da Silva *et al. Algorithms for Molecular Biology* 2013, **8**:21
http://www.almob.org/content/8/1/21

AMB ALGORITHMS FOR MOLECULAR BIOLOGY

RESEARCH Open Access

DCJ-indel and DCJ-substitution distances with distinct operation costs

Poly H da Silva[1,2], Raphael Machado[2], Simone Dantas[1*] and Marília DV Braga[2]

Abstract

Background: Classical approaches to compute the genomic distance are usually limited to genomes with the same content and take into consideration only rearrangements that change the organization of the genome (i.e. positions and orientation of pieces of DNA, number and type of chromosomes, etc.), such as inversions, translocations, fusions and fissions. These operations are generically represented by the double-cut and join (DCJ) operation. The distance between two genomes, in terms of number of DCJ operations, can be computed in linear time. In order to handle genomes with distinct contents, also insertions and deletions of fragments of DNA – named *indels* – must be allowed. More powerful than an indel is a *substitution* of a fragment of DNA by another fragment of DNA. Indels and substitutions are called *content-modifying* operations. It has been shown that both the DCJ-indel and the DCJ-substitution distances can also be computed in linear time, assuming that the same cost is assigned to any DCJ or content-modifying operation.

Results: In the present study we extend the DCJ-indel and the DCJ-substitution models, considering that the content-modifying cost is distinct from and upper bounded by the DCJ cost, and show that the distance in both models can still be computed in linear time. Although the triangular inequality can be disrupted in both models, we also show how to efficiently fix this problem *a posteriori*.

Keywords: Double cut and join (DCJ), Insertions and deletions (indels), Substitution, Genome rearrangements, Genomic distance, Evolution, Comparative genomics, Combinatorics, Algorithms

Background

The distance between two genomes is often computed using only the common markers, that occur in both genomes. Such distance allows rearrangements that change the organization of the genome, that is, the positions and orientations of markers, number and types of chromosomes. Inversions, translocations, fusions and fissions are some of these operations [1]. All these rearrangements can be generically represented as a *double-cut-and-join* (DCJ) operation [2]. The DCJ distance, which takes into consideration only DCJ operations, can be computed in linear time [3].

Nevertheless, genomes with the same content are rare, and differences in gene content may reflect important evolutionary aspects. In order to handle genomes with unequal contents, one has to take into consideration *content-modifying* operations, that change the contents of the genomes. These operations can be an *insertion* or a *deletion* of a piece of DNA. Insertions and deletions are also called *indels*. Some extensions of the classical approaches lead to models that handle genomes with unequal contents, but without duplicated markers, allowing rearrangements and indels. In 2001, El Mabrouk [4] extended the classical sorting by inversions approach [5] and developed a method to compare unichromosomal genomes with unequal contents, considering only inversions and indels. She provided an exact algorithm that deals with insertions and deletions asymmetrically, and a heuristic that handles the operations symmetrically. Then, in 2009, a model to sort multichromosomal genomes with unequal contents, using both DCJ and indel operations was introduced by Yancopoulos and Friedberg [6]. Later, Braga *et al.* [7] presented an exact formula for the DCJ-indel distance, that can be computed in linear time handling indels symmetrically.

*Correspondence: sdantas@im.uff.br
[1] IME, Universidade Federal Fluminense, Niterói, Brazil
Full list of author information is available at the end of the article

Recently, in 2011, a more powerful content-modifying operation has also been considered: a *substitution* allows a piece of DNA to be substituted by another piece of DNA [8]. Observe that it is not suggested that a substitution occurs in a precise moment in evolution, but instead it represents a region that underwent continuous mutations (duplications, losses and gene mutations), so that a group of genes is transformed into a different group of genes (either of which may also be empty, allowing a substitution to represent an insertion or a deletion). Other studies also represent continuous mutations as a rearrangement event [9,10]. By minimizing substitutions we are able to establish a relation between indels that could have occurred in the same position of the compared genomes, identifying genomic regions that could be subject to these continuous mutations. It has been shown that the DCJ-substitution distance can also be computed in linear time [8].

The approaches mentioned above [4,6-8] assign the same cost to any rearrangement or content-modifying operation. However, during the evolution of many organisms, content-modifying operations are said to occur more often than rearrangements and, consequently, should be assigned to a lower cost. Examples are bacteria that are obligate intracellular parasites, such as *Rickettsia* [11]. The genomes of such intracellular parasites are observed to have a reductive evolution, that is, the process by which genomes shrink and undergo extreme levels of gene degradation and loss. In the present work, we refine the DCJ-indel [7] and the DCJ-substitution [8] models, by adopting a distinct content-modifying cost that is upper bounded by the DCJ cost. For simplicity, we assign a cost of 1 to DCJ and a positive cost of $w \leq 1$ to content-modifying operations. We are then able to give exact formulas for both the DCJ-indel and the DCJ-substitution distances, for any positive $w \leq 1$.

Content-modifying operations are applied to pieces of DNA of any size, and a side effect of this fact is that the triangular inequality often does not hold for distances that consider these operations [4,6-8,12]. In the case of the models we study here, it is possible to do an *a posteriori* correction, using an approach similar to the one described in [12].

This paper is an extension of [13] and is organized as follows. In the remainder of this section we give definitions and previous results used in this work. We will then present our results, including the formulas for the distances with distinct DCJ and content-modifying costs and the correction to establish the triangular inequality.

Genomes

We deal with models in which duplicated markers are not allowed. Given two genomes A and B, possibly with unequal content, let $\mathcal{G}$, $\mathcal{A}$ and $\mathcal{B}$ be three disjoint sets, such that $\mathcal{G}$ is the set of markers that occur both in A and B, $\mathcal{A}$ is the set of markers that occur only in A, and $\mathcal{B}$ is the set of markers that occur only in B. The markers in sets $\mathcal{A}$ and $\mathcal{B}$ are also called *unique markers*. We denote by $u(A,B) = |\mathcal{A}| + |\mathcal{B}|$ the number of unique markers in genomes A and B.

Each marker g in a genome is a DNA fragment and is represented by the symbol g, if it is read in direct orientation, or by the symbol $\overline{g}$, if it is read in reverse orientation. Each one of the two extremities of a linear chromosome is called a *telomere*, represented by the symbol $\circ$. Each chromosome in a genome can be then represented by a string that can be circular, if the chromosome is circular, or linear and flanked by the symbols $\circ$ if the chromosome is linear. In general, a genome is either circular (composed of circular chromosomes) or linear (composed of linear chromosomes). As an example, consider the linear genomes $A = \left\{\circ bsu\bar{c}av\bar{d}e\circ\right\}$ and $B = \left\{\circ awb\bar{x}c\circ, \circ ydze\circ\right\}$, represented in Figure 1. Here we have $\mathcal{G} = \{a,b,c,d,e\}$, $\mathcal{A} = \{s,u,v\}$ and $\mathcal{B} = \{w,x,y,z\}$.

The DCJ model

In this section we will summarize the DCJ model, that allows the sorting of the common content of two genomes, also called *DCJ-sorting*. We will also show how the DCJ distance can be easily computed with the help of the *adjacency graph*.

Given two genomes A and B, we denote the two extremities of each $g \in \mathcal{G}$ by g^t (tail) and g^h (head). Then, a *$\mathcal{G}$-adjacency* or simply *adjacency* [7] in genome A (respectively in genome B) is a string $v = \gamma_1 \ell \gamma_2 \equiv \gamma_2 \bar{\ell} \gamma_1$, such that each γ_i can be a telomere or an extremity of a marker from $\mathcal{G}$ and ℓ is a substring composed of the markers that are between γ_1 and γ_2 in A (respectively in B) and contains no marker that also belongs to $\mathcal{G}$. The substring ℓ is the *label* of v. If ℓ is empty, the adjacency is said to be *clean*, otherwise it is said to be *labeled*. If a linear chromosome is composed only of unique markers, it is represented by an adjacency $\circ\ell\circ$. Similarly, a circular chromosome composed only of unique markers is represented by a (circular) adjacency ℓ. For the linear genomes represented in Figure 1, the set of adjacencies in A is

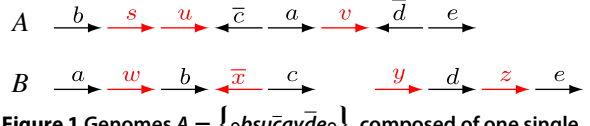

Figure 1 Genomes $A = \left\{\circ bsu\bar{c}av\bar{d}e\circ\right\}$, composed of one single linear chromosome, and $B = \{\circ awb\bar{x}c\circ, \circ ydze\circ\}$, composed of two linear chromosomes. The markers in $\mathcal{G}$ are represented in black, while the unique markers in $\mathcal{A}$ and in $\mathcal{B}$ are represented in red.

$\{\circ b^t,\ b^h s u c^h,\ c^t a^t,\ a^h v d^h,\ d^t e^t,\ e^h \circ\}$ and the set of adjacencies in B is $\{\circ a^t,\ a^h w b^t,\ b^h \overline{x} c^t,\ c^h \circ,\ \circ y d^t,\ d^h z e^t,\ e^h \circ\}$.

Adjacency graph

Given two genomes A and B, the *adjacency graph* $AG(A,B)$ [3] is the bipartite multigraph whose vertices are the adjacencies of A and of B and that has one edge for each common extremity of a pair of vertices. Each of the connected components of $AG(A,B)$ alternate vertices in genome A and in genome B. Each component can be either a cycle, or an *AB-path* (that has one endpoint in genome A and the other in B), or an *AA-path* (that has both endpoints in genome A), or a *BB-path* (that has both endpoints in B). A special case of an *AA* or a *BB*-path is a *linear singleton*, that is a linear chromosome represented by an adjacency of type $\circ \ell \circ$, where ℓ contains only unique markers. Paths occur when the genomes are linear. For circular genomes, the graph $AG(A,B)$ is composed of cycles only, and may also have a special type of component composed of a single vertex, that corresponds to a circular chromosome composed only of markers that are not in $\mathcal{G}$, called *circular singleton*. In Figure 2 we show the adjacency graph built over the linear genomes represented in Figure 1.

DCJ operations

A *cut* performed on a genome A separates two adjacent markers of A. A cut affects a single adjacency ν in A: it is done between two symbols of ν, creating two open ends. In general a cut can be performed between two markers of a label, but the DCJ-indel distance can be computed considering only cuts that do not "break" labels. A *double-cut and join* or *DCJ* applied on a genome A is the operation that performs cuts in two different adjacencies in A, creating four open ends, and joins these open ends in a different way. In other words, a DCJ rearranges two adjacencies in A, transforming them into two new adjacencies. As an example consider a DCJ applied to genome A (from Figure 1), that rearranges the adjacencies $a^h v d^h$ and $d^t e^t$ into the new adjacencies $a^h v d^t$ and $d^h e^t$. Observe that this operation corresponds to the inversion of marker d in genome A. Indeed, a DCJ operation can correspond to several rearrangements, such as an inversion, a translocation, a fusion or a fission [2].

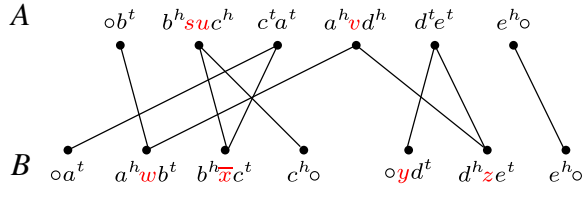

Figure 2 For genomes *A* and *B* (Figure 1), the graph has one *BB* and two *AB*-paths.

DCJ-sorting and DCJ distance

Given two genomes A and B, the components of $AG(A,B)$ with 3 or more vertices need to be reduced, by applying DCJ operations, to components with only 2 vertices, that can be cycles or AB-paths [14]. This procedure is called *DCJ-sorting* of A into B. The number of AB-paths in $AG(A,B)$ is always even and a DCJ can be of three types [7]: it can either decrease the number of cycles by one, or the number of AB-paths by two (*counter-optimal*); or it does not affect the number of cycles and AB-paths (*neutral*); or it can either increase the number of cycles by one, or the number of AB-paths by two (*optimal*). The DCJ distance of A and B, denoted by $d_{DCJ}(A,B)$, is the minimum number of steps required to do a DCJ-sorting of A into B, given by the following theorem.

