The genesis of the DCJ formula

Anne Bergeron[1] and Jens Stoye[2]

[1] Lacim, Université du Québec à Montréal, Montréal, Canada
[2] Technische Fakultät, Universität Bielefeld, Bielefeld, Germany

Abstract. The formula $N-(C+I/2)$ to compute the number of Double-Cut-and-Join operations needed to transform one genome into another is both simple and easy to prove. When it was published, in 2006, we omitted all details on how it was constructed. In this chapter, we will give an elementary treatment on the intuitions and methods underlying the formula, showing that simplicity is sometimes difficult to achieve. We will also prove that this formula is one among an infinite number of candidates, and that the techniques can be applied to other genomic distances.

1 Introduction

In May 2005, the authors attended the Recomb meeting in Boston, Mass. They had an accepted paper on a tamed variant of the genome rearrangement problem. Happily for them, the presenter was a young graduate student, Julia Mixtacki, and the authors had plenty of lounging time. On the sunny terraces, cafés and salons of the MIT campus, David Sankoff managed to introduce us to Sophia Yancopoulos, who had an original, thrilling, radical, but very informal view on genome rearrangements, presented on a poster at that conference. Her paper [10], written with O. Attie and R. Friedberg, appeared in the same month in Bioinformatics, but was a bit of a challenge to read.

The authors' team, including Julia who would play a determinant role in the sequel, felt that there should exist a more formal way of computing this distance. The road was bumpy. We first had to understand the real power of the Double-Cut-and-Join (DCJ) operation introduced by Yancopoulos et al. The original paper focused on the usefulness of the new concepts to explain known results, rather than exploring the consequences of the new definition. It was necessary to forget, for the time being, the results of the former decade that explored rearrangement operations on linear genomes.

Immediately after Recomb, the authors and Julia spent three days together in Montréal and started trying to understand and formalize the ideas they had been introduced to. However, we completely failed, as we were too closely following the Yancopoulos 'recipe', instead of starting from scratch and re-phrase the DCJ model in our own language. Therefore, it was possibly a good idea to wait for eight months before continuing, so we could leave behind most of that early attempt.

The first breakthrough occurred in February 2006 in Lisbon, Portugal, when AB was giving a series of five lectures to graduate students at the Instituto Gulbenkian de Ciencia. The lectures were given in the morning, and the lecturer had the afternoons to herself to pursue her own research. Julia came from Bielefeld to Lisbon for a week. This move resulted in the definition of the adjacency graph, and in a deep grasp of the DCJ operations. While the shift of our underlying data structure, from the breakpoint graph to the adjacency graph, was rather formal, the understanding of the nature of the DCJ operations relied on a toy genome, made of black and white electrical chords, together with male and female connections that stood for double-stranded DNA, gene orientation, breaks and repairs. At one point, the two researchers 'executed' all the variants of a DCJ operation, using their four hands and the model genome. They were so concentrated on the 'proof' that it took them a certain time to realize that maintenance people were peering at them through the door's window. These results are described in Sections 2 and 3.

The next, and crucial, development happened in Montréal, in Spring 2006, when JS came to visit as part of his sabbatical. At that point, the problem was not to develop one formula, but to cope with too many formulas! One of the frustrated authors decided, on one evening, to rely on a dirty mathematical trick, discussed in Section 4, to come up with the 'simplest' formula among those candidates. The result was suddenly quite simple, and the proof of the DCJ distance became elementary, which is discussed in Section 5.

In fact, the DCJ model is only one out of many where the same technique can be applied to quickly derive general and simple distance formulas, as we will show in Section 6 for the algebraic, the breakpoint and two single-cut distances.

2 Rearrangement operations and the adjacency graph

Here we will briefly recall the notation we use to represent and manipulate genomes. While it may nowadays seem natural and many other authors have adopted the terminology, in 2005-06 we spent probably more time on the development of this notation than on the derivation and proving of the DCJ distance formula and sorting algorithm.

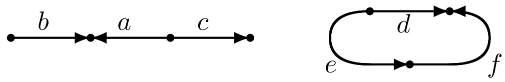
A *gene* is a piece of DNA with two extremities, its *head* and its *tail.* For a gene a we denote its head by a^h and its tail by a^t. A *genome* for a given set of genes G is a set of *adjacencies*, consisting of pairs of gene extremities, where each extremity of each gene in G is contained in exactly one adjacency. One of the two gene extremities in an adjacency can be replaced by the *telomere marker* $\circ$, indicating the end of a linear chromosome. Such an adjacency is called a *telomere.*

Example 1. Consider the gene set $G = \{a, b, c, d, e, f\}$. Then the following set A is a genome for G:

$$A \;=\; \{\; \{\circ, b^t\}, \{b^h, a^h\}, \{a^t, c^t\}, \{c^h, \circ\}, \{e^t, d^t\}, \{d^h, f^h\}, \{f^t, e^h\} \;\}$$

A genome can be represented as a graph, called the *genome graph*, whose vertices are the adjacencies and whose edges connect for each gene the adjacency containing its head with the adjacency containing its tail. Clearly, each vertex of the genome graph has degree one or two, and therefore the connected components are either *cycles*, representing circular chromosomes, or *paths*, representing linear chromosomes.

Example 1 (cont'd). The genome graph of A looks as follows:



It is easy to see that A has two connected components, one of which is linear and the other one is circular.

We will also use a notation to represent the genome graph, in which a linear chromosome is written as the sequence of its genes from one of its telomeres to the other, where a gene is indicated by its name when it is read in tail-head direction, and by its overlined name when it is read in head-tail direction. A circular chromosome is represented similarly but, as it has no ends, spelling the genes can start anywhere, in any of the two possible directions, and all these representations are equivalent.

