Rank Distance Sheds Light on Genome Evolution

Joao Meidanis [1] Leonid Chindelevitch [2]

[1]University of Campinas, Brazil

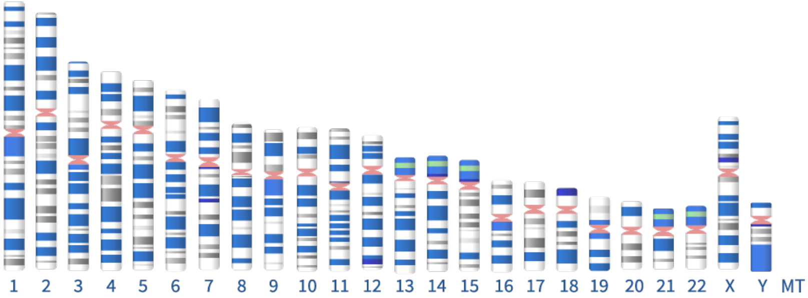
[2]Simon Fraser University, Canada

June 2019

Summary

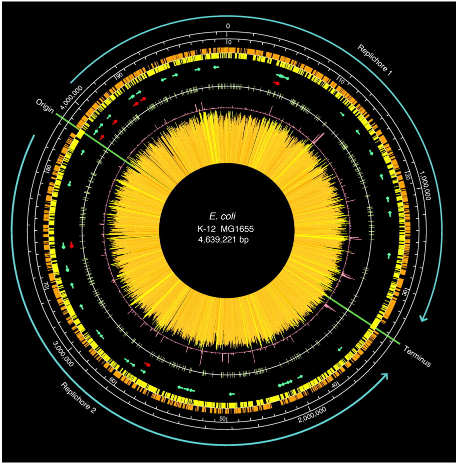
Genomes, Distances, Trees, Ancestors

The Human Genome



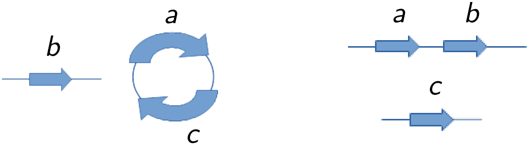
Source: National Center for Biotechnology Information (NCBI), USA

A Circular Genome: *E. coli*



Source: Science, 05 Sep 1997: Vol. 277, Issue 5331, pp. 1453-1462

General Scheme



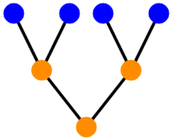
Genomes

Distances

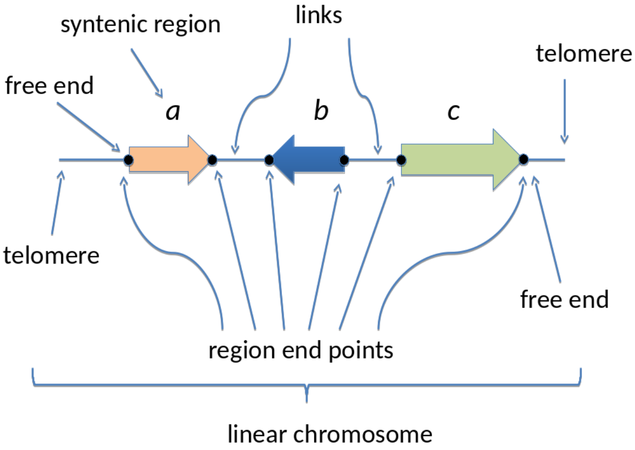
rearrangement distance $= 3$

Trees

Ancestors



Genome elements



- Links: $\{a_h, b_h\}, \{b_t, c_t\}$; free ends: a_t, c_h

Representing genomes as matrices

- Links: $\{a_h, b_h\}, \{b_t, c_t\}$; free ends: a_t, c_h

$$
\begin{array}{c} \\ a_t \\ a_h \\ b_t \\ b_h \\ c_t \\ c_h \end{array}
\begin{array}{c} \begin{array}{cccccc} a_t & a_h & b_t & b_h & c_t & c_h \end{array} \\
\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{array}
$$

Properties

- symmetric matrix ($A = A^t$)
- orthogonal matrix ($A^t = A^{-1}$)
- involution ($A^2 = I$)

Distance

- Distance between two genome matrices is the rank of their difference

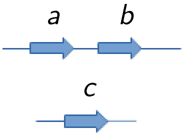
$$d(A, B) = r(A - B)$$

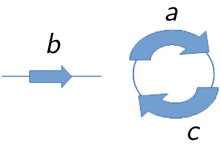
Properties

- Rank is the maximum number of linearly independent rows
- $d(A, B) = 0$ if and only if $A = B$
- $d(A, B) = d(B, A)$
- $d(A, C) \leq d(A, B) + d(B, C)$

