

Topics of today:

1. Canonical DCJ distance and sorting
2. Relational graph
3. Restricted canonical DCJ sorting

Canonical DCJ

Given two canonical genomes \mathbb{A} and \mathbb{B} ,...

Canonical DCJ Distance Problem: Compute the minimum number of DCJ operations required to transform \mathbb{A} into \mathbb{B} .

Denote by $d_{\text{DCJ}}(\mathbb{A}, \mathbb{B})$ the DCJ distance of \mathbb{A} and \mathbb{B} .

Canonical DCJ Sorting Problem: Find a sequence of $d_{\text{DCJ}}(\mathbb{A}, \mathbb{B})$ DCJ operations that transform \mathbb{A} into \mathbb{B} .

Let $\xi(\mathbb{G}) = \{g^t : g \in \mathcal{G}(\mathbb{G})\} \cup \{g^h : g \in \mathcal{G}(\mathbb{G})\}$ be the set of extremities of all genes in genome \mathbb{G} .

Ex: $\mathbb{G} = [2 \bar{3} 1]$, $\mathcal{G}(\mathbb{G}) = \{1, 2, 3\}$ and $\xi(\mathbb{G}) = \{1^t, 1^h, 2^t, 2^h, 3^t, 3^h\}$.

Note that, if genomes \mathbb{A} and \mathbb{B} are canonical, then $\xi(\mathbb{A}) = \xi(\mathbb{B})$.

Relational graph of canonical genomes

Given two canonical genomes \mathbb{A} and \mathbb{B} , their **relational graph** $RG(\mathbb{A}, \mathbb{B}) = (V, E)$ is described as follows:

1. $V = V(\xi(\mathbb{A})) \cup V(\xi(\mathbb{B}))$: there is a vertex for each extremity of each gene in \mathbb{A}
and a vertex for each extremity of each gene in \mathbb{B}

Each vertex v has a label $\ell(v)$, that corresponds to the extremity it represents.

For a given gene g , let $\begin{cases} \text{vertices } u \text{ and } v \text{ represent } g^t \text{ and } g^h \text{ in genome } \mathbb{A} \text{ and} \\ \text{vertices } u' \text{ and } v' \text{ represent } g^t \text{ and } g^h \text{ in genome } \mathbb{B} \end{cases}$

Then: $\ell(u) = \ell(u') = g^t$ and $\ell(v) = \ell(v') = g^h$

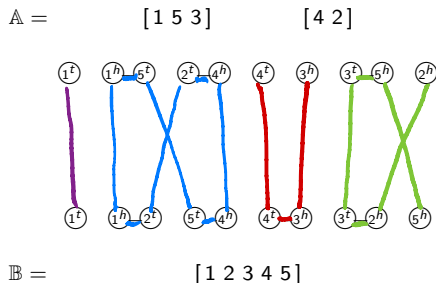
2. $E = E_\alpha(\mathbb{A}) \cup E_\alpha(\mathbb{B}) \cup E_\xi$, where:

- **Adjacency edges:** $\begin{cases} E_\alpha(\mathbb{A}) = \{uv : u, v \in V(\xi(\mathbb{A})) \text{ and } \ell(u)\ell(v) \in \alpha(\mathbb{A})\} \\ E_\alpha(\mathbb{B}) = \{uv : u, v \in V(\xi(\mathbb{B})) \text{ and } \ell(u)\ell(v) \in \alpha(\mathbb{B})\} \end{cases}$
- **Extremity edges:** $E_\xi = \{uv : u \in V(\xi(\mathbb{A})) \text{ and } v \in V(\xi(\mathbb{B})) \text{ and } \ell(u) = \ell(v)\}$

Note that:

- Let $n = |\mathcal{G}_\star|$. The number of edges in E_ξ is $2n$ (two edges per element of \mathcal{G}_\star).

