$\beta\alpha^{-1} = (u \cdots v \cdots w \cdots)(w v u) = (u \cdots)(v \cdots)(w \cdots)$$

Então temos $o(\beta\alpha^{-1}, E) = o(\beta, E) + 2$ e usando a fórmula $\|\alpha\| = |E| - o(\alpha, E)$ e sabendo que $\|\alpha\| = 2$ temos que

$$|E| - \|\beta\alpha^{-1}\| = |E| - \|\beta\| + \|\alpha\| \Rightarrow \|\beta\alpha^{-1}\| = \|\beta\| - \|\alpha\|.$$

Daí, $\alpha|\beta$.

□

Dadas duas permutações π e σ sobre E , uma transposição τ aplicável a π é um x -**movimento** quando $o(\sigma\pi^{-1}\tau^{-1}, E) = o(\sigma\pi^{-1}, E) + x$.

Lema 4.1.2 *Se π e σ são duas permutações sobre E , uma transposição τ aplicável a π é um 2-movimento se e somente se $\tau|\sigma\pi^{-1}$.*

Prova Se τ é um 2-movimento então por definição temos $o(\sigma\pi^{-1}\tau^{-1}, E) = o(\sigma\pi^{-1}, E) + 2$. Usando a fórmula $\|\alpha\| = |E| - o(\alpha)$ para uma permutação α , temos $|E| - \|\sigma\pi^{-1}\tau^{-1}\| = |E| - \|\sigma\pi^{-1}\| + 2$. Conseqüentemente, temos $\|\sigma\pi^{-1}\tau^{-1}\| = \|\sigma\pi^{-1}\| - 2$. Como $\|\tau\| = 2$ para qualquer 3-ciclo τ , então $\|\sigma\pi^{-1}\tau^{-1}\| = \|\sigma\pi^{-1}\| - \|\tau\|$. Conseqüentemente, por definição de divisibilidade, temos $\tau|\sigma\pi^{-1}$.

Inversamente, se $\tau|\sigma\pi^{-1}$ então $\|\sigma\pi^{-1}\tau^{-1}\| = \|\sigma\pi^{-1}\| - \|\tau\|$. Usando $\|\alpha\| = |E| - o(\alpha)$ temos $|E| - o(\sigma\pi^{-1}\tau^{-1}) = |E| - o(\sigma\pi^{-1}) - 2$. Conseqüentemente, temos $o(\sigma\pi^{-1}\tau^{-1}, E) = o(\sigma\pi^{-1}, E) + 2$.

Assim uma transposição τ aplicável a π é um 2-movimento se e somente se $\tau|\sigma\pi^{-1}$. \square

Lema 4.1.3 *Se π e σ são duas permutações sobre E , uma transposição $\tau = (u v w)$, $u, v, w \in E$, aplicável a π é um 0-movimento se e somente se*

- $\tau \nmid \sigma\pi^{-1}$ ou
- $v, w \notin orb(u, \sigma\pi^{-1})$, e $v \in orb(w, \sigma\pi^{-1})$.

Prova Se τ é um 0-movimento aplicável a π então por definição $o(\sigma\pi^{-1}\tau^{-1}, E) = o(\sigma\pi^{-1}, E)$. Também temos $\tau \nmid \sigma\pi^{-1}$, pois τ é um 0-movimento. Temos dois casos:

1. Se $v \rightarrow_{u, \sigma\pi^{-1}\tau^{-1}} w$, então pela Proposição 4.1.1 temos $w \rightarrow_{u, \sigma\pi^{-1}} v$. Conseqüentemente pelo Lema 4.1.1 temos $(u w v)|\sigma\pi^{-1}$.
2. Se $u, w \notin orb(v, \sigma\pi^{-1}\tau^{-1})$, e $u \in orb(w, \sigma\pi^{-1}\tau^{-1})$, então $u, v \notin orb(w, \sigma\pi^{-1})$ e $u \in orb(v, \sigma\pi^{-1})$.

Conseqüentemente temos $\tau \nmid \sigma\pi^{-1}$ ou $v, w \notin orb(u, \sigma\pi^{-1})$, e $v \in orb(w, \sigma\pi^{-1})$.

Inversamente, se $\tau \nmid \sigma\pi^{-1}$ então τ é um 0-movimento ou um -2 -movimento aplicável a π . Temos dois casos:

1. Se $\tau^{-1}|\sigma\pi^{-1}$ então $w \rightarrow_{u, \sigma\pi^{-1}} v$ para $(u, \sigma\pi^{-1})$. Assim, pela Proposição 4.1.1, temos $o(\tau^{-1}\sigma\pi^{-1}, E) = o(\sigma\pi^{-1}, E)$. Como $\|\sigma\pi^{-1}\tau^{-1}\| = \|\tau^{-1}\sigma\pi^{-1}\|$ então $o(\sigma\pi^{-1}\tau^{-1}, E) = o(\tau^{-1}\sigma\pi^{-1}, E)$. Então, $o(\sigma\pi^{-1}\tau^{-1}, E) = o(\sigma\pi^{-1}, E)$.
2. Se $v, w \notin orb(u, \sigma\pi^{-1})$, e $v \in orb(w, \sigma\pi^{-1})$ então, pela Proposição 4.1.1 (ítem 2), temos $o(\sigma\pi^{-1}\tau^{-1}, E) = o(\sigma\pi^{-1}, E)$.

Então a transposição τ é um 0-movimento aplicável a π . Assim τ é um 0-movimento aplicável a π se e somente se $\tau \nmid \sigma\pi^{-1}$ e

- $\tau^{-1}|\sigma\pi^{-1}$ ou
- $v, w \notin orb(u, \sigma\pi^{-1})$, e $v \in orb(w, \sigma\pi^{-1})$.

□

Dadas duas permutações π e σ sobre E , uma transposição τ aplicável a π é um **x -movimento válido** quando:

$$o_{\text{impar}}(\sigma\pi^{-1}\tau^{-1}, E) = o_{\text{impar}}(\sigma\pi^{-1}, E) + x \quad e$$

$$o(\sigma\pi^{-1}\tau^{-1}, E) = o(\sigma\pi^{-1}, E) + x$$

Dizemos que a permutação α **3-divide** β , denotado por $\alpha|_3\beta$, quando:

$$\|\beta\alpha^{-1}\|_3 = \|\beta\|_3 - \|\alpha\|_3.$$

Lema 4.1.4 *Se π e σ são duas permutações sobre E , uma transposição τ aplicável a π é um 2-movimento válido se e somente se $\tau|_3\sigma\pi^{-1}$.*

Prova Se τ é um 2-movimento válido, então $o_{\text{impar}}(\sigma\pi^{-1}\tau^{-1}, E) = o_{\text{impar}}(\sigma\pi^{-1}, E) + 2$. Como $\|\alpha\|_3 = (|E| - o_{\text{impar}}(\alpha))/2$ para uma permutação α sobre E , temos $|E| - 2\|\sigma\pi^{-1}\tau^{-1}\|_3 = |E| - 2\|\sigma\pi^{-1}\|_3 + 2$. Logo, temos $\|\sigma\pi^{-1}\tau^{-1}\|_3 = \|\sigma\pi^{-1}\|_3 - 1$. Como $\|\tau\|_3 = 1$ para qualquer transposição, então $\|\sigma\pi^{-1}\tau^{-1}\|_3 = \|\sigma\pi^{-1}\|_3 - \|\tau\|_3$. Assim, por definição de 3-divisibilidade, temos $\tau|_3\sigma\pi^{-1}$.

Inversamente, se $\tau|_3\sigma\pi^{-1}$ então $\|\sigma\pi^{-1}\tau^{-1}\|_3 = \|\sigma\pi^{-1}\|_3 - \|\tau\|_3$. Usando $\|\alpha\|_3 = (|E| - o_{\text{impar}}(\alpha))/2$ temos $(|E| - o_{\text{impar}}(\sigma\pi^{-1}\tau^{-1}, E))/2 = (|E| - o_{\text{impar}}(\sigma\pi^{-1}, E) - 2)/2$. Como Conseqüência temos $o_{\text{impar}}(\sigma\pi^{-1}\tau^{-1}, E) = o_{\text{impar}}(\sigma\pi^{-1}, E) + 2$.

Portanto uma transposição τ aplicável a π é um 2-movimento válido se e somente se $\tau|_3\sigma\pi^{-1}$. □

Dadas duas permutações π e σ sobre E , dois ciclos α e β de $\sigma\pi^{-1}$ são **ciclos ligados** para $\sigma\pi^{-1}$ quando $x \rightarrow_{u,\pi} \pi\sigma^{-1}u \rightarrow_{u,\pi} \sigma\pi^{-1}x$, onde u pertence a órbita de α , e x pertence a órbita de β .

Lema 4.1.5 *Se π e σ são duas permutações sobre E tal que não há um 2-movimento aplicável a π , se há dois ciclos ligados $\alpha = (\pi\sigma^{-1}u \ u \ \dots)$ e $\beta = (x \ \sigma\pi^{-1}x \ \dots)$ para $\sigma\pi^{-1}$ e um 0-movimento $\tau = (u \ x \ \sigma\pi^{-1}x)$ aplicável a π , então há um 2-movimento κ aplicável a $\tau\pi$.*

Prova A transposição $\tau = (u \ x \ \sigma\pi^{-1}x)$ é um 0-movimento aplicável a π pela definição de ciclos ligados, Proposição 4.1.1, e Lema 4.1.1. Como α e β são ciclos ligados para $\sigma\pi^{-1}$ então $x \rightarrow_{u,\pi} \pi\sigma^{-1}u \rightarrow_{u,\pi} \sigma\pi^{-1}x$, isto é, temos $x = \pi^{k_1}u$, $\pi\sigma^{-1}u = \pi^{k_2}u$, e $\sigma\pi^{-1}x = \pi^{k_3}u$ (por definição) para $0 < k_1 < k_2 < k_3 < n$ e $n = |\pi|$.

Mostraremos que $\kappa = (u \ x \ \pi\sigma^{-1}u)$ é uma transposição aplicável a $\tau\pi$, e que κ é um 2-movimento. Como a transposição τ troca os blocos consecutivos de x para $\pi^{-1}\sigma\pi^{-1}x$ e de $\sigma\pi^{-1}x$ para u , então, para $\theta = \tau\pi$, temos $x = \theta^{n-k_3+k_1}u$, $\pi\sigma^{-1}u = \theta^{n-k_3+k_2}u$, e $\sigma\pi^{-1}x = \theta^{k_1}u$. Como $k_1 < k_2$, e $n - k_3 > 0$, e $n > k_i$ para

$1 \leq i \leq 3$, então $\sigma\pi^{-1}x \xrightarrow{u, \tau\pi} x \xrightarrow{u, \tau\pi} \pi\sigma^{-1}u$. Conseqüentemente há inteiros m_1 and m_2 tal que $x = \theta^{m_1}u$ e $\pi\sigma^{-1}u = \theta^{m_2}u$, ou equivalentemente $x \xrightarrow{u, \tau\pi} \pi\sigma^{-1}u$. Assim, pelo Lema 4.1.1, temos $(u \ x \ \pi\sigma^{-1}u)|\tau\pi$.

