

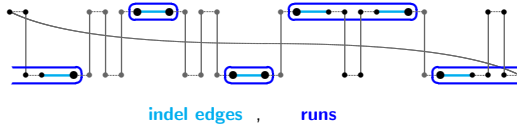
Topics of today:

Singular DCJ-indel distance and sorting:

1. Indel-potential
2. Deducing path recombinations
3. Restricted DCJ-indel model
4. The diameter of the DCJ-indel distance
5. Establishing the triangular inequality

Runs of indel-edges

One indel-enclosing cycle:



$$\Lambda = 4$$

$\Lambda(C)$ is the number of **runs** in component C

Λ	
0	cycles or paths
1	cycles, paths and singletons
2	cycles, paths
3	paths
4	cycles, paths
5	paths
6	cycles, paths
\vdots	\vdots
\vdots	\vdots

Runs of indel-edges

Types of DCJ operation $\left\{ \begin{array}{l} \Delta_{\text{DCJ}} = 0 \text{ (gaining): creates one cycle or two } \mathbb{A}\mathbb{B}\text{-paths} \\ \Delta_{\text{DCJ}} = 1 \text{ (neutral): does not change the number of cycles nor of } \mathbb{A}\mathbb{B}\text{-paths} \\ \Delta_{\text{DCJ}} = 2 \text{ (losing): destroys one cycle or two } \mathbb{A}\mathbb{B}\text{-paths} \end{array} \right.$

Each **run** can be **accumulated** with gaining DCJ operations and then inserted/deleted at once

\Rightarrow Second upper bound:

$$d_{\text{DCJ}}^{\text{ID}}(\mathbb{A}, \mathbb{B}) \leq n - |C| - \frac{|\mathcal{P}_{\mathbb{A}\mathbb{B}}|}{2} + \sum_{C \in \mathcal{R}G} \Lambda(C)$$

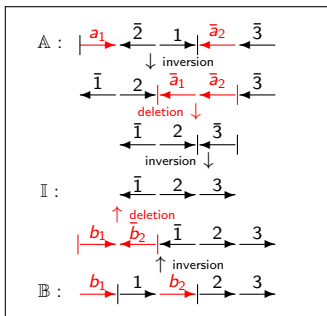
DCJ operations can modify the number of runs:

A DCJ operation can have $\left\{ \begin{array}{l} \Delta_{\Lambda} = -2 \text{ (merges two pairs of runs)} \\ \Delta_{\Lambda} = -1 \text{ (merges one pair of runs)} \\ \Delta_{\Lambda} = 0 \text{ (preserves the runs)} \\ \Delta_{\Lambda} = 1 \text{ (splits one run)} \\ \Delta_{\Lambda} = 2 \text{ (splits two runs)} \end{array} \right.$

Runs can be merged and accumulated in both genomes

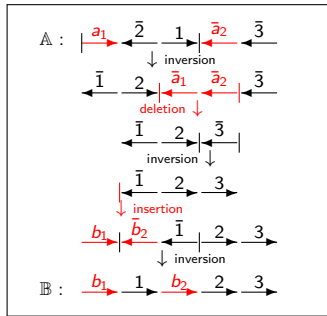
$$\text{Singular genomes } \left\{ \begin{array}{l} \mathbb{A} = [a_1 \quad \bar{2} \quad 1 \quad \bar{a}_2 \quad \bar{3}] \\ \mathbb{B} = [b_1 \quad 1 \quad b_2 \quad 2 \quad 3] \end{array} \right.$$

A sequence of 3 operations
sorting \mathbb{A} into $\mathbb{I} = [\bar{1} \ 2 \ 3]$



A sequence of 2 operations
sorting \mathbb{B} into $\mathbb{I} = [\bar{1} \ 2 \ 3]$

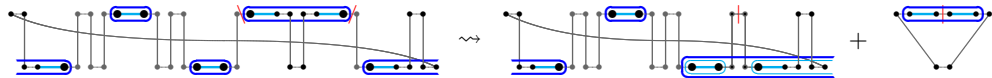
A sequence of 5 operations
sorting \mathbb{A} into \mathbb{B}



\Rightarrow

Merging runs with “internal” gaining DCJ operations

An **gaining DCJ operation** applied to two adjacency-edges belonging to the same indel-enclosing component can **decrease** the number of runs:



$\Lambda = 4$	\rightsquigarrow	2	+	1 = 3 ($\Delta_\Lambda = -1$)
---------------	--------------------	---	---	---------------------------------

DCJ-sorted (or short) components: 2-cycles and 1-paths (and 0-cycles and 0-paths)

Long components: k -cycles (with $k \geq 4$) and k -paths (with $k \geq 2$)

DCJ-sorting a long component C : transforming C into a set of DCJ-sorted components

Indel-potential $\lambda(C)$ of a component C :

minimum number of runs that we can obtain by DCJ-sorting C with gaining DCJ operations

Indel-potential λ' of a cycle C

$$\Lambda(C) = 0, 1, 2, 4, 6, 8, \dots$$

We will show that $\lambda'(C)$ depends only on the value $\Lambda(C)$: denote $\lambda'(C) = \lambda'(\Lambda(C))$

$$\Lambda(C) = 1 \Rightarrow \lambda'(1) = 1$$

$$\Lambda(C) = 2 \Rightarrow \lambda'(2) = 2$$

$$\Lambda(C) \geq 4 : \Lambda(C) = o_1 + o_2 \text{ such that } o_1 \text{ and } o_2 \text{ are odd, and assume } o_1 \geq o_2$$

two resulting cycles: $\begin{cases} \text{one with } o_1 - 1 \text{ runs} \\ \text{one with either 1 run (if } o_2 = 1) \text{ or with } o_2 - 1 \text{ runs (if } o_2 \geq 3) \end{cases}$

$$\Rightarrow \lambda'(4) = \lambda'(2) + \lambda'(1) = 2 + 1 = 3$$

$$\Rightarrow \lambda'(6) = \begin{cases} \lambda'(2) + \lambda'(2) = 2 + 2 = 4 \\ \lambda'(4) + \lambda'(1) = 3 + 1 = 4 \end{cases}$$

$$\Rightarrow \lambda'(8) = \begin{cases} \lambda'(4) + \lambda'(2) = 3 + 2 = 5 \\ \lambda'(6) + \lambda'(1) = 4 + 1 = 5 \end{cases}$$