Relational graph of canonical genomes



$$n = |\mathcal{G}_\star| = 5, \quad \kappa(\mathbb{A}) = 2 \quad \text{and} \quad \kappa(\mathbb{B}) = 1$$

Every vertex has degree one or two:

$RG(\mathbb{A}, \mathbb{B})$ is a collection of paths and cycles (alternating edges in E_ξ and in $E_\alpha(\mathbb{A}) \cup E_\alpha(\mathbb{B})$)

cycle with k edges in E_ξ : k -cycle or c_k

path with k edges in E_ξ : k -path or p_k

$$\left\{ \begin{array}{l}
 \mathcal{C} = \{c_k\} : \text{set of cycles } (k \text{ is even}) \\
 \mathcal{P}_{\mathbb{A}\mathbb{A}} = \{p_k : \text{starts and ends in } \mathbb{A}\} : \\
 \quad \text{set of } \mathbb{A}\mathbb{A}\text{-paths } (k \text{ is even}) \\
 \mathcal{P}_{\mathbb{B}\mathbb{B}} = \{p_k : \text{starts and ends in } \mathbb{B}\} : \\
 \quad \text{set of } \mathbb{B}\mathbb{B}\text{-paths } (k \text{ is even}) \\
 \mathcal{P}_{\mathbb{A}\mathbb{B}} = \{p_k : \text{starts in } \mathbb{A} \text{ and ends in } \mathbb{B}\} : \\
 \quad \text{set of } \mathbb{A}\mathbb{B}\text{-paths } (k \text{ is odd})
 \end{array} \right.$$

$|\mathcal{P}_{\mathbb{A}\mathbb{B}}|$ is even (E_ξ has $2n$ edges)

$$|\mathcal{P}_{\mathbb{A}\mathbb{A}}| + |\mathcal{P}_{\mathbb{B}\mathbb{B}}| + |\mathcal{P}_{\mathbb{A}\mathbb{B}}| = \kappa(\mathbb{A}) + \kappa(\mathbb{B})$$

If $\mathbb{A} = \mathbb{B}$,

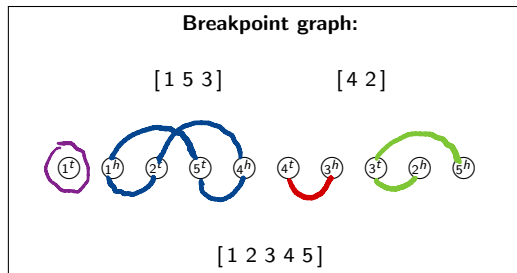
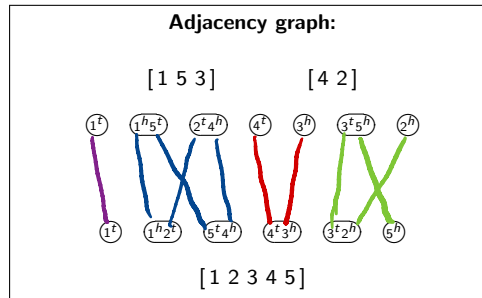
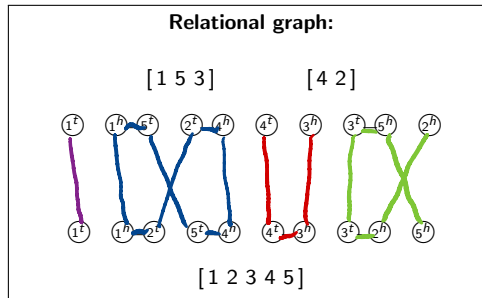
$RG(\mathbb{A}, \mathbb{B})$ has only 2-cycles and 1-paths:

$$2n = 2|\mathcal{C}| + |\mathcal{P}_{\mathbb{A}\mathbb{B}}| \Rightarrow n = |\mathcal{C}| + \frac{|\mathcal{P}_{\mathbb{A}\mathbb{B}}|}{2}$$

Otherwise, if $\mathbb{A} \neq \mathbb{B}$:

$$n > |\mathcal{C}| + \frac{|\mathcal{P}_{\mathbb{A}\mathbb{B}}|}{2}$$

Relational graph \cong Adjacency graph \cong Breakpoint graph

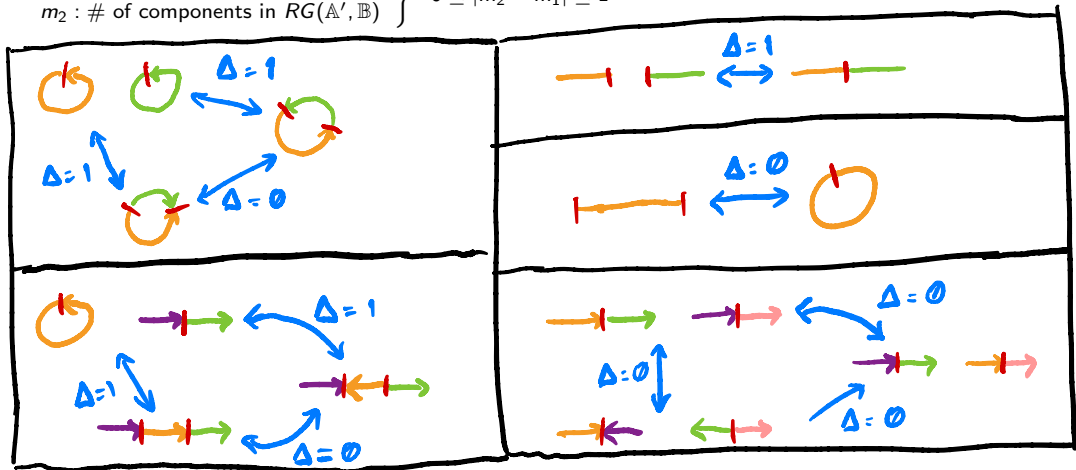


DCJ operations
are applied
only in A
or
only in B

Types of DCJ operation

Let a DCJ operation transform a genome \mathbb{A} into another genome \mathbb{A}' :

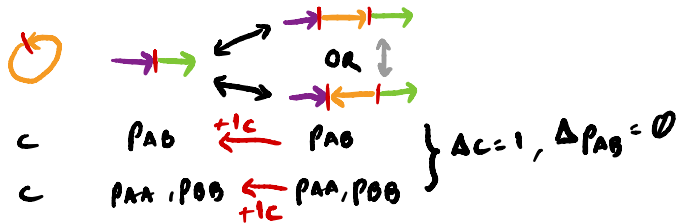
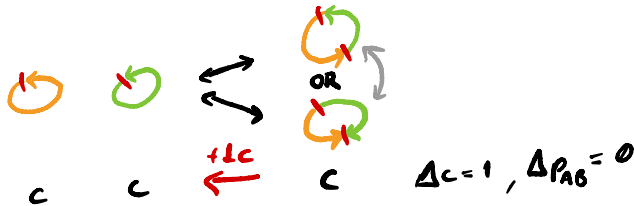
$$\left. \begin{array}{l} m_1 : \# \text{ of components in } RG(\mathbb{A}, \mathbb{B}) \\ m_2 : \# \text{ of components in } RG(\mathbb{A}', \mathbb{B}) \end{array} \right\} 0 \leq \overbrace{|m_2 - m_1|}^{\Delta} \leq 1$$



Goal: increase the number of cycles ($|\mathcal{C}|$) and/or the number of $\mathbb{A}\mathbb{B}$ -paths ($|\mathcal{P}_{\mathbb{A}\mathbb{B}}|$) in RG

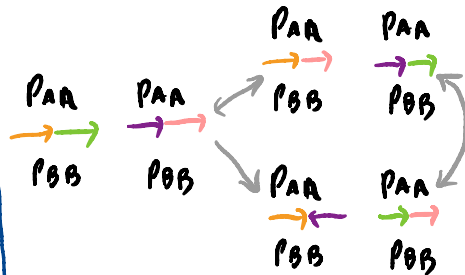
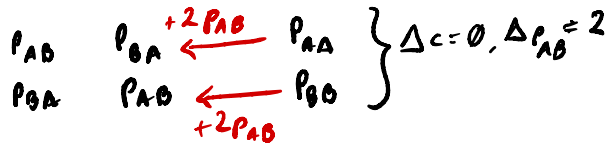
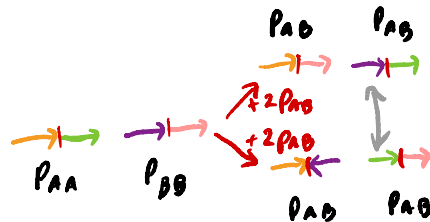
Types of DCJ operation

Goal: increase the number of cycles ($|C|$) and/or the number of AB -paths ($|\mathcal{P}_{AB}|$) in RG



Types of DCJ operation

Goal: increase the number of cycles ($|C|$) and/or the number of AB -paths ($|\mathcal{P}_{AB}|$) in RG



Canonical DCJ Distance & Sorting

Recall that, if $\mathbb{A} = \mathbb{B}$, we have $n = |\mathcal{C}| + \frac{|\mathcal{P}_{\mathbb{A}\mathbb{B}}|}{2}$, otherwise $n > |\mathcal{C}| + \frac{|\mathcal{P}_{\mathbb{A}\mathbb{B}}|}{2}$

A DCJ operation ρ is called **optimal** if $\left\{ \begin{array}{l} \rho \text{ increases the number of cycles by one, or} \\ \rho \text{ increases the number of } \mathbb{A}\mathbb{B}\text{-paths by two.} \end{array} \right.$