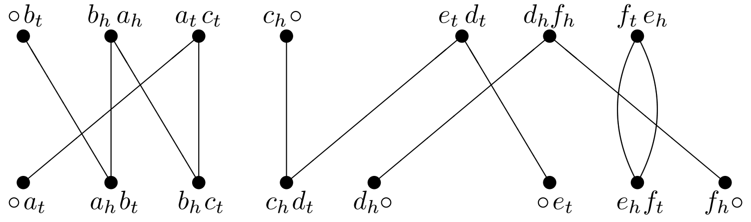
Example 1 (cont'd). In the linear notation, our genome looks as follows:

$$A = \{ \; (\circ \; b \; \overline{a} \; c \; \circ) \; \; (d \; \overline{f} \; \overline{e}) \; \}$$

Note that our genome model is very general in the sense that a genome can be a mix of circular and linear chromosomes. Other models have been considered, restricting genomes to contain only linear or only circular chromosomes. These constraints can be added at any time to the general model in order to reflect biological reality. However, as we will see in Section 3, rearrangement operations are independent of chromosome structure.

Another graph that will be very useful in the sequel is the *adjacency graph* for two genomes A and B containing the same genes. It will be the essential tool when calculating their rearrangement distance. The vertices of the adjacency graph are the adjacencies of the two genomes, and for each extremity of a gene from G we have an edge, connecting the two adjacencies (one from A and one from B) in which it is contained. Note that all vertices of the adjacency graph also have degree one or two, thus its connected components are again paths or cycles. However, because the graph is bipartite, all cycles have even length.

Example 1 (cont'd). For A as above and genome $B = \{ \; (\circ \; a \; b \; c \; d \; \circ) \; (\circ \; e \; f \; \circ) \; \}$, we have the following adjacency graph:



In the sequel we will consider various models of genome comparison, most of which realize some kind of edit distance, which in general can be phrased as follows.

Definition 1 (Genomic distance problem). *Given two genomes A and B and a set of operations to manipulate them, what is the minimum number of operations to transform A into B?*

If a corresponding sequence of operations realizing this number is actually reported, this is called the *genomic sorting problem*, but we will not discuss it further in this chapter.

3 An Illustrated Guide to the Double-Cut-and-Join Operation

In order to understand rearrangement operations it is necessary to have an idea of the mechanisms underlying them. When a double-stranded DNA sequence is broken, the cell is usually able to repair the damage by joining the two hanging ends together. However, as one wikipedian wrote in 2007 under the pseudonym *Amazinglarry* [9]:

> "*Double-strand breaks, in which both strands in the double helix are severed, are particularly hazardous to the cell because they can lead to genome rearrangements.*"

Indeed, when a genome breaks at two positions that are physically close, creating four hanging ends of double-strand DNA, the repair mechanisms may join the alternative ends together. This yields a deceptively simple definition of the Double-Cut-and-Join (DCJ) operation as: *the genome is cut in two places, and the pieces are joined in a different way.* This definition is correct, but it should be treated with the care deserved to informal definitions: the oriented nature of a double-stranded DNA sequence, and the fact that pieces may be lost or misplaced, introduce subtle constraints that need to be formalized.

The basic DCJ operation. Let's begin with Figure 1. The top genome is a circular chromosome broken at two physically close positions. Two genes are

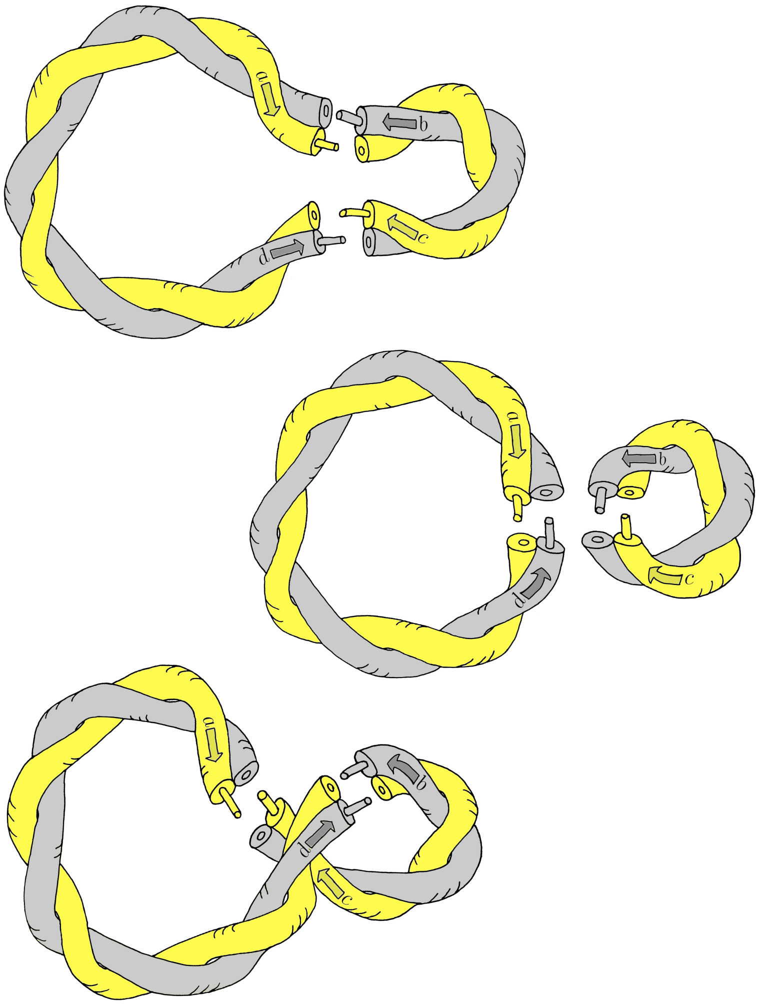


Fig. 1. The top drawing represents a genome with one double stranded circular chromosome and 4 genes, a, b, c, and d. Genes are represented by arrows, and genes on opposite strands have opposite orientation. This genome can be represented as $(a\ \overline{b}\ c\ \overline{d})$. Suppose the chromosome is broken in two places, as illustrated, between genes a and b, and between genes c and d. The strands may be repaired in three different ways: the original arrangement $(a\ \overline{b}\ c\ \overline{d})$, the middle genome $(a\ \overline{d})\ (\overline{b}\ c)$ which is a *fission* of the top chromosome, and the bottom one $(a\ \overline{c}\ b\ \overline{d})$ that contains an *inversion* with respect to the original arrangement.