Λ	λ'
0	0
1	1
2	2
4	3
6	4
8	5
⋮	⋮
⋮	⋮

Induction: $\begin{cases} \text{hypothesis: } \lambda'(\Lambda(C)) = \frac{\Lambda(C)}{2} + 1 \\ \text{base cases: } \lambda'(1) = 1 \text{ and } \lambda'(2) = 2 \end{cases}$

Induction step: in general, for $\Lambda(C) \geq 4$, we can state $\lambda'(\Lambda(C)) = \lambda'(\Lambda(C) - 2) + \lambda'(1)$

$$= \left(\frac{\Lambda(C) - 2}{2} + 1 \right) + 1$$
$$= \frac{\Lambda(C)}{2} + 1$$

Indel-potential λ'' of a path P

$$\Lambda(P) = 0, 1, 2, 3, 4, 5, 6, 7, 8, \dots$$

We will show that $\lambda''(P)$ depends only on the value $\Lambda(P)$: denote $\lambda''(P) = \lambda''(\Lambda(P))$

$$\Lambda(P) = 1 \Rightarrow \lambda''(1) = 1$$

$$\Lambda(P) = 2 \Rightarrow \lambda''(2) = 2$$

$\Lambda(P) \geq 3$: $\Lambda(P) = o_1 + o_2$ such that $o_1 \geq 1$ and o_2 is odd

two resulting components: $\begin{cases} \text{one path with either 1 run (if } o_1 = 1) \text{ or with } o_1 - 1 \text{ runs (if } o_1 \geq 2) \\ \text{one cycle with either 1 run (if } o_2 = 1) \text{ or with } o_2 - 1 \text{ runs (if } o_2 \in \{3, 5, \dots\}) \end{cases}$

but we can get the same indel-potential if we extract **all runs into a cycle**:

$$\lambda''(3) = \begin{cases} \lambda''(1) + \lambda'(1) = 1 + 1 = 2 \\ \lambda'(2) = 2 \end{cases} \quad \lambda''(5) = \begin{cases} \lambda''(3) + \lambda'(1) = 2 + 1 = 3 \\ \lambda''(1) + \lambda'(2) = 1 + 2 = 3 \\ \lambda'(4) = 3 \end{cases}$$

$$\lambda''(4) = \begin{cases} \lambda''(2) + \lambda'(1) = 2 + 1 = 3 \\ \lambda''(1) + \lambda'(2) = 1 + 2 = 3 \\ \lambda'(4) = 3 \end{cases} \quad \lambda''(6) = \begin{cases} \dots \\ \lambda'(6) = 4 \end{cases}$$

Λ	λ''
0	0
1	1
2	2
3	2
4	3
5	3
6	4
7	4
⋮	⋮
⋮	⋮

In general, for $\Lambda(P) \geq 2$, we can state $\lambda''(\Lambda(P)) = \begin{cases} \lambda'(\Lambda(P)) & \text{if } \Lambda(P) \text{ is even} \\ \lambda'(\Lambda(P) - 1) & \text{if } \Lambda(P) \text{ is odd} \end{cases}$

$$\lambda''(\Lambda(P)) = \left\lceil \frac{\Lambda(P) + 1}{2} \right\rceil$$

Indel-potential λ of a component C

If C is a singleton: $\lambda(C) = 1$

If C is a cycle:

$$\lambda(C) = \begin{cases} 0 & \text{if } \Lambda(C) = 0 \text{ (} C \text{ is indel-free)} \\ 1 & \text{if } \Lambda(C) = 1 \\ \frac{\Lambda(C)}{2} + 1 & \text{if } \Lambda(C) \geq 2 \end{cases}$$

If C is a path:

$$\lambda(C) = \begin{cases} 0 & \text{if } \Lambda(C) = 0 \text{ (} C \text{ is indel-free)} \\ \lceil \frac{\Lambda(C)+1}{2} \rceil & \text{if } \Lambda(C) \geq 1 \end{cases}$$

In general, for any component C :

$$\lambda(C) = \begin{cases} 0 & \text{if } \Lambda(C) = 0 \text{ (} C \text{ is indel-free)} \\ \lceil \frac{\Lambda(C)+1}{2} \rceil & \text{if } \Lambda(C) \geq 1 \end{cases}$$

Λ	λ
0	0
1	1
2	2
3	2
4	3
5	3
6	4
7	4
\vdots	\vdots
\vdots	\vdots

paths and cycles
 paths, cycles and singletons
 paths and cycles
 paths
 paths and cycles
 paths
 paths and cycles
 paths

Third upper bound:

$$d_{DCJ}^{\text{ID}}(\mathbb{A}, \mathbb{B}) \leq n - |C| - \frac{|\mathcal{P}_{AB}|}{2} + \sum_{C \in RG} \lambda(C)$$

(gaining DCJ operations + indels sorting components separately)

Types of DCJ operation

DCJ-types of DCJ operation $\left\{ \begin{array}{l} \Delta_{\text{DCJ}} = 0 \text{ (gaining): creates one cycle or two AB-paths} \\ \Delta_{\text{DCJ}} = 1 \text{ (neutral): does not change the number of cycles nor of AB-paths} \\ \Delta_{\text{DCJ}} = 2 \text{ (losing): destroys one cycle or two AB-paths} \end{array} \right.$

Indel-types of DCJ operation $\left\{ \begin{array}{l} \Delta_{\lambda} = -2 : \text{decreases the overall indel-potential by two} \\ \Delta_{\lambda} = -1 : \text{decreases the overall indel-potential by one} \\ \Delta_{\lambda} = 0 : \text{does not change the overall indel-potential} \\ \Delta_{\lambda} = 1 : \text{increases the overall indel-potential by one} \\ \Delta_{\lambda} = 2 : \text{increases the overall indel-potential by two} \end{array} \right.$

Effect of a DCJ operation ρ on the third upper bound: $\Delta_{\text{DCJ}}^{\lambda}(\rho) = \Delta_{\text{DCJ}}(\rho) + \Delta_{\lambda}(\rho)$