Para mostrar que κ é um 2-movimento aplicável a $\tau\pi$, verificamos que

$$x \xrightarrow{u, \sigma\pi^{-1}\tau^{-1}} \pi\sigma^{-1}u.$$

Neste caso temos $\kappa|\sigma\pi^{-1}\tau^{-1}$ pelo Lema 4.1.1, e κ é um 2-movimento aplicável a $\tau\pi$ pelo Lema 4.1.2.

Temos $\sigma\pi^{-1}\tau^{-1}(\pi\sigma^{-1}u) = u$ porque $\tau^{-1}(\pi\sigma^{-1}u) = \pi\sigma^{-1}u$ e $\sigma\pi^{-1}\pi\sigma^{-1}u = u$. Então, se $x \in orb(u, \sigma\pi^{-1}\tau^{-1})$ então temos $x \xrightarrow{u, \sigma\pi^{-1}\tau^{-1}} \pi\sigma^{-1}u$.

Observe que temos $\sigma\pi^{-1}\tau^{-1}(u) = (\sigma\pi^{-1})^2x$ e $\sigma\pi^{-1}\tau^{-1}((\sigma\pi^{-1})^i x) = (\sigma\pi^{-1})^{i+1}x$ para $2 \leq i \leq |orb(x, \sigma\pi^{-1}x)| - 1$. Como $(\sigma\pi^{-1})^{|orb(x, \sigma\pi^{-1}x)|-1}x = (\sigma\pi^{-1})^{-1}x$ então $\sigma\pi^{-1}\tau^{-1}((\sigma\pi^{-1})^{-1}x) = x$. Conseqüentemente temos $x = (\sigma\pi^{-1}\tau^{-1})^m u$, para $m = |orb(x, \sigma\pi^{-1}x)|$, e então $x \in orb(u, \sigma\pi^{-1}\tau^{-1})$.

Então $(u \ x \ \pi\sigma^{-1}u)|\sigma\pi^{-1}\tau^{-1}$ e a transposição $(u \ x \ \pi\sigma^{-1}u)$ é um 2-movimento aplicável a $\tau\pi$. \square

Proposição 4.1.2 (Bafna and Pevzner(1998)) *Se π e σ são duas permutações sobre E tais que não há um 2-movimento aplicável a π , então há dois ciclos ligados em $\sigma\pi^{-1}$.*

A prova do teorema seguinte decorre dos Lemas 4.1.2, 4.1.4, 4.1.5.

Teorema 4.1.1 *Se π e σ são duas permutações sobre E , então existe um 2-movimento aplicável a π ou um 0-movimento seguido de um 2-movimento aplicável a π .*

Os lemas 4.1.2 a 4.1.5 e as duas proposições desta seção provam a correção do algoritmo *TranspositionSorting2*. O Teorema 4.1.1 prova a razão de aproximação 2.0.

Algorithm TranspositionSorting2 ()

Input: π , $n = |\pi|$, σ

Output: τ_1, \dots, τ_k , $d_t(\pi) \leq k \leq 2d_t(\pi)$

```

1:  $k = 0$ 
2:  $\theta_k = \pi$ 
3: while  $\theta_k \neq \sigma$  do
4:    $k++$ 
5:   if existe  $\tau$  tal que  $\tau|\pi$  e  $\tau|_3\sigma\pi^{-1}$  then
6:      $\tau_k = \tau$ 
7:   else
8:     if existe  $\tau$  tal que  $\tau|\pi$  e  $\tau|\sigma\pi^{-1}$  then
9:        $\tau_k = \tau$ 
10:    else
11:      tome  $(\pi\sigma^{-1}u \ u \ \dots)$  e  $(x \ \sigma\pi^{-1}x \ \dots)$  em  $\sigma\pi^{-1}$ 

```

```

12:          $\tau_k = (u \ x \ \pi\sigma^{-1}u)$ 
13:     end if
14: end if
15:      $\theta_k = \tau_k\theta_{k-1}$ 
16: end while

```

O algoritmo foi implementado utilizando-se a linguagem de programação Java. Para armazenar as permutações e as transposições τ que ordenam π , além dos produtos $\tau\pi$, foram utilizadas estruturas de dados do tipo *vector* da linguagem Java.

Foram implementadas classes para as operações de divisão e produto de permutações e para a classificação dos ciclos (orientado, cruzado, ligado, não ligado) e para a decomposição em ciclos de π .

- *TauPiU*: armazena todas as transposições τ_k , $k > 0$ aplicadas a π , e também o resultado do produto $\tau_k\theta_{k-1}$.
- *Ciclo*: armazena a decomposição em ciclos de uma permutação π . Esta classe tem como propriedades, além da representação da decomposição em ciclos de π , o número de órbitas e o tamanho de cada órbita.
- *Divisao*: nesta classe estão implementadas as definições e os resultados da operação divisão. Ela possui os métodos para calcular norma, norma-3, permutação inversa, número de órbitas ímpares e verificar se uma permutação α divide uma permutação β , se uma permutação α divide-3 uma permutação β , se uma transposição é um 0-movimento, 0-movimento válido, 2-movimento ou 2-movimento válido.
- *Produto*: esta classe possui métodos necessários para calcular o produto entre duas permutações, fornecendo como saída a permutação produto.
- *PropriedadeCiclos*: nesta classe são classificados os ciclos da decomposição em ciclos de uma permutação. Tem como objetivo identificar se os ciclos de uma permutação são ligados, se é possível um 0-movimento ou um 2-movimento.

A complexidade de espaço é $O(kn)$, para $k = d_t(\pi, \sigma)$ e $n = |E|$, pois são necessários $k + 1$ *vectors* para armazenar as permutações, k *vectors* para armazenar as transposições e cada um deles tem tamanho $n = |E|$. O próximo lema mostra a complexidade de tempo da implementação.

Lema 4.1.6 *Dados π e σ sobre E , $|E| = n$, o algoritmo *TranspositionSorting2* ordena o genoma π em tempo $O(n^4)$.*

Prova As linhas 5, 8 e 11 do algoritmo *TranspositionSorting2* são $O(n^3)$, e a linha 3 é $O(n)$. Então, a complexidade de tempo é $O(n^4)$. \square

4.2 Algoritmo de razão 1.5

Nesta seção, inicialmente caracterizamos os ciclos orientados e não orientados de acordo com o formalismo algébrico e depois utilizamos estes resultados para construir o algoritmo de aproximação 1.5.

Dadas as permutações π e σ sobre E , sejam $u, v, w \in \beta$ um k -ciclo, $k \geq 3$, na decomposição em ciclos de $\sigma\pi^{-1}$ tal que $v \rightarrow_{u,\pi} w$ e tomando o 3-ciclo $\alpha = (u v w)$ temos que:

1. β é **ciclo orientado** se $o(\sigma\pi^{-1}\alpha^{-1}, E) = o(\sigma\pi^{-1}, E) + 2$.
2. β é **ciclo não orientado** se $o(\sigma\pi^{-1}\alpha^{-1}, E) = o(\sigma\pi^{-1}, E)$.

Exemplo 4.2.1 Sejam $\pi = (0 3 2 1 5 4 8 7 6)$, $\sigma = (0 1 2 3 4 5 6 7 8)$ e $\sigma\pi^{-1} = (0 7)(1 3)(2 4 6 8 5)$. Então $\beta = (2 4 6 8 5)$ é orientado, pois tomando $u = 2$, $v = 4$ e $w = 6 \in \beta$, temos $\alpha = (2 4 6)$ tal que $\alpha\pi = (2 4 6)(0 3 2 1 5 4 8 7 6) = (0 3 4 8 7 2 1 5 6)$ e $\sigma\pi^{-1}\alpha^{-1} = (0 7)(1 3)(2 8 5)(4)(6)$. Então temos $o(\sigma\pi^{-1}\alpha^{-1}, E) = o(\sigma\pi^{-1}, E) + 2$.

Exemplo 4.2.2 Sejam $\pi = (0 7 6 5 4 3 2 1)$, $\sigma = (0 1 2 3 4 5 6 7)$ e $\sigma\pi^{-1} = (0 2 4 6)(1 3 5 7)$. Então $\beta = (1 3 5 7)$ é não orientado, pois para quaisquer $u, v, w \in \beta$ tais que $\alpha = (u v w)$, $v \rightarrow_u w$, temos $o(\sigma\pi^{-1}\alpha^{-1}, E) = o(\sigma\pi^{-1}, E)$.

Dadas duas permutações π e σ sobre E . Sejam $C = (u \dots)$ um k_1 -ciclo, $k_1 \geq 3$, e $C' = (x \dots)$ um k_2 -ciclo, $k_2 \geq 3$, tais que C e C' pertencem a decomposição em ciclos de $\sigma\pi^{-1}$ e $n = |\pi|$. Dizemos que C e C' são **ciclos cruzados** se existirem $x \in orb(C')$ e $u \in orb(C)$ tais que

$$\pi^{i_1}(u) \rightarrow_{u,\pi} \pi^{i_2}(u) \rightarrow_{u,\pi} \pi^{i_3}(u) \rightarrow_{u,\pi} \pi^{i_4}(u) \rightarrow_{u,\pi} x$$

e

- $\pi^{i_1}(u) \rightarrow_{u,\pi} \pi^{f_1}(u)$, $1 \leq i_1 \leq f_1$, $\pi^{i_1}(u) \in orb(x, \sigma\pi^{-1})$ e $1 \leq f_1 \leq n$ e
- $\pi^{i_2}(u) \rightarrow_{u,\pi} \pi^{f_2}(u)$, $i_1+1 \leq i_2 \leq f_2$, $\pi^{i_2}(u) \in orb(u, \sigma\pi^{-1})$ e $(i_1+1) \bmod n \leq f_2 \leq i_1 \bmod n$ e
- $\pi^{i_3}(u) \rightarrow_{u,\pi} \pi^{f_3}(u)$, $i_2+1 \leq i_3 \leq f_3$ e $\pi^{i_3}(u) \in orb(x, \sigma\pi^{-1})$ e $(i_2+1) \bmod n \leq f_3 \leq i_2 \bmod n$, e
- $\pi^{i_4}(u) \rightarrow_{u,\pi} \pi^{f_4}(u)$, $i_3+1 \leq i_4 \leq f_4$, $\pi^{i_4}(u) \in orb(u, \sigma\pi^{-1})$ e $(i_3+1) \bmod n \leq f_4 \leq n$.

Exemplo 4.2.3 Sejam $\pi = (0 7 6 5 4 3 2 1)$, $\sigma = (0 1 2 3 4 5 6 7)$ e $\sigma\pi^{-1} = (0 2 4 6)(1 3 5 7)$ tal que $C = (0 2 4 6)$ e $C' = (1 3 5 7)$. Os ciclos C e C' são ciclos cruzados, pois se tomarmos $u = 0$ e $x = 1$ temos $i_1 = 1$, $i_2 = 2$, $i_3 = 3$, $i_4 = 4$. Então, $7 \rightarrow_{u,\pi} 6 \rightarrow_{u,\pi} 5 \rightarrow_{u,\pi} 4 \rightarrow_{u,\pi} 1$.

