Reconstructing an ancestral genome using minimum segments duplications and reversals

Nadia El-Mabrouk
mabrouk@iro.umontreal.ca

Département d'informatique et de recherche opérationnelle, Université de Montréal, CP6128 succursale Centre-Ville, Montréal, Québec H3C 3J7

Abstract

We consider a particular model of genomic rearrangements that takes paralogous and orthologous genes into account. Given a particular model of evolution and an optimization criterion, the problem is to recover an ancestor of a modern genome modeled as an ordered sequence of signed genes. One direct application is to infer gene orders at the ancestral nodes of a phylogenetic tree.

Implicit in the rearrangement literature is that each gene has exactly one copy in each genome. This hypothesis is clearly false for species containing several copies of highly paralogous genes, e.g. multigene families. One of the most important regional event by which gene duplication can occur has been referred to as duplication transposition. Our model of evolution takes such duplications into account. For a genome G with gene families of different sizes, the implicit hypothesis is that G has an ancestor containing exactly one copy of each gene, and that G has evolved from this ancestor through a series of duplication transpositions and substring reversals. The question is : how can we reconstruct an ancestral genome giving rise to the minimal number of duplication transpositions and reversals? The key idea is to reduce the problem to a series of sub-problems involving genomes containing at most two copies of each gene. For this simpler version, we provide tight bounds, and we describe an algorithm, based on Hannenhalli and Pevzner graph and result, that is exact when certain conditions are verified. We then show how to use this algorithm to recover gene orders at the ancestral nodes of a phylogenetic tree.

1 Introduction

A fundamental problem arising in comparative genomics is to determine the evolutionary distance between two or more genomes. Traditionally, evolutionary relationships have been deduced from comparisons of single gene sequences. The availability of complete genome sequence data now enables evolutionary relationships to be inferred from entire genetic material, by comparing gene orders. This approach infers divergence history, not in terms of local mutations, but more global genomic mutations, involving the movement, inversion and duplication of chromosomal segments of various sizes.

Gene orders can be compared according to a variety of criteria. The breakpoint distance between two genomes G and H measures the number of disruptions between conserved segments of G and H. This metric has been used to infer phylogenetic trees [BKS99, SBD$^+$00]. Other metrics, **rearrangement distances**, are based on specific models of evolution. They measure the minimal number of genomic rearrangements necessary to transform one linear order of genes into another. The rearrangement operations that have been considered most often are: the inversion (reversal) of a chromosomal segment [HP95a, KST97, Ber01], the transposition of a segment from one site to another in a chromosome [BP98, WDM98] and the translocation (exchange) of terminal segments between two chromosomes [KR95, Han95, HP95b]. In the most realistic version of the rearrangement problem, a sign (+ or -) is associated with each object in the linear order, representing the transcriptional direction of the corresponding gene. The exact polynomial algorithm of Hannenhalli and Pevzner (hereafter "HP") for sorting signed permutations by reversals [HP95a] was a breakthrough for the formal analysis of evolutionary genome rearrangement. Moreover, they were able to extend their approach to include the analysis of reciprocal translocations [Han95, HP95b]. We have further extended the HP approach to include insertions and deletions of gene blocks, allowing to compare genomes with different gene contents [EM00b].

Implicit in the rearrangement literature, and in most tree reconstruction methods based on gene orders, is that each gene is present exactly once in each genome. While this hypothesis of unique genes may be appropriate for small genomes, e.g. viruses and organelles, it is clearly unguaranteed for divergent species containing several copies of highly paralogous and orthologous genes, scattered across the genomes. In comparing two or more genomes, it is, therefore, important to introduce the possibility of having different copies (paralogous) of the same gene in one genome, e.g. **multigene families**. The only existing version of the genomic rearrangement problem accounting for gene families is the exemplar approach [San99]. The idea is to remove all but one member of each gene family in each of the two genomes being compared, so as to minimize a rearrangement distance. Such an approach is not based on any specific evolutionary model. Our goal here is to consider a realistic model of duplication.

Several models have been proposed to account for the origin of gene duplications. These models fall into two categories: genome-wide doubling events, and duplications at a regional level. In [EMS01, EM00a, EMBS99], we have considered the genome duplication model and developed exact algorithms for reconstructing the ancestral doubled genome minimizing the number of reversals and/or translocations required to derive the observed order of genes along the present-day chromosomes. Here, we consider more localized duplication events. One of the most important regional events by which gene duplications can occur has been referred to as **duplication transposition** [OE00]. In this model, entire regions are duplicated from one location of the genome to another. Studies from human genomic sequence indicate that many of these segments have been duplicatively transposed in very recent evolutionary time [EAR99]. Many of these duplications play a role in both human disease and human evolution [MS98].

In this paper, we consider the problem of reconstructing an ancestral genome of a modern genome arising through duplication transpositions and reversals. For a genome G with gene families of different sizes, the implicit hypothesis is that G has an ancestor containing exactly one copy of each gene, and that G has evolved from this ancestor through a series of duplication transpositions and substring reversals. The question is : how can we reconstruct an ancestral genome giving rise to the minimal number $RD(G)$ of duplication transpositions and reversals? The key idea is to reduce the problem to a series of simpler sub-problems involving genomes containing at most two copies of each gene. To do so, each gene is either paired with one of its paralogous, or considered as a singleton. Such a pairing can be done in a relevant and efficient way if one have a preliminary information about evolutionary relationship between all genes of a gene family, summarized by a gene tree. This simpler version of the problem can be solved by using a method that resembles in many ways the technique we have developed previously to find an ancestral duplicated genome [EMBS99, EMS01]. It is based on the HP graph for sorting signed permutations by reversals. We provide tight bounds for $RD(G)$ and present an algorithm for recovering an ancestral genome H. When certain conditions are verified, the algorithm is exact, e.g. the reversal and duplication transposition distance between G and H is minimal over all possible ancestral genomes. We then show how to use our approach to reconstruct gene orders at the ancestral nodes of a species tree

$\mathcal{T}$, given N genomes (gene orders) and F gene trees.

We formalize our problem in Section 2. In Section 3, we summarize the essential concepts of the HP theory, and introduce some basic definitions. We provide, in Section 4, bounds for the minimal number of reversals/duplications required to obtained a modern genome G from and ancestral genome. Section 5 describes an algorithm for recovering an ancestor of G in the case of a genome G containing gene families of size one or two. Section 6 shows how to use the former algorithm to recover the ancestor of a genome G with gene families of any size. Finally, Section 7 shows how this approach can be used to recover gene orders at the ancestral nodes of a species tree.

2 Formalizing the problem

In this paper, we restrict ourselves to circular genomes, as most one-chromosomal genomes are circular. However, this hypothesis is not fundamental in our approach, and similar methods can be developed for linear genomes. A string is a sequence of signed ($+$ or $-$) terms (genes) from a set $\mathcal{B}$. We define a **genome** as a circular string, such that each gene of $\mathcal{B}$ can appear more than once in G, i.e. a genome can contain paralogous genes. Though we will write a genome as a linear string, it is understood that the first gene is adjacent to the right of the last gene. For example,

$$\Pi = +a\ +b\ +x\ +h\ +f\ +e\ +g\ -c\ -a\ -b\ -d\ -h\ -z\ +d\ +e\ -g\ -c\ -f\ +y$$

is a genome on the set of genes $\mathcal{B} = \{a, b, c, d, e, f, g, h, x, y, z\}$.

An **ambiguous genome** is a genome containing at least one gene of $\mathcal{B}$ in more than one copy. Otherwise, the genome is said **non-ambiguous**. The above genome Π is an ambiguous genome.

For a string $X = x_1 x_2 \cdots x_r$, denote by $-X$ the reverse string $-x_r - x_{r-1} \cdots - x_1$. A **reversal** transforms some proper substring of a genome into its reverse. A **duplication transposition** (direct or indirect), or simply a **duplication**, is an operation that transforms a genome $G = ABCD$ into a genome $G' = ABCBD$ or $G' = ABC - BD$, where A, B, C, D are four substrings of G, possibly empty except for B. Notice that B can be restricted to one gene.

The problem is to find the **reversal/duplication distance** of a modern ambiguous genome G, that is the minimal number $\mathbf{RD(G)}$ of reversals and duplications that transforms an unknown non-ambiguous genome H into G, and simulate a possible sequence of such mutations. The key idea is to reduce the problem to a series of sub-problems involving simplified data. A **semi-ambiguous genome** G is an ambiguous genome such that each gene of $\mathcal{B}$ appears at most twice in G. A gene that has only one copy in G is called a **singleton**, otherwise it is called a **duplicated gene**. We denote by $\mathcal{B}_s$ the set of singletons, and by $\mathcal{B}_d$ the set of duplicated genes of G. For example, the above genome Π is a semi-ambiguous genome, with $\mathcal{B}_d = \{a, b, c, d, e, f, g, h\}$ and $\mathcal{B}_s = \{x, y, z\}$.

A **repeat** is a maximal substring of G that is present twice in the genome, or such that its reverse is also a substring of G. If one copy is denoted S, then the other copy, called the **complementary** of S, is denoted $\overline{S}$. S and $\overline{S}$ are considered as the same repeat. We denote by $\mathbf{D(G)}$ the number of repeats of G. For example, the following genome:

$$\underbrace{+a\ -b\ +c}_{S_1}\ +x\ \underbrace{+d - e}_{S_2}\ \underbrace{+e\ -d}_{\overline{S_2}}\ \underbrace{+a\ -b\ +c}_{\overline{S_1}}\ +y$$

contains two repeats. In the case of the genome Π, as each repeat contains exactly one gene, $D(\Pi) = 8$.

We consider the following evolutionary model for semi-ambiguous genomes: a semi-ambiguous genome G on a gene set $\mathcal{B}$ has an ancestor H containing exactly one copy of each gene, and G has evolved from H through a series of duplications, giving rise to an **intermediate ancestral**

genome I, which is a genome containing exactly the same genes as those in G in the same number of copies, followed by a series of reversals (see Figure 1). Then, our problem is just to reconstruct an intermediate ancestral genome I such that $D(I) + R(G, I)$ is minimal over all possible intermediate ancestral genomes, where $\mathbf{R(G, I)}$ is the reversal distance between G and I. Indeed, it is straightforward to recover, from I, a genome H giving rise to $D(I)$ duplications. Thus, the only ancestral genome which is of interest is I. Therefore, in the rest of this paper, an **ancestral genome** of G will refer to a genome containing exactly the same genes than G, in the same number of copies.

Hereafter, we use the notations: $\mathbf{RD(G, I) = D(I) + R(G, I)}$. Then, $RD(G)$ is the minimal value of $RD(G, I)$ over all possible ancestral genomes I of G.

$$\begin{array}{c}
H \\
+a\ \ +b\ \ +c\ \ +d \\
\Big\downarrow \text{Duplications} \\
I \\
+a\ \ +b\ \ +c\ \ +a\ \ +b\ \ +d\ \ -c \\
\Big\downarrow \text{Reversals} \\
G \\
+a\ \ -c\ \ -d\ \ -b\ \ -a\ \ +b\ \ -c
\end{array}$$

Figure 1: G has evolved from a genome H through two duplications, giving rise to an ancestral genome I, followed by a series of reversals. I has two repeats: $\{+a+b, +c\}$; G has 3 repeats: $\{+a, +b, +c\}$.