Example

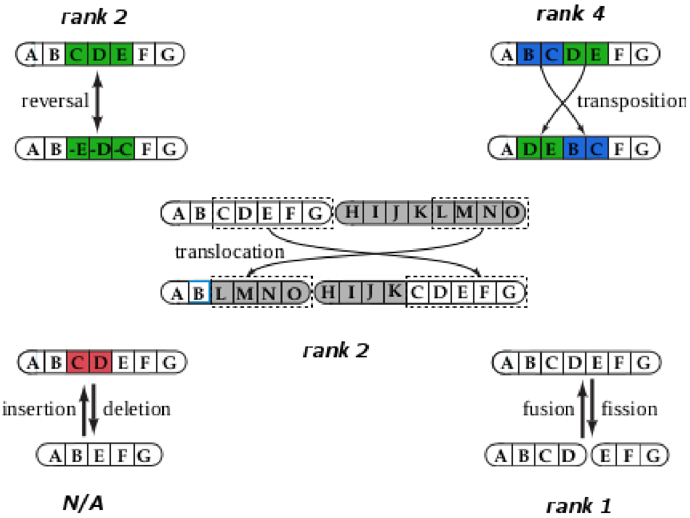
$$\begin{array}{c} a_t \\ a_h \\ b_t \\ b_h \\ c_t \\ c_h \end{array} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$





$$\begin{matrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{matrix} - \begin{matrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{matrix} = \begin{matrix} 1 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 1 \end{matrix}$$

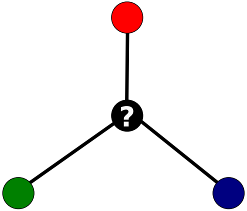
Genome Rearrangements



Ancestors: The Median Problem

Median Problem

Useful for ancestor reconstruction



Definition

Given three input genome matrices A, B, and C, find matrix M minimizing $d(M, A) + d(M, B) + d(M, C)$.

Median may not be genomic

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \quad \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

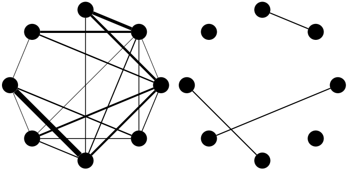
$$\downarrow$$

$$\begin{bmatrix} -0.5 & 0.5 & 0.5 & 0.5 \\ 0.5 & -0.5 & 0.5 & 0.5 \\ 0.5 & 0.5 & -0.5 & 0.5 \\ 0.5 & 0.5 & 0.5 & -0.5 \end{bmatrix}$$

- Need ways to go back from matrices to genomes

From matrices back to genomes

$$\begin{bmatrix} 0.2 & 0.8 & 0.5 & 0 & 0 & 0.4 & 0 & 0.1 \\ 0.4 & 0 & 0 & 0 & 0 & 0.3 & 0 & 0.6 \\ 0.3 & 0 & 0.5 & 0.2 & 0 & 0 & 0 & 0.3 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0.1 & 0 & 0 & 0.1 & 0.1 & 0.4 & 0.2 & 0.7 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0.3 & 0 & 0 & 0.5 & 0.1 & 0 & 0.4 & 0.1 \\ 0 & 0.8 & 0.2 & 0 & 0 & 0.8 & 0.2 & 0.3 \end{bmatrix}$$



$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

- Assign weight $|a_{ij}| + |a_{ji}|$ to edge ij
- Take a maximum weight matching as your solution
- A genome is a matching of gene extremities

Genomic Median is NP-hard

If we insist on genomic medians:

Definition

Given three input genome matrices A, B, and C, find a **genomic** matrix M minimizing $d(M, A) + d(M, B) + d(M, C)$.

Then the problem becomes HP-hard

Sarkis *et al.*, 2019, submitted

Orthogonal Median is Polynomially Solvable

Definition

Given three input genome matrices A, B, and C, find an **orthogonal** matrix M minimizing $d(M, A) + d(M, B) + d(M, C)$.