marked on each strand, a strand with genes a and c, and, in the opposite direction, a strand with genes b and d. Starting from gene a, and going around the chromosome, the gene organization of this chromosome can be represented as

$$(a\ \overline{b}\ c\ \overline{d}).$$

The position of a break in the double-strand is described by the severed adjacency, which we say to be *cut*. In Figure 1, the two adjacencies of the top genome:

$$\{a^h, b^h\} \text{ and } \{c^h, d^h\}$$

are cut. If those two breaks are sufficiently close, the repair mechanisms, which have a very restricted understanding of the global situation, may *join* the a^h extremity with any of b^h, c^h, or d^h, whichever comes handy. The two remaining extremities are joined together, provided no pieces are lost. Thus there are three possible results of the repair, illustrated in Figure 1:

1) The original configuration, when b^h is chosen. The shape and gene order of the original genome are restored.

2) An alternative configuration, when d^h is chosen. The circular chromosome is split in two circular chromosomes, in an event called a *fission*. The corresponding genome can be represented by $(a\ \overline{d})\ (\overline{b}\ c)$. The reverse event is called a *fusion*.

3) An alternative configuration, when c^h is chosen. The chromosome is still circular, but the original strands are mixed: genes a and b are now on the same strand, opposite to genes c and d. This event is called an *inversion*. The new chromosome can be represented as $(a\ \overline{c}\ b\ \overline{d})$. Note the change in order and orientation of genes c and b with respect to the original genome.

This is it! The essence of the DCJ operation is contained in this example. The vast majority of identified rearrangement operations are based on this series of events, and the variations in terminology usually come from factors that are not directly related to the rearrangement operation itself.

A DCJ within a single linear chromosome. Figure 2 is a reproduction of Figure 1 in which the circular genome has been transformed into a linear genome by replacing a small segment of the double-stranded DNA with two telomeres. This modification has been done far from the breaks, and the rest of the picture is exactly the same. The genome organization would now be represented as

$$(\circ\ a\ \overline{b}\ c\ \overline{d}\ \circ),$$

to account for the new shape of the chromosome.

As in the circular case, the rearrangment operation between the top and bottom chromosome is called an *inversion*: the original strands are mixed, but the shape and gene content of the chromosome is the same. This type of rearrangements was first identified on fruit fly chromosomes, at the beginning of the last century [3], giving a founding example of rearranged genomes.

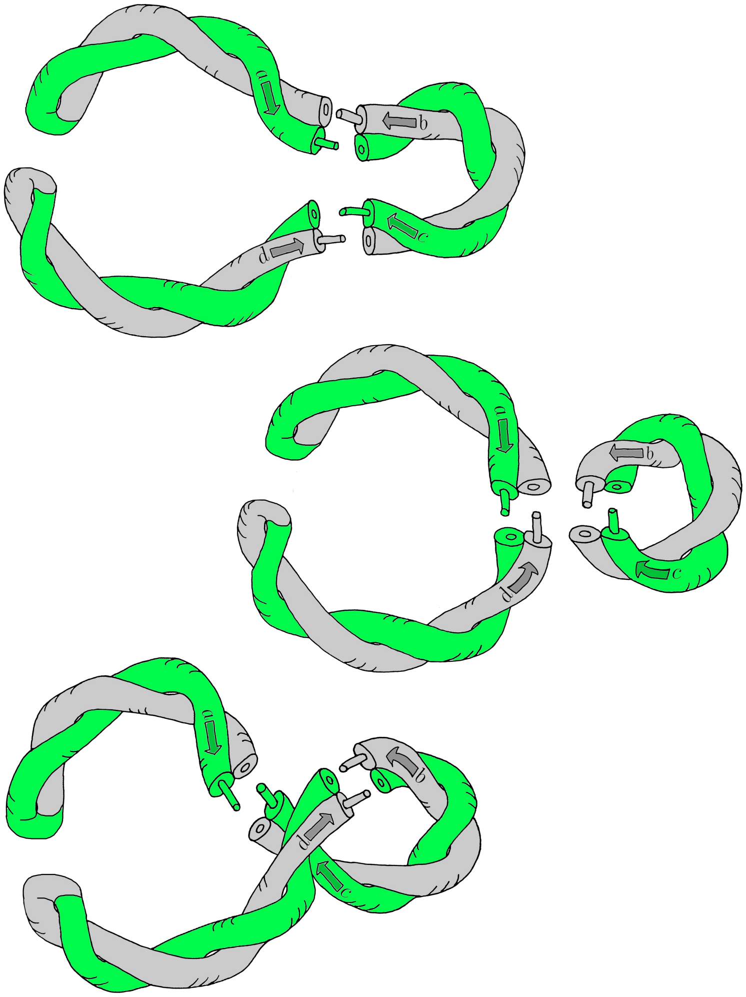


Fig. 2. In this figure, the top drawing represents a genome with one double stranded linear chromosome. It was obtained by a slight modification of the genome in Figure 1, consisting in removing a segment from the larger loop and capping the extremities with *telomeres*. This genome contains the same genes as the one in Figure 1, and is now represented as ($\circ\ a\ \overline{b}\ c\ \overline{d}\ \circ$) to account for the telomeres, but the breaks and repairs are exactly at the same positions. The bottom genome ($\circ\ a\ \overline{c}\ b\ \overline{d}\ \circ$) contains an inversion with respect to the original arrangement. The middle genome has two chromosomes, one linear and one circular: ($\circ\ a\ \overline{d}\ \circ$) ($\overline{b}\ c$).

