

Defesa de Tese de Doutorado

Algoritmos para Problemas de Ordenação por Reversões ou Transposições, com Aplicações em Rearranjo de Genomas

Aluno: Gustavo Rodrigues Galvão

Orientador: Prof. Dr. Zanoni Dias



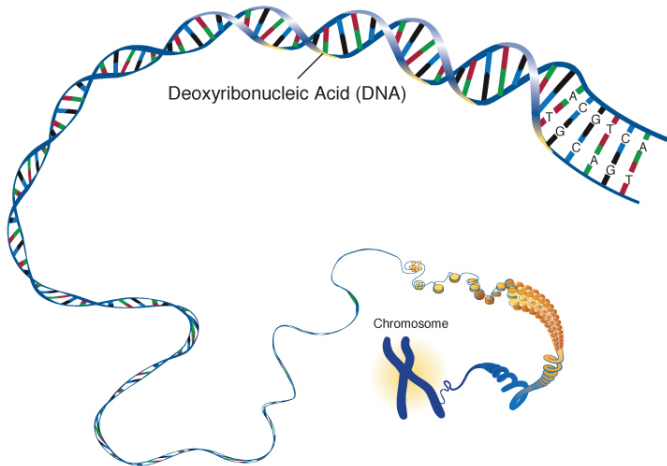
02 de Outubro de 2015

Agenda

- 1 Introdução
- 2 Ferramenta de Auditoria para Algoritmos de Rearranjo de Genomas
- 3 Heurística Genérica para Problemas de Rearranjo de Genomas
- 4 Abordagens Alternativas para Aproximar a Distância de Transposição
- 5 Ordenação de Permutações com Sinais por Operações Curtas
- 6 Considerações Finais

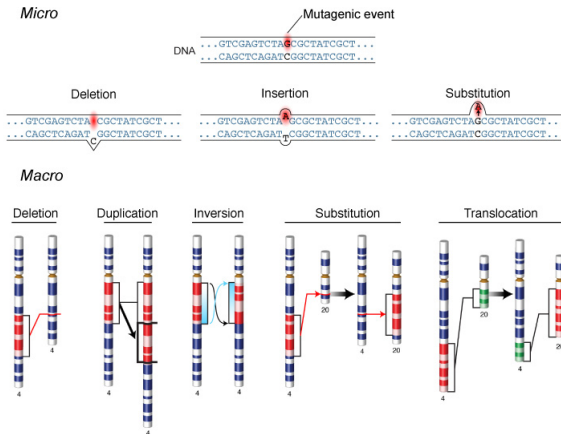
Introdução

DNA, Gene, Cromossomo e Genoma



Fonte: National Institutes of Health. "Talking Glossary of Genetic Terms." National Human Genome Research Institute. Disponível em: <http://www.genome.gov/glossary/>

Mutações



Fonte: National Institutes of Health. "Talking Glossary of Genetic Terms." National Human Genome Research Institute. Disponível em: <http://www.genome.gov/glossary/>

Problema Motivador

Estimar a distância evolutiva entre espécies.

Proposta de Solução

Dados dois genomas relativos a duas espécies, calcula-se a distância de rearranjo entre eles, que é igual ao tamanho da menor sequência de eventos de rearranjo que transforma um genoma no outro. Utiliza-se, então, a distância de rearranjo como estimativa para a distância evolutiva entre as espécies.

- Premissas quanto aos genomas:
 - Genoma unicromossomal.
 - Mesmo conjunto de genes.
 - Sem repetição de genes.
- Representação clássica: permutações de números inteiros.
 - $\pi = (\pi_1 \pi_2 \dots \pi_n)$, $1 \leq |\pi_i| \leq n$ e $|\pi_i| \neq |\pi_j| \leftrightarrow i \neq j$.
- Cada elemento π_i possui um sinal, + ou -, que indica a orientação do gene que ele representa.
 - O sinal é omitido caso a orientação não seja considerada.

Reversão

A reversão $\rho(i, j)$, $1 \leq i < j \leq n$, aplicada na permutação sem sinal

$$(\pi_1 \ \pi_2 \ \dots \ \pi_{i-1} \ \underline{\pi_i \ \pi_{i+1} \ \dots \ \pi_{j-1} \ \pi_j} \ \pi_{j+1} \ \dots \ \pi_n)$$

a transforma na permutação sem sinal

$$(\pi_1 \ \pi_2 \ \dots \ \pi_{i-1} \ \underline{\pi_j \ \pi_{j-1} \ \dots \ \pi_{i+1} \ \pi_i} \ \pi_{j+1} \ \dots \ \pi_n).$$

Caso a permutação original tivesse sinal, então a reversão a transformaria na permutação com sinal

$$(\pi_1 \ \pi_2 \ \dots \ \pi_{i-1} \ \underline{-\pi_j \ -\pi_{j-1} \ \dots \ -\pi_{i+1} \ -\pi_i} \ \pi_{j+1} \ \dots \ \pi_n).$$