Lema 4.2.1 Dadas as permutações π e σ sobre E . Sejam um ciclo

$$C = (x \cdots y \cdots z \cdots)$$

na decomposição em ciclos de $\sigma\pi^{-1}$ e $i, j, k \notin C$. Se $\tau = (i j k)$, $j \rightarrow_{i,\pi} k$ e τ é cruzado com relação ao ciclo C então τ inverte a orientação de C .

Prova Temos dois casos:

1. Se o ciclo $C = (x \cdots y \cdots z \cdots)$ for orientado em $\sigma\pi^{-1}$. Como τ é cruzado com C em $\sigma\pi^{-1}$ então tomemos $\pi = (i \cdots x \cdots j \cdots y \cdots k \cdots z \cdots)$ e $\tau\pi = (i \cdots x \cdots k \cdots z \cdots j \cdots y \cdots)$. Então $C\tau^{-1} = (x \cdots y \cdots z \cdots)$. τ cria um ciclo $C' = (x \cdots y \cdots z \cdots)$ em $\sigma\pi^{-1}\tau^{-1}$. Tomemos $\tau' = (x z y)$ e $z \rightarrow_{x,\tau\pi} y$:

$$\sigma\pi^{-1}\tau^{-1}\tau'^{-1} = \cdots(x \cdots y \cdots z \cdots) \cdots (y z x) = (y \cdots x \cdots z \cdots).$$

Daí, $o(\sigma\pi^{-1}\tau^{-1}\tau'^{-1}) = o(\sigma\pi^{-1}\tau^{-1})$. Portanto $C' = (x \cdots y \cdots z \cdots)$ em $\sigma\pi^{-1}\tau^{-1}$ é não orientado.

2. Se o ciclo $C = (x \cdots y \cdots z \cdots)$ for não orientado em $\sigma\pi^{-1}$. Como τ é cruzado com C então tomemos $\pi = (i \cdots x \cdots j \cdots z \cdots k \cdots y \cdots)$ e $\tau\pi = (i \cdots x \cdots k \cdots y \cdots j \cdots z \cdots)$. Então $C\tau^{-1} = \cdots(x \cdots y \cdots z \cdots) \cdots$. τ cria um ciclo $C' = (x \cdots y \cdots z \cdots)$ em $\sigma\pi^{-1}\tau^{-1}$. Tomemos $\tau' = (x y z)$ e $y \rightarrow_{x,\tau\pi} z$:

$$\sigma\pi^{-1}\tau^{-1}\tau'^{-1} = \cdots(x \cdots y \cdots z \cdots) \cdots (z y x) = (x \cdots)(y \cdots)(z \cdots).$$

Daí, $o(\sigma\pi^{-1}\tau^{-1}\tau'^{-1}) = o(\sigma\pi^{-1}\tau^{-1}) + 2$. Portanto $C' = (x \cdots y \cdots z \cdots)$ em $\sigma\pi^{-1}\tau^{-1}$ é orientado.

□

Exemplo 4.2.4 Sejam $\pi = (0 9 8 3 2 7 6 5 4 1)$, $\sigma = (0 1 2 3 4 5 6 7 8 9)$ e $\sigma\pi^{-1} = (0 2 4 6 8)(1 5 7 3 9)$. Tomamos o ciclo não orientado $C = (1 5 7 3 9)$ e $x = 3, y = 5, z = 1 \in C$ e $i = 0, j = 2, k = 4 \notin C$, tais que $2 \rightarrow_{0,\pi} 4$ e $3 \rightarrow_{0,\pi} 2 \rightarrow_{0,\pi} 5 \rightarrow_{0,\pi} 4 \rightarrow_{0,\pi} 1$. Então $\tau = (0 2 4)$ é cruzado com relação ao ciclo $C = (1 5 7 3 9)$, $\tau\pi = (0 2 4)(0 9 8 3 2 7 6 5 4 1) = (0 9 8 3 4 1 2 7 6 5)$ e $\sigma\pi^{-1}\tau^{-1} = (0 6 8)(2)(4)(1 5 7 3 9)$. Note que τ inverte a orientação de C em $\sigma\pi^{-1}\tau^{-1}$, pois $\tau' = (3 1 5)$, $1 \rightarrow_{3,\tau\pi} 5$ é orientado em $C' = (1 5 7 3 9)$ e $C' \in \sigma\pi^{-1}\tau^{-1}$.

Lema 4.2.2 Dadas as permutações $\pi = (z \cdots y \cdots x \cdots)$ e σ sobre E e $C = (x \cdots y \cdots z \cdots)$ um ciclo não orientado na decomposição em ciclos de $\sigma\pi^{-1}$. Para todo $x, y, z \in C$, o 3-ciclo $\tau = (z y x)$ tal que $y \rightarrow_{z,\pi} x$ transforma C em um ciclo não orientado em $\sigma\pi^{-1}\tau^{-1}$.

Prova

Seja $C = (x \cdots y \cdots z \cdots)$ um ciclo não orientado em $\sigma\pi^{-1}$. Como $y \rightarrow_{z,\pi} x$ então $\tau\pi = (z \cdots x \cdots y \cdots)$. Logo,

$$C\tau^{-1} = (x \cdots y \cdots z \cdots)(x y z) = (x \cdots z \cdots y \cdots) = C'.$$

Tomemos $\tau' = (z \ x \ y)$, $x \rightarrow_{z,\tau\pi} y$ então:

$$C'\tau'^{-1} = (x \cdots z \cdots y \cdots)(y \ x \ z) = (y \cdots z \cdots x \cdots).$$

Como, $o(\sigma\pi^{-1}\tau^{-1}\tau'^{-1}) = o(\sigma\pi^{-1}\tau^{-1})$, temos que C' é não orientado, pois x , y e z são escolhidos arbitrariamente. \square

Exemplo 4.2.5 *Sejam $\pi = (0 \ 9 \ 8 \ 3 \ 2 \ 7 \ 6 \ 5 \ 4 \ 1)$, $\sigma = (0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8 \ 9)$ e $\sigma\pi^{-1} = (0 \ 2 \ 4 \ 6 \ 8)(1 \ 5 \ 7 \ 3 \ 9)$. Tomamos um ciclo $C = (1 \ 5 \ 7 \ 3 \ 9)$ não orientado em $\sigma\pi^{-1}$. Para todo $x, y, z \in C$ tal que $\tau = (z \ y \ x)$, com $y \rightarrow_{z,\pi} x$, aplicada a π geram um novo ciclo também não orientado.*

Dadas duas permutações π e σ sobre E , C e C' dois ciclos orientados na decomposição em ciclos de $\sigma\pi^{-1}$. Dizemos que C e C' são **ciclos não-interferentes** quando C e C' não forem cruzados.

Exemplo 4.2.6 *Sejam $\pi = (0 \ 4 \ 1 \ 3 \ 5 \ 2)$, $\sigma = (0 \ 1 \ 2 \ 3 \ 4 \ 5)$ e $\sigma\pi^{-1} = (0 \ 3 \ 2)(1 \ 5 \ 4)$. Os ciclos $C = (0 \ 3 \ 2)$ e $C' = (1 \ 5 \ 4)$ são orientados em $\sigma\pi^{-1}$ e C e C' não são cruzados. Então C e C' são ciclos não interferentes.*

Lema 4.2.3 *Dadas as permutações π e σ sobre E e os ciclos $C = (x \ \cdots \ y \ \cdots)$ e $D = (x' \ \cdots \ y' \ \cdots)$ na decomposição em ciclos de $\sigma\pi^{-1}$. Seja o 3-ciclo τ formado por três dos quatro elementos x, y, x' e y' e τ é uma transposição aplicável a π . Então:*

1. *Se o ciclo C não é ligado com o ciclo D então τ cria um ciclo não orientado em $\sigma\pi^{-1}\tau^{-1}$.*
2. *Se o ciclo C é ligado com o ciclo D então τ cria um ciclo orientado em $\sigma\pi^{-1}\tau^{-1}$.*

Prova

1. Como C e D não são ligados então por definição não temos $x' \rightarrow_{x,\pi} y \rightarrow_{x,\pi} y'$. Temos os seguintes casos:

- (a) Para $\pi = (y \cdots x \cdots y' \cdots x' \cdots)$. Tomemos $\tau = (y \ x \ y')$, $x \rightarrow_{y,\pi} y'$. Então:

$$\sigma\pi^{-1}\tau^{-1} = (x \cdots y \cdots)(x' \cdots y' \cdots)(y' \ x \ y) = (y' \cdots y \cdots x' \cdots)(x \cdots).$$

τ transforma C e D nos ciclos $F = (y' \cdots y \cdots x' \cdots)$ e $G = (x \cdots)$ e $\tau\pi = (y \cdots y' \cdots x' \cdots x \cdots)$.

Tomemos $\tau' = (y \ y' \ x')$, $y' \rightarrow_{y,\tau\pi} x'$, temos:

$$\sigma\pi^{-1}\tau^{-1}\tau'^{-1} = (y' \cdots y \cdots x' \cdots)(x \cdots)(x' \ y' \ y) = (x' \cdots y \cdots y' \cdots)(x \cdots).$$

Logo $o(\sigma\pi^{-1}\tau^{-1}\tau'^{-1}) = o(\sigma\pi^{-1}\tau^{-1})$ então F é um ciclo não orientado.

Para as transposições $\tau = (y \ x \ x')$, $x \rightarrow_{y,\pi} x'$, $\tau = (x \ y' \ x')$, $y' \rightarrow_{x,\pi} x'$ e $\tau = (y \ y' \ x')$, $y' \rightarrow_{y,\pi} x'$, as demonstrações são equivalentes.

(b) Para $\pi = (y' \cdots y \cdots x \cdots x' \cdots)$. Tomemos $\tau = (y \ x \ x')$, $x \rightarrow_{y,\pi} x'$.
Então:

$$\sigma\pi^{-1}\tau^{-1} = (x \cdots y \cdots)(x' \cdots y' \cdots)(x' \ x \ y) = (x' \cdots y \cdots y' \cdots)(x \cdots)$$

e $\tau\pi = (y \cdots x' \cdots y' \cdots x \cdots)$. Tomemos $\tau' = (y \ x' \ y')$, $x' \rightarrow_{y,\tau\pi} y'$
então temos:

$$\sigma\pi^{-1}\tau^{-1}\tau'^{-1} = (x' \cdots y \cdots y' \cdots)(x \cdots)(y' \ x' \ y) = (y' \cdots y \cdots x' \cdots)(x \cdots).$$

Daí, $o(\sigma\pi^{-1}\tau^{-1}\tau'^{-1}) = o(\sigma\pi^{-1}\tau^{-1})$ então $F = (y' \cdots y \cdots x' \cdots)$ é um
ciclo não orientado.