Remark 1 *The constraint, in our evolutionary model, to have all duplications first and then all reversals can be seen as restrictive. However, notice that the semi-ambiguous genome problem is just a sub-problem of the general ambiguous genome problem. In section 6, we will show how gene trees can be used to determine which duplications have occurred at the same historical time. Only the genes involved in some contemporaneous duplications will be considered as duplicated genes, and all other genes will be considered as singletons. Therefore, for such sub-problems, it is justified to consider that all duplications have occurred simultaneously, and then a series of reversals have disrupted gene orders. However, as the general problem is solved as a series of such sub-problem, the general model of evolution for ambiguous genomes is a mix of reversals and duplications: a series of duplications, followed by a series of reversals, followed by a series of duplications, followed by a series of reversals$\cdots$ Another reason for this restriction is to simplify the presentation of our algorithm. Notice also that our goal is not to output all possible sequences of reversals/duplications, but just one optimal sequence.*

3 Preliminaries

Our method is based on Hannenhalli and Pevzner (HP) graph and result for sorting signed permutations by reversals [HP95a]. We briefly introduce the basic concepts of this theory.

3.1 The Hannenhalli and Pevzner graph and result

Given two genomes H_1, H_2 on the same set of genes $\mathcal{B}$, each gene of $\mathcal{B}$ appearing exactly once in each genome, the HP problem is to find the minimal number $R(H_1, H_2)$ of reversals required to transform H_1 to H_2. The HP result and algorithm depends on a bicoloured cycle graph $\mathcal{G}_{12}$ constructed from H_1 and H_2 as follows : if gene x of H_1 has positive sign, replace it by the pair

$x^t x^h$, and if it is negative, by $x^h x^t$. Then the vertices of $\mathcal{G}_{12}$ are just the x^t and the x^h for all x in $\mathcal{B}$. Any two vertices which are adjacent in H_1, other than x^t and x^h deriving from the same x, are connected by a black edge, and any two adjacent in H_2, by a gray edge. This graph decomposes naturally into a set of c_{12} disjoint colour-alternating cycles. By the **size of a cycle** we will mean the number of black edges (or similarly of gray edges) of the cycle. Note that c_{12} is maximized when $H_1 = H_2$, in which case each cycle is of size 1. Therefore, minimizing the number of reversals can be seen in terms of increasing the number of cycles as fast as possible. Let ρ be a reversal, and $\Delta(c)$ the difference between the number of cycles of the graph, before and after applying the operation ρ. Hannenhalli and Pevzner show that $\Delta(c)$ may take on values 1, 0 or -1, in which cases they called ρ **proper, improper** or **bad**, respectively. Roughly speaking, a reversal acting on two black edges in two different cycles will be bad, while one acting on two black edges within the same cycle may be proper or improper, depending on the type of cycle and the type of edges considered. Key to the HP approach are the graph components. Two cycles, say Cycles 1 and 2 containing gray edges that "cross", e.g., gene i linked to gene j by a black edge (i.e. in H_1) in Cycle 1, gene k linked to gene t by a black edge in Cycle 2, but ordered i, k, j, t in H_2, are **connected**. A **component** of $\mathcal{G}_{12}$ is a maximal set of connected cycles (see Figure 2).

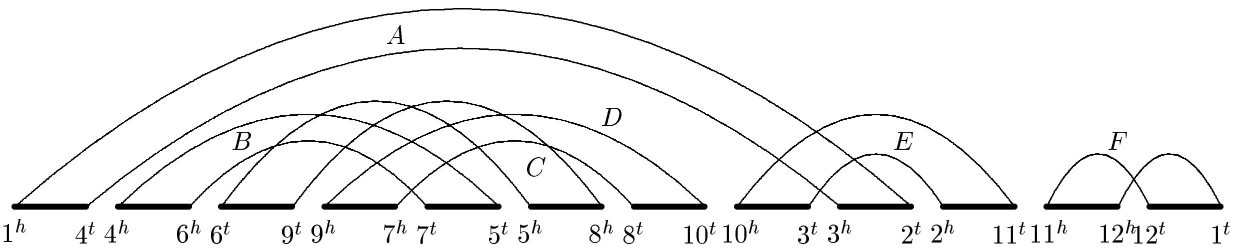


Figure 2: Graph $\mathcal{G}_{12}$ corresponding to circular genomes (i.e. first gene is adjacent to last gene) $H_1 = +1 +4 -6 +9 -7 +5 -8 +10 +3 +2 +11 -12$ and $H_2 = +1 +2 +3 \cdots +12$. A, B, C, D, E and F are the 6 cycles of $\mathcal{G}_{12}$. $\{A, E\}, \{B, C, D\}$ and $\{F\}$ are the three components of $\mathcal{G}_{12}$. $\{B, C, D\}, \{F\}$ are good components, and $\{A, E\}$ is a bad one.

A gray edge is **oriented** if it links two left vertices of two black edges, or two right vertices of two black edges. A component is **good** if it contains at least one oriented gray edge, and **bad** otherwise. HP showed that a good component can be transformed to a set of cycles of size 1 by a series of proper reversals. As for bad components, some of them can still be solved by proper reversals, whereas others, called **hurdles**, require bad reversals to be solved.

The HP final result is:

$$R(H_1, H_2) = |A| - c(\mathcal{G}_{12}) + h(\mathcal{G}_{12}) + fr(\mathcal{G}_{12}) \qquad \textbf{(HP formulae)}$$

where $|A|$ is the cardinality of the set A of black edges, $c(\mathcal{G}_{12})$ is the number of cycles, $h(\mathcal{G}_{12})$ is the number of hurdles of $\mathcal{G}_{12}$, and $fr(\mathcal{G}_{12})$ is a parameter which is 0 or 1 depending on the type of hurdles in $\mathcal{G}_{12}$ [HP95a].

As the probability for a given component to be good is bigger than its probability to be bad, the number of hurdles is usually close to 0. Therefore, the number of cycles of $\mathcal{G}_{12}$ is the dominant parameter in the HP formulae for $R(H_1, H_2)$, if $b(\mathcal{G}_{12})$ is considered as a constant. In other words, the more cycles there are, the less reversals we need to transform H_1 into H_2.

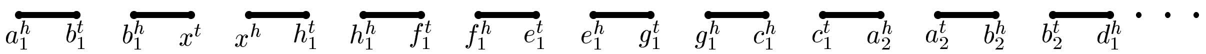
3.2 Notations and definitions

To make use of the HP graph structure in our problem of reconstructing an ancestor of a semi-ambiguous genome G, we introduce arbitrarily a distinction within each pair of paralogous genes, labeling one occurrence x_1 and the other x_2 for each duplicated gene x. Moreover, each gene x_j (either a duplicate or a singleton) is replaced by x_j^t and x_j^h as in the HP construction. For example, our example genome Π is rewritten:

$$G = a_1^{th}\ b_1^{th}\ x^{th}\ h_1^{th}\ f_1^{th}\ e_1^{th}\ g_1^{th}\ c_1^{ht}\ a_2^{ht}\ b_2^{ht}\ d_1^{ht}\ h_2^{ht}\ z^{ht}\ d_2^{th}\ e_2^{th}\ g_2^{ht}\ c_2^{ht}\ f_2^{ht}\ y^{th}$$

(each $x^{s_1 s_2}$ represents x^{s_1} followed by x^{s_2}). We use the notation $\overline{1} = 2$, $\overline{2} = 1$, $\widetilde{t} = h$, $\widetilde{h} = t$.

Define the sets of vertices $S = \{x^s\}_{x \in \mathcal{B}_s}^{s \in \{h,t\}}$ and $V = S \cup \{x_j^s\}_{\substack{x \in \mathcal{B}_d \\ j=1,2}}^{s \in \{h,t\}}$. The **partial graph** $\mathcal{G}$ associated with G has the vertices set V, and the edge set A of (black) edges linking adjacent terms (other than x^t and x^h for the same gene x) in G. For example, the partial graph associated to the genome Π is as follows:



In the case of reconstructing an ancestral genome of a modern genome G, only the partial graph associated with G is known. The problem is then to **complete** the partial graph by a set Γ of gray edges satisfying the properties: (1) every vertex in V is incident to exactly one gray edge of Γ; (2) a gray edge does not link x^t and x^h for the same gene x. Such a set of gray edges does not necessarily correspond to a single circular genome. If not, it should be modified. We denote by $\mathcal{G}(\Gamma)$ the graph $\mathcal{G}$ completed by a set Γ of gray edges.

Recall that the goal is to reconstruct an ancestral genome I such that $D(I)+R(G,I)$ is minimal over all possible ancestral genomes of G. As the number of cycles is the dominant parameter in the HP formulae, we first consider the problem of constructing a set Γ of gray edges giving rise to a genome I minimizing $D(I) + |A| - c(\Gamma)$, where $c(\Gamma)$ is the number of cycles of the corresponding completed graph $\mathcal{G}(\Gamma)$. Denote $\mathbf{\Delta(G, I) = D(I) + |A| - c(\Gamma)}$. Then $RD(G, I)$ is just $\Delta(G, I) + h(\Gamma) + f(\Gamma)$, where $h(\Gamma)$ is the number of hurdles of $\mathcal{G}(\Gamma)$, and $f(\Gamma)$ is 0 or 1 depending on the set of hurdles. Also denote by $\mathbf{\Delta(G)}$ the minimal value of $\Delta(G, I)$ over all possible ancestral genomes I of G.

The key observation is that an ancestral genome I of G can be modeled as a sequence of duplicated segments (chromosome-like segments) separated by singletons. It can be viewed as a multichromosomal genome, and the elements of S as "dummy genes" added to the ends of the chromosomes. Therefore, there are some similarities between this problem and that of recovering a perfect duplicated genome in the case of the multichromosomal model [EMS01, EMBS99]. However, additional difficulties are due to the fact that the final genome should be a circular, one-chromosomal genome. Indeed, as these chromosomes-like strings are concatenated, the "dummy genes" can not be added randomly at the ends of the individual duplications.

As in [EMS01], we decompose the partial graph $\mathcal{G}$ into a set of subgraphs, defined as follows.

Definition 1 *Let $e = (u, v) \in A$. Define A_e recursively by $(u, v) \in A_e$, and if $(x, y) \in A_e$ then both the edge of A adjacent to $\overline{x}$ (if x is not in S) and the edge of A adjacent to $\overline{y}$ (if y is not in S) are also in A_e. Then the subgraph $\mathcal{G}_e$ of $\mathcal{G}$, made up of the set A_e of black edges and of the set V_e of vertices incident to the edges of A_e, is the* **natural graph** *(of* **size** *$|A_e|$) of $\mathcal{G}$ generated by e. Note that if $f \in A_e$, then $A_f = A_e$.*

For example, the natural graphs corresponding to the genome Π are shown in Figure 3. We denote by $\mathcal{E}$ the set of natural graphs of $\mathcal{G}$ containing those graphs of even size that do not contain any vertex in S (i.e. with all vertices corresponding to duplicated genes). Also we denote by $\mathcal{N}$

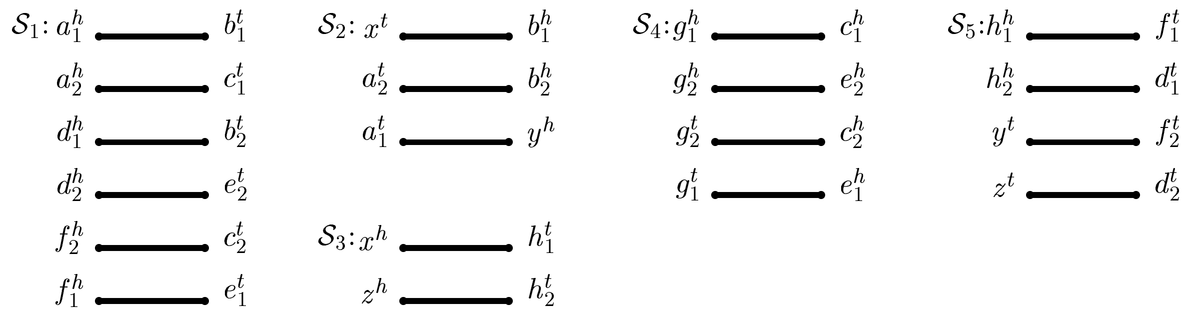


Figure 3: The natural graphs corresponding to the genome Π. $\mathcal{S}_1$ and $\mathcal{S}_4$ are in $\mathcal{E}$, and the three other natural graphs are in $\mathcal{N}$.

the set of all natural graphs of $\mathcal{G}$ that are not in $\mathcal{E}$. In Figure 3, $\mathcal{S}_1$ and $\mathcal{S}_4$ are in $\mathcal{E}$, and the three other natural graphs are in $\mathcal{N}$.