Transposição

A transposição $\rho(i, j, k)$, $1 \leq i < j < k \leq n + 1$, aplicada na permutação (com ou sem sinal)

$$(\pi_1 \dots \pi_{i-1} \underline{\pi_i \dots \pi_{j-1}} \underline{\pi_j \dots \pi_{k-1}} \pi_k \dots \pi_n)$$

a transforma na permutação (com ou sem sinal)

$$(\pi_1 \dots \pi_{i-1} \underline{\pi_j \dots \pi_{k-1}} \underline{\pi_i \dots \pi_{j-1}} \pi_k \dots \pi_n).$$

Modelo de Rearranjo

Conjunto de eventos de rearranjo permitidos para transformar uma permutação em outra.

Distância de Rearranjo

A distância de rearranjo entre duas permutações π e σ é igual ao tamanho da menor sequência de eventos de rearranjo pertencentes a um determinado modelo de rearranjo que transforma π em σ (ou σ em π).

Problema da Ordenação por Rearranjo

Encontrar a menor sequência de eventos de rearranjo pertencentes a um determinado modelo de rearranjo que transforma uma permutação π na permutação identidade $\iota = (1\ 2\ \dots\ n)$.

Estado da Arte de Algumas Variações

Modelo de Rearranjo	Complexidade	Melhor Solução Teórica
Reversão com Sinal	P	Algoritmo exato $O(n^{\frac{3}{2}} \sqrt{\log n})$
Reversão	NP-Difícil	Algoritmo 1.375-aproximado
Transposição	NP-Difícil	Algoritmo 1.375-aproximado
Reversão de Prefixo	NP-Difícil	Algoritmo 2-aproximado
Transposição de Prefixo	?	Algoritmo 2-aproximado
Reversão Curta	?	Algoritmo 2-aproximado
Reversão com Sinal e Transposição	?	Algoritmo 2-aproximado
Reversão e Transposição	?	Algoritmo $2k$ -aproximado

Uma Ferramenta de Auditoria para Algoritmos de Rearranjo de Genomas

- Dificuldade do Problema da Ordenação por Rearranjo:
 - Heurísticas.
 - Algoritmos aproximados.
- Construção de uma ferramenta de auditoria.

- Etapas da construção da ferramenta:
 - 1 Definição do algoritmo para calcular distâncias de rearranjo.
 - 2 Cálculo das distâncias de rearranjo.
 - 3 Desenvolvimento da ferramenta (GRAAu).
- Aplicação da ferramenta e resultados.

Algoritmo para Calcular as Distâncias

- Algoritmo de busca em largura:
 - Simplicidade.
 - Flexibilidade.
- Diminuição do uso de memória:
 - Representação de permutações como inteiros.
 - 12x melhor do que o algoritmo de Dias e Meidanis¹.
 - 7x melhor do que o algoritmo de Walter, Dias e Meidanis².
- Diminuição do tempo de execução:
 - Paralelização.

¹Z. Dias and J. Meidanis. *Sorting by prefix transpositions*. In *Proceedings of the SPIRE'2002*, volume 2476 of *LNCS*, pages 65–76, Lisbon, Portugal, 2002. Springer-Verlag.

²M. E. M. T. Walter, Z. Dias, and J. Meidanis. *A new approach for approximating the transposition distance*. In *Proceedings of the SPIRE'2000*, pages 199–208, Washington, DC, USA, 2000. IEEE Computer Society.

- Foram considerados 11 modelos de rearranjo:
 - Permutação sem sinal: $n \leq 13$.
 - Permutação com sinal: $n \leq 10$.
- Base de dados de distâncias de rearranjo:
 - Interface web.

- Implementação em Java:
 - Arquitetura cliente-servidor.
- Estatísticas produzidas:
 - Diâmetro.
 - Distância média.
 - Razão média.
 - Razão máxima.
 - Igualdade.
- Permutações que apresentaram a razão máxima.

- Auditoria de 4 algoritmos aproximados:
 - 2 para o Problema da Ordenação por Reversões de Prefixo.
 - 2 para o Problema da Ordenação por Transposições de Prefixo.
- Justeza do fator de aproximação:
 - Mostramos que o fator de aproximação de 3 algoritmos é justo.
 - Conjecturamos que o fator do outro algoritmo é justo.
- Contraposição da hipótese de Fischer e Ginzinger³.

³ J. Fischer and S. W. Ginzinger. A 2-approximation algorithm for sorting by prefix reversals. In *Proceedings of the ESA'2005*, volume 3669 of LNCS, pages 415–425, Mallorca, Spain, 2005. Springer-Verlag.

