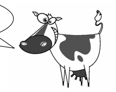
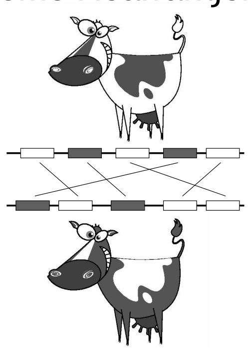
Computational Biology

Lecture 16



Saad Mneimneh

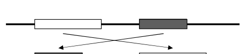
Genome Rearrangements



Saad Mneimneh

Genome Rearrangement

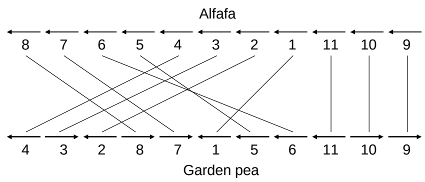
- Especially when comparing different species, a piece of chromosome can be moved or copied to another location.



- So far we used alignment algorithms to compare two sequences, possibly coming from different species, but alignments do not capture genome rearrangements
  - Computing local alignments could capture this notion (we will see later) but does not tell how one chromosome is obtained from the other
- A genome is obtained from another by a number of a special kind of rearrangements: *Reversals*
- Distance between two genomes: minimum number of reversals needed

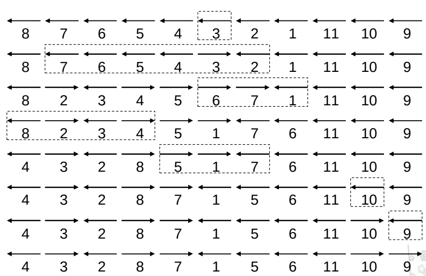
Saad Mneimneh

Example



- Each block, possibly containing more than one gene, is transcribed as a unit
- Orientation of a block denotes the strand where transcription occurs
- Blocks with the same number are homologous

Reversals



Formalism

- A signed permutation α over the set $L = \{1, 2, \ldots, n\}$ is a permutation such that $\alpha(i) = +a$ or $-a$, where $a \in L$.
- Example: +3, –2, –1, +4, –5 is a signed permutation over $L = \{1, 2, 3, 4, 5\}$.
- A reversal $[i,j]$ ($i \leq j$) of a signed permutation α, is a signed permutation

$$\alpha' = \alpha[i,j] = \alpha(1), \ldots, \alpha(i-1), -\alpha(j), \ldots, -\alpha(i), \alpha(j+1), \ldots, \alpha(n)$$

- *Sorting by reversals*: Given two signed permutations α and β, find the **minimum** number of reversals $\rho_1, \ldots, \rho_t$ that will transform α into β, i.e.

$$\alpha\rho_1\ldots\rho_t = \beta$$

- Define reversal distance $d_\beta(\alpha) = t$ ($d_\beta(\alpha) = d_\alpha(\beta)$)

Question…

Q: Is it always possible to transform α into β by using reversals only?

A: Yes

How?

If $\beta(1) = -\alpha(i)$, then let $\alpha' = \alpha[1,i]$.
If $\beta(1) = +\alpha(i)$, then let $\alpha' = \alpha[1,i][1,1]$.

Now $\beta(1) = \alpha'(1)$.

Work similarly on $\beta(2), \dots, \beta(n)$ and $\alpha'(2), \dots, \alpha'(n)$

Breakpoints

- Let $\alpha = \alpha(1), \dots, \alpha(n)$
- The extended version of α is
  $\alpha(0), \alpha(1), \dots, \alpha(n), \alpha(n+1) = 0, \alpha(1), \dots, \alpha(n), n+1$
- if (x,y) appears in the extended version of α but neither (x,y) nor (–y,–x) appear in β, then (x,y) is a breakpoint of α with respect to β.
- Example:
  extended α: 0, –2, –3, +1, +6, –5, –4, 7
  extended β: 0, +1, +2, +3, +4, +5, +6, 7

  Breakpoints: (0,–2), (–2,–3), (–3,+1), (+1,+6), (6,–5), (–4,7)

  $b_\beta(\alpha) = 6$

  $b_\beta(\beta) = 0$ (always true)

A lower bound

- If (x,y) is a breakpoint of α, then in order to transform α into β, some reversal must separate x and y.
- A reversal can reduce the number of breakpoints by at most 2.
- Let $\rho_1, \dots, \rho_t$ be such that $\alpha\rho_1 \dots \rho_t = \beta$, then:

$$\left.\begin{array}{l} b(\alpha) - b(\alpha\rho_1) \le 2 \\ b(\alpha\rho_1) - b(\alpha\rho_1\rho_2) \le 2 \\ \dots \\ b(\alpha\rho_1 \dots \rho_{t-1}) - b(\alpha\rho_1 \dots \rho_t) \le 2 \end{array}\right| \Rightarrow b(\alpha) - b(\alpha\rho_1 \dots \rho_t) = b(\alpha) \le 2t$$

- Let $t = d(\alpha)$, then $\boxed{d(\alpha) \ge b(\alpha)/2}$

Another representation

- Represent each element in the permutation as a tuple:

  $+a$: (–a +a) 
  $-a$: (+a –a) 

- A permutation is a sequence of adjacent tuples:

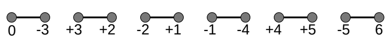
  Example: α = +3, –2, –1, +4, –5 can be represented as:

  α = 0---(–3 +3)---(+2 –2)---(+1 –1)---(–4 +4)---(+5 –5)---6

Saad Mneimneh

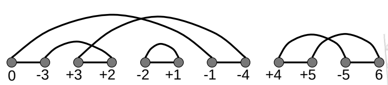
Graph representation

α = 0---(–3 +3)---(+2 –2)---(+1 –1)---(–4 +4)---(+5 –5)---6



β = 0---(–1 +1)---(–2 +2)---(–3 +3)---(–4 +4)---(–5 +5)---6

We can represent both α and βon the same (multi) graph

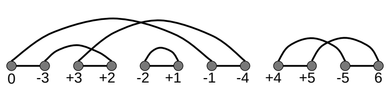


Saad Mneimneh

Diagram of Reality and Desire

- α is the reality
- β is what is desired

- The diagram $RD_\beta(\alpha)$ is the diagram of reality and desire:

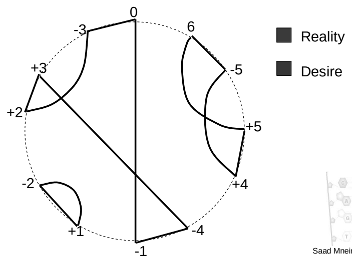


Saad Mneimneh

Diagram of Reality and Desire

(for better visualization)

- Extended α: 0, +3, –2, –1, +4, –5, 6
- Extended β: 0, +1, +2, +3, +4, +5, 6



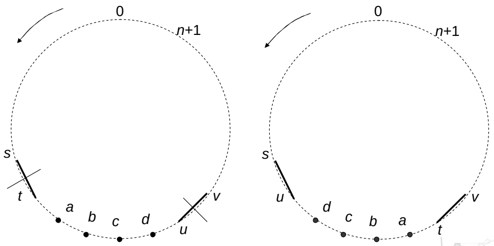
Properties of $RD_\beta(\alpha)$

- Each vertex has degree 2
- The connected components of the graph are alternating cycles (edges alternate between reality and desire)
- Let $c_\beta(\alpha)$ be the number of cycles in $RD_\beta(\alpha)$
- $c_\beta(\beta) = n+1$ (each desire is a reality)
- Therefore, transforming α into β can be seen as transforming $RD_\beta(\alpha)$ into a graph with as many cycles as possible (i.e. $n+1$).

Reversals and RD(α)

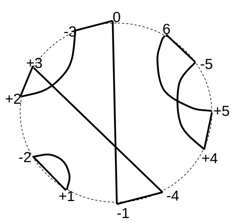
- A reversal is characterized by the two points where it cuts the permutation, each defined by a reality edge.
- Let ρ be a reversal defined by two reality edges (s,t) and (u,v), then $RD(\alpha\rho)$ differs from $RD(\alpha)$ as follows:
  - Reality edges (s,t) and (u,v) are replaced by (s,u) and (t,v)
  - Desire edges remain unchanged
  - Vertices $u, \ldots, t$ are reversed

Illustration



Definition

Two reality edges on the same cycle *converge* if they are traversed in the same direction (either clockwise or counterclockwise) on the cycle.



Example:

(+3,+2) and (-1,-4) converge

(0,-3) and (+3,+2) diverge

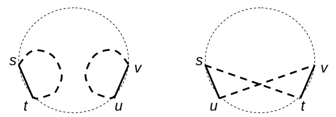
Reversals and $c(\alpha)$

Let ρ be a reversal defined by two reality edges e and f, then:

- If e and f belong to different cycles, $c(\alpha\rho) = c(\alpha) - 1$
- If e and f belong to the same cycle and converge, $c(\alpha\rho) = c(\alpha)$
- If e and f belong to the same cycle and diverge, $c(\alpha\rho) = c(\alpha) + 1$

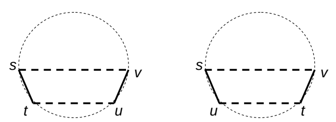
Proof – case 1

If *e* and *f* belong to different cycles,
$c(\alpha\rho) = c(\alpha) - 1$



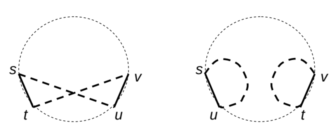
Proof – case 2

If *e* and *f* belong to the same cycle and converge, $c(\alpha\rho) = c(\alpha)$



Proof – case 3

If *e* and *f* belong to the same cycle and diverge,
$c(\alpha\rho) = c(\alpha) +1$



Another lower bound

- Let $\rho_1, \ldots, \rho_t$ be such that $\alpha\rho_1 \ldots \rho_t = \beta$, then:

$$
\left.\begin{array}{l}
c(\alpha\rho_1) - c(\alpha) \le 1 \\
c(\alpha\rho_1\rho_2) - c(\alpha\rho_1) \le 1 \\
\ldots \\
c(\alpha\rho_1 \ldots \rho_t) - c(\alpha\rho_1 \ldots \rho_{t-1}) \le 1
\end{array}\right| \Rightarrow c(\alpha\rho_1 \ldots \rho_t) - c(\alpha) = n + 1 - c(\alpha) \le t
$$