We require a final definition. A **fragment** of a genome G is just a substring of G, that can be circular or linear. For example, $+f_1\ +e_1\ +g_1\ -c_1$ is a fragment of the genome Π. If this fragment is linear, then it has two **endpoints** which are f_1^t and c_1^t. A **circular fragment** does not have any endpoint.

4 Bounds on the number of duplications and reversals

Let G be a semi-ambiguous genome.

Lemma 1 *The reversal/duplication distance $RD(G)$ of G is at most $D(G)$.*

Proof: Let $X_1, \cdots, X_d$ be the $D(G)$ repeats of G, and $\overline{X_1}, \cdots, \overline{X_d}$ be their complementary sequences. Let H be the non-ambiguous genome obtained from G by removing all the substrings $\overline{X_i}$, for $1 \leq i \leq d$. Then, the genome G can be obtained from H by performing $D(G)$ duplications, and 0 reversals □

Lemma 2 *A reversal can decrease by at most two the number of repeats of a genome G.*

Proof: Let $X = aEb$ be a substring of G, where a, b are duplicated genes, and E is a string (possibly empty). Let ρ be the reversal giving rise to the substring $X^r = a\ \ -E\ b$. Denote by $\rho(G)$ the genome obtained after applying ρ.

- Let $E = c$ be a duplicated gene. Suppose G has a substring of form $Y = \overline{a}\ {-\overline{c}}\ \overline{b}$, and $G \neq XY$. In that case, a, b, c are part of three different repeats in G, but are part of the same repeat in $\rho(G)$. Thus, $\rho(G)$ decreases by two the number of repeats of G.

- Let $E = cFd$, where c, d are duplicated genes. Suppose G has a substring of form $Y = \overline{a}\ \ -\overline{d}\ K\ \ -\overline{c}\overline{b}$, and $G \neq XY$. In that case, a, b, c, d are part of four different repeats in G, but are part of only two different repeats in $\rho(G)$. Thus, $\rho(G)$ decreases by two the number of repeats of G.

It is straightforward to see that these two cases are the only ones that give rise to a reversal decreasing by two the number of repeats of G. Moreover, as a reversal can act on only two breakpoints, it is clear that a reversal can not decrease by more than two the number of repeats □

As a reversal can decrease by 2 the number of repeats of a genome G, the reversal/duplication distance of G can be strictly lower than $D(G)$.

As for the problem of recovering a duplicated genome [EMS01], the reversal/duplication distance can be computed by analyzing separately each natural graph of $\mathcal{G}$. We require some additional definitions. Let E be a set of black edges or a set of gray edges linking vertices of V. We define the **fragments set** $\mathcal{F}$ **of** E as follows: $\mathcal{F}$ contains all maximal length strings $x_1 \cdots x_p$ such that, for any i, x_i, x_{i+1} are linked by an edge of E. For example, the fragments set corresponding to the set of black edges of the natural graph $\mathcal{S}_4$ in Figure 3 is $\mathcal{F}_4 = \{+e_1 + g_1 - c_1, +c_2 + g_2 - e_2\}$. A **repeat of a fragments set** $\mathcal{F}$ is defined as for a repeat of a genome. The above set $\mathcal{F}_4$ contains 3 repeats. We will denote by $D(\mathcal{F})$, or equivalently $D(E)$ the number of repeats of the fragments set $\mathcal{F}$ corresponding to the set E of edges.

Now, let $\mathcal{G}_\alpha$ be a natural graph of $\mathcal{G}$, with the set of vertices V_α and the set of black edges A_α. This natural graph represents a fragments set $\mathcal{F}_\alpha$. We want to complete $\mathcal{G}_\alpha$ by a set of gray edges. Let Γ_α be such a set and $\mathcal{I}_\alpha$ be its corresponding fragments set. Denote $\Delta(A_\alpha, \Gamma_\alpha) = D(\Gamma_\alpha) + |A_\alpha| - c(\Gamma_\alpha)$, where $c(\Gamma_\alpha)$ is the number of cycles of $\mathcal{G}_\alpha(\Gamma_\alpha)$. Denote also by $\Delta(A_\alpha)$ the minimal value of $\Delta(A_\alpha, \Gamma_\alpha)$ over all possible set of gray edges Γ_α.

Lemma 3 *If $\mathcal{G}_\alpha$ is in $\mathcal{E}$, then $\Delta(A_\alpha) \geq D(A_\alpha) - 1$.*

Proof: For a natural graph $\mathcal{G}_\alpha$ of $\mathcal{E}$ of size 2, it is clear that $D(A_\alpha) = 1$ and that $\Delta(A_\alpha) \geq 0$. Therefore, the inequality $\Delta(A_\alpha) \geq D(A_\alpha) - 1$ is true. In the rest of this proof we consider a natural graph $\mathcal{G}_\alpha$ of $\mathcal{E}$ of size greater than 2.

Each vertex of V_α is of form x_i^s, with $1 \leq i \leq 2$ and $s \in \{t, h\}$, and x is a gene of $\mathcal{B}_d$. Let v be the number of xs, i.e. genes, represented by the vertices of V_α. Then, $D(A_\alpha) = v$. Indeed, if $D(A_\alpha) < v$, then the vertices of V_α would have been scattered in more than one natural graph.

Let v' be the number of xs verifying: exactly one of the two vertices x_1^t, x_1^h is in V_α. As $|A_\alpha| = v' + 2(v - v') = 2v - v'$, v' is necessarily even. Moreover, $\mathcal{F}_\alpha$ contains v' fragments. Therefore, $\mathcal{I}_\alpha$ contains at least $v'/2$ repeats, that is $D(\Gamma_\alpha) \geq v'/2$.

Suppose that Γ_α is such that $D(\Gamma_\alpha) = v'/2$ (such a genome exists). Then, from [EMS01], $c(\Gamma_\alpha) \leq |A_\alpha|/2 + 1$, thus $|A_\alpha| - c(\Gamma_\alpha) \geq |A_\alpha|/2 - 1$. Therefore, $\Delta(A_\alpha, \Gamma_\alpha) \geq v'/2 + |A_\alpha|/2 - 1$. Now, $v'/2 + |A_\alpha|/2 - 1 = v - 1$, thus $\Delta(A_\alpha, \Gamma_\alpha) \geq v - 1$ $(*)$.

Suppose now that there exist a fragments set $\mathcal{I}'_\alpha$, induced by a set Γ' of gray edges, such that $D(\Gamma'_\alpha) + |A_\alpha| - c(\Gamma'_\alpha) < v - 1$. We deduce from $(*)$ that $D(\Gamma'_\alpha) - c(\Gamma'_\alpha) < D(\Gamma_\alpha) - c(\Gamma_\alpha)$, that is $\Delta(d) = D(\Gamma'_\alpha) - D(\Gamma_\alpha) < c(\Gamma'_\alpha) - c(\Gamma_\alpha) = \Delta(c)$. As $D(\Gamma'_\alpha)$ can not be less than $D(\Gamma_\alpha)$, we should have $c(\Gamma'_\alpha) > c(\Gamma_\alpha)$.

We prove by induction that $\Delta(d) \geq \Delta(c)$:

1: Suppose that $\Delta(c) = 1$. Then, from [EMS01], $\mathcal{I}'_\alpha$ can not be subdivided into pairs of duplicated fragments, and thus $D(\Gamma'_\alpha) > D(\Gamma_\alpha) \Longrightarrow \Delta(d) \geq 1 = \Delta(c)$.

n-1: Suppose that for any $p < n$, if $\Delta(c) = p$, then $\Delta(d) > \Delta(c)$.

n: Let $\Delta(c) = n$. As $c(\Gamma'_\alpha) > c(\Gamma_\alpha)$, $\mathcal{I}'_\alpha$ can not be a duplicated fragments set. Thus there exist two vertices $a, \overline{a}$ of V_α, and two gray edges $(a, b), (\overline{a}, c)$ of Γ'_α such that $c(\Gamma_\alpha) \neq \overline{b}$. Let $(\overline{b}, d)$ be the gray edge of Γ'_α adjacent to $\overline{b}$.

Let Γ''_α be the set of gray edges obtained from Γ'_α by removing the edges $(\overline{a}, c), (\overline{b}, d)$, and replacing them by the edges $(\overline{a}, \overline{b}), (c, d)$. Then, $c(\Gamma''_\alpha) = c(\Gamma'_\alpha) - 1$ and $D(\Gamma''_\alpha) = D(\Gamma'_\alpha) - 1$. From the induction hypothesis, we have $D(\Gamma''_\alpha) - D(\Gamma_\alpha) > c(\Gamma''_\alpha) - c(\Gamma_\alpha)$, and thus $\Delta(d) = D(\Gamma'_\alpha) - D(\Gamma_\alpha) > c(\Gamma'_\alpha) - c(\Gamma_\alpha) = \Delta(c)$ □

Lemma 4 *If $\mathcal{G}_\alpha$ is in $\mathcal{N}$, then $\Delta(A_\alpha) = D(A_\alpha)$.*

Proof: Suppose that $\mathcal{G}_\alpha$ is a natural graph with an even number of black edges. Then, V_α contains exactly two vertices corresponding to singletons. Let v be the number of genes of $\mathcal{B}_d$ represented by vertices of V_α. Then, $D(A_\alpha) = v$.

Let v' be the number of xs corresponding to duplicated genes verifying: exactly one of the two vertices x_1^t, x_1^h is in V_α. Then $\mathcal{F}_\alpha$ contains $v'+1$ fragments, and $\mathcal{I}_\alpha$ contains at least $(v'+1)/2$ repeats, that is $D(\Gamma_\alpha) \geq (v'+1)/2$.

Suppose that $\mathcal{I}_\alpha$ contains exactly $(v'+1)/2$ repeats (such a genome exists). Then, from [EMS01], $c(\Gamma_\alpha) \leq |A_\alpha|/2+1$, thus $|A_\alpha| - c(\Gamma_\alpha) \geq |A_\alpha|/2-1$. Therefore, $\Delta(A_\alpha, \Gamma_\alpha) \geq (v'+1)/2 + A_\alpha|/2-1$. As $|A_\alpha| = v' + 2(v-v') + 1 = 2v - v' + 1$, $(v'+1)/2 + |A_\alpha|/2 - 1 = v$, and thus $\Delta(A_\alpha, \Gamma_\alpha) \geq v$.

We prove, as for lemma 3, that we can not construct a completed graph $\mathcal{G}_\alpha(\Gamma'_\alpha)$ such that $D(\Gamma'_\alpha) - D(\Gamma_\alpha) \leq c(\Gamma'_\alpha) - c(\Gamma_\alpha)$. We deduce that $\Delta(A_\alpha) \geq D(A_\alpha)$. Now, as $\Delta(A_\alpha) \leq D(A_\alpha)$, we have $\Delta(A_\alpha) = D(A_\alpha)$.