Theorem 1 (from [3])**.** *Given two genomes A and B, we have* $d_{DCJ}(A,B) = |\mathcal{G}| - c - \frac{b}{2}$*, where* $\mathcal{G}$ *is the set of common markers and c and b are, respectively, the number of cycles and of AB-paths in* $AG(A,B)$*.*

Internal DCJ operations and recombinations

Observe that a DCJ operation ρ acts on two different adjacencies, that can be in the same or in two distinct connected components of the graph. The components on which the cuts are applied are called *sources* and the components obtained after the joinings are called *resultants* of ρ. With respect to the adjacency graph, ρ can be of two types: *internal*, when ρ is applied to two adjacencies belonging to a single component; and *recombination*, when ρ is applied to adjacencies belonging to two distinct components.

Any recombination applied to a vertex of an *AA*-path and a vertex of a *BB*-path is optimal [14]. A recombination applied to vertices of two distinct *AB*-paths can be either neutral, when the resultants are also *AB*-paths, or counter-optimal, when the resultants are an *AA*-path and a *BB*-path. All other types of path recombinations are neutral and all recombinations involving at least one cycle are counter-optimal.

It is possible to do a separate DCJ-sorting in any component P of $AG(A,B)$ [14] by applying DCJs internal to P. We denote by $d_{DCJ}(P)$ the number of optimal DCJ operations used for DCJ-sorting P separately ($d_{DCJ}(P)$ depends only on the number of vertices or, equivalently, the number of edges of P [14]). Thus, the DCJ distance can also be re-written in terms of the sum of the distance per component:

Lemma 1 (derived from [14])**.** *Given two genomes A and B, we have* $d_{DCJ}(A,B) = \sum_{P \in AG(A,B)} d_{DCJ}(P)$*.*

Only optimal DCJs, counted in the equivalent formulas given by Theorem 1 and Lemma 1, are necessary

to do a DCJ-sorting. Given a DCJ ρ, the *DCJ variation* of ρ, denoted by $\Delta_{DCJ}(\rho)$, is defined to be respectively 0, 1 and 2 depending whether ρ is optimal, neutral or counter-optimal.

Modifying the content of a genome

In the previous section, the unique markers appeared as labels of adjacencies, but the DCJ operations are only able to change the organization of the genomes. Here we introduce the operations that are applied to the labels and change the content of the genomes.

Indel operations

The most classical content-modifying operations are *insertions* and *deletions* of blocks of contiguous markers [4,6]. We refer to insertions and deletions as *indel* operations. In the model we consider, an indel only affects the label of one single adjacency, by deleting or inserting contiguous markers in this label, with the restriction that an insertion cannot produce duplicated markers [7]. Thus, while sorting A into B, the indels are the steps in which the markers in $\mathcal{A}$ are deleted and the markers in $\mathcal{B}$ are inserted. At most one chromosome can be entirely deleted or inserted at once. We illustrate an indel with the following example: the deletion of markers su from adjacency $b^h su c^h$ of genome A (Figure 2), which results into the clean adjacency $b^h c^h$. The opposite operation would be an insertion.

Substitutions

Substitutions are more powerful content-modifying operations, that allow blocks of contiguous markers to be substituted by other blocks of contiguous markers [8]. In other words, a deletion and a subsequent insertion that occur at the same position of the genome can be modeled as a substitution, counting together for one single sorting step.

A substitution only affects the label of one single adjacency, by substituting contiguous markers in this label, with the restriction that it cannot produce duplicated markers [8]. An example is the substitution of markers su in adjacency $b^h su c^h$ by $\overline{x}$, which results into adjacency $b^h \overline{x} c^h$. At most one chromosome can be entirely substituted at once (but we do not allow the substitution of a linear by a circular chromosome nor *vice-versa*). As previously mentioned, insertions and deletions are special cases of substitutions. If a block of markers is substituted by the empty string, we have a deletion. Analogously, if the empty string is substituted by a block of markers, we have an insertion.

Runs, indel- and substitution-potentials

In this section we introduce some definitions and concepts that will help us to integrate the DCJ model with content-modifying operations. These concepts will be very useful in our results, when we will show how to use DCJ operations to minimize the number of content-modifying operations to be performed.

First, let us recall the concept of *run*, introduced in [7]. Given two genomes A and B and a component P of $AG(A,B)$, a *run* is a maximal subpath of P, in which the first and the last vertices are labeled and all labeled vertices belong to the same genome (or partition). An example is given in Figure 3. A run in genome A is also called an $\mathcal{A}$-run, and a run in genome B is called a $\mathcal{B}$-run. We denote by $\Lambda(P)$ the number of runs in a component P. While a path can have any number or runs, a cycle has either 0, 1, or an even number of runs.

A set of labels of one genome can be accumulated with DCJs. For example, take the adjacencies $d^h z e^t$ and $d^t \overline{y}\circ$ from genome B (Figure 3). A DCJ applied to these two adjacencies could result into $d^t e^t$ and $d^h z \overline{y}\circ$, in which the label $z\overline{y}$ resulted from the accumulation of the labels of the two original adjacencies. In particular, when we apply optimal DCJs internal to a single component of the adjacency graph, we can accumulate an entire run into a single adjacency [7].

Runs can be merged by DCJ operations. Consequently, during the optimal DCJ-sorting of a component P, we can reduce its number of runs. The *indel-potential* of P, denoted by $\lambda(P)$, is defined in [7] as the minimum number of runs that we can obtain by DCJ-sorting P with optimal DCJ operations. An example is given in Figure 4.

The indel-potential of a component depends only on its number of runs:

Proposition 1 (from [7]). *Given two genomes A and B and a component P of $AG(A,B)$, the indel-potential of P is given by $\lambda(P) = \lceil \frac{\Lambda(P)+1}{2} \rceil$, if $\Lambda(P) \geq 1$. Otherwise, if $\Lambda(P) = 0$, then $\lambda(P) = 0$.*

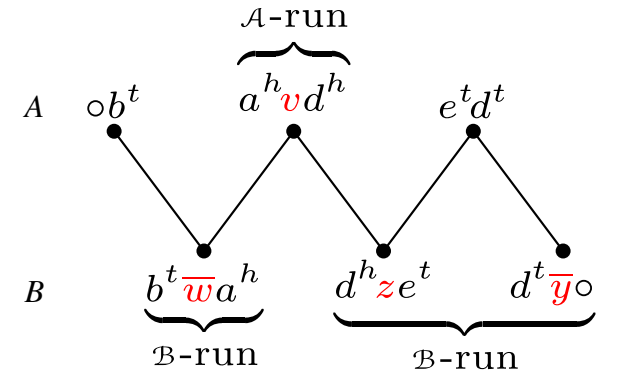

Figure 3 An *AB*-path with 3 runs (extracted from Figure 2).

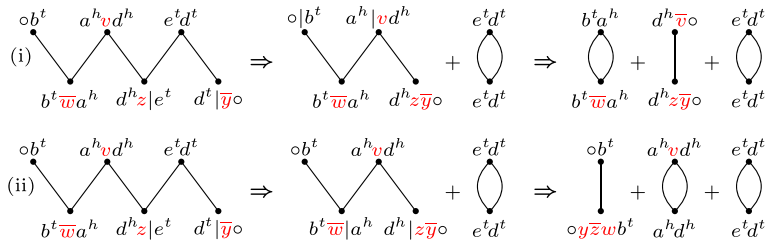

Figure 4 Two optimal sequences for DCJ-sorting an *AB*-path with $\Lambda = 3$ (the cuts of each DCJ in each sequence are represented by "|"). In **(i)** the overall number of runs in the resulting components is three, while in **(ii)** the resulting components have only two runs. Indeed, in this case, the best we can have is the indel-potential $\lambda = 2$.

Similarly, the *substitution-potential* of a component P is the minimum number of substitutions that we can obtain by DCJ-sorting P with optimal DCJ operations. The substitution-potential is denoted by $\sigma(P)$ and can be computed as follows:

Proposition 2 (from [8])**.** *Given genomes A and B and a component P of AG(A,B), the substitution-potential of P is given by* $\sigma(P) = \lceil \frac{\Lambda(P)+1}{4} \rceil$*, if* $\Lambda(P) \geq 1$*. Otherwise, if* $\Lambda(P) = 0$*, then* $\sigma(P) = 0$*.*

Results

In this section we show how to compute the DCJ-indel and the DCJ-substitution distances, considering that the content-modifying cost is distinct from and upper bounded by the DCJ cost. We assign the cost of 1 to each DCJ and a positive cost $w \leq 1$ to each content-modifying operation.

The DCJ-indel model with distinct operation costs

First we consider the case in which only indels are allowed as content-modifying operations. Given two genomes A and B, we define the *DCJ-indel distance* of A and B, denoted by $d_{DCJ}^{id}(A,B)$, as the minimum cost of a DCJ-indel sequence of operations that sorts A into B. If $w = 1$, the DCJ-indel distance corresponds exactly to the minimum number of steps required to sort A into B. To compute the distance in this case, a linear algorithm was given in [7]. Here we present a more general method to compute the DCJ-indel distance for any positive $w \leq 1$.

An upper bound for the DCJ-indel distance

We can obtain a good upper bound for the DCJ-indel distance by showing how to compute the DCJ-indel distance per component. Given a DCJ operation ρ, let λ_0 and λ_1 be, respectively, the sum of the indel-potentials for the components of the adjacency graph before and after ρ, and let $\Delta\lambda(\rho) = \lambda_1 - \lambda_0$. If ρ is an optimal DCJ internal to a single component of the graph, the definition of indel-potential implies $\Delta\lambda(\rho) \geq 0$. We also have $\Delta\lambda(\rho) \geq 0$, if ρ is counter-optimal, and $\Delta\lambda(\rho) \geq -1$, if ρ is neutral [7]. Recall that $\Delta_{DCJ}(\rho)$ is, respectively, 0, 1 and 2, depending whether the DCJ ρ is optimal, neutral or counter-optimal. We define $\Delta_{DCJ\text{-}\lambda}(\rho) = \Delta_{DCJ}(\rho) + w\Delta\lambda(\rho)$.

We know that each component P of $AG(A,B)$ can be DCJ-sorted separately, and the labels can then be easily sorted with indel operations. Let $d_{DCJ}^{id}(P)$ be the DCJ-indel distance of P, that is the minimum cost of a DCJ-indel sequence of operations sorting P separately. This can be computed according to the following proposition.

Proposition 3. *For each* $P \in AG(A,B)$, $d_{DCJ}^{id}(P) = d_{DCJ}(P) + w\lambda(P)$.

Proof. By the definition of λ, the best we can do with optimal DCJs is $d_{DCJ}(P) + w\lambda(P)$. From [7], we have $\Delta_{DCJ\text{-}\lambda}(\rho) \geq 2$ if ρ is counter-optimal, thus we can only get more expensive sorting scenarios if we use such operation. We also know that, if ρ is neutral $\Delta_{DCJ\text{-}\lambda}(\rho) \geq 1 - w \geq 0$, for any positive $w \leq 1$. □

This allows us to get a good upper bound for the DCJ-indel distance with distinct operation costs:

Lemma 2. *Given two genomes A and B and a positive indel cost* $w \leq 1$*, we have*

$$d_{DCJ}^{id}(A,B) \leq d_{DCJ}(A,B) + w \sum_{P \in AG(A,B)} \lambda(P).$$

Proof. If we sort the components separately we have $d_{DCJ}^{id}(A,B) \leq \sum_{P \in AG(A,B)} d_{DCJ}^{id}(P)$, which, according to Lemma 1 and Proposition 3, corresponds exactly to $d_{DCJ}(A,B) + w\sum_{P \in AG(A,B)} \lambda(P)$. □

Recombinations and the exact DCJ-indel distance

Until this point, we have explored the possible effects of any DCJ that is internal to a single component

of the graph. Now we will analyze the effect of recombinations, that have $\Delta\lambda \geq -2$ [7]. We saw previously that any recombination involving cycles is counter-optimal. Since any counter-optimal recombination has $\Delta_{DCJ\text{-}\lambda} \geq 2 - 2w \geq 0$, only path recombinations can have $\Delta_{DCJ\text{-}\lambda} < 0$.