DCJ Operations that can decrease the third upper bound: $\left\{ \begin{array}{l} \Delta_{\text{DCJ}} = 0 \text{ (gaining) and } \Delta_{\lambda} = -2 : \Delta_{\text{DCJ}}^{\lambda} = -2 \\ \Delta_{\text{DCJ}} = 0 \text{ (gaining) and } \Delta_{\lambda} = -1 : \Delta_{\text{DCJ}}^{\lambda} = -1 \\ \Delta_{\text{DCJ}} = 1 \text{ (neutral) and } \Delta_{\lambda} = -2 : \Delta_{\text{DCJ}}^{\lambda} = -1 \end{array} \right.$

- ▶ By definition: any “internal” gaining DCJ operation ρ (applied to a single component) has $\Delta_{\lambda}(\rho) \geq 0$ and, consequently, $\Delta_{\text{DCJ}}^{\lambda}(\rho) \geq 0$
- ▶ Any losing DCJ operation ρ has $\Delta_{\text{DCJ}}^{\lambda}(\rho) \geq 0$

DCJ operations involving cycles

Λ	λ
0	0
1	1
2	2
4	3
6	4
8	5
⋮	⋮
⋮	⋮

▶ Any DCJ operation involving two cycles is losing and has $\Delta_{\text{DCJ}}^{\lambda} \geq 0$
(cannot decrease the DCJ-indel distance)

▶ A DCJ operation ρ applied to a single cycle C can be:

▶ Gaining, with $\Delta_{\text{DCJ}}^{\lambda}(\rho) \geq 0$ (cannot decrease the DCJ-indel distance)

▶ Neutral ($\Delta_{\text{DCJ}}(\rho) = 1$):

If $\Lambda(C) \geq 4$, the DCJ ρ can merge at most two pairs of runs: $\Delta_{\Lambda}(\rho) \geq -2$ and $\Delta_{\lambda}(\rho) \geq -1$

\Rightarrow Any neutral DCJ operation applied to a single cycle has $\Delta_{\text{DCJ}}^{\lambda} \geq 0$
(cannot decrease the DCJ-indel distance)

If singular genomes \mathbb{A} and \mathbb{B} are circular, the graph $RG(\mathbb{A}, \mathbb{B})$ has only cycles (and eventually singletons).

In this case:

$$d_{\text{DCJ}}^{\text{ID}}(\mathbb{A}, \mathbb{B}) = n - |\mathcal{C}| + \sum_{C \in \mathcal{C}} \lambda(C)$$

DCJ operations involving paths

- ▶ Any DCJ operation involving a path and a cycle is losing and has $\Delta_{\text{DCJ}}^\lambda \geq 0$ (cannot decrease the DCJ-indel distance)

Λ	λ
0	0
1	1
2	2
3	2
4	3
5	3
6	4
7	4
⋮	⋮
⋮	⋮

- ▶ A DCJ operation ρ applied to a single path P can be:
 - ▶ Gaining, with $\Delta_{\text{DCJ}}^\lambda(\rho) \geq 0$ (cannot decrease the DCJ-indel distance)
 - ▶ Neutral ($\Delta_{\text{DCJ}}(\rho) = 1$):
 - If $\Lambda(P) \geq 4$, the DCJ ρ can merge at most two pairs of runs: $\Delta_\Lambda(\rho) \geq -2$ and $\Delta_\lambda(\rho) \geq -1$
- ⇒ Any neutral DCJ operation applied to a single path has $\Delta_{\text{DCJ}}^\lambda \geq 0$ (cannot decrease the DCJ-indel distance)

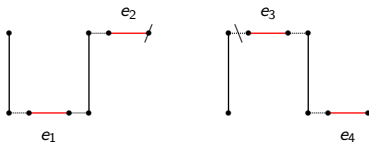
Path recombinations can have $\Delta_{DCJ}^\lambda \leq -1$

An gaining (**deducting**) path recombination with $\Delta_{DCJ}^\lambda = -2$:

Sources

$$(\sum \lambda = 2 + 2 = 4)$$

AA + BB
2 runs + 2 runs

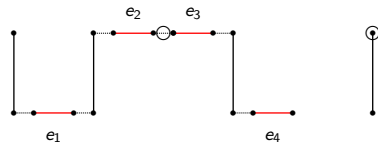


→
gaining
DCJ

Resultants

$$(\sum \lambda = 2 + 0 = 2)$$

AB + AB
3 runs + no run



$$AA_{BA} + BB_{AB} = \begin{cases} AB_{BAB} + AB_\epsilon \\ (AB_{ABA} + AB_\epsilon) \\ (AB_A + AB_B) \end{cases} \quad (\text{all variants have } \Delta_{DCJ}^\lambda = -2)$$

Deducting path recombinations

have $\Delta_{DCJ}^\lambda \leq -1$

General DCJ-indel distance formula:

$$d_{DCJ}^{ID}(A, B) = n - |C| - \frac{|P_{AB}|}{2} + \sum_{C \in RG} \lambda(C) - \delta,$$

where δ is the value obtained by optimizing deducting path recombinations

Optimizing deducing path recombinations (for computing δ)

$$\text{Run-type of a path} \begin{cases} \varepsilon & \equiv \varepsilon \text{ (empty)} \\ \mathcal{A}\mathcal{B}\mathcal{A}\mathcal{B} \dots \mathcal{A} & \equiv \mathcal{A} \text{ (odd)} \\ \mathcal{B}\mathcal{A}\mathcal{B}\mathcal{A} \dots \mathcal{B} & \equiv \mathcal{B} \text{ (odd)} \\ \mathcal{A}\mathcal{B}\mathcal{A}\mathcal{B} \dots \mathcal{A}\mathcal{B} & \equiv \mathcal{A}\mathcal{B} \text{ (even)} \\ \mathcal{B}\mathcal{A}\mathcal{B}\mathcal{A} \dots \mathcal{B}\mathcal{A} & \equiv \mathcal{B}\mathcal{A} \text{ (even)} \end{cases} \quad \text{Path types} \begin{cases} \mathcal{A}\mathcal{A}_\varepsilon, \mathcal{A}\mathcal{A}_\mathcal{A}, \mathcal{A}\mathcal{A}_\mathcal{B}, \mathcal{A}\mathcal{A}_{\mathcal{A}\mathcal{B}} (\equiv \mathcal{A}\mathcal{A}_{\mathcal{B}\mathcal{A}}) \\ \mathcal{B}\mathcal{B}_\varepsilon, \mathcal{B}\mathcal{B}_\mathcal{A}, \mathcal{B}\mathcal{B}_\mathcal{B}, \mathcal{B}\mathcal{B}_{\mathcal{A}\mathcal{B}} (\equiv \mathcal{B}\mathcal{B}_{\mathcal{B}\mathcal{A}}) \\ \mathcal{A}\mathcal{B}_\varepsilon, \mathcal{A}\mathcal{B}_\mathcal{A}, \mathcal{A}\mathcal{B}_\mathcal{B}, \mathcal{A}\mathcal{B}_{\mathcal{A}\mathcal{B}}, \mathcal{A}\mathcal{B}_{\mathcal{B}\mathcal{A}} \\ \Rightarrow \text{an } \mathcal{A}\mathcal{B}\text{-path is always read from } \mathcal{A} \text{ to } \mathcal{B} \end{cases}$$