The middle genome of Figure 2 is the only example in which linear and circular chromosomes are mixed. The DCJ operations that transform a linear chromosome into such a genome is called a *circular excision*, and its reverse, a *reincorporation.* Many models of genome evolution explicitly forbid this type of rearrangement, arguing that, for example, the transformation of a genome consisting of linear chromosomes into a similar genome should not involve circular chromosomes. This is a quite natural requirement, but there is a tradeoff in the complexity of deriving the distance formula [6].

A DCJ between two linear chromosomes. In Figure 3, telomeres are inserted in both ends of the circular genome of Figure 1, resulting in a genome consisting of two linear chromosomes, represented by:

$$(\circ\, a\, \overline{b}\, \circ)\ \ (\circ\, c\, \overline{d}\, \circ).$$

In this case, the DCJ operations are referred to as *reciprocal translocations.* Here again, the modifications have been done far from the breaks, and the rest of the picture is the same, showing that the basic mechanics of DCJ cover a vast range of rearrangement operations.

Reciprocal translocations change the sets of genes associated with a particular chromosome, but do not modify the number of chromosomes in a genome. This can be annoying, since there are examples of really close species, with virtually the same set of genes, that have different number of chromosomes. This is the case, for example, with the human and chimpanzee genome, the latter having an extra chromosome. The DCJ model can be extended to cover this possibility as explained in the next paragraph.

Single breaks and lost pieces. As we have seen, the vast majority of genome rearrangements are caused by double breaks, but sometimes single breaks lead to genome modifications. With a single break, the repair mechanism usually restores the DNA strand but, in some rare instances, the break is never repaired. If this event occurs in a linear chromosome, we model the operation as:

$$(\circ\, a\ \ b\, \circ) \longrightarrow (\circ\, a\, \circ)\, (\circ\, b\, \circ),$$

which is called a *fission.* When such an event occurs in a circular chromosome, it is called a *linearization*: the number of chromosomes is unchanged, but the genome is clearly modified. Despite involving only one break, these two operations are included in the DCJ model.

On the other hand, the reverse of these two operations, *fusion* of linear chromosomes and *circularization*, require two breaks and can be explained using the standard DCJ model and the loss of some hopefully redundant genetic material. Figure 3 contains many instances of fusions of chromosome segments belonging to different chromosomes: if all the necessary genetic information is contained in two fused segments, the remaining segments can be lost without consequences. As expected, we model this operation as the reverse of a fission:

$$(\circ\, a\, \circ)\, (\circ\, b\, \circ) \longrightarrow (\circ\, a\ \ b\, \circ),$$

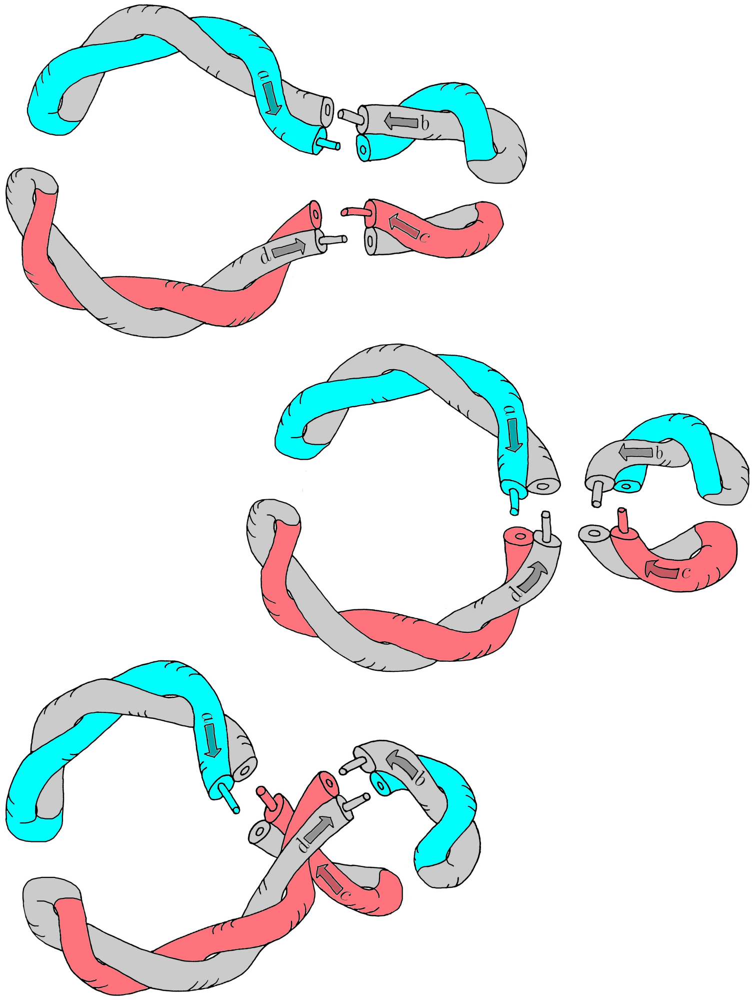


Fig. 3. This figure is another photoshopped version of Figure 1 which produced two linear chromosomes out of the original circular ones. The two breaks and the gene labels are untouched. In this case, the operation that transforms one genome into any of the two others is called a *reciprocal translocation*. The top genome is represented by $(\circ\ a\ \overline{b}\ \circ)$ $(\circ\ c\ \overline{d}\ \circ)$, the middle one by $(\circ\ a\ \overline{d}\ \circ)$ $(\circ\ c\ \overline{b}\ \circ)$, and the bottom one by $(\circ\ a\ \overline{c}\ \circ)$ $(\circ\ b\ \overline{d}\ \circ)$.

where the telomere markers '○' stand in for the lost material. The circularization of a segment of a linear chromosome is central in Figure 2: again, if all the necessary genetic information is contained in this circular segment, the two parts that contain telomeres can be lost.