Although the space of recombinations is not small, some observations allow us to explore it efficiently. Proposition 1 shows that the indel-potential increases of one when the number of runs increases of two. Furthermore, when we decrease the number of runs of a path by one, it will decrease the indel-potential only if its initial number of runs is one or a multiple of two. However, the exact number of runs does not really matter. In the path recombination analysis, we only have to consider the following properties for each path:

- whether it is an AA, or a BB, or an AB-path;
- whether it has zero, or an odd or an even number of runs; and
- whether its first run is in A or in B (by convention, an AB-path is always read from A to B).

An empty sequence (with no run) is represented by ε. For the benefit of the reader, for an integer $i \geq 0$, let $\mathcal{A}$ (respectively $\mathcal{B}$) be a sequence with odd $2i+1$ runs, starting and ending with an $\mathcal{A}$-run (respectively $\mathcal{B}$-run). Similarly, let $\mathcal{AB}$ (respectively $\mathcal{BA}$), be a sequence with even $2i+2$ runs, starting with an $\mathcal{A}$-run (respectively $\mathcal{B}$-run) and ending with a $\mathcal{B}$-run (respectively $\mathcal{A}$-run). Then each one of the notations AA_{ε}, $AA_{\mathcal{A}}$, $AA_{\mathcal{B}}$, $AA_{\mathcal{AB}} \equiv AA_{\mathcal{BA}}$, BB_{ε}, $BB_{\mathcal{A}}$, $BB_{\mathcal{B}}$, $BB_{\mathcal{AB}} \equiv BB_{\mathcal{BA}}$, AB_{ε}, $AB_{\mathcal{A}}$, $AB_{\mathcal{B}}$, $AB_{\mathcal{AB}}$ and $AB_{\mathcal{BA}}$ represents a particular type of path (AA, BB or AB) with a particular structure of runs (ε, $\mathcal{A}$, $\mathcal{B}$, $\mathcal{AB}$ or $\mathcal{BA}$). An example of this notation is given in Figure 5, which represents a neutral recombination possibly with $\Delta_{DCJ\text{-}\lambda} < 0$.

Each type of recombination can lead to different resultants, depending on where the cuts are applied. However, it is always possible to choose the "best" resultants in each case: we take the recombination with the smallest $\Delta_{DCJ\text{-}\lambda}$, whose resultants can be better reused in further recombinations. The main observations to guide this task are: only recombinations of paths whose runs are $\mathcal{AB}$ or $\mathcal{BA}$ have $\Delta\lambda = -2$ and only recombinations of type $AA + BB$ are optimal and have $\Delta_{DCJ} = 0$. In Table 1, we list all path recombinations that can have $\Delta_{DCJ\text{-}\lambda} < 0$, together with neutral recombinations that have $\Delta_{DCJ\text{-}\lambda} = 1 - w \geq 0$, but produce an $AA_{\mathcal{AB}}$ or a $BB_{\mathcal{AB}}$ path. We denote by $\bullet$ an AB-path that never appears as a source of a recombination in this table (these paths are AB_{ε}, $AB_{\mathcal{A}}$ and $AB_{\mathcal{B}}$).

The DCJ-indel distance formula By analyzing the whole universe of operations, we could identify groups of recombinations, as listed in Table 2. Since some resultants of recombinations can be used in other recombinations, the groups can have more than one recombination. Groups $\mathcal{P}$, $\mathcal{S}_1$ and $\mathcal{S}_2$ are composed of a single recombination, while groups $\mathcal{T}$, $\mathcal{N}_1$ and $\mathcal{N}_2$ are composed of two recombinations and groups $\mathcal{Q}$ and $\mathcal{M}$ are composed of three recombinations. recombination is not an associative operation, thus, in column 'DCJ seq.' of Table 2, we indicate how the sequence of DCJs must be applied in each group (the symbol $\prec$ separates preceeding and succeeding recombinations).

While in groups $\mathcal{Q}$ and $\mathcal{T}$ the preceding recombinations have lower $\Delta_{DCJ\text{-}\lambda}$, in groups $\mathcal{M}$, $\mathcal{N}_1$ and $\mathcal{N}_2$ we need to use operations of type $n_{\text{-}1}$ in order to prepare better recombinations. Another important observation concerning groups $\mathcal{Q}$ and $\mathcal{T}$ is that, although their $\Delta_{DCJ\text{-}\lambda}$ indicate that $\mathcal{Q}$ could be applied for $w > 1/4$ and $\mathcal{T}$ could be applied for $w > 1/3$, the last operation of these groups is of type $n_{\text{-}2}$ and actually increases $\Delta_{DCJ\text{-}\lambda}$ for $w \leq 1/2$. For this reason, we skip groups $\mathcal{Q}$ and $\mathcal{T}$ for $w \leq 1/2$ (there is no loss with this approach, since their optimal operations are then counted in $\mathcal{S}_1$).

The deductions shown in Table 2 can be computed with an approach that greedily maximizes the number of occurrences in $\mathcal{P}$, $\mathcal{Q}$, $\mathcal{T}$, $\mathcal{S}_1$, $\mathcal{S}_2$, $\mathcal{M}$, $\mathcal{N}_1$ and $\mathcal{N}_2$ in this order. The two groups in $\mathcal{Q}$ are mutually exclusive after maximizing $\mathcal{P}$. The lines in $\mathcal{T}$ are subgroups of the lines in $\mathcal{Q}$, that is, they are only computed when there are enough remaining components after maximizing $\mathcal{Q}$. Similarly, each one of the remaining groups are computed when there are enough remaining components after maximizing the upper groups. With the results presented in

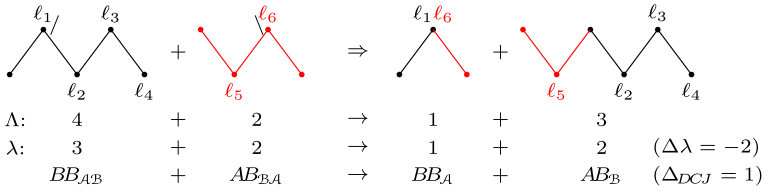

Figure 5 Neutral recombination that has $\Delta_{DCJ\text{-}\lambda} = 1 - 2w$ (we represent only the labels of the adjacencies, the cuts of the recombination are represented by "/" and "\").

Table 1 Path recombinations that are used to compute the DCJ-indel distance

	Sources	Resultants	$\Delta\lambda$	Δ_{DCJ}	$\Delta_{DCJ\text{-}\lambda}$		Sources	Resultants	$\Delta\lambda$	Δ_{DCJ}	$\Delta_{DCJ\text{-}\lambda}$
$o_{\text{-}2}$	$AA_{\mathcal{AB}} + BB_{\mathcal{AB}}$	$\bullet + \bullet$	-2	0	$-2w$						
						$n_{\text{-}2}$	$AA_{\mathcal{AB}} + AA_{\mathcal{AB}}$	$AA_{\mathcal{A}} + AA_{\mathcal{B}}$	-2	1	$1-2w$
$o_{\text{-}1}$	$AA_{\mathcal{A}} + BB_{\mathcal{AB}}$	$\bullet + AB_{\mathcal{AB}}$	-1	0	$-w$	$n_{\text{-}2}$	$BB_{\mathcal{AB}} + BB_{\mathcal{AB}}$	$BB_{\mathcal{A}} + BB_{\mathcal{B}}$	-2	1	$1-2w$
$o_{\text{-}1}$	$BB_{\mathcal{A}} + AA_{\mathcal{AB}}$	$\bullet + AB_{\mathcal{BA}}$	-1	0	$-w$	$n_{\text{-}2}$	$AA_{\mathcal{AB}} + AB_{\mathcal{AB}}$	$\bullet + AA_{\mathcal{A}}$	-2	1	$1-2w$
$o_{\text{-}1}$	$AA_{\mathcal{B}} + BB_{\mathcal{AB}}$	$\bullet + AB_{\mathcal{BA}}$	-1	0	$-w$	$n_{\text{-}2}$	$AA_{\mathcal{AB}} + AB_{\mathcal{BA}}$	$\bullet + AA_{\mathcal{B}}$	-2	1	$1-2w$
$o_{\text{-}1}$	$BB_{\mathcal{B}} + AA_{\mathcal{AB}}$	$\bullet + AB_{\mathcal{AB}}$	-1	0	$-w$	$n_{\text{-}2}$	$BB_{\mathcal{AB}} + AB_{\mathcal{AB}}$	$\bullet + BB_{\mathcal{B}}$	-2	1	$1-2w$
$o_{\text{-}1}$	$AA_{\mathcal{A}} + BB_{\mathcal{A}}$	$\bullet + \bullet$	-1	0	$-w$	$n_{\text{-}2}$	$BB_{\mathcal{AB}} + AB_{\mathcal{BA}}$	$\bullet + BB_{\mathcal{A}}$	-2	1	$1-2w$
$o_{\text{-}1}$	$AA_{\mathcal{B}} + BB_{\mathcal{B}}$	$\bullet + \bullet$	-1	0	$-w$	$n_{\text{-}2}$	$AB_{\mathcal{AB}} + AB_{\mathcal{BA}}$	$\bullet + \bullet$	-2	1	$1-2w$
$n_{\text{-}1}$	$AA_{\mathcal{A}} + AB_{\mathcal{BA}}$	$\bullet + AA_{\mathcal{AB}}$	-1	1	$1-w$	$n_{\text{-}1}$	$BB_{\mathcal{A}} + AB_{\mathcal{AB}}$	$\bullet + BB_{\mathcal{AB}}$	-1	1	$1-w$
$n_{\text{-}1}$	$AA_{\mathcal{B}} + AB_{\mathcal{AB}}$	$\bullet + AA_{\mathcal{AB}}$	-1	1	$1-w$	$n_{\text{-}1}$	$BB_{\mathcal{B}} + AB_{\mathcal{BA}}$	$\bullet + BB_{\mathcal{AB}}$	-1	1	$1-w$

Recombinations of type $o_{\text{-}2}$ (optimal with $\Delta\lambda = -2$), $o_{\text{-}1}$ (optimal with $\Delta\lambda = -1$) and $n_{\text{-}2}$ (neutral with $\Delta\lambda = -2$) can have $\Delta_{DCJ\text{-}\lambda} < 0$. Recombinations of type $n_{\text{-}1}$ (neutral with $\Delta\lambda = -1$) have $\Delta_{DCJ\text{-}\lambda} = 1 - w \geq 0$, but produce an $AA_{\mathcal{AB}}$ or a $BB_{\mathcal{AB}}$ path.