Deducing path recombinations that allow the best reuse of the resultants:

sources	resultants	Δ_λ	Δ_{DCJ}	$\Delta_{\text{DCJ}}^\lambda$
$\mathcal{A}\mathcal{A}_{\mathcal{A}\mathcal{B}} + \mathcal{B}\mathcal{B}_{\mathcal{A}\mathcal{B}}$	$\bullet + \bullet$	-2	0	-2
$\mathcal{A}\mathcal{A}_{\mathcal{A}\mathcal{B}} + \mathcal{B}\mathcal{B}_\mathcal{A}$	$\bullet + \mathcal{A}\mathcal{B}_{\mathcal{B}\mathcal{A}}$	-1	0	-1
$\mathcal{A}\mathcal{A}_{\mathcal{A}\mathcal{B}} + \mathcal{B}\mathcal{B}_\mathcal{B}$	$\bullet + \mathcal{A}\mathcal{B}_{\mathcal{A}\mathcal{B}}$	-1	0	-1
$\mathcal{A}\mathcal{A}_\mathcal{A} + \mathcal{B}\mathcal{B}_{\mathcal{A}\mathcal{B}}$	$\bullet + \mathcal{A}\mathcal{B}_{\mathcal{A}\mathcal{B}}$	-1	0	-1
$\mathcal{A}\mathcal{A}_\mathcal{B} + \mathcal{B}\mathcal{B}_{\mathcal{A}\mathcal{B}}$	$\bullet + \mathcal{A}\mathcal{B}_{\mathcal{B}\mathcal{A}}$	-1	0	-1
$\mathcal{A}\mathcal{A}_\mathcal{A} + \mathcal{B}\mathcal{B}_\mathcal{A}$	$\bullet + \bullet$	-1	0	-1
$\mathcal{A}\mathcal{A}_\mathcal{B} + \mathcal{B}\mathcal{B}_\mathcal{B}$	$\bullet + \bullet$	-1	0	-1

sources	resultants	Δ_λ	Δ_{DCJ}	$\Delta_{\text{DCJ}}^\lambda$
$\mathcal{A}\mathcal{A}_{\mathcal{A}\mathcal{B}} + \mathcal{A}\mathcal{A}_{\mathcal{A}\mathcal{B}}$	$\mathcal{A}\mathcal{A}_\mathcal{A} + \mathcal{A}\mathcal{A}_\mathcal{B}$	-2	+1	-1
$\mathcal{B}\mathcal{B}_{\mathcal{A}\mathcal{B}} + \mathcal{B}\mathcal{B}_{\mathcal{A}\mathcal{B}}$	$\mathcal{B}\mathcal{B}_\mathcal{A} + \mathcal{B}\mathcal{B}_\mathcal{B}$	-2	+1	-1
$\mathcal{A}\mathcal{A}_{\mathcal{A}\mathcal{B}} + \mathcal{A}\mathcal{B}_{\mathcal{A}\mathcal{B}}$	$\bullet + \mathcal{A}\mathcal{A}_\mathcal{A}$	-2	+1	-1
$\mathcal{A}\mathcal{A}_{\mathcal{A}\mathcal{B}} + \mathcal{A}\mathcal{B}_{\mathcal{B}\mathcal{A}}$	$\bullet + \mathcal{A}\mathcal{A}_\mathcal{B}$	-2	+1	-1
$\mathcal{B}\mathcal{B}_{\mathcal{A}\mathcal{B}} + \mathcal{A}\mathcal{B}_{\mathcal{A}\mathcal{B}}$	$\bullet + \mathcal{B}\mathcal{B}_\mathcal{B}$	-2	+1	-1
$\mathcal{B}\mathcal{B}_{\mathcal{A}\mathcal{B}} + \mathcal{A}\mathcal{B}_{\mathcal{B}\mathcal{A}}$	$\bullet + \mathcal{B}\mathcal{B}_\mathcal{A}$	-2	+1	-1
$\mathcal{A}\mathcal{B}_{\mathcal{A}\mathcal{B}} + \mathcal{A}\mathcal{B}_{\mathcal{B}\mathcal{A}}$	$\bullet + \bullet$	-2	+1	-1

Sources:

$\mathcal{W} : \mathcal{A}\mathcal{A}_{\mathcal{A}\mathcal{B}}$

$\bar{\mathcal{W}} : \mathcal{A}\mathcal{A}_\mathcal{A}$

$\underline{\mathcal{W}} : \mathcal{A}\mathcal{A}_\mathcal{B}$

$\mathcal{M} : \mathcal{B}\mathcal{B}_{\mathcal{A}\mathcal{B}}$

$\bar{\mathcal{M}} : \mathcal{B}\mathcal{B}_\mathcal{A}$

$\underline{\mathcal{M}} : \mathcal{B}\mathcal{B}_\mathcal{B}$

$\mathcal{Z} : \mathcal{A}\mathcal{B}_{\mathcal{A}\mathcal{B}}$

$\mathcal{N} : \mathcal{A}\mathcal{B}_{\mathcal{B}\mathcal{A}}$

Path recombinations with $\Delta_{\text{DCJ}}^\lambda = 0$ creating resultants that can be used in deducing recombinations:

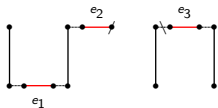
sources	resultants	Δ_λ	Δ_{DCJ}	$\Delta_{\text{DCJ}}^\lambda$
$\mathcal{A}\mathcal{A}_\mathcal{A} + \mathcal{A}\mathcal{B}_{\mathcal{B}\mathcal{A}}$	$\bullet + \mathcal{A}\mathcal{A}_{\mathcal{A}\mathcal{B}}$	-1	+1	0
$\mathcal{A}\mathcal{A}_\mathcal{B} + \mathcal{A}\mathcal{B}_{\mathcal{A}\mathcal{B}}$	$\bullet + \mathcal{A}\mathcal{A}_{\mathcal{A}\mathcal{B}}$	-1	+1	0
$\mathcal{B}\mathcal{B}_\mathcal{A} + \mathcal{A}\mathcal{B}_{\mathcal{A}\mathcal{B}}$	$\bullet + \mathcal{B}\mathcal{B}_{\mathcal{A}\mathcal{B}}$	-1	+1	0
$\mathcal{B}\mathcal{B}_\mathcal{B} + \mathcal{A}\mathcal{B}_{\mathcal{B}\mathcal{A}}$	$\bullet + \mathcal{B}\mathcal{B}_{\mathcal{A}\mathcal{B}}$	-1	+1	0

sources	resultants	Δ_λ	Δ_{DCJ}	$\Delta_{\text{DCJ}}^\lambda$
$\mathcal{A}\mathcal{A}_\mathcal{A} + \mathcal{B}\mathcal{B}_\mathcal{B}$	$\bullet + \mathcal{A}\mathcal{B}_{\mathcal{A}\mathcal{B}}$	0	0	0
$\mathcal{A}\mathcal{A}_\mathcal{B} + \mathcal{B}\mathcal{B}_\mathcal{A}$	$\bullet + \mathcal{A}\mathcal{B}_{\mathcal{B}\mathcal{A}}$	0	0	0
$\mathcal{A}\mathcal{B}_{\mathcal{A}\mathcal{B}} + \mathcal{A}\mathcal{B}_{\mathcal{A}\mathcal{B}}$	$\mathcal{A}\mathcal{A}_\mathcal{A} + \mathcal{B}\mathcal{B}_\mathcal{B}$	-2	+2	0
$\mathcal{A}\mathcal{B}_{\mathcal{B}\mathcal{A}} + \mathcal{A}\mathcal{B}_{\mathcal{B}\mathcal{A}}$	$\mathcal{A}\mathcal{A}_\mathcal{B} + \mathcal{B}\mathcal{B}_\mathcal{A}$	-2	+2	0

Optimizing deducing path recombinations (for computing δ)

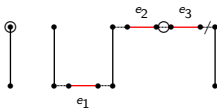
Deducing chain of path recombinations $\left\{ \begin{array}{l} \text{transforming } 2 \times AA_{AB} + BB_A + BB_B \\ \text{into } 3 \times AB_\varepsilon + AB_B \\ \text{with overall } \Delta_{DCJ}^\lambda = -3 \end{array} \right.$

$AA_{AB} + BB_A$
2 runs + 1 run
 $\lambda = 2 + \lambda = 1$



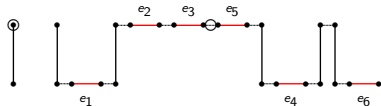
$(\Delta_{DCJ}^\lambda = -1)$
gaining DCJ

$AB_\varepsilon + AB_{BA}$
no run + 2 runs
 $\lambda = 0 + \lambda = 2$



\searrow

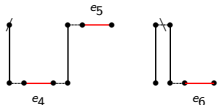
$AB_\varepsilon + AB_B$
no run + 3 runs
 $\lambda = 0 + \lambda = 2$



$(\Delta_{DCJ}^\lambda = -1)$
neutral DCJ

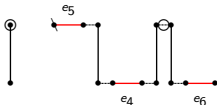
\nearrow

$AA_{AB} + BB_B$
2 runs + 1 run
 $\lambda = 2 + \lambda = 1$



$(\Delta_{DCJ}^\lambda = -1)$
gaining DCJ

$AB_\varepsilon + AB_{AB}$
no run + 2 runs
 $\lambda = 0 + \lambda = 2$



id		sources			resultants			Δ_{DCJ}^λ	scr	
\mathcal{P}	WM	AA_{AB}	BB_{AB}	—	—	—	$2 \times \bullet$	-2	-1	
\mathcal{Q}	WWM	$2 \times AA_{AB}$	$BB_A + BB_B$	—	—	—	$4 \times \bullet$	-3	-3/4	
	MMW	$AA_A + AA_B$	$2 \times BB_{AB}$	—	—	—	$4 \times \bullet$	-3	-3/4	
\mathcal{T}	WZM	AA_{AB}	BB_A	AB_{AB}	—	—	$3 \times \bullet$	-2	-2/3	
	WWM	$2 \times AA_{AB}$	BB_A	—	AA_B	—	$2 \times \bullet$	-2	-2/3	
	WNM	AA_{AB}	BB_B	AB_{BA}	—	—	$3 \times \bullet$	-2	-2/3	
	WWM	$2 \times AA_{AB}$	BB_B	—	AA_A	—	$2 \times \bullet$	-2	-2/3	
	MNW	AA_A	BB_{AB}	AB_{BA}	—	—	$3 \times \bullet$	-2	-2/3	
	MMW	AA_A	$2 \times BB_{AB}$	—	—	BB_B	$2 \times \bullet$	-2	-2/3	
	MZW	AA_B	BB_{AB}	AB_{AB}	—	—	$3 \times \bullet$	-2	-2/3	
	MMW	AA_B	$2 \times BB_{AB}$	—	—	BB_A	$2 \times \bullet$	-2	-2/3	
\mathcal{S}	ZN	—	—	$AB_{AB} + AB_{BA}$	—	—	$2 \times \bullet$	-1	-1/2	
	WM	AA_A	BB_A	—	—	—	$2 \times \bullet$	-1	-1/2	
	WM	AA_B	BB_B	—	—	—	$2 \times \bullet$	-1	-1/2	
	WM	AA_{AB}	BB_A	—	—	AB_{BA}	\bullet	-1	-1/2	
	WM	AA_{AB}	BB_B	—	—	AB_{AB}	\bullet	-1	-1/2	
	WZ	AA_{AB}	—	AB_{AB}	AA_A	—	\bullet	-1	-1/2	
	WN	AA_{AB}	—	AB_{BA}	AA_B	—	\bullet	-1	-1/2	
	WW	$2 \times AA_{AB}$	—	—	$AA_A + AA_B$	—	—	-1	-1/2	
	MW	AA_A	BB_{AB}	—	—	—	AB_{AB}	\bullet	-1	-1/2
	MW	AA_B	BB_{AB}	—	—	—	AB_{BA}	\bullet	-1	-1/2
	MZ	—	BB_{AB}	AB_{AB}	—	BB_B	—	\bullet	-1	-1/2
	MN	—	BB_{AB}	AB_{BA}	—	BB_A	—	\bullet	-1	-1/2
	MM	—	$2 \times BB_{AB}$	—	—	$BB_A + BB_B$	—	-1	-1/2	