Uma Heurística Genérica para Problemas de Rearranjo de Genomas

- Base de dados de distâncias de rearranjo:
 - Bilhões de permutações com ordenação ótima para diversos modelos de rearranjo.
- Como usar essa informação?
 - Heurística de melhoria.

Permutação Estendida

Dada uma permutação $\pi \in S_n$, nós a estendemos com dois elementos $\pi_0 = 0$ e $\pi_{n+1} = n + 1$. A permutação estendida também é denota por π .

Breakpoint de Transposição

Um breakpoint de transposição de uma permutação $\pi \in S_n$ é um par de elementos adjacentes (π_i, π_{i+1}) tal que $\pi_{i+1} - \pi_i \neq 1$, $0 \leq i \leq n$.

Breakpoint de Reversão

Um breakpoint de reversão de uma permutação $\pi \in S_n$ é um par de elementos adjacentes (π_i, π_{i+1}) tal que $|\pi_{i+1} - \pi_i| \neq 1$, $0 \leq i \leq n$.

Strips e Permutações Reduzidas

Strip

Sequência máxima de elementos na qual pares de elementos adjacentes não formam breakpoints.

Breakpoints de transposição definem strips de transposição.

Breakpoints de reversão definem strips de reversão.

Permutação Reduzida

Uma permutação é dita reduzida se ela possui apenas strips de transposição de tamanho 1.

Exemplos: (0 4 3 2 1 5)

(0 3 4 1 2 5)

Permutação Parcialmente Reduzida

Uma permutação é dita parcialmente reduzida se ela possui apenas strips de reversão de tamanho 1 ou 2.

Exemplos: (0 4 3 2 1 5)

(0 3 4 1 2 5)

Reduzindo Permutações

Permutações que não são (parcialmente) reduzidas podem ser (parcialmente) reduzidas.

Exemplo de Redução

$$\begin{aligned}\pi &= (\underline{0} \ \underline{2} \ \underline{3} \ \underline{1} \ \underline{4} \ \underline{5} \ \underline{6}) \\ &\quad (\underline{0} \ \underline{2} \ \underline{1} \ \underline{3} \ \underline{4} \ \underline{5}) \\ \pi^r &= (\underline{0} \ \underline{2} \ \underline{1} \ \underline{3})\end{aligned}$$

Exemplo de Redução Parcial

$$\begin{aligned}\sigma &= (\underline{0} \ \underline{2} \ \underline{3} \ \underline{1} \ \underline{4} \ \underline{5} \ \underline{6} \ \underline{7} \ \underline{10} \ \underline{9} \ \underline{8} \ \underline{11}) \\ &\quad (\underline{0} \ \underline{2} \ \underline{3} \ \underline{1} \ \underline{4} \ \underline{5} \ \underline{8} \ \underline{7} \ \underline{6} \ \underline{9}) \\ \sigma^{pr} &= (\underline{0} \ \underline{2} \ \underline{3} \ \underline{1} \ \underline{4} \ \underline{5} \ \underline{7} \ \underline{6} \ \underline{8})\end{aligned}$$

A redução pode ser revertida se contabilizarmos os elementos descartados.

Descartados

$$[0, 1, 0, 2]$$

Descartados

$$[0, 0, 0, 2, 1, 0]$$

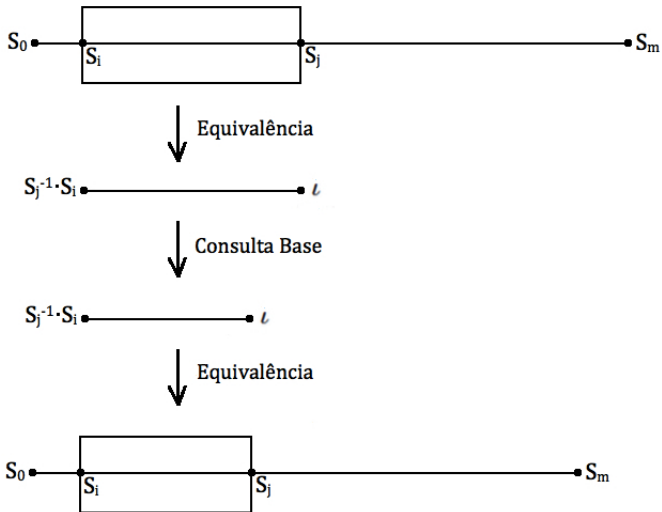
Seja $S = [S_0, S_1, \dots, S_m]$ uma solução para o problema de transformar a permutação S_0 na permutação S_m .

A subsequência $S' = [S_i, S_{i+1}, \dots, S_j]$, $0 \leq i < j \leq m$, é uma solução para o problema de transformar S_i em S_j .

Suponha que i e j sejam escolhidos de tal modo que a permutação $\pi = S_j^{-1} \cdot S_i$ possa ser (parcialmente) reduzida para uma permutação $\sigma \in S_n$, $n \leq 12$.