Table 2 All recombination groups that determine the deductions for computing the DCJ-indel distance

	Sources	Resultants	DCJ seq.	$\Delta_{DCJ\text{-}\lambda}$	skip if
$\mathcal{P}$	$AA_{\mathcal{AB}} + BB_{\mathcal{AB}}$	$2\bullet$	$o_{\text{-}2}$	$-2w$	
$\mathcal{Q}$	$2AA_{\mathcal{AB}} + BB_{\mathcal{A}} + BB_{\mathcal{B}}$	$4\bullet$	$2o_{\text{-}1} \prec n_{\text{-}2}$	$1-4w$	$w \leq \frac{1}{2}$
	$2BB_{\mathcal{AB}} + AA_{\mathcal{A}} + AA_{\mathcal{B}}$	$4\bullet$		$1-4w$	
$\mathcal{T}$	$AA_{\mathcal{AB}} + BB_{\mathcal{A}} + AB_{\mathcal{AB}}$	$3\bullet$	$o_{\text{-}1} \prec n_{\text{-}2}$	$1-3w$	$w \leq \frac{1}{2}$
	$AA_{\mathcal{AB}} + BB_{\mathcal{B}} + AB_{\mathcal{BA}}$	$3\bullet$		$1-3w$	
	$BB_{\mathcal{AB}} + AA_{\mathcal{A}} + AB_{\mathcal{BA}}$	$3\bullet$		$1-3w$	
	$BB_{\mathcal{AB}} + AA_{\mathcal{B}} + AB_{\mathcal{AB}}$	$3\bullet$		$1-3w$	
	$2BB_{\mathcal{AB}} + AA_{\mathcal{A}}$	$2\bullet + BB_{\mathcal{B}}$		$1-3w$	
	$2BB_{\mathcal{AB}} + AA_{\mathcal{B}}$	$2\bullet + BB_{\mathcal{A}}$		$1-3w$	
	$2AA_{\mathcal{AB}} + BB_{\mathcal{A}}$	$2\bullet + AA_{\mathcal{B}}$		$1-3w$	
	$2AA_{\mathcal{AB}} + BB_{\mathcal{B}}$	$2\bullet + AA_{\mathcal{A}}$		$1-3w$	
$\mathcal{S}_1$	$AA_{\mathcal{A}} + BB_{\mathcal{A}}$	$2\bullet$	$o_{\text{-}1}$	$-w$	
	$AA_{\mathcal{B}} + BB_{\mathcal{B}}$	$2\bullet$		$-w$	
	$AA_{\mathcal{AB}} + BB_{\mathcal{A}}$	$\bullet + AB_{\mathcal{BA}}$		$-w$	
	$AA_{\mathcal{AB}} + BB_{\mathcal{B}}$	$\bullet + AB_{\mathcal{AB}}$		$-w$	
	$BB_{\mathcal{AB}} + AA_{\mathcal{A}}$	$\bullet + AB_{\mathcal{AB}}$		$-w$	
	$BB_{\mathcal{AB}} + AA_{\mathcal{B}}$	$\bullet + AB_{\mathcal{BA}}$		$-w$	
$\mathcal{S}_2$	$AB_{\mathcal{AB}} + AB_{\mathcal{BA}}$	$2\bullet$	$n_{\text{-}2}$	$1-2w$	$w \leq \frac{1}{2}$
	$AA_{\mathcal{AB}} + AB_{\mathcal{AB}}$	$\bullet + AA_{\mathcal{A}}$		$1-2w$	
	$AA_{\mathcal{AB}} + AB_{\mathcal{BA}}$	$\bullet + AA_{\mathcal{B}}$		$1-2w$	
	$BB_{\mathcal{AB}} + AB_{\mathcal{AB}}$	$\bullet + BB_{\mathcal{B}}$		$1-2w$	
	$BB_{\mathcal{AB}} + AB_{\mathcal{BA}}$	$\bullet + BB_{\mathcal{A}}$		$1-2w$	

Table 2 All recombination groups that determine the deductions for computing the DCJ-indel distance *(continued)*

	$AA_{\mathcal{AB}} + AA_{\mathcal{AB}}$	$AA_{\mathcal{A}} + AA_{\mathcal{B}}$		$1 - 2w$	
	$BB_{\mathcal{AB}} + BB_{\mathcal{AB}}$	$BB_{\mathcal{A}} + BB_{\mathcal{B}}$		$1 - 2w$	
$\mathcal{M}$	$2AB_{\mathcal{AB}} + AA_{\mathcal{B}} + BB_{\mathcal{A}}$	$4\bullet$	$2n_{-1} \prec o_{-2}$	$2 - 4w$	$w \leq \frac{1}{2}$
	$2AB_{\mathcal{BA}} + AA_{\mathcal{A}} + BB_{\mathcal{B}}$	$4\bullet$		$2 - 4w$	
$\mathcal{N}_1$	$AB_{\mathcal{AB}} + AA_{\mathcal{B}} + BB_{\mathcal{A}}$	$2\bullet + AB_{\mathcal{BA}}$	$n_{-1} \prec o_{-1}$	$1 - 2w$	$w \leq \frac{1}{2}$
	$AB_{\mathcal{BA}} + AA_{\mathcal{A}} + BB_{\mathcal{B}}$	$2\bullet + AB_{\mathcal{AB}}$		$1 - 2w$	
$\mathcal{N}_2$	$2AB_{\mathcal{AB}} + AA_{\mathcal{B}}$	$2\bullet + AA_{\mathcal{A}}$	$n_{-1} \prec n_{-2}$	$2 - 3w$	$w \leq \frac{2}{3}$
	$2AB_{\mathcal{AB}} + BB_{\mathcal{A}}$	$2\bullet + BB_{\mathcal{B}}$		$2 - 3w$	
	$2AB_{\mathcal{BA}} + AA_{\mathcal{A}}$	$2\bullet + AA_{\mathcal{B}}$		$2 - 3w$	
	$2AB_{\mathcal{BA}} + BB_{\mathcal{B}}$	$2\bullet + BB_{\mathcal{A}}$		$2 - 3w$	

this section we have an exact formula to compute the DCJ-indel distance:

Theorem 2. *Given two genomes A and B and a positive indel cost* $w \leq 1$*,*

$$\begin{aligned} d_{DCJ}^{id}(A,B) = d_{DCJ}(A,B) + w \sum_{P \in AG(A,B)} \lambda(P) - 2w\mathcal{P} \\ - (4w-1)\mathcal{Q} - (3w-1)\mathcal{T} \\ - w\mathcal{S}_1 - (2w-1)(\mathcal{S}_2 + 2\mathcal{M} + \mathcal{N}_1) \\ - (3w-2)\mathcal{N}_2, \end{aligned}$$

where $\mathcal{P}$*,* $\mathcal{Q}$*,* $\mathcal{T}$*,* $\mathcal{S}_1$*,* $\mathcal{S}_2$*,* $\mathcal{M}$*,* $\mathcal{N}_1$ *and* $\mathcal{N}_2$ *are computed as described above.*

As we mentioned before, the groups $\mathcal{Q}$ and $\mathcal{T}$ are skipped ($\mathcal{Q} = \mathcal{T} = 0$) for $w \leq 1/2$. Furthermore, we also have $\mathcal{S}_2 = \mathcal{M} = \mathcal{N}_1 = 0$ if $w \leq 1/2$ and $\mathcal{N}_2 = 0$ if $w \leq 2/3$. Although some groups have reusable resultants, those are actually never reused (if groups that are lower in the table use as sources resultants from higher groups, the sources of all referred groups would be previously consumed in groups that occupy even higher positions in the table). Due to this fact, the number of occurrences in each group depends only on w and the initial number of each type of component.

Observe that, for $w = 1$, our formula is identical to the one proposed in [7]. Actually, for any $2/3 < w \leq 1$, the two formulas are equivalent, since the same occurrences of groups of recombinations and an equivalent upper bound are taken into account.

We illustrate the result of our formula with an example. Let $AG(A,B)$ have only the following labeled paths: two $AA_{\mathcal{AB}}$, one $BB_{\mathcal{A}}$ and one $BB_{\mathcal{B}}$. In this case, there are no occurrences of $\mathcal{P}$, thus we have $\mathcal{P} = 0$. If we take $w > \frac{1}{2}$, all labeled paths are consumed in one occurrence of $\mathcal{Q}$. We have $\mathcal{Q} = 1$, while all other values are zero, resulting in $\Delta_{DCJ\text{-}\lambda} = 1 - 4w$. On the other hand, if $w \leq \frac{1}{2}$, we automatically set $\mathcal{Q} = \mathcal{T} = \mathcal{S}_2 = \mathcal{M} = \mathcal{N}_1 = \mathcal{N}_2 = 0$. The labeled paths are consumed in two occurrences of $\mathcal{S}_1$, that is, $\mathcal{S}_1 = 2$, resulting in $\Delta_{DCJ\text{-}\lambda} = -2w$. For sure, $-2w \leq 1 - 4w$ only if $w \leq \frac{1}{2}$.

The DCJ-substitution model with distinct operation costs

Now we consider a different model in which substitutions are the content-modifying operations. Recall that substitutions include indels. Again we assign the cost of 1 to each DCJ and the cost of $w \leq 1$ to each substitution. The *DCJ-substitution distance* of genomes A and B, denoted by $d_{DCJ}^{sb}(A,B)$, is then the minimum cost of a DCJ-substitution sequence that sorts A into B. If $w = 1$, this corresponds exactly to the minimum number of steps required to sort A into B and can be computed in linear time [8]. Here we present a general method to compute the DCJ-substitution distance for any positive $w \leq 1$. Similarly to the approach used with the DCJ-indel model, we will first use internal DCJs to obtain a good upper bound and then analyze recombinations to compute the exact DCJ-substitution distance.

An upper bound for the DCJ-substitution distance

We can also obtain a good upper bound for the DCJ-substitution distance by showing how to compute the DCJ-substitution distance per component. Given a DCJ operation ρ, let σ_0 and σ_1 be, respectively, the sum of the substitution-potentials for the components of the adjacency graph before and after ρ, and let $\Delta\sigma(\rho) = \sigma_1 - \sigma_0$. If ρ is an optimal DCJ internal to a single component of the graph, the definition of substitution-potential implies

$\Delta\sigma(\rho) \geq 0$. We also have $\Delta\sigma(\rho) \geq 0$, if ρ is counter-optimal, and $\Delta\sigma(\rho) \geq -1$, if ρ is neutral [8]. We define $\Delta_{DCJ\text{-}\sigma}(\rho) = \Delta_{DCJ}(\rho) + w\Delta\sigma(\rho)$.

After DCJ-sorting a component P of $AG(A,B)$, the remaining labels can be easily sorted with substitutions. Let $d^{sb}_{DCJ}(P)$ be the DCJ-substitution distance of P, that is the minimum cost of a DCJ-substitution sequence of operations sorting P separately. This is given by the following proposition.

Proposition 4. *For each* $P \in AG(A,B)$, $d^{sb}_{DCJ}(P) = d_{DCJ}(P) + w\sigma(P)$.

Proof. Analogous to the proof of Proposition 3. □

If P is a singleton in $AG(A,B)$, $d^{sb}_{DCJ}(P) = w$ (the indel of the whole chromosome). A linear cannot be substituted by a circular singleton and *vice-versa*. However, a pair composed by a singleton in genome A and a singleton in genome B, such that both are linear or both are circular, can be sorted with one substitution (which saves one sorting step per pair). Let P_{LS} and P_{CS} be, respectively, the maximum number of disjoint pairs of linear and circular singletons in $AG(A,B)$. Together with Proposition 4, these numbers give a good upper bound for the DCJ-substitution distance:

Lemma 3. *Given genomes A and B and a positive substitution cost* $w \leq 1$,

$$d^{sb}_{DCJ}(A,B) \leq d_{DCJ}(A,B) + w \sum_{P \in AG(A,B)} \sigma(P) - w(P_{LS} + P_{CS}),$$

where P_{LS} *and* P_{CS} *are the numbers of disjoint pairs of linear and circular singletons.*

Proof. If we sort the components separately we have $d^{sb}_{DCJ}(A,B) \leq \sum_{P \in AG(A,B)} d^{sb}_{DCJ}(P)$, which, according to Lemma 1 and Proposition 4, corresponds exactly to $d_{DCJ}(A,B) + w\sum_{P \in AG(A,B)} \sigma(P)$. □

Recombinations and the exact DCJ-substitution distance

Now we also need to analyze the effect of path recombinations, that have $\Delta\sigma(\rho) \geq -2$ [8], in the DCJ-substitution distance. Here the space of recombinations is even larger, but can still be efficiently explored. Proposition 2 shows that the substitution-potential increases of one when the number of runs increases of four. Furthermore, when we decrease the number of runs of a path by one, it will decrease the indel-potential only if its initial number of runs is one or a multiple of four. Again, the exact number of runs does not really matter. We have to consider the following properties for each path:

- whether it is an AA, or a BB, or an AB-path;
- whether it has zero, or a number of runs that is a multiple of four, or a multiple of four plus 1, or a multiple of four plus 2, or a multiple of four plus 3; and
- whether its first run is in A or in B (by convention, an AB-path is always read from A to B).