	id	sources			resultants				Δ_{DCJ}^λ	scr
\mathcal{M}	$ZZ\bar{W}\bar{M}$	AA_B	BB_A	$2 \times AB_{AB}$	—	—	—	$4 \times \bullet$	-2	-1/2
	$NN\bar{W}\bar{M}$	AA_A	BB_B	$2 \times AB_{BA}$	—	—	—	$4 \times \bullet$	-2	-1/2
\mathcal{N}	$Z\bar{W}\bar{M}$	AA_B	BB_A	AB_{AB}	—	—	AB_{BA}	$2 \times \bullet$	-1	-1/3
	$ZZ\bar{W}$	AA_B	—	$2 \times AB_{AB}$	AA_A	—	—	$2 \times \bullet$	-1	-1/3
	$ZZ\bar{M}$	—	BB_A	$2 \times AB_{AB}$	—	BB_B	—	$2 \times \bullet$	-1	-1/3
	$N\bar{W}\bar{M}$	AA_A	BB_B	AB_{BA}	—	—	AB_{AB}	$2 \times \bullet$	-1	-1/3
	$NN\bar{W}$	AA_A	—	$2 \times AB_{BA}$	AA_B	—	—	$2 \times \bullet$	-1	-1/3
	$NN\bar{M}$	—	BB_B	$2 \times AB_{BA}$	—	BB_A	—	$2 \times \bullet$	-1	-1/3

Sources:

W : AA_{AB}

\bar{W} : AA_A

\underline{W} : AA_B

M : BB_{AB}

\bar{M} : BB_A

\underline{M} : BB_B

Z : AB_{AB}

N : AB_{BA}

DCJ-indel distance formula:

$$d_{DCJ}^{ID}(A, B) = n - |C| - \frac{|P_{AB}|}{2} + \sum_{C \in RG} \lambda(C) - \delta,$$

where δ is the value obtained by optimizing deducting path recombinations:

$$\delta = 2P + 3Q + 2T + S + 2M + N$$

the values P , Q , T , S , M and N refer to the corresponding number of chains of deducting path recombinations of each type and can be obtained by a greedy approach (simple top-down screening of the table)

Singular DCJ-indel model - summary

DCJ-indel distance: $d_{\text{DCJ}}^{\text{ID}}(\mathbb{A}, \mathbb{B}) = n - |\mathcal{C}| - \frac{|\mathcal{P}_{\mathbb{A}\mathbb{B}}|}{2} + \sum_{C \in \text{RG}} \lambda(C) - \delta$, where δ is the value obtained by optimizing deducting path recombinations

\mathbb{A} and \mathbb{B} are circular: $d_{\text{DCJ}}^{\text{ID}}(\mathbb{A}, \mathbb{B}) = n - |\mathcal{C}| + \sum_{C \in \text{RG}} \lambda(C)$

Sorting genome \mathbb{A} into genome \mathbb{B} (with a minimum number of DCJs):

1. Apply all \mathcal{P} , \mathcal{Q} , \mathcal{T} , \mathcal{S} , \mathcal{M} and \mathcal{N} chains of deducting path recombinations, in this order.
2. For each component $C \in \text{RG}(\mathbb{A}, \mathbb{B})$:
 - 2.1 Split C with **gaining** DCJs (that have $\Delta_{\lambda} = \mathbf{0}$) until only components with at most two runs are obtained and the total number of runs in all new components is equal to $\lambda(C)$.
 - 2.2 Accumulate all runs in the smaller components derived from C with **gaining** DCJ operations (that have $\Delta_{\lambda} = \mathbf{0}$).
 - 2.3 Apply **gaining** DCJ operations (that have $\Delta_{\lambda} = \mathbf{0}$) in the smaller components derived from C until only DCJ-sorted components exist.
 - 2.4 **Delete** all runs in the DCJ-sorted components derived from C .

Computing the distance and sorting can be done in **linear time**.

Singular DCJ-indel sorting: trade-off between DCJ and indels

The presented sorting algorithm maximizes gaining DCJs with $\Delta_\lambda = 0$ (minimizing indels).

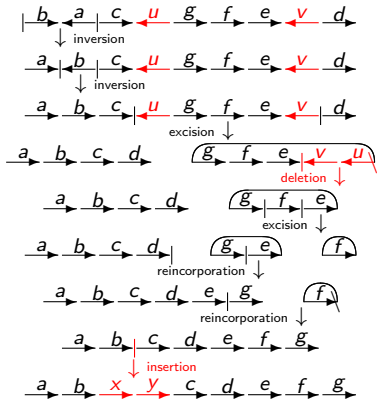
However, these gaining DCJs can often be replaced by $\begin{cases} \text{neutral DCJs with } \Delta_\lambda = -1 \\ \text{losing DCJs with } \Delta_\lambda = -2 \end{cases}$