Given two canonical genomes \mathbb{A} and \mathbb{B} , it is possible to find an optimal DCJ operation at each sorting step.
Therefore:

$$d_{\text{DCJ}}(\mathbb{A}, \mathbb{B}) = n - |\mathcal{C}| - \frac{|\mathcal{P}_{\mathbb{A}\mathbb{B}}|}{2}$$

Quiz 1

1 Which of the following statements about the Relational Graph are true?

odd: AB

even: AA or BB

- ☒ A Closing an even path into a cycle is always optimal.
- ☒ B Breaking an odd path into two paths is always optimal.
- ☒ C Breaking an even path into two odd paths is always optimal.
- ☒ D Breaking an even cycle into two cycles is always optimal.
- ☒ E Recombining two even paths into two odd paths is always optimal.

Compute the DCJ distance for the following pairs of genomes:

2 $A = [1 \ 3 \ 2 \ 4]$ and
 $B = [1 \ 2 \ 3 \ 4]$

A 0

B 1

☒ C 2

D 2,5

E 3

$d = 4 - 1 - 1 = 2$



3 $A = [1 \ \bar{3} \ 2 \ 4]$ and
 $B = [1 \ 2 \ 3 \ 4]$

A 0

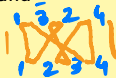
B 1

☒ C 2

D 2,5

E 3

$d = 4 - 1 - 1 = 2$



4 $A = [1 \ \bar{3} \ \bar{2} \ 4]$ and
 $B = [1 \ 2 \ 3 \ 4]$

A 0

☒ B 1

C 1,5

D 2

E 3

$d = 4 - 2 - 1 = 1$



Computing the canonical DCJ Distance in linear time

A. Telomeres and adjacencies of genome A

Pos	1	2	3	4	5	6	7
1st	1^t	1^h	5^h	3^h	4^t	6^h	2^h
2nd	-	5^t	3^t	-	-	2^t	-

Positions of gene extremities in Tab. A

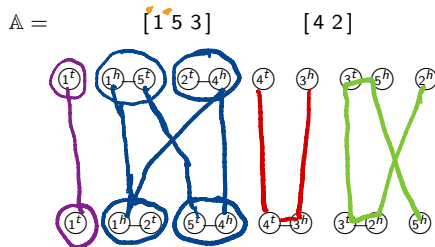
gene	1	2	3	4	5
head	2	7	4	6	3
tail	1	6	3	5	2

B. Telomeres and adjacencies of genome B

Pos	1	2	3	4	5	6
1st	1^t	1^h	2^h	3^h	4^h	5^h
2nd	-	2^t	3^t	4^t	5^t	-

Positions of gene extremities in Tab. B

gene	1	2	3	4	5
head	2	3	4	5	6
tail	1	2	3	4	5



B = [1 2 3 4 5]

$$p_{AB} = 11$$

$$c = 1$$

$$d = 5 - 1 - 1 = 3$$

Canonical DCJ Sorting

Algorithm 2 (Greedy sorting by DCJ)

```
1: for each adjacency  $\{p, q\}$  in genome  $B$  do
2:   let  $u$  be the element of genome  $A$  that contains  $p$ 
3:   let  $v$  be the element of genome  $A$  that contains  $q$ 
4:   if  $u \neq v$  then
5:     replace  $u$  and  $v$  in  $A$  by  $\{p, q\}$  and  $(u \setminus \{p\}) \cup (v \setminus \{q\})$ 
6:   end if
7: end for
8: for each telomere  $\{p\}$  in genome  $B$  do
9:   let  $u$  be the element of genome  $A$  that contains  $p$ 
10:  if  $u$  is an adjacency then
11:    replace  $u$  in  $A$  by  $\{p\}$  and  $(u \setminus \{p\})$ 
12:  end if
13: end for
```

Canonical DCJ Sorting

$$\mathbb{A} = [1|5\ 3] \quad [4|2]$$

Handwritten steps showing the sorting process:

$$[1\ 2|] \quad [4\ 5|3]$$

$$[1\ 2\ 3|] \quad [4\ 5]$$

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

Red arrows indicate the sequence of operations: first a swap of 2 and 5 in the first permutation, then a swap of 3 and 4 in the second, and finally a swap of 2 and 3 in the first, leading to the sorted state.