We prove in the same way that if $\mathcal{G}_\alpha$ is a natural graph of odd size, then $\Delta(A_\alpha) = D(A_\alpha)$ □

Let $\mathcal{I}$ be the fragments set corresponding to the set Γ of gray edges of a completed graph $\mathcal{G}(\Gamma)$. Notice that such a fragments set does not necessarily correspond to an intermediate genome, that is to a unique circular fragment.

Denote $\Delta(A, \Gamma) = D(\Gamma) + |A| - c(\Gamma)$. Denote also by $\Delta(G)$ the minimal value of $\Delta(A, \Gamma)$ over all possible sets Γ of gray edges.

Theorem 1 $\Delta(G) \geq D(G) - |\mathcal{E}|$.

Proof: Let $\mathcal{G}(\Gamma)$ be a completed graph obtained by completing individually each natural graph, and $\mathcal{I}$ be the fragments set corresponding to Γ. For each natural graph $\mathcal{G}_\alpha$, let Γ_α be its corresponding set of gray edges. We have:

$$\begin{aligned}
\Delta(A,\Gamma) - D(G) &= |A| - c(\Gamma) - (D(G) - D(\Gamma)) \\
&= \Sigma_\alpha \left(|A_\alpha| - c(\Gamma_\alpha) - (D(A_\alpha) - D(\Gamma_\alpha))\right) \\
&= \Sigma_\alpha \Delta(A_\alpha, \Gamma_\alpha) - \Sigma_\alpha D(A_\alpha) \\
&\geq \Sigma_\alpha D(A_\alpha) - |\mathcal{E}| - \Sigma_\alpha D(A_\alpha) \text{ from lemma 3 and 4} \\
&= -|\mathcal{E}|
\end{aligned}$$

$\Longrightarrow \Delta(G, \mathcal{I}) \geq D(G) - |\mathcal{E}|$.

Therefore, for any set Γ obtained by completing each natural graph individually, we have $\Delta(A, \Gamma) \geq D(G) - |\mathcal{E}|$.

Suppose now that Γ contains bad edges, that is edges linking two vertices of two different natural graphs. Let $b(\Gamma)$ be the number of such bad edges. Then, we will show that we can replace Γ by a set Γ', such that the number $b(\Gamma')$ of bad edges of Γ' is at most $b(\Gamma) - 1$ bad edges, and such that $\Delta(A, \Gamma) \geq \Delta(A, \Gamma')$. By iterating this procedure until no bad edges remain, we can then deduce that $\Delta(A, \Gamma) \geq D(G) - |\mathcal{E}|$. We use the notation $\Delta(c) = c(\Gamma') - c(\Gamma)$, and $\Delta(d) = D(\Gamma') - D(\Gamma)$.

So, let (a, b) be an edge of Γ such that a, b are, respectively, in the two disjoint natural graphs $\mathcal{G}_a$ and $\mathcal{G}_b$. Let C be the cycle of $\mathcal{G}(\Gamma)$ containing (a, b). Then, C contains one bad gray edge (c, d) such that c is in $\mathcal{G}_a$, and one bad gray edge (e, f) such that e is in $\mathcal{G}_b$ (see Figure 4). Notice that (c, d) can be equal to (e, f). Suppose first that $(c, d) \neq (e, f)$.

Consider the set Γ' obtained from Γ by removing $(a, b), (c, d), (e, f)$, and adding $(a, c), (d, f)$, (b, e). Then we have obviously $b(\Gamma') \leq b(\Gamma) - 1$, and $\Delta(c) = 2$. Therefore, if $\Delta(d) \leq 2$, $\Delta(c) - \Delta(d) \geq 0$, and therefore $\Delta(A, \Gamma) \geq \Delta(A, \Gamma')$.

Suppose now that $\Delta(d) > 2$. As the only modified edges are those linking the vertices a, b, c, d, e, f, the only modified repeats are those containing these vertices. $\Delta(d) > 2$ means that at least three new repeats are created by removing the edges $(a, b), (c, d), (e, f)$. This means that the two genes of each of these pairs belong to one repeat of $\mathcal{I}$ (a, b belong to the same repeat, and the same holds for c, d and e, f), in other words, the vertices a, b, c, d, e, f belong to at most three repeats. This is verified if and only if Γ contains also the three gray edges $(\overline{a}, \overline{b}), (\overline{c}, \overline{d}), (\overline{e}, \overline{f})$. In that case, removing the edges $(a, b), (c, d), (e, f)$ and replacing them by the edges $(a, c), (d, f), (b, e)$ gives rise to the six adjacencies: $ac, df, be, \overline{a}\overline{b}, \overline{c}\overline{d}, \overline{e}\overline{f}$. To solve this case, we construct the set Γ'' from Γ' by removing $(\overline{a}, \overline{b}), (\overline{c}, \overline{d}), (\overline{e}, \overline{f})$, and adding $(\overline{a}, \overline{c}), (\overline{d}, \overline{f}), (\overline{b}, \overline{e})$. This gives rise to the six adjacencies: $ac, df, be, \overline{ac}, \overline{df}, \overline{be}$, and thus a, b, c, d, e, f belong to at most three repeats of the new fragments set $\mathcal{I}''$ (see Figure 4). Now, for $\Delta(d) = D(\Gamma'') - D(\Gamma)$, we have $\Delta(d) \leq 2$.

Suppose now that $(c, d) = (e, f)$. Then a similar approach than for $(c, d) \neq (e, f)$ shows that Γ can be replace by Γ' such that $b(\Gamma') \leq b(\Gamma) - 1$, $\Delta(c) = 1$ and $\Delta(d) \leq 1$. From the two last inequalities, we deduce that $\Delta(c) - \Delta(d) \geq 0$, and therefore $\Delta(A, \Gamma) \geq \Delta(A, \Gamma')$ $\square$

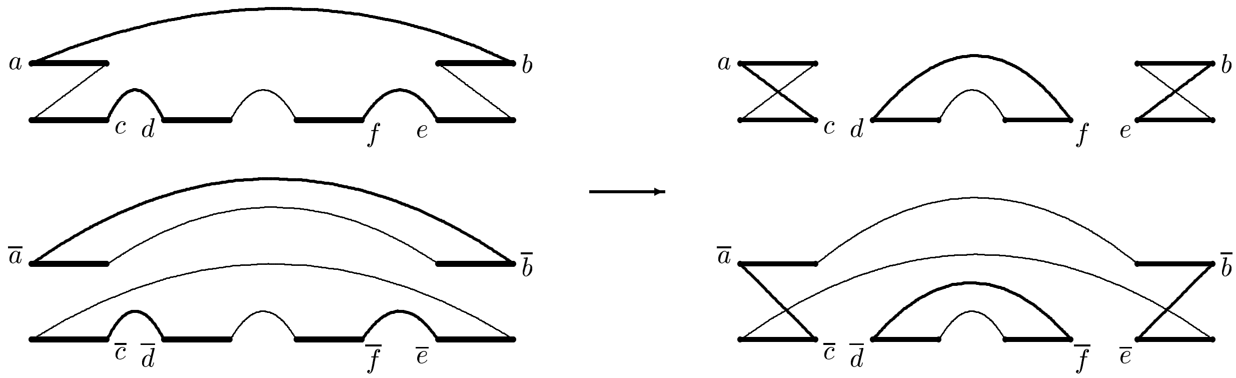


Figure 4: The graph on the left is a subgraph of $\mathcal{G}(\Gamma)$, and the one on the right is the corresponding subgraph of $\mathcal{G}(\Gamma')$ (see the text for more details). Black edges are represented by horizontal lines, and gray edges by thick lines. The other thin lines represent paths between two vertices. All black edges on the same column belong to the same natural graph. For example, the two black edges linking a and $\overline{a}$ are in the same natural graph. The subgraph of $\mathcal{G}(\Gamma)$ has 3 cycles, whereas the subgraph of $\mathcal{G}(\Gamma')$ has 5 cycles. Moreover, the two sets of gray edges give rise to the same number of repeats.

Corollary 1 $D(G) - |\mathcal{E}| \leq RD(G) \leq D(G)$.

Proof: Deduced from Lemma 1, Theorem 1 and the fact that $RD(G) \geq \Delta(G)$ $\square$

The intuitive signification of the former result is that an ancestral genome can be obtained from G by performing $D(G)$ duplications and no reversals, and that reversals can improve this distance by at most $|\mathcal{E}|$.

5 An algorithm for recovering the ancestor of a semi-ambiguous genome

We will complete the natural graphs of $\mathcal{E} \cup \mathcal{N}$ one at a time. Let $\mathcal{E}_2$ be the set containing all natural graphs of $\mathcal{E}$ of size 2, and let $\mathcal{E}_0 = \mathcal{E} \setminus \mathcal{E}_2$. Notice first that no reversal can reduce the number of repeats of the fragments set corresponding to the black edges of a natural graph of $\mathcal{E}_2$. It also follows from Lemma 4 that no reversal can reduce the number of repeats of a natural graph of $\mathcal{N}$. Therefore, for these graphs, we just construct a set of gray edges corresponding to the set of black edges, that is: for any black edge (x, y), construct the gray edge (x, y). Let F_0 be the fragments set resulting from the set of gray edges obtained after completing, as described above, all natural graphs of $\mathcal{N} \cup \mathcal{E}_2$. As these fragments are substrings of the input genome G, none of them is circular.

We will use the following notation: for any set U of natural graphs, we denote by V_U the set of vertices of all natural graphs of U, and by A_U the set of all black edges of U. For example, $V_{\mathcal{E}}$ will be the set of vertices of $\mathcal{E}$. Also, for any set E, we denote by $|E|$ the cardinality of E.

5.1 An algorithm for completing each natural graph of $\mathcal{E}_0$

The goal is now to complete each natural graph $\mathcal{G}_\alpha$ of $\mathcal{E}_0$ in a way minimizing $\Delta(A_\alpha)$ (see Lemma 3). To do that, whenever an edge (x, y) is constructed, the edge $(\overline{x}, \overline{y})$ is also constructed.

Suppose we have reached a certain step s in the construction, Γ_s is the set of gray edges already constructed at this step, and $\mathcal{G}(\Gamma_s)$ is the obtained graph. Suppose also that the natural graph being considered at this step is $\mathcal{G}_\alpha$, the set of gray edges linking vertices of $\mathcal{G}_\alpha$ already constructed is $\Gamma_{s,\alpha}$, and $\mathcal{G}_\alpha(\Gamma_{s,\alpha})$ is the obtained "partially completed" natural graph. A vertex of V is said **unlinked** if it is not yet linked by a gray edge at the current step of the algorithm.

We denote by $\mathcal{F}$ the fragments set resulting from Γ_s. At the outset, $\mathcal{F}$ is made up of the fragments of F_0, and of the **unitary fragments** which are all $x_i^t x_i^h$, such that x_i^t and x_i^h are in $\mathcal{E}_0$. As the construction proceeds, whenever a gray edge (x, y) is created, the fragment containing x and the one containing y are joined together. A **repeated fragment** is a fragment made up exclusively of duplicated genes, that is a fragment with all vertices in $V_{\mathcal{E}}$. In particular, all unitary fragments are repeated fragments.

Definition 2 *Let V_s be one particular subset of the set of unlinked vertices at step s of the algorithm. The* **border** *of V_s is the set of all vertices x of V_s such that, if x is an endpoint of a repeated fragment $F \in \mathcal{F}$, then the second endpoint of F is not in V_s.*

Let O be the set of unlinked vertices of $V_{\mathcal{E}}$ at step 1 of the algorithm, that is after obtaining the fragments set F_0. Notice that, if a vertex x is in O, then $\overline{x}$ is also in O. Also, as $|A_{\mathcal{E}}|$ is even, and that the number of vertices of $V_{\mathcal{E}}$ outside the border is divisible by 4, $|O|$ should also be divisible by 4. At the end of the algorithm, all vertices of $V_{\mathcal{E}}$ will be scattered in, at least, $|O|/4$ repeats of the final fragments. In order to obtain as few repeats as possible, that is exactly $|O|/4$ repeats, we should be careful, during the construction, not to construct a **bad fragment**, that is a circular fragment with all vertices in $V_{\mathcal{E}}$.

