Algorithms in Comparative Genomics

Universität Bielefeld, WS 2021/2022
Dr. Marília D. V. Braga · Leonard Bohnenkämper

https://gi.cebitec.uni-bielefeld.de/teaching/2021winter/cg

Exercise sheet 6, 25.11.2021

Exercise 1 (Canonical inversion model) (7 pts)

Given the canonical circular chromosomes

$$\mathbb{A} = (\ 1\ 10\ \overline{12}\ 11\ 13\ 15\ 14\ 16\ 2\ 4\ 3\ 5\ 8\ 7\ 6\ 9\ 17\ ) \quad \text{and}$$

$$\mathbb{B} = (\ 1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ 10\ 11\ 12\ 13\ 14\ 15\ 16\ 17\ ).$$

1. Draw the breakpoint diagram $BD(\mathbb{A}, \mathbb{B})$.
2. Identify all cycles on the diagram, giving their lengths and their types (good / bad).
3. Identify all interleaving components on the diagram, giving their numbers of cycles and their types (good / bad / hurdle / super hurdle). Is the diagram of $\mathbb{A}$ and $\mathbb{B}$ a fortress?
4. What is the inversion distance $d_{\text{INV}}(\mathbb{A}, \mathbb{B})$?

Exercise 2 (Optimal cover of component tree) (7 pts)

Given canonical circular chromosomes

$$\mathbb{A} = (0\ 2\ \overline{7}\ 6\ \overline{5}\ 3\ 4\ 8\ 10\ \overline{12}\ 9\ \overline{11}\ 13\ \overline{1}\ 14\ 16\ 21\ 17\ 19\ 18\ 20\ 22\ 27\ 23\ 25\ 24\ 26\ 28\ 15)$$

$$\mathbb{B} = (0\ 1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ 10\ 11\ 12\ 13\ 14\ 15\ 16\ 17\ 18\ 19\ 20\ 21\ 22\ 23\ 24\ 25\ 26\ 27\ 28)$$

1. Draw the relational (or the breakpoint) diagram of $\mathbb{A}$ and $\mathbb{B}$.

   (You can use the Java program InversionVisualization provided on the course website: enter the values for chromosome $\mathbb{A}$, without the first value (0) and assume that the first vertex of the outputted diagram is 0^h and the last vertex is 0^t.)
2. Based on the diagram, construct both the chained component tree $\Upsilon_{\blacksquare}(\mathbb{A}, \mathbb{B})$ and the component tree $\Upsilon_{\circ}(\mathbb{A}, \mathbb{B})$.
3. Find an optimal cover (i.e. a cover with minimum cost) for $\Upsilon_{\circ}(\mathbb{A}, \mathbb{B})$.
4. Compute the inversion distance $d_{\text{INV}}(\mathbb{A}, \mathbb{B})$.

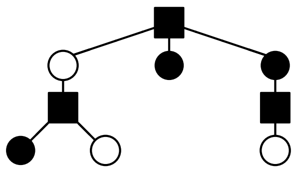
Exercise 3 (Canonical inversion sorting) (4 pts)

Sort circular chromosome $\mathbb{A} = (0\ 3\ \overline{1}\ \overline{4}\ 2)$ into $\mathbb{B} = (0\ 1\ 2\ 3\ 4)$. Indicate all intermediate steps by drawing the respective overlap graphs, annotating each vertex with its corresponding score. Indicate your choice of a safe inversion by marking the corresponding vertex in the overlap graph.

(You may use InversionVisualization again! - again omitting the first value (0) and assuming that the first vertex of the outputted diagram is 0^h and the last vertex is 0^t.)

Exercise 4 (From a chained component tree to canonical chromosomes) (4 pts)

Given the following chained component tree $\Upsilon'_{\blacksquare}$:



Find canonical circular chromosomes $\mathbb{A}$ and $\mathbb{B}$ with the minimum number of genes such that

$$\Upsilon'_{\blacksquare} = \Upsilon_{\blacksquare}(\mathbb{A}, \mathbb{B}).$$

Exercise 5 (Swaping elements) (6* extra pts)

let $\iota(n) = \langle 1, 2, 3 \ldots n-1, n \rangle$ be the *identity* permutation of n elements.

1. Let $\pi(n)$ be any permutation of n elements - for instance, we could have $\pi(5) = \langle 2, 5, 1, 3, 4 \rangle$.

   The *swap* operation simply swaps two consecutive elements of a permutation.

   Give an algorithm that finds the minimum number of swap operations that sort a permutation $\pi(n)$ into the identity $\iota(n)$.

2. Not let $\sigma(n)$ be a signed permutation of n elements - for instance, we could have $\sigma(5) = \langle 2, -5, -1, 3, 4 \rangle$.

   The *signed swap* operation swaps two consecutive elements of a signed permutation and inverts their signs.

   Give an algorithm that finds the minimum number of signed swap operations that sort a signed permutation $\sigma(n)$ into the identity $\iota(n)$.