$$\mathbb{B} = [1\ 2\ 3\ 4\ 5]$$

A. Telomeres and adjacencies of genome \mathbb{A}

Pos	1	2	3	4	5	6	7
1st	1^t	1^h	5^h	3^h	4^t	4^h	2^h
2nd	-	5^t	3^t	-	-	2^t	-

Positions of gene extremities in Tab. A

gene	1	2	3	4	5
head	2	7	4	6	3
tail	1	6	3	5	2

B. Telomeres and adjacencies of genome \mathbb{B}

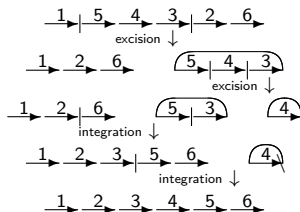
Pos	1	2	3	4	5	6
1st	1^t	1^h	2^h	3^h	4^h	5^h
2nd	-	2^t	3^t	4^t	5^t	-

Positions of gene extremities in Tab. B

gene	1	2	3	4	5
head	2	3	4	5	6
tail	1	2	3	4	5

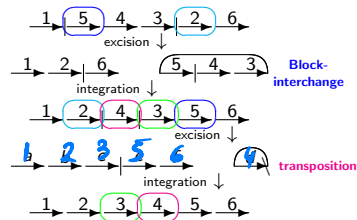
DCJ model - circular excision/integration

Canonical DCJ model



Many circular chromosomes can coexist in the intermediate genomes.

Restricted canonical DCJ model



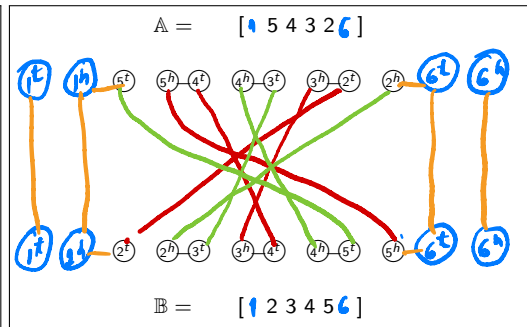
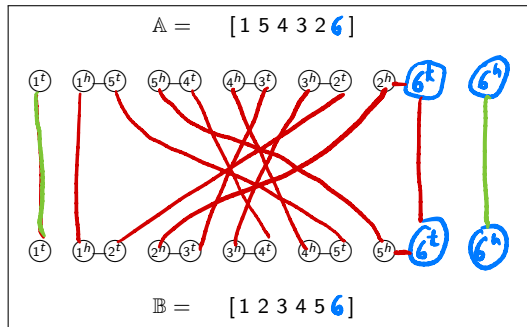
A circular chromosome is immediately reintegrated after its excision.

The DCJ distance is the same for both the general and the restricted DCJ models

Restricted canonical DCJ sorting

Canonical genomes \mathbb{A} and \mathbb{B} are linear.

Transform **A** and **B** into **co-tailed** genomes **A'** and **B'**:



n: 5 \rightarrow 6 (+1)

$$P_{AB} = 2 \rightarrow 2$$

$$c: 0 \rightarrow 1 \quad (-9)$$

Assume that the genes in chromosomes of \mathbb{B}' are consecutive numbers
(otherwise renumber the genes of \mathbb{A}' and of \mathbb{B}')

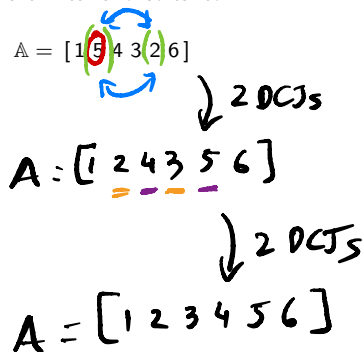
$n: 4 \rightarrow 6 (+2)$

$P_{A_2}: 0 \rightarrow 2 \quad (-1)$

$$c: \emptyset \rightarrow 1 \quad (-1)$$

Restricted canonical DCJ sorting

Canonical genomes \mathbb{A}' and \mathbb{B}' are linear and co-tailed.



$$\mathbb{B} = [1 \text{ } 2 \text{ } 3 \text{ } 4 \text{ } 5 \text{ } 6]$$

Sort \mathbb{A}' into \mathbb{B}' from left to right:

Once we have transformed the beginning of a chromosome in \mathbb{A}' to $k \ k+i \ \dots \ \ell$, we extend it by moving $\ell+1$ next to ℓ . Cases:

1. $\ell+1$ is on a different chromosome: translocation

Otherwise the situation is $\ell \dots \ell+1 \dots$

2. ℓ and $\ell+1$ have distinct orientations: inversion

Otherwise, find the highest gene m between ℓ and $\ell+1$ and find $m+1$

3. $m+1$ is on a different chromosome: translocation to move it next to m ; this operation also moves $\ell+1$ to another chromosome, with another translocation we put it next to ℓ

Otherwise the situation is $\ell \dots m \dots \ell+1 \dots m+1$

4. m and $m+1$ have distinct orientations: inversion to move $m+1$ next to m ; this also changes the orientation of $\ell+1$, with another inversion we put it next to ℓ

5. m and $m+1$ have the same orientation: interchange blocks $\ell \langle \dots m \rangle \dots \langle \ell+1 \dots \rangle m+1 \rightsquigarrow \ell \ell+1 \dots m \ m+1$ (if both m and $m+1$ have positive orientation)
or
 $\ell \langle \dots \overline{m} \dots \overline{\ell+1} \dots \overline{m+1} \rangle \rightsquigarrow \ell \ell+1 \dots \overline{m+1} \ \overline{m}$ (if m and $m+1$ have both reverse orientation)

With two operations we put $\ell+1$ next to ℓ and m next to $m+1$.

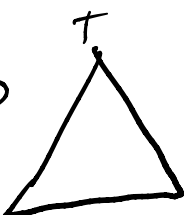
$$\begin{array}{l} [\dots x \dots \overline{x+1} \dots] \\ [\dots x \overline{x+1} \dots] \end{array} \left. \vphantom{\begin{array}{l} [\dots x \dots \overline{x+1} \dots] \\ [\dots x \overline{x+1} \dots] \end{array}} \right\} \text{INVERSION}$$

$$\begin{array}{l} [\dots x \dots] \dots [\dots \overline{x+1} \dots] \\ [\dots x \dots] \dots [\dots \overline{x+1} \dots] \end{array} \left. \vphantom{\begin{array}{l} [\dots x \dots] \dots [\dots \overline{x+1} \dots] \\ [\dots x \dots] \dots [\dots \overline{x+1} \dots] \end{array}} \right\} \begin{array}{l} \text{TRANSLOCATION} \\ 1-2 \text{ INVERSIONS} \end{array}$$

$$\begin{array}{l} [\dots x] \dots [\dots \overline{x+1}] \\ [\dots x \overline{x+1} \dots] \dots [] \\ [\dots x] \dots [\overline{x+1} \dots] \end{array} \left. \vphantom{\begin{array}{l} [\dots x] \dots [\dots \overline{x+1}] \\ [\dots x \overline{x+1} \dots] \dots [] \\ [\dots x] \dots [\overline{x+1} \dots] \end{array}} \right\} \begin{array}{l} \text{FUSION/FISSION} \\ 1-2 \text{ INVERSIONS} \end{array}$$

$$\begin{array}{l} [\dots x \dots m \dots x+1 \dots m+1] \\ [\dots x \dots \overline{x+1} \dots \overline{m} \dots m+1] \end{array} \left. \vphantom{\begin{array}{l} [\dots x \dots m \dots x+1 \dots m+1] \\ [\dots x \dots \overline{x+1} \dots \overline{m} \dots m+1] \end{array}} \right\} \begin{array}{l} \text{B.I.} \\ 3- \text{INVERSIONS} \end{array}$$

BALANCED TREE



SPLIT



INVERSION



MERGE



$O(n \log n)$

Quiz 2

1 Which of the following statements about the DCJ model are true?

☒ A Computing the canonical DCJ distance can be done in linear time.

☒ B Sorting a canonical genome \mathbb{A} into another canonical genome \mathbb{B} has the same complexity as computing the DCJ distance of \mathbb{A} and \mathbb{B} . *GREEDY SORTING*

☐ C DCJ distance and sorting are asymmetric.

☐ D The canonical DCJ distance and the restricted canonical DCJ distance are distinct.

☒ E The canonical DCJ sorting can be done in linear time.

☐ F The restricted canonical DCJ sorting can be done in linear time.

References

A Unifying View of Genome Rearrangements

(Anne Bergeron, Julia Mixtacki and Jens Stoye)

LNCS, volume 4175, pages 163-173 (2006)

Restricted DCJ Model: Rearrangement Problems with Chromosome Reincorporation

(Jakub Kováč, Robert Warren, Marília D. V. Braga, and Jens Stoye)

JCB Volume 18, Number 9, 2011