$\mathcal{G}(\Gamma_s)$ is a **bad graph** if there exist a subset U of natural graphs of $\mathcal{E}$ such that the border of $V_{s,U}$ is empty, where $V_{s,U}$ is the set of unlinked vertices of V_U at step s. Otherwise $\mathcal{G}(\Gamma_s)$ is a **good graph**.

Lemma 5 *Any set of gray edges linking the remaining unlinked vertices of a bad graph creates at least one bad fragment.*

Proof: Suppose that U is a subset of $\mathcal{E}$ such that the border of $V_{s,U}$ is empty. Then, there is a set $\mathcal{F}_d$ of repeated fragments such that the set of endpoints of $\mathcal{F}_d$ is exactly $V_{s,U}$. Then, by linking the vertices of $V_{s,U}$ by gray edges, all we can do is to close all the fragments of $\mathcal{F}_d$, that is, create at least one circular fragment $\Box$

The above lemma implies that we have to be careful, during the execution of the algorithm, not to end up with a bad graph. Now suppose that $\mathcal{G}(\Gamma_s)$ is a good graph. Let $x, y, \overline{x}, \overline{y}$ be four unlinked vertices of $\mathcal{G}_\alpha(\Gamma_{s,\alpha})$. The pair of "potential" gray edges $\{(x, y), (\overline{x}, \overline{y})\}$ will be termed **impossible** if, when constructed, it creates, either a bad fragment, or a bad graph, and **possible** otherwise. It is easy to see that a pair of edges $\{(x,y),(\overline{x},\overline{y})\}$ creates a bad fragment if and only if one of the following properties is verified (see Figure 5):

Property I: The vertices $\{x,y\}$ are the endpoints of a repeated fragment of $\mathcal{F}$.
Property II: The pairs of vertices $\{x,\overline{y}\},\{\overline{x},y\}$ are the endpoints of two repeated fragments of $\mathcal{F}$.

Figure 5: Left and right figures respectively represent Properties I and II. Bold lines represent fragments, and thin lines represent the "potential" gray edges $(x,y),(\overline{x},\overline{y})$. In any of these cases the resulting fragment is circular, and thus bad.

Now, let consider a third property of a pair $\{(x,y),(\overline{x},\overline{y})\}$ of potential gray edges:
Property III: x, y are two endpoints of two different fragments F_1, F_2 of $\mathcal{F}$, and neither one of the two other endpoints of F_1, F_2 are in $\mathcal{G}_\alpha$.

Lemma 6 *Suppose that $\mathcal{G}(\Gamma_s)$ is good. Suppose that, at step $s+1$, we construct the two gray edges $(x,y),(\overline{x},\overline{y})$. If these gray edges do not satisfy Property III, then $\mathcal{G}(\Gamma_{s+1})$ is good.*

Proof: Let F_1, F_2 be the two fragments such that x is an endpoint of F_1 and y is an endpoint of F_2. Suppose that x, y do not satisfy Property III, and that $\mathcal{G}(\Gamma_{s+1})$ is bad. Let U be a subset of $\mathcal{E}$ such that the border of $V_{U,s+1}$ is empty.

- Suppose the four endpoints of F_1, F_2 are in $\mathcal{G}_\alpha$. Then, it is easy to see that linking F_1 to F_2 does not modify, neither the border corresponding to $\mathcal{G}_\alpha$, nor that corresponding to U. Thus, the state of U could not have changed between steps s and $s+1$.
- Suppose that three of the endpoints of F_1, F_2 are in $\mathcal{G}_\alpha$. U is bad if and only if $V_{U,s+1}$ contains the fourth endpoint of F_1, F_2 not in $\mathcal{G}_\alpha$, and that the border of $V_{U,s+1}$ is empty. But this would implies that U was also bad at step s, which is a contradiction $\square$

Let x be an unlinked vertex of $\mathcal{G}_\alpha$. x is one of the two endpoints of a path C (made up of a succession of black and gray edges) completely contained in $\mathcal{G}_\alpha$. We denote by x^c the second endpoint of this path.

To simplify the description of the algorithm, we use a particular representation of each natural graph $\mathcal{G}_\alpha$ of $\mathcal{E}_0$. Let V_α be its set of vertices and A_α its set of black edges. Relabeling the vertices in V_α allows us to define a **suitable order** for the edges in A_α (see Figure 6). $A_\alpha = \{e_1, e'_1, \cdots, e_n, e'_n\}$ such that:

- $e_1 = (a_1, b_1)$; $e'_1 = (\overline{a_1}, b_2)$.
- For all i, $1 < i < n$, $e_i = (a_i, \overline{b_{i-1}})$ and $e'_i = (\overline{a_i}, b_{i+1})$.
- $e_n = (a_n, \overline{b_{n-1}})$; $e'_n = (\overline{a_n}, \overline{b_n})$.

The *Algorithm complete-natural-graph* described below is used to complete each natural graph of $\mathcal{E}_0$.

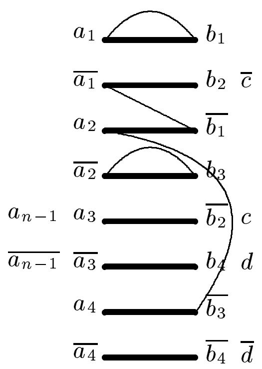


Figure 6: A partially completed natural graph $\mathcal{G}_\alpha(\Gamma_{s,\alpha})$, represented with a suitable order of black edges. At this step, it remains 8 unlinked vertices. It corresponds to the last step of *Algorithm complete-natural-graph*. The vertices corresponding to a_{n-1}, c, d (see the algorithm) are indicated.

Algorithm complete-natural-graph:
For $i = 1$ to $n-2$ do
  Set $c = a_i^c$ and $d = \overline{a_i}^c$;
  If $p_{i,1} = \{(a_i, c), (\overline{a_i}, \overline{c})\}$ is possible
    Construct the gray edges of $p_{i,1}$;
  Otherwise
    Construct the gray edges of $p_{i,2} = \{(\overline{a_i}, d), (a_i, \overline{d})\}$;
Set $c = a_{n-1}^c$ and $d = \overline{a_{n-1}}^c$;
If $p_{n-1,1} = \{(a_{n-1}, c), (\overline{a_{n-1}}, \overline{c})\}$ and $p_{n,1} = \{(a_n, d), (\overline{a_n}, \overline{d})\}$ are possible
  Construct the gray edges of $p_{n-1,1}, p_{n,1}$;
Otherwise
  Construct the gray edges of $p_{n-1,2} = \{(\overline{a_{n-1}}, d), (a_{n-1}, \overline{d})\}$ and $p_{n,2} = \{(a_n, \overline{c}), (\overline{a_n}, c)\}$;

Lemma 7 *At each step, the algorithm constructs possible pairs of gray edges.*

Proof: Before completing any of the natural graphs of $\mathcal{E}_0$, $\mathcal{G}(\Gamma_s)$ is clearly good. Suppose that we have reached a good graph $\mathcal{G}(\Gamma_s)$ with a certain number of completed natural graphs, that we are completing the natural graph $\mathcal{G}_\alpha$, and that the current vertices to be considered are $a_i, \overline{a_i}$. Suppose first that $i \leq n-2$. It is easy to see, from the construction, that $a_i^c \neq \overline{a_i}^c$, and thus, the two pairs of gray edges $p_{i,1}$ and $p_{i,2}$ are different.

Now, suppose that $p_{i,1}$ is impossible. We want to prove that $p_{i,2}$ is possible. Suppose $p_{i,1}$ verifies Property I. That means that a_i, c are the endpoints of the same fragment F. Therefore, $a_i, \overline{d}$ can not be the endpoints of the same fragment, which means that $p_{i,1}$ does not verify Property I. a_i, d are not the endpoints of the same fragment neither, which means that $p_{i,1}$ does not verify Property II. Now, as $a_i, \overline{d}$ are two endpoints of two fragments, where one of them is F and have both endpoints in $\mathcal{G}_\alpha$, $p_{I,2}$ does not verify Property III neither.

We prove similarly that, if $p_{i,1}$ verifies Property II, then $p_{i,1}$ can not verify neither one of the three properties.

Suppose now that $p_{i,1}$ creates a bad graph. That means that there exist a subset U of $\mathcal{E}$ such that the border of $V_{U,s}$ is $B(U,s) = \{a_i, \overline{a_i}, c, \overline{c}\}$. Also, $p_{i,1}$ should satisfy Property III, that is, a_i, c should belong to two different fragments with the two other endpoints not in $\mathcal{G}_\alpha$. Then, clearly $p_{i,2}$ can not satisfy Property I or Property II. Suppose that it satisfies Property III. That means that there exist a subset U' of $\mathcal{E}$ such that the border of $V_{U',s}$ is $B(U', s) = \{a_i, \overline{a_i}, d, \overline{d}\}$. Therefore, the border of $U \cup U'$ is restricted to $\{a_i, \overline{a_i}\}$, and is of size 2. But this is impossible as the number of vertices of $U \cup U'$ remaining unlinked should be divisible by 4.

To finish the proof, we have to show that, if $p_{n-1,1}$ and $p_{n,1}$ are impossible, then $p_{n-1,2}, p_{n,2}$ are possible. To do so, we need to consider more cases than above. Indeed, $p_{n-1,1}$ and $p_{n,1}$ can be impossible due to different reasons: Property I or II can be verified for $p_{n-1,1}$ or $p_{n,1}$; the four gray edges of $p_{n-1,1}$, $p_{n,1}$ can form a single circular fragment; $p_{n-1,1}$, $p_{n,1}$ can give rise to a bad graph. We prove with the same arguments than before, that, in any of these cases, $p_{n-1,2}, p_{n,2}$ are possible $\square$

Now, let *Algorithm complete-graph* be the general algorithm consisting in completing each natural graph of $\mathcal{N} \cup \mathcal{E}_2$ as described above, that is by constructing a set of gray edges identical to the set of black edges, and then in applying *Algorithm complete-natural-graph* to each natural graph of $\mathcal{E}_0$ (see Figure 7 for an example). The set Γ of gray edges resulting from this algorithm represents a set $\mathcal{F} = \{F_1, \cdots F_n\}$ of fragments, possibly circular, of an ancestral genome of G. If this set is reduced to one circular fragment, then this fragment represents an ancestral genome. Otherwise, this fragments set should be modified.

We denote by $C_{\mathcal{N}}$ the set of all duplicated genes x such that both x_1^t and x_1^h (or similarly x_2^t, x_2^h) are in $V_{\mathcal{N}}$. We denote by $B_{\mathcal{E}}$ the set of all duplicated genes x such that exactly one of the two vertices x_1^t, x_1^h is in $V_{\mathcal{E}}$. For example, the sets corresponding to the genome represented by the natural graphs of Figure 3 (and Figure 7) are $C_{\mathcal{N}} = \{h\}$ and $B_{\mathcal{E}} = \{a, b, d, f\}$. It is easy to see that $|B_{\mathcal{E}}|$ is even.