- Let $t = d(\alpha)$, then $\boxed{\boldsymbol{d(\alpha) \ge n + 1 - c(\alpha)}}$

Better lower bound

- The lower bound $n + 1 - c(\alpha)$ is better than $b(\alpha)/2$.
- For most signed permutations, it comes very close to the actual reversal distance.
- It does not always work. We cannot always choose two divergent edges (doing so will increase the number of cycles by 1 each time and achieve the exact bound).
- To understand why, we need to define some concepts.

Good / Bad cycles

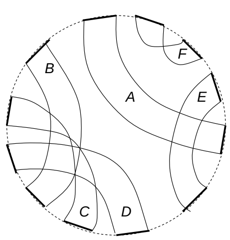
The easy stuff first….

- A cycle is **good** iff it has two reality edges that diverge
- A cycle is **bad** iff all its reality edges converge
- A **proper** cycle is a cycle with at least 4 edges
  – We need not worry about non-proper cycles (2 edges only), they represent the non-breakpoints (reality = desire)

Interleaving

- Assume the special way of drawing *RD*(α)
  - either the counterclockwise circle where reality edges are on the circumference and desire edges are chords
  - or reality edges are on the line and desire edges on top
- Two cycles interleave iff a desire edge of one crosses a desire edge of the other
- In general we can define this using intervals without assuming a special way of drawing *RD*(α)

Example



A and *E* interleave

B and *C* interleave

C and *D* interleave

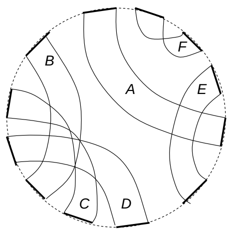
B and *D* interleave

Components

Construct the interleaving graph

- Vertices are the cycles
- An edge exists between two vertices if the corresponding cycles interleave
- The connected components of the interleaving graph are the components

Example



We have three components:

{F}

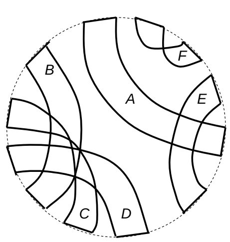
{A, E}

{B, C, D}

Good / Bad components

- A component is good iff it contains at least one good cycle
- A component is bad iff all its cycles are bad

Example



We have three components:

{F} : good

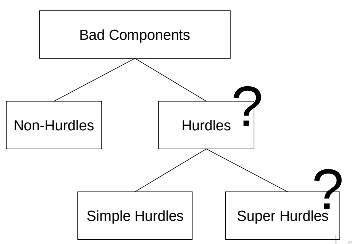
{A, E} : bad

{B, C, D} : good

Bad component

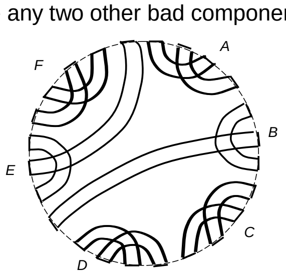
- A reversal that increases the number of cycles must act on a good cycle in a (good) component
- Such a reversal can possibly transform a bad component into a good one by twisting some cycles
- A cycle will be twisted when performing a reversal on another only if the two cycles interleave (i.e. same component)
- Therefore, a bad component will remain until we perform a reversal on one of its cycles, or a reversal on two distinct cycles
- In either case we do not increase the number of cycles
- Therefore, if $RD(\alpha)$ contains a bad component, then we will have to perform a reversal that does not increase the number of cycles and the lower bound $n + 1 - c(\alpha)$ will be exceeded.

Classification of bad components



Hurdles

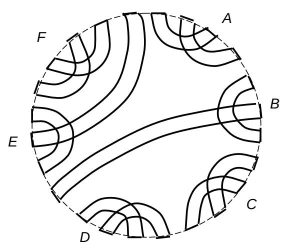
A bad component is a hurdle iff it does not separate any two other bad components



Super / Simple hurdle

- A super hurdle is a hurdle *A* such its removal will cause a non-hurdle *B* to become a hurdle, we say *A protects B*
- Otherwise, a hurdle is a simple hurdle

Hurdles

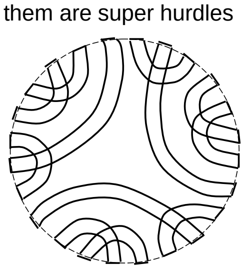


F is a super hurdle, it protects *E*

All other hurdles are Simple hurdles

Fortress

A signed permutation α is called a fortress iff $RD(\alpha)$ has an odd number of hurdles and all of them are super hurdles



The smallest fortress, with 3 super hurdles

Next time…

$d(\alpha) = n + 1 - c(\alpha) + h(\alpha)$, α non-fortress

$d(\alpha) = n + 1 - c(\alpha) + h(\alpha) + 1$, α fortress

where $h(\alpha)$ is the number of hurdles

and an algorithm…