Orthogonal algorithm: finds many solutions fast (nondeterministic)

Chindelevitch, Zanetti, Meidanis. *BMC Bioinformatics 2018*

M_I **algorithm**: finds one solution faster (deterministic)

Chindelevitch, Meidanis. *RECOMB-CG 2018*

Practical Experiments

Data Sets

Simulation

- Start with random genome
- Apply random rearrangement operations
- Repeat to get A, B, C

Parameters

- sizes: 12, 16, 20, 30, 50, 100, 200, 300, 500, 100 extremities
- type of operation: Add/remove adjacencies (near) or DCJ (far)
- number of operations: 5% to 30%
- $10 \times$ each
- 1,080 instances

Results

Near

- For 595/600 instances, the algorithms find genomic medians
- In 5 remaining cases, heuristics find genomic medians

Far

- For 263 cases, the algorihtms find genomic medians
- In 135 remaining cases, heuristics find genomic medians (diff 0–21, avg 3)
- In 102 remaining cases, heuristics find genomic medians (diff 1–173, avg 19)

Running Times

- M_I algorithm: 1 second, $n = 500$ (cubic algorithm)
- Orthogonal: 1 minute, $n = 500$ (quartic algorithm)

Campanulaceae, family of flowering plants

Real genomes

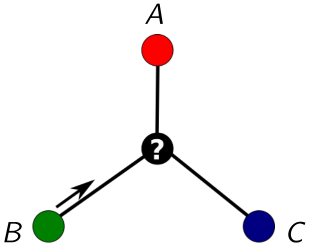
Campanulaceae chloroplast genomes

- 286 instances
- $n = 210$ extremities
- Score of output very close to theoretical minimum (1% off in average)
- Running time close to 1 sec per instance

Algorithms

Orthogonal algorithm

- Specific for **orthogonal matrices**
- Exact, efficient algorithm



- "Walk towards the median"
- Find rank 1 matrix H such that $B + H$ is closer to both A and C
- Always possible!

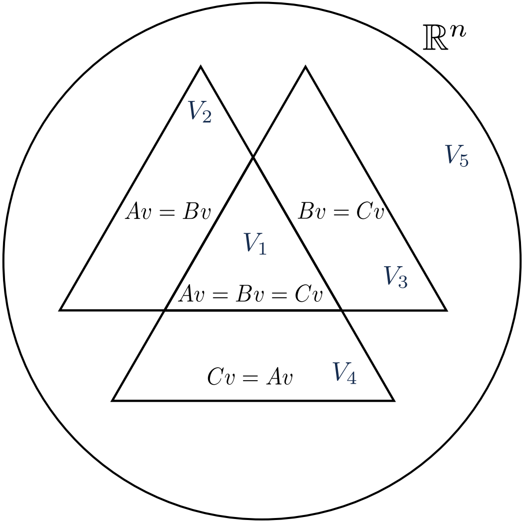
Orthogonal algorithm

- Algorithm

while $d(A, B) + d(B, C) > d(A, C)$ **do**
  Find non-zero $u \in \text{im}(A - B) \cap \text{im}(C - B)$
  $B \leftarrow B - 2uu^T B / u^T u$
end
return B

- Nondeterministic
- Reaches **all orthogonal** medians

Division into subspaces



M_I Median — $O(n^\omega)$

- Specific for **genome matrices**
- M_I follows majority in V_1 through V_4
- M_I follows I in V_5

Subspaces	V_1	V_2	V_3	V_4	V_5
	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$
Bases	B_1	B_2	B_3	B_4	B_5
	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$
Projection Matrices	P_1	P_2	P_3	P_4	P_5
	$\swarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\searrow$

Median $\quad M_I = AP_1 + AP_2 + BP_3 + AP_4 + IP_5$

Genomes with Unequal Content

Genomes with Indels (Insertions/Deletions)

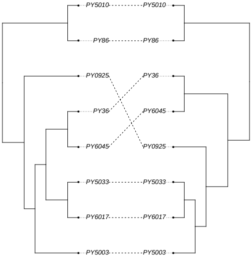
- Assume a universal ground set with all gene ends
- A missing gene end is represented by a 0 row and 0 column
- A: Genes: $\{a, b, d\}$; links: $\{a_h, b_t\}, \{b_h, d_h\}$
- B: Genes: $\{b, c, d\}$; links: $\{b_h, c_t\}, \{c_h, d_t\}$

$$A - B = \begin{array}{c} a_t \\ a_h \\ b_t \\ b_h \\ c_t \\ c_h \\ d_t \\ d_h \end{array} \begin{array}{c} \begin{array}{cccccccc} a_t & a_h & b_t & b_h & c_t & c_h & d_t & d_h \end{array} \\ \left[\begin{array}{cccccccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 \end{array} \right] \end{array}$$

Genomes with Indels

- Fast distance computation
- Biological interpretation seems to require **semi-chromosomes**: a tail without a head or vice-versa
- Initial tests with fungal genomes (∼6000 genes) are encouraging

rank distance



classical method

- Next step: median

Future Work

Future Work

Main challenges

- Incorporate point mutations + rearrangements in analysis
- Study median problem with indels
- Interpretation of fractional/negative entries in matrices
- Interpetation of semi-chromosomes

Get this presentation:

`http://www.ic.unicamp.br/~meidanis/research/rear/`