Lemma 8 *Let $\mathcal{G}(\Gamma)$ be the completed graph obtained by* Algorithm complete-graph *and $\mathcal{F}$ be the fragments set resulting from Γ. Then, $\mathcal{G}(\Gamma)$ contains $c(\Gamma) = |A| - \dfrac{|A_{\mathcal{E}}|}{2} + |\mathcal{E}|$ cycles, and $\mathcal{F}$ contains $D(\Gamma) = |C_{\mathcal{N}}| + \dfrac{|B_{\mathcal{E}}|}{2}$ repeats. Moreover: $\Delta(A, \Gamma) = D(G) - |\mathcal{E}|$.*

Proof: First, the set of gray edges exactly corresponds to the set of black edges for natural graphs of $\mathcal{N}$, the number of cycles constructed in these natural graphs is equal to $|A_{\mathcal{N}}|$. Let $\mathcal{G}'$ be a natural graph of $\mathcal{E}$ containing $2n$ edges. Then, each time a vertex $x \in \mathcal{G}'$ is considered, the algorithm constructs, either $(x, x^c), (\overline{x}, \overline{x^c})$, or $(\overline{x}, \overline{x}^c), (x, \overline{\overline{x}^c})$, and then, at least one path is closed to form a cycle. As $2n$ gray edges are required to complete $\mathcal{G}'$ and that two gray edges are constructed at each step, n steps are necessary to complete $\mathcal{G}'$. Moreover, at the last step, the two last gray edges close two paths (form two cycles). Thus, at least $n+1$ cycles are formed in $\mathcal{G}'$. But, as our construction give rise to a set of repeated fragments, according to Lemma 6.3 in [EMS01] , the maximal number of cycles of a completed graph of $\mathcal{G}'$ is $n+1$. Therefore, the number of cycles of a completed graph of $\mathcal{G}'$ is $n+1$. Thus, the total number of cycles of all completed natural graphs of $\mathcal{E}$ is $\frac{|A_{\mathcal{E}}|}{2} + |\mathcal{E}|$. We deduce that the completed graph constructed by *Algorithm complete-graph* contains $c(D) = |A_{\mathcal{N}}| + \dfrac{|A_{\mathcal{E}}|}{2} + |\mathcal{E}| = |A| - \dfrac{|A_{\mathcal{E}}|}{2} + |\mathcal{E}|$ cycles.

As the set $\Gamma_{\mathcal{N}}$ of gray edges constructed in the natural graphs of $\mathcal{N}$ exactly corresponds to the set of black edges, the fragments set $\mathcal{F}(\Gamma_{\mathcal{N}})$ of $\Gamma_{\mathcal{N}}$ is exactly the fragments set $\mathcal{F}(A_{\mathcal{N}})$ determined by the black edges $A_{\mathcal{N}}$. As the number of repeats of $\mathcal{F}(A_{\mathcal{N}})$ is $|C_{\mathcal{N}}|$, the number of repeats of $\mathcal{F}(\Gamma_{\mathcal{N}})$ is also $|C_{\mathcal{N}}|$.

Now, as $2|B_{\mathcal{E}}|$ is the number of elements in O, that each fragment constructed by the algorithm has two endpoints in O, and that only repeated fragments are constructed by the algorithm, the number of repeats resulting from *Algorithm complete-graph* is $|B_{\mathcal{E}}|/2$. Therefore, the total number of repeats of $\mathcal{I}$ is $|C_{\mathcal{N}}| + |B_{\mathcal{E}}|/2$.

From the above results we have $\Delta(A, \Gamma) = |A| - |A| + \frac{|A_{\mathcal{E}}|}{2} - |\mathcal{E}| + |C_{\mathcal{N}}| + \frac{|B_{\mathcal{E}}|}{2} = \frac{|A_{\mathcal{E}}|}{2} + |C_{\mathcal{N}}| + \frac{|B_{\mathcal{E}}|}{2} - |\mathcal{E}|(*)$.

Now, as $D(G)$ is the number of genes of $\mathcal{B}_d$, it is the number of genes exclusively contained in the natural graphs of $\mathcal{N}$, that is $|C_{\mathcal{N}}|$, plus the number of genes exclusively contained in the natural graphs of $\mathcal{E}$, that is $\frac{|A_{\mathcal{E}}|}{2} - \frac{|B_{\mathcal{E}}|}{2}$, plus the number of genes with one end in $\mathcal{N}$, and one end in $\mathcal{E}$, that is $|B_{\mathcal{E}}|$. Therefore $D(G) = |C_{\mathcal{N}}| + \frac{|A_{\mathcal{E}}|}{2} - \frac{|B_{\mathcal{E}}|}{2} + |B_{\mathcal{E}}| = |C_{\mathcal{N}}| + \frac{|A_{\mathcal{E}}|}{2} + \frac{|B_{\mathcal{E}}|}{2}$. Thus, from $(*)$ we have $\Delta(A,\Gamma) = D(G) - |\mathcal{E}|$ $\square$.

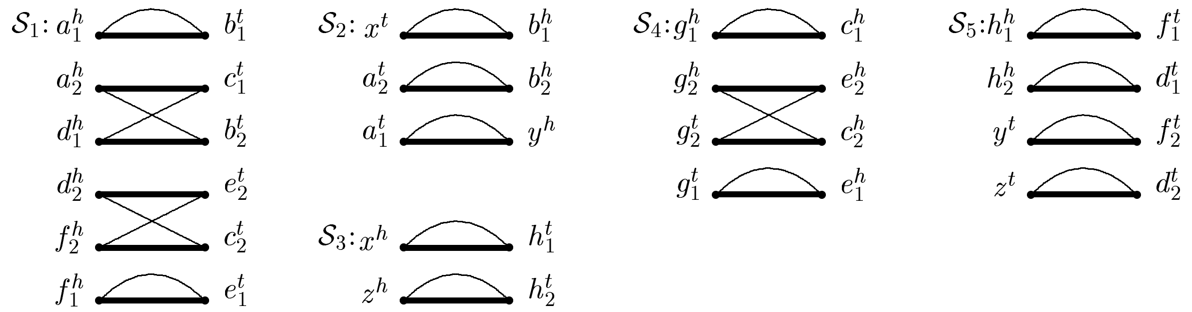


Figure 7: The completed graph obtained by applying *Algorithm complete-graph* to the partial graph of Figure 3. The resulting fragments set contains two circular fragments: $\mathcal{F} = \{+a_2\ +b_2\ ,\ +a_1\ +b_1\ +x\ +h_1\ +f_1\ +e_1\ +g_1\ -c_1\ -d_1\ -h_2\ -z\ +d_2\ +c_2\ -g_2\ -e_2\ -f_2\ +y\}$. Here, $c(\Gamma) = 16$ and $D(\mathcal{F}) = 3$.

Theorem 2 *If* Algorithm complete-graph *gives rise to a unique circular fragment I, then this fragment is an ancestral genome of G. Moreover, if the completed graph $\mathcal{G}(\Gamma)$ does not contain any hurdle, then the reversal/duplication distance of G is:*

$$RD(G) = RD(G,I) = D(G) - |\mathcal{E}|.$$

Proof: The final obtained fragment I is circular and contains exactly the same genes than G in the same number of copies. Therefore it is a potential ancestral genome of G.

From Lemma 8, we have $\Delta(A,\Gamma) = D(G) - |\mathcal{E}|$. Now, from HP formulae: $RD(G,I) = R(G,I) + D(I) = |A| - c(\Gamma) + h(\Gamma) + f(\Gamma) + D(I) = \Delta(A,\Gamma) + h(\Gamma) + f(\Gamma)$. Therefore, if $\mathcal{G}(\Gamma)$ does not contain any hurdle, $h(\Gamma) = f(\Gamma) = 0$ and $RD(G,I) = \Delta(A,\Gamma)$ $\square$

5.2 Recovering an ancestral genome

Let $\mathcal{G}(\Gamma)$ be the completed graph obtained by *Algorithm complete-graph*, and $\mathcal{F}$ be the fragments set corresponding to Γ. If $\mathcal{F}$ contains more than one fragment, then we apply *Algorithm correction* described below. An **adjacency** will refer to a pair of vertices $\{a,b\}$, not corresponding to the same gene (i.e. not x^t, x^h for the same x), that are linked by a gray edge. Two adjacent vertices belong necessarily to the same natural graph.

Algorithm correction:
(1) While $\mathcal{F}$ is not restricted to one fragment do
(2) Choose two arbitrary fragments F_1, F_2 in $\mathcal{F}$;
(3) Choose an adjacency $\{a,b\}$ in F_1 with b not in $V_{\mathcal{E}}$;
(4) Choose an adjacency $\{c,d\}$ in F_2 with d not in $V_{\mathcal{E}}$;
(5) Replace the two gray edges $(a,b),(c,d)$ by:
(6) $(a,c),(b,d)$ if one of these edges is oriented;
(7) $(a,d),(b,c)$ otherwise;
(8) Remove from $\mathcal{F}$ the two fragments F_1, F_2, and add the new obtained fragment.

At lines (3) and (4), an adjacency verifying the given property exists because, from the construction, $\mathcal{F}$ can not contain a circular fragment with all vertices in $V_{\mathcal{E}}$. Let $\mathcal{G}(\Gamma_f)$ be the final completed graph obtained by applying *Algorithm correction* to $\mathcal{G}(\Gamma)$.

Lemma 9 *$\mathcal{G}(\Gamma_f)$ gives rise to an ancestral genome I of G. Moreover: $c(\Gamma)-|\mathcal{F}|+1 \leq c(\Gamma_f) \leq c(\Gamma)$, $D(I) \leq D(\mathcal{F})$, and thus:*

$$\Delta(G,I) \leq D(G) - |\mathcal{E}| + |\mathcal{F}| - 1 \leq D(G) - |\mathcal{E}| + |B_{\mathcal{E}}| - 1$$

Proof: Notice that $\Delta(G,I)$ is just $\Delta(A,\Gamma_f)$. At each step, *Algorithm correction* choose two circular fragments and merge them into one circular fragment. Thus, the final obtained genome I is circular and contains exactly the same genes than G in the same number of copies. Therefore it is a potential ancestral genome of G.

Let F_1, F_2 be the two fragments being merged at step s of the algorithm, and let $(a,b) \in F_1$, $(c,d) \in F_2$ be the two considered gray edges. a,b,c,d belong to at most two cycles. If they belong to one cycle, then replacing these two edges by $(a,c),(b,d)$ $(a,d),(b,c)$ creates one more cycle. Otherwise, the new constructed edges merge the two cycles into one. Therefore, at each step, the algorithm merges two fragments, and the number of cycles is reduced by at most one. Thus, $c(\Gamma) - (\lfloor|\mathcal{F}| - 1)\rfloor \leq c(\Gamma_f) \leq c(\Gamma)$.

Now, as the edges to be replaced are chosen not to link two vertices of $\mathcal{E}$, we are sure not to destroy a repeated fragment, and thus, $D(\mathcal{F}) \leq D(I)$.

The inequality $\Delta(A,\Gamma_f) \leq D(G) - |\mathcal{E}| + |\mathcal{F}| - 1$ is a direct consequence of the above results and Lemma 8. Now, to see that $\Delta(A,\Gamma_f) \leq D(G) - |\mathcal{E}| + |B_{\mathcal{E}}| - 1$, we should prove that $|\mathcal{F}| \leq |B_{\mathcal{E}}|$. This is deduced from the fact that the algorithm never constructs a bad fragment, that is a circular fragment containing no vertex of O. Therefore, each fragment of $\mathcal{F}$ should contain at least two vertices of O, thus $\mathcal{F}$ contains at most $|O|/2$ fragments. The final result is deduced from the fact that $|O|/2 = |B_{\mathcal{E}}|$ $\square$

The following corollary gives tight bounds for the reversal/duplication distance.