Recall that an empty sequence (with no run) is represented by ε. For labeled paths we adopt a different meaning for $\mathcal{A}$, $\mathcal{B}$, $\mathcal{AB}$, $\mathcal{BA}$: for an integer $i \geq 0$, let $\mathcal{A}$ (respectively $\mathcal{B}$) be a sequence with odd $4i+1$ runs, starting and ending with an $\mathcal{A}$-run (respectively $\mathcal{B}$-run), and let $\mathcal{AB}$ (respectively $\mathcal{BA}$), be a sequence with even $4i+2$ runs, starting with an $\mathcal{A}$-run (respectively $\mathcal{B}$-run) and ending with a $\mathcal{B}$-run (respectively $\mathcal{A}$-run). Here we still have some additional cases: let $\mathcal{ABA}$ (respectively $\mathcal{BAB}$) be a sequence with odd $4i+3$ runs, starting and ending with an $\mathcal{A}$-run (respectively $\mathcal{B}$-run), and let $\mathcal{ABAB}$ (respectively $\mathcal{BABA}$), be a sequence with even $4i+4$ runs, starting with an $\mathcal{A}$-run (respectively $\mathcal{B}$-run) and ending with a $\mathcal{B}$-run (respectively $\mathcal{A}$-run). Then, for each type of path (AA, BB or AB) with a particular structure of runs ($\mathcal{A}$, $\mathcal{B}$, $\mathcal{AB}$, $\mathcal{BA}$, $\mathcal{ABA}$, $\mathcal{BAB}$, $\mathcal{ABAB}$, or $\mathcal{BABA}$), we have a particular notation. An example of this notation is given in Figure 6, which represents a neutral recombination with $\Delta_{DCJ\text{-}\sigma} = 1 - w$.

Again, although each type of recombination can lead to different resultants, it is always possible to choose the "best" resultants in each case: we take the recombination with the smallest $\Delta_{DCJ\text{-}\sigma}$, whose resultants can be better reused. In Table 3, we list all recombinations that can have $\Delta_{DCJ\text{-}\sigma} < 0$, together with those that have $\Delta_{DCJ\text{-}\sigma} = 1 - w \geq 0$, but produce an AA or a BB-path with runs $\mathcal{ABAB}$ or $\mathcal{A}$ or $\mathcal{B}$. We denote by $\bullet$ an AB-path that never appears as a source in this table (these are all AB paths, with the exception of $AB_{\mathcal{AB},\mathcal{AB}}$ and $AB_{\mathcal{B},\mathcal{ABA}}$).

The DCJ-substitution distance formula In Table 4 we list groups of recombinations, which allow the computation of the exact DCJ-substitution distance, with an approach that greedily maximizes the number of occurrences in $\mathcal{U}, \mathcal{V}, \mathcal{W}, \mathcal{X}_1, \mathcal{X}_2, \mathcal{Y}, \mathcal{Z}_1$ and $\mathcal{Z}_2$ in this order. The two groups in $\mathcal{V}$ are mutually exclusive after maximizing $\mathcal{U}$, while those in $\mathcal{W}$ are subgroups of $\mathcal{V}$ (they are only computed when there are enough remaining components after maximizing $\mathcal{V}$). Similarly, each one of the remaining groups are computed when there are enough remaining components after maximizing the upper groups. As previously observed, the recombination is not associative, thus the column 'DCJ seq.' determines in which order the sequence of DCJs must be applied in each group. Here we also need to skip some recombinations depending on the value of w. In particular, although $\Delta_{DCJ\text{-}\sigma}$ indicates that $\mathcal{W}$ could be applied for $w > 1/3$ and $\mathcal{V}$ for $w > 1/4$, the last operation of these groups is of type $n_{\text{-}2}$ and

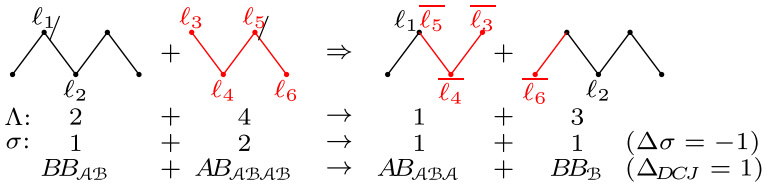

Figure 6 Neutral recombination that has $\Delta_{DCJ\text{-}\sigma} = 1-w$ (we represent only the labels of the adjacencies, the cuts of the recombination are represented by "/").

increases $\Delta_{DCJ\text{-}\sigma}$ for $w \leq 1/2$. Groups $\mathcal{V}$ and $\mathcal{W}$ are skipped for $w \leq 1/2$, and their optimal operations are then counted in $\mathcal{X}_1$.

The recombinations allow us to obtain an exact formula for the DCJ-substitution distance:

Theorem 3. *Given genomes A and B and a positive substitution cost $w \leq 1$,*

$$\begin{aligned} d^{sb}_{DCJ}(A,B) = d_{DCJ}(A,B) + w \sum_{P \in AG(A,B)} \sigma(P) \\ - 2w\mathcal{U} - (4w-1)\mathcal{V} - (3w-1)\mathcal{W} \\ - w\mathcal{X}_1 - (2w-1)(\mathcal{X}_2 + 2\mathcal{Y} + \mathcal{Z}_1) \\ - (3w-2)\mathcal{Z}_2 - w(P_{IS} + P_{CS}), \end{aligned}$$

where $\mathcal{U}$, $\mathcal{V}$, $\mathcal{W}$, $\mathcal{X}_1$, $\mathcal{X}_2$, $\mathcal{Y}$, $\mathcal{Z}_1$ and $\mathcal{Z}_2$ are computed as described above and P_{LS} and P_{CS} are the numbers of disjoint pairs of linear and circular singletons.

Observe that the number of occurrences in each group depends only on w and the initial number of each type of component and, for any $2/3 < w \leq 1$, our formula is equivalent to the one proposed in [8], since the same occurrences of groups of recombinations and an equivalent upper bound are taken into account.

Complexity

Both $AG(A,B)$ and $d_{DCJ}(A,B)$ can be computed in linear time [3]. The occurrences in each recombination group

Table 3 Path recombinations that are used to compute the DCJ-substitution distance

	Sources	Result.	$\Delta\sigma$	Δ_{DCJ}	$\Delta_{DCJ\text{-}\sigma}$		Sources	Result.	$\Delta\sigma$	Δ_{DCJ}	$\Delta_{DCJ\text{-}\sigma}$
						$o_{\text{-}1}$	$AA_{\mathcal{A}} + BB_{\mathcal{AB},\mathcal{AB}}$	$\bullet + AB_{\mathcal{AB},\mathcal{AB}}$	−1	0	$-w$
$o_{\text{-}2}$	$AA_{\mathcal{AB},\mathcal{AB}} + BB_{\mathcal{AB},\mathcal{AB}}$	$\bullet + \bullet$	−2	0	$-2w$	$o_{\text{-}1}$	$AA_{\mathcal{B}} + BB_{\mathcal{AB},\mathcal{AB}}$	$\bullet + AB_{\mathcal{B},\mathcal{AB},\mathcal{A}}$	−1	0	$-w$
						$o_{\text{-}1}$	$AA_{\mathcal{AB},\mathcal{AB}} + BB_{\mathcal{A}}$	$\bullet + AB_{\mathcal{B},\mathcal{AB},\mathcal{A}}$	−1	0	$-w$
						$o_{\text{-}1}$	$AA_{\mathcal{AB},\mathcal{AB}} + BB_{\mathcal{B}}$	$\bullet + AB_{\mathcal{AB},\mathcal{AB}}$	−1	0	$-w$
$o_{\text{-}1}$	$AA_{\mathcal{A}} + BB_{\mathcal{AB},\mathcal{A}}$	$\bullet + \bullet$	−1	0	$-w$						
$o_{\text{-}1}$	$AA_{\mathcal{B}} + BB_{\mathcal{B},\mathcal{AB}}$	$\bullet + \bullet$	−1	0	$-w$	$n_{\text{-}2}$	$AA_{\mathcal{AB},\mathcal{AB}} + AA_{\mathcal{AB},\mathcal{AB}}$	$AA_{\mathcal{AB},\mathcal{A}} + AA_{\mathcal{B},\mathcal{AB}}$	−2	1	$1-2w$
$o_{\text{-}1}$	$AA_{\mathcal{AB},\mathcal{A}} + BB_{\mathcal{A}}$	$\bullet + \bullet$	−1	0	$-w$	$n_{\text{-}2}$	$BB_{\mathcal{AB},\mathcal{AB}} + BB_{\mathcal{AB},\mathcal{AB}}$	$BB_{\mathcal{AB},\mathcal{A}} + BB_{\mathcal{B},\mathcal{AB}}$	−2	1	$1-2w$
$o_{\text{-}1}$	$AA_{\mathcal{B},\mathcal{AB}} + BB_{\mathcal{B}}$	$\bullet + \bullet$	−1	0	$-w$	$n_{\text{-}2}$	$AA_{\mathcal{AB},\mathcal{AB}} + AB_{\mathcal{AB},\mathcal{AB}}$	$\bullet + AA_{\mathcal{AB},\mathcal{A}}$	−2	1	$1-2w$
$o_{\text{-}1}$	$AA_{\mathcal{AB}} + BB_{\mathcal{AB},\mathcal{AB}}$	$\bullet + \bullet$	−1	0	$-w$	$n_{\text{-}2}$	$AA_{\mathcal{AB},\mathcal{AB}} + AB_{\mathcal{B},\mathcal{AB},\mathcal{A}}$	$\bullet + AA_{\mathcal{B},\mathcal{AB}}$	−2	1	$1-2w$
$o_{\text{-}1}$	$AA_{\mathcal{AB},\mathcal{AB}} + BB_{\mathcal{AB}}$	$\bullet + \bullet$	−1	0	$-w$	$n_{\text{-}2}$	$BB_{\mathcal{AB},\mathcal{AB}} + AB_{\mathcal{AB},\mathcal{AB}}$	$\bullet + BB_{\mathcal{B},\mathcal{AB}}$	−2	1	$1-2w$
$o_{\text{-}1}$	$AA_{\mathcal{AB}} + BB_{\mathcal{AB}}$	$\bullet + \bullet$	−1	0	$-w$	$n_{\text{-}2}$	$BB_{\mathcal{AB},\mathcal{AB}} + AB_{\mathcal{B},\mathcal{AB},\mathcal{A}}$	$\bullet + BB_{\mathcal{AB},\mathcal{A}}$	−2	1	$1-2w$
$o_{\text{-}1}$	$AA_{\mathcal{AB},\mathcal{A}} + BB_{\mathcal{AB},\mathcal{AB}}$	$\bullet + \bullet$	−1	0	$-w$	$n_{\text{-}2}$	$AB_{\mathcal{AB},\mathcal{AB}} + AB_{\mathcal{B},\mathcal{AB},\mathcal{A}}$	$\bullet + \bullet$	−2	1	$1-2w$
$o_{\text{-}1}$	$AA_{\mathcal{B},\mathcal{AB}} + BB_{\mathcal{AB},\mathcal{AB}}$	$\bullet + \bullet$	−1	0	$-w$						
$o_{\text{-}1}$	$AA_{\mathcal{AB},\mathcal{AB}} + BB_{\mathcal{AB},\mathcal{A}}$	$\bullet + \bullet$	−1	0	$-w$	$n_{\text{-}1}$	$AA_{\mathcal{A}} + AB_{\mathcal{B},\mathcal{AB},\mathcal{A}}$	$\bullet + AA_{\mathcal{AB},\mathcal{AB}}$	−1	1	$1-w$
$o_{\text{-}1}$	$AA_{\mathcal{AB},\mathcal{AB}} + BB_{\mathcal{B},\mathcal{AB}}$	$\bullet + \bullet$	−1	0	$-w$	$n_{\text{-}1}$	$AA_{\mathcal{B}} + AB_{\mathcal{AB},\mathcal{AB}}$	$\bullet + AA_{\mathcal{AB},\mathcal{AB}}$	−1	1	$1-w$
$o_{\text{-}1}$	$AA_{\mathcal{A}} + BB_{\mathcal{A}}$	$\bullet + \bullet$	−1	0	$-w$	$n_{\text{-}1}$	$BB_{\mathcal{A}} + AB_{\mathcal{AB},\mathcal{AB}}$	$\bullet + BB_{\mathcal{AB},\mathcal{AB}}$	−1	1	$1-w$
$o_{\text{-}1}$	$AA_{\mathcal{B}} + BB_{\mathcal{B}}$	$\bullet + \bullet$	−1	0	$-w$	$n_{\text{-}1}$	$BB_{\mathcal{B}} + AB_{\mathcal{B},\mathcal{AB},\mathcal{A}}$	$\bullet + BB_{\mathcal{AB},\mathcal{AB}}$	−1	1	$1-w$
$o_{\text{-}1}$	$AA_{\mathcal{A}} + BB_{\mathcal{AB}}$	$\bullet + \bullet$	−1	0	$-w$	$n_{\text{-}1}$	$AA_{\mathcal{AB}} + AB_{\mathcal{AB},\mathcal{AB}}$	$\bullet + AA_{\mathcal{A}}$	−1	1	$1-w$
$o_{\text{-}1}$	$AA_{\mathcal{B}} + BB_{\mathcal{AB}}$	$\bullet + \bullet$	−1	0	$-w$	$n_{\text{-}1}$	$AA_{\mathcal{AB}} + AB_{\mathcal{B},\mathcal{AB},\mathcal{A}}$	$\bullet + AA_{\mathcal{B}}$	−1	1	$1-w$
$o_{\text{-}1}$	$AA_{\mathcal{AB}} + BB_{\mathcal{A}}$	$\bullet + \bullet$	−1	0	$-w$	$n_{\text{-}1}$	$BB_{\mathcal{AB}} + AB_{\mathcal{B},\mathcal{AB},\mathcal{A}}$	$\bullet + BB_{\mathcal{A}}$	−1	1	$1-w$
$o_{\text{-}1}$	$AA_{\mathcal{AB}} + BB_{\mathcal{B}}$	$\bullet + \bullet$	−1	0	$-w$	$n_{\text{-}1}$	$BB_{\mathcal{AB}} + AB_{\mathcal{AB},\mathcal{AB}}$	$\bullet + BB_{\mathcal{B}}$	−1	1	$1-w$