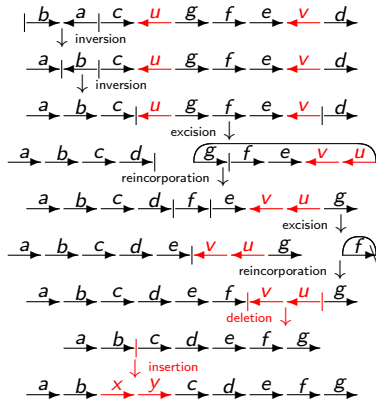
There is a big range of possibilities between the presented sorting algorithm and a sorting algorithm that minimizes gaining DCJs with $\Delta_\lambda = 0$ (maximizing indels)

Restricted DCJ-indel-distance (singular linear genomes)

general DCJ-indel sorting



restricted DCJ-indel sorting



In any sorting sequence, it is always possible to

- move **deletions** down
- move **insertions** up

S : general sequence of DCJ and indel operations sorting linear \mathbb{A} into linear \mathbb{B}

$$S \rightsquigarrow S' = S_{\text{INS}} \oplus S_{\text{DCJ}} \oplus S_{\text{DEL}} \rightsquigarrow R = S_{\text{INS}} \oplus R_{\text{DCJ}} \oplus S_{\text{DEL}} \quad \text{and} \quad |S| = |S'| = |R|$$

The diameter D_{DCJ}^{ID} of the DCJ-indel-distance

For a given component C in a relational graph, let a **segment** of C be

- $\left\{ \begin{array}{l} C \text{ itself (if } C \text{ is a 0-cycle or a 0-path)} \\ \text{a minimal path flanked by two extremity-edges} \\ \text{a minimal path at the extremity of a path and connected to an extremity edge} \end{array} \right.$

$s(C)$: number of segments in component C

$s(C)$	$d_{DCJ}(C)$	$\lambda_{MAX}(C)$	$\lambda_{MAX}(C)$
1	0	1	1
2	0	2	2
3	1	3	2
4	1	4	3
5	2	5	3
6	2	6	4
7	3	7	4
\vdots	\vdots	\vdots	\vdots
$s(C)$	$\lfloor \frac{s(C)-1}{2} \rfloor$	$s(C)$	$\lceil \frac{s(C)+1}{2} \rceil$

if $s(C)$ is odd:

$$d_{DCJ}(C) + \lambda_{MAX}(C) = \frac{s(C)-1}{2} + \frac{s(C)+1}{2} = s(C)$$

if $s(C)$ is even:

$$d_{DCJ}(C) + \lambda_{MAX}(C) = \frac{s(C)-2}{2} + \frac{s(C)+2}{2} = s(C)$$

$$\text{Let } \left\{ \begin{array}{l} \kappa(A) : \# \text{ linear chromosomes in } A \\ S(A) : \# \text{ (circular) singletons in } A \\ \kappa(B) : \# \text{ linear chromosomes in } B \\ S(B) : \# \text{ (circular) singletons in } B \end{array} \right.$$

The number of segments in $RG(A, B)$ is

$$s(RG(A, B)) = 2n + \kappa(A) + S(A) + \kappa(B) + S(B)$$

$$\begin{aligned} D_{DCJ}^{ID}(A, B) &= \sum_{C \in RG(A, B)} (d_{DCJ}(C) + \lambda_{MAX}(C)) \\ &= \sum_{C \in RG(A, B)} s(C) \\ &= s(RG(A, B)) \end{aligned}$$

$$D_{DCJ}^{ID}(A, B) = 2n + \kappa(A) + S(A) + \kappa(B) + S(B)$$

The triangular inequality does not hold for the DCJ-indel distance

$$\text{Three singular genomes } \begin{cases} \mathbb{A} = [1 \ 2 \ 3 \ 4 \ 5] \\ \mathbb{B} = [1 \ 3 \ \bar{4} \ 2 \ 5] \\ \mathbb{C} = [1 \ 5] \end{cases} .$$

$$\begin{array}{l} \text{The triangular inequality} \\ d_{\text{DCJ}}^{\text{ID}}(\mathbb{A}, \mathbb{B}) \leq d_{\text{DCJ}}^{\text{ID}}(\mathbb{A}, \mathbb{C}) + d_{\text{DCJ}}^{\text{ID}}(\mathbb{B}, \mathbb{C}) \\ \text{does not hold} \end{array} \begin{cases} d_{\text{DCJ}}^{\text{ID}}(\mathbb{A}, \mathbb{B}) = 3 \\ d_{\text{DCJ}}^{\text{ID}}(\mathbb{A}, \mathbb{C}) = 1 \\ d_{\text{DCJ}}^{\text{ID}}(\mathbb{B}, \mathbb{C}) = 1 \end{cases}$$

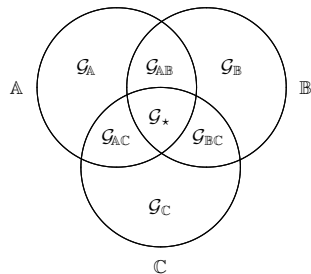
“Free lunch”:
while sorting \mathbb{A} into \mathbb{C} and then \mathbb{C} into \mathbb{B} ,
a set of common genes of \mathbb{A} and \mathbb{B}
are deleted and then reinserted

In the comparison of two genomes, our model prevents this problem:
common genes cannot be deleted or inserted

However, the triangular inequality is essential in other problems involving the DCJ-indel distance
for the comparison of three or more genomes (e.g. median)

Establishing the triangular inequality

Disjoint sets of genes \mathcal{G}_A , \mathcal{G}_B , \mathcal{G}_C , \mathcal{G}_{AB} , \mathcal{G}_{BC} , \mathcal{G}_{AC} and \mathcal{G}_*
for three genomes A, B and C



For each pair of genomes, we define the **corrected distance** dk_{DCJ}^{ID} :

$$dk_{DCJ}^{ID}(A, B) = d_{DCJ}^{ID}(A, B) + k(|\mathcal{G}_A| + |\mathcal{G}_{AC}| + |\mathcal{G}_B| + |\mathcal{G}_{BC}|)$$

$$dk_{DCJ}^{ID}(A, C) = d_{DCJ}^{ID}(A, C) + k(|\mathcal{G}_A| + |\mathcal{G}_{AB}| + |\mathcal{G}_C| + |\mathcal{G}_{BC}|)$$

$$dk_{DCJ}^{ID}(B, C) = d_{DCJ}^{ID}(B, C) + k(|\mathcal{G}_B| + |\mathcal{G}_{AB}| + |\mathcal{G}_C| + |\mathcal{G}_{AC}|)$$