These four rearrangement operations are often described as “standard” DCJ operations by introducing imaginary $\{\circ, \circ\}$ adjacencies: a DCJ operation applied to adjacencies $\{a^h, b^t\}$ and $\{\circ, \circ\}$ yields $\{a^h, \circ\}$ and $\{b^t, \circ\}$ and models fissions and linearizations. The reverse operation models fusions and circularizations.

4 Deriving the DCJ formula

It was a clever observation by Sophia Yancopoulos that the DCJ operation subsumes the two operations that have most prominently been discussed in the genome rearrangement literature up to 2005: inversions and translocations. In their paper [10], the authors also addressed the question of distance computation and gave the formula $D = b - c$ where $b := N - 1$ is the number of *initial breakpoints* between the N genes in the input genomes and c is a parameter closely related to the number of cycles in our adjacency graph. Nevertheless, their argument was rather informal. As said in the Introduction, it was our goal to formalize their approach and, if possible, simplify the argument and solution.

It was somehow clear to us that this should be possible, but even after almost a year of working on it, the exact way and the general DCJ formula still eluded our grasp. Therefore, in May 2005, we resorted to a ‘dirty trick’, based on just a few simple (and fortunately true) assumptions. Consider the following six parameters computed on the genomes and on the adjacency graph:

N : number of genes in each genome
C : number of cycles in the adjacency graph
I : number of odd paths in the adjacency graph
P : number of even paths in the adjacency graph
L : total number of linear chromosomes
R : total number of circular chromosomes

We begin with a well known mathematical technique called *guessing the solution*. Here, the educated guess is that the formula for the DCJ distance depends linearly on each of the above parameters, that is:

$$nN + cC + iI + pP + \ell L + rR = D,$$

where the coefficients n, c, i, p, ℓ and r are real numbers. Then we try to find the values of the coefficients.

These values are not necessarily independent. The most obvious relation is that the number L of linear chromosomes is related to the number of paths I and P by the equation:

$$L = I + P.$$

This means that we can keep one of the coefficient as an arbitrary constant, which we choose to be ℓ in the sequel. We will wait until all the values of the other coefficients are known, and then choose a value of ℓ that will make the formula look ‘simple’.

The next step is to consider a series of examples for which the DCJ distance is known. Each example will give a linear equation relating the values of the coefficients. The examples that we used in 2006 were lost in various paper baskets. This turns out to be a blessing, since recreating a suitable set of examples shows that only five very elementary examples are sufficient to determine the DCJ distance.

The simplest circular chromosomes. In these first two examples, we apply DCJ operations to a circular genome with two genes a and b. The graphs of Figures 4 and 5 represent the two possible operations. Since a single DCJ has been applied in each case, the distance is $D = 1$.

Example 2. Consider genomes $A = (a)\ (b)$ and $B = (a\ b)$, as in Figure 4. The corresponding equation is:

$$2n + c + 3r = 1.$$

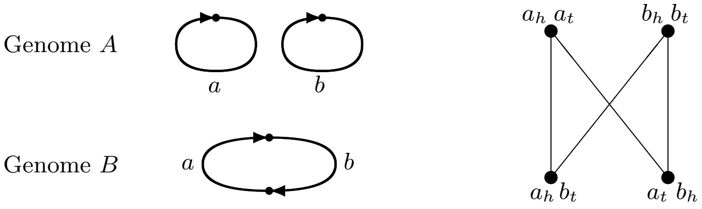


Fig. 4. A fusion of two circular chromosomes.

Example 3. Consider genomes $A = (a\ \bar{b})$ and $B = (a\ b)$, as in Figure 5, with the corresponding equation:

$$2n + c + 2r = 1.$$

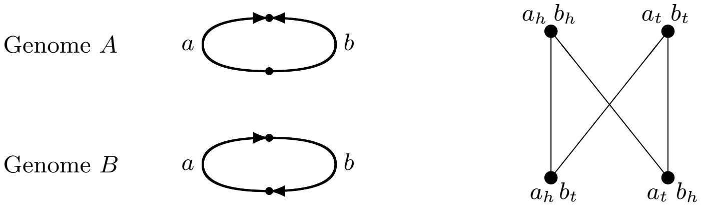


Fig. 5. An inversion within a circular chromosome.

From these equations we immediately conclude that $r = 0$, meaning that the distance is independent of the number of circular chromosomes. Setting r to its value yields the equation:

$$2n + c = 1. \tag{1}$$

Equality with one circular chromosome. The next equation is obtained by drawing the adjacency graph of a circular chromosome compared to itself. Obviously, the distance is $D = 0$ in this case, as in Figure 6.

Example 4. Consider genomes $A = (a)$ and $B = (a)$. The corresponding equation is:

$$n + c = 0. \tag{2}$$

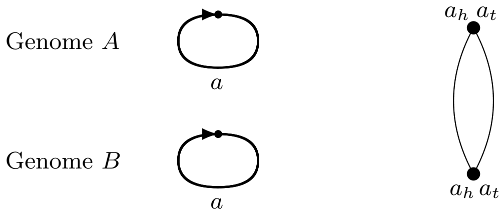


Fig. 6. Comparing a circular chromosome to itself.

Equations (1) and (2) yield $n = 1$ and $c = -1$.

Equality with one linear chromosome. When comparing a linear chromosome to itself, as in Figure 7, we also get a distance of $D = 0$.

Example 5. Consider genomes $A = (\circ\ a\ \circ)$ and $B = (\circ\ a\ \circ)$. The corresponding equation is:

$$n + 2i + 2\ell = 0. \tag{3}$$

Knowing that $n = 1$ and using ℓ as a constant, we get $i = -1/2 - \ell$.