Recombinations of type $o_{\text{-}2}$, $o_{\text{-}1}$ and $n_{\text{-}2}$ can have $\Delta_{DCJ\text{-}\sigma} < 0$. Recombinations of type $n_{\text{-}1}$ have $\Delta_{DCJ\text{-}\sigma} \geq 0$, but produce an AA or a BB-path with runs $\mathcal{AB},\mathcal{AB}$ or $\mathcal{A}$ or $\mathcal{B}$.

Table 4 All recombination groups that determine the deductions for computing the DCJ-substitution distance

	Sources	Resultants	DCJ seq.	$\Delta_{DCJ\text{-}\sigma}$	skip if
$\mathcal{U}$	$AA_{\mathcal{AB},\mathcal{AB}} + BB_{\mathcal{AB},\mathcal{AB}}$	$2\bullet$	o_{-2}	$-2w$	
$\mathcal{V}$	$2AA_{\mathcal{AB},\mathcal{AB}} + BB_{\mathcal{A}} + BB_{\mathcal{B}}$	$4\bullet$	$2o_{-1} \prec n_{-2}$	$1-4w$	$w \geq \frac{1}{2}$
	$2BB_{\mathcal{AB},\mathcal{AB}} + AA_{\mathcal{A}} + AA_{\mathcal{B}}$	$4\bullet$		$1-4w$	
$\mathcal{W}$	$AA_{\mathcal{AB},\mathcal{AB}} + BB_{\mathcal{A}} + AB_{\mathcal{AB},\mathcal{AB}}$	$3\bullet$	$o_{-1} \prec n_{-2}$	$1-3w$	$w \geq \frac{1}{2}$
	$AA_{\mathcal{AB},\mathcal{AB}} + BB_{\mathcal{B}} + AB_{\mathcal{B},\mathcal{AB},\mathcal{A}}$	$3\bullet$		$1-3w$	
	$BB_{\mathcal{AB},\mathcal{AB}} + AA_{\mathcal{A}} + AB_{\mathcal{B},\mathcal{AB},\mathcal{A}}$	$3\bullet$		$1-3w$	
	$BB_{\mathcal{AB},\mathcal{AB}} + AA_{\mathcal{B}} + AB_{\mathcal{AB},\mathcal{AB}}$	$3\bullet$		$1-3w$	
	$2AA_{\mathcal{AB},\mathcal{AB}} + BB_{\mathcal{A}}$	$2\bullet + AA_{\mathcal{B},\mathcal{AB}}$		$1-3w$	
	$2AA_{\mathcal{AB},\mathcal{AB}} + BB_{\mathcal{B}}$	$2\bullet + AA_{\mathcal{AB},\mathcal{A}}$		$1-3w$	
	$2BB_{\mathcal{AB},\mathcal{AB}} + AA_{\mathcal{A}}$	$2\bullet + BB_{\mathcal{B},\mathcal{AB}}$		$1-3w$	
	$2BB_{\mathcal{AB},\mathcal{AB}} + AA_{\mathcal{B}}$	$2\bullet + BB_{\mathcal{AB},\mathcal{A}}$		$1-3w$	
$\mathcal{X}_1$	$AA_{\mathcal{A}} + BB_{\mathcal{AB},\mathcal{AB}}$	$\bullet + AB_{\mathcal{AB},\mathcal{AB}}$	o_{-1}	$-w$	
	$AA_{\mathcal{B}} + BB_{\mathcal{AB},\mathcal{AB}}$	$\bullet + AB_{\mathcal{B},\mathcal{AB},\mathcal{A}}$		$-w$	
	$AA_{\mathcal{AB},\mathcal{AB}} + BB_{\mathcal{A}}$	$\bullet + AB_{\mathcal{B},\mathcal{AB},\mathcal{A}}$		$-w$	
	$AA_{\mathcal{AB},\mathcal{AB}} + BB_{\mathcal{B}}$	$\bullet + AB_{\mathcal{AB},\mathcal{AB}}$		$-w$	
	$AA_{\mathcal{AB}} + BB_{\mathcal{AB},\mathcal{AB}}$	$\bullet + \bullet$		$-w$	
	$AA_{\mathcal{AB},\mathcal{AB}} + BB_{\mathcal{AB}}$	$\bullet + \bullet$		$-w$	
	$AA_{\mathcal{AB}} + BB_{\mathcal{AB}}$	$\bullet + \bullet$		$-w$	
	$AA_{\mathcal{AB},\mathcal{A}} + BB_{\mathcal{AB},\mathcal{AB}}$	$\bullet + \bullet$		$-w$	
	$AA_{\mathcal{B},\mathcal{AB}} + BB_{\mathcal{AB},\mathcal{AB}}$	$\bullet + \bullet$		$-w$	
	$AA_{\mathcal{AB},\mathcal{AB}} + BB_{\mathcal{AB},\mathcal{A}}$	$\bullet + \bullet$		$-w$	
	$AA_{\mathcal{AB},\mathcal{AB}} + BB_{\mathcal{B},\mathcal{AB}}$	$\bullet + \bullet$		$-w$	
	$AA_{\mathcal{A}} + BB_{\mathcal{A}}$	$\bullet + \bullet$		$-w$	
	$AA_{\mathcal{B}} + BB_{\mathcal{B}}$	$\bullet + \bullet$		$-w$	
	$AA_{\mathcal{A}} + BB_{\mathcal{AB}}$	$\bullet + \bullet$		$-w$	
	$AA_{\mathcal{B}} + BB_{\mathcal{AB}}$	$\bullet + \bullet$		$-w$	
	$AA_{\mathcal{AB}} + BB_{\mathcal{A}}$	$\bullet + \bullet$		$-w$	
	$AA_{\mathcal{AB}} + BB_{\mathcal{B}}$	$\bullet + \bullet$		$-w$	
	$AA_{\mathcal{A}} + BB_{\mathcal{AB},\mathcal{A}}$	$\bullet + \bullet$		$-w$	
	$AA_{\mathcal{B}} + BB_{\mathcal{B},\mathcal{AB}}$	$\bullet + \bullet$		$-w$	
	$AA_{\mathcal{AB},\mathcal{A}} + BB_{\mathcal{A}}$	$\bullet + \bullet$		$-w$	
	$AA_{\mathcal{B},\mathcal{AB}} + BB_{\mathcal{B}}$	$\bullet + \bullet$		$-w$	
$\mathcal{X}_2$	$AA_{\mathcal{AB},\mathcal{AB}} + AA_{\mathcal{AB},\mathcal{AB}}$	$AA_{\mathcal{AB},\mathcal{A}} + AA_{\mathcal{B},\mathcal{AB}}$	n_{-2}	$1-2w$	$w \geq \frac{1}{2}$
	$BB_{\mathcal{AB},\mathcal{AB}} + BB_{\mathcal{AB},\mathcal{AB}}$	$BB_{\mathcal{AB},\mathcal{A}} + BB_{\mathcal{B},\mathcal{AB}}$		$1-2w$	
	$AA_{\mathcal{AB},\mathcal{AB}} + AB_{\mathcal{AB},\mathcal{AB}}$	$\bullet + AA_{\mathcal{AB},\mathcal{A}}$		$1-2w$	
	$AA_{\mathcal{AB},\mathcal{AB}} + AB_{\mathcal{B},\mathcal{AB},\mathcal{A}}$	$\bullet + AA_{\mathcal{B},\mathcal{AB}}$		$1-2w$	
	$BB_{\mathcal{AB},\mathcal{AB}} + AB_{\mathcal{AB},\mathcal{AB}}$	$\bullet + BB_{\mathcal{B},\mathcal{AB}}$		$1-2w$	
	$BB_{\mathcal{AB},\mathcal{AB}} + AB_{\mathcal{B},\mathcal{AB},\mathcal{A}}$	$\bullet + BB_{\mathcal{AB},\mathcal{A}}$		$1-2w$	
	$AB_{\mathcal{AB},\mathcal{AB}} + AB_{\mathcal{B},\mathcal{AB},\mathcal{A}}$	$\bullet + \bullet$		$1-2w$	

Table 4 All recombination groups that determine the deductions for computing the DCJ-substitution distance *(continued)*