Para as transposições $\tau = (y' \ y \ x')$, $y \rightarrow_{y',\pi} x'$ e $\tau = (y' \ x \ x')$,
 $x \rightarrow_{y',\pi} x'$, as demonstrações são equivalentes.

2. Como C e D são ligados então temos: $x' \rightarrow_{x,\pi} y \rightarrow_{x,\pi} y'$. Daí, $\pi =$
 $(x \cdots x' \cdots y \cdots y' \cdots)$. Temos duas possibilidades:

(a) Tomemos $\tau = (y \ x \ x')$, $x \rightarrow_{y,\pi} x'$. Então:

$$\sigma\pi^{-1}\tau^{-1} = (x \cdots y \cdots)(x' \cdots y' \cdots)(x' \ x \ y) = (x' \cdots y \cdots y' \cdots)(x \cdots).$$

e $\tau\pi = (x \cdots y \cdots y' \cdots x' \cdots)$. Tomemos $\tau' = (y \ y' \ x')$, $y' \rightarrow_{y,\tau\pi} x'$
então:

$$\sigma\pi^{-1}\tau^{-1}\tau'^{-1} = (x' \cdots y \cdots y' \cdots)(x \cdots)(x' \ y' \ y) = (x' \cdots)(y' \cdots)(y \cdots)(x \cdots).$$

Daí, $o(\sigma\pi^{-1}\tau^{-1}\tau'^{-1}) = o(\sigma\pi^{-1}\tau^{-1})+2$, então o ciclo $F = (x' \cdots y \cdots y' \cdots)$
em $\sigma\pi^{-1}\tau^{-1}$ é orientado.

(b) Tomemos $\tau = (y' \ x \ x')$, $x \rightarrow_{y',\pi} x'$. Então:

$$\sigma\pi^{-1}\tau^{-1} = (x \cdots y \cdots)(x' \cdots y' \cdots)(x' \ x \ y') = (x' \cdots y \cdots x \cdots)(y' \cdots)$$

e $\tau\pi = (x \cdots y' \cdots x' \cdots y \cdots)$. Tomemos $\tau' = (x \ x' \ y)$, $x' \rightarrow_{x,\tau\pi} y$
então

$$\sigma\pi^{-1}\tau^{-1}\tau'^{-1} = (x' \cdots y \cdots x \cdots)(y' \cdots)(y \ x' \ x) = (y \cdots)(x' \cdots)(x \cdots)(y' \cdots).$$

Daí, $o(\sigma\pi^{-1}\tau^{-1}\tau'^{-1}) = o(\sigma\pi^{-1}\tau^{-1})+2$, então o ciclo $F = (x' \cdots y \cdots x \cdots)$
em $\sigma\pi^{-1}\tau^{-1}$ é orientado.

□

Exemplo 4.2.7 Sejam $\pi = (0 \ 5 \ 6 \ 4 \ 1 \ 2 \ 3)$ e $\sigma\pi^{-1} = (0 \ 4)(1 \ 5)(2)(3)(6)$ onde
 $(0 \ 4)$ e $(1 \ 5)$ são ligados. Tomando $\tau = (0 \ 4 \ 1)$ onde $4 \rightarrow_{0,\pi} 1$ temos $\tau\pi =$
 $(0 \ 4 \ 1)(0 \ 5 \ 6 \ 4 \ 1 \ 2 \ 3) = (0 \ 5 \ 6 \ 1 \ 2 \ 3 \ 4)$. Daí, $\sigma\pi^{-1}\tau^{-1} = (0 \ 5 \ 1)(2)(3)(4)(6)$.
Como $o(\sigma\pi^{-1}\tau^{-1}) = o(\sigma\pi^{-1}) + 2$ então $\tau' = (0 \ 5 \ 1)$ é um ciclo orientado em
 $\tau'\tau\pi$.

Dado um ciclo $C = (\dots i \dots j \dots)$ na decomposição em ciclos de $\sigma\pi^{-1}$, definimos a **distância** entre os elementos $i, j \in C$, denotada por $d(i, j)$, como k tal que $j = (\sigma\pi^{-1})^k(i)$.

Lema 4.2.4 *Dadas as permutações π e σ sobre E . Para todo ciclo $C = (\dots a \dots b \dots)$, $b = \pi^{k_1}(a)$, $k_1 > 1$, não orientado na decomposição em ciclos de $\sigma\pi^{-1}$, então existe um k -ciclo $D = (\dots c \dots d \dots)$, $k \geq 2$, tal que $c \rightarrow_{a,\pi} b \rightarrow_{a,\pi} d$.*

Prova Seja $k_1 = \max_{d(a,b)}$ e $d = k_1 + 1 \pmod{n+1}$. A escolha de k_1 implica que $d \notin C$, porque C é não orientado e $d \neq a$ então $d < k_1$. Daí, $c \rightarrow_{a,\pi} b \rightarrow_{a,\pi} d$. \square

Lema 4.2.5 *Dadas as permutações π e σ sobre E , se existe um ciclo orientado na decomposição em ciclos de $\sigma\pi^{-1}$ então existe uma transposição τ aplicável a π tal que τ é um 2-movimento válido ou existem τ_1, τ_2, τ_3 tais que τ_1 é um 0-movimento válido e τ_2, τ_3 são dois 2-movimentos válidos consecutivos.*

Prova Suponhamos que não seja possível um 2-movimento válido em

$$\pi = (\dots y \dots z \dots x \dots)$$

com $z \rightarrow_{y,\pi} x$, e tomemos um ciclo C na decomposição em ciclos de $\sigma\pi^{-1}$. Consideremos o seguinte conjunto S de transposições aplicáveis a π , $\tau = (x y z)$, $y \rightarrow_{x,\pi} z$, tal que $x, y, z \in C$ e $o(\sigma\pi^{-1}\tau^{-1}) = o(\sigma\pi^{-1}) + 2$ e $y = (\sigma\pi^{-1})^k(x)$ e k é ímpar:

$$S = \{(x y z) : x, y, z \in C, d(x, y) \text{ é ímpar} \}$$

Tomemos $\tau = (y z x)$, $z \rightarrow_{y,\pi} x$. Sabemos que existe C em $\sigma\pi^{-1}$ tal que $C = (\dots x \dots y \dots z)$ e $d(x, y)$ é ímpar. Como $x, y, z \in C$ então $\tau = (y z x)$ é orientada e aplicável a π e age apenas em C então $\pi' = \tau\pi$ é tal que o ciclo C em $\sigma\pi^{-1}$ é transformado em três ciclos C_1, C_2 e C_3 em $\sigma\pi^{-1}\tau^{-1}$, isto é,

$$\sigma\pi^{-1}\tau^{-1} = \dots(\dots x \dots y \dots z) \dots (x z y) = (x \dots)(y \dots)(z \dots)$$

sendo $C_1 = (x \dots)$, $C_2 = (y \dots)$ e $C_3 = (z \dots)$.

Como $\tau \in S$ e $d(x, y)$ é ímpar então C_2 é ímpar. Se C_1 ou C_3 for ímpar então $\Delta_{c_{\text{ímpar}}}(\tau) = 2$ e τ é um 2-movimento válido, o que contradiz a hipótese de que não há 2-movimentos válidos em π . Então C_1 e C_3 são pares e C_1 e C_3 têm tamanho no mínimo dois, isto é, $x = (\sigma\pi^{-1})^k(z)$, $k \geq 2$ e $z = (\sigma\pi^{-1})^l(y)$, $l \geq 2$.

Se existirem a e b em $C \in \sigma\pi^{-1}$ tal que:

1. $a = (\sigma\pi^{-1})^{-i}(z)$, $a = (\sigma\pi^{-1})^{k_1}(x)$, $z = (\sigma\pi^{-1})^{k_2}(x)$ tal que $1 \leq k_1 < k_2$
2. $b = (\sigma\pi^{-1})^j(z)$, $b = (\sigma\pi^{-1})^{-k_3}(x)$.

Analisaremos as seguintes possibilidades:

1. $a \rightarrow_{y,\pi} x$ então $C = (\dots b \dots x \dots y \dots a \dots z)$, $C \in \sigma\pi^{-1}$, e para a permutação $\pi = [\dots y \dots b \dots z \dots a \dots x]$, a transposição $\tau = (y a x)$ é aplicável a π e C é transformado nos seguinte ciclos em $\sigma\pi^{-1}\tau^{-1}$:

$$\begin{aligned}\sigma\pi^{-1}\tau^{-1} &= (\dots x \dots y \dots a \dots z)(x a y) \\ &= (x \dots z \dots)(a \dots)(y \dots)\end{aligned}$$

e $C_1 = (x \dots z \dots)$, $C_2 = (a \dots)$ e $C_3 = (y \dots)$. $C_3 = d(x, y)$ que é ímpar, $C_2 = d(y, a)$ e $C_1 = d(a, z) + 1 + d(z, x)$. Mas, $d(y, z) = d(y, a) + 1 + d(a, z)$ e $d(y, z)$ é par. Então $d(y, a) + d(a, z)$ é ímpar. Daí, temos: se $d(y, a)$ é ímpar e $d(a, z)$ é par então C_2 também é ímpar. E se $d(a, z)$ é ímpar e $d(y, a)$ é par então C_1 também é ímpar, pois $d(z, x)$ é par. Portanto $\tau = (y a x)$ é um 2-movimento válido o que contradiz a hipótese inicial. Consequentemente $a \rightarrow_{y,C} x$ não é verdade. Argumentos simétricos demonstram que $b \rightarrow_{y,C} x$ não é verdade.

2. Se $a \rightarrow_{x,\pi} z$ então $C = (x \dots y \dots a z b \dots)$, $C \in \sigma\pi^{-1}$, $\pi = [b a \dots y \dots z \dots x \dots]$ e $\tau = (a z x)$ é uma transposição aplicável a π , $a = (\sigma\pi^{-1})^{-1}(z)$ em C e C é transformado nos seguintes ciclos em $\sigma\pi^{-1}\tau^{-1}$:

$$\sigma\pi^{-1}\tau^{-1} = \dots(x \dots y \dots a z b \dots) \dots (x z a) = (x b \dots)(z)(a \dots y \dots)$$

sendo $C_1 = (x b \dots)$, $C_2 = (z)$ e $C_3 = (a \dots y \dots)$. O ciclo C_3 tem tamanho $d(x, y) + d(y, a) + 1 = d(x, y) + d(y, z) - 1 + 1$. Daí, $y = (\sigma\pi^{-1})^k(x)$ e k é ímpar e por hipótese $z = (\sigma\pi^{-1})^l(y)$ e l é par, então C_3 é ímpar. Portanto $\tau = (a z x)$ é um 2-movimento válido o que contradiz a hipótese inicial de $a \rightarrow_{x,C} z$.