Fusion/fission of linear chromosomes. The last example is given by the fusion of two linear chromosomes, and its dual operation, fission. In this case the distance is $D = 1$, and the adjacency graph is shown in Figure 8.

Example 6. Consider genomes $A = (\circ\ a\ \circ)\ (\circ\ b\ \circ)$ and $B = (\circ\ a\ b\ \circ)$. The corresponding equation is:

$$2n + 2i + p + 3\ell = 1, \tag{4}$$

which gives $p = -\ell$.

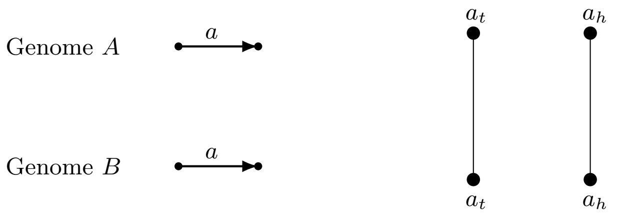


Fig. 7. Comparing a linear chromosome to itself.

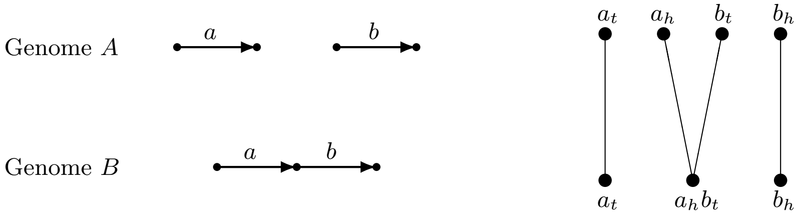


Fig. 8. Fusion/fission of linear chromosomes.

A simple formula. Summing up the work thus far, we get the following family of distance formulas, indexed with ℓ:

$$D = N - C - (1/2 + \ell)I - \ell P + \ell L.$$

Clearly, the 'simplest' formula is obtained by setting $\ell = 0$.

The formula $D = N - C - I/2$ is simple in the sense that it has the fewest number of parameters. However, this simplicity does not impose an intrinsic value on the parameter I and we will also see, in the subsections of Section 6, that setting $\ell = 0$ does not always yield distance formulas with the least number of parameters.

5 Proving the DCJ formula

Obtaining a formula based on a few examples does not mean that it computes the correct distance between arbitrary genomes: a general proof is still needed. In this section we discuss various topics associated with proving distance formulas, and we refer the reader to [2] for formal proofs.

The formula $D = N - C - I/2$ of the preceding section is not only the simplest but, as a nice added benefit, provides a roadmap for the general proof. The first step is to prove that the distance between two genomes is 0 if and only if the two genomes are equal. Here, this statement translates as:

Two genomes A and B are equal if and only if $N = C + I/2$.

The best informal justification of this result is to show the adjacency graph of two equal genomes with one circular and one linear chromosome, as in Figure 9 with genomes $A = B = (a\ b)\ (\circ\ c\ d\ \circ)$. This figure also illustrates that the graph is a collection of cycles of length 2 and paths of length 1.

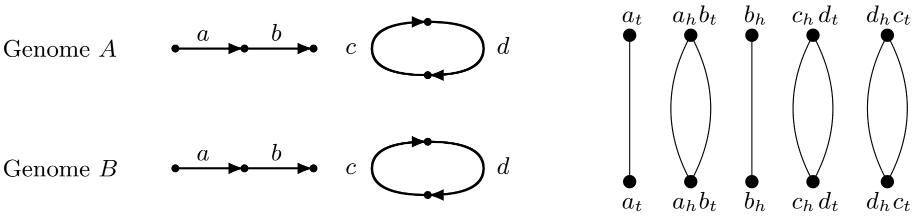


Fig. 9. The genome and adjacency graphs of two equal genomes.

The next step in proving the distance formula is to show that $D \geq N - C - I/2$. This is done by considering the quantity $C + I/2$ in the adjacency graph of genomes A and B, and showing that it increases by at most 1 for *any* DCJ operation applied to genome A. A nice way to enumerate all the cases is to realize that a DCJ operation applied to genome A is also a DCJ operation applied to the adjacency graph. Thus, for example, the number of cycles can be increased either by extracting a cycle, or by creating a cycle from a path. In the first two cases, the number of paths is unchanged, and, in the third case, the length of the path must be even, since the lengths of all cycles of an adjacency graph are even. The reader is welcome to complete the details for the case of odd paths.

The final step in the proof is to show that $D \leq N - C - I/2$. The easiest way to prove this is to construct an algorithm that effectively sorts genome A into genome B in exactly $N - C - I/2$ steps, by showing that there always exists a DCJ operation on genome A that either increases the number of cycles by 1, or the number of odd paths by 2. In fact, *any* adjacency of genome B that is not an adjacency of genome A can be created in one DCJ operation on genome A: this operation creates a cycle of length 2, and increases the number of cycles by 1. Once all adjacencies of genome B are created, the two genomes have the same adjacencies. If they are not equal, genome B has more telomeres than genome A, and a few fissions should solve the problem, each creating two paths of length 1.

6 Algebraic, single-cut and breakpoint distance formulas

As mentioned in the Introduction, DCJ is just one of several rearrangement models by which genomes can be compared. Since many of the alternative models are closely related to DCJ, it is not surprising that their distance formulas are somewhat related as well. In this section we show in a systematic way how to derive the distance formulas for four other genome rearrangement models, using the techniques introduced in Section 4. In fact, we can re-use much of the derivation of the DCJ distance formula and only small modifications are necessary.

6.1 The algebraic distance

A line of research in genome rearrangement that has been introduced by Meidanis and Dias [7] uses algebraic operations acting on genomes represented as permutations. Several "traditional" results can similarly be derived in that formalism, and some new models have also been introduced, including the so-called *algebraic* (ALG) distance [5]. In its most general version including linear and circular chromosomes, it is identical to the DCJ distance, except that fissions and fusions weigh 1/2 instead of 1.