Corollary 2 *If the final graph $\mathcal{G}(\Gamma_f)$ does not contain any hurdle, then:*

$$D(G) - |\mathcal{E}| \leq RD(G) \leq \min\{D(G), D(G) - |\mathcal{E}| + |B_{\mathcal{E}}| - 1\}$$

Proof: The first inequality comes from corollary 1, and the secondinequality is deduced from the fact that $RD(G) \leq RD(G,I) \leq D(G) - |\mathcal{E}| + |B_{\mathcal{E}}| - 1$ and Lemma 9 $\square$

In order to make *Algorithm correction* more efficient, we can merge fragments in a way decreasing the number of repeats. Indeed, suppose that $\mathcal{F}$ contains two repeats A, B, and that the four substrings of form $A, B, \overline{A}, \overline{B}$ are localized in at least three fragments of $\mathcal{F}$. Then these fragments can be merged, in two steps, into one fragment containing the repeat AB (see Figure 8). If we can merge the fragments of $\mathcal{F}$ in a way guaranteeing that, at each step, at least three fragments containing four substrings of form $A, B, \overline{A}, \overline{B}$ remain, then, at the end of the algorithm, the ancestral genome I obtained is such that $\Delta(G,I) = D(G) - |\mathcal{E}|$.

The steps (5), (6), (7) of *Algorithm correction* ensures that the components obtained by merging fragments are good.

Hereafter, *Algorithm complete-graph* will refer to the algorithm obtained by incorporating *Algorithm correction* to the end of the initial one.

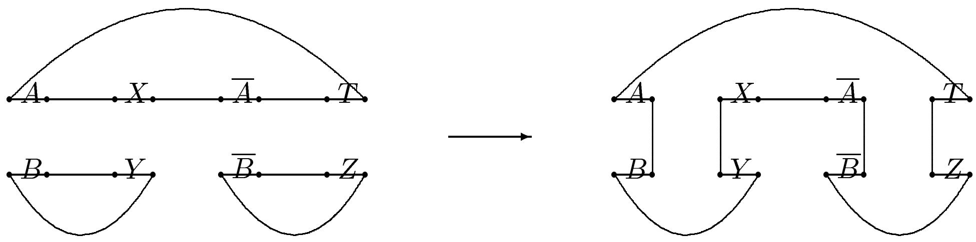


Figure 8: This example shows that merging fragments can reduce the number of repeats. The left figure represents three fragments of $\mathcal{F}$. Letters and lines represent substrings of these fragments. A, B are two repeats. After two merges, a single circular fragment (right figure) is obtained, with the two repeats A, B merged into one $A - B$.

5.3 Complexity of the algorithm

For any natural graph of $\mathcal{N}$, the algorithm constructs a set of gray edges that is a copy of the set of black edges. Therefore, the time required to complete all these natural graphs is constant.

Let $\mathcal{G}_\alpha$ be a natural graph of $\mathcal{E}$ of size $2n$. *Algorithm complete-natural-graph* proceeds in n steps. At each step, the only non-constant instruction is to check whether or not a pair of gray edges creates a bad graph. To do that, we can proceed in the following way: Check if Property III is satisfied for the considered pair of gray edges. This can be done in constant time. If not, the graph is good. Otherwise, it is easy to see that we have to test at most $|\mathcal{E}|$ subsets of natural graphs. Therefore, the time required by *Algorithm complete-natural-graph* to complete the natural graph $\mathcal{G}_\alpha$ is in $O(n|\mathcal{E}|)$. Thus, the total time required to complete all natural graphs of $\mathcal{E}$ is in $O(|\mathcal{E}||A_\mathcal{E}|)$.

Now, as the maximal number of circular fragments resulting from the completed graph is $|B_\mathcal{E}|$, the complexity of *Algorithm correction* is clearly $O(|B_\mathcal{E}|)$.

Therefore, the time complexity of the whole algorithm is $O(|\mathcal{E}||A_\mathcal{E}| + |B_\mathcal{E}|)$.

5.4 Bad components

Results of Theorem 2 and Corollary 2 are conditional on the fact that the obtained completed graph does not contain any hurdle. As the probability for a component to be good is higher than its probability to be bad, this assumption is reasonable, especially that *Algorithm correction* is designed to avoid creating bad components when merging circular fragments.

For the genome duplication model, we developed in [EMS01, EM00a, EMS99] rigorous methods for avoiding as many bad components as possible. We have shown that, if the modern genome G does not contain a particular kind of substrings, called *local subpermutations*, then the completed graph can be corrected in a way eliminating all bad components. Otherwise, the number of hurdles is perfectly determined by the modern genome G. The same kind of results hold for our problem involving local duplications. More precisely, in most cases (the genome does not contain local subpermutations), the completed graph $\mathcal{G}(\Gamma)$ obtained by applying *Algorithm complete-graph* can be corrected to remove all bad components, without modifying the value of $\Delta(A, \Gamma)$.

6 Recovering the ancestor of an ambiguous genome

So far, we have shown how to reconstruct an ancestral genome of a semi-ambiguous modern genome G. Our aim now is to face the general problem of an ambiguous genome, that is a genome with gene families of any size. The key idea is to subdivide this problem into a series of sub-problems involving semi-ambiguous genomes, by pairing duplicated genes in a certain way. One possibility is to try all possible pairings, and to choose the one that gives rise to the minimal number of reversals/duplications. Such a method is, of course, highly exponential, and does not take into account any meaningful biological information. This could be avoided if one have a preliminary information about the evolutionary relationship between all genes of a gene family, summarized by a gene tree. Such a gene tree can be constructed from an alignment of nucleotides or amino acid sequences of paralogous and orthologous genes.

Example 1 *Let G be a genome containing 8 copies of a gene a. Figure 9 represents a possible gene tree for this gene family. Leaves represent the different copies of a, and internal nodes represent gene duplication events.*

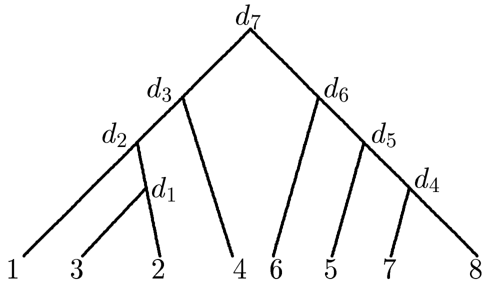


Figure 9: A gene tree of a gene family of size 8. Leaves represent gene copies, and internal nodes represent gene duplication events.

For our purpose, we need to know, not only the tree topology, but also the approximate time of divergence events. In that case, we can subdivide the set of internal nodes into subsets corresponding to the same historical time t. For example, suppose that, in the above example, the tree branches length is proportional to the time passed since divergence. Then, we can notice that gene duplications d_1, d_4 occurred at the same historical time. Therefore, they can be the result of a unique duplication transposition event. The same observation holds for d_2, d_5 and d_3, d_6. The internal nodes are, therefore, subdivided into the following subsets: $\{d_1, d_4\}, \{d_2, d_5\}, \{d_3, d_6\}, \{d_7\}$.

Let G be an ambiguous genome on a set $\mathcal{B}$ of size b, and suppose we have b gene trees summarizing the results of independent phylogenetic analysis within each of the b multigene families. Let $t_1, \cdots, t_p$ be the sequence of all times corresponding to all nodes of all these gene trees, in increasing order. Our general algorithm used to reconstruct a non-ambiguous genome from G follows p steps. At each step we apply *Algorithm complete-graph* to a particular semi-ambiguous genome G_i. Let $\mathcal{F}_a$ be a gene tree containing n leaves labeled by numbers from 1 to n. During the execution of the algorithm, the nodes of $\mathcal{F}_a$ are successively labeled by an integer in $[1, n]$. At the beginning of the algorithm, the only labeled nodes are the leaves. We denote by $\lambda(u)$ the label of a node u. Each step i of the algorithm is subdivided into two procedures:

1. **Gene pairing:** Consider the set S_i of all pairs of nodes (u, v) of all gene trees, such that u, v have the same father f and the time of f is t_i. For each pair $(u, v) \in S_i$, set $\overline{\lambda(u)} = \lambda(v)$. In other words, $\lambda(u), \lambda(v)$ are considered as duplicated genes. All the other genes are considered as singletons.

2. **Algorithm complete-graph:** Apply *Algorithm complete-graph* to the semi-ambiguous genome G' obtained from (1). Let H be the resulting non-ambiguous genome. For any pair $(u, v) \in S_i$, H contains exactly one of the two genes $\lambda(u), \lambda(v)$. The father of u, v is labeled by this gene copy.

$$1,2,3,4,5,6,7,8 \xrightarrow{(1)} 1,\{2,3\},4,5,6,\{7,8\} \xrightarrow{(2)} 1,2,4,5,6,7 \xrightarrow{(1)} \{1,2\},4,6,\{5,7\} \xrightarrow{(2)} 1,4,6,5$$

$$\xrightarrow{(1)} \{1,4\},\{6,5\} \xrightarrow{(2)} 1,5 \xrightarrow{(1)} \{1,5\} \xrightarrow{(2)} 1$$

Figure 10: Possible steps in proceeding the gene family represented by the tree of Example 1. (1): Gene pairing; (2): Algorithm complete-graph.

7 Recovering the ancestral nodes of a species tree, using gene trees

The approach described in the last section can be used in a fundamental phylogenetic problem: given a species tree $\mathcal{T}$, as well as N ambiguous genomes corresponding to these species and F gene trees summarizing the results of independent phylogenetic analysis within each of the F multigene families represented in these genomes, how to reconstruct gene orders at the ancestral nodes of $\mathcal{T}$? A method to solve this problem has been presented in [SEM00]. It integrates three approaches to genomic evolution: reconciliation, exemplar analysis and breakpoint-based phylogeny. This method is summarized in the next paragraph.

7.1 The general method

The first step of the method is to assign the right number of copies of each gene at each internal node of $\mathcal{T}$. The reconciliation of a gene tree with a species tree is designed to explain the non-congruence between these two trees by duplication events, localized in the gene tree, that have affected some lineages in the tree but not others [PC97, PC00]. In [SEM00], we have shown how to use this approach to find the number of gene copies at each ancestral node of a species tree, as well as the historical relationship among these copies.

After assigning the right number of copies of each gene at each internal node of $\mathcal{T}$, the right gene orders should be found. Starting with an initial assignment of genomes to all the internal nodes, recalculation of the internal nodes is carried out one by one, each time using the most recently calculated versions of the neighboring internal nodes. Iteration continues until no improvement can be made at any node. At each step, gene order at each internal node X is obtained by using the median and the exemplar approach, described below.

Given three non-ambiguous genomes A, B and C, the **median problem** is to find a genome X that minimizes $d(A,X) + d(B,X) + d(C,X)$ for a distance d. Efficient heuristics exist for the breakpoint distance, even if the genomes do not have the same set of genes [SB97, SBD$^+$00]. However, these heuristics can not be applied to genomes containing several copies of the same gene. Thus, ambiguous genomes should be converted to non-ambiguous genomes, before applying the median approach. To do so, the exemplar approach is considered [San99]. It is used to compare each of the pairs (A,X), (B,X) and (C,X). The central idea, based on a model of gene copy movement, is the deletion of all but one member of each gene family -its exemplar- in each of the two genomes being compared, so as to minimize some rearrangement distance between the two reduced genomes thus derived. The alternating application of exemplar and median analysis is shown in Figure 11.

7.2 Using the ambiguous genome ancestor approach

We show here how to replace, in the general method described above, the exemplar approach by our new approach of recovering a non-ambiguous genome that minimizes the number of reversals and duplication transpositions.