3. Se $b \rightarrow_{x,\pi} z$ então $C = (x \dots y \dots a z b \dots)$, $C \in \sigma\pi^{-1}$. Tomemos a permutação $\pi = [b a \dots y \dots z \dots x \dots]$, a transposição $\tau = (b a z)$ é aplicável a π e $z = \sigma\pi^{-1}(b)$. C é transformado nos seguintes ciclos em $\sigma\pi^{-1}\tau^{-1}$: $\sigma\pi^{-1}\tau^{-1} = \dots(x \dots y \dots a z b \dots) \dots (z a b) = (z)(a \dots x \dots y \dots)(b)$ e $C_1 = (z)$, $C_2 = (a \dots x \dots y \dots)$ e $C_3 = (b)$. Os ciclos C_1 e C_3 têm tamanho um, C_2 tem tamanho ímpar, pois $C_2 = d(z, x) - 1 + d(x, y) + d(y, z) - 1$ e como $d(z, x)$ e $d(y, z)$ têm tamanho par (por hipótese) então $d(z, x) - 1$, $d(y, z) - 1$ e $d(x, y)$ são ímpares. Daí, $\tau = (b a z)$ é um 2-movimento válido o que contradiz a hipótese inicial e $a \rightarrow_{z,C} y$ e $b \rightarrow_{z,C} y$.

4. Se $b \rightarrow_{x,\pi} a$, então temos o mesmo caso descrito em $b \rightarrow_{x,\pi} z$ e $\tau = (b a z)$ é um 2-movimento válido, o que contradiz a hipótese inicial.

5. Se $a \rightarrow_{x,\pi} b$ então $C = (x \dots y \dots a z b \dots)$, $C \in \sigma\pi^{-1}$ e para a permutação $\pi = [a \dots b \dots y \dots z \dots x \dots]$ a transposição $\tau = (b z x)$ é aplicável a π . C é transformado no seguinte ciclo em $\sigma\pi^{-1}\tau^{-1}$:

$$\sigma\pi^{-1}\tau^{-1} = \dots(x \dots y \dots a z b \dots) \dots (x z b)$$

$$= \cdots(x b \cdots y \cdots a z \cdots) \cdots = C'$$

C' é formado pelos mesmos elementos que estão em C , isto é, C tem tamanho $d(x, y) + d(y, z) + d(z, x)$ que é ímpar. Então τ é um 0-movimento válido. A transposição $\tau_1 = (b y x)$ transforma C' nos seguintes ciclos:

$$\begin{aligned} \sigma\pi^{-1}\tau^{-1}\tau_1^{-1} &= \cdots(x b \cdots y \cdots a z \cdots) \cdots (x y b) \\ &= \cdots(x \cdots a z \cdots)(y \cdots)(b) \cdots \end{aligned}$$

sendo $C_1 = (x \cdots a z \cdots)$, $C_2 = (y \cdots)$ e $C_3 = (b)$. O ciclo C_2 tem tamanho ímpar, pois é formado pelos elementos entre x e y , C_3 tem tamanho um e C_1 tem tamanho $d(y, z) + 1 + d(z, x)$. $d(y, z) + 1$ é ímpar, pois $d(y, z)$ é par e como $d(z, x)$ é par então C_3 tem tamanho ímpar. Logo, τ_1 é um 2-movimento válido. $\tau_2 = (a z x)$ transforma C_1 nos seguintes ciclos: $C_1\tau_2^{-1} = (x \cdots a z \cdots)(x z a) = (x \cdots)(z)(a \cdots)$ e $C_4 = (x \cdots)$, $C_5 = (z)$ e $C_6 = (a \cdots)$. C_6 tem tamanho $d(y, z) - 1$, que é ímpar. Daí, τ_2 é um 2-movimento válido. □

Na demonstração do lema acima, o caso em que $a \rightarrow_{x,\pi} b$ é um exemplo de ciclo orientado que não permite um 2-movimento válido. Para tentar evitar criar estes casos, definimos ciclos fortemente orientados.

Dadas as permutações π e σ sobre E , seja $\alpha = (x \cdots y \cdots z \cdots)$ um k -ciclo, $k \geq 3$ e $d(x, y)$ é ímpar, na decomposição em ciclos de $\sigma\pi^{-1}$. Dizemos que α é um **ciclo fortemente orientado** quando:

- α é orientado;
- Existe pelo menos uma transposição τ que é um 2-movimento válido.

Exemplo 4.2.8 *Sejam $\pi = (0 2 3 1)$ e $\sigma = (0 1 2 3)$. Temos $\sigma\pi^{-1} = (0 2 1)(3)$. O ciclo $\alpha = (0 2 1)$ é orientado. Se tomarmos $\tau = (2 1 0)$, temos $\sigma\pi^{-1}\tau^{-1} = (0 2 1)(3)(1 2 0) = (0)(1)(2)(3)$. Portanto τ é um 2-movimento válido e α é fortemente orientado.*

Os dois lemas a seguir mostram como ciclos fortemente orientados resultam de ciclos não orientados.

Lema 4.2.6 *Dadas as permutações π e σ sobre E , sejam o 3-ciclo $\tau = (u v w)$ tal que $v \rightarrow_{u,\pi} w$ e tomemos $x, y, z \in C$, tal que $C = (x \cdots y \cdots z \cdots)$ é um k -ciclo, $k \geq 3$, com $d(x, y)$ ímpar. O ciclo C é não orientado na decomposição em ciclos de $\sigma\pi^{-1}$ e τ é cruzado com o ciclo C . Então τ transforma C em um ciclo fortemente orientado em $\sigma\pi^{-1}\tau^{-1}$.*

Prova $\tau = (u v w)$ é cruzado com o ciclo $C = (x \cdots y \cdots z \cdots)$ então pelo Lema 4.2.1 τ transforma C em um ciclo orientado D em $\sigma\pi^{-1}\tau^{-1}$. Temos duas possibilidades:

1. Se $u \rightarrow_{x,\pi} z \rightarrow_{x,\pi} v \rightarrow_{x,\pi} y \rightarrow_{x,\pi} w$ e $u, v, w \notin C$, e

$$\sigma\pi^{-1}\tau^{-1} = \cdots (x \cdots y \cdots z \cdots) \cdots (w v u) = \cdots (x \cdots y \cdots z \cdots) \cdots$$

e $D = (x \cdots y \cdots z \cdots)$ é um k -ciclo, $k \geq 3$, orientado e $d(x, y)$ é ímpar, pois as distâncias em D não são alteradas em relação a C .

Tomemos $\tau' = (x y z)$ uma transposição aplicável a $\tau\pi$ então $\sigma\pi^{-1}\tau^{-1}\tau'^{-1} = \cdots (x \cdots y \cdots z \cdots) \cdots (z y x) = (x \cdots)(y \cdots)(z \cdots)$ e tal que $D_1 = (x \cdots)$, $D_2 = (y \cdots)$, $D_3 = (z \cdots)$ e D_2 é ímpar, pois tem tamanho $d(x, y)$. Se D_1 ou D_3 é ímpar então τ' é um 2-movimento válido.

Se D_1 e D_3 são pares então $d(z, x)$ e $d(y, z)$ são pares, isto é D_1 e D_3 têm tamanho no mínimo 2. Portanto existem pelo menos dois outros elementos a e b em $D \in \sigma\pi^{-1}\tau^{-1}$ tal que $a = (\sigma\pi^{-1}\tau^{-1})^{-i}(z)$ e $b = (\sigma\pi^{-1}\tau^{-1})^j(z)$ (a e b nas mesmas condições do Lema 4.2.5). De acordo com o Lema 4.2.5 se existe um ciclo orientado somente é possível um 0-movimento válido seguido de dois 2-movimentos válidos se $a \rightarrow_{x,\pi} b$ e $z \rightarrow_{y,\pi} x$, caso contrário sempre é possível um 2-movimento válido. No nosso caso temos $x \rightarrow_{y,\pi} z$, contrariando o lema, pois D_1 e D_3 são pares. Então, só podemos ter D_1 ou D_3 ímpares, o que configura um 2-movimento válido. Daí, D é um ciclo fortemente orientado em $\sigma\pi^{-1}\tau^{-1}$.

2. Se $x \rightarrow_{u,\pi} v \rightarrow_{u,\pi} z \rightarrow_{u,\pi} w \rightarrow_{u,\pi} y$, mostramos o resultado de forma análoga. □

Lema 4.2.7 *Dadas as permutações $\pi = (\cdots x' \cdots y \cdots y' \cdots x)$ e σ sobre E . Sejam $D = (x \cdots y \cdots)$ e $D' = (x' \cdots y' \cdots)$ dois ciclos não orientados na decomposição em ciclos de $\sigma\pi^{-1}$ e D e D' não são cruzados e D e D' são ligados e $d(x, y)$ é ímpar. Seja τ formado por três dos quatro elementos x, y, x' e y' tal que τ é aplicável a π então τ cria um ciclo fortemente orientado.*

Prova O Lema 4.2.4 garante que para o ciclo D não orientado existe um k -ciclo D' tal que D e D' são ligados, pois $x' \rightarrow_{x,\pi} y \rightarrow_{x,\pi} y'$.

Como $\tau = (x' y y')$ é uma transposição formada por três dos quatro elementos x, y, x', y' e D e D' são ligados então pelo Lema 4.2.3 τ cria um ciclo orientado C em $\sigma\pi^{-1}\tau^{-1}$:

$$\sigma\pi^{-1}\tau^{-1} = \cdots (x \cdots y \cdots)(x' \cdots y' \cdots)(y' y x') = (y' \cdots x \cdots y \cdots)(x' \cdots)$$

e $C = (y' \cdots x \cdots y \cdots)$ é orientado.

Tomemos então $\tau' = (x y y')$ e temos que C é transformado nos seguintes ciclos em $\sigma\pi^{-1}\tau^{-1}\tau'^{-1}$:

$\sigma\pi^{-1}\tau^{-1}\tau'^{-1} = \cdots (y' \cdots x \cdots y \cdots) \cdots (y' y x) = (y' \cdots)(y \cdots)(x \cdots)$ e tal que $C_1 = (y' \cdots)$, $C_2 = (y \cdots)$ e $C_3 = (x \cdots)$. C_2 tem tamanho ímpar, pois $d(x, y)$ é ímpar. Se C_1 ou C_2 tem tamanho ímpar então τ' é um 2-movimento válido. Se C_1 e C_3 têm tamanho par então tomemos $a = (\sigma\pi^{-1}\tau^{-1})^{-i}(y')$ e $b = (\sigma\pi^{-1}\tau^{-1})^j(y')$ (nas mesmas condições da prova do Lema 4.2.5). De acordo com o

Lema 4.2.5 como temos um ciclo orientado em $\sigma\pi^{-1}\tau^{-1}$ então somente é possível um 0-movimento válido seguido de dois 2-movimentos válidos consecutivos se $a \rightarrow_{x,\pi} b$ e $y \rightarrow_{y',\pi} x$. Mas, em $\sigma\pi^{-1}\tau^{-1}$ temos $x \rightarrow_{y',\tau\pi} y$. Portanto τ' é um 2-movimento válido e τ cria um ciclo fortemente orientado em $\sigma\pi^{-1}\tau^{-1}$. \square

Dadas as permutações π e σ sobre E , sejam C um k_1 -ciclo, $k_1 \geq 3$ e C' um k_2 -ciclo, $k_2 \geq 3$, na decomposição em ciclos de $\sigma\pi^{-1}$ tais que C é fortemente orientado e C' é não orientado. Dizemos que C e C' são **fortemente cruzados** se C e C' são ciclos cruzados.