Então, podemos verificar a otimalidade de S' consultando a distância da permutação σ na base de dados de distâncias de rearranjo. Caso não seja uma solução ótima, trocamos S' pela solução armazenada na base de dados.

Ideia Geral da Heurística



Executamos 23 algoritmos aproximados sobre um mesmo conjunto de permutações e aplicamos a heurística nas soluções obtidas.

Problem	Code	Authors	Ratio
Prefix Reversal	FG	Fischer and Ginzinger	2
	LD	Lintzmayer and Dias	2
Prefix Reversal and Prefix Transposition	DD	Dias and Dias	2
	LD	Lintzmayer and Dias	3
	SEA	Sharmin <i>et al.</i>	3
Prefix Reversal, Prefix Transposition, Suffix Reversal and Suffix Transposition	LD1	Lintzmayer and Dias	2
	LD2	Lintzmayer and Dias	2
Prefix Reversal and Suffix Reversal	LD1	Lintzmayer and Dias	2
	LD2	Lintzmayer and Dias	2
Prefix Transposition and Suffix Transposition	LD1	Lintzmayer and Dias	2
	LD2	Lintzmayer and Dias	2
Prefix Transposition	DM	Dias and Meidanis	2
	GD	Galvão and Dias	2
Reversal	C	Christie	1.5
	KS	Kececioğlu and Sankoff	2
Reversal and Transposition	DEA	Dias <i>et al.</i>	3
	RSH	Rahman, Shatabda and Hasan	3
	WDM	Walter, Dias and Meidanis	3
Transposition	BP	Bafna and Pevzner	1.5
	DD1	Dias and Dias	1.5
	DD2	Dias and Dias	1.375
	EH	Elias and Hartman	1.375
	WDM	Walter, Dias and Meidanis	2.25

Resumo dos Resultados

Problem	Code	% of Improvement
Prefix Reversal	FG	47.09
	LD	68.68
Prefix Reversal and Prefix Transposition	DD	77.52
	LD	70.80
	SEA	86.13
Prefix Reversal, Prefix Transposition, Suffix Reversal and Suffix Transposition	LD1	86.18
	LD2	79.34
Prefix Reversal and Suffix Reversal	LD1	68.57
	LD2	76.93
Prefix Transposition and Suffix Transposition	LD1	99.99
	LD2	76.77
Prefix Transposition	DM	99.95
	GD	66.44
Reversal	C	15.08
	KS	74.93
Reversal and Transposition	DEA	53.54
	RSH	93.42
	WDM	99.71
Transposition	BP	23.53
	DD1	3.89
	DD2	2.47
	EH	18.26
	WDM	30.90

Abordagens Alternativas para Aproximar a Distância de Transposição

- Problema da Ordenação por Transposições:
 - NP-Difícil.
 - Melhores algoritmos se baseiam no grafo de ciclos.
- Abordagens alternativas ao grafo de ciclos:
 - Algoritmo de Walter, Dias e Meidanis⁴.
 - Algoritmo de Benoît-Gagné e Hamel⁵.
 - Heurística de Guyer, Heath e Vergara⁶.
- Estudo das abordagens alternativas:
 - É possível melhorar?
 - Quão boas elas são na prática?

⁴ M. E. M. T. Walter, Z. Dias, and J. Meidanis. A new approach for approximating the transposition distance. In *Proceedings of the SPIRE'2000*, pages 199–208, Washington, DC, USA, 2000. IEEE Computer Society.

⁵ M. Benoît-Gagné and S. Hamel. A new and faster method of sorting by transpositions. In *Proceedings of the CPM'2007*, volume 4580 of LNCS, pages 131–141, London, Ontario, Canada, 2007. Springer-Verlag.

⁶ S. A. Guyer, L. S. Heath, and J. P. C. Vergara. *Subsequence and run heuristics for sorting by transpositions*. Technical Report TR-97-20, Virginia Polytechnic Institute & State University, 1997.

Estado da Arte Anterior

Algoritmo	Complexidade	Aproximação
Walter, Dias e Meidanis	$O(n^2)$	2.25
Benoît-Gagné e Hamel	$O(n^2)$	3
Guyer, Heath e Vergara	$O(n^5 \log n)$?

Melhorias

- Algoritmo de Benoît-Gagné e Hamel:
 - Preenchemos uma lacuna na prova do fator de aproximação.
 - Mostramos como implementar em tempo $O(n \log n)$.
- Algoritmo de Guyer, Heath e Vergara:
 - Provamos ter fator de aproximação igual a 3.

- Experimentos com permutações pequenas:
 - Auditamos os 3 algoritmos estudados com o GRAAu.
- Experimentos com permutações grandes:
 - Permutações de tamanho variando entre 10 e 300.
 - Consideramos algoritmos^{7,8,9,10} baseados no grafo de ciclos.