$\mathcal{Y}$	$2AB_{\mathcal{AB}\mathcal{AB}} + AA_{\mathcal{B}} + BB_{\mathcal{A}}$	4 •	$2n_{-1} \prec o_{-2}$	$2 - 4w$	$w \geq \frac{1}{2}$
	$2AB_{\mathcal{BAB}\mathcal{A}} + AA_{\mathcal{A}} + BB_{\mathcal{B}}$	4 •		$2 - 4w$	
$\mathcal{Z}_1$	$AB_{\mathcal{AB}\mathcal{AB}} + AA_{\mathcal{AB}} + BB_{\mathcal{ABA}}$	3 •	$n_{-1} \prec o_{-1}$	$1 - 2w$	$w \geq \frac{1}{2}$
	$AB_{\mathcal{BAB}\mathcal{A}} + AA_{\mathcal{AB}} + BB_{\mathcal{BAB}}$	3 •		$1 - 2w$	
	$AB_{\mathcal{BAB}\mathcal{A}} + AA_{\mathcal{ABA}} + BB_{\mathcal{AB}}$	3 •		$1 - 2w$	
	$AB_{\mathcal{AB}\mathcal{AB}} + AA_{\mathcal{BAB}} + BB_{\mathcal{AB}}$	3 •		$1 - 2w$	
	$AB_{\mathcal{AB}\mathcal{AB}} + AA_{\mathcal{B}} + BB_{\mathcal{ABA}}$	3 •		$1 - 2w$	
	$AB_{\mathcal{AB}\mathcal{AB}} + AA_{\mathcal{BAB}} + BB_{\mathcal{A}}$	3 •		$1 - 2w$	
	$AB_{\mathcal{BAB}\mathcal{A}} + AA_{\mathcal{A}} + BB_{\mathcal{BAB}}$	3 •		$1 - 2w$	
	$AB_{\mathcal{BAB}\mathcal{A}} + AA_{\mathcal{ABA}} + BB_{\mathcal{B}}$	3 •		$1 - 2w$	
	$AB_{\mathcal{AB}\mathcal{AB}} + AA_{\mathcal{B}} + BB_{\mathcal{A}}$	2 • $+AB_{\mathcal{BAB}\mathcal{A}}$		$1 - 2w$	
	$AB_{\mathcal{BAB}\mathcal{A}} + AA_{\mathcal{A}} + BB_{\mathcal{B}}$	2 • $+AB_{\mathcal{AB}\mathcal{AB}}$		$1 - 2w$	
$\mathcal{Z}_2$	$2AB_{\mathcal{AB}\mathcal{AB}} + AA_{\mathcal{B}}$	2 • $+AA_{\mathcal{ABA}}$	$n_{-1} \prec n_{-2}$	$2 - 3w$	$w \geq \frac{2}{3}$
	$2AB_{\mathcal{AB}\mathcal{AB}} + BB_{\mathcal{A}}$	2 • $+BB_{\mathcal{BAB}}$		$2 - 3w$	
	$2AB_{\mathcal{BAB}\mathcal{A}} + AA_{\mathcal{A}}$	2 • $+AA_{\mathcal{BAB}}$		$2 - 3w$	
	$2AB_{\mathcal{BAB}\mathcal{A}} + BB_{\mathcal{B}}$	2 • $+BB_{\mathcal{ABA}}$		$2 - 3w$	

depends only on w and the initial components. The runs are obtained by a single walk through each path, thus the whole procedure takes linear time for both models.

Establishing the triangular inequality

We have presented two genomic distances that combine DCJ and content-modifying operations and can be computed in linear time. However, content-modifying operations are applied to pieces of DNA of any size, and a side effect of this fact is that the triangular inequality often does not hold for distances that consider these operations [4,6-8,12].

Let A, B and C be three genomes, with unequal contents, and consider, without loss of generality, that $d_{DCJ}^{id}(A,B) \geq d_{DCJ}^{id}(A,C)$ and $d_{DCJ}^{id}(A,B) \geq d_{DCJ}^{id}(B,C)$. The triangular inequality is then the property which guarantees that the inequality $d_{DCJ}^{id}(A,B) \leq d_{DCJ}^{id}(A,C) + d_{DCJ}^{id}(B,C)$ also holds. Unfortunately this is not the case for the DCJ-indel distance, and also not the case for the DCJ-substitution distance. Take for example the genomes $A = \{\circ abcde\circ\}$, $B = \{\circ ac\overline{d}be\circ\}$ and $C = \{\circ ae\circ\}$ [6]. While the cost of sorting A (or B) into C is w (one indel), the minimum number of DCJs (that are inversions in this case) required to sort A into B is three. We have $d_{DCJ}^{id}(A,B) = 3$, $d_{DCJ}^{id}(A,C) = w$, $d_{DCJ}^{id}(B,C) = w$ and the triangular inequality is disrupted.

Denote by $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$, $\mathcal{D}$, $\mathcal{E}$, $\mathcal{F}$ and $\mathcal{G}$ the disjoint sets of markers such that: $\mathcal{A}$, $\mathcal{B}$ or $\mathcal{C}$ are the sets of markers that occur respectively only in genome A, B or C, the markers in $\mathcal{D}$ are common only to genomes A and B, the markers in $\mathcal{E}$ are common only to B and C, the markers in $\mathcal{F}$ are common only to A and C, and, $\mathcal{G}$ is the set of markers that are common to all three genomes A, B and C. These sets are represented in Figure 7.

When $\mathcal{D} = \emptyset$, meaning that genomes A and B have no common marker that does not occur in C, the triangular inequality holds for both DCJ-indel and DCJ-substitution

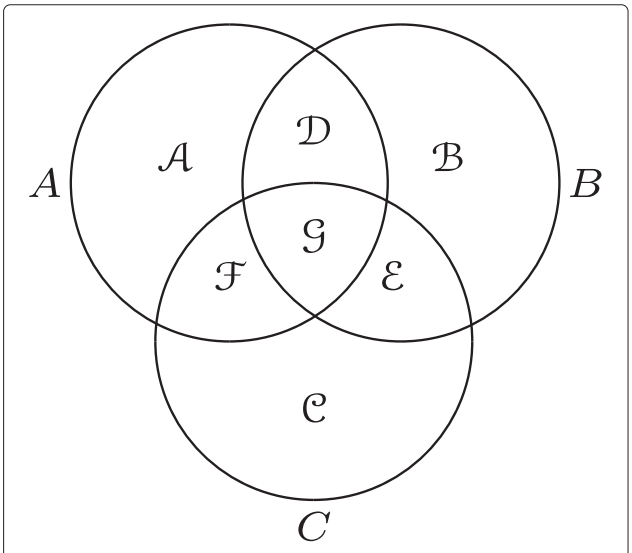

Figure 7 The set of markers of each genome is represented by a circle.

distances [12]. However, if $\mathcal{D} \neq \emptyset$, the triangular inequality can be disrupted for d^{id}_{DCJ} and d^{sb}_{DCJ}, and this may be an obstacle if one intends to use these distances to compute the median of three or more genomes and in phylogenetic reconstructions.

It is possible to establish the triangular inequality in our two models *a posteriori*, by adapting an approach proposed in [12]: we simply sum to each distance a surcharge that depends on the number of unique markers, as we will see in the following subsections.

We define the diameter as the maximum distance between any pair of genomes, usually as a function on the size of the genomes. We use this definition in the next results.

Correction for the DCJ-indel distance

For genomes A and B and a positive constant k, let $m^{id}(A,B) = d^{id}_{DCJ}(A,B) + k \cdot u(A,B)$, where $u(A,B)$ is the number of unique markers between A and B [7,12]. We then have $m^{id}(A,B) = d^{id}_{DCJ}(A,B)+k(|\mathcal{A}|+|\mathcal{F}|+|\mathcal{B}|+|\mathcal{E}|)$, $m^{id}(A,C) = d^{id}_{DCJ}(A,C) + k(|\mathcal{A}| + |\mathcal{D}| + |\mathcal{C}| + |\mathcal{E}|)$ and $m^{id}(B,C) = d^{id}_{DCJ}(B,C) + k(|\mathcal{B}| + |\mathcal{D}| + |\mathcal{C}| + |\mathcal{F}|)$. From this definition we can derive a simpler inequality that can be used to determine the value of the constant k:

Proposition 5 (from [12])**.** *Given three genomes A, B and C without duplicated markers, the inequality $m^{id}(A,B) \leq m^{id}(A,C) + m^{id}(B,C)$ holds if, and only if, $d^{id}_{DCJ}(A,B) \leq d^{id}_{DCJ}(A,C) + d^{id}_{DCJ}(B,C) + 2k|\mathcal{D}|$, where $\mathcal{D}$ is the set of markers common only to A and B.*

The problem now is to find the minimum value of k for which the inequality of Proposition 5 holds. In order to accomplish this task, the first step is to determine the diameter of the DCJ-indel distance.

Lemma 4. *Given a positive indel cost $w \leq 1$ and two genomes A and B with n common markers, then*

$$d^{id}_{DCJ}(A,B) \leq (w+1)n + w(L_A + S_A + L_B + S_B),$$

where L_A, S_A and L_B, S_B are, respectively, the number of linear chromosomes and circular singletons in genomes A and B.

Proof. Let $|P|$ be the number of vertices in component P, that is DCJ-sorted with $\lfloor \frac{|P|-1}{2} \rfloor$ DCJs [14]. If $|P|$ is even, P is sorted with $\frac{|P|}{2} - 1$ DCJs and $\lambda(P) \leq \frac{|P|}{2} + 1$ indels, then $d^{id}_{DCJ}(P) \leq \frac{|P|}{2} - 1 + w(\frac{|P|}{2} + 1) = \frac{(w+1)|P|}{2} + w - 1$. If $|P|$ is odd, P is sorted with $\frac{|P|-1}{2}$ DCJs and $\lambda(P) \leq \frac{|P|+1}{2}$ indels, then $d^{id}_{DCJ}(P) \leq \frac{|P|-1}{2} + w\frac{|P|+1}{2} = \frac{(w+1)|P|+w-1}{2}$. As $w \leq 1$ implies $w - 1 \leq \frac{w-1}{2} \leq 0$, for any component P we have $d^{id}_{DCJ}(P) \leq \frac{(w+1)|P|+w-1}{2}$. Then, $d^{id}_{DCJ}(A,B) \leq \sum_{P \in AG(A,B)} d^{id}_{DCJ}(P) \leq \sum_{P \in AG(A,B)} \frac{(w+1)|P|+w-1}{2} = \frac{w+1}{2} \sum_{P \in AG(A,B)} |P| + \sum_{P \in AG(A,B)} \frac{w-1}{2}$. Each linear chromosome corresponds to one path in $AG(A,B)$, thus the number of components is at least $(L_A + S_A + L_B + S_B)$ and $\sum_{P \in AG(A,B)} \frac{w-1}{2} \leq \frac{(L_A+S_A+L_B+S_B)(w-1)}{2} \leq 0$. Furthermore, from [12] we know that $\sum_{P \in AG(A,B)} |P| = 2n + L_A + S_A + L_B + S_B$. □

We are ready to generalize the result of [12], and determine the minimum possible value of k.

Theorem 4. *For any positive indel cost $w \leq 1$, the function m^{id} satisfies the triangular inequality if and only if $k \geq \frac{w+1}{2}$.*

Proof. Recall that, to prove the triangular inequality for m^{id}, we only need to find a k such that $d^{id}_{DCJ}(A,B) \leq d^{id}_{DCJ}(A,C) + d^{id}_{DCJ}(B,C) + 2k|\mathcal{D}|$ holds (Proposition 5). We know that the inequality holds when $\mathcal{D} = \emptyset$ [12]. It remains to examine the case in which $\mathcal{D} \neq \emptyset$. The worst case would be to have an empty genome C [12]. Let X_A and X_B be the number of chromosomes in A and B. Since C is empty, we know that $d^{id}_{DCJ}(A,C) = wX_A$ and $d^{id}_{DCJ}(B,C) = wX_B$. From Lemma 4, we have $d^{id}_{DCJ}(A,B) \leq (w+1)|\mathcal{D}|+w(L_A+S_A+L_B+S_B)$. This gives $(w+1)|\mathcal{D}|+w(L_A+S_A+L_B+S_B) \leq w(X_A+X_B)+2k|\mathcal{D}|$. Since $L_A+S_A+L_B+S_B \leq X_A+X_B$, we have $(w+1)|\mathcal{D}| \leq 2k|\mathcal{D}|$, which holds for any $k \geq \frac{w+1}{2}$.