Exemplo 4.2.9 *Sejam $\pi = (0\ 3\ 2\ 5\ 4\ 1)$, $\sigma = (0\ 1\ 2\ 3\ 4\ 5)$ e $\sigma\pi^{-1} = (1\ 5\ 3)(2\ 4\ 0)$. O ciclo $C' = (1\ 5\ 3)$ é não orientado, o ciclo $C = (2\ 4\ 0)$ é fortemente orientado e C e C' são cruzados, pois*

$$3 \rightarrow_{0,\pi} 2 \rightarrow_{0,\pi} 5 \rightarrow_{0,\pi} 4 \rightarrow_{0,\pi} 1.$$

Portanto, o ciclo $C' = (1\ 5\ 3)$ é fortemente cruzado com o ciclo $C = (2\ 4\ 0)$.

Lema 4.2.8 *Dadas as permutações π e σ sobre E , se em $\sigma\pi^{-1}$ existem ciclos fortemente cruzados então temos dois 2-movimentos válidos consecutivos em $\sigma\pi^{-1}$.*

Prova Tomemos os ciclos C e D em $\sigma\pi^{-1}$ tais que C é fortemente orientado e D é não orientado e C e D são fortemente cruzados. Então, por definição, existem $x, y, z \in D$ e $u, v, w \in C$ tal que $x \rightarrow_{u,\pi} v \rightarrow_{u,\pi} y \rightarrow_{u,\pi} w \rightarrow_{u,\pi} z$.

Seja $\tau = (u\ v\ w)$, então pelo Lema 4.2.6 τ transforma D em um ciclo D' fortemente orientado em $\sigma\pi^{-1}\tau^{-1}$ e como C é um ciclo fortemente orientado então τ gera um 2-movimento válido em $\sigma\pi^{-1}\tau^{-1}$.

D' em $\sigma\pi^{-1}\tau^{-1}$ é fortemente orientado então existe τ' em D' que é um 2-movimento válido. \square

Exemplo 4.2.10 *Sejam $\pi = (0\ 9\ 8\ 3\ 2\ 7\ 6\ 5\ 4\ 1)$, $\sigma = (0\ 1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9)$ e $\sigma\pi^{-1} = (0\ 2\ 4\ 6\ 8)(1\ 5\ 7\ 3\ 9)$. Temos que $C = (0\ 2\ 4\ 6\ 8)$ é fortemente orientado, e $\tau = (0\ 2\ 4)$ gera um 2-movimento válido em $\tau\pi$, pois: $(0\ 2\ 4)(0\ 9\ 8\ 3\ 2\ 7\ 6\ 5\ 4\ 1) = (0\ 9\ 8\ 3\ 4\ 1\ 2\ 7\ 6\ 5)$ e $\sigma\pi^{-1}\tau^{-1} = (0\ 6\ 8)(1\ 5\ 7\ 3\ 9)(2)(4)$. O ciclo $(0\ 6\ 8)$ é não orientado e o ciclo $(1\ 5\ 7\ 3\ 9)$ é fortemente orientado então $\tau' = (1\ 5\ 9)$ gera um segundo 2-movimento válido em $\tau'\tau\pi$, pois $\tau'\tau\pi = (1\ 5\ 9)(0\ 9\ 8\ 3\ 4\ 1\ 2\ 7\ 6\ 5) = (0\ 1\ 2\ 7\ 6\ 9\ 8\ 3\ 4\ 5)$ e $\sigma\pi^{-1}\tau^{-1}\tau'^{-1} = (0\ 6\ 8)(1)(2)(3\ 9\ 7)(4)(5)$.*

Dados dois ciclos fortemente orientados C e C' . Dizemos que C é **fortemente não interferente** com relação a C' se C e C' são não interferentes.

Exemplo 4.2.11 *Sejam $\pi = (0\ 2\ 4\ 1\ 5\ 3)$, $\sigma = (0\ 1\ 2\ 3\ 4\ 5)$ e $\sigma\pi^{-1} = (2\ 1\ 5)(0\ 4\ 3)$ tal que $C = (2\ 1\ 5)$ e $C' = (0\ 4\ 3)$. Os ciclos C e C' são fortemente orientados e não são cruzados, então, são não interferentes. Neste caso, C é fortemente não interferente com relação a C' .*

Lema 4.2.9 *Dadas as permutações π e σ sobre E , se existem C e C' na decomposição em ciclos de $\sigma\pi^{-1}$ tal que C é fortemente não interferente com relação a C' então existem dois 2-movimentos válidos consecutivos.*

Prova Temos que C é fortemente não interferente com relação a C' então C e C' não são cruzados. Por definição, como C é fortemente orientado então existem $a, b, c \in C$ tais que $\tau_1 = (a b c)$, $b \rightarrow_{a,\pi} c$, que é um 2-movimento válido.

C e C' são fortemente não interferentes. Então τ_1 não altera a orientação de C' . Como C' é fortemente orientado em $\sigma\pi^{-1}$, então continua fortemente orientado em $\sigma\pi^{-1}\tau_1^{-1}$. Portanto, existem $x, y, z \in C'$ tais que $\tau_2 = (x y z)$, $y \rightarrow_{x,\tau_1\pi} z$, que é um 2-movimento válido. \square

Lema 4.2.10 *Dadas as permutações π e σ sobre E , se existe pelo menos um k -ciclo, $k \geq 3$, não orientado, $C = (x \cdots y \cdots z \cdots)$, $d(x, y)$ ímpar e não existem ciclos orientados na decomposição em ciclos de $\sigma\pi^{-1}$, então existem τ_1 , τ_2 e τ_3 tais que τ_1 é um 0-movimento válido e τ_2, τ_3 são dois 2-movimentos válidos consecutivos.*

Prova

Vamos analisar dois casos.

1. Tomemos $D = (u \cdots v \cdots w \cdots)$ um k_1 -ciclo, $k_1 \geq 3$, não orientado tal que

$$\pi^{i_1}(u) \rightarrow_{u,\pi} \pi^{i_2}(u) \rightarrow_{u,\pi} \pi^{i_3}(u) \rightarrow_{u,\pi} \pi^{i_4}(u) \rightarrow_{u,\pi} x$$

Daí, C e D são ciclos cruzados e denotaremos $y = \pi^{i_3}(u)$, $z = \pi^{i_1}(u)$, $v = \pi^{i_4}(u)$ e $w = \pi^{i_2}(u)$.

Tomemos $\tau = (u \pi^{i_2}(u) \pi^{i_4}(u))$, $\pi^{i_2}(u) \rightarrow_{u,\pi} \pi^{i_4}(u)$. τ é um 0-movimento válido em $\sigma\pi^{-1}\tau^{-1}$:

$$\sigma\pi^{-1}\tau^{-1} = \cdots (x \cdots y \cdots z \cdots)(u \cdots v \cdots w \cdots)(vwu) =$$

$$(x \cdots y \cdots z \cdots)(v \cdots u \cdots w \cdots)$$

tal que $C' = (x \cdots y \cdots z \cdots)$, $D' = (v \cdots u \cdots w \cdots)$ e

$$\tau\pi = (u \cdots \pi^{i_1}(u) \cdots \pi^{i_4}(u) \cdots x \cdots \pi^{i_2}(u) \cdots \pi^{i_3}(u) \cdots)$$

C' é fortemente orientado (Lema 4.2.6) e D' é não orientado (Lema 4.2.2) e C' e D' são cruzados então C' e D' são fortemente cruzados. Logo, existem dois 2-movimentos válidos consecutivos em $\sigma\pi^{-1}\tau^{-1}$ (Lema 4.2.8).

2. Supomos então que não exista nenhum ciclo D tal que C e D sejam cruzados. Sejam os ciclos não orientados $D = (u \cdots)$ um k_1 -ciclo, $k_1 \geq 2$ e $F = (v \cdots)$ um k_2 -ciclo, $k_2 \geq 2$.

Tomemos u, v, w tal que $u \xrightarrow{z,\pi} y, v \xrightarrow{y,\pi} x$ e $w \xrightarrow{x,\pi} z$ e $u, w \in D$ e $v \in F$. Então $\tau = (u v w)$ tal que $v \xrightarrow{u,\pi} w$ e $u \xrightarrow{z,\pi} y \xrightarrow{z,\pi} v \xrightarrow{z,\pi} x \xrightarrow{z,\pi} w$. Portanto τ é uma transposição aplicável a π e cruzada com o ciclo $C = (x \cdots y \cdots z \cdots)$.

τ transforma C em um ciclo C' fortemente orientado em $\sigma\pi^{-1}\tau^{-1}$ (Lema 4.2.6 e temos duas situações (Lema 4.2.3):

- (a) Se D e F são ciclos ligados então τ cria um ciclo D' orientado em $\sigma\pi^{-1}\tau^{-1}$. Tomemos $a = (\sigma\pi^{-1}\tau^{-1})^{-j_1}(u)$, $b = (\sigma\pi^{-1}\tau^{-1})^{j_2}(u)$ (nas mesmas condições da prova do Lema 4.2.5). O Lema 4.2.5 mostra que um ciclo orientado não é fortemente orientado apenas se $a \xrightarrow{u,\tau\pi} b$ e $w \xrightarrow{u,\tau\pi} v$. Mas em D' temos $v \xrightarrow{u,\tau\pi}$, portanto D' é fortemente orientado. Como C' e D' são ciclo fortemente orientados em $\sigma\pi^{-1}\tau^{-1}$ e não são cruzados então C' e D' são fortemente não interferentes e existem dois 2-movimentos válidos consecutivos em $\sigma\pi^{-1}\tau^{-1}$ (Lema 4.2.9).
- (b) Se D e F são dois ciclos não ligados então τ cria um ciclo D' não orientado em $\sigma\pi^{-1}\tau^{-1}$ (Lema 4.2.3). Os ciclos C' e D' mantêm o cruzamento entre o ciclo C e τ em $\sigma\pi^{-1}\tau^{-1}$, então C' e D' são fortemente cruzados e existem dois 2-movimentos válidos consecutivos em $\sigma\pi^{-1}\tau^{-1}$ (Lema 4.2.8).

□

Teorema 4.2.1 *Se $\pi \neq \sigma$ e se na decomposição em ciclos de $\sigma\pi^{-1}$ temos somente k -ciclos, $k \leq 2$, então é possível um 0-movimento válido seguido por um 2-movimento válido.*

Prova Sejam $C = (u v)$ um k_1 -ciclo, $k_1 = 2$, não orientado e $D = (r s)$ um k_2 -ciclo, $k_2 = 2$, não orientado tais que C e D são ligados (Lema 4.2.4).