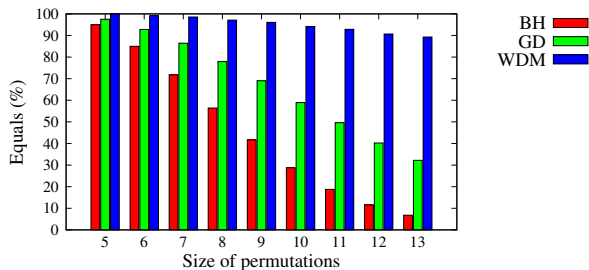
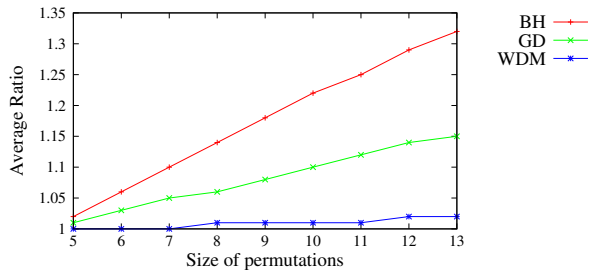
⁷ V. Bafna and P. A. Pevzner. *Sorting by transpositions*. *SIAM Journal on Discrete Mathematics*, 11(2):224–240, 1998.

⁸ I. Eliás and T. Hartman. *A 1.375-approximation algorithm for sorting by transpositions*. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 3(4):369–379, 2006.

⁹ U. Dias and Z. Dias. *Extending Bafna-Pevzner algorithm*. In *Proceedings of the ISB'2010*, pages 23:1–23:8, Calicut, Kerala, India, 2010. ACM Press.

¹⁰ U. Dias and Z. Dias. *An improved 1.375-approximation algorithm for the transposition distance problem*. In *Proceedings of the BCB'2010*, pages 334–337, Niagara Falls, New York, 2010. ACM Press.

Resultados Obtidos com o GRAAu



Análise da Razão Máxima

Algoritmo de Walter, Dias e Meidanis

	n									
	4	5	6	7	8	9	10	11	12	13
Maximum Ratio	1.00	1.00	1.33	1.33	1.50	1.50	1.50	1.60	1.60	1.60

- A razão máxima parece seguir a progressão

$$\frac{2}{2}, \frac{4}{3}, \frac{6}{4}, \frac{8}{5}, \dots, \frac{2k}{k+1}.$$

- Indicação de que o fator de aproximação pode ser baixado.

Análise da Razão Máxima

Algoritmo de Benoît-Gagné e Hamel

	n									
	4	5	6	7	8	9	10	11	12	13
Maximum Ratio	1.00	1.50	1.67	2.00	2.00	2.00	2.25	2.25	2.25	2.40

- A razão máxima parece seguir a progressão

$$\frac{6}{3}, \frac{9}{4}, \frac{12}{5}, \dots, \frac{3k}{k+1}.$$

- Encontramos permutações π de tamanho $n = 3k + 1$, $k \in \{5, 6, 7\}$, para as quais $\frac{p(\pi)}{d(\pi)} = \frac{3k}{k+1}$.
 - Para $k = 7$, $\frac{p(\pi)}{d(\pi)} = 2.625$.
- Contraposição da hipótese de Benoît-Gagné e Hamel.

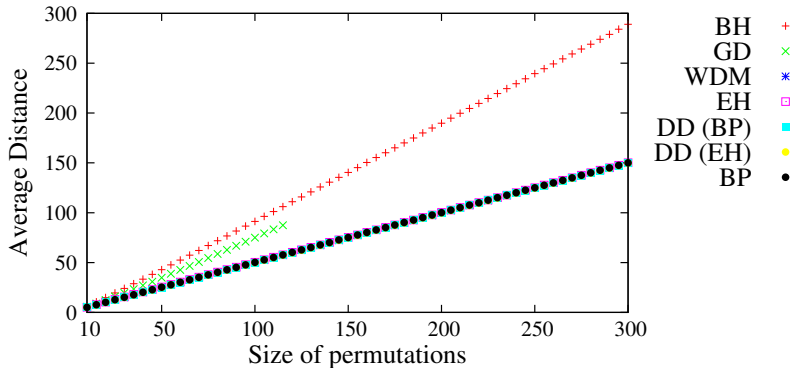
Análise da Razão Máxima

Versão Adaptada da Heurística de Guyer, Heath e Vergara

	n										
	4	5	6	7	8	9	10	11	12	13	
Maximum Ratio	1.00	1.50	1.50	1.67	1.67	2.00	2.00	2.00	2.00	2.25	

- Fator de aproximação pelo menos 2.25.