Therefore, Equations (1), (2) and (3) are valid as well, the only difference is that Equation (4) becomes

$$2n + 2i + p + 3\ell = 1/2.$$

The values of n, c and i are the same as in the DCJ model, but p becomes:

$$p = -1/2 - \ell.$$

Thus the corresponding family of distance formulas is:

$$D_{ALG} = N - C - (1/2 + \ell)I - (1/2 + \ell)P + \ell L.$$

In this case, the 'simplest' formula is obtained by setting $\ell = -1/2$:

$$D_{ALG} = N - C - L/2.$$

However, this formula mixes parameters from the genome graph, L, and from the adjacency graph, C. In Section 5, we saw that choosing both parameters from the adjacency graph gave us a big advantage in interpreting and proving the DCJ distance formula. It might be wise to do the same in this case.

6.2 The Single-Cut-or-Join distance

Another rearrangement distance, that is particularly charming because it allows efficient computational solutions to complicated multi-genome comparison, is the Single-Cut-or-Join (SCorJ) distance [4]. Here, any cut in a chromosome, and any join of two chromosome ends is considered as an individual operation, each of weight 1.

When deriving the SCorJ distance formula, we need one more pair of genomes in order to distinguish the roles of the long and short cycles, and we must rewrite the first four equations accordingly. The parameter C is split in two:

C_s : number of cycles of length 2
C_ℓ : number of long cycles

and the first four equations become:

$$
\begin{aligned}
&2n + c_\ell = 4\\
&n + c_s = 0\\
&n + 2i + 2\ell = 0\\
&2n + 2i + p + 3\ell = 1
\end{aligned}
$$

The simplest rearrangement operation that gives the required new equation is a *transposition*, which exchanges two consecutive blocks of genes. In order to transpose blocks in a circular chromosome with Single-Cut-or-Join operations, it is necessary to cut three adjacencies, and join them at three different places.

Example 7. Consider the circular genomes $A = (a\ b\ c)$ and $B = (a\ c\ b)$, as in Figure 10. The corresponding equation is:

$$3n + c_\ell = 6. \tag{5}$$

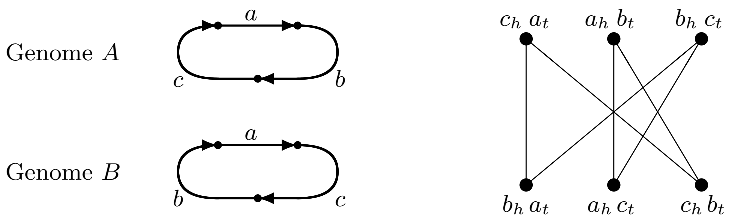


Fig. 10. A transposition within a circular chromosome.

The solution of the system is given by $n = 2$, $c_\ell = 0$, $c_s = -2$, $i = -1 - \ell$ and $p = -1 - \ell$, yielding the general formula

$$D_{SCorJ} = 2N - 2C_s - (1 + \ell)I - (1 + \ell)P + \ell L.$$

Here we would want to set $\ell = -1$, to get the 'simplest' distance formula:

$$D_{SCorJ} = 2N - 2C_s - L.$$

Again, here, this simplest formula might not be the wisest, since it mixes parameters from both the genome and the adjacency graphs.

6.3 The Single-Cut-and-Join distance

Similar by name to SCorJ, but more closely related to the DCJ distance is the Single-Cut-and-Join (SCandJ) distance [1] where a single operation comprises at most one cut, followed by at most one join.

Again, we distinguish long and short cycles and get the following first four equations:

$$\begin{aligned} 2n + c_\ell &= 3 \\ n + c_s &= 0 \\ n + 2i + 2\ell &= 0 \\ 2n + 2i + p + 3\ell &= 1 \end{aligned}$$

The final equation can be derived from a transposition as in Example 7, which has distance[3] $D = 4$, yielding a fifth equation:

$$3n + c_\ell = 4$$

Therefore the solution is $n = 1$, $c_\ell = 1$, $c_s = -1$, $2i+2\ell = -1$, $2i+p+3\ell = -1$, and we get the general formula:

$$D_{SCandJ} = N + C_\ell - C_s - (1/2 + \ell)I - \ell P + \ell L.$$

Setting $\ell = 0$, we find:

$$D_{SCandJ} = N + C_\ell - C_s - I/2.$$

6.4 The breakpoint distance

Finally we consider the *breakpoint* (BRK) distance, which is possibly the most classical and simplest genomic distance. Different from the previous ones, the breakpoint distance is not an edit distance, but just defined as the number of adjacencies that are present in one, but not in the other of the two input genomes [8].

The breakpoint distance needs to distinguish the odd paths of length 1, that are shared telomeres between genomes, from the longer odd paths. Thus we need two more coefficients:

I_s : number of paths of length 1
I_ℓ : number of long odd paths

[3] Since Single-Cut-and-Join operations are sometimes less intuitive, here is a scenario that sorts genome $(a\ b\ c)$ to genome $(a\ c\ b)$ in 4 steps. Cuts are indicated by vertical bars: $(a\ b\ c\ |) \longrightarrow (\circ\ a\ |\ b\ c\ \circ) \longrightarrow (\circ\ a\ \circ)\ (b\ |\ c) \longrightarrow (\circ\ a\ c\ b\ \circ) \longrightarrow (a\ c\ b)$.

We also need a revised set of equations, reflecting the new values on our pet examples for the breakpoint distance:

$$\begin{aligned}
&2n + c_\ell = 2\\
&n + c_s = 0\\
&n + 2i_s + 2\ell = 0\\
&2n + 2i_s + p + 3\ell = 1\\
&3n + c_\ell = 3
\end{aligned}$$

Finally, we need a last example to distinguish the roles of the short and long odd paths. The simplest one is the linear variant of Example 2.