Tomemos $\tau = (u v s)$ uma transposição aplicável a π formada por três dos quatro elementos u, v, r, s . Então, τ gera um ciclo orientado em $\sigma\pi^{-1}\tau^{-1}$ (Lema 4.2.3).

Como C e D são k -ciclos, $k = 2$, então é possível um 2-movimento válido em $\sigma\pi^{-1}\tau^{-1}$ (Lema 4.2.5). □

Os Lemas 4.2.5, 4.2.10 e o Teorema 4.2.1 provam a correção do algoritmo *TranspositionSorting15*.

Algorithm TranspositionSorting15 ()

Input: $\pi, n = |\pi|, \sigma$

Output: $\tau_1, \dots, \tau_k, d(\pi) \leq k \leq 1.5d(\pi)$

1: $j = 0$

2: $\pi_j = \pi$

3: **while** existe ciclo orientado C em $\sigma\pi_j^{-1}$ **do**

```

4:   {Lema 4.2.5}
5:   j++
6:   if existe um 2-movimento válido em  $C$  then
7:      $\pi_j = \tau\pi_{j-1}$ 
8:   else
9:     Localiza em  $C$  um 0-movimento válido  $\tau$ 
10:     $\pi_j = \tau\pi_{j-1}$ 
11:    j++
12:    Localiza em  $C'$  um 2-movimento válido  $\tau$ 
13:     $\pi_j = \tau\pi_{j-1}$ 
14:    j++
15:    Localiza em  $D'$  um 2-movimento válido  $\tau$ 
16:     $\pi_j = \tau\pi_{j-1}$ 
17:  end if
18: end while
19: while existe ciclo  $C = (x \cdots y \cdots z \cdots)$  não orientado,  $|C| \geq 3$ , em  $\sigma\pi_j^{-1}$  e
     $\pi_j \neq \sigma$  do
20:   {Lema 4.2.10}
21:   if existe ciclo  $D = (u \cdots v \cdots w \cdots)$ ,  $|D| \geq 3$  em  $\sigma\pi_j^{-1}$  e  $C$  e  $D$  são
    cruzados then
22:      $\tau = (\pi_j^{i_2}(u) \pi_j^{i_4}(u) u)$  é um 0-movimento válido
23:     j++
24:      $\pi_j = \tau\pi_{j-1}$ 
25:     {Lema 4.2.8}
26:      $\tau = (x \pi_j^{i_3}(u) \pi_j^{i_1}(u))$  é um 2-movimento válido
27:     j++
28:      $\pi_j = \tau\pi_{j-1}$ 
29:      $\tau = (u \pi_j^{i_2}(u) \pi_j^{i_4}(u))$  é um 2-movimento válido
30:     j++
31:      $\pi_j = \tau\pi_{j-1}$ 
32:   else
33:     if não existem ciclos cruzados e existem ciclos não orientados
       $D = (u \cdots)$ ,  $F = (v \cdots)$  then
34:        $\tau = (u v w)$  é um 0-movimento válido
35:       j++
36:        $\pi_j = \tau\pi_{j-1}$ 
37:       {Lema 4.2.8 ou Lema 4.2.9}
38:        $\tau = (x y z)$  é um 2-movimento válido
39:       j++
40:        $\pi_j = \tau\pi_{j-1}$ 
41:       Localiza um 2-movimento válido  $\tau$  no ciclo fortemente orientado
42:       j++
43:        $\pi_j = \tau\pi_{j-1}$ 
44:     end if
45:   end if
46: end while
47: while existem apenas  $k$ -ciclos,  $k \leq 2$ , em  $\sigma\pi_j^{-1}$  e  $\pi_j \neq \sigma$  do

```

```

48:  {Teorema 4.2.1}
49:  Localiza um 0-movimento válido  $\tau$  em dois ciclos ligados
50:  j++
51:   $\pi_j = \tau\pi_{j-1}$ 
52:  Localiza um 2-movimento válido  $\tau$  no ciclo fortemente orientado
53:  j++
54:   $\pi_j = \tau\pi_{j-1}$ 
55:  end while

```

O algoritmo foi implementado utilizando a linguagem de programação Java. Foram usadas as mesmas estruturas de dados e operações básicas do formalismo algébrico desenvolvido para o algoritmo de razão 2.0. Foram incluídas novas operações para localizar um 2-movimento válido em um ciclo fortemente orientado, um 0-movimento válido que seja cruzado com um ciclo não orientado e as transposições indicadas nas provas dos lemas e teoremas desta seção.

A complexidade de espaço da nossa implementação continua sendo $O(kn)$, para $k = d_t(\pi, \sigma)$ e $n = |E|$, devido aos $k + 1$ *vectors* para armazenar as permutações, k *vectors* para armazenar as transposições, tendo cada um deles o tamanho $n = |E|$. O próximo lema mostra a complexidade de tempo da nossa implementação.

Lema 4.2.11 *Dados π e σ sobre E , $|E| = n$, o algoritmo *TranspositionSorting15* ordena o genoma π em tempo $O(n^8)$.*

Prova Primeiramente o algoritmo verifica se existem ciclos orientados em $\sigma\pi^{-1}$ (linha 3). O pior caso que temos para ciclos orientados é quando $\pi = (0 \ n \ (n - 1) \ \dots \ 2 \ 1)$, $n = |\pi| - 1$, $n \geq 4$ e n par. Assim, linha 3 tem complexidade de tempo $O(n)$. A linha 6 verifica a existência de um 2-movimento válido em π . O nosso pior caso é a inexistência dele. O algoritmo pesquisa todas as possibilidades de transposição formada pelos elementos pertencentes aos ciclos orientados em $\sigma\pi^{-1}$, e neste caso a complexidade de tempo $O(n^4)$. Para localizar um 0-movimento válido (linha 9) a complexidade de tempo é $O(n^4)$ e para localizar um 2-movimento válido (linhas 12 e 15) a complexidade de tempo também é $O(n^4)$. Logo, a complexidade de tempo para aplicar em um ciclo orientado um 2-movimento válido ou um 0-movimento válido seguido de dois 2-movimentos válidos é $O(n^5)$.

Para verificar se existem k -ciclos, $k \geq 3$, não orientados (linha 19) temos complexidade de tempo $O(n)$. Para localizar um 0-movimento válido em ciclos cruzados (linha 21), o pior caso é o da permutação $\pi = (0 \ n \ (n - 1) \ \dots \ 2 \ 1)$, $n = |\pi| - 1$, $n \geq 5$, n ímpar. Neste caso o algoritmo terá complexidade $O(n^7)$. Para localizar uma transposição cruzada ($\tau = (u \ v \ w)$) com o ciclo C que é um 0-movimento válido (linha 34) e $\tau = (x \ y \ z)$ que é um 2-movimento válido (linha 38) a complexidade de tempo é $O(n^4)$ e para localizar um 2-movimento válido (linha 41) a complexidade de tempo também é $O(n^4)$. Logo, a complexidade de tempo para aplicar em um ciclo não orientado um 0-movimento válido seguido de dois 2-movimentos válidos é $O(n^8)$.

As linhas 49 e 52 têm complexidade de tempo $O(n^4)$.

Portanto a nossa implementação do algoritmo *TranspositionSorting15* tem complexidade de tempo $O(n^8)$. \square

Capítulo 5

Experimentos e análise dos resultados

Como estudo de caso para a teoria desenvolvida no capítulo anterior, executamos os algoritmos de aproximação 2.0 e 1.5 utilizando o formalismo algébrico para todas as permutações de tamanho 2 a 11. Estes experimentos foram realizados em uma máquina com processador Pentium 4, 3GHz, 512Mb de Ram e 40Gb de disco rígido.

Para analisar os resultados, comparamos os valores obtidos dos algoritmos de aproximação com a distância de transposição, contando o número de permutações em que estes dois valores foram diferentes. A Tabela 5.1 mostra nas colunas 3 a 5 os resultados dos nossos algoritmos, respectivamente de razões 2.0 e 1.5. Além disso, executamos o algoritmo de razão 1.5 incluindo a heurística descrita no próximo parágrafo.

Tabela 5.1: Comparação dos resultados do algoritmo de aproximação 2.0 (af2), 1.5, sem (af15) e com heurística (afh15) com os resultados do algoritmo exato.

$ E $	número de permutações	af2	af15	afh15
2	2	0	0	0
3	6	0	0	0
4	24	0	0	0
5	120	7	0	0
6	720	65	0	0
7	5040	756	1	0
8	40320	7399	133	124
9	362880	84715	3023	2977
10	3265920	-	69353	69297
11	39916800	-	1235108	1234709

Para permutações estritamente decrescentes aplicamos o resultado de Meidanis, Walter e Dias [14] adaptado ao nosso algoritmo. Dada uma permutação $\pi_n = (0 \cdots i \cdots)$, uma *adjacência* é uma posição i em π tal que $1 \leq i \leq n + 1$ e $\pi(i) - \pi(i - 1) = 1$. Uma *faixa* é uma subsequência $\pi[i \cdots j]$ de π , onde $i < j$, i e $j + 1$ são pontos-de-quebra e não existem quaisquer pontos-de-quebra entre as

posições i até j [6]. Seja r o número de faixas em π_n . A permutação π_r será construída com as r faixas de acordo com a posição em π_n . Por exemplo, tomamos $\pi_5 = (0\ 1\ 3\ 4\ 2\ 5)$. Os três pontos de quebra de π em relação a ι são $(1, 3)$, $(4, 2)$ e $(2, 5)$. As quatro faixas de π são respectivamente 0 1, 3 4, 2 e 5. Então, $r = 4$ e $\pi_r = (0\ 3\ 2\ 1)$. A prova do lema seguinte vem diretamente da definição de π_r [14].

Lema 5.0.12 *Seja $\pi_r = (0\ r\ r-1\ \dots\ 1)$. Então, para todo $i \geq 0$, temos:*

1. Para $r = 2$, então $d_t(\pi_r) = 1$
2. Para $r \geq 3$, então $d_t(\pi_r) = \lfloor \frac{r}{2} \rfloor + 1$

Na Tabela 5.2, as colunas 3 a 7 mostram as diferenças entre $d_t(\pi)$ e o valor computado pelo respectivo algoritmo.

Tabela 5.2: Comparação dos resultados dos algoritmo de WDM-Walter, Dias e Meidanis [27], Ch-Christie [5] com heurísticas (implementado por Walter, Curado and Oliveira [26]), H-Hartman [10] (implementado por Honda [11]), BP-Bafna e Pevzner [4] sem heurísticas (implementado por Oliveira [7]) e BPh-Bafna e Pevzner [4] com heurísticas (implementado por Soares [24]) com os resultados do algoritmo exato.