Resultados Obtidos para Permutações Grandes



Ordenação de Permutações com Sinais por Operações Curtas

- Questão relevante: tamanho das operações.
- Prevalência e significância de reversões curtas em:
 - Genomas bacterianos^{11,12,13}.
 - Genomas eucarióticos¹⁴.
- Abordagens para endereçar essa questão na literatura:
 - Atribuir peso às operações em função do tamanho.
 - Restringir operações baseando-se no tamanho.
- Ordenação por Operações Curtas:
 - Resultados conhecidos para permutações sem sinais.
 - E quanto às permutações com sinais?

¹¹ D. A. Dalevi, N. Eriksen, K. Eriksson, and S. G. E. Andersson. *Measuring genome divergence in bacteria: A case study using chlamydian data.* *Journal of Molecular Evolution*, 55(1):24–36, 2002.

¹² J. F. Lefebvre, N. El-Mabrouk, E. Tillier, and D. Sankoff. *Detection and validation of single gene inversions.* *Bioinformatics*, 19(suppl 1):i190–i196, 2003.

¹³ A. E. Darling, I. Miklós, and M. A. Ragan. *Dynamics of genome rearrangement in bacterial populations.* *PLoS Genetics*, 4(7):e1000128, 2008.

¹⁴ C. Seoighe et al. *Prevalence of small inversions in yeast gene order evolution.* *Proceedings of the National Academy of Sciences*, 97(26):14433–14437, 2000.

- Permutações lineares:
 - Ordenação por Reversões Super Curtas.
 - Ordenação por Reversões Curtas.
 - Ordenação por Reversões e Transposições Super Curtas.
 - Ordenação por Reversões e Transposições Curtas.
- Permutações circulares:
 - Ordenação por Reversões Super Curtas.

Ordenação por Reversões (Super) Curtas

Permutações Lineares

k-Reversão

Uma reversão $\rho(i, j)$ é dita ser uma k -reversão se $k = j - i + 1$.

Reversão Super Curta

Reversão super curta é uma k -reversão tal que $k \leq 2$.

Reversão Curta

Reversão curta é uma k -reversão tal que $k \leq 3$.

Resultados Conhecidos

- Ordenação por Reversões Super Curtas sem Sinal:
 - Jerrum¹⁵ apresentou algoritmo exato polinomial.
- Ordenação por Reversões Curtas sem Sinal:
 - Heath e Vergara¹⁶ apresentaram algoritmo 2-aproximado.

Nossa Contribuição

- Ordenação por Reversões Super Curtas com Sinal:
 - Algoritmo exato polinomial.
- Ordenação por Reversões Curtas com Sinal:
 - Algoritmo 5-aproximado.
 - Fator de aproximação esperado ≤ 3 para $n > 12$.

¹⁵M. R. Jerrum. *The complexity of finding minimum-length generator sequences*. *Theoretical Computer Science*, 36:265–289, 1985.

¹⁶L. S. Heath and J. P. C. Vergara. *Sorting by short swaps*. *Journal of Computational Biology*, 10(5):775–789, 2003.

Ordenação por Operações (Super) Curtas

Permutações Lineares

(x,y) -Transposição

Uma transposição $\rho(i, j, k)$ é dita ser uma (x, y) -transposição caso $x = j - i$ e $y = k - j$.

Operação Super Curta

Uma transposição super curta é uma (x, y) -transposição tal que $x + y = 2$. Uma operação super curta é uma reversão/transposição super curta.

Operação Curta

Uma transposição curta é uma (x, y) -transposição tal que $x + y \leq 3$. Uma operação curta é uma reversão/transposição curta.

Resultados Conhecidos

- Ordenação por Operações Super Curtas sem Sinal:
 - Algoritmo exato polinomial (Jerrum¹⁷).
- Ordenação por Operações Curtas sem Sinal:
 - Vergara¹⁸ apresentou algoritmo 2-aproximado.

Nossa Contribuição

- Ordenação por Operações Super Curtas com Sinal:
 - Algoritmo exato polinomial.
- Ordenação por Operações Curtas com Sinal:
 - Algoritmo 3-aproximado.

¹⁷ M. R. Jerrum. *The complexity of finding minimum-length generator sequences*. *Theoretical Computer Science*, 36:265–289, 1985.

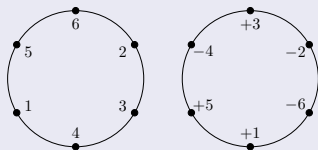
¹⁸ J. P. C. Vergara. *Sorting by Bounded Permutations*. PhD thesis, Virginia Polytechnic Institute & State University, 1998.