The triangular inequality must hold for dk_{DCJ}^{ID} :

$$dk_{DCJ}^{ID}(A, B) \leq dk_{DCJ}^{ID}(A, C) + dk_{DCJ}^{ID}(B, C)$$

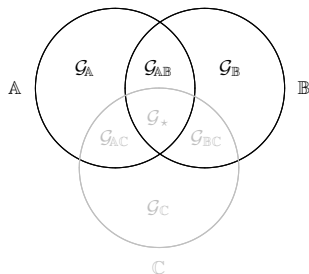
$$d_{DCJ}^{ID}(A, B) + k(|\mathcal{G}_A| + |\mathcal{G}_{AC}| + |\mathcal{G}_B| + |\mathcal{G}_{BC}|) \leq d_{DCJ}^{ID}(A, C) + k(|\mathcal{G}_A| + |\mathcal{G}_{AB}| + |\mathcal{G}_C| + |\mathcal{G}_{BC}|) + d_{DCJ}^{ID}(B, C) + k(|\mathcal{G}_B| + |\mathcal{G}_{AB}| + |\mathcal{G}_C| + |\mathcal{G}_{AC}|)$$

$$d_{DCJ}^{ID}(A, B) \leq d_{DCJ}^{ID}(A, C) + k(|\mathcal{G}_{AB}| + |\mathcal{G}_C|) + d_{DCJ}^{ID}(B, C) + k(|\mathcal{G}_{AB}| + |\mathcal{G}_C|)$$

$$d_{DCJ}^{ID}(A, B) \leq d_{DCJ}^{ID}(A, C) + d_{DCJ}^{ID}(B, C) + 2k(|\mathcal{G}_{AB}| + |\mathcal{G}_C|)$$

Establishing the triangular inequality

$$\begin{cases} d_{DCJ}^{ID}(A, B) \leq d_{DCJ}^{ID}(A, C) + d_{DCJ}^{ID}(B, C) + 2k(|\mathcal{G}_{AB}| + |\mathcal{G}_C|) \\ d_{DCJ}^{ID}(A, C) \leq d_{DCJ}^{ID}(A, B) + d_{DCJ}^{ID}(B, C) + 2k(|\mathcal{G}_{AC}| + |\mathcal{G}_B|) \\ d_{DCJ}^{ID}(B, C) \leq d_{DCJ}^{ID}(A, B) + d_{DCJ}^{ID}(A, C) + 2k(|\mathcal{G}_{BC}| + |\mathcal{G}_A|) \end{cases}$$



Assume $\begin{cases} d_{DCJ}^{ID}(A, B) \geq d_{DCJ}^{ID}(A, C) \\ d_{DCJ}^{ID}(A, B) \geq d_{DCJ}^{ID}(B, C) \end{cases}$ Let $\begin{cases} \xi(A) : \# \text{ chromosomes in } A \\ \kappa(A) : \# \text{ linear chromosomes in } A \\ \mathcal{S}(A) : \# \text{ (circular) singletons in } A \\ \xi(B) : \# \text{ chromosomes in } B \\ \kappa(B) : \# \text{ linear chromosomes in } B \\ \mathcal{S}(B) : \# \text{ (circular) singletons in } B \end{cases}$ $\begin{cases} \kappa(A) + \mathcal{S}(A) \leq \xi(A) \\ \text{and} \\ \kappa(B) + \mathcal{S}(B) \leq \xi(B) \end{cases}$

We need to find a value k that guarantees:

$$d_{DCJ}^{ID}(A, B) \leq d_{DCJ}^{ID}(A, C) + d_{DCJ}^{ID}(B, C) + 2k(|\mathcal{G}_{AB}| + |\mathcal{G}_C|)$$

In the worst case genome C is empty:

$$d_{DCJ}^{ID}(A, C) = \xi(A) \quad \text{and} \quad d_{DCJ}^{ID}(B, C) = \xi(B)$$

$$d_{DCJ}^{ID}(A, B) = 2|\mathcal{G}_{AB}| + \kappa(A) + \mathcal{S}(A) + \kappa(B) + \mathcal{S}(B)$$

$$D_{DCJ}^{ID}(A, B) \leq \xi(A) + \xi(B) + 2k|\mathcal{G}_{AB}|$$

\vdots

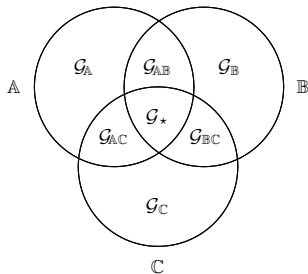
$$2|\mathcal{G}_{AB}| \leq 2k|\mathcal{G}_{AB}| \Rightarrow \boxed{k \geq 1}$$

Establishing the triangular inequality

$$dk_{DCJ}^{ID}(A, B) = d_{DCJ}^{ID}(A, B) + k(|\mathcal{G}_A| + |\mathcal{G}_{AC}| + |\mathcal{G}_B| + |\mathcal{G}_{BC}|)$$

$$dk_{DCJ}^{ID}(A, C) = d_{DCJ}^{ID}(A, C) + k(|\mathcal{G}_A| + |\mathcal{G}_{AB}| + |\mathcal{G}_C| + |\mathcal{G}_{BC}|)$$

$$dk_{DCJ}^{ID}(B, C) = d_{DCJ}^{ID}(B, C) + k(|\mathcal{G}_B| + |\mathcal{G}_{AB}| + |\mathcal{G}_C| + |\mathcal{G}_{AC}|)$$



The triangular inequality holds for the corrected distance dk_{DCJ}^{ID} for any $k \geq 1$

References

Double Cut and Join with Insertions and Deletions

(Marília D.V. Braga, Eyla Willing and Jens Stoye)

JCB, Vol. 18, No. 9 (2011)

Sorting Linear Genomes with Rearrangements and Indels

(Marília D. V. Braga and Jens Stoye)

TCBB, vol 12, issue 3, pp. 500-506 (2015)

On the weight of indels in genomic distances

(Marília D. V. Braga, Raphael Machado, Leonardo C. Ribeiro and Jens Stoye)

BMC Bioinformatics, vol. 12, S13 (2011)