$ E $	número de permutações	WDM	Ch	H	BP	BPh
2	2	0	0	0	0	0
3	6	0	0	0	0	0
4	24	0	0	0	0	0
5	120	0	0	0	0	0
6	720	6	0	2	0	0
7	5040	72	0	108	1	0
8	40320	1167	40	1517	135	0
9	362880	14327	1182	25425	4361	490
10	3265920	-	-	-	-	17449

Podemos observar que o nosso algoritmo de razão 1.5 produziu resultados bem melhores quando comparamos com o algoritmo de Bafna e Pevzner [7] sem heurísticas. Nosso algoritmo produziu resultados piores do que o algoritmo de Bafna e Pevzner com heurísticas, porém a melhora significativa obtida no algoritmo implementado por Soares [24] foi devida à inclusão da heurística dos ciclos pares, implementada utilizando a técnica *branch-and-bound*, ainda não incluída na presente implementação.

Por fim, apresentamos na Figura 5.1 o tempo de execução do algoritmo de razão 1.5 em relação ao algoritmo de razão 2.0. O algoritmo de razão 1.5 teve o tempo de execução menor que o algoritmo de razão 2.0, pois ele localiza as transposições que são um 2-movimento válido somente em ciclos orientados em $\sigma\pi^{-1}$ e para ciclos não orientados em $\sigma\pi^{-1}$ ele utiliza as transposições determinadas nos resultados que provam a correção do algoritmo, enquanto que o algoritmo de razão 2.0 localiza as transposições que são um 2-movimento de forma exaustiva

na decomposição em ciclos de π . A Figura 5.2 mostra o tempo de execução do nosso algoritmo de razão 1.5 em relação aos algoritmos de BP-Bafna e Pevzner [4] sem heurísticas (implementado por Oliveira [7]) e BPh-Bafna e Pevzner [4] com heurísticas (implementado por Soares [24]).

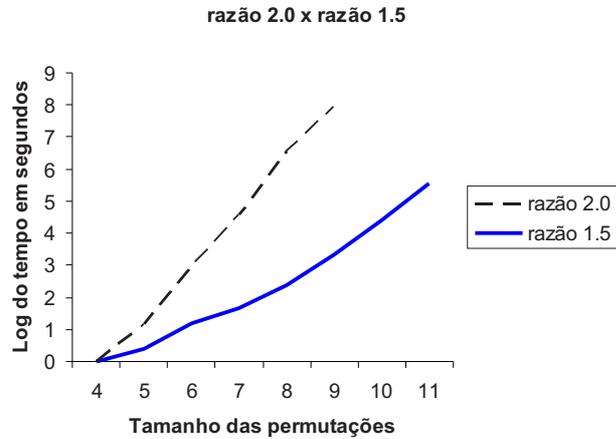


Figura 5.1: Comparação do tempo de execução dos algoritmos de razões 2.0 e 1.5.

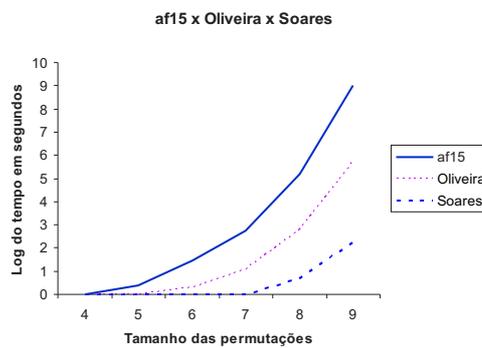


Figura 5.2: Comparação do tempo de execução da nossa implementação (af15) com as implementações dos algoritmos BP-Bafna e Pevzner [4] sem heurísticas (implementado por Oliveira [7]) e BPh-Bafna e Pevzner [4] com heurísticas (implementado por Soares [24])

Acreditamos que a diferença de tempo de processamento é devida ao fato de termos escrito nossos programas utilizando a linguagem de programação Java enquanto que as outras implementações foram escritas utilizando a linguagem C.

Capítulo 6

Conclusões e trabalhos futuros

Neste trabalho propusemos e implementamos os algoritmos de aproximação 2.0 e 1.5 para o problema de ordenação por transposições, utilizando o formalismo algébrico de Dias, Meidanis e Mira [13, 16, 15]. Realizamos experimentos com todas as permutações de tamanho 2 a 11, incluindo ainda a heurística de Meidanis, Walter e Dias [14] adaptada ao nosso algoritmo de razão 1.5, e comparamos os resultados obtidos com os algoritmos de Walter, Dias e Meidanis [27], Christie [5] com heurísticas (implementado por Walter, Curado and Oliveira [26]), Hartman [10] (implementado por Honda [11]), Bafna e Pevzner [4] sem heurísticas (implementado por Oliveira [7]) e Bafna e Pevzner [4] com heurísticas (implementado por Soares [24]). Comparando as implementações baseadas no grafo de ciclos de Bafna e Pevzner [4] àquelas baseadas no formalismo algébrico de Dias, Meidanis e Mira [13, 16, 15] obtivemos melhores resultados.

O trabalho realizado nesta dissertação visa contribuir para a descoberta da complexidade do problema da ordenação por transposições, ainda não conhecida.

Melhores resultados para o problema poderiam ser encontrados através do estudo e desenvolvimento de novas propriedades e resultados teóricos no formalismo algébrico. Outro caminho interessante seria a proposição de um algoritmo de aproximação 1.375, que poderia utilizar estratégias como as apresentadas por Elias e Hartman [9]. A implementação do nosso algoritmo de razão 1.5 poderia ser aprimorada para baixar a complexidade de tempo, utilizando outras estruturas de dados ou novas técnicas. Finalmente, uma outra direção de pesquisa é encontrar o diâmetro de transposição.

Referências

- [1] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman. Gapped blast and psi-blast: a new generation of protein database search programs. *Nucleic Acids Research*, 25:3389–3402, 1997.
- [2] V. Bafna and P. A. Pevzner. Sorting by reversals: Genome rearrangements in plant organelles and evolutionary history of x chromosome. *Molecular Biology and Evolution*, 12(2):239–246, 1995.
- [3] V. Bafna and P. A. Pevzner. Sorting by transpositions. In *Proceedings of the Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 614–623, San Francisco, USA, January 1995.
- [4] V. Bafna and P. A. Pevzner. Sorting by transpositions. *SIAM Journal on Discrete Mathematics*, 11(2):224–240, May 1998.
- [5] D. A. Christie. *Genome rearrangements problems*. PhD thesis, Glasgow University, Scotland, 1998.
- [6] L. R. A. F. Curado. Problema da ordenação por transposições em rearranjos de genomas. Master’s thesis, Universidade de Brasília, Brasil, 2001. In Portuguese.
- [7] E. T. G. de Oliveira. Implementação de algoritmos para o problema de ordenação de transposições. Master’s thesis, Universidade de Brasília, Brasil, 2001. In Portuguese.
- [8] Z. Dias and J. Meidanis. Sorting by prefix transpositions. In A. H. F. Laender and A. L. Oliveira, editors, *Proceedings of the String Processing and Information Retrieval (SPIRE’2002)*, number 2476 in Lecture Notes in Computer Science, pages 65–76, Lisboa, Portugal, September 2002. Springer-Verlag, Berlin.
- [9] I. Elias and T. Hartman. A 1.375-approximation algorithm for sorting by transpositions. In *WABI*, volume 3692 of *Lecture Notes in Computer Science*, pages 204–215. Springer, 2005.
- [10] T. Hartman. A simpler 1.5-approximation algorithm for sorting by transpositions. In *14th Annual Symposium, CPM 2003, Morelia, Michoacán, Mexico*, pages 156–169, 2003.

- [11] M. I. Honda. Implementation of the algorithm of Hartman for the problem of sorting by transpositions. Master's thesis, Universidade de Brasilia (in portuguese), 2004.
- [12] S. MacLane and G. Birkhoff. *Algebra*. The Macmillan Company, London, sixth printing edition, 1971.
- [13] J. Meidanis and Z. Dias. An alternative algebraic formalism for genome rearrangements. In D. Sankoff and J. H. Nadeau, editors, *Comparative Genomics: Empirical and Analytical Approaches to Gene Order Dynamics, Map Alignment and Evolution of Gene Families*, pages 213–223. Kluwer Academic Publishers, November 2000.
- [14] J. Meidanis, M. E. M. T. Walter, and Z. Dias. Transposition distance between a permutation and its reverse. In R. Baeza-Yates, editor, *Proceedings of the 4th South American Workshop on String Processing (WSP'97)*, pages 70–79, Valparaiso, Chile, 1997. Carleton University Press.
- [15] C. Mira and J. Meidanis. Algebraic formalism for genome rearrangements (part 1). Technical Report IC-05-10, Institute of Computing - University of Campinas, June 2005.
- [16] C. V. G. Mira and J. Meidanis. Analysis of sorting by transpositions based on algebraic formalism, March 2004. Eighth Annual International Conference on Research in Computational Molecular Biology (RECOMB2004).
- [17] J. Monteiro. *Elementos de Álgebra*. Livros Técnicos e Científicos, 1969.
- [18] J. H. Nadeau and B. A. Taylor. Lengths of chromosomal segments conserved since divergence of man and mouse. *Proceedings of the National Academy of Sciences, USA*, 81:814–818, 1984.
- [19] S. B. Needleman and C. D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of molecular biology*, 48:443–53, 1970.
- [20] J. D. Palmer and L. A. Herbon. Plant mitochondrial DNA evolves rapidly in structure, but slowly in sequence. *Journal of Molecular Evolution*, 27:87–97, 1988.
- [21] D. Sankoff, G. Leduc, N. Antoine, B. Paquin, B. F. Lang, and R. Cedergren. Gene order comparisons for phylogenetic inference: Evolution of the mitochondrial genome. *Proceedings of the National Academy of Sciences, USA*, (89):6575–6579, 1992.
- [22] J. C. Setubal and J. Meidanis. *Introduction to Computational Molecular Biology*. PWS Publishing Company, 1997.
- [23] T. F. Smith and M. S. Waterman. Identification of common molecular subsequences. *J. Mol. Biol.*, 147:195–197, 1981.

- [24] L. S. N. Soares. A heuristic for the algorithm Bafna e Pevzner to the problem of sorting by transpositions. Master's thesis, Universidade de Brasília, Brazil, 2004. In Portuguese.
- [25] M. E. M. T. Walter. *Algoritmos para Problemas em Rearranjo de Genomas*. PhD thesis, Universidade de Campinas, Brazil, 1999. In Portuguese.
- [26] M. E. M. T. Walter, L. R. A. F. Curado, and A. G. Oliveira. Working on the problem of sorting by transpositions on genome rearrangements. In *14th Annual Symposium, CPM 2003, Morelia, Michoacán, Mexico*, pages 372–383, 2003.
- [27] M. E. M. T. Walter, Z. Dias, and J. Meidanis. A new approach for approximating the transposition distance. In *String Processing and Information Retrieval - SPIRE 2000*, pages 199–208, 2000.
- [28] M. E. M. T. Walter, L. Soares, and Z. Dias. Branch-and-bound algorithms for the problem of sorting by transpositions on genome rearrangements. *XXXIII Seminário Integrado de Software e Hardware - SEMISH*, 2006.
- [29] J. Watson and F. Crick. A structure for Deoxyribose Nucleic Acid. *Nature*, 171(4356):737–738, 1953.