Ordenação por Reversões Super Curtas

Permutações Circulares

Permutação Circular

Uma permutação circular sem sinal é um arranjo dos elementos em $\{1, 2, \dots, n\}$ ao redor do círculo. Similarmente, uma permutação circular com sinal é um arranjo de n elementos em $\{-n, \dots, -2, -1, 1, 2, \dots, n\}$ ao redor do círculo tal que $|i| \neq |j|$ para quaisquer elementos distintos i e j no arranjo.



Reversão Super Curta

Uma reversão super curta troca de lugar dois elementos adjacentes. Caso a permutação tenha sinal, o sinal dos elementos é invertido.

Linearizando Permutações Circulares

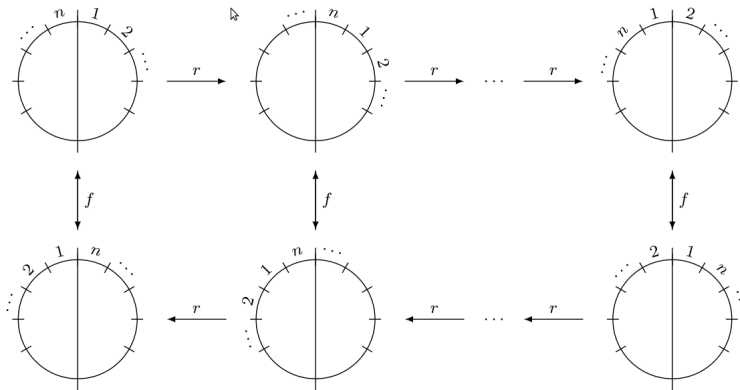


Figura: Alguns dos $2n$ pontos de vista para se observar uma permutação circular. Esta figura foi retirada do trabalho de Egri-Nagy *et al.*¹⁹

¹⁹ A. Egri-Nagy, V. Gebhardt, M. M. Tanaka, and A. R. Francis. *Group-theoretic models of the inversion process in bacterial genomes.* *Journal of Mathematical Biology*, 69(1):243–265, 2014.

Reversão Cíclica

Reversão cíclica é similar a uma reversão $\rho(i, j)$, isto é,

- i) *se $1 \leq i < j \leq n$, então é equivalente a uma reversão;*
- ii) *se $1 \leq j < i \leq n$, então ela transforma a permutação*

$$\left(\underline{\pi_1 \ \pi_2 \ \dots \ \pi_j} \ \pi_{j+1} \ \dots \ \pi_{i-1} \ \underline{\pi_i \ \pi_{i+1} \ \dots \ \pi_n} \right)$$

na permutação

$$\left(\underline{\pi_n \ \pi_{n-1} \ \dots \ \pi_i} \ \pi_{j+1} \ \dots \ \pi_{i-1} \ \underline{\pi_j \ \pi_{j-1} \ \dots \ \pi_1} \right).$$

Reversão Cíclica Super Curta

Reversão cíclica super curta é uma reversão cíclica $\rho(i, j)$ tal que $j - i \equiv 1 \pmod{n}$.

Reversão Cíclica com Sinal

Reversão cíclica com sinal é similar a uma reversão $\rho(i, j)$, isto é,

- i) *se $1 \leq i \leq j \leq n$, então é equivalente a uma reversão;*
- ii) *se $1 \leq j < i \leq n$, então ela transforma a permutação*

$$\left(\underline{\pi_1 \ \pi_2 \ \dots \ \pi_j} \ \pi_{j+1} \ \dots \ \pi_{i-1} \ \underline{\pi_i \ \pi_{i+1} \ \dots \ \pi_n} \right)$$

na permutação

$$\left(\underline{-\pi_n \ -\pi_{n-1} \ \dots \ -\pi_i} \ \pi_{j+1} \ \dots \ \pi_{i-1} \ \underline{-\pi_j \ -\pi_{j-1} \ \dots \ -\pi_1} \right).$$

Reversão Cíclica Super Curta com Sinal

Reversão cíclica super curta com sinal é uma reversão cíclica com sinal $\rho(i, j)$ tal que $j = i$ ou $j - i \equiv 1 \pmod{n}$.

Resultados Conhecidos

- Algoritmo exato polinomial para o caso sem sinal:
 - Solução do Jerrum²⁰: abordagem combinatória.
 - Solução de Egri-Nagy *et al.*²¹: abordagem algébrica.

Nossa Contribuição

- Algoritmo exato polinomial para o caso com sinal:
 - Extensão da solução do Jerrum.

²⁰ M. R. Jerrum. *The complexity of finding minimum-length generator sequences*. *Theoretical Computer Science*, 36:265–289, 1985.

²¹ A. Egri-Nagy, V. Gebhardt, M. M. Tanaka, and A. R. Francis. *Group-theoretic models of the inversion process in bacterial genomes*. *Journal of Mathematical Biology*, 69(1):243–265, 2014.