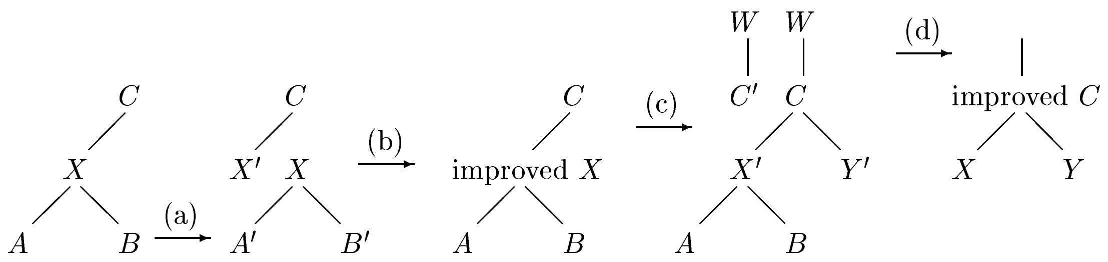


Figure 11: Alternating application of exemplar and median analyses. (a) and (c): Exemplar extraction. (b) and (d): calculation of the median.

The reconciliation approach described in [SEM00] gives rise to groupings at each node of the species tree. Each grouping at a node u represents one gene copy whose descendants are just the copies listed in the grouping. Figure 12.a is an example of groupings obtained for three adjacent nodes of a species tree. The genome A contains 7 copies of a gene a. As copies 1,2,3 are grouped in ρ, and 5,6 are grouped in τ, the sets $\{1,2,3\},\{4\},\{5,6\},\{7\}$ are considered separately, and the exemplar analysis is used to choose an exemplar gene from each set.

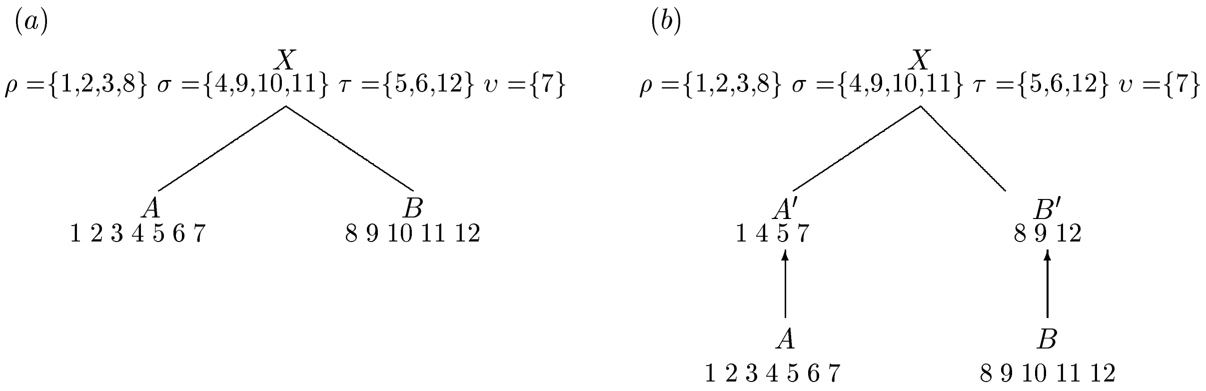


Figure 12: Subtree consisting of genomes A, B, and their common immediate ancestor X. Each grouping represents a gene copy whose descendants in A and B are just the copies listed between the braces.

Now let consider our ambiguous genome approach. The reconciliation analysis is applied as described before. Let X, A be two gene orders respectively assigned to two nodes u, v, where u is the father of v. Then, each grouping of X is considered as a gene family on its own, and the method described in section 6 is applied to reconstruct a non-ambiguous genome A'. Now, as A' contain just one member of each considered gene family, the median approach can be applied (see Figure 12.b). To incorporate this ambiguous genome approach to the general method described in the last section, it suffices to replace the exemplar extraction by this new approach in Figure 11.

7.3 Advantages of our approach versus the exemplar approach

The implicit evolutionary model considered for the exemplar approach is that the two genomes G, H being compared (containing the same set of genes, but in a different number of copies) evolved from a common ancestor F containing exactly one copy of each gene, through a series of independent gene duplications and reversals. The central idea of the exemplar approach is just to keep members of each gene family (in G and H), its true exemplars, which best reflect the original position of the ancestral gene in the common ancestor F. The deletion procedure used does not rely on a real mutational process. In contrast, our approach based on duplication transpositions and reversals, not only find an ancestral non-ambiguous genome, but also produces a possible sequence of mutations. Therefore, the pertinence of the obtained genome is better justified with our approach.

Another advantage of our approach is that it is designed to find the ancestor of a modern genome, and therefore it is directly applicable to the problem of recovering the ancestral nodes of a species tree. In contrast, the exemplar analysis is used to compare two genomes. In the context of recovering ancestral nodes of a species tree, at each step i the exemplar approach should be applied to a "good" node (a leaf, or an internal node already recalculated) and a "bad" node (an initial assignment, or a node not yet recalculated at step i). In contrast, our new approach is applied to only good nodes.

8 Conclusion

We have developed one of the first genomic rearrangement method that can be applied to genomes containing several copies of the same gene. As homologous genes are commonly encountered in real data, a genome rearrangement approach that takes gene families into account is essential. We considered a general model of duplication, including the duplication of single genes or entire chromosomal segments, in direct or indirect order, in tandem or not.

As a first study, we restricted ourselves to circular genomes, and to an evolutionary model involving segments reversals and duplications. We have developed an algorithm for recovering an ancestral genome I of a modern genome G. This algorithm is exact, i.e. it gives rise to a genome H minimizing the distance between G and any potential ancestor of G, when certain conditions are verified. Otherwise, we provide bounds that show that the obtained genome is not too far from an optimal one. To find an actual sequence of reversals and duplications that transform H into G, it suffices to apply the HP algorithm to the completed graph obtained by our algorithm.

Generalizations of our approach, involving gene losses, transpositions, or translocations in the case of multichromosomal genomes, can be considered.

References

[Ber01] A. Bergeron. A very elementary presentation of the Hannenhalli-Pevzner theory. In *12th Annual Symposium on Combinatorial Pattern Matching*, Lecture Notes in Computer Science. Springer Verlag, 2001.

[BKS99] M. Blanchette, T. Kunisawa, and D. Sankoff. Gene order breakpoint evidence in animal mitochondrial phylogeny. *Journal of Molecular Evolution*, 49:193-203, 1999.

[BP98] Vineet Bafna and Pavel A. Pevzner. Sorting by transpositions. *SIAM Journal on Discrete Mathematics*, 11(2):224–240, 1998.

[EAR99] E. E. Eichler, N. Archidiacono, and M. Rocchi. CAGGG repeats and the pericentromeric duplication of the hominoid genome. *Genome Research*, 9:1048–1058, 1999.

[EM00a] N. El-Mabrouk. Recovery of ancestral tetraploids. In David Sankoff and Joseph H. Nadeau, editors, *Comparative Genomics: Gene Order Dynamics, Map Alignment and the Evolution of Gene Families*, volume 1 of *Series in Computational Biology*, pages 465- 477, Dordrecht, NL, 2000. Kluwer Academic Press.

[EM00b] N. El-Mabrouk. Sorting signed permutations by reversals and insertions/deletions of contiguous segments. *Journal of Discrete Algorithms*, 1(1):105-122, 2000.

[EMBS99] N. El-Mabrouk, D. Bryant, and D. Sankoff. Reconstructing the pre-doubling genome. In *Proceedings of the third annual international conference on computational molecular biology (RECOMB'99)*, pages 154–163. ACM press, April 1999.

[EMS99] N. El-Mabrouk and D. Sankoff. On the reconstruction of ancient doubled circular genomes using minimum reversals. In S. Miyano K. Asai and T. Takagi, editors, *Genome Informatics*, volume 10, pages 83-93, Tokyo, 1999. Universal Academy Press.

[EMS01] N. El-Mabrouk and D. Sankoff. The reconstruction of doubled genomes. submitted to SIAM, Journal on Computing, 2001.

[Han95] S. Hannenhalli. Polynomial-time algorithm for computing translocation distance between genomes. In Z. Galil and E. Ukkonen, editors, *Sixth Annual Symposium on Combinatorial Pattern Matching*, volume 937 of *Lecture Notes in Computer Science*, pages 162–176, Berlin, 1995. Springer.

[HP95a] S. Hannenhalli and P.A. Pevzner. Transforming cabbage into turnip. (polynomial algorithm for sorting signed permutations by reversals). In *Proceedings of the 27th Annual ACM-SIAM Symposium on the Theory of Computing*, pages 178–189, 1995.

[HP95b] S. Hannenhalli and P.A. Pevzner. Transforming men into mice. In *Proceedings of the IEEE 36th Annual Symposium on Foundations of Computer Science*, pages 581–592, 1995.

[KR95] J. D. Kececioglu and R. Ravi. Of mice and men: algorithms for evolutionary distance between genomes with translocations. In *Proceedings of Sixth ACM-SIAM Symposium on Discrete Algorithms*, pages 604–613, 1995.

[KST97] H. Kaplan, R. Shamir, and R.E. Tarjan. Faster and simpler algorithm for sorting signed permutations by reversals. In *Proceedings of the 8th Annual ACM-SIAM Symposium on Discrete Algorithms*, 1997.

[MS98] R. Mazzarella and D. Schlessinger. Pathological consequences of sequence duplications in the human genome. *Genome Research*, 8:1007–1021, 1998.

[OE00] Christine O'Keefe and Evan Eichler. The pathological consequences and evolutionary implications of recent human genomic duplications. In David Sankoff and Joseph H. Nadeau, editors, *Comparative Genomics: Gene Order Dynamics, Map Alignment and the Evolution of Gene Families*, volume 1 of *Series in Computational Biology*, pages 29 - 46, Dordrecht, NL, 2000. Kluwer Academic Press.

[PC97] R.D.M Page and M.A. Charleston. Reconciled trees and incongruent gene and species trees. *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, 37:57–70, 1997.

[PC00] Roderic D. M. Page and James A. Cotton. GENETREE: A tool for exploring gene family evolution. In David Sankoff and Joseph H. Nadeau, editors, *Comparative Genomics: Gene Order Dynamics, Map Alignment and the Evolution of Gene Families*, volume 1 of *Series in Computational Biology*, Dordrecht, NL, 2000. Kluwer Academic Press.

[San99] David Sankoff. Genome rearrangements with gene families. *Bioinformatics*, 15:909–917, 1999.

[SB97] David Sankoff and Mathieu Blanchette. The median problem for breakpoints in comparative genomics. In T. Jiang and D.T. Lee, editors, *Computing and Combinatorics, Proceeedings of COCOON '97*, number 1276 in Lecture Notes in Computer Science, pages 251–263, Berlin, 1997. Springer.

[SBD+00] D. Sankoff, D. Bryant, M. Deneault, B.F. Lang, and G. Burger. Early eukaryote evolution based on mitochondrial gene order breakpoints. In *Proceedings of the Fourth Annual International Conference on Computational Molecular Biology (RECOMB 2000)*, New York, 2000. ACM.

[SEM00] D. Sankoff and N. El-Mabrouk. Duplication, rearrangement and reconciliation. In D.Sankoff and J.H. Nadeau, editors, *Comparative Genomics: Empirical and Analytical Approaches to Gene Order Dynamics, Map alignment and the Evolution of Gene Families (DCAF)*, volume 1 of *Series in Computational Biology*, pages 537- 550, Dordrecht, NL, september 2000. Kluwer Academic Press.

[WDM98] M. E. Walter, Z. Dias, and J. Meidanis. Reversal and transposition distance of linear chromosomes. In *String Processing and Information Retrieval: A South American Symposium (SPIRE '98)*, 1998.