For the necessity, take A and B with n common markers and let each genome be composed of one circular chromosome, meaning that we have one adjacency per common marker in each genome (or n adjacencies per genome). Then let $AG(A,B)$ have one single cycle with $2n$ vertices and let each vertex be labeled, so that the number of runs in the cycle is $2n$ and the number of unique markers in each genome is n. Thus, we have $d^{id}_{DCJ}(A,B) = (n-1)+w(n+1) = (w+1)n+(w-1)$ and the corrected distance is $m^{id}(A,B) = (w+1)n+(w-1)+2kn$. Take C as an empty genome, so that $d^{id}_{DCJ}(A,C) = d^{id}_{DCJ}(B,C) = w$ and $m^{id}(A,C) = m^{id}(B,C) = w + 2kn$. The inequality $m^{id}(A,B) \leq m^{id}(A,C)+m^{id}(B,C)$ corresponds to $(w+1)n+(w-1)+2kn \leq 2w+4kn$ or, equivalently, $2kn \geq (w+1)n-w-1$, that is $k \geq \frac{w+1}{2}\left(1 - \frac{1}{n}\right)$, which holds for all n only if $k \geq \frac{w+1}{2}$. □

Correction for the DCJ-substitution distance

Similarly, in the case of the DCJ-substitution distance, for genomes A and B and a positive constant k', let $m^{sb}(A,B) = d^{sb}_{DCJ}(A,B) + k' \cdot u(A,B)$, where $u(A,B)$ is the number of unique markers between A and B [7,12]. We then have $m^{sb}(A,B) = d^{sb}_{DCJ}(A,B)+k'(|\mathcal{A}|+|\mathcal{F}|+|\mathcal{B}|+|\mathcal{E}|)$, $m^{sb}(A,C) = d^{sb}_{DCJ}(A,C) + k'(|\mathcal{A}| + |\mathcal{D}| + |\mathcal{C}| + |\mathcal{E}|)$ and

$m^{sb}(B,C) = d^{sb}_{DCJ}(B,C)+k'(|\mathcal{B}|+|\mathcal{D}|+|\mathcal{C}|+|\mathcal{F}|)$. Again, from this definition we can derive a simpler inequality that can be used to determine the value of the constant k':

Proposition 6 (from [12]). *Given three genomes A, B and C without duplicated markers, the inequality $m^{sb}(A,B) \leq m^{sb}(A,C) + m^{sb}(B,C)$ holds if, and only if, $d^{sb}_{DCJ}(A,B) \leq d^{sb}_{DCJ}(A,C) + d^{sb}_{DCJ}(B,C) + 2k'|\mathcal{D}|$, where $\mathcal{D}$ is the set of markers common only to A and B.*

In order to find the minimum value of k' for which the inequality of Proposition 6 holds, we need to determine the diameter of the DCJ-substitution distance, that is given by the following lemma.

Lemma 5. *If A and B are genomes with n common markers, then*

$$d^{sb}_{DCJ}(A,B) \leq \frac{(w+2)}{2}n + w(L_A + S_A + L_B + S_B),$$

where L_A, S_A, L_B and S_B are, respectively, the number of linear chromosomes and circular singletons in genomes A and B.

Proof. Let $|P|$ be the number of vertices in component P, that is DCJ-sorted with $\lfloor \frac{|P|-1}{2} \rfloor$ DCJs [14]. If $|P|$ is even, then P can be DCJ-sorted with $\frac{|P|}{2} - 1$ DCJs. We have to analyze two cases: (i) if $|P| = 4x + 4$, then $\sigma(P) \leq \frac{|P|}{4} + 1$ and $d^{sb}_{DCJ}(P) \leq (\frac{|P|}{2} - 1) + w(\frac{|P|}{4} + 1) = \frac{(w+2)|P|}{4} + w - 1$; (ii) if $|P| = 4x + 2$, then $\sigma(P) \leq \frac{|P|-2}{4} + 1$ and $d^{sb}_{DCJ}(P) \leq (\frac{|P|}{2} - 1) + w(\frac{|P|-2}{4} + 1) = \frac{(w+2)|P|}{4} + \frac{w-2}{2}$. As $w \leq 1$ implies $\frac{w-2}{2} \leq w - 1 \leq 0$. If $|P|$ is odd, then P is an AA- or a BB-path and can be DCJ-sorted with $\frac{|P|-1}{2}$ DCJs. Again, we have to analyze two cases: (i) if $|P| = 4x + 3$, then $\sigma(P) \leq \frac{|P|+1}{4}$ and $d^{sb}_{DCJ}(P) \leq \frac{|P|-1}{2} + w(\frac{|P|+1}{4}) = \frac{(w+2)|P|}{4} + \frac{w-2}{4}$; (ii) if $|P| = 4x + 1$, then $\sigma(P) \leq \frac{|P|+3}{4}$ and $d^{sb}_{DCJ}(P) \leq \frac{|P|-1}{2} + w(\frac{|P|+3}{4}) = \frac{(w+2)|P|}{4} + \frac{3w-2}{4}$. In this last case we could have $d^{sb}_{DCJ}(P) > \frac{(w+2)|P|}{4}$. Observe however that the numbers of AA- and BB-paths are bounded, respectively, by L_A and L_B. Then, $d^{sb}_{DCJ}(A,B) \leq \sum_{P \in AG(A,B)} d^{sb}_{DCJ}(P) \leq \sum_{P \in AG(A,B)} \frac{(w+2)|P|}{4} + \frac{(3w-2)(L_A+L_B)}{4} = \frac{w+2}{4} \sum_{P \in AG(A,B)} |P| + \frac{(3w-2)(L_A+L_B)}{4}$. From [12] we know that $\sum_{P \in AG(A,B)} |P| = 2n+L_A+S_A+L_B+S_B$. Therefore, $d^{sb}_{DCJ}(A,B) \leq \frac{w+2}{4}(2n+L_A+S_A+L_B+S_B) + \frac{(3w-2)(L_A+L_B)}{4} = \frac{(w+2)}{2}n + w(L_A + L_B) + \frac{(w+2)(S_A+S_B)}{4} \leq \frac{(w+2)}{2}n + w(L_A + L_B + S_A + S_B)$. □

We are ready to generalize the result of [12], and determine the minimum possible value of k'.

Theorem 5. *For any positive substitution cost $w \leq 1$, the function m^{sb} satisfies the triangular inequality if and only if $k' \geq \frac{w+2}{4}$.*

Proof. Recall that, to prove the triangular inequality for m^{sb}, we only need to find a k' such that $d^{sb}_{DCJ}(A,B) \leq d^{sb}_{DCJ}(A,C) + d^{sb}_{DCJ}(B,C) + 2k'|\mathcal{D}|$ holds (Proposition 6). We know that the inequality holds when $\mathcal{D} = \emptyset$ [12]. It remains to examine the case in which $\mathcal{D} \neq \emptyset$. The worst case would be to have an empty genome C [12]. Let X_A and X_B be the number of chromosomes in A and B. Since C is empty, we know that $d^{sb}_{DCJ}(A,C) = wX_A$ and $d^{sb}_{DCJ}(B,C) = wX_B$. From Lemma 5, we have $d^{sb}_{DCJ}(A,B) \leq \frac{(w+2)|\mathcal{D}|}{2} + w(L_A+S_A+L_B+S_B)$. This gives $\frac{(w+2)|\mathcal{D}|}{2}+w(L_A+S_A+L_B+S_B) \leq w(X_A+X_B)+2k'|\mathcal{D}|$. Since $L_A+S_A+L_B+S_B \leq X_A+X_B$, we have $\frac{(w+2)|\mathcal{D}|}{2} \leq 2k'|\mathcal{D}|$, which holds for any $k' \geq \frac{w+2}{4}$.

For the necessity, take A and B with n common markers, for n even, and let each genome be composed of one circular chromosome, meaning that we have one adjacency per common marker in each genome (or n adjacencies per genome). Then let $AG(A,B)$ have one single cycle with $2n$ vertices and let each vertex be labeled, so that the number of runs in the cycle is $2n$ and the number of unique markers in each genome is n. Thus, we have $d^{sb}_{DCJ}(A,B) = (n-1)+w(\frac{n}{2}+1) = \frac{(w+2)n}{2} + (w-1)$ and the corrected distance is $m^{sb}(A,B) = \frac{(w+2)n}{2} + (w-1) + 2k'n$. Take C as an empty genome, so that $d^{sb}_{DCJ}(A,C) = d^{sb}_{DCJ}(B,C) = w$ and $m^{sb}(A,C) = m^{sb}(B,C) = w + 2k'n$. The inequality $m^{sb}(A,B) \leq m^{sb}(A,C)+m^{sb}(B,C)$ corresponds to $\frac{(w+2)n}{2} + (w-1)+2k'n \leq 2w+4k'n$ or, equivalently, $2k'n \geq (\frac{w+2}{2})n - w - 1$, that is $k' \geq \frac{w+2}{4} - \frac{w+1}{2n}$, which holds for all n only if $k' \geq \frac{w+2}{4}$. □

Conclusions

In this work we have presented methods to compute in linear time the DCJ-indel and DCJ-substitution distances between two genomes without duplicated markers, when the content-modifying cost is distinct from and upper bounded by the DCJ cost. Content-modifying operations can be applied to pieces of DNA of any size, and a side effect of this property is that the triangular inequality does not hold for our distance formulas. However we have shown that an *a posteriori* correction can be applied to establish the triangular inequality in both DCJ-indel and DCJ-substitution distances.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

PHS, RM, SD and MDVB have elaborated the model, proved the results and written the paper. All authors read and approved the final manuscript.

Acknowledgements
This research was partially supported by the Brazilian research agencies CNPq and FAPERJ.

Author details
[1]IME, Universidade Federal Fluminense, Niterói, Brazil. [2]Inmetro – Instituto Nacional de Metrologia, Qualidade e Tecnologia, Duque de Caxias, Brazil.

Received: 21 December 2012 Accepted: 29 June 2013
Published: 23 July 2013

References

1. Hannenhalli S, Pevzner P: **Transforming men into mice (polynomial algorithm for genomic distance problem).** In *36th Annual IEEE Symposium on Foundations of Computer Science.* 1995:581–592.
2. Yancopoulos S, Attie O, Friedberg R: **Efficient sorting of genomic permutations by translocation, inversion and block interchange.** *Bioinformatics* 2005, **21:**3340–3346.
3. Bergeron A, Mixtacki J, Stoye J: **A unifying view of genome rearrangements.** In *Algorithms in Bioinformatics, Lecture Notes in Computer Science, Volume 4175.* Springer; 2006:163–173.
4. El-Mabrouk N: **Sorting signed permutations by reversals and insertions/deletions of contiguous segments.** *J Discrete Algorithms* 2001, **1:**105–122.
5. Hannenhalli S, Pevzner P: **Transforming cabbage into turnip: polynomial algorithm for sorting signed permutations by reversals.** *J ACM* 1999, **46:**1–27.
6. Yancopoulos S, Friedberg R: **DCJ path formulation for genome transformations which include insertions, deletions, and duplications.** *J Comput Biol* 2009, **16**(10):1311–1338.
7. Braga MDV, Willing E, Stoye J: **Double Cut and Join with Insertions and Deletions.** *J Comput Biol* 2011, **18**(9):1167–1184. (a preliminary version appeared in proceedings of WABI 2010, LNBI vol. 6293, p. 90–101).
8. Braga MDV, Machado R, Ribeiro LC, Stoye J: **Genomic distance under gene substitutions.** *BMC Bioinformatics* 2011, **12**(Suppl 9):S8.
9. Boore JL: **The duplication/random loss model for gene rearrangement exemplified by mitochondrial genomes of deuterostome animals.** In *Comparative Genomics, Volume 1.* Springer; 2000:133–147.
10. Moritz C, Dowling TE, Brown WM: **Evolution of animal mitochondrial DNA: relevance for population biology and systematics.** In *Annual Review of Ecology and Systematics, Volume 18.* Annual Reviews Inc; 1987:269–292.
11. Blanc G, Ogata H, Robert C, et al: **Reductive genome evolution from the mother of Rickettsia.** *PLoS Genet* 2007, **3:**e14.
12. Braga MDV, Machado R, Ribeiro LC, Stoye J: **On the weight of indels in genomic distances.** *BMC Bioinformatics* 2011, **12**(Suppl 9):S13.
13. da Silva PH, Braga MDV, Machado R, Dantas S: **DCJ-indel distance with distinct operation costs.** In *Algorithms in Bioinformatics, Lecture Notes in Computer Science, Volume 7534.* Springer; 2012:378–390.
14. Braga MDV, Stoye J: **The solution space of sorting by DCJ.** *J Comput Biol* 2010, **17**(9):1145–1165.

doi:10.1186/1748-7188-8-21
Cite this article as: da Silva *et al.*: DCJ-indel and DCJ-substitution distances with distinct operation costs. *Algorithms for Molecular Biology* 2013 **8**:21.