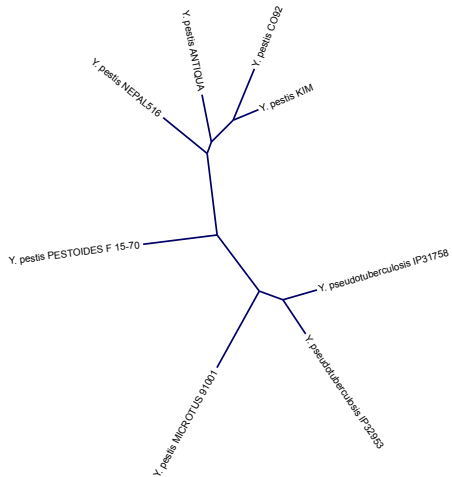
- O experimento:
 - Obtivemos, a partir do trabalho de Darling *et al.*²², permutações circulares com sinal que representam 8 genomas de bactérias *Yersinia*.
 - Computamos a matriz de distância de reversão super curta com sinal.
 - Construimos uma árvore filogenética usando Neighbor-Joining²³.
- Mesmo experimento feito por Egri-Nagy *et al.*²⁴, exceto que consideramos a orientação dos genes.

²² A. E. Darling, I. Miklós, and M. A. Ragan. *Dynamics of genome rearrangement in bacterial populations*. *PLoS Genetics*, 4(7):e1000128, 2008.

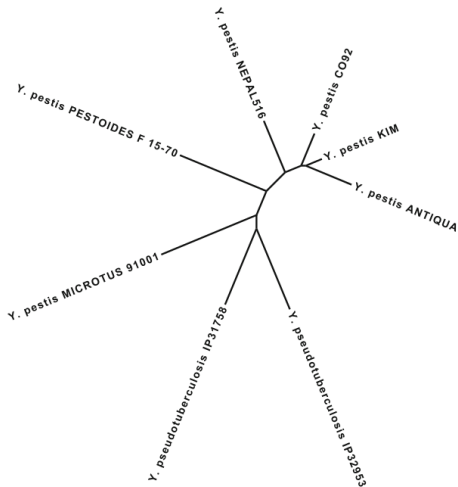
²³ N. Saitou and M. Nei. *The neighbor-joining method: a new method for reconstructing phylogenetic trees*. *Molecular Biology and Evolution*, 4(4):406–425, 1987.

²⁴ A. Egri-Nagy, V. Gebhardt, M. M. Tanaka, and A. R. Francis. *Group-theoretic models of the inversion process in bacterial genomes*. *Journal of Mathematical Biology*, 69(1):243–265, 2014.

Resultado



(a) Nossa filogenia



(b) Filogenia de Egri-Nagy *et al.*

Considerações Finais

- Resumo:
 - Ferramenta para validação e comparação.
 - Algoritmo para calcular as distâncias de rearranjo.
 - Base de dados de distâncias de rearranjo.
- Trabalhos futuros:
 - Algoritmos mais eficientes para calcular as distâncias.
 - Considerar permutações grandes.

- Resumo:
 - Heurística de melhoria.
 - Desempenho variável: menos de 5% a quase 100% de melhoria.
- Trabalhos futuros:
 - Integrar com o modelo de Programação Inteira de Lancia, Rinaldi e Serafini²⁵.

²⁵ G. Lancia, F. Rinaldi, and P. Serafini. A unified integer programming model for genome rearrangement problems. In *Proceedings of the IWBBIO'2015*, volume 9043 of *Lecture Notes in Computer Science*, pages 491–502, Granada, Spain, 2015. Springer International Publishing.

- Resumo:
 - Estudo de abordagens alternativas ao grafo de ciclos.
 - Melhorias teóricas.
 - Investigação experimental.
- Trabalhos futuros:
 - É possível chegar a uma aproximação < 2 ?

- Resumo:
 - Relevância do tamanho das operações.
 - Ordenação de Permutações com Sinais por Operações Curtas.
 - Algoritmos exatos e aproximados para cinco variações.
 - Aplicação: filogenia de bactérias *Yersinia*.
- Trabalhos futuros:
 - Algoritmos melhores (tempo e aproximação).
 - Considerar outras variações.
 - Mais experimentos com dados reais.

Artigos Publicados/Submetidos para Periódicos

Capítulo	Periódico	Qualis
2	ACM Journal of Experimental Algorithmics	B3
3	Journal of Bioinformatics and Computational Biology	B1
4	Journal of Universal Computer Science	B1
5	Algorithms for Molecular Biology	A1
6	IEEE/ACM Transactions on Computational Biology and Bioinformatics	A2

Artigos Apresentados em Conferências

Ano	Conferência
2012	7th Brazilian Symposium on Bioinformatics (BSB)
2014	5th ACM Conference on Bioinformatics, Computational Biology, and Health Informatics (ACM-BCB)
2015	11th International Symposium on Bioinformatics Research and Applications (ISBRA)