Example 8. Consider genomes $A = (\circ\ a\ \bar{b}\ \circ)$ and $B = (\circ\ a\ b\ \circ)$ with the corresponding equation

$$2n + i_s + i_\ell + 2l = 3/2, \tag{6}$$

as in Figure 11. The value 3/2 comes from the definition of the general breakpoint distance for mixed multichromosomal genomes given in [8]. The solution is given by $n = 1, c_\ell = 0, c_s = -1, i_s = -1/2 - \ell, i_\ell = -\ell, p = -\ell$ with the general formula:

$$D_{BRK} = N - C_s - (1/2 + \ell)I_s - \ell I_\ell - \ell P + \ell L.$$

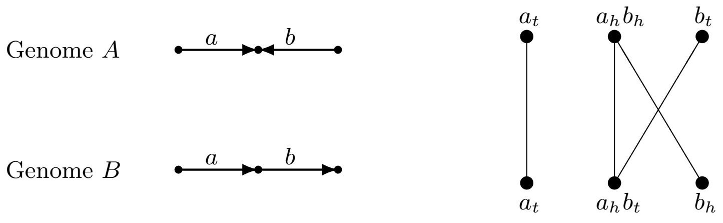


Fig. 11. An inversion inside a linear chromosome.

Setting $\ell = 0$, as in the DCJ distance, we get the breakpoint distance formula of David Sankoff and co-authors [8]:

$$D_{BRK} = N - C_s - I_s/2.$$

7 Conclusion

In this paper, we showed that many standard formulas for computing the rearrangement distance between genomes can be obtained using a handful of very simple genomes and a little linear algebra. Table 1 summarizes the principal results. It should be noted, though, that the formulas must be treated as conjectures. As for the DCJ formula, independent correctness proofs are needed, and are available in the literature.

Table 1. Distance parameters for various genomes and rearrangement models

Genomes	Equation	DCJ	ALG	SCorJ	SCandJ	BRK
$(a)\ (b)$ & $(a\ b)$	$2n + c_\ell$	1	1	4	3	2
(a) & (a)	$n + c_s$	0	0	0	0	0
$(\circ\ a\ \circ)$ & $(\circ\ a\ \circ)$	$n + 2i_s + 2\ell$	0	0	0	0	0
$(\circ\ a\ \circ)\ (\circ\ b\ \circ)$ & $(\circ\ a\ b\ \circ)$	$2n + 2i_s + p + 3\ell$	1	1/2	1	1	1
$(a\ b\ c)$ & $(a\ c\ b)$	$3n + c_\ell$	2	2	6	4	3
$(\circ\ a\ \bar{b}\ \circ)$ & $(\circ\ a\ b\ \circ)$	$2n + i_s + i_\ell + 2\ell$	1	1	2	4	3/2
Parameter	Coefficient					
Genes	n	1	1	2	1	1
Long cycles	c_ℓ	-1	-1	0	1	0
Short cycles	c_s	-1	-1	-2	-1	-1
Long odd paths	i_ℓ	$-1/2 - \ell$	$-1/2 - \ell$	$-1 - \ell$	0	$-\ell$
Short odd paths	i_s	$-1/2 - \ell$	$-1/2 - \ell$	$-1 - \ell$	0	$-1/2 - \ell$
Even paths	p	$-\ell$	$-1/2 - \ell$	$-2 - 2\ell$	$-\ell$	-2ℓ
Linear chromosomes	ℓ	ℓ	ℓ	ℓ	ℓ	ℓ

These distance formulas are similar, in the sense that they all depend linearly on the same set of parameters. However, in this setting, each model yields an infinite number of formulas: finding the simplest, or one that depends on given parameters, is just a mathematical trick. The important point is that the parameters should be chosen for their practical implications on the formulae, such as ease of interpretation or propensity toward elegant proofs.

References

1. A. Bergeron, P. Medvedev, and J. Stoye. Rearrangement models and single-cut operations. *Journal of Computational Biology*, 17(9):1213–1225, 2010.
2. A. Bergeron, J. Mixtacki, and J. Stoye. A unifying view of genome rearrangements. In *Proceedings of WABI 2006*, volume 4175 of *LNBI*, pages 163–173, 2006.
3. T. Dobzhansky and A. H. Sturtevant. Inversions in the Chromosomes of Drosophila Pseudoobscura. *Genetics*, 23(1):28–64, 1938.
4. P. Feijão and J. Meidanis. SCJ: A breakpoint-like distance that simplifies several rearrangement problems. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 8(5):1318–1329, 2011.
5. P. Feijão and J. Meidanis. Extending the algebraic formalism for genome rearrangements to include linear chromosomes. In *Proceedings of BSB 2012*, volume 7409 of *LNBI*, pages 13–24, 2012.
6. S. Hannenhalli and P. A. Pevzner. Transforming men into mice (polynomial algorithm for genomic distance problem). In *Proceedings of FOCS 1995*, pages 581–592, 1995.

7. J. Meidanis and Z. Dias. An alternative algebraic formalism for genome rearrangements. In *Comparative Genomics: Empirical and Analytical Approaches to Gene Order Dynamics, Map Alignment and the Evolution of Gene Families*, pages 213–223. Kluwer Academic Press, 2000.
8. E. Tannier, C. Zheng, and D. Sankoff. Multichromosomal median and halving problems under different genomic distances. *BMC Bioinformatics*, 10:120, 2009.
9. Wikipedia. http://en.wikipedia.org/wiki/DNA_repair.
10. S. Yancopoulos, O. Attie, and R. Friedberg. Efficient sorting of genomic permutations by translocation, inversion and block interchange. *Bioinformatics*, 21(16):3